

中国科学技术大学

University of Science and Technology of China

博士学位论文



论文题目 复杂动态行人场景下的机器人导航

作者姓名 陈广大

学科专业 计算机应用技术

导师姓名 陈小平 教授

完成时间 二〇二一年五月

中国科学技术大学

博士学位论文



复杂动态行人场景下的机器人导航

作者姓名：陈广大

学科专业：计算机应用技术

导师姓名：陈小平 教授

完成时间：二〇二一年五月二十七日

University of Science and Technology of China
A dissertation for doctor's degree



**Robot Navigation in Complex and
Dynamic Pedestrian Scenarios**

Author: Guangda Chen

Speciality: Computer Application Technology

Supervisor: Prof. Xiaoping Chen

Finished time: May 27, 2021

中国科学技术大学学位论文原创性声明

本人声明所呈交的学位论文，是本人在导师指导下进行研究工作所取得的成果。除已特别加以标注和致谢的地方外，论文中不包含任何他人已经发表或撰写过的研究成果。与我一同工作的同志对本研究所做的贡献均已在论文中作了明确的说明。

作者签名: 陈广大

签字日期: 2021.5.29

中国科学技术大学学位论文授权使用声明

作为申请学位的条件之一，学位论文著作权拥有者授权中国科学技术大学拥有学位论文的部分使用权，即：学校有权按有关规定向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，可以将学位论文编入《中国学位论文全文数据库》等有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。本人提交的电子文档的内容和纸质论文的内容相一致。

保密的学位论文在解密后也遵守此规定。

公开 保密（ 年）

作者签名: 陈广大

导师签名: 陈小平

签字日期: 2021.5.29

签字日期: 2021.5.29

摘要

机器人导航是移动机器人非常重要的基础能力。随着人工智能、机器人技术和传感器技术的快速发展，越来越多的移动机器人被应用到开放的人群环境中。为了应对如此复杂的动态场景，机器人配备了越来越多的传感器来完成环境感知任务，传感器的精确标定是机器人能够成功导航的前提。基于动作捕获系统的通用批量标定框架可以解决多传感器标定问题，但目前该系统仅限于激光与机器人标定且核心外参标定方法准确度不高。在本研究中，我们基于通用批量标定框架和动作捕获系统提出了可靠精确的 RGB-D 相机标定方法，使通用批量标定框架扩展到彩色或深度相机标定领域。在传感器外参标定中，我们充分利用运动捕捉系统的精准特性来定制手眼标定的全局优化步骤，进一步提高了外参标定的准确性。此外，我们提出了一种非递归的新颖数据获取方法来计算深度图像中每个像素的真实值，并采用一种无模型的单步深度校准方法来获得深度校准模型集的参数。与现有的深度校准技术相比，我们的方法可以同时估计不同测量距离处深度误差的均值和方差。实验结果表明我们的标定方法大大提高了彩色点云的匹配与测量精度。

移动机器人的自主移动导航依靠安全高效的避障算法。近年来的大量研究工作开始探索深度强化学习在动态环境下的机器人导航应用，已经开发了基于原始传感器数据输入或智能体位置信息输入的端到端动态环境导航。在本文中，我们首先提出了一种在无通信环境中基于栅格地图和深度强化学习的异质多机器人避障算法，其中以自我为中心的局部栅格图用于表示机器人周围的环境信息和机器人自身形状大小，环境信息可以通过使用多个传感器或传感器融合轻松地生成。与其他方法相比，基于栅格图的方法对嘈杂的传感器数据具有更强的鲁棒性，不需要机器人的运动状态数据，并且考虑了相关机器人的大小和形状，从而使其更高效且更容易部署到实体机器人上。我们首先使用分布式近端策略优化算法在包含多个移动机器人的定制仿真器中训练神经网络，其中使用针对多种场景的多阶段课程学习策略来提高性能，然后我们将训练后的模型直接部署到现实的机器人上而无需进行繁琐的参数调整。我们在多种仿真和现实场景中评估了该方法。定性和定量实验均表明，我们的方法是有效的，并且在许多指标上都优于现有的传统的和基于深度强化学习的避障方法。

然而，在人群密集的场景中，如果避障算法把行人也视为普通的障碍物或机器人，可能会导致近距离的避障效果，给行人带来不适。同时行人的行走策略与机器人的避障策略一般不同，机器人在人群中的避障不完全符合同策略多机器人避障算法适用条件。因此我们扩展了基于栅格地图的深度强化学习多机器人

摘 要

避障方法，在传感器栅格地图中添加了由多传感器行人感知追踪模块生成的行人地图通道，设计了新的远离行人的回报函数部分和多种行人仿真策略训练环境，实验结果表明我们的方法进一步提升了机器人在不同行人策略环境中的避障成功率。同时在动态行人环境中，机器人还会遇到更严重的定位丢失问题。因此我们提出基于行人感知过滤的动态行人环境定位方法，改善了机器人定位的准确性。然而被动定位方法仍不能完全解决大量动态人腿遮挡问题，因此我们提出了基于二维码激光联合主动重定位方法。当机器人定位误差较大时，机器人自主移动到二维码恢复区校准自身定位信息。实验结果显示我们的重定位方法可以在定位完全丢失时仍能非常准确地重新定位机器人全局位姿。最后，根据上述研究成果和“封闭性准则”，本文提出了试图解决开放人群环境下的机器人长期自主导航的具体部署方案，并成功在成都大熊猫研究繁育基地的博物馆导览机器人上进行了大规模长时间测试。

关键词：机器人导航；传感器标定；多传感器融合；多机器人避障；深度强化学习；栅格地图；行人追踪；行人导航；二维码定位；机器人重定位

ABSTRACT

Navigation is a very important and fundamental ability for mobile robots. With the development of artificial intelligence, robotics, and sensor technology, more and more mobile robots are applied to open and crowd environments. In order to cope with such dynamic and complex scenarios, robots are equipped with more and more sensors to perform complex perception tasks. The accurate calibration of the sensors is a prerequisite for the successful navigation of the robot. The general batch calibration framework based on motion capture system can solve the problem of multi-sensor calibration, but the current system is only limited to laser and robot calibration, and the accuracy of the core extrinsic calibration method is not high. In this dissertation, we propose a reliable and accurate calibration method of RGB-D cameras based on the general batch calibration framework and motion capture system (MoCap), which extends the general batch calibration framework to the field of color or depth camera calibration. In the extrinsic calibration, we make full use of the characteristics of the MoCap system to customize the global refinement step for the hand-eye calibration, and it further improves the accuracy of the extrinsic calibration. Moreover, a non-recursive and novel data acquirement method is used to get the ground truth of every pixel in depth images and a one-step, model-free depth correction approach is applied to obtain the parameters of the depth correction models. Compared with the existing depth correction techniques, our method can simultaneously estimate the mean and variance of the depth error at different measurement distances. The experimental results show that our calibration method greatly improves the matching and measurement accuracy of color point cloud.

Autonomous navigation of mobile robots relies on safe and efficient obstacle avoidance algorithms. In recent years, a large amount of research work has begun to explore the application of deep reinforcement learning (DRL) for robot navigation in the dynamic environments, and end-to-end navigation approaches based on raw sensor data or pedestrian position information has been developed. In this dissertation, we propose a heterogeneous multi-robot obstacle avoidance algorithm based on grid map and DRL in a communication-free environment. We use the egocentric local grid map to represent the environmental information including robot's shape and observable appearances of other robots and obstacles, which can be easily generated by using multiple sensors or sensor fusion. Compared to other methods, the map-based approach is more robust to sensor noise, does not require robots' movement data and more efficient and easier

Abstract

to be deployed to real robots. We first train the neural network in a specified simulator of multiple mobile robots using DPPO, where a multi-stage curriculum learning strategy for multiple scenarios is used to improve the performance. Then we deploy the trained model to real robots to perform collision avoidance in their navigation without tedious parameter tuning. We evaluate the approach with multiple scenarios both in the simulator and the real world. Both qualitative and quantitative experiments show that our approach is efficient and outperforms existing DRL-based approaches in many indicators.

However, if the obstacle avoidance algorithm treats pedestrians as ordinary obstacles or robots, it may lead to close obstacle avoidance effects and bring discomfort to pedestrians. At the same time, the walking strategy of pedestrians is generally different from the obstacle avoidance strategy of robots. The crowd navigation does not completely meet the applicable conditions of the multi-robot obstacle avoidance algorithm. Therefore, we expand our map-based multi-robot obstacle avoidance approach, add the pedestrian map channel generated by the multi-sensor pedestrian perception module, and design a new reward function part and the multi-strategy pedestrian training environments. Experiments show that our method further improves the success rate of obstacle avoidance in dynamic pedestrian environments. At the same time, robots will also encounter more serious localization failure problems in a dynamic pedestrian environment. Therefore, we propose a dynamic localization method based on pedestrian perception filtering to improve the accuracy of robot localization. However, the passive localization method still cannot completely solve the problem of sensors obscured by a large number of dynamic human legs, so we propose a joint active relocation method based on QR code and laser sensor. When the localization error is large, the robot moves to the QR code recovery area and calibrates the global pose autonomously. Experimental results show that our relocation method can relocate the robot's global pose very accurately even when the localization is completely lost. Finally, based on the above research results and "criteria of closeness", we propose a specific deployment plan that attempts to solve the long-term autonomous navigation in open and crowd environments, and successfully conducted a large-scale long-term navigation test at the Chengdu Research Base of Giant Panda Breeding.

Key Words: Robot navigation; Calibration; Multi-sensor fusion; Multi-robot collision avoidance; Deep reinforcement learning; Grid map; Pedestrian tracking; Pedestrian navigation; QR-code localization; Robot relocation

目 录

第 1 章 绪论 ······	1
1.1 研究背景 ······	1
1.2 面临的困难与挑战 ······	7
1.3 本文主要工作和贡献 ······	8
1.4 文章组织及章节安排 ······	10
第 2 章 基于动作捕获系统的精确深度相机标定 ······	11
2.1 深度相机标定研究背景 ······	11
2.2 测距传感器简介 ······	13
2.2.1 超声传感器 ······	13
2.2.2 激光测距仪 ······	14
2.2.3 飞行时间相机 ······	14
2.2.4 结构光相机 ······	15
2.3 系统框架 ······	15
2.4 标定过程 ······	16
2.4.1 内参标定 ······	16
2.4.2 外参标定 ······	18
2.4.3 深度校准 ······	20
2.5 实验及结果分析 ······	24
2.5.1 手眼标定测试 ······	25
2.5.2 深度校准测试 ······	25
2.5.3 视觉 SLAM 实验 ······	28
2.5.4 对比实验 ······	29
2.6 本章小结 ······	30
第 3 章 基于栅格地图信息融合的多机器人导航 ······	31
3.1 多机器人避障研究背景 ······	31
3.2 传统机器人避障算法 ······	35
3.2.1 动态窗口法 (DWA) ······	35
3.2.2 速度障碍物法 (VO) ······	36
3.3 多机器人避障问题定义 ······	39
3.4 基于栅格地图的多机器人避障算法 ······	40
3.4.1 强化学习组件定义 ······	40

目 录

3.4.2 分布式近端策略优化算法 ······	43
3.4.3 多阶段课程学习 ······	48
3.5 仿真环境实验 ······	50
3.5.1 超参数设置 ······	51
3.5.2 实现细节 ······	52
3.5.3 泛化能力测试 ······	53
3.5.4 性能对比实验 ······	54
3.5.5 鲁棒性测试 ······	57
3.6 真实环境实验 ······	62
3.6.1 实验机器人硬件配置 ······	62
3.6.2 障碍物环境测试 ······	63
3.6.3 多机器人环境测试 ······	65
3.6.4 多传感器融合避障实验 ······	68
3.6.5 可佳机器人部署实验 ······	70
3.7 长距离导航系统 ······	71
3.7.1 系统框架 ······	71
3.7.2 长距离导航实验 ······	72
3.7.3 楼宇配送场景实验 ······	73
3.8 本章小结 ······	74
第 4 章 动态行人环境下的机器人导航系统 ······	75
4.1 动态行人导航研究背景 ······	75
4.2 基于多传感器融合的行人感知追踪系统 ······	77
4.2.1 行人检测 ······	78
4.2.2 最近邻融合 ······	78
4.2.3 行人追踪 ······	79
4.3 动态行人环境下的机器人鲁棒定位方法 ······	80
4.3.1 二维码识别 ······	82
4.3.2 二维码激光重定位 ······	83
4.3.3 实体机器人测试 ······	87
4.4 动态行人环境下的强化学习避障算法 ······	88
4.4.1 行人仿真 ······	88
4.4.2 避障策略学习 ······	89
4.4.3 仿真实验 ······	91
4.4.4 实体实验 ······	94

目 录

4.5 本章小结 ······	96
第 5 章 示范应用：成都大熊猫博物馆导览机器人 ······	97
5.1 导览机器人研究背景 ······	97
5.2 应用挑战分析与解决方案设计 ······	98
5.3 大熊猫导览机器人应用 ······	101
5.3.1 项目背景与需求分析 ······	101
5.3.2 机器人软硬件系统 ······	102
5.4 部署过程与测试结果 ······	104
5.4.1 测试环境介绍 ······	104
5.4.2 机器人与多传感器标定 ······	105
5.4.3 地图构建 ······	106
5.4.4 二维码部署 ······	107
5.4.5 行人感知测试 ······	108
5.4.6 密集人群导航测试 ······	109
5.5 本章小结 ······	113
第 6 章 总结与展望 ······	114
6.1 工作总结 ······	114
6.2 未来工作展望 ······	115
参考文献 ······	117
致谢 ······	133
在读期间发表的学术论文与取得的研究成果 ······	135

插 图 清 单

1.1	早期的半遥控移动机器人	1
1.2	以 ROS Navigation 为代表的机器人导航功能模块流程图	2
1.3	现阶段机器人导航技术的应用场景	3
1.4	携带多种传感器的移动机器人与无人机	6
2.1	不同种类的测距传感器	13
2.2	RGB-D 相机标定方法流程图	16
2.3	(a) 动作捕获系统; (b) 和 (c) 是动作捕获系统的标定工具; (d) 是粘有三个反光标记点的 Kinect 设备; (e) 用于内参和外参标定的棋盘格	18
2.4	深度图与 IR 图偏移标定	18
2.5	内参和外参标定场景示意图	19
2.6	不同距离的深度误差图	21
2.7	摄像头图像像素点投影到追踪平面 P 的示意图	23
2.8	标准 GP 与异方差 GP 预测对比	24
2.9	全局优化前后的重投影误差对比	25
2.10	不同标定方法的彩色图和深度图对齐效果和点云校准效果	26
2.11	五个实例像素点的多项式拟合曲线和 GP 拟合曲线	26
2.12	深度校准前后的深度误差统计	27
2.13	使用不同方法校准后的点云效果对比	27
2.14	RGB-D SLAM v2 实验结果	28
2.15	对比实验用空心立方体和标定精度对比	29
3.1	DWA 避障算法示意图	35
3.2	速度障碍物 VO 求解示意图	37
3.3	ORCA 求解示意图	38
3.4	障碍物的 ORCA 求解示意图	39
3.5	仿真器环境与生成的局部栅格图	41
3.6	策略网络架构图	47
3.7	输入帧数与网络层数对训练效果的影响	48
3.8	两种多机器人训练场景	49
3.9	课程学习对训练效果的影响和基于地图与传感器级方法的训练效果对比	50
3.10	非协作机器人仿真测试实验	53

3.11 大规模场景仿真测试实验	54
3.12 异质机器人仿真测试实验	55
3.13 不同避障策略定量测试实验效果图	57
3.14 不同避障算法抵抗传感器噪音能力的对比	58
3.15 使用点云生成避障栅格图	59
3.16 使用深度相机在 Gazebo 仿真器中测试避障实验的轨迹图	59
3.17 不同传感器视野大小对我们避障方法的影响	60
3.18 复杂静态训练环境 (a-b) 与训练过程曲线 (c)	61
3.19 复杂静态测试环境 (a-g) 与螺旋测试场景 (h) 和之字形测试场景 (i)	61
3.20 带有 2D 激光的多机器人实验平台	63
3.21 单机器人在静态和动态障碍物环境中的实体实验	64
3.22 两到三个机器人复杂环境实体实验	66
3.23 四机器人复杂环境实体实验	67
3.24 配备深度相机和 2D 激光的实验机器人	68
3.25 多传感器融合 3D 避障实验	69
3.26 可佳机器人实体实验	70
3.27 多传感器复杂环境长距离导航系统	71
3.28 基于深度强化学习的远距离走廊导航实验	72
3.29 机器人跨楼层大范围楼宇配送实验	73
3.30 楼宇配送避障场景	73
4.1 动态行人环境下的机器人导航系统框架	76
4.2 基于多传感器融合的行人感知追踪系统框架	77
4.3 (a) YOLO v3 的行人检测实例; (b) 多传感器融合的行人追踪效果实例	80
4.4 激光人腿过滤前后机器人定位效果对比	80
4.5 动态行人环境下的机器人鲁棒定位系统框架	81
4.6 激光或二维码定位与里程计纯定位示意图	82
4.7 ArUco 二维码的检测与识别流程	83
4.8 激光扫描匹配算法应用前后定位效果对比	87
4.9 二维码定位轨迹图对比	88
4.10 动态行人环境下的强化学习避障算法框架与策略网络	88
4.11 人类行走不同阶段的简单腿部运动模型	89
4.12 行人仿真训练场景	91
4.13 不同 PPO 算法训练过程的期望回报曲线和成功率曲线对比	92
4.14 不同方法的机器人测试轨迹图	93

4.15 三种 PPO 策略在五种不同行人策略环境的成功率对比 ······	94
4.16 动态行人避障测试实验的机器人与测试环境 ······	95
4.17 动态行人环境实体测试 ······	95
4.18 动态行人走廊场景实体测试 ······	96
5.1 导览机器人实例 ······	98
5.2 典型的 SLAM 框架图 ······	99
5.3 三种定位方式的切换策略 ······	100
5.4 大熊猫导览机器人硬件组成 ······	102
5.5 大熊猫导览机器人软件系统架构 ······	103
5.6 成都大熊猫博物馆大厅 ······	104
5.7 里程计线速度误差 ······	105
5.8 里程计纯定位前后误差 ······	106
5.9 大熊猫场馆定位导航地图 ······	107
5.10 二维码重定位部署与测试 ······	107
5.11 定位异常时二维码定位恢复策略测试 ······	108
5.12 行人识别与追踪测试 ······	109
5.13 多行为人群避障算法的切换策略 ······	110
5.14 人群导航算法示范性测试 ······	110
5.15 长时间机器人导航测试轨迹图 ······	111
5.16 机器人避让行人与小推车 ······	111
5.17 成都大熊猫博物馆的大量行人测试场景 ······	112
5.18 语音提示行人注意 ······	112
5.19 成都大熊猫博物馆的儿童测试场景 ······	113
5.20 儿童对机器人的干扰 ······	113

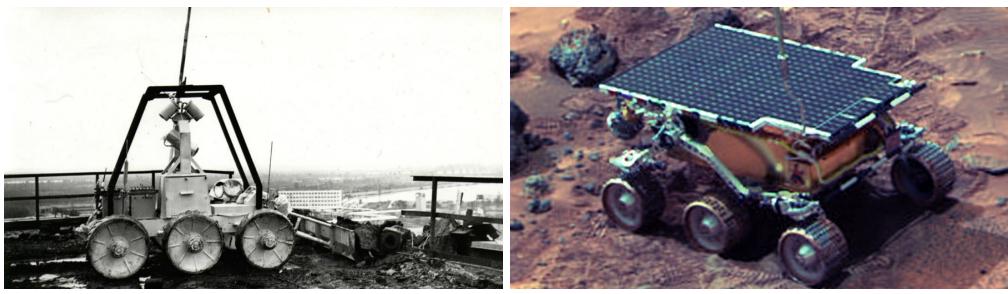
表 格 清 单

2.1 视觉 SLAM 定位误差 ATE 和 RPE 的均方根误差 (RMSE) 值	28
3.1 对比实验中 NH-ORCA 算法的超参数设置	51
3.2 训练算法 3.1 和算法 3.2 中的超参数设置	52
3.3 不同多机器人避障方法在多种测试环境中评估的性能指标	56
3.4 Hokuyo 二维激光传感器详细参数	62
3.5 四机器人场景的测试结果	65
3.6 机器人与传感器标定结果	68
4.1 二维码重定位准确度	87
4.2 不同方法的测试指标	92
5.1 机器人与多传感器外参标定结果	105

第1章 绪论

1.1 研究背景

几十年来，机器人学已经在工业制造中取得了巨大成功。机器人手臂或者机械手，构成了约 200 亿美元的全球市场规模。早在 1980 年代，机械臂技术就已成功地应用于汽车制造业。在机器人技术领域，它是使用最广泛的自动化机械设备，并且在许多工业生产工序中（例如装配、喷涂、焊接、高温铸造和锻造等）都可以用机械臂代替人工操作。但对于这些成功的工业应用，商业机器人存在一个共同的根本缺点：缺少移动能力。固定的机械手臂被安置在流水线固定的位置，其运动范围受很大限制。而且这些位置都要事先人工标定，一经确定改变代价很大。相反，移动机器人可以在整个工作区域中移动和行走，在其有效位置灵活地施展技能。早期阶段，低复杂度的移动机器人主要用于人类无法到达的区域。如图1.1(a) 所示为 1986 年在切尔诺贝利专门探索石棺用于清理核辐射现场的遥控推土机器人和图1.1(b) 中描绘的 1997 年在探险者探索金星期间所使用的旅居者机器人，它几乎完全由地面人类操作员遥控，但依靠自身机载传感器来进行障碍检测和运动控制支持。



(a) 遥控推土机器人

(b) 旅居者机器人

图 1.1 早期的半遥控移动机器人

随着人工智能与机器人技术的快速发展，越来越多的移动机器人在我们人类日常生活环境中得到应用，如在商场或养老院提供服务^[1]、在工厂中与人协作^[2]、甚至在医疗保健中提供辅助^[3]。这些机器人之所以引人注目，并不是因为它们的机动性，而是因为它们的高度自主性^[4]。机器人在其环境中自主移动导航的能力是其智能程度的重要体现，因此机器人导航技术也是机器人研究领域的重点研究内容之一。机器人导航是指移动机器人通过其自身携带的传感器感知周围环境信息和自身位置状态，实现在有障碍（人）的环境中面向目标的自主运动^[5]。其核心在于解决三个问题：所处的环境是什么样子的？当前在环境中的什

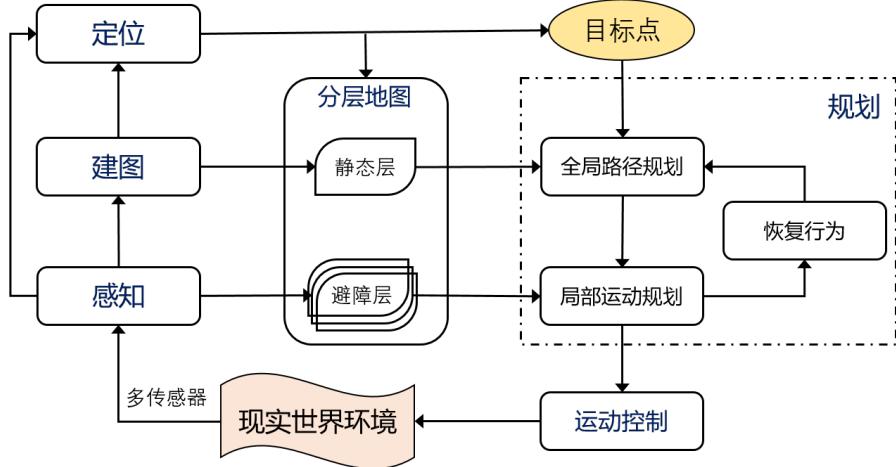


图 1.2 以 ROS Navigation 为代表的机器人导航功能模块流程图

么位置？如何到达目的地？图1.2粗略描绘了以开源项目 ROS Navigation^①为代表的机器人导航功能模块间的关系，主要包括感知、建图、定位、规划和运动控制等模块。任何种类的自主系统的最重要任务之一是获取关于其周围环境的知识，感知模块负责从不同传感器感受的测量中提取有意义的信息。用在机器人上的传感器种类丰富，有些传感器只用于测量简单的值，如测量电机转速的编码器等。而其他更复杂的传感器可以用来获取关于机器人的环境信息，甚至直接测量机器人的全局位置。因为机器人有可能在未知的开放环境中移动，它常常碰见未预料的环境特征，所以这种对环境的感知能力是非常重要的。建图模块指的是建立环境的栅格地图^[6]的过程，简单的问题定义为：假设机器人的位姿已知，如何使用嘈杂和不确定的测量数据来生成一致的环境信息。栅格地图是常用的环境信息表示方式，它使用一系列环境随机变量来表示环境信息。每个随机变量是指示该栅格位置是否被障碍物占用的二进制数据，栅格地图的构建是对以上随机变量进行近似后验估计。但当机器人进入未知的环境中，不能得到环境地图且也不知道自身的位置时，即时定位与建图^[7]（Simultaneous Localization and Mapping, SLAM）问题就被提出来。该问题定义为机器人试图同时获得环境地图并确定自己相对于该地图的位置。由 SLAM 事先建立好全局地图后，我们确定机器人在地图中的位姿的问题称为机器人定位，也称为位姿估计。定位问题也分为很多种难易不同的种类，包括机器人起始位姿已知的位置追踪问题、起始位姿未知的全局定位问题和定位失败后的机器人绑架问题（kidnapped robot problem），也包括静态环境下的定位和动态环境下的定位，和主动与被动定位问题^[7]。由 SLAM 建立全局静态层地图和感知实时更新的避障层地图，机器人知道了所处的环境是什么样子。定位模块由传感器感知数据和静态层地图可以帮助机器人确定自身位姿，解决了当前所在位置在哪的问题。机器人导航的剩下一

^①<http://wiki.ros.org/navigation>

个问题是如何让机器人安全到达目标点，这需要机器人规划模块来解决。在实际的机器人导航中，机器人有规划和反应两种互补状态，各自对另一方都是极其重要的。机器人为了到达它的目标位置，需要首先进行路径规划，然后机器人在执行期间必须对一些规划未预见的事件（如突然出现的障碍物和行人）作出反应，使之仍然能够安全无碰撞地到达目标点。规划模块输出的一般是机器人的速度命令，这种命令被运动控制模块接收后根据机器人硬件特点转换为电机的转速指令。其中运动估计模型可以计算出里程计信息，这是机器人对自身运动位置的粗略估计，对定位模块有辅助作用。除此之外，机器人运动控制中还会涉及机器人的工作空间，一般认为是运动规划算法需要考虑的机器人形状和大小。机器人的运动改变了环境，继而产生新的多传感器感知数据，从而影响下一步机器人的运动决策。

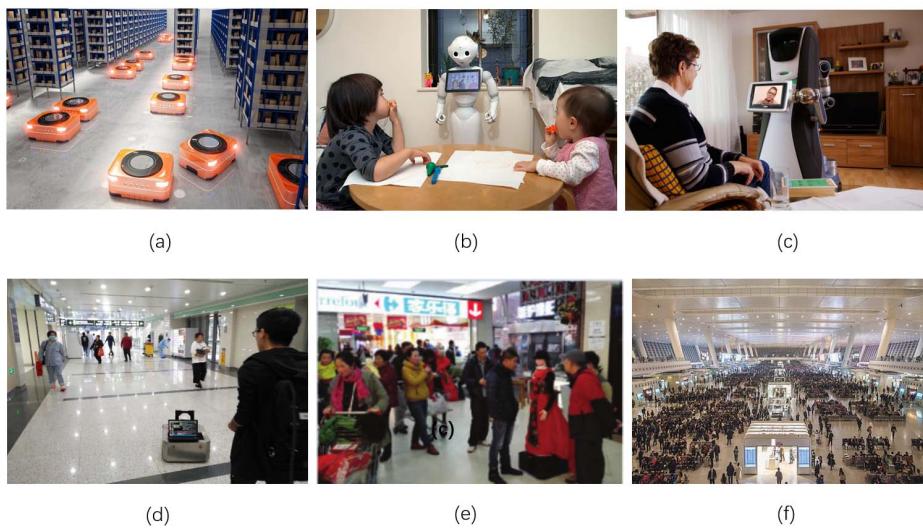


图 1.3 现阶段机器人导航技术的应用场景

根据机器人应用场景，机器人导航技术大致分为工业机器人导航阶段、家庭服务机器人导航阶段和公共服务机器人导航阶段。图1.3(a)中描绘了AGV (Autonomous guided vehicle, 自主导向车) 通过跟踪安装在地面的专用导向线，或使用机载激光在用户专用地图范围内定位，在仓库中自主地搬运物品。这是典型的工业机器人导航场景，这类导航技术用在结构化的场景中，只需要按照指定的路线行进，不需要复杂的避障技术。在实验室环境工作的标准研究型机器人平台可以深入研究诸如机器人感知、定位和导航之类的高级问题，这是目前移动机器人最大的应用市场之一。图1.3(b) 和1.3(c) 描述了移动机器人走进家庭为孩子教育和老人护理提供服务，这类机器人往往配备多种传感器来应对复杂的非结构化家庭环境，同时对机器人的避障和与人交互的能力有很高要求。图1.3(d), 1.3(e) 和1.3(f) 是更具挑战性的移动机器人公共服务应用场景，它们分别描绘了机器人

在医院、商场和火车站等密集人群环境下提供公共服务。随着传感器技术和概率机器人理论^[8]的发展，普通简单室内场景中的机器人建图、定位和避障都已经有基本的解决方案。但如何安全地与环境中其他动态智能体（行人或移动障碍物）交互仍然是移动机器人或自动驾驶汽车的一个基本挑战。尽管导航技术在过去十年中取得了令人瞩目的进步，但全自主导航仍然面临挑战，特别是在与其他移动智能体共同相处的不确定动态环境中。之所以出现部署困难，主要是因为规划系统通常不了解其他智能体的意图和策略（即智能体的移动目标和期望的路径），这也促使人们开始重点研究动态行人避障算法。

现有关于动态行人避障算法的工作可以分为合作和非合作的方法。非合作的方法首先预测其他智能体的动作，然后相对于其他智能体的预测动作规划机器人的无碰撞路径。但是这可能会导致机器人冻结问题^[9]（freezing robot problem），即因为其他智能体的预测路径占据了可穿越空间的很大一部分，机器人无法找到任何可行的路径。合作方法通过在规划程序中对交互行为进行建模来解决此问题，使机器人的动作可以影响其他智能体的动作，从而让所有智能体共同承担避障的责任。然而传统的动态行人避障算法可能需要大量的计算代价或实践中无法满足的一组假设。这些方法还涉及许多参数，这些参数需要手动调整，而不是能够从过去的经验中自主学习，这使得它们更难处理不可预测的情况。基于学习的方法尝试优化参数化的策略，以避免各种环境中的碰撞。深度神经网络已广泛应用于有监督的学习范式中，以训练一种避免碰撞的策略，该策略将传感器输入映射到机器人的控制命令，以在具有静态障碍物的环境中控制机器人移动导航^[10-12]。但是，基于监督学习的方法需要大量标记的训练数据集，并且训练数据集和测试环境之间的分布可能不匹配，这限制了它们在现实世界中的应用。另一方面，深度强化学习方法在许多具有挑战性的任务中都取得了显著成功^[13-16]。相反，基于深度强化学习的方法能够从大量的试验和相应的反馈（奖励）中学习。机器人需要和环境进行大量的交互，直接在现实世界中收集经验代价非常大且很耗时，还会给机器人和环境带来安全问题，因此在仿真器中训练再迁移到真实世界是一个更合适的方法。现有基于深度强化学习解决动态行人避障问题的仿真环境主要分为两类：多机器人仿真环境和动态行人仿真环境。基于多机器人的动态仿真环境是基于自我对弈^[17]（self-play）的设计理念，环境中的智能体共享相同的策略网络。多机器人仿真训练环境不需要考虑仿真复杂的行人运动策略，且经验的收集随着机器人数的增加更加高效。然而，由于行人智能体的特殊性，动态行人避障算法有别于多机器人避障算法。如果把行人都视为普通的障碍物或机器人，机器人可能会导致近距离的避障效果，给行人带来不适^[18]。同时行人的行走策略与机器人的避障策略一般不同，机器人在动态行人环境中的避障不完全符合共享相同策略多机器人避障算法适用条件^[19]。因此，仿真行人

运动策略并搭建动态行人环境进行训练是解决动态行人避障策略训练的常用方法。然而现实和仿真世界中策略网络的输入和输出的差异决定了仿真器训练策略能否更好地迁移到实体，因此网络输入和输出的设计要让仿真器训练的策略更好应用到真实环境，同时也应该兼顾各种传感器输入并很好地融合它们的信息，这在当前基于深度强化学习的行人与多机器人避障问题中仍未得到很好地解决。在更加复杂的开放人群环境中，机器人还要面临和人群的攻防博弈问题。为了顺利完成任务，机器人需要在人群中主动与行人交互以到达目标点。因此本文将分别从多机器人避障、动态行人避障和开放人群避障三个不同维度探讨机器人在复杂动态行人环境中的避障问题，它们依次为后者的研究基础，最终我们在真实的博物馆场景中测试了我们的人群避障模块。

动态行人环境下的机器人导航系统还存在更严重的定位丢失问题^[20]。当机器人在给定的地图上由于较大的定位不确定性或错误而无法准确定位时，就会出现定位丢失问题。在当前的定位方法中，一个相当常见的假设是：当机器人移动时，世界保持不变(换句话说，地标是静态的)。这种静态世界的假设^[21-23]在小规模动态场景中运行短期定位是正确的，但在高度动态的行人环境中，激光和地图会发生错误的数据关联，从而导致定位发生错误。通常最直接的解决方法是试图在感知数据中过滤运动物体^[22,24]，他们认为运动物体的感知追踪是解决动态环境定位的关键。但在密集的人群环境中，此类方法会过滤掉大量传感器数据，从而可能导致定位失败。另一类方法通过使用视觉传感器观察不会被行人遮挡的天花板来实现机器人定位。CV-SLAM^[25]是最早实现该想法的工作，他们及其他后续工作^[26-28]都是通过提取天花板特定视觉特征来实现定位的。虽然此类方法容易部署，但当天花板纹理较少时，此类定位方法会产生较大误差。为了能在纹理较少的环境中鲁棒定位，文献^[29-30]通过在天花板上粘贴大量二维码来计算出当前机器人在全局坐标系中的位姿。然而此类方法需要粘贴大量的二维码标签，部署麻烦且影响环境美观，甚至在某些大型场馆的很高天花板上部署二维码标签是不可行的。由于基于视觉感知，此类方法受光照影响严重。同时产业界（如送餐机器人）尝试通过其他改变环境的方法来部署稳定的导航系统，如有轨磁导航方案、RFID 地标定位、天花板红外标签定位等，但此类方法环境改造大，部署困难。以上解决动态环境定位丢失问题的方法都是被动方法^[31-33]，他们假设机器人的运动和传感器指向的方向不能被动态控制，有选择地集中利用传感器流来最小化定位的不确定性或误差。然而在人群密集的高度动态场景中，定位所需的显著特征可能都被遮挡或不适合大规模改造环境。因此机器人必须主动决定“移动到哪里”来解决遮挡问题，并“向哪里看”从丢失的定位中恢复^[8,34-35]。一个理想的长期稳定定位解决方案应该是故障感知的并提供恢复机制来主动校准定位，然而现有的机器人定位方法都没有提供鲁棒有效且易于部署

的解决方案^[7]。

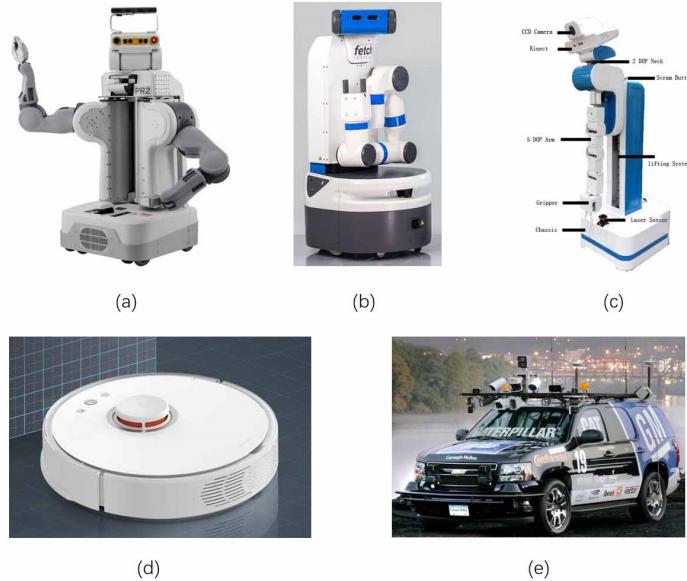


图 1.4 携带多种传感器的移动机器人与无人车：(a) Willow Garage 公司生产的 PR2 机器人；(b) Fetch 机器人；(c) 中科大可佳机器人；(d) 小米扫地机器人；(e) 卡内基梅隆大学无人车

另一方面，为了对复杂环境进行全面的感知，现有机器人或者无人车配备了越来越多的传感器。如图1.4所示，图中的移动机器人或者无人车都包含大量的诸如摄像头、雷达等传感器。其中著名的 PR2 机器人（图 (a)）头部携带广角和窄角立体摄像头、Kinect 传感器和五百万像素彩色摄像头，躯干上配备上下转动的 Hokuyo UTM-30LX 激光测距仪和 Microstrain 3DM-GX2 IMU，手臂上安装有以太网摄像机、三轴加速度计和指尖压力传感器，移动底座则安装了 Hokuyo UTM-30LX 激光测距仪。这些传感器模型包含许多参数需要精确定定，同时为了更好融合多传感器感知信息，他们之间的空间位置关系也需要标定。因此机器人标定技术是机器人导航（甚至是机器人系统）能够稳定运行的首要保障^[36]。通常机器人标定大致分为传感器内参标定（传感器模型参数），传感器外参标定（传感器之间或与机器人坐标系之间的空间转换关系）和机器人硬件标定（如手臂关节的标定、移动轮子的参数标定等）。尽管有许多文献提供了多种标定解决方案，但是仍没有针对一般机器人或者智能汽车的多个传感器进行标定的简单快速解决方案^[37]。标定的大部分工作集中于传感器到传感器成对的标定，如摄像头之间的标定^[38-42] 或者摄像头与激光之间的标定^[43-47]。当考虑成对的传感器组合时，根据成对传感器中每个传感器的形式有几种可能的组合，其中大多数组合已有相应的论文提供了解决方法，如 RGB 摄像头之间的组合^[38-42,44]、RGB 和深度摄像头的组合^[48-50]、摄像头和 2D 激光之间^[43-44,51]、2D 激光和 3D 激光

之间^[46]、摄像头和 3D 激光之间^[52] 和摄像头和雷达之间^[53]。但是所有这些方法都有明显的缺点，即只能标定其中一对传感器组合，这不适用于智能汽车或者更复杂的机器人系统。有一些工作提出了多传感器的同时标定，在文献^[54] 中，作者提出了一个联合目标函数来相对于 RGB-D 摄像机同时标定与三个 RGB 摄像机的关系。在文献^[55] 中，作者提出了一种联合估计传感器之间的时间偏移和空间变换的方法，这个方法不依赖于特定传感器的独特属性，而且不用特定的标定工具，只使用场景中存在的平面。在文献^[56] 中，作者提出了针对 PR2 机器人的关节偏移和传感器位置的标定，该方法考虑了传感器的不确定性，而且用束调整类似的方法进行建模。文献^[57] 也是基于束调整的方法，但他们的方法更加通用，可以用在任何机器人系统。然而不幸的是，大多数标定工作都是由专业操作人员手动完成的，标定过程既费时又费力。标定的效果通常取决于操作员以及所使用的设备或者工具的经验和知识。更糟糕的是，对于量产的机器人产品可能需要一遍又一遍重复的相同人工操作。因此在文献^[58] 中，作者提出了随着服务机器人数量激增而必须解决的批量标定问题。为了解决这个问题，作者基于动作捕获系统提出了通用批量标定框架，他们的目标是建立一个自动化快速机器人标定平台。他们提出的标定平台具有自动化、批量化和通用化等特点，可以适配不同种类和数量的机器人与多传感器标定。在文献^[58] 中，作者还对里程计、激光进行了标定测试实验。但是他们没有提出针对 RGB 摄像机或者深度相机的标定方法且核心外参标定方法准确度不高。

综上所述，本文主要研究动态行人环境下的鲁棒导航系统，主要涉及导航系统的标定、避障与定位等相关模块，包含机器人与多传感器标定、动态行人避障和动态环境鲁棒定位等具体问题。

1.2 面临的困难与挑战

在过去的几十年里，机器人导航算法的各个模块都取得了很大发展，出现了许多先进的研究成果和成熟的技术方案。但随着移动机器人被用于动态开放的密集人群环境中，这些原有的导航相关技术方法也面临着一些困难和挑战：

1. 快速准确地标定机器人携带的多个或多种传感器。越来越多的机器人为提高其感知周围环境的能力，配备了越来越多种类和数量的传感器。这些传感器与机器人的标定是融合它们感知信息的前提。现有的传感器标定方法大多需要专业人员操作且只能成对的标定，并且标定的传感器通常都要同时观察同一个标定装置，这在某些情况下无法满足。通用批量标定框架可以解决快速精确机器人与多传感器标定问题，但他们的方法仅限于激光与机器人标定且核心外参标定方法准确度不高。然而 RGB-D 相机传感器

可以提供融合的彩色和深度信息（彩色点云）且价格相对低廉，是目前移动机器人上应用广泛的三维测距传感器，通用批量标定框架急需扩展到深度相机标定领域。快速准确的标定这些传感器的内参和外参是使用它们进行导航的首要前提。

2. 复杂动态行人场景下安全高效自主移动。在动态、静态形状各异的障碍物和动态行人并存的复杂环境下，机器人需要区分对待环境中的行人和障碍物，当接近行人和接近障碍物或者其他机器人时，机器人应该表现出不同的避障方式。同时当场景的动态复杂性增加时，由于环境不确定性的累积和出于安全性考虑，机器人倾向于保守的行为来最小化风险。从而可能无法找到可行的前进路径，最终机器人将在原地停止或在两个方向之间摆动，产生机器人冻结问题。另外在更加开放密集的人群环境中，机器人还要面临和行人的攻防博弈问题。为了高效完成任务，机器人需要在人群中主动与行人交互以到达目标点。因此机器人需要具有在动态开放行人环境中安全高效的导航避障能力。
3. 动态行人场景下的长期鲁棒定位。在动态行人环境中，激光和地图会发生错误的数据关联，机器人还会遇到更严重的定位丢失问题。虽然通过改造环境和天花板定位方式可以很大程度上解决动态环境定位丢失问题，但该方法需要较高的部署代价和环境要求，不适用于部署到所有的现实环境。同时机器人定位系统出现异常时是很脆弱的，主要是因为几乎所有鲁棒的定位技术都基于迭代优化或滤波。一个理想的长期稳定定位解决方案应该是故障感知的并提供恢复机制来主动校准定位，然而现有的机器人定位方法都没有提供鲁棒有效且易于部署的解决方案。因此如何改善机器人在动态行人环境中定位的准确性以及如何自主校准机器人全局定位误差是动态行人环境下成功导航的首要问题。

1.3 本文主要工作和贡献

本文的研究目标是提高移动机器人在复杂动态行人场景下的自主导航能力，即试图解决1.2节提到的三大困难与挑战问题。本文的主要研究内容和贡献概括如下：

1. 用于解决多传感器标定问题的通用批量标定框架仅限于激光与机器人标定且核心外参标定方法准确度不高。在本文中我们基于通用批量标定框架和动作捕获系统提出了精确的 RGB-D 相机标定方法，使通用批量标定框架扩展到彩色或深度相机标定领域。我们利用动作捕获系统的精准特性来定制手眼标定的全局优化步骤，改进了外参标定精度。在深度校准中，我们

使用非递归的新颖数据获取方法来获取深度图像中每个像素的真实值，并采用基于异方差高斯过程的校准方法来估计不同测量距离处深度误差的均值和方差。实验结果表明我们的标定方法大大提高了彩色点云的匹配精度与深度测量的准确度，并且该方法基于通用批量标定框架可以轻松推广到更多种类与数量的传感器间的外参标定。

2. 现有的基于深度强化学习的多机器人避障算法不能兼容各种种类与数量的传感器或者需要复杂的前端感知处理。我们提出了一种在无通信环境中基于栅格地图和深度强化学习的多机器人避障方法，基于栅格地图的方法适用于多种传感器，易于在仿真环境中训练，对嘈杂的传感器数据更鲁棒，不需要精确复杂的前端感知过程，考虑了机器人的大小和形状，并且更易于部署到实际机器人中。我们在仿真器中以及在现实世界中的四个差动机器人上，都针对多种场景评估了该方法。定性和定量实验均表明，我们的方法在许多指标上都优于传统的和其他基于深度强化学习的避障方法。
3. 机器人在人群中的避障不完全符合同策略多机器人避障算法适用条件，而且当接近行人和接近障碍物或者其他机器人时，机器人应该表现出不同的避障方式。因此我们扩展了基于栅格地图的深度强化学习多机器人避障方法，在传感器栅格地图中添加了由多传感器行人感知追踪模块生成的行人地图通道，设计了新的远离行人的回报函数部分和基于社会力模型与速度障碍物法仿真的多策略行人仿真训练环境，提出了适用于包含形状各异的动态、静态障碍物和动态行人复杂场景下的避障算法。实验表明我们的方法进一步提升了机器人在不同行人策略的复杂环境中的避障成功率。
4. 大多数定位算法基于世界静止假设，但在高度动态的行人环境中，该假设不成立。因此我们提出了基于行人感知过滤的动态行人环境定位方法，改进了动态行人环境下机器人定位的准确性。同时一个理想的长期稳定定位解决方案应该是故障感知的并提供恢复机制来主动校准定位，然而现有的机器人定位方法都没有提供鲁棒有效且易于部署的解决方案。因此我们提出了二维码激光联合主动全局重定位方法和自主检测校准机器人全局定位误差策略，以解决在动态行人复杂环境下机器人长期稳定定位问题。实验结果显示我们的重定位方法可以在定位完全丢失时仍能非常准确地重新定位机器人全局位姿。
5. 最后我们在成都大熊猫博物馆进行了大规模长时间实地场景测试来验证我们提出的复杂动态行人导航系统的实际价值，同时我们根据“封闭性准则”，提出了我们试图解决开放环境机器人长期自主导航的具体部署方案，包括基于二维码激光重定位的多定位方式切换策略和多行为导航避障策略。在这个高度复杂动态的开放密集人群环境中，机器人顺利完成导航任务 3000

多次，其中不间断导航测试时长高达7个小时。

1.4 文章组织及章节安排

本文的剩余部分组织如下：在第二章中，我们介绍了基于通用批量标定框架和动作捕获系统的精确深度相机标定方法，并进行了大量验证实验证明我们提出的深度相机标定方法的效果与优势。在第三章，我们详细介绍了基于栅格地图和深度强化学习的异质多机器人避障方法的细节，这项工作旨在说明基于栅格地图信息输入的强化学习避障算法的优势，且该多机器人避障算法是我们提出的基于深度强化学习的动态行人环境下避障算法基础，同时此多机器人避障算法本身具有不需要复杂前端感知处理等优点和实际应用价值，可以应用于多机器人无人仓储等实际场景。本文进行了大量的仿真与实体测试来说明我们的避障算法性能。最后我们把多机器人避障算法集成到了导航系统中并做了长距离导航实体实验、多传感器融合导航实体实验、异质多机器人实验和楼宇配送场景实验。在第四章，我们主要介绍动态行人环境下基于多传感器信息的机器人导航系统。首先我们介绍基于多传感器信息融合的行人感知追踪模块，然后介绍基于行人感知过滤的行人环境定位方法以及基于二维码激光联合主动全局重定位方法，紧接着我们介绍扩展的面向多种行人运动策略的多传感器感知多地图通道深度强化学习动态行人避障模块。在第五章，我们将论文中的方法实际应用于成都大熊猫基地的博物馆导览机器人项目中，并在成都大熊猫基地博物馆进行了大量实地开放场景测试。在文章最后的第六章，我们总结了文章内容并对未来研究方向进行了展望。

第 2 章 基于动作捕获系统的精确深度相机标定

机器人与多传感器的精确标定是机器人导航的前提。本章基于动作捕获系统和通用批量标定框架提出了精确 RGB-D 相机（深度相机）标定方法，我们详细介绍了该深度相机标定方法的背景、内容和实现，最后通过大量实验验证了我们方法的优点。

2.1 深度相机标定研究背景

由于家庭服务机器人的许多核心功能（例如视觉 SLAM^[59]，机器人导航^[60]，物体识别^[61] 和物体操作^[62]）都需要融合的彩色和深度信息，越来越多的机器人配备了彩色（RGB）和深度（D）相机。基本上，此类 RGB-D 传感器由刚性连接的彩色相机和深度相机组成，其中深度传感器可以是飞行时间（ToF）相机或基于结构光的传感器。本文中我们将重点放在类似 Kinect 的传感器（结构光传感器，如 Intel Realsense SR300 和 Asus Xtion Pro Live）上，因为它们已广泛用于移动机器人中。尽管这些设备已经过出厂校准，但是此校准的质量只是对于大多数视频游戏来说是足够的。然而对于鲁棒的机器人应用，我们仍然需要适当的标定程序。对于这类深度传感器，我们需要标定彩色和深度相机的内部参数（焦距、光心和镜头畸变参数）以及它们之间的空间对准（外部参数，即相对空间位置和方向）。此外，我们注意到深度测量中的偏差和不确定性会随着测量距离的增加而增加，并且在不同的像素坐标中，测量的偏差和不确定性的变化并不是平稳的。因此，RGB-D 相机的完整标定主要分为内部和外部参数的标定以及深度测量的校准。

为了获得深度相机的内部和外部参数，学术研究中存在基于深度相机不同输出数据的各种标定方法，例如基于红外（IR）图像、视差图像或深度图像。诸如文献^[63] 之类的早期研究是基于 IR 图像和标准 RGB 相机标定技术的，他们使用棋盘格图案来标定 RGB 和 IR 相机的内部参数，以及它们之间的外部转换。但是，他们的方法要求不同的相机同时观察相同的棋盘格图案，并且由于广泛使用的 ROS Kinect 驱动程序^[64] 不支持同时输出 IR 和 RGB 图像，标定过程变得繁琐，结果不精确。Herrera 等人^[65] 的工作是一种基于视差图像的典型方法，标定过程中他们让 Kinect 在距离不同的地方观察附有棋盘格图案的大型平面。他们的工作有两个主要弱点：首先，他们要求专业人员来手动选择深度图像中的相应特征。其次，他们的方法严重依赖于特定的 Kinect 型视差数据。基于深度图像的其他方法则使用 RGB 和深度传感器同时观察特定标定对象。在文献^[66] 中，作

者使用了具有数十个圆形孔的大型定制木板，并且需要大量的手动步骤才能将每个 RGB 和深度图像对中的孔中心关联起来。此外，在文献^[67]中，作者使用了球形物体篮球来作为标定工具。基于深度图像的标定方法的共同局限性在于深度图的噪声非常大，因此标定结果不够准确。

对于深度校准，Smisek 等人^[68]表明 Kinect 传感器受到某些径向对称变形的影响，然后 Herrera 等人^[65]对标定过程中的失真校准进行了首次尝试，但是他们的方法使用外部高清相机观察棋盘格图案，因此不适用于 RGB 相机和深度传感器之间具有长距离间隔的情况。Zhang 等人^[69]的研究表明 Kinect 提供的深度测量值是实际深度值的线性函数。深度校准的主要问题是如何准确地获得深度图像中每个像素点对应的真实深度值。Teichman 等人^[70]假设由 SLAM 生成的特写点云图可以用作相应位置的深度测量的真实值。显然，他们的方法在很大程度上依赖于 SLAM 的定位和建图的精度。近期 Basso 等人^[71]将深度误差分为两个不同的部分：失真误差和全局误差，并分别对其进行了处理。他们假设最近的深度测量没有误差，并且在同一像素位置的相近深度测量的深度误差相同。由于他们使用多项式曲线拟合来校准深度值，因此需要提前人为确定多项式形式。

受流行的 RGB-D SLAM 基准评估工作^[72]的启发，我们提出了一种新颖的 RGB-D 相机标定方法，该方法基于使用高精度动作捕获系统（MoCap）的通用批量标定框架^[58]，消除了获取深度测量真实值的系统假设。与文献^[72]中提出的外参标定方法不同，我们的方法基于手眼标定技术，消除了由于粘贴在棋盘上的标记点位置不正确而导致的不可避免误差。本文我们仍然使用红外图像标定深度相机，主要是因为与深度图相比它具有低噪声的特性，可以估计内部和外部参数。与传统的基于红外图像的方法相比，我们的方法不是直接估计深度相机和彩色相机之间的空间关系，从而不需要深度相机和彩色相机同时观察同一块棋盘格图案，而是标定动作捕获系统提供的全局世界坐标系和各个相机坐标系之间的空间转换关系。这种做法的另一个好处是，我们可以轻松扩展到标定多个相机之间的外部参数^[73-74]或与其他类型传感器^[75-76]的外部参数。对于深度传感器提供的深度测量，我们采用了一种可以同时减少变形和系统误差的深度校准模型来获得对环境更精确的深度测量，该模型表示为一组无需手动调参的高斯过程（Gaussian Process，GP）。为了获得模型参数，我们充分利用动作捕获系统来获得不同距离处每个像素的相应深度真实性和测量值。更重要的是，我们发现随着测量距离的增加，测量深度值的不确定性也增加，即误差的方差增加。因此，我们采用稀疏的异方差高斯过程^[77]来同时估计测量误差的均值和方差，即深度误差相对于测量距离的概率分布，这对于机器人研究中的状态估计问题至关重要。我们的主要贡献概述如下：

- 提出了 RGB-D 传感器的完整精确标定流程，包括内参标定、外参标定和深

度校准。

- 提出了相机的内参和外参联合标定方法，该方法主要基于全局优化的手眼标定技术。
- 提出了直接且非递归的逐像素深度测量真实值数据获取方法，并基于异方差高斯过程来校准深度测量值，可以同时估计深度误差的均值和方差。

2.2 测距传感器简介

用在移动机器人上的传感器种类丰富，有些传感器只用来测量简单的值（如电机转速），但还有不少传感器可以用来测量周围环境信息。本小节介绍下常见的感知环境信息并对导航避障特别相关的测距传感器，此类测距传感器可以返回障碍物的距离信息，如超声传感器、激光测距仪、飞行时间相机和结构光相机等。

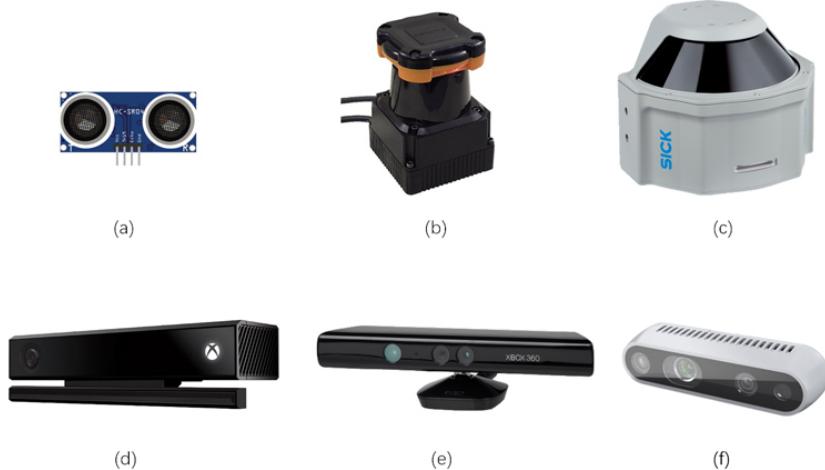


图 2.1 不同种类的测距传感器

2.2.1 超声传感器

超声传感器通过发射超声压力波包并测量其返回到接收器所占有的时间来计算障碍物距离。引起反射的物体距离 d 可以根据超声传播速度 c 和飞行时间 t 计算得到：

$$d = \frac{ct}{2}.$$

典型的超声波具有 40-180kHz 之间的频率，超声波具有频率高、波长短、衍射现象小、方向性特别好的特性，并且可以成为射线而定向传播。当超声波击中杂质或界面时，它将产生明显的反射并形成反射回波。在移动机器人中，普通的超声波传感器可以实现大约 2cm 的分辨率。常用的超声波传感器由压电晶片组成，可以发射和接收超声波。低功率超声探头主要用于检测，它有许多不同的结构，

它可以分为直探头（纵向波）、斜探头（横向波）、表面波探头（表面波）、兰姆波探头（兰姆波）、和双探头（发射一个探头，接收一个探头，如图2.1(a)所示）等。由于声音以锥形方式传播，有大约 $20^\circ - 40^\circ$ 的孔径角，因此，超声波传感器只能告诉我们在测量锥内一定距离处有障碍物，而不是点测量。超声传感器主要还有一些缺点，即误差区域、带宽和交叉灵敏度。超声器的准确度是基于离开障碍物的声波是成功垂直反射的名义值。但现实中，超声探测返回的角度可能远非垂直，还有一些可能不会产生单元感知很强的反射或吸收声波的物体，如泡沫和衣服等。另外，单个超声波传感器在中等开放空间中的周期时间较慢，如测量3m远的障碍物需要 20ms，而且如果机器人配备具有 20 个超声传感器组成的环，为了使传感器之间干涉最小，传感器需要顺序地激发和测量，那么所有传感器更新一遍需要 0.4s，传感器的更新频率仅有 2.5hz，这对于移动机器人来说具有较大延迟。但超声传感器可以探测类似透明玻璃的障碍物，这是激光传感器不能做到的。

2.2.2 激光测距仪

激光测距仪是飞行时间传感器的一种，由于发射激光而不是超声，相对于超声传感器有了很大改进。激光测距仪通常由发射器和接收器组成，发射器使用激光来照亮物体，接收器检测光的分量。激光测距仪也常常被称为激光雷达，激光雷达根据激光到达目标然后返回接收器所需的时间来计算障碍物距离。激光雷达使用带有镜子的机械结构扫掠光束以覆盖平面中的场景，或者在三维世界中使用旋转和摆动的镜子结构，有三种测量光束飞行时间的方法。第一种方法是使用脉冲激光直接测量占用的时间，第二种方法是测量调频连续波和接收到的反射之间的差频，第三种方法是测量反射光的相移。激光测距仪的角分辨率远高于超声传感器，而且超声测距的一个重要误差涉及到能量的相干反射，而光只会发生在测量高度抛光表面的物体的时候，激光测距仪不能测量光学上透明的物体。如图2.1(b) 所示为 Hokuyo 公司生产的 UTM-30LX 2D 激光雷达，有效测量范围达到 30m，270 度范围。图2.1(c) 为 SICK MRS6000 3D 激光雷达，可以为无人车提供丰富的 3D 环境信息。

2.2.3 飞行时间相机

飞行时间相机（TOF 相机）工作原理和激光测距仪类似。TOF 相机测距的原理是连续将光脉冲发送到目标，然后使用传感器接收从对象返回的光，通过探测光脉冲的飞行（往返）时间来得到测量物体距离。激光逐点扫描，而 TOF 相机同时获取整个图像的深度（距离）信息，并且没有运动部分。TOF 相机通常包括照射单元、光学透镜、成像传感器、控制单元和计算单元等模块。这种传感器

使用一个受调制的红外光源来确定光子混合器件各像素点的距离，它具有非常紧密的结构。在存在背景光源的时候，图像传感器会接受附加的照明信号，干扰距离的测量。为了消除信号的背景，每关掉照明光完成一次信号采集。TOF 相机会受多重反射、散射光、环境光和温度的影响。图2.1(d) 为基于飞行时间原理的微软 Kinect 二代深度相机，它可以提供 30 FPS 1920×1080 彩色图像和 30 FPS 512×424 深度图像，可测量范围为 0.5m 到 4.5m。

2.2.4 结构光相机

结构光相机是基于三角剖分的测距传感器，它们把已知的光模式或结构光映射到环境，已知模式的反射被接收器接收，并与已知几何值合在一起利用简单的三角剖分建立距离的测量。相对于立体相机，它会不断地投射结构光以至于可以不间断地工作在黑暗无纹理无特征的环境中。著名的第一代 Kinect 深度相机是基于结构光原理的传感器，它于 2010 年推出，并首次用于 Microsoft Xbox 360 视频游戏。但随后打开了机器人学应用中的新可能，它是一款非常廉价的测距传感器，使用一个红外线激光发射器，反射模式肉眼不可见。基于结构光原理的深度相机有智能机器人广泛使用的微软 Kinect 一代（图2.1(e)）和英特尔 realsense 系列（图2.1(f)）。

2.3 系统框架

从一组深度图 τ_D 和彩色图 τ_R 恢复出彩色 3D 点云信息的方法如下：对于深度图 τ_D 中的第 i 个像素点 (u_D^i, v_D^i) ，其深度测量值表示为 $z(u_D^i, v_D^i)$ ，并且对应于 IR 图像的像素点记为 (u_I^i, v_I^i) 。因此 (u_I^i, v_I^i) 像素坐标对应的 3D 空间点可以根据相机小孔成像模型和深度测量值计算得到：

$$\overset{o}{\mathbf{M}}_I = project(\mathbf{K}_I, \mathbf{D}_I, u_I^i, v_I^i) \quad (2.1)$$

$$\mathbf{M}_I = \overset{o}{\mathbf{M}}_I \cdot z(u_I^i, v_I^i) \quad (2.2)$$

其中 \mathbf{K}_I 是 IR 相机的内参， \mathbf{D}_I 是 IR 相机的畸变参数， $\overset{o}{\mathbf{M}}_I$ 是 \mathbf{M}_I 的归一化形式 ($z = 1$)。然后，我们可以根据从 IR 相机坐标系到 RGB 相机坐标系的转换关系 ${}^R_I \mathbf{T}$ 把 IR 坐标系中的点 \mathbf{M}_I 转换到 RGB 坐标系中的点 \mathbf{M}_R ：

$$\mathbf{M}_R = {}^R_I \mathbf{T} \cdot \mathbf{M}_I \quad (2.3)$$

随后，我们重投影空间点 \mathbf{M}_R 到 RGB 图像像素点 (u_R^i, v_R^i) ，并且得到彩色图像 τ_R 中相应的彩色信息 $c(u_R^i, v_R^i)$ ：

$$(u_R^i, v_R^i) = repr(\mathbf{K}_R, \mathbf{D}_R, \mathbf{M}_R) \quad (2.4)$$

其中 \mathbf{K}_R 和 \mathbf{D}_R 分别是 RGB 相机的内参和畸变参数，重投影函数 $repr()$ 是 $project()$ 的逆函数。

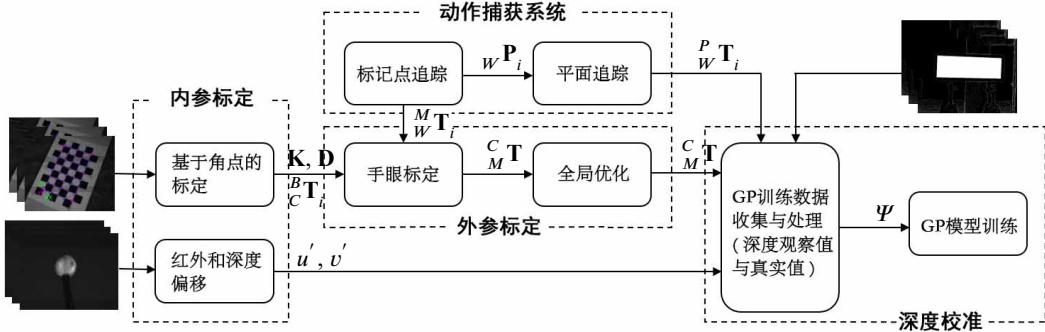


图 2.2 RGB-D 相机标定方法流程图

我们标定方法的目标是使 RGB-D 相机能够从 RGB 和深度图像对中恢复更准确的彩色 3D 环境信息，这意味着更准确的深度测量以及所有彩色 3D 点 ($\mathbf{M}_R, c(u_R^i, v_R^i)$) 的颜色和空间点之间更好的对齐。如图 2.2 所示，我们的工作主要包括三个部分：内参标定、外参标定和深度校准。首先，在 2.4.1 部分中，我们标定了 IR 和 RGB 相机的内在参数，其中包括 $\mathbf{K}_I, \mathbf{D}_I$ 和 $\mathbf{K}_R, \mathbf{D}_R$ ，以及 IR 图像和深度图像中相应像素坐标之间的关系。同时，外参 ${}^R\mathbf{T}_I$ 在 2.4.2 部分中获得。最后在 2.4.3 小节中，我们提供了一种方法来获得更精确的深度测量值 $z(u_D^i, v_D^i)$ 。

2.4 标定过程

2.4.1 内参标定

RGB 和 IR 相机通常都遵循具有径向变形和切向变形的所谓针孔相机模型。也就是说，3D 点 $\mathbf{M} = [X, Y, Z, 1]^T$ 与它的图像投影像素坐标 $\mathbf{m} = [u, v, 1]^T$ 关系如下：

$$s\mathbf{m} = \mathbf{K}[\mathbf{R}, \mathbf{t}]\mathbf{M} \quad (2.5)$$

其中 s 是一个比例因子， \mathbf{R} 和 \mathbf{t} 分别是将其他坐标系与相机坐标系相关联的旋转和平移，并且一起称为相机的外参。本质矩阵 \mathbf{K} 具体形式如下：

$$\mathbf{K} = \begin{bmatrix} f_x & \gamma & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix},$$

其中 f_x 和 f_y 是图像 x 和 y 轴的焦距， γ 是描述两个轴的偏斜度参数， (u_0, v_0) 是光心的像素坐标。此外，实际的相机图像通常会有些变形，主要是径向变形和轻微的切向变形。理想图像坐标 (x_n, y_n) 和真实（失真）图像坐标 (x_t, y_t) 之间的关

系为：

$$\begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} x_n(1 + k_1 r^2 + k_2 r^4) \\ y_n(1 + k_1 r^2 + k_2 r^4) \end{bmatrix} \quad (2.6)$$

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \begin{bmatrix} 2p_1 x_r y_r + p_2(r^2 + 2x_r^2) \\ p_1(r^2 + 2y^2) + 2p_2 x_r y_r \end{bmatrix} \quad (2.7)$$

其中 $r^2 = x_n^2 + y_n^2$, k_1 和 k_2 是径向失真系数, 而 p_1 和 p_2 是切向失真系数。

1. 基于角点的标定和目标提取

单个相机的内参标定是一个经过充分研究的问题。在 Zhang 的方法^[78] 中, 他们不失一般性地假设棋盘格模型平面位于世界坐标系的 $Z = 0$ 上。从收集的图像中提取棋盘角点, 然后使用世界坐标中的已知角点位置和图像中的角点位置为每个图像计算相应的单应性矩阵。相关的线性方程组用于解决每个单应性矩阵所施加的约束, 之后可以通过最小化以下目标函数来估算完整的参数集:

$$\sum_{i=1}^n \sum_{j=1}^m \|\mathbf{m}_{ij} - \check{\mathbf{m}}(\mathbf{K}, k_1, k_2, p_1, p_2, \mathbf{M}_{ij}, {}^B\mathbf{T}_i)\|^2 \quad (2.8)$$

其中函数 $\check{\mathbf{m}}$ 是根据等式2.5在图像 i 中第 j 个点 \mathbf{M}_{ij} 伴随着径向变形 (等式2.6) 和切向变形 (等式2.7) 的投影函数。非线性最小化通过 Levenberg-Marquardt 算法求解。空间坐标转换 ${}^B\mathbf{T}_i$ (从相机坐标系 C 到棋盘坐标系 B 的空间转换, 包含 Rodrigues 格式的平移和旋转部分) 的初始值和内参 \mathbf{K} 是使用先前的估计获得的。失真参数 k_1, k_2, p_1, p_2 用 Zhang 的方法^[78] 中描述的技术初始化, 或者只是简单设为 0。通过执行此标定, 我们不仅可以获取相机内参, 还可以获取棋盘格坐标系中相机的所有姿态 ${}^B\mathbf{T}_i$, 可用于下一节所述的外参标定。

2. 红外图像和深度图像之间的偏移

如文献^[79] 中所述, IR 图像和深度图像中的相应像素坐标之间存在固定的偏移量。也就是说,

$$\mathbf{m}_I = \mathbf{m}_D + [u' \ v' \ 0]^T \quad (2.9)$$

为了估算该值, 我们将相机朝向标定工具的反光标记点, 如图2.3(a)或图2.3(b)所示, 并使反光标记点保持静止。然后, 记录 180 对 IR 和深度图像。对于每组图像对, 我们首先使用 Canny 边缘检测^[80], 然后使用 Hough 圆检测^[81] 来检测 IR 图像 (一些实际效果如图2.4(a), 2.4(b), 2.4(c)) 和深度图像 (如图2.4(d), 2.4(e), 2.4(f)) 中反光标记点的中心点。由于 Kinect 的深度图像非常嘈杂, 因此我们几乎无法获得 u', v' 的准确结果。我们在实验中获得的 u', v' 平均估计值约为 (4, 4)。

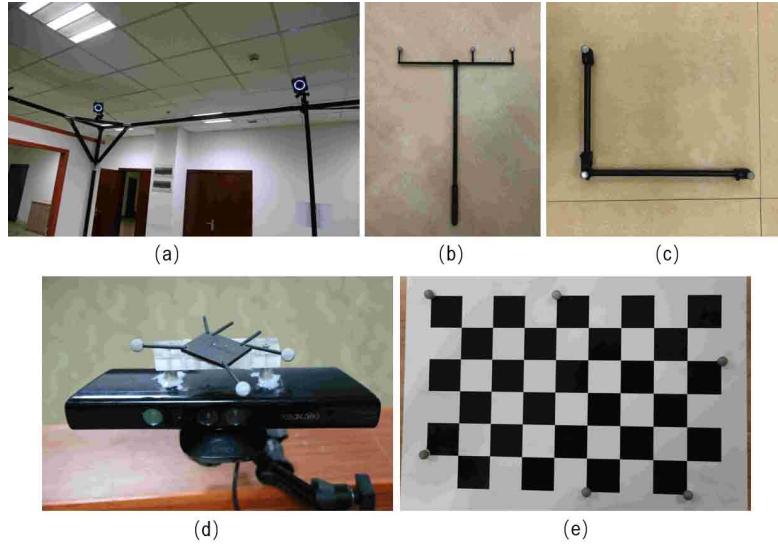


图 2.3 (a) 动作捕获系统; (b) 和 (c) 是动作捕获系统的标定工具; (d) 是粘有三个反光标记点的 Kinect 设备; (e) 用于内参和外参标定的棋盘格, 上面贴有为了评估标定效果的六个反光标记点

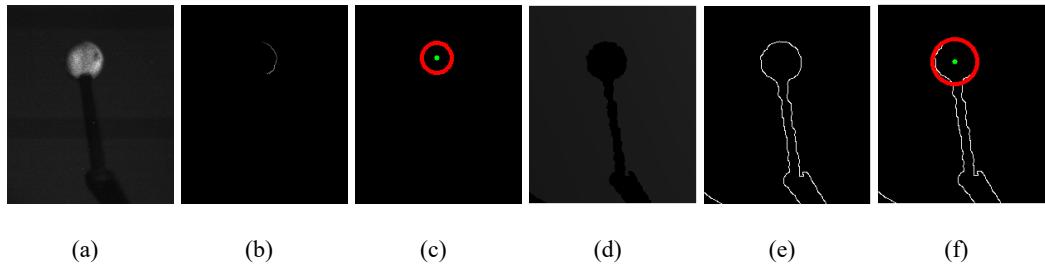


图 2.4 不同标定处理后的 IR(a-c) 和深度图像 (d-f): (a) 原始的 IR 图像; (b) 经过 Canny 边缘检测处理后的 IR 图像; (c) Hough 圆检测处理后的 IR 图像; (d) 原始的深度图像; (e) 经过 Canny 边缘检测处理后的深度图像; (f) Hough 圆检测处理后的深度图像

2.4.2 外参标定

在内参标定过程中, 我们将相机移动到保持静止不动的棋盘格图案前面并不断移动相机。如图2.5所示, 相机光学中心的坐标系从 C_1 移动到 C_2 , 同时贴在相机上的反光标记坐标系从 M_1 相应地转换为 M_2 。所以我们有:

$$\frac{B}{W} \mathbf{T} = \frac{C_1}{B} \mathbf{T}^{-1} \cdot \frac{C_1}{M_1} \mathbf{T} \cdot \frac{M_1}{W} \mathbf{T} = \frac{C_2}{B} \mathbf{T}^{-1} \cdot \frac{C_2}{M_2} \mathbf{T} \cdot \frac{M_2}{W} \mathbf{T},$$

由于反光标记点牢固地固定在相机上, 我们有 $\frac{M_1}{C_1} \mathbf{T} = \frac{M_2}{C_2} \mathbf{T} = \frac{M}{C} \mathbf{T}$, 因此上面的公式可以简化为:

$$\frac{C_2}{C_1} \mathbf{T} \cdot \frac{C}{M} \mathbf{T} = \frac{C}{M} \mathbf{T} \cdot \frac{M_2}{M_1} \mathbf{T} \quad (2.10)$$

其中 $\frac{C_2}{C_1} \mathbf{T} = \frac{C_2}{B} \mathbf{T} \cdot \frac{C_1}{B} \mathbf{T}^{-1}$, 并且它们的值可以通过第2.4.1节的第1小节提到的方法计算得到。 $\frac{M_2}{M_1} \mathbf{T} = \frac{M_2}{W} \mathbf{T} \cdot \frac{M_1}{W} \mathbf{T}^{-1}$, 并且它们的值可以通过动作捕获系统测量得到。

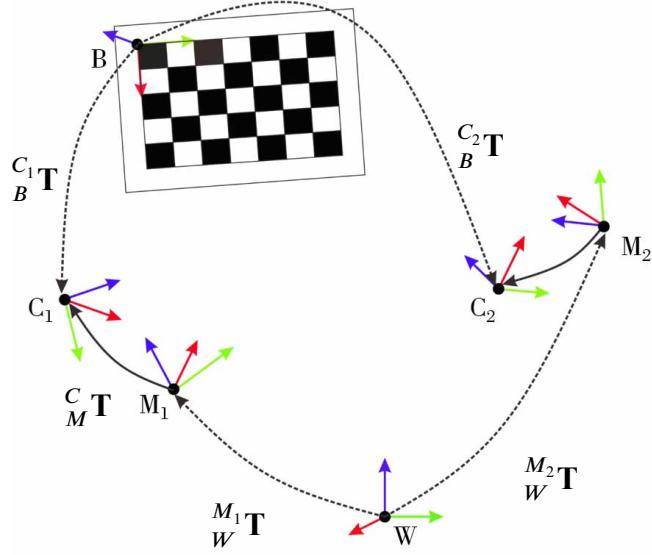


图 2.5 内参和外参标定场景示意图。摄像头坐标系从 C_1 移至 C_2 ，与摄像头刚性相连的标记点坐标系相应地从 M_1 移至 M_2 。 W 是 MoCap 系统的世界坐标系， B 坐标系固定在棋盘格图案上。在标定期间，我们保持棋盘格不动。

1. 手眼标定

显然，等式2.10是著名的手眼标定问题，它等效于解决等式 $\mathbf{AX} = \mathbf{XB}$ ，其中 \mathbf{A} , \mathbf{B} 和 \mathbf{X} 都是 4×4 空间转换矩阵。方程2.10可以进一步分解为两个方程：取决于旋转 \mathbf{R} 和平移 \mathbf{t} 的旋转方程和位移方程：

$$\mathbf{R}_A \mathbf{R}_X = \mathbf{R}_X \mathbf{R}_B \quad (2.11)$$

$$(\mathbf{R}_A - \mathbf{I})\mathbf{t}_X = \mathbf{R}_X \mathbf{t}_B - \mathbf{t}_A \quad (2.12)$$

其中 \mathbf{I} 是 3×3 单位矩阵。

根据文献^[82] 中提到的数学技巧，我们可以再把上式改写成线性方程：

$$\begin{bmatrix} \mathbf{I}_9 - \mathbf{R}_A \otimes \mathbf{R}_B & \mathbf{0}_{9 \times 3} \\ \mathbf{I}_3 \otimes \mathbf{t}_B^T & \mathbf{I}_3 - \mathbf{R}_A \end{bmatrix} \begin{bmatrix} \text{vec}(\mathbf{R}_X) \\ \mathbf{t}_X \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{9 \times 1} \\ \mathbf{t}_A \end{bmatrix} \quad (2.13)$$

其中，涉及 Kronecker 乘法 \otimes 和列向量操作符 vec 定义如下：

$$\mathbf{M} \otimes \mathbf{N} = \begin{bmatrix} M_{11}\mathbf{N} & \cdots & M_{1n}\mathbf{N} \\ \vdots & \ddots & \vdots \\ M_{m1}\mathbf{N} & \cdots & M_{mn}\mathbf{N} \end{bmatrix},$$

$$\text{vec}(\mathbf{M}) = [M_{11}, \dots, M_{1n}, \dots, M_{21}, \dots, M_{mn}]^T.$$

也就是求解对应的线性方程：

$$\mathbf{C}_i \mathbf{x} = \mathbf{d}_i \quad (2.14)$$

我们可以采集多组 \mathbf{C}_i 与 \mathbf{d}_i , 构建误差函数 $f_i(\mathbf{x}) = \|\mathbf{C}_i \mathbf{x} - \mathbf{d}_i\|_2$, 并采用一般的凸优化方法优化出 \mathbf{x} 。我们通过使用 CVXPY^[83] 等相关开源项目实际解决该凸优化问题。

2. 全局优化

手眼标定精度受到限制主要是因为假定的姿态轨迹是单独估算的, 并且在对齐它们以找到手眼变换时保持固定。因此, 在本小节中, 鉴于动作捕获系统测量结果具有很高的精度, 我们提出了标定和轨迹的联合最大似然优化。这种优化很难作为一个整体问题解决, 但是使用第1部分的结果作为初始猜测, 局部似然最大化可以提高标定的准确性。我们使用李群值 B 样条^[84] 表示连续时间 SO(3) 轨迹, 并使用 Levenberg-Marquardt 算法在李群的扩展来求解非线性优化^[85]。

但是在我们的标定场景中, 动作捕获系统测量的转换矩阵 ${}^W_W \mathbf{T}$ 的值非常准确, 并且标定误差的主要来源是转换矩阵 ${}^B_B \mathbf{T}$ 的估计值。因此, 本文中我们为移动标记点坐标系 M 的移动轨迹建模为 ${}^M_W \mathbf{T}(t) =: \mathbf{X}(t)$ 。并且文献^[85] 中提到的负对数似然函数 l 修改为:

$$\begin{aligned} l &= l_1(\mathbf{X}|({}^{M_i}_W \mathbf{T}, t_{M_i})_{i=1}^I) + l_2({}^C_B \mathbf{T}, {}^C_M \mathbf{T}|({}^{C_i}_B \mathbf{T}, t_{C_i})_{i=1}^K) \\ &= \sum_{i=2}^{k_M} \rho(||d_M({}^{M_{i-1}}_W \mathbf{T}, {}^{M_i}_W \mathbf{T}, {}^M_W \mathbf{T}(t_{M_{i-1}}), {}^M_W \mathbf{T}(t_{M_i}))||_{\Sigma_M}^2) \\ &\quad + \sum_{i=2}^{k_C} \rho(||d_C({}^{C_{i-1}}_B \mathbf{T}, {}^{C_i}_B \mathbf{T}, {}^C_B \mathbf{T}(t_{C_{i-1}}), {}^C_B \mathbf{T}(t_{C_i}))||_{\Sigma_C}^2) \end{aligned} \quad (2.15)$$

其中 ${}^C_B \mathbf{T}(t) := {}^W_B \mathbf{T} \cdot {}^M_W \mathbf{T}(t) \cdot {}^C_M \mathbf{T}$, ${}^M_W \mathbf{T}(t) = \mathbf{X}(t)$, $\rho(s) = \log(1+s)$, d_M 和 d_C 是相对的关系, $d(\mathbf{A}, \mathbf{A}', \mathbf{B}, \mathbf{B}') = d(\mathbf{A}^{-1} \mathbf{A}', \mathbf{B}^{-1} \mathbf{B}')$ 。作为 SE(3) 上位移矢量 $d(\mathbf{A}, \mathbf{B}) \in \mathbb{R}^6$, 我们使用基于固定正交基础的 $(\log_{SO(3)}(\mathbf{R}), \mathbf{u})$ 坐标, 其中 \mathbf{u} 是平移值, 而 \mathbf{R} 是适当的旋转值, 使得 (唯一) $\mathbf{u} \circ \mathbf{R} := \mathbf{A}^{-1} \cdot \mathbf{B}$ 。我们首先优化 l_1 来估算标记点坐标系轨迹函数 $\mathbf{X}(t)$ 。然后优化 l_2 以计算 ${}^C_M \mathbf{T}$ 的最佳估计值。

2.4.3 深度校准

许多深度传感器具有近视特性, 这意味着错误的参数设置会导致深度测量误差随距离增加而增加。更糟糕的是, 在不同像素位置的深度测量误差也不同 (如图 2.6 所示)。为了对这种失真的影响进行建模, 在最新的工作^[50,71] 中, 作者将误差分为两个不同的部分并分别进行处理, 包括失真误差 (该误差是导致观测物体形状局部更改的原因) 和全局误差 (即平均深度的系统估计误差)。也就是说, 实际深度 d^* 估计为:

$$d^* = (g \odot u)_{(u,v)}(d),$$

其中函数 $u(\cdot)$ 考虑了局部失真, 函数 $g(\cdot)$ 对深度量度进行了全局校准。

在这项工作中，2.4.2小节中提到的方法用于实时获取IR相机的姿态。在本节中，首先我们将展示如何追踪平面以获取深度测量的真实值。然后我们提出了一种稀疏的高斯过程方法来对后验概率 $P(d^*|d)$ 建模，也就是位于不同距离 d 的深度测量值 d^* 的均值 $\mu(d^*)$ 和方差 $\sigma^2(d^*)$ 。

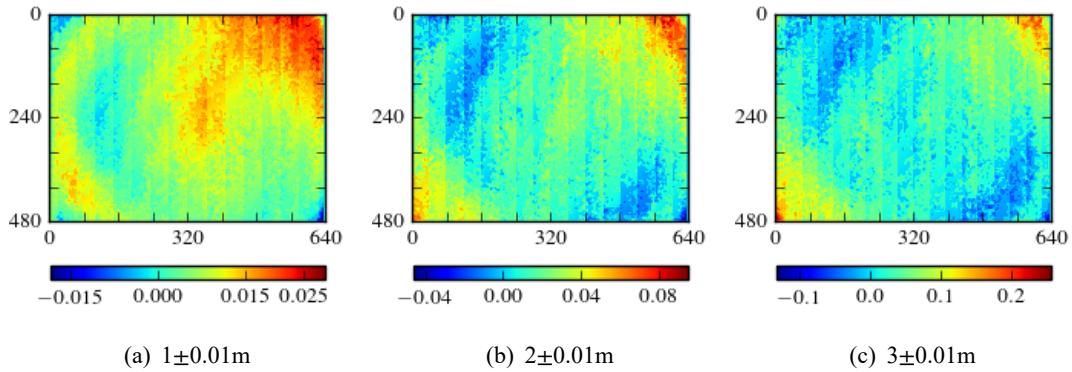


图 2.6 不同距离的深度误差图，请注意每个图像的色标都不同。

1. 平面追踪

首先，我们要追踪平面坐标系 P ，这意味着实时获取转换矩阵 ${}^P_W \mathbf{T}$ 。我们在平面板上贴上三个反射标记 A , B 和 C ，或使用如图 2.3(c) 中所示的官方标定工具。MoCap 系统可实时为我们提供三个标记的三维坐标： $P_A = [x_A, y_A, z_A]^T$, $P_B = [x_B, y_B, z_B]^T$ 和 $P_C = [x_C, y_C, z_C]^T$ 。下面阐述如何根据这些反光标记点计算从世界坐标系 W 到由这些标记点形成的平面坐标系 P 的转换矩阵。

第一步，我们不失一般性地选择 \overrightarrow{AB} 的单位方向向量 \vec{x} 作为坐标系 P 的 X 轴的方向。我们还可以通过使用几何属性 $\vec{z} \perp \overrightarrow{AB}$ 和 $\vec{z} \perp \overrightarrow{AC}$ 找到平面单位法向量 \vec{z} (即 Z 轴的方向)。 Y 轴的方向可以估计为 $\vec{y} = \vec{z} \times \vec{x}$ 。如果没有 MoCap 系统的测量误差和标记点的放置偏差，则可以轻松获得从世界坐标系 W 到平面坐标系 P 的 3×3 旋转矩阵 \mathbf{R} ：

$$\mathbf{R} = (\vec{x}, \vec{y}, \vec{z}) \quad (2.16)$$

但是实际的三个方向向量并不绝对相互垂直。因此奇异值分解(SVD) $\mathbf{R} = \mathbf{U} \Sigma \mathbf{V}^T$ 用于获得旋转矩阵 \mathbf{R} 的近似估计 \mathbf{R}' (详细的证明可以在文献^[78]中找到)。也就是

$$\mathbf{R}' = \mathbf{U} \cdot \mathbf{V}^T \quad (2.17)$$

最后，从坐标系 W 到 P 的 4×4 的转换矩阵为：

$${}^P_W \mathbf{T} = \begin{pmatrix} \mathbf{R}' & P_A \\ \mathbf{0} & 1 \end{pmatrix} \quad (2.18)$$

同时为了过滤投影到平面外的深度数据，我们还需要提供平面范围

$(x_{min}, x_{max}, y_{min}, y_{max})$, 这些值不需要非常精确, 只要保证在实际平面范围内即可。

算法 2.1 深度数据获取

Input: ${}^P_W\mathbf{T}$, ${}^M_W\mathbf{T}$, 测量的深度图 \mathbf{D} ;

Output: 用于训练 GP 模型的数据集 Ψ ;

```

1  ${}^I_M\mathbf{T} = hand\_eye\_calibration();$ 
2 初始化平面范围  $(x_{min}, x_{max}, y_{min}, y_{max})$ ;
3 初始化 IR 图像和深度图的偏移  $u', v'$ ;
4 for  $i \in range(\mathbf{D})$  do
5    ${}^P_I\mathbf{T}_i = {}^P_W\mathbf{T}_i \cdot {}^M_W\mathbf{T}_i^{-1} \cdot {}^I_M\mathbf{T}^{-1};$ 
6    $\mathbf{P}_i = plane({}^P_I\mathbf{T}_i);$ 
7   for  $pixel(u, v) \in \mathbf{D}_i.size()$  do
8      $u_I = u + u';$ 
9      $v_I = v + v';$ 
10     $\mathbf{M}_I = project(u_I, v_I);$ 
11     $\mathbf{M}'_I = cross(\mathbf{M}_I, \mathbf{P}_i);$ 
12     $\mathbf{M}'_P = {}^P_I\mathbf{T} \cdot \mathbf{M}'_I;$ 
13    if  $(\mathbf{M}'_P.x \in [x_{min}, x_{max}] \wedge \mathbf{M}'_P.y \in [y_{min}, y_{max}])$  then
14       $\Psi = \Psi \cup \{(u, v, \mathbf{D}_i(u, v), \mathbf{M}'_P.z)\}$ 
15    end
16  end
17 end
18 return  $\Psi$ ;
```

2. 深度数据获取

上一节中提到的平面可以选择为不能移动的较大平面 (如墙), 也可以选择为可以自由移动的较小平面 (如图 2.3(e) 中所示)。这是因为我们可以使用 2.4.2 和上一小节中提到的方法实时追踪平面和相机的 3D 姿态。很明显, 更大的平面可以提高数据采集的效率。

我们首先记录 Kinect 在离平面不同距离处的深度图像, 以及相应的 ${}^P_W\mathbf{T}$ 和 ${}^M_W\mathbf{T}$, 然后离线处理记录的数据以生成用于训练 GP 模型的数据集。离线处理过程在算法 2.1 中进行了简要描述。首先, 我们使用 ${}^P_I\mathbf{T}$ 恢复平面方程 \mathbf{P} (l. 6, 其中 $plane()$ 是获得 XY 轴平面方程的函数)。对于深度图中给定的像素坐标 $\mathbf{m} = (u, v)$, 我们可以使用公式 2.9 (ll. 8-9) 获得 IR 图像中的相应像素坐标。然后使用公式 2.5、2.6 和 2.7 (l. 10) 计算对应的 3D 点 $\mathbf{M} = (x, y, 1)$, 这意味着一条从光心射向 \mathbf{M} 的

光束。我们可以找到射线与平面 P 的交点 M' (I. 11)。最后，我们使用平面范围 $(x_{min}, x_{max}, y_{min}, y_{max})$ 来确定交点是否落在平面中，并决定是否将其保留在数据集 Ψ 中 (II. 12-15) (如图 2.7 中所示，图中三维交点 a 和 e 超出了平面范围，应该被滤除)。

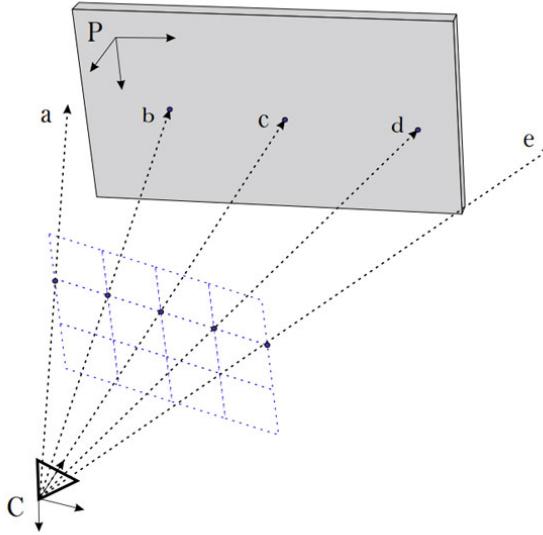


图 2.7 摄像头图像像素点投影到追踪平面 P 的示意图

3. 异方差高斯过程

高斯过程 (Gaussian Process, GP) 是对函数的正态分布的概括，描述了有限维随机变量的函数。与文献^[71] 中使用的曲线拟合方法相比，高斯过程不需要事先知道模型的具体形式。稀疏高斯过程可以降低高斯过程求解代价，核函数模型 (BFM) 是其中一类稀疏高斯过程求解方法，BFM 的基本假设为：给定一组输入 $\mathbf{X} = \{x_i\}_{i=1}^n \in \mathbb{R}^{n \times d}$ 和一组目标输出 $y = \{y_i\}_{i=1}^n \in \mathbb{R}^n$ ，观测目标 y_i 由 m 个非线性函数 $\phi(x_i) = [\phi_1(x_i), \dots, \phi_m(x_i)] \in \mathbb{R}^m$ 的线性组合函数的输出加上噪声 $\epsilon_i \sim \mathcal{N}(0, \beta^{-1})$ 生成：

$$y_i = \phi(x_i)\omega + \epsilon_i,$$

其中 ω 是长为 m 的参数向量，即模型的参数。这里假设我们的观测具有恒定精度 β 和均值为零的噪声。显然不符合此工作中遇到的方差随输入变化的特征。为了将此信息整合到系统中，开源代码项目 GPz^[77] 通过将噪声方差建模为与输入相关的函数 $\beta_i = f(x_i)$ 来增强模型的预测方差估计。具体而言，他们把这个函数定义为核函数的线性组合 $\beta_i = \exp(\phi(x_i)u + b)$ ，指数形式来保证 β_i 的正数性。注意，如果 $u = 0$ 且 $b = \ln\beta$ ，模型将退化为固定精度 β 的原始假设。模型的参数可以根据 GPz 论文^[77] 中定义的对数边际似然进行优化。一旦参数被推断，不可

见测试用例 x_* 的预测分布如下：

$$\begin{aligned} p(y_*|y) &= \mathcal{N}(\mu_*, \sigma_*^2), \\ \mu_* &= \phi(x_*)\bar{\omega}, \\ \sigma_*^2 &= v_* + \beta_*^{-1}, \\ v_* &= \phi(x_*)\Sigma^{-1}\phi(x_*)^T, \\ \beta_* &= \exp(\phi(x_*)u + b). \end{aligned}$$

其中，模型方差 v_* 取决于 x_* 处训练样本的数据密度，理论上，随着数据集的增加，模型方差的这一组成部分将趋于零。因此， v_* 建模了我们对均值函数的潜在不确定性。当在 x_* 处呈现大量样本时，模型对后验均值变得非常有信心。 β_*^{-1} 是与输入相关的噪声不确定性。

在该标定工作的深度校准中，我们使用开源代码项目 GPz^[77] 为异方差后验概率 $P(d^*|d)$ 训练一组稀疏的高斯过程。为了说明标准 GP 与异方差 GP 之间的差异，我们展示了具有方差 $v = (x + 1)/15$ 的示例函数 $y = \sin(x)$ 产生的预测。图2.8显示了使用异方差 GP 成功预测了示例函数的均值和方差。

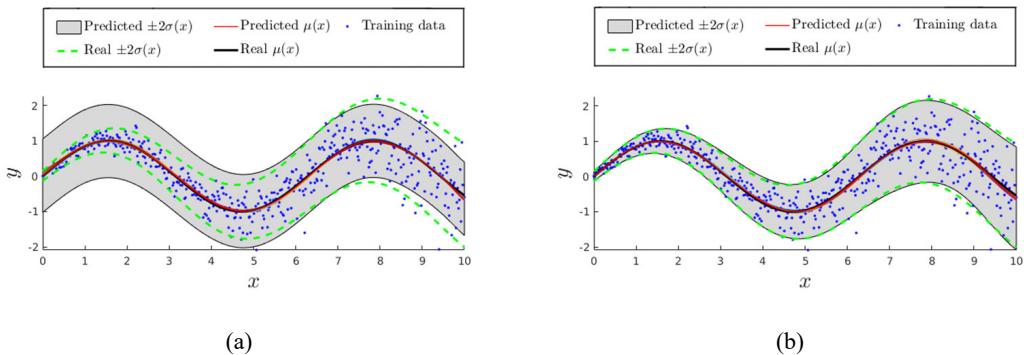


图 2.8 针对具有方差 $v = (x + 1)/15$ 的示例函数 $y = \sin(x)$ 生成的预测，(a) 为标准 GP，(b) 为异方差 GP。

2.5 实验及结果分析

在本节中，我们将证明我们的方法能够为类似 Kinect 的传感器提供更准确的标定结果。我们独立分析我们的手眼标定（第2.4.2节）和深度校准（第2.4.3节）的结果。然后，我们报告了将 RGB-D 视觉 SLAM 应用于安装了 Kinect 设备的移动机器人的一些实际实验，这些实验表明，通过使用我们的标定方法校准的 RGB-D 数据，对定位和建图的准确性提高非常有用。最后展示了与其他相关工作的性能比较。

2.5.1 手眼标定测试

实际情况下准确评估手眼标定结果的准确性是十分困难的，因此在该项测试中，我们通过重投影误差来评估内参和外参标定的整体效果。我们在棋盘格的外角上尽可能准确地放置了六个反光标记点（如图 2.3(e) 所示），这样我们就知道了棋盘格坐标系和标记点坐标系之间的转换关系。给定这些点的观测值和相应的点模型，我们可以根据迭代最近点法（ICP）计算其相对于动作捕获系统世界坐标系的位姿 ${}^W \mathbf{T}'$ 。对于每个图像 i 中每个角点 j 的 3D 位置 \mathbf{M}_{ij} ，我们可以在相机坐标系中计算其对应的坐标，并比较与检测到的角点像素坐标 \mathbf{m}_{ij} 的偏差。也就是

$$e_{ij} = \text{repr}({}^C \mathbf{T} \cdot {}^W \mathbf{T}_i \cdot {}^W \mathbf{T}' \cdot \mathbf{M}_{ij}) - \mathbf{m}_{ij}.$$

我们记录了 44 张 8x5 的棋盘图像，共有 1760 个角点。图 2.9(a) 绘制了在外参标定的第1部分中估计的 ${}^C \mathbf{T}$ 值对应的角点重投影误差 e_{ij} ，图 2.9(b) 绘制了经过外参标定的第2部分中的全局优化方法处理后对应的角点重投影误差 e_{ij} 。图 2.9(c) 显示了重投影误差 $|e_{ij}|$ 的分布。图 2.9 中的所有子图都证明了外参标定第2小节中提到的全局优化步骤可以对外参进行更准确的估计。需要说明的另一点是，这里的重投影错误不仅是由于对 ${}^C \mathbf{T}$ 的不正确估计引起的，而且还包括粘贴标记点的位置不正确、角点检测不准确以及 MoCap 系统的测量误差等因素。我们方法的重投影误差在 2 个像素以内，我们认为深度校准的精度是可以接受的。在图 2.10 中还提供了一些定性分析，这些分析表明，使用我们的 ${}^R \mathbf{T}$ 估计值可以使 RGB 和深度相机之间更好对齐，从而使彩色和深度信息更准确地融合在一起。

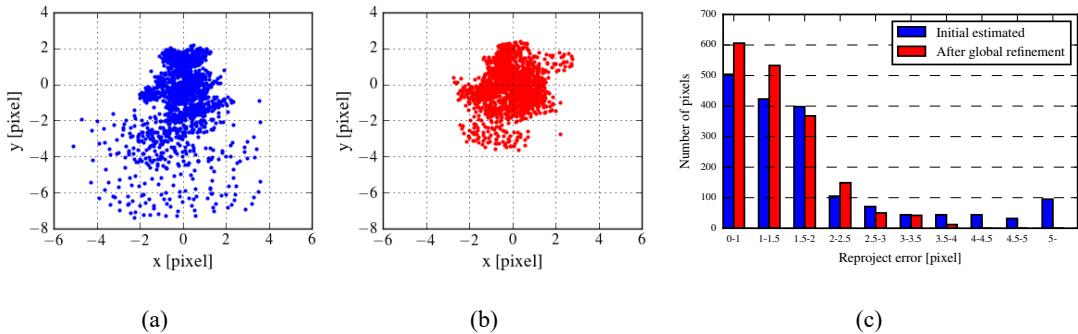


图 2.9 所有测试图像中所有像素的重投影误差：(a) 是没有全局优化的误差；(b) 是经过全局优化后的最终结果；(c) 重投影误差的分布。

2.5.2 深度校准测试

为了验证我们提出的深度校准方法的性能，我们收集了两个数据集（包括深度和彩色图像对、 ${}^M \mathbf{T}_i$ 和 ${}^P \mathbf{T}_i$ ）：训练集（包含 597 个图像对，用于生成校准函数）和测试集（包含 428 个图像对，用于评估深度校准精度）。在训练集上部

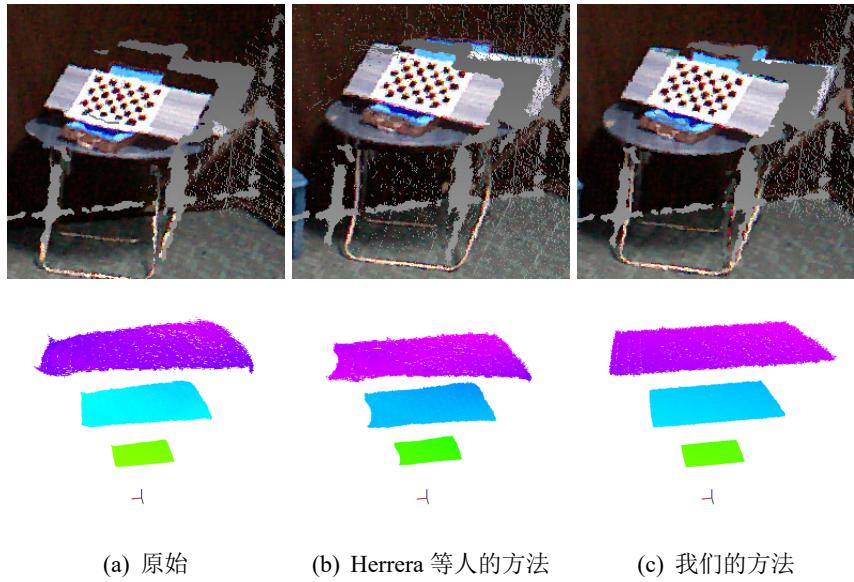


图 2.10 不同标定方法的彩色图和深度图对齐效果和点云校准效果：第一行：彩色点云是由 RGB 相机和深度相机之间的不同对齐导致的；第二行：使用不同深度校准方法的三组平面点云（距离平面分别为 1m, 2m, 3m）。

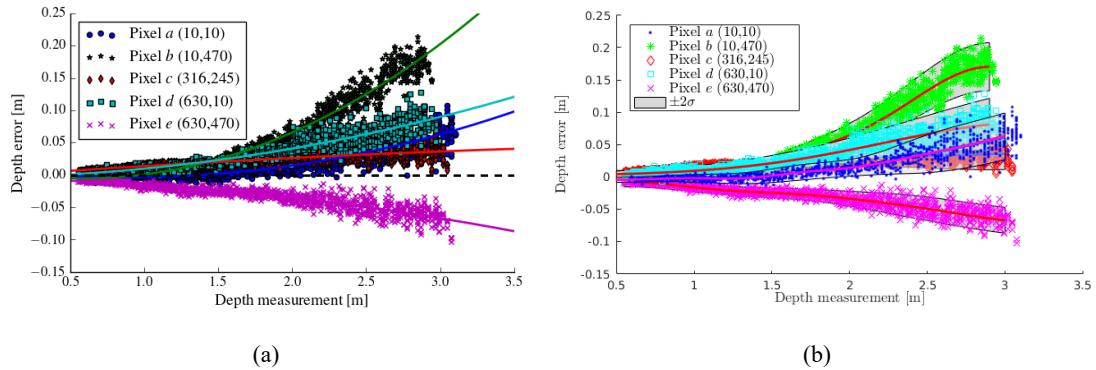


图 2.11 五个不同像素坐标（像素 c 是光心的坐标）的深度误差和相应的多项式拟合曲线（a）或 GP 模型（b）。注意不同的像素坐标具有不同的拟合曲线或 GP 模型参数。

署深度校准算法后，我们获得了 480x640 个深度校准 GP 模型。为了与最新的深度校准方法^[71]进行比较，假设深度误差是通过二次多项式校准，我们还获得了 480x640 个函数，该函数总共包含 480x640x3 个参数（在图 2.11 中举例说明了一些示例）。然后，我们使用这些模型或函数来校准测试数据集中的深度图像。总的深度误差绘制在图 2.12 中，其中图 2.12(a) 是原始深度误差，图 2.12(b) 是采用曲线拟合方法^[71]校准后的深度误差，图 2.12(d) 是使用我们 GP 模型提供的平均值校准的深度误差。由我们的 GP 方法获得的深度误差的标准偏差也绘制在图 2.12(c) 中。我们可以显然地发现我们的方法显著地减少了 Kinect 深度测量的误差，并且优于曲线拟合方法。在图 2.10 中，我们选择了三组点云，它们的测量距离大约为 1m、2m 和 3m，且它们的表面保持平坦。我们发现，与使用 Herrera 的

方法^[65]相比，经我们的方法校准后三组点云变得更加准确。图2.13分别是其俯视图和正视图，进一步显示了使用我们提出的深度校准算法得到的更好效果。

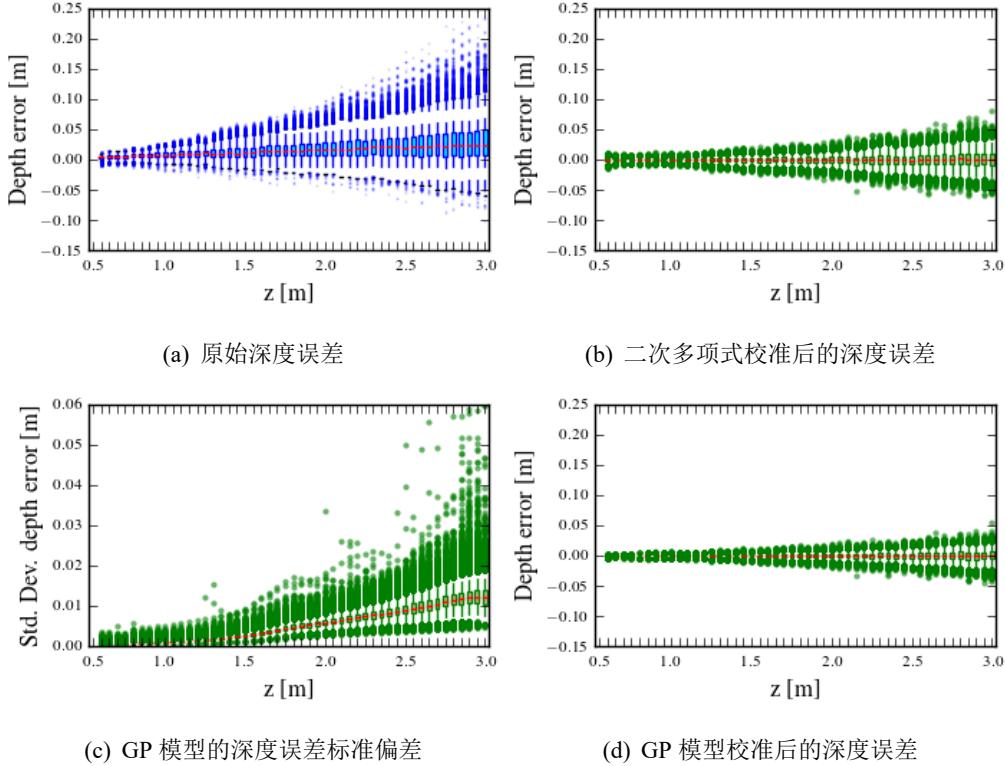


图2.12 深度校准前后的深度误差统计

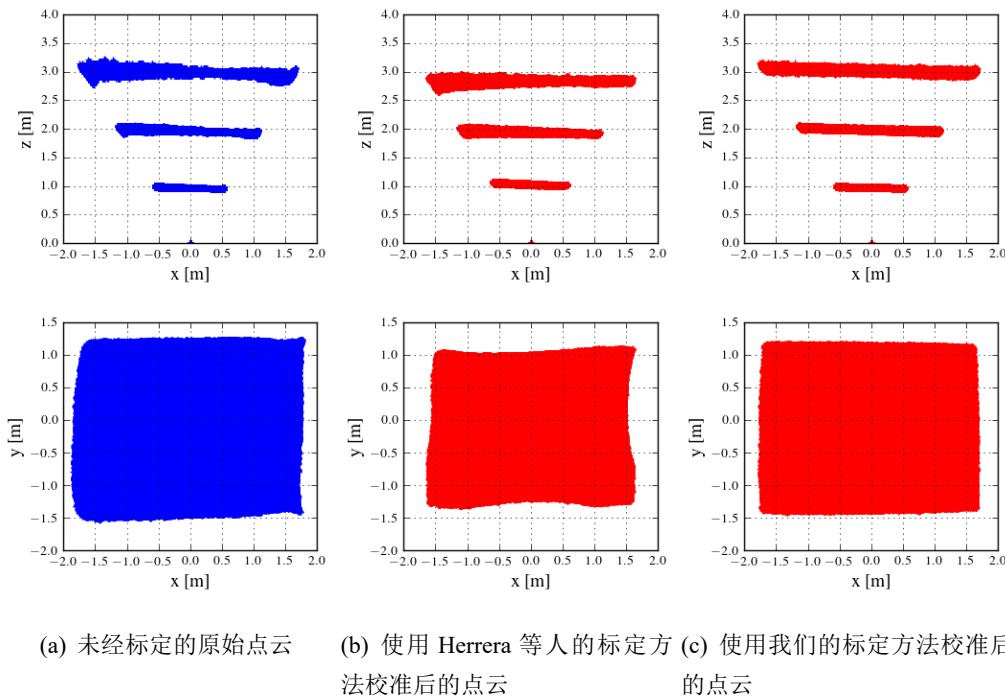


图2.13 不同视角下的示例点云。第一行：距离大约为1m, 2m 和 3m 的三个点云侧视图；第二行：距离3m 的点云的正视图。

2.5.3 视觉 SLAM 实验

为了进一步验证我们方法的有效性，我们展示了使用我们的家庭服务机器人（Kejia F2^[86]）进行的 RGB-D 视觉 SLAM 实验（RGB-D SLAM v2^[59]）。并且，我们使用评估指标定量评估了通过我们的方法校准的 RGB-D 数据进行定位的准确性，定位评估指标包括绝对轨迹误差（ATE）和相对姿态误差（RPE）^[72]，最终建立的 3D 点云图也做了具体展示。

我们以 0.02 m/s 的速度将移动机器人向前移动 2m，然后旋转 90 度并再向前移动 3m。同时，我们使用手眼标定结果 ${}^C_M \mathbf{T}$ 记录附着在 Kinect 上的标记点位姿，并且我们将其视为 Kinect 轨迹的真实值。表2.1展示了实验中 ATE 和 RPE 的均方根误差（RMSE）：ATE 的降低非常明显，RPE 也略有降低。图 2.14(a) 展示了在有无使用我们的标定方法的情况下估算轨迹的和真实轨迹的俯视图，以及生成的点云图（图 2.14(b)）。与使用原始数据估计的轨迹相比，使用标定数据估计的轨迹显然更接近真实轨迹，并且通过使用标定数据也可以提高重构点云图的质量和精度。

表 2.1 视觉 SLAM 定位误差 ATE 和 RPE 的均方根误差（RMSE）值

	原始数据	标定处理后的数据
ATE [m]	0.092	0.041
RPE.translation [m]	0.0082	0.0073
RPE.rotation [deg]	0.3576	0.3517

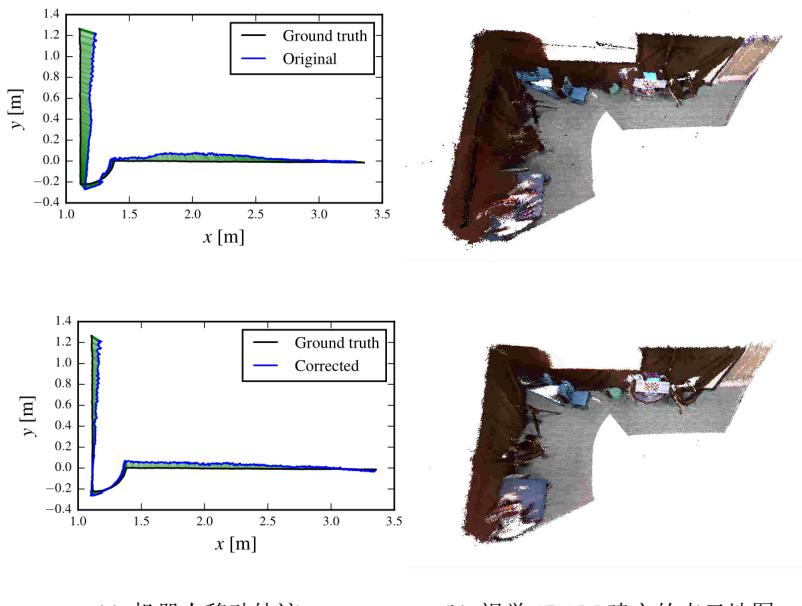


图 2.14 RGB-D SLAM v2 实验结果

2.5.4 对比实验

在本小节中，我们将我们提出的标定方法的校准精度与两种最先进的标定方法进行了比较，一种来自 Basso 等人的标定方法^[71]，另一种来自 Herrera 等人的标定方法^[65]，工程上均使用作者提供的原始实现。对于每种方法，我们都收集大量训练集分别根据具体标定步骤来对同一个 Kinect 深度相机进行标定。为了比较校准精度，我们收集了一个测试数据集，该测试数据集依靠一个大的空心立方体（图 2.15(a)），其每个可见面均附有 AprilTags^[87] 二维码。我们可以使用

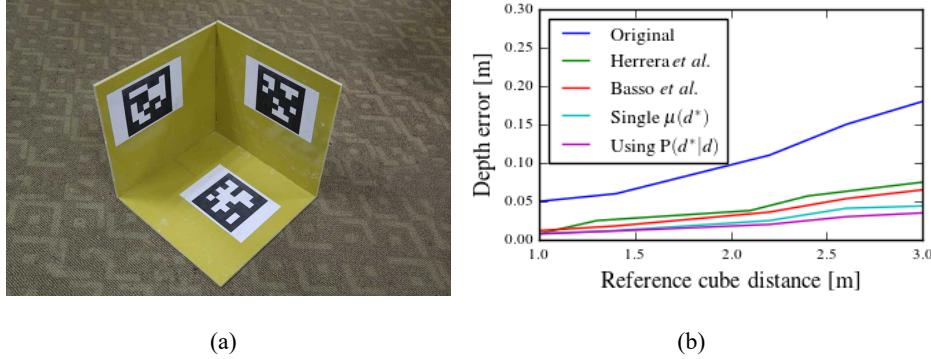


图 2.15 (a) 在性能评估中用作真实值的空心参照立方体；(b) 不同标定方法精度比较。

获得的 AprilTag 彩色图像来精确地计算三个侧面的平面方程，这些平面的交点 x 作为深度测量真实值。然后我们使用三种不同深度校准方法校准三个平面收集的点云，并计算它们的交点 x' ，即为深度校准后的测量值。为了进一步证明概率分布在状态估计问题中的作用，我们在每个相同的位置记录了三个深度图。真实深度 d^* 的概率分布为：

$$P(d^*|d_1, d_2, d_3) = P(d^*|d_1) \cdot P(d^*|d_2) \cdot P(d^*|d_3),$$

其中 d_1 , d_2 和 d_3 是三个深度图中交点 x 的深度测量。由于所有概率分布都近似于高斯分布，因此我们有：

$$P(d^*|d_i) \cdot P(d^*|d_j) = \mathcal{N} \left(\frac{\sigma^2(d_i) \cdot \mu(d_j) + \sigma^2(d_j) \cdot \mu(d_i)}{\sigma^2(d_i) + \sigma^2(d_j)}, \frac{\sigma^2(d_i) \cdot \sigma^2(d_j)}{\sigma^2(d_i) + \sigma^2(d_j)} \right),$$

其中 $\mu(d)$ 和 $\sigma^2(d)$ 是均值和方差值，可以从我们的 GP 模型中获得。图 2.15(b)显示了在空心立方体的不同距离处不同方法的深度误差对比。我们可以看出不同标定方法均可以大幅度降低深度相机的测量误差，但我们的方法比 Herrera 等人和 Basso 等人的标定方法更好，进一步降低了整体标定误差。同时使用我们的深度校准 GP 模型可以获得深度误差均值和方差值，这可以在多帧深度测量基础上进一步获取更加准确的深度估计。

2.6 本章小结

在本章中，我们完善了通用批量标定框架，提出了一种使用动作捕获系统标定 RGB-D 传感器的可靠准确方法。所提出的标定程序不仅可以提供每个相机的内部和外部参数，还可以基于异方差高斯过程生成一组像素级深度校准模型。在内参标定中，我们完全标定了深度图像和 IR 图像之间的像素偏移。在我们的外参标定中，我们充分利用动作捕获系统的特性来定制手眼标定的全局优化步骤，进一步提高了外参标定的准确性。此外，采用一种非递归的新颖的数据获取方法来获取深度图像中每个像素的真实值，并采用一种无模型的基于异方差高斯过程的单步深度校准方法来获取深度校准模型集的参数。所提出的校准方法比以前的相关工作获得了更好的结果，并且整个标定结果大大改善了基于 Kinect 传感器的视觉 SLAM 的定位和建图精度。

第3章 基于栅格地图信息融合的多机器人导航

本章主要介绍基于栅格地图信息融合的多机器人避障技术，该项工作旨在说明基于栅格地图信息输入的强化学习避障算法的优势，且该多机器人避障算法是我们在下一章提出的基于深度强化学习的动态行人环境下避障算法基础，同时此多机器人避障算法本身不需要复杂前端感知处理且具有实际应用价值，可以应用于多机器人无人仓储等实际场景。我们首先介绍了多机器人避障技术的研究背景和现状，然后我们回顾了两种传统的机器人避障算法：动态窗口法和速度障碍物法，详细阐述了它们的原理与缺陷。随后我们用数学方法和符号定义了多机器人避障问题。在第3.4节中，我们详细介绍了我们提出的基于深度强化学习和栅格地图的多机器人避障算法，包括强化学习算法的关键要素定义、多机器人避障策略的网络架构和训练过程及多阶段课程学习的应用。最后我们通过大量的仿真实验和实体实验验证了我们方法的优势。

3.1 多机器人避障研究背景

近年来，随着自主移动机器人的飞速发展，人们对多机器人避障技术的关注也越来越多，该技术在多智能体的许多应用中至关重要。例如多机器人搜索与救援^[88]、多机器人智能仓库系统^[89]、机器人在人群中自主导航^[90]和自动驾驶汽车^[91]。多机器人避障允许每个机器人从起始位置到达其目标位置，同时避免与其他机器人和障碍物发生碰撞。各种自主机器人之间的动态交互作用以及环境中的不确定性使该问题极具挑战性。

多机器人避障的方法通常可以分为两类：集中式方法和分散式方法。集中式方法通常提供一个中心服务器，该中心服务器在收集所有相关信息之后使用优化算法确定每个机器人的动作，这种方法已广泛应用于许多应用中，例如任务分配^[92]，编队控制^[93]和物体运输^[94-95]等。这些集中式避障方法假定每个机器人的动作由中央服务器确定，该服务器了解每个机器人的意图（如初始状态和目标，及其周围环境）。基于此知识，中央服务器将通过为每个机器人同时规划最佳路径来计算避免碰撞的动作。集中式方法通常可以保证安全性、完整性和近似最佳性。然而，这些方法是脆弱的并且难以扩展到具有大量机器人的复杂系统，这主要是由于以下原因：首先，随着机器人数量的增加，集中控制和调度变得越来越具有挑战性。其次，中央服务器与每个机器人之间都需要可靠的同步通信，这对于大型系统而言既不经济也不可行。第三，集中式系统易受各种故障的影响，例如中央服务器的干扰、机器人之间的通信延迟或传感器之间的通信错误。

此外，当在非结构化未知环境中部署多个机器人时，尤其是在人机共存的环境中，这些集中式避障算法将不适用。

与集中式方法不同，分散式方法允许每个机器人依靠对周围环境的本地感知来执行避障，而不再需要中央服务器。在这种情况下，移动机器人将需要合作，而不必了解其他机器人的意图。根据每个机器人的本地感知中指定的信息，去中心化（分散式）方法大致可以分为两组，即智能体级（agent-level）和传感器级（sensor-level）。具体而言，智能体级方法会考虑其他机器人的位置和运动数据，例如速度，加速度和路径。传感器级别的方法直接使用传感器数据。

关于智能体级分散式避障方法的现有工作可以进一步分为三类，即反应式的、基于轨迹的和基于学习的。特别地，反应式的智能体级分散方法为当前环境的几何配置指定了当前交互规则。速度障碍物^[96]（Velocity Obstacles, VO）被广泛用于指定这样的几何配置，这需要其他机器人的位置和速度来诱导 VO 来规划无碰撞运动。最优相互碰撞避免^[97]（Optimal Reciprocal Collision Avoidance, ORCA）是最成功的基于 VO 的方法之一，该方法可确保在密集环境中生成无碰撞的运动轨迹。具体来说，ORCA 为多机器人提供了能够防止它们在短时间内相互碰撞的充分条件，并且该算法可以自由地扩展到大型多机器人系统。ORCA 及其变体^[98-99]已广泛用于人群仿真和多智能体系统中。但是，这些方法使用启发式方法来构建参数化的碰撞避免模型，而这些模型很难进行调整以获得令人满意的性能。这些方法还需要完美感应周围机器人的位置、速度和形状，这在实际应用中很难实现。一些扩展方法^[100-102]要求通信网络共享机器人的位置和速度，这会损害系统的鲁棒性和灵活性。通过使用动作捕获系统作为全局定位设施，Zhou 等人^[103]提出的基于其缓冲 Voronoi 单元方法仅需要周围机器人的位置信息。ORCA 也已扩展到非全向轮机器人上。ORCA-DD^[104]使每个机器人的有效半径加倍，以确保在不同约束条件下机器人的无碰撞和平滑路径，然而这在狭窄通道和非结构化环境中会造成麻烦。NH-ORCA^[105-106]使差动驱动机器人能够在错误范围内遵循所需的速度命令，这仅会稍微增加机器人的有效半径，并且在避免碰撞方面的性能优于 ORCA-DD。

另一方面，基于轨迹规划的智能体级分散方法通过预测其他机器人的未来轨迹来显式评估机器人的未来状态。非合作方法的一个子类^[107-108]首先估计其他机器人的预测路径，然后为自我机器人规划无冲突的路径。但是，这些预测的路径通常会在拥挤的环境中标记出很大一部分空间是不安全的，这会导致机器人冻结问题^[9]。为了克服这个问题，研究者们提出了一种协作方法的子类，其中考虑了机器人之间的交互作用。基于协作轨迹的方法首先推断其他机器人的意图，共同计算所有附近机器人的可行路径，然后选择质量更高的路径，例如，每个机器人达到其相应目标的时间更短。但是，估计其他机器人的意图是困难的，

并且很难为每个机器人生生成路径。此外，随着时间的增加，每个机器人的预测路径与实际路径之间的差异会增大，系统需要对该方法进行快速更新。

以上讨论的分散式传统避障方法可能需要大量的计算代价或实践中无法满足的一些假设。这些方法还涉及许多需要手动调整的参数，而且不能够从过去的经验中自主学习，这使得它们更难处理不可预测的情况。基于学习的方法尝试优化参数化的策略，以避免各种环境中的碰撞。深度神经网络已广泛应用于有监督的学习范式中，以训练一种避免碰撞的策略，该策略将传感器输入映射到机器人的控制命令，以在具有静态障碍物的环境中进行机器人导航。Giusti 等人^[10] 应用神经网络对输入的彩色图像进行分类，并确定使四旋翼飞行器在丛林中无碰撞飞行的动作。为了获取大量的训练样本，一些徒步旅行者配备了三个头戴式摄像头，并且被要求在山路上行走以收集数据。Gandhi 等人^[11] 创建了无人机最大的碰撞数据集之一，以训练无人机飞行。Tai 等人^[109] 提出了一种将决策过程与卷积神经网络相结合的分层结构。Pfeiffer 等人^[12] 提出了一种模型，该模型能够直接学习复杂的从 2D 激光测距结果到移动室内机器人的相应控制命令映射。但是，基于监督学习的方法需要大量标记的训练数据集，并且训练数据集和测试环境之间的分布可能不匹配，这限制了它们在现实世界中的应用。其中一些工作仅能在空旷的走廊中避开简单的静态障碍物。

另一方面，深度强化学习（Deep Reinforcement Learning, DRL）算法在许多具有挑战性的任务中都取得了显著成功，例如围棋^[13-14]，视频游戏^[15-16] 和机器人领域^[110-111]。相反，基于 DRL 的方法能够从大量的试验和相应的反馈（奖励）中学习。为了通过 DRL 学习复杂的控制策略，机器人需要与训练环境进行长时间互动，以积累在不同状态下采取不同措施的后果经验。但是，在现实世界中收集此类交互式数据非常昂贵费时，有时由于安全原因甚至是不可能的^[112]。受基于 VO 的方法启发，Chen 等人^[113] 提供了一种基于 DRL 的方法来训练智能体级碰撞避免策略，该方法中网络仍然需要难以获得的自我机器人、附近机器人和移动障碍物的运动数据作为其输入。在他们的扩展工作^[114] 中，基于多个传感器执行了复杂感知任务，例如分割、识别和跟踪，以估计附近机器人和移动障碍物的运动数据。但是，这些感知任务需要大量的计算能力才能在线使用。尽管智能体级方法的缺点是需要精确且复杂的前端感知过程，但它们具有适用于各种传感器，易于在仿真环境中进行训练以及易于迁移到真实机器人的优点。

为了缓解智能体级分散式避障方法的缺点，Long 等人^[115] 和 Fan 等人^[116] 提出了直接使用原始传感器数据的传感器级分散式避障策略，即直接用二维激光扫描测距数据来计算机器人的控制命令。需要注意的是，仿真器中的激光数据与现实世界之间的差异是有限的，这是他们的避障策略模型能够顺利迁移到实体的保障。他们在多种模拟环境中训练这些避免碰撞策略，并随后将其部署到现实

世界中的机器人中。但是这些方法仅限于特定的传感器，即 2D 激光传感器。此外，移动机器人自主安全地导航需要多个传感器及其融合功能^[117]。例如，2D 激光传感器在类似办公室的环境中难以避开 3D 障碍物，在这种环境中，像 Kinect 这样的 RGB-D 相机可能会有所帮助。

在本文中，我们提出了一种分散式基于栅格地图的深度强化学习方法^[118]，用于在分布式和无通信环境中的多机器人避障。与智能体级^[114] 的和传感器级^[116] 的方法相比，我们使用以机器人自我为中心的局部栅格图来表示其周围的环境信息，包括自身形状信息和对其周围机器人和障碍物的感知信息，这些信息可以通过使用多个传感器或传感器融合技术轻松生成。然后，我们应用分布式近端策略优化（Distributed Proximal Policy Optimization, DPPO）算法来训练一个卷积神经网络，该网络将三帧以自我为中心的局部栅格图和机器人的相对局部目标位置直接映射到低级机器人控制命令中。本文提出的基于栅格地图的方法既具有智能体级方法的优势又具有传感器级方法的优势，基于栅格地图的方法适用于各种传感器，易于在模拟环境中进行训练，对嘈杂的传感器数据更鲁棒，不需要机器人运动数据和精确而复杂的前端感知过程，并考虑了相关机器人的大小和形状，从而使其更高效、更强大，并且更易于部署到实际机器人中。我们首先使用 DPPO 在多个移动机器人的定制仿真器中训练神经网络，其中针对两种情况使用两阶段课程学习策略来提高性能。然后，我们将训练后的模型部署到实际的机器人中，从而在导航过程中避免碰撞，而无需进行繁琐的参数调整。我们在仿真器中以及在现实世界中的四个差动驱动移动机器人上，都针对多种场景评估了该方法。定性和定量实验均表明我们的方法是有效的，并且在许多指标上都优于现有的基于 DRL 的方法。我们还进行了消融研究，显示了使用以自我为中心的栅格地图和多阶段课程学习的积极作用。我们的主要贡献概述如下：

- 我们提出了一种在无通信环境中基于栅格地图的深度强化学习多机器人避障方法，其中以自我为中心的局部栅格图用于表示机器人周围的环境信息，该信息可以通过使用多个传感器或传感器融合轻松地生成。
- 我们使用分布式近端策略优化算法在多个仿真环境中训练了避障策略，可以将其部署到真实的机器人中，而无需进行繁琐的参数调整，我们的策略网络将以机器人自我为中心的本地栅格图作为输入，并直接输出低级机器人控制命令。
- 我们在仿真器中以及现实世界中的多个差动驱动移动机器人上的多种场景中评估了我们的方法。定性和定量实验均表明，我们的方法是有效的，并且在许多指标上都优于现有的相关方法。
- 我们进行消融研究，以说明使用以自我为中心的栅格地图和多阶段课程学习的积极作用。

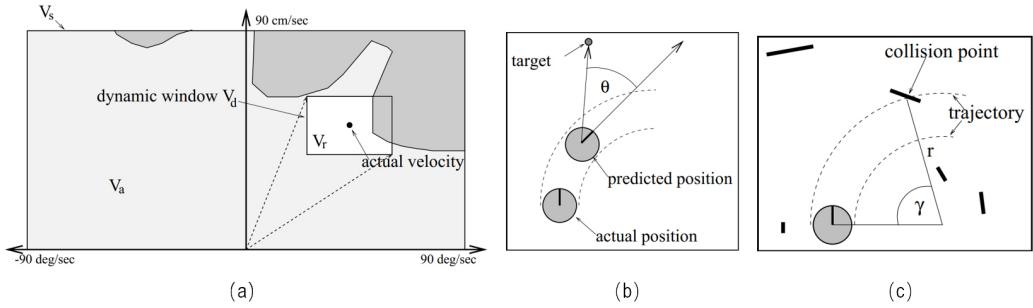


图 3.1 (a) 动态窗口示意图; (b) 方位角; (c) 障碍物距离。

本章的其余部分安排如下：在第3.2小节我们介绍两种著名的传统机器人避障方法；3.3小节介绍了多机器人避障的形式化定义；3.4小节中介绍了基于自我中心的本地栅格图用于多机器人碰撞避免的深度强化学习算法；3.5小节和3.6小节分别描述了仿真实验结果和真实世界的实验。最后，在3.7小节，我们把多机器人避障模块融入导航系统进行了长距离导航测试和百米楼宇物流实验。

3.2 传统机器人避障算法

3.2.1 动态窗口法 (DWA)

动态窗口算法^[119] (Dynamic Window Approach, DWA) 是机器人局部避障常用的算法之一，对于没有出现在全局地图中的障碍物，DWA 算法可以较好地解决。动态窗口法首先在机器人速度空间中进行速度采样，然后计算在相应采样速度下一段时间内的运动轨迹，再对这些轨迹进行评估，最终选择评估最好的一个轨迹对应的速度去执行。算法的优点是计算复杂度较低，动态窗口的含义是指机器人采样速度是在加减速度的动态可行范围内。速度采样是 DWA 算法中重要的组成部分，其主要思想是在速度 (v, ω) 的二维空间中，存在着无数组可能的速度。但是根据机器人自身和环境因素的影响，可以采样的速度被限制在了一定的范围内，主要限制因素包括三个方面：首先，移动机器人受到自身速度阈值的限制，即

$$V_m = \{v \in [v_{min}, v_{max}], \omega \in [\omega_{min}, \omega_{max}]\},$$

再者，移动机器人受到电机硬件性能影响，导致机器人的加减速速度也是有限的，在机器人运动的时间周期内，如图3.1 (a) 所示，存在一个机器人能够实际达到速度的动态窗口：

$$V_d = \{(v, \omega) | v \in [v_c - \dot{v}_b \Delta t, v_c + \dot{v}_a \Delta t], \omega \in [\omega_c - \dot{\omega}_b \Delta t, \omega_c + \dot{\omega}_a \Delta t]\},$$

其中 v_c 和 ω_c 表示当前速度， \dot{v}_a 表示最大加速度， \dot{v}_b 表示最大减速度。最后一点是基于移动机器人的安全性考虑。由于机器人的制动能力有限和最大减速度

限制，机器人需要在碰撞障碍物前停下来，因此运行速度在有障碍物的环境中需要在一定的范围内：

$$V_a = \{(v, \omega) | v \leq \sqrt{2 \cdot \text{dist}(v, \omega) \cdot \dot{v}_b} \wedge \omega \leq \sqrt{2 \cdot \text{dist}(v, \omega) \cdot \dot{\omega}_b}\},$$

其中 $\text{dist}(v, \omega)$ 为速度 (v, ω) 对应运动轨迹离障碍物的最近距离。当我们采样一组速度之后，首先模拟出机器人的轨迹，找到障碍物的位置，计算出机器人到障碍物之间的距离，之后计算得出这组速度能否使机器人在碰撞障碍物之前停下来。如果能停下来这组速度能够接受，否则应当抛弃这组采样。DWA 算法采样一组速度后，得到若干组可行轨迹，对每一条轨迹分别进行评估。评估方法一般是采用多因素加权求和的方式，各个方面影响因素求出一个评价指标，

$$G(v, \omega) = \sigma(\alpha \cdot \text{heading}(v, \omega) + \beta \cdot \text{dist}(v, \omega) + \gamma \cdot \text{velocity}(v, \omega)),$$

具体评价指标包含以下部分：第一部分是方位角指标。机器人轨迹末端的机器人朝向方向与目标方向之间有一个角度差（如图3.1 (b) 所示）。如果我们使用 180 减去角度差，评估的结果是 $\text{heading}(v, \omega)$ ，角度差越小得分越高，角度差越大，则评分越小。第二部分是空隙指标。 $\text{dist}(v, \omega)$ 求出的是当前轨迹上与最近障碍物的距离（如图3.1 (c) 所示），如果这一条轨迹上没有障碍物，那么其输出值为一个常数。第三部分是速度指标， $\text{velocity}(v, \omega)$ 用以评价当前轨迹的速度大小。最后是平滑处理，平滑处理即归一化，每组采样速度轨迹的每一评价指标项，需要除以对于所有采样轨迹该指标项的总和。DWA 算法这样做，是因为不同评价指标的量级具有差距，归一化之后能够更加客观公平的评估每组轨迹。DWA 此类单机器人反应式避障算法把环境看成瞬时静态，依靠高频计算保证实时反映，但由于不考虑其他机器人的移动轨迹，在动态环境中表现极差。

3.2.2 速度障碍物法 (VO)

速度障碍物^[96] (Velocity Obstacles, VO) 是一种基于速度规划的可以避免与多机器人或移动障碍物发生碰撞的著名算法，该算法为机器人选择与已知速度的动态障碍物避免碰撞的速度提供了充要条件。为了避免机器人间的碰撞，该方法扩展为相互速度障碍物法^[120-121] (Reciprocal Velocity Obstacles, RVO)，其中假定两个机器人都会选择另一个机器人引起的 RVO 之外的速度。但 RVO 仅能保证在特定条件下的避障，而不能为一般情况下的碰撞避免提供充分条件。最优相互避障法^[97] (Optimal Reciprocal Collision Avoidance, ORCA) 为多机器人避障提供了充分条件。下面我们详细介绍 ORCA 的算法内容。

假设有机器人 A 和 B，在时间 τ 内由 B 引起的 A 的速度障碍物 $VO_{A|B}^\tau$ 定义为一系列相对于 B 的会在 τ 时间内与 B 发生碰撞的 A 的相对速度，

$$VO_{A|B}^\tau = \{\mathbf{v} | \exists t \in [0, \tau], \mathbf{v}t \in D(p_B - p_A, r_A + r_B)\},$$

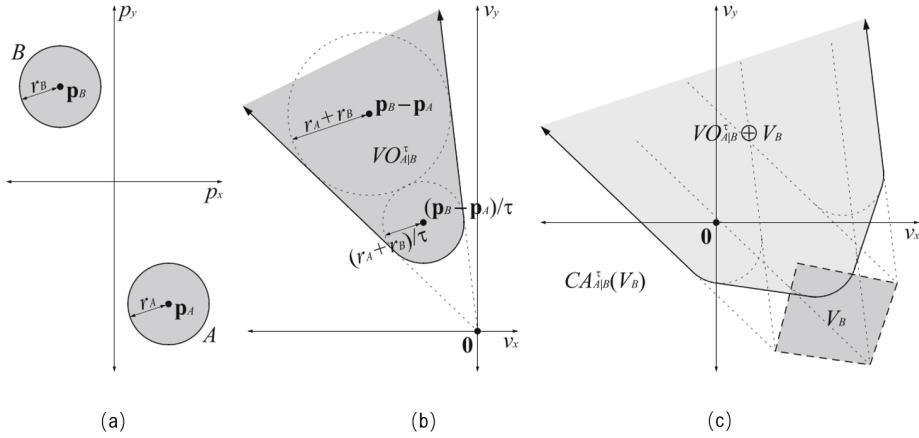


图 3.2 (a) 两个机器人 A 和 B; (b) 速度障碍物 $VO_{A|B}^\tau$ (灰色区域) 示意图; (c) 由 B 的速度集合 V_B (深灰区域) 生成的 A 的碰撞避免速度 $CA_{A|B}^\tau(V_B)$ (浅灰区域) 示意图。

其中 $D(\mathbf{p}, r) = \{\mathbf{q} \mid \|\mathbf{q} - \mathbf{p}\| < r\}$, 定义为以 \mathbf{p} 为原点半径为 r 的圆盘。速度障碍物的几何解释如图3.2所示, $VO_{A|B}^\tau$ 和 $VO_{B|A}^\tau$ 成原点对称。假设当前机器人 A 和 B 的速度分别为 \mathbf{V}_A 和 \mathbf{V}_B , 速度障碍物 VO 的定义暗示着如果 $\mathbf{V}_A - \mathbf{V}_B \in VO_{A|B}^\tau$ 或者 $\mathbf{V}_B - \mathbf{V}_A \in VO_{B|A}^\tau$, 如果他们持续以这个速度前进, 则机器人 A 和 B 将会在时间 τ 内发生碰撞。相反如果 $\mathbf{V}_A - \mathbf{V}_B \notin VO_{A|B}^\tau$, 则机器人 A 和 B 将不会在时间 τ 内发生碰撞。如果我们把 $X \oplus Y$ 定义为集合 X 和 Y 的 Minkowski 和, 即

$$X \oplus Y = \{\mathbf{x} + \mathbf{y} \mid \mathbf{x} \in X, \mathbf{y} \in Y\},$$

则对于任意的集合 V_B , 如果 $\mathbf{v}_B \in V_B$ 且 $\mathbf{v}_A \notin VO_{A|B}^\tau \oplus V_B$, 则机器人 A 和 B 以当前速度将会在时间 τ 内发生碰撞。即定义对于 B 从 V_B 选择速度的 A 的碰撞避免速度如下:

$$CA_{A|B}^\tau(V_B) = \{\mathbf{v} \mid \mathbf{v} \notin VO_{A|B}^\tau \oplus V_B\},$$

如果 $V_A \subseteq CA_{A|B}^\tau(V_B)$ 且 $V_B \subseteq CA_{B|A}^\tau(V_A)$, 则称 V_A 和 V_B 为 A 和 B 的互惠避障, 如果 $V_A = CA_{A|B}^\tau(V_B)$ 且 $V_B = CA_{B|A}^\tau(V_A)$, 则称 V_A 和 V_B 为 A 和 B 的最大互惠避障。因为 A 和 B 是不能相互通信且相对独立的机器人, 所以他们需要各自自主推断出自己被允许执行的速度范围, 这里存在许多对的 V_A 和 V_B 在要遵循着这些要求。但在这些成对的 V_A 和 V_B 中, 在互为最大化的可选避障速度集合中选择一对接近最优的速度, 称 A 的为 \mathbf{v}_A^{opt} , B 的为 \mathbf{v}_B^{opt} 。我们称以上这些避免碰撞可选范围 A 和 B 相互最大化的速度集合 A 的 $ORCA_{A|B}^\tau$, B 的为 $ORCA_{B|A}^\tau$, 即 $CA_{A|B}^\tau(ORCA_{B|A}^\tau) = ORCA_{A|B}^\tau$ 且 $CA_{B|A}^\tau(ORCA_{A|B}^\tau) = ORCA_{B|A}^\tau$ 。如图3.3 (a) 所示, 设 \mathbf{u} 为从 $\mathbf{v}_A^{opt} - \mathbf{v}_B^{opt}$ 到 VO 边界的最近点, 设 \mathbf{n} 为在点 $\mathbf{v}_A^{opt} - \mathbf{v}_B^{opt} + \mathbf{u}$ 处边界 $VO_{A|B}^\tau$ 的外向法线, 则

$$ORCA_{A|B}^\tau = \{\mathbf{v} \mid (\mathbf{v} - (\mathbf{v}_A^{opt} + \frac{1}{2}\mathbf{u})) \cdot \mathbf{n} \geq 0\}.$$

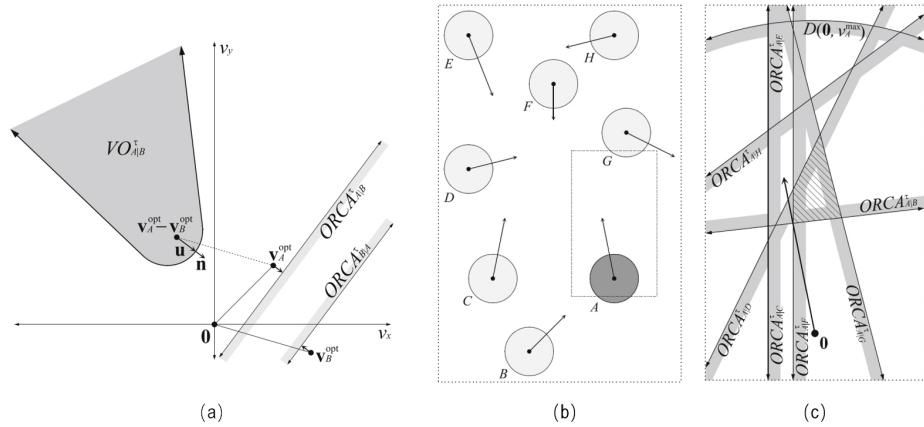


图 3.3 (a) ORCA 求解示意图; (b) 八机器人环境实例; (c) 由其他所有机器人影响的机器人 A 的允许速度半平面。

每个机器人 A 在时间 Δt 内执行连循环往复的感知和运动。机器人在每次循环中需要知道自己的和其它机器人的半径、当前位置和当前速度，基于这些信息，机器人 A 推断出自己的对于机器人 B 的 $ORCA_{A|B}^{\tau}$ 半平面。在这个半平面内允许的速度范围内，A 又和其它机器人产生新的半平面，如此不断叠加，最后的集合定义为：

$$ORCA_A^{\tau} = D(\mathbf{0}, \mathbf{v}_A^{\max}) \cap \bigcap_{B \neq A} ORCA_{A|B}^{\tau},$$

如图3.3 (a) 和 (b) 描绘了八机器人环境中 A 机器人的 $ORCA_A^{\tau}$ 求解示意图。然后机器人从所有允许的速度中选择最接近期望速度 $\mathbf{v}_A^{\text{pref}}$ 的新速度 $\mathbf{v}_A^{\text{new}}$ ：

$$\mathbf{v}_A^{\text{new}} = \arg \min_{\mathbf{v} \in ORCA_A^{\tau}} \left\| \mathbf{v} - \mathbf{v}_A^{\text{pref}} \right\|,$$

最后机器人执行此速度到达新的位置，感知行动循环进行。

对于环境中的障碍物，我们可以采取和上述处理机器人间避障基本相同的方式处理。主要不同的是障碍物是静止的不会移动，因此机器人应完全承担避免与障碍物碰撞的责任。我们通常可以用线段的集合来对障碍物进行建模。如图3.4所示，设 O 是这样的线段之一，A 是半径为 r_A ，位于点 \mathbf{p}_A 的机器人。那么由障碍物 O 影响的障碍速度 $VO_{A|O}^{\tau}$ 定义如下：

$$VO_{A|O}^{\tau} = \{ \mathbf{v} | \exists t \in [0, \tau], \mathbf{v}t \in O \oplus -D(\mathbf{p}_A, r_A) \},$$

如果其速度在 $VO_{A|O}^{\tau}$ 内，则机器人 A 将在 τ 时间内与障碍物 O 碰撞。但由于 $VO_{A|O}^{\tau}$ 的补集是非凸区域，所以线性规划算法无法直接使用，为此，对于障碍物 O，我们定义 $ORCA_{A|O}^{\tau}$ 的半平面为划分线为切线过 $VO_{A|O}^{\tau}$ 边界上一个最接近 $\mathbf{v}_A^{\text{opt}}$ 的点。 VO 类避障方法有个很大的缺点是需要精确感知周围机器人的位置和速度，在实际使用时往往需要复杂的前端感知处理。

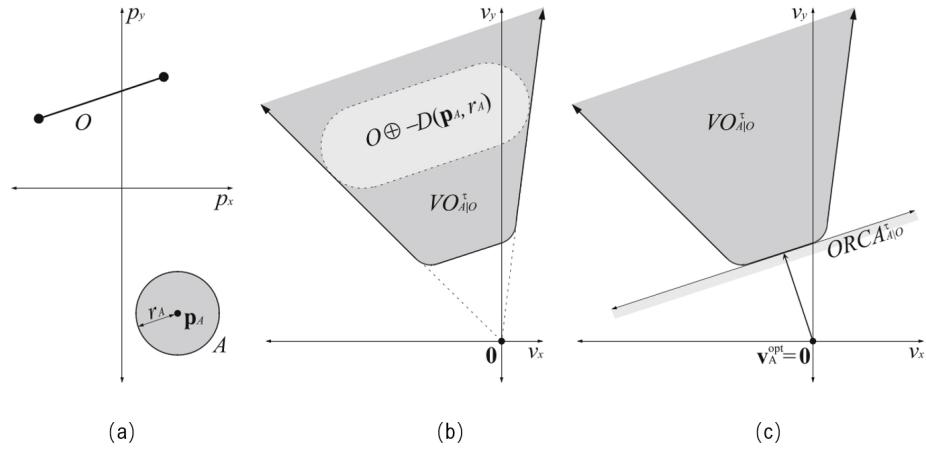


图 3.4 (a) 机器人 A 和线障碍物 O; (b) $\tau = 2$ 的速度障碍物 $VO_{A|O}^\tau$; (c) $ORCA_{A|O}^\tau$ 几何示意图。

3.3 多机器人避障问题定义

多机器人避障问题为一组具有完整或非完整约束的 N 个移动机器人在到达目标位置时执行相应的轨迹，同时避免彼此碰撞以及与环境中的障碍物发生碰撞。特别地，在每个时间步 t 处，每个机器人 i ($0 \leq i \leq N - 1$) 首先接收其传感数据 \mathbf{c}_t^i ，相对于其起始位姿的偏移角 α_t^i 和其坐标系中的本地目标位置 \mathbf{g}_t^i ，即机器人 i 在其全局规划器生成到达目标点的路径中的相对位置。然后机器人选择一个动作 \mathbf{a}_t^i 移向其本地目标 \mathbf{g}_t^i 。需要注意的是，感知数据 \mathbf{c}_t^i 不需要对环境有完整的观察。因此无需假设完美的感应，我们在这里只需要部分观察即可。另一方面，如第 3.1 节中所述，我们也没有直接将原始传感器数据用作避障策略的输入。相反，我们使用以自我为中心的局部栅格图 \mathbf{M}_t^i 来表示机器人 i 周围的环境信息，包括其形状 Ω^i 和感知结果 \mathbf{c}_t^i 。具体来说，我们考虑

$$\mathbf{M}_t^i = f_\lambda(\mathbf{c}_t^i, \Omega^i),$$

其中 f_λ 表示由参数 λ 指定的栅格图生成器，以根据机器人的形状 Ω^i 和感知结果 \mathbf{c}_t^i 生成以自我为中心的局部栅格图 \mathbf{M}_t^i 。

需要注意的是，此处不需要其他机器人的意图和运动数据。但是，我们可以从以自我为中心的局部栅格图的连续帧中隐式估计其他机器人的运动信息，即速度、加速度和路径等。我们还可以根据其局部目标和偏移方位角的连续帧来估算自我机器人的历史轨迹和速度（即线速度和角速度）。特别是，我们将在时间 t 处对机器人 i 的观察指定为 $\mathbf{o}_t^i = (\mathbf{M}_t^i, \mathbf{g}_t^i, \alpha_t^i)$ 。我们可以将这样的观察状态的 $m \geq 1$ 个连续帧用作避障策略 π_θ 的输入。具体来说，我们考虑

$$\mathbf{a}_t^i = \pi_\theta(\mathbf{o}_{t-m+1}^i, \dots, \mathbf{o}_t^i),$$

其中 π_θ 表示由参数 θ 指定的避障策略，以基于观察的 m 个连续帧来选择动作 \mathbf{a}_t^i 。在本文中，我们表示动作 $\mathbf{a}_t^i = (v_t^i, \omega_t^i)$ ，其中 v_t^i 是线速度， ω_t^i 是角速度，它们是机器人 i 直到下一个时间步 $t+1$ 所需要执行的速度。

在多机器人避障中，每个机器人 i 都从起始位置 \mathbf{p}_0^i 移至目标位置 \mathbf{p}_g^i ，同时避免彼此之间的碰撞和与障碍物 B_k ($1 \leq k \leq M$) 的碰撞，即在环境中可观察到的障碍物外观。此外， B_k 可以被视为栅格图中相应障碍物占据的一组单元格，并且符号 $\Omega^i(\mathbf{p}_t^i)$ 表示在时间 t 被具有形状 Ω^i 的机器人 i 在 \mathbf{p}_t^i 位置占据的一组单元格。我们打算在不与其他机器人和障碍物发生碰撞的约束下，使每个机器人 $0 \leq i \leq N-1$ 的到达时间 t_g^i 的期望最小化。具体来说，

$$\begin{aligned} & \arg \min_{\theta} \mathbb{E} \left[\sum_{i=0}^{N-1} t_g^i \mid \pi_\theta \right], \\ & \text{s.t. for each } 0 \leq i \leq N-1, 0 \leq j \leq N-1, i \neq j, \text{ and } 1 \leq k \leq M, \\ & \mathbf{p}_{t+1}^i = \mathbf{p}_t^i + \Delta t \cdot \pi_\theta(\mathbf{o}_{t-m+1}^i, \dots, \mathbf{o}_t^i), \mathbf{p}_{t_g^i}^i = \mathbf{p}_g^i, \Omega^i(\mathbf{p}_t^i) \cap \Omega^j(\mathbf{p}_t^j) = \emptyset, \text{ and } \Omega^i(\mathbf{p}_t^i) \cap B_k = \emptyset. \end{aligned}$$

其中第二行条件要求每个机器人都必须达到目标而不发生碰撞。在下文中，我们介绍实现此目的的具体方法。

3.4 基于栅格地图的多机器人避障算法

我们通过描述多机器人分布式近端策略优化 (DPPO) 强化学习算法的关键要素来开始本节。然后，我们描述了有关多机器人避障策略的网络架构和训练过程的详细信息。最后，我们详细介绍了用于优化策略的训练过程和方案。

3.4.1 强化学习组件定义

如第 3.3 节中所述，多机器人避障问题可以被视为不确定性下的部分可观察决策问题，可以将其定义为部分可观察的马尔可夫决策过程^[122] (POMDP)，并使用强化学习算法解决。具体来说，POMDP 问题由 6 个元组 $\langle S, A, P, R, \Omega, O \rangle$ 组成，其中 S 是状态空间， A 是动作空间， P 是转移函数，用于描述转移到下一个状态的概率分布， R 是说明即时状态动作奖励信号的奖励函数， Ω 是观察空间，其中 $\mathbf{o} \in \Omega$ ，而 O 是观察概率函数，描述了观察 \mathbf{o} 的概率分布。

在下文中，我们将描述基于栅格地图的多机器人碰撞避免方法的细节，包括观察空间、动作空间和奖励函数设计。

1. 观察空间

如第 3.3 节中所述，观测值 \mathbf{o}_t^i 由相应机器人 i 的生成栅格图 \mathbf{M}_t^i 、目标相对位置 \mathbf{g}_t^i 和偏移定向角 α_t^i 组成。我们使用最新的三帧观测值 $\vec{\mathbf{o}}_t^i = (\mathbf{o}_{t-2}^i, \mathbf{o}_{t-1}^i, \mathbf{o}_t^i)$ 作为

避障策略的神经网络输入。在本文中，以机器人为中心的局部栅格图 \mathbf{M}_t^i 是通过 2D 激光扫描测距生成的，该二维激光具有 180 度水平视场（FOV），可对机器人 i 的形状以及附近障碍物和其他机器人的可见外观进行编码。以机器人为中心的本地栅格地图是直接从栅格地图^①中构建的，可以通过使用各种传感器或传感器融合轻松生成。相对局部目标位置 \mathbf{g}_t^i 是表示机器人的当前位置 (x, y) 的二维向量，它表示由全局规划器生成的以到达环境中机器人 i 的最终目标的路径中局部目标点，或者是距离不是特别远的最终目标点。角度 α_t^i 表示机器人 i 相对于其起始位姿的方向变化。需要注意的是，策略网络可以从以自我为中心的局部栅格图的连续帧中隐式提取其他机器人的运动信息，并且可以从其局部目标和偏移方位角的连续帧中隐式地提取自我机器人的历史轨迹和速度。然而我们需要在估算的准确性和策略的效率之间权衡利弊，在我们的实验中，当将三帧连续观测值视为网络的输入时，可以获得令人满意的性能。

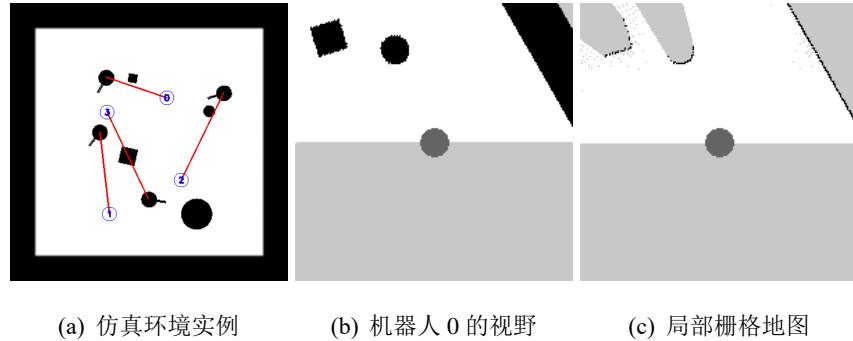


图 3.5 (a) 仿真器环境实例，其中蓝色数字圆圈代表具有相应编号的机器人目标位置，红色线代表机器人从当前位置到目标的直线，其他黑色像素代表各种障碍物；(b) 环境中机器人 0 的环境局部视图；(c) 使用 2D 激光传感器从局部视图生成的以自我为中心的局部栅格图。

以自我为中心的局部栅格图是由栅格地图构造而成的，栅格地图已被广泛应用于机器人导航^[123-124]中，并且可以由各种传感器或传感器融合信息生成且具有抗噪性。我们首先生成一份机器人 i 的本地栅格地图，那么我们可以通过在栅格地图中添加机器人的配置（即机器人的尺寸和形状）来构建以自我为中心的局部栅格图 \mathbf{M}_t^i 。 \mathbf{M}_t^i 中的每个单元中都有一个值来表示周围的环境。具体来说，值为 0 的单元表示机器人周围的障碍物或其他机器人。值 255 的单元表示空白空间，值 100 的单元表示对于机器人未知的或未检测到的位置，值 200 的单元表示由机器人自身占据的空间。图 3.5(a)中展示了我们定制的仿真器中一种训练环境示例。在该示例中，机器人 0 生成的以自我为中心的局部栅格图显示在图 3.5(c)中。

^①http://wiki.ros.org/costmap_2d

在本文中，我们将以自我为中心的局部栅格图转换为灰度图像，将像素值除以 255 以进行归一化和简化网络设置。在不失一般性的前提下，我们在本文中提到的栅格图也就是相应的灰度图像。

2. 动作空间

机器人的动作空间是在连续空间中的一组允许速度。差动轮机器人 i 的动作 \mathbf{a}_t^i 由线速度 v_t^i 和角速度 ω_t^i 组成，即 $\mathbf{a}_t^i = (v_t^i, \omega_t^i)$ 。在本文中，我们设置 $v_t^i \in [0, 0.6]$ （以米/秒为单位）， $\omega_t^i \in [-0.9, 0.9]$ （以弧度/秒为单位），这样可以由我们实验中使用的差动轮机器人直接执行。由于机器人后方缺少传感器，因此 $v_t^i \geq 0$ ，即不允许机器人向后移动。

3. 回报函数

在强化学习中，智能体的目标是最大化其累积奖励，奖励函数指定智能体需要实现的目标，而不是如何实现的目标。在我们的长距离多机器人避障任务中，目标是在无冲突约束的情况下最小化每个机器人到达其本地目标点的平均到达时间，其中除非达到本地目标点或发生碰撞，否则奖励信号通常是微弱且稀疏的。为了缓解此问题，我们通过使用奖励塑造技术^[125]（Reward Shaping），添加额外的奖励信号 r_t^g 来指导机器人朝其目标点移动。

在我们的设置中，每个机器人都有相同的奖励函数。我们使用 r_t 表示机器人在时间 t 收到的奖励。我们在本文中使用以下奖励函数：

$$r_t = r_t^g + r_t^a + r_t^c + r_t^s,$$

其中 r_t 是 r_t^g, r_t^a, r_t^c 和 r_t^s 的四部分之和。

具体来说， r_t^g 指当机器人远离其本地目标时的惩罚。我们将 r_t^g 定义为：

$$r_t^g = \eta(\|\mathbf{p}_{t-1} - \mathbf{p}_g\| - \|\mathbf{p}_t - \mathbf{p}_g\|),$$

其中 \mathbf{p}_t 是机器人在时间 t 的位置， \mathbf{p}_g 是相对目标点的位置， η 是控制惩罚的超参数。

r_t^a 表示机器人到达其本地目标时的奖励，即机器人与其本地目标之间的距离小于 d_{arr} ，其中 $r_{arr} > 0$ 是固定的奖励。

$$r_t^a = \begin{cases} r_{arr} & \text{if } \|\mathbf{p}_t - \mathbf{p}_g\| < d_{arr}, \\ 0 & \text{otherwise.} \end{cases}$$

r_t^c 指机器人遇到碰撞时的惩罚。我们将 r_t^c 定义为：

$$r_t^c = \begin{cases} r_{col} & \text{if collision,} \\ 0 & \text{otherwise,} \end{cases}$$

其中 $r_{col} < 0$ 是一个针对碰撞的固定惩罚。

最后，我们对每个时间步长施加较小的负惩罚，即 $r_t^s < 0$ ，以鼓励机器人缩短路径。在训练过程中我们设置 $r_{arr} = 500$, $\eta = 200$, $r_{col} = -500$ 和 $r_t^s = -5$ 。

3.4.2 分布式近端策略优化算法

在 POMDP 问题中，要避免多机器人碰撞，每个机器人 i 都需要一个随机策略 $\pi_\theta(\mathbf{a}_t^i | \vec{\mathbf{o}}_t^i)$ ，它指在时间 t 映射最近三个观测值 $\vec{\mathbf{o}}_t^i$ （即 $(\mathbf{o}_{t-2}^i, \mathbf{o}_{t-1}^i, \mathbf{o}_t^i)$ ）到动作 \mathbf{a}_t^i 的概率。需要注意的是，在我们的工作中每个机器人都有相同的随机策略。然后在不失一般性的情况下，以下我们省略下标 i 。

在深度强化学习中，策略梯度方法通过使用随机梯度上升最大化期望收益 $J(\pi_\theta)$ 来优化随机策略 π_θ 。具体来说，

$$\nabla_\theta J(\pi_\theta) = \nabla_\theta \mathbb{E}_{\tau \sim \pi_\theta} [R(\tau)],$$

$$\theta_{k+1} = \theta_k + \alpha \nabla_\theta J(\pi_{\theta_k}),$$

其中 τ 表示一条轨迹， $R(\tau)$ 是有限折扣收益 ($R(\tau) = \sum_{t=0}^{\infty} \gamma^t r_t$ ，其中 $\gamma \in (0, 1)$ 是折扣因子)， $\nabla_\theta J(\pi_\theta)$ 为策略梯度，而 α 是学习率。通过引入轨迹概率 $P(\tau|\theta)$ 并使用数学技巧^[126]，我们可以从上面的方程式进一步得出以下解析性梯度表达式，

$$\begin{aligned} \nabla_\theta J(\pi_\theta) &= \mathbb{E}_{\tau \sim \pi_\theta} [\nabla_\theta \log P(\tau|\theta) R(\tau)] \\ &= \mathbb{E}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^T \nabla_\theta \log \pi_\theta (\mathbf{a}_t | \vec{\mathbf{o}}_t) R(\tau) \right]. \end{aligned}$$

通常，我们不需要描述一个动作的绝对含义，而只需要描述它平均比其他动作好多少。换句话说，我们需要知道该动作的优势。因此，我们将 \hat{A}^{π_θ} 定义为策略 π_θ 的优势函数，并将上面的等式进一步改写为：

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{\tau \sim \pi_\theta} \left[\sum_{t=0}^T \nabla_\theta \log \pi_\theta (\mathbf{a}_t | \vec{\mathbf{o}}_t) \hat{A}^{\pi_\theta} (\vec{\mathbf{o}}_t, \mathbf{a}_t) \right].$$

尽管使用相同的轨迹在 $J(\pi_\theta)$ 上执行多个优化步骤很吸引人，但这并没有得到很好结果，而且常常会导致破坏性的大幅度策略更新。近端策略优化^[127] (PPO) 通过引入重要性采样并使用剪切方法限制策略更新的大小来解决此问题。具体而言，具有剪切操作更新策略的 PPO 通过

$$\theta_{k+1} = \arg \max_{\theta} \mathbb{E}_{\vec{\mathbf{o}}, \mathbf{a} \sim \pi_{\theta_k}} [L(\vec{\mathbf{o}}, \mathbf{a}, \theta_k, \theta)], \quad (3.1)$$

其中 L 定义为

$$L(\vec{\mathbf{o}}, \mathbf{a}, \theta_k, \theta) = \min \left(\frac{\pi_\theta(\mathbf{a} | \vec{\mathbf{o}})}{\pi_{\theta_k}(\mathbf{a} | \vec{\mathbf{o}})} \hat{A}^{\pi_{\theta_k}}(\vec{\mathbf{o}}, \mathbf{a}), g(\epsilon, \hat{A}^{\pi_{\theta_k}}(\vec{\mathbf{o}}, \mathbf{a})) \right), \quad (3.2)$$

$$g(\epsilon, \hat{A}) = \begin{cases} (1 + \epsilon)\hat{A} & \hat{A} \geq 0, \\ (1 - \epsilon)\hat{A} & \hat{A} < 0, \end{cases}$$

其中 ϵ 是剪切函数的比率。

我们可以使用学习的状态值函数 $V_\phi(\vec{\mathbf{o}}_t)$ 来构造方差降低的优势函数估计量 $\hat{A}^{\pi_\theta}(\vec{\mathbf{o}}_t, \mathbf{a}_t)$ 。特别是对于给定长度 T 的轨迹段，我们使用广义优势估算^[128] (Generalized Advantage Estimation, GAE) 的剪切版本来计算优势函数。

$$\hat{A}^{\pi_\theta}(\vec{\mathbf{o}}_t, \mathbf{a}_t) = \delta_t + (\gamma \lambda) \delta_{t+1} + \dots + (\gamma \lambda)^{T-1-t} \delta_{T-1}, \quad (3.3)$$

其中 $\delta_t = r_t + \gamma V_\phi(\vec{\mathbf{o}}_{t+1}) - V_\phi(\vec{\mathbf{o}}_t)$, V_ϕ 表示状态值函数，折扣因子 $0 \leq \gamma < 1$ ，并且参数 $0 \leq \lambda \leq 1$ 。

在本文中，我们应用分布式近端策略优化 (DPPO) 算法来训练用于多机器人避障的随机策略。DPPO 是 PPO 的扩展，它通过在多种环境中收集分布式环境中的经验来实现，在该环境中，多个机器人共享同一策略 π_θ 以交互地执行操作。然后，这些收集的经验将用于更新策略 π_θ 和状态值函数 V_ϕ 的参数。

下面，我们详细介绍 DPPO 算法的部署细节，包括训练过程，神经网络架构和多阶段课程学习的使用。

1. 训练过程

我们通过集中式学习分散执行范式来实现 DPPO 算法，其中同类学习智能体同时从多个模拟环境中采样，以训练代表其避障策略的相同共享神经网络。算法 3.1 描述了这种训练过程，该过程通过并行执行策略对轨迹进行采样，并使用采样数据训练更新策略网络 π_θ 。每个机器人都利用相同的策略在相应的环境中生成轨迹。当每个机器人到达其目标或遇到与障碍物或其他机器人的碰撞时，终止在环境中运行的机器人的回合，然后环境也会重置。具体来说，每个机器人 i 在每个时间步 t 处接收自己的输入 $\vec{\mathbf{o}}_t^i$ ，即最近的三帧观测值，并执行在多个环境中从共享策略 π_θ 生成的动作 \mathbf{a}_t^i ，然后返回 r_t^i 和新的状态观测值 $\vec{\mathbf{o}}_{t+1}^i$ ，并且我们把 $(\vec{\mathbf{o}}_t^i, \mathbf{a}_t^i, r_t^i, V_t^i)$ 存储在缓冲区队列中 (ll. 7-11)。当缓冲区队列已满或轨迹长度达到回合中的最大轨迹长度限制 T_m 时，我们需要设置 $V_{t+1}^i = V_\phi(\vec{\mathbf{o}}_{t+1}^i)$ 来强制剪断轨迹以结束回合 (ll. 12-13)。当机器人到达目标点 ($\|\mathbf{p}_t^i - \mathbf{p}_g^i\| < d_{arr}$) 或与其他障碍物碰撞 (there exists some $1 \leq k \leq M$, s.t., $\Omega^i(\mathbf{p}_t^i) \cap B_k \neq \emptyset$)，或与其他机器人碰撞时 (there exists some $0 \leq j \leq N - 1$, s.t., $i \neq j$ and $\Omega^i(\mathbf{p}_t^i) \cap \Omega^j(\mathbf{p}_t^j) \neq \emptyset$)，我们将设置 $V_{t+1}^i = 0$ 来结束此回合 (ll. 14-15)。当该回合内所有机器人都结束时，

算法3.1 分布式近端策略优化

```

1 初始化策略网络  $\pi_\theta$  和状态值网络  $V_\phi$  参数。
2 初始化空白经验缓存池  $Buffer$ .
3 for  $epoch = 1, \dots, E$  do
4   // 并行收集数据
5   for  $step t = 1, \dots, T_{ep}$  do
6     for  $robot i = 0, \dots, N - 1$  do
7        $\mathbf{a}_t^i = \pi_\theta(\vec{\mathbf{o}}_t^i)$ 
8        $V_t^i = V_\phi(\vec{\mathbf{o}}_t^i)$ 
9        $\vec{\mathbf{o}}_{t+1}^i, r_t^i = step(\mathbf{a}_t^i)$ 
10      Add  $(\vec{\mathbf{o}}_t^i, \mathbf{a}_t^i, r_t^i, V_t^i)$  to  $Buffer$ 
11       $\vec{\mathbf{o}}_t^i \leftarrow \vec{\mathbf{o}}_{t+1}^i$ 
12      if  $Buffer$  is full or trajectory length  $> T_m$  then
13        Cut trajectory by  $V_{t+1}^i = V_\phi(\vec{\mathbf{o}}_{t+1}^i)$ .
14      else if  $\|\mathbf{p}_t^i - \mathbf{p}_g^i\| < d_{arr}$  or there exists some  $1 \leq k \leq M$ , s.t.,
15         $\Omega^i(\mathbf{p}_t^i) \cap B_k \neq \emptyset$  or there exists some  $0 \leq j \leq N - 1$ , s.t.,  $i \neq j$ 
16        and  $\Omega^i(\mathbf{p}_t^i) \cap \Omega^j(\mathbf{p}_t^j) \neq \emptyset$  then
17          Finish episode by  $V_{t+1}^i = 0$ .
18      end
19    end
20    if all robots have finished then
21      Estimate advantages using GAE in Equation (3.3), i.e.,
22       $\hat{A}_t = \sum_{l=t}^{T-1} (\gamma \lambda)^{l-t} \delta_l$ , where  $\delta_l = r_l + \gamma v_{l+1} - v_l$ .
23      Add  $\hat{A}_t$  to  $Buffer$ 
24       $\vec{\mathbf{o}}_t = reset()$ 
25    end
26  end
27   $\pi_{\theta_{old}} \leftarrow \pi_\theta$ 
28  训练策略和值网络;
29  清空经验缓存池  $Buffer$ ;
30 end

```

算法3.2 策略网络和值网络的训练

```

1 // 更新策略网络
2 for  $m = 1, \dots, E_\pi$  do
3   Compute  $L^{PPO}(\theta)$  by Equation (3.2), i.e.,

$$L^{PPO}(\theta) = \sum_{t=1}^{T_{ep}} \min \left( \frac{\pi_\theta(\mathbf{a}_t | \vec{\mathbf{o}}_t)}{\pi_{\theta_{old}}(\mathbf{a}_t | \vec{\mathbf{o}}_t)} \hat{A}_t, g(\epsilon, \hat{A}_t) \right), \text{ where}$$


$$g(\epsilon, \hat{A}) = \begin{cases} (1 + \epsilon)\hat{A} & \hat{A} \geq 0, \\ (1 - \epsilon)\hat{A} & \hat{A} < 0, \end{cases}$$

4   if  $KL[\pi_{\theta_{old}} | \pi_\theta] > 1.5KL_{target}$  then
5     break
6   end
7   Update  $\theta$  with the learning rate  $lr_\theta$  using Adam w.r.t.  $L^{PPO}(\theta)$  by
      Equation (3.1).
8 end
9 // 更新值网络
10 for  $n = 1, \dots, E_V$  do
11    $L^V(\phi) = - \sum_{t=1}^{T_{ep}} (\sum_{t'>t} \gamma^{t'-t} r_{t'} - V_\phi(\vec{\mathbf{o}}_t))^2$ 
12   Update  $\phi$  with the learning rate  $lr_\phi$  using Adam w.r.t.  $L^V(\phi)$ .
13 end

```

我们估计存储在缓冲区中每步的优势值，然后将其保存到缓冲区中的相应位置，然后重置仿真环境 (ll. 18-22)。

当收集的经验已填满整个经验缓冲区时，将更新随机策略 π_θ 的策略网络和状态值函数 V_ϕ 的值网络。算法 3.2 描述了这一训练过程。为了更新策略 π_θ ，算法 3.2 中的替代损失 $L^{PPO}(\theta)$ 是根据收集的轨迹构造的，并在 Kullback–Leiber (KL) 散度约束下使用 Adam 优化器^[129] 进行了优化 (ll. 2-7)。要更新状态值函数 V_ϕ ，即估计优势函数 \hat{A}_t 的基线，算法 3.2 中的损失平方误差 $L^V(\phi)$ 也是根据这些收集的轨迹构造的，也使用 Adam 优化器针对 E_V 个回合进行了优化 (ll. 10-13)。最后，我们清除缓冲区并重新收集用于训练的经验 (ll. 26)。价值网络与策略网络具有相同的网络架构，不同之处在于最后一层被修改为仅输出 $\vec{\mathbf{o}}_t$ 的值，即具有线性激活功能的一个单元。需要注意的是，策略网络和价值网络均独立更新，并且不共享其参数，这是由于使用两个单独的网络通常会在实践中带来更好的性能。

我们可以轻松地将算法 3.1 和 3.2 中的 DPPO 算法扩展为具有一百多个机器人的复杂多机器人系统，该系统处于分布式和无通信场景。我们还在实验中说明了

这种情况，其中每个机器人都独立收集其经验。分散执行不仅大大减少了样本收集的时间，而且使该算法适合在多种场景下训练大量机器人。

2. 网络架构

图3.6中显示了避障策略 π_θ 的网络架构。如第3.4.1节中所述， \vec{o}_t 被记为网络的输入，它由三部分组成，即三个以自我为中心的局部栅格图的连续帧 ($\mathbf{M}_{t-2}, \mathbf{M}_{t-1}, \mathbf{M}_t$)，三个连续的相对局部目标位置帧 ($\mathbf{g}_{t-2}, \mathbf{g}_{t-1}, \mathbf{g}_t$) 和三个连续的偏移定向角帧 ($\alpha_{t-2}, \alpha_{t-1}, \alpha_t$)。

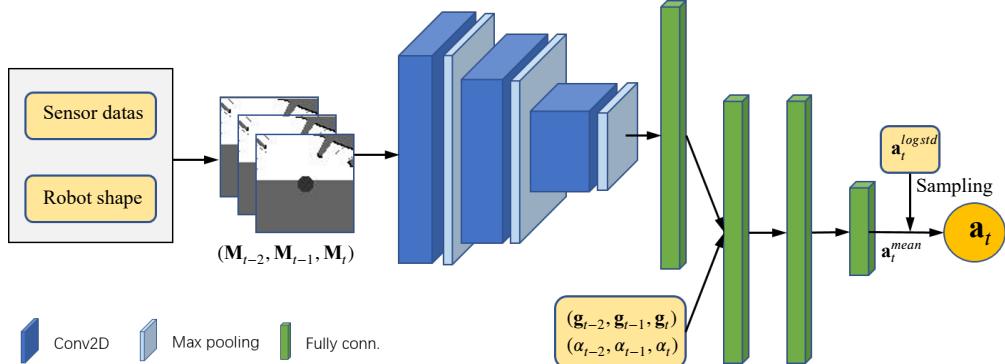


图 3.6 策略网络的架构图。网络的输入是 \vec{o}_t ，它由三帧以自我为中心的局部栅格图和三帧局部目标位置和偏移方位角组成。网络计算动作的平均值，该平均值由线速度和角速度组成。产生的动作是来自高斯分布的样本。

网络首先使用三层卷积层 L^{CV} 和三层最大池化层 L^{MP} 生成栅格图 ($\mathbf{M}_{t-2}, \mathbf{M}_{t-1}, \mathbf{M}_t$) 的特征图。输入大小为 $(N, C^{l-1}, H^{l-1}, W^{l-1})$ 和输出大小为 (N, C^l, H^l, W^l) 的二维卷积层 L^{CV} 可以精确地描述为：

$$\mathbf{x}^l(N_i, C_j^l) = \text{ReLU}\left(\sum_{k=0}^{C^{l-1}-1} \mathbf{W}^l(C_j^l, k) \star \mathbf{x}^{l-1}(N_i, k) + \mathbf{b}^l(C_j^l)\right),$$

其中 $\text{ReLU}(x) = \max(0, x)$ 表示校正线性单位函数^[130]， \star 是有效的 2D 互相关运算符， \mathbf{x}^l 是网络层 l 的输出张量， N 是批处理大小， C 表示通道数量， H 是输入高度（以像素为单位）， W 是宽度（以像素为单位）。输入大小为 (N, C, H^{l-1}, W^{l-1}) ，输出大小为 (N, C, H^l, W^l) ，内核大小 (kH, kW) 的 2D 最大池化层 L^{MP} 可以精确地描述为：

$$\mathbf{x}^l(N_i, C_j, h, w) = \max_{m=0}^{kH-1} \max_{n=0}^{kW-1} \mathbf{x}^{l-1}(N_i, C_j, s_0 \times h + m, s_1 \times w + n), \quad m, n \in \mathbb{N},$$

其中 (s_0, s_1) 表示要计算最大值的窗口的跨度， $h \in [0, H^l]$ ， $w \in [0, W^l]$ 。第一个 2D 卷积层 L_0^{CV} 包含 64 个二维过滤器（Conv2D）对三个栅格图进行卷积，内核为 3×3 ，跨度为 1，并应用 ReLU 激活，然后输出再由最大池层 L_0^{MP} 处理。第二层 L_1^{CV} 包含 128 个二维过滤器，内核大小 3×3 ，跨度为 1，然后再紧接着一个内核大小为 2×2 ，跨度为 2 的最大池层 L_1^{MP} 。第三层 L_2^{CV} 则包含 256 个二维

过滤器，内核大小为 3×3 ，跨步为 1，然后再跟一个最大池层 L_2^{MP} ，它的内核大小也为 2×2 ，跨步为 2。然后，这些特征图由具有 512 单元的全连接层 L_0^{FC} 转换为 512 维矢量单位。全连接层 L^{FC} 对输入数据 \mathbf{x}^{l-1} 进行如下线性变换：

$$\mathbf{x}^l = \text{ReLU}(\mathbf{W}^l \mathbf{x}^{l-1} + \mathbf{b}^l).$$

该网络还包含三个局部目标和偏移定向角的输入，即 $(\mathbf{g}_{t-2}, \mathbf{g}_{t-1}, \mathbf{g}_t)$ 和 $(\alpha_{t-2}, \alpha_{t-1}, \alpha_t)$ 。网络将这两个向量组合在一起，并将它们输送到具有 512 个单位的两个全连接层 L_1^{FC} 和 L_2^{FC} 。然后网络应用具有 2 个单元的全连接层 L_3^{FC} 而不使用激活函数，以产生输出，即动作的平均值 $\mathbf{a}_t^{mean} = (v_t^{mean}, \omega_t^{mean})$ ，其中 v_t^{mean} 是线速度的平均值，而 ω_t^{mean} 是角速度的平均值。整个策略网络 π_θ 可以表示为：

$$\mathbf{a}_t^{mean} = \prod_{k=1}^3 L_k^{FC}(L_0^{FC}(\prod_{l=0}^2 L_l^{MP} \cdot L_l^{CV}(\vec{\mathbf{M}}_t)) \oplus \vec{\mathbf{g}}_t \oplus \vec{\alpha}_t),$$

其中 \oplus 是拼接运算符。最后，从高斯分布 $\mathcal{N}(\mathbf{a}_t^{mean}, \mathbf{a}_t^{logstd})$ 中采样网络的输出，即机器人控制动作。其中 \mathbf{a}_t^{logstd} 是引用对数标准偏差的一组独立参数，这些参数将在训练过程中进行更新。我们还使用剪切函数来确保生成的动作在有效的动作空间中。值网络与策略网络具有相同的体系结构，不同之处在于最后一层被修改为仅输出状态值 $V_\phi(\vec{\mathbf{M}}_t, \vec{\mathbf{g}}_t, \vec{\alpha}_t)$ 。

对于网络超参数的设计，我们参考了现有相关工作并在随机场景中做了测试。如图3.7所示为我们在仿真器中测试不同输入帧数与网络层数对训练效果影响的对比结果。通过实验我们发现三帧状态输入和三层卷积网络架构比简单的一帧状态输入和一层网络架构要好。

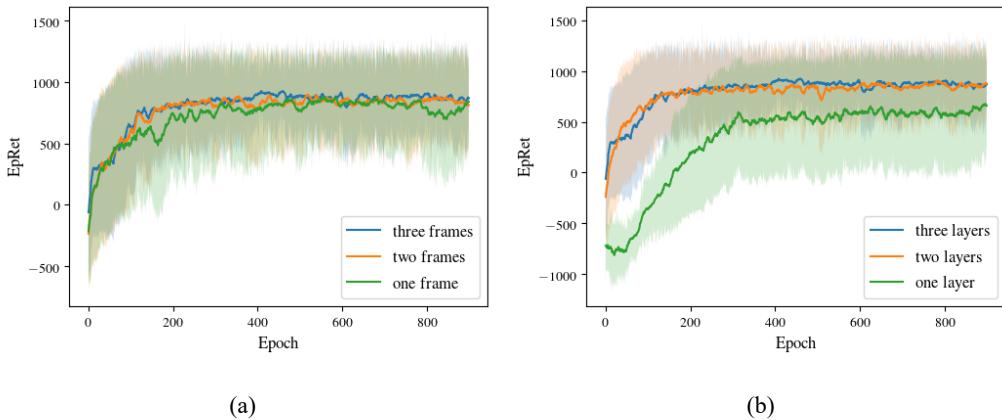


图 3.7 输入帧数 (a) 与网络层数 (b) 分别对训练效果的影响

3.4.3 多阶段课程学习

课程学习^[131]旨在通过为从简单到困难的进阶学习者设计适当的课程来提高学习成绩。Elman 等人^[132]提出了这样的课程学习想法，即逐渐艰巨的任务课

程可以显著加速神经网络的训练。课程学习最近在机器学习领域变得越来越普遍，它假定课程学习可以提高训练过程的收敛速度，并且可以找到比现有求解器更好的局部最小值。可以在文献^[131]中找到该想法的正式数学表达。步骤 λ 的相应训练分布为：

$$Q_\lambda(z) \propto W_\lambda(z)P(z) \quad \forall z,$$

其中 z 是代表学习者示例的随机变量， $P(z)$ 代表目标训练分布， $W_\lambda(z) \in [0, 1]$ 是在课程顺序中 $\lambda \in [0, 1]$ 步骤所应用到实例 z 的权重， $W_1(z) = 1$ 。如果这些分布的熵增加，则分布 Q_λ 的序列称为一个课程，

$$H(Q_\lambda) < H(Q_{\lambda+\xi}) \quad \forall \xi > 0,$$

并且 $W_\lambda(z)$ 单调递增于 λ ，

$$W_{\lambda+\xi}(z) \geq W_\lambda(z) \quad \forall z, \forall \xi > 0.$$

熵的增加意味着训练示例的多样性，我们希望特定示例的权重随着其添加到训练集中而增加。课程学习策略可以在常规任务中分两步完成：首先是一组简单的示例，然后是目标训练集。

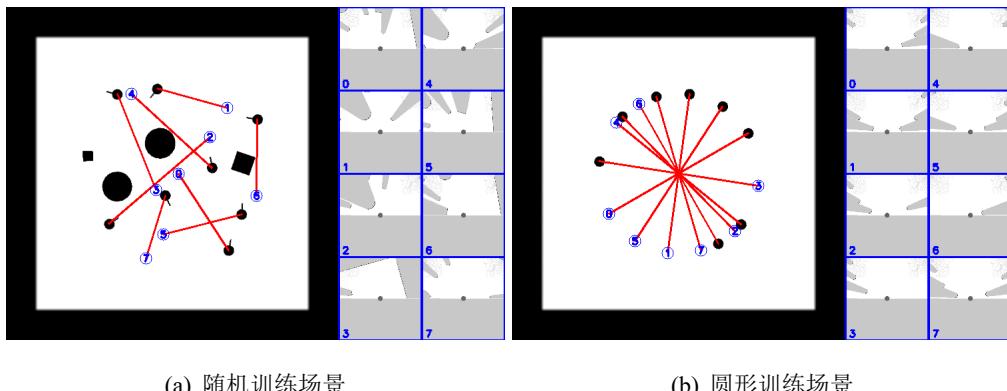


图 3.8 用于收集多个机器人避障经验的两种场景，其中蓝色数字圆圈代表具有相应编号的机器人的目标，红色直线代表机器人从当前位置到目标的连线，其他黑色像素代表各种障碍。右侧的蓝色框显示了每个机器人的以自我为中心的局部栅格图。

在本文中，我们引入了课程学习的两阶段训练过程。具体来说，我们提出了两种场景，即随机训练场景和圆形训练场景，它们在我们的自定义仿真器中生成随机环境（如图3.8中所示），以收集多个机器人的避障经验。图3.8(a)展示了随机选择障碍物位置、机器人的起始位置和目标位置的随机训练环境，这将有助于机器人充分学习避开障碍物的能力。随机场景中的环境是在 $6 \times 6 m^2$ 空间中构建的，其中有八个机器人和四个障碍物，每个机器人的目标位置是在其起点距离 2m 到 4m 范围内随机生成的。图3.8(b)说明了将机器人随机放置在具有随机

半径圆上的环境（也称为圆形训练场景），这有助于该机器人与其他机器人进行交互。圆形训练场景中的环境将八个机器人随机放置在一个半径范围为 1.8m 到 3.0m 的圆上。机器人可以在这些复杂丰富且具有代表性的训练环境中充分探索其观察空间，有利于提高避障策略的鲁棒性。同时按照集中学习和分散执行机制，机器人在每次探索训练迭代中都有效地优化了避障策略。

遵循课程学习的思想，我们将训练过程分为两个阶段。在第一阶段，我们在具有八个机器人和四个障碍物的随机训练场景（如图3.8(a)中所示）的环境中训练策略。这使机器人可以快速学习到初级的避障策略。一旦该避障策略达到可接受的效果，我们将保存学习到的策略，然后进入第二阶段。在第二阶段中，我们同时在随机训练场景和圆形训练场景（如图3.8(b)中所示）的环境中训练策略，其中机器人数量增加到 16。

图3.9(a)显示了消融研究的学习曲线，以说明两阶段课程学习的效果。其中“阶段 1”(Stage-1) 表示之前所述的第一阶段训练过程，“阶段 2”(Stage-2) 表示之前所述的第二阶段训练过程，“从头开始”(From scratch) 表示不使用两阶段课程学习而直接从最开始就在两种训练场景中训练的方式。从训练曲线我们可以清晰地看出，两阶段课程学习可以帮助策略收敛于更高的预期回报，即更好的避障性能。

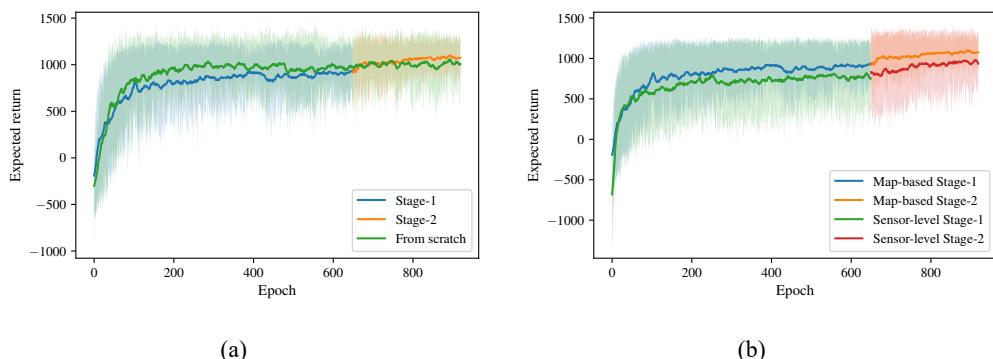


图 3.9 (a) 两阶段课程学习的训练过程和没有使用课程学习的训练过程（即从头开始）的期望回报。(b) 基于地图的策略训练过程和传感器级策略的训练过程的期望回报。

3.5 仿真环境实验

在本节中，我们将在多种模拟环境中评估基于栅格地图的多机器人避障方法。我们首先叙述实现的细节，包括定制的仿真器、使用的超参数和用于训练网络的硬件和软件。然后，我们在多种仿真环境中定量评估基于栅格地图的多机器人避障策略的性能，并将其与其他现有方法进行比较。有关实验中机器人的具体表现，请参见<https://www.bilibili.com/video/BV12f4y1Q7cx/>上的演示视频。

表3.1 对比实验中 NH-ORCA 算法的超参数设置

参数	值
VO 类型	HRVO
是否使用 ORCA	是
是否使用 clearpath	是
Epsilon	0.1
时间范围 (Time horizon)	10
完整约束时间 (Time to holonomic)	0.4
最小追踪误差 (Minimum tracking error)	0.02
最大追踪误差 (Maximum tracking error)	0.1

在实验中，我们在多种情况下比较了三种不同的避障方法并评估这些方法的性能。具体来说，我们考虑以下避障策略：

- NH-ORCA 策略: Alonso-Mora 等人^[105-106] 提出的最新的基于规则的智能体级多机器人避障策略。表3.1中列出了对比实验中使用的 NH-ORCA 算法的超参数。有关每个参数的详细定义与解释，请参考文献^[101]。
- 传感器级策略: Long 等人^[115] 和 Fan 等人^[116] 提出的基于 DRL 的多机器人避障策略。与我们的方法不同，他们的网络将原始 2D 激光数据（以 1D 形式）视为输入，并使用一维卷积处理输入。为了进行公平的比较，该传感器级方法使用了与我们的方法相同的训练过程（即两阶段训练过程）。图3.9(b)中显示了其训练过程的学习曲线，其中我们将他们的方法称为“传感器级”(Sensor-level)，将我们的方法称为“基于地图”(Map-based)。需要注意的是，我们的方法在训练过程中收敛到了更高的期望回报。
- 基于地图的策略: 本文中我们提出的多机器人避障策略，该策略以自我为中心的本地栅格图视为输入。

为了进一步评估我们的方法中的两阶段课程学习的作用，我们将在第一阶段训练后学到的避障策略记为“基于地图，阶段 1”，在第二阶段之后学到的避障策略称作“基于地图，阶段 2”。

我们从不同的角度比较了上述三种避障策略，包括泛化到看不见的场景、导航的效率以及对机器人密度以及机器人的各种形状和动力学的鲁棒性。这些实验表明，我们提出的基于栅格地图的避障方法在许多指标上都优于其他方法。

3.5.1 超参数设置

我们根据算法3.1和算法3.2中的 DPPO 算法，使用表3.2中列出的超参数训练了基于地图的避障策略。

表 3.2 训练算法3.1和算法3.2中的超参数设置

参数	值
算法3.1第 5 行的 T_{ep}	2000
算法3.1第 12 行的 T_m	200
算法3.1第 14 行的 d_{arr}	0.2
算法3.1第 19 行的 λ	0.95
算法3.1第 19 行和算法3.2第 11 行的 γ	0.99
算法3.2第 2 行的 E_π	80
算法3.2第 3 行的 ϵ	0.2
算法3.2第 4 行的 KL_{target}	0.01
算法3.2第 7 行的 lr_θ	3.0×10^{-4} (阶段 1), 1.0×10^{-4} (阶段 2)
算法3.2第 10 行的 E_v	80
算法3.2第 12 行的 lr_ϕ	1.0×10^{-3}

3.5.2 实现细节

与我们以前的工作^[133] 和其他大多数使用 Gazebo^[134] 或 Stage^[135] 作为仿真器的相关工作不同，我们的训练环境是基于 OpenCV^①定制的。具体而言，仿真器可以将环境地图加载为灰色图像，其中障碍物和机器人被表示为图像中的相应像素。此外，碰撞的发生可以根据这些像素是否重合来判断。与其他仿真器（如 Gazebo）不同，我们定制的仿真器高效、灵活且不用考虑通信延迟。

我们还在仿真器中实现了差动驱动机器人，其位姿 (x, y, θ) 根据差动轮的速度运动模型进行更新：

$$\begin{pmatrix} x' \\ y' \\ \theta' \end{pmatrix} = \begin{pmatrix} x \\ y \\ \theta \end{pmatrix} + \begin{pmatrix} -\frac{v}{\omega} \sin \theta + \frac{v}{\omega} \sin(\theta + \omega \Delta t) \\ \frac{v}{\omega} \cos \theta - \frac{v}{\omega} \cos(\theta + \omega \Delta t) \\ \omega \Delta t \end{pmatrix},$$

其中 Δt 是速度控制的间隔时间， v 和 ω 分别表示线速度和角速度。在每个时间步 t ，我们根据机器人 i 的最新位姿截取世界地图以生成以自我为中心的栅格图的草图 $\bar{\mathbf{M}}_t^i$ 。然后，我们使用布雷森汉姆直线算法^[136] 从激光扫描仪估算出以自我为中心的局部栅格图 \mathbf{M}_t^i 。需要注意的是，在训练过程中，我们不会在仿真器

^①<https://opencv.org/>

构建的环境中添加任何噪声，这有助于我们优化策略到低方差水平。

策略网络和值网络都用 TensorFlow^②实现，并使用 Adam 优化器^[129]进行训练。训练使用配备 i7-9900 CPU 和英伟达 Titan RTX GPU 的计算机。在算法3.1和算法3.2中运行大约 900 次迭代需要大约 12 个小时来训练网络以在所有训练场景中收敛。如表3.2中表述的那样，策略网络的学习率 lr_θ 在第一个训练阶段设置为 3×10^{-4} ，然后在第二个训练阶段中降低为 1×10^{-4} 。

可以在仿真和现实世界中的机器人上实时执行学习策略。在仿真器中，策略网络在 GPU 上花费 2 毫秒为 8 个机器人计算新动作。在实际工作中将策略部署到机器人之后，在英伟达 Jetson TX2 上花费约 25 毫秒用于策略网络计算新动作，并花费约 30 毫秒从 2D 激光扫描数据生成以自我为中心的局部栅格图。

3.5.3 泛化能力测试

我们仅使用随机场景和圆形场景中的环境来训练基于栅格地图的避障策略。在本小节中，通过展示所学习的策略在一系列未见过的场景中表现良好，并且在训练中引入了不同大小和形状的机器人之后，对于异类机器人也表现良好，我们证明了我们方法的泛化能力。

1. 非合作多机器人环境

首先，我们介绍一种包含非协作机器人的场景，即场景中包含不受其他机器人影响的移动机器人。图3.10展示了此场景的三种环境，环境中部署了由我们的避障策略控制的八台机器人，其中的黄色机器人沿直线以 0.5 m/s 的速度运动，不受其他机器人影响。实验结果表明，尽管在训练过程中没有引入非合作机器人，但是学习的策略仍然可以使其他机器人避免与之碰撞。

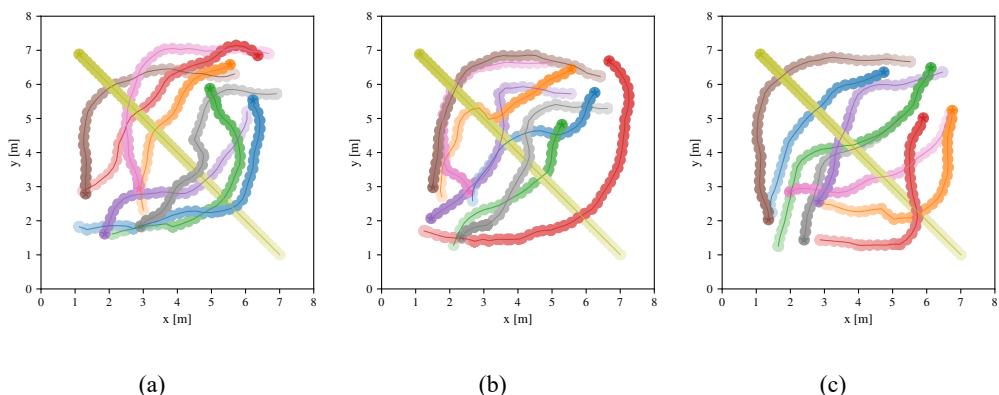


图 3.10 非协作机器人场景测试实验，非合作型机器人（黄色）以 0.5 m/s 的速度沿直线运动，其他颜色代表的机器人由相同的学习到的避障策略控制。

^②<https://www.tensorflow.org/>

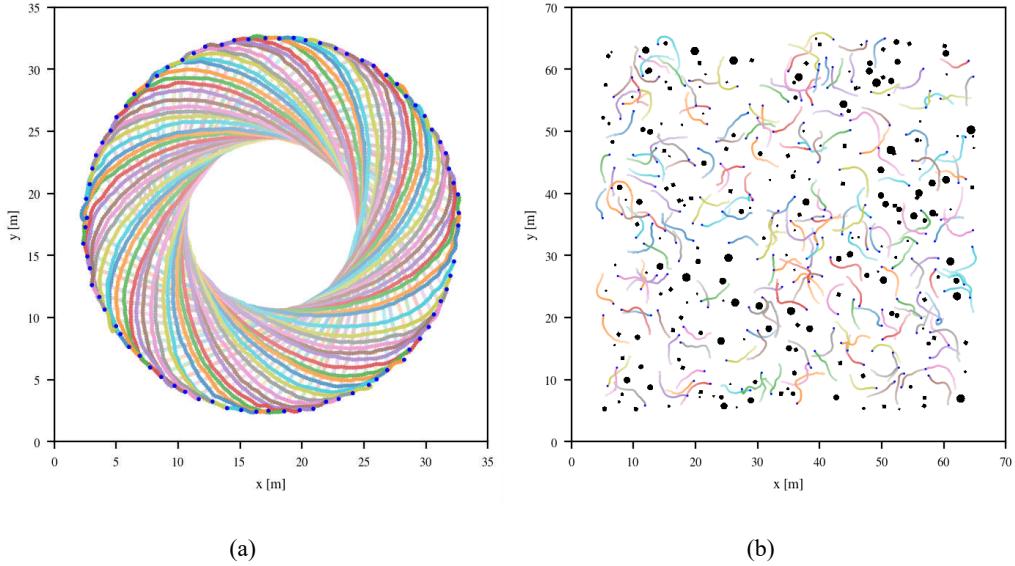


图 3.11 (a) 从圆形场景扩展而来的大规模场景，该场景将 80 个机器人都均匀地放置在一个大圆圈上，并要求它们移动到对面的位置。蓝点表示机器人的目标位置。(b) 从随机场景扩展而来的大型规模场景，该场景将 200 个机器人和 200 个大小不同的障碍物随机放置在 $60 \times 60 \text{ m}^2$ 的区域中，并要求他们移动到相对于相应的起始位置在 4.5 m 至 5 m 范围内随机选择的目标位置。蓝点也是表示机器人的目标位置。

2. 超大规模环境

我们还介绍了两个大规模场景来说明所学策略的泛化能力。具体而言，第一个大规模场景是从圆形场景扩展而来的，该场景将 80 个机器人都均匀地放置在一个大圆上，并要求它们移动到圆心的对立圆上位置。图3.11(a)中展示了该场景的环境。第二个大规模场景是从随机场景扩展而来的，该场景随机将 200 个机器人和 200 个大小不同的障碍物放置在 $60 \times 60 \text{ m}^2$ 的区域中，并要求他们移动到相对于相应的起始位置 4.5 m 到 5 m 范围内随机选择的目标点。图3.11(b)中展示了该场景的环境。

实验结果表明，尽管学习策略在训练过程中没有遇到这两种大规模场景，但在两种场景的环境中仍可以很好地运行而无需进行微调。

3. 异质多机器人环境

最后，我们介绍一种含有异质机器人的场景，即具有不同大小和形状的机器人。图3.12展示了该场景的六个环境，其中包含三个矩形机器人，三个圆形机器人以及具有不同形状和大小的障碍物。

实验结果表明，在训练中引入不同形状的相应机器人后，所学习的策略在这种情况下的环境中也能表现良好。这项实验说明我们的避障策略可以兼容不同形状的机器人或障碍物，同时异质机器人也可以等同于不同形状大小的移动障碍物，这对于增加训练环境的多样性具有重要意义。

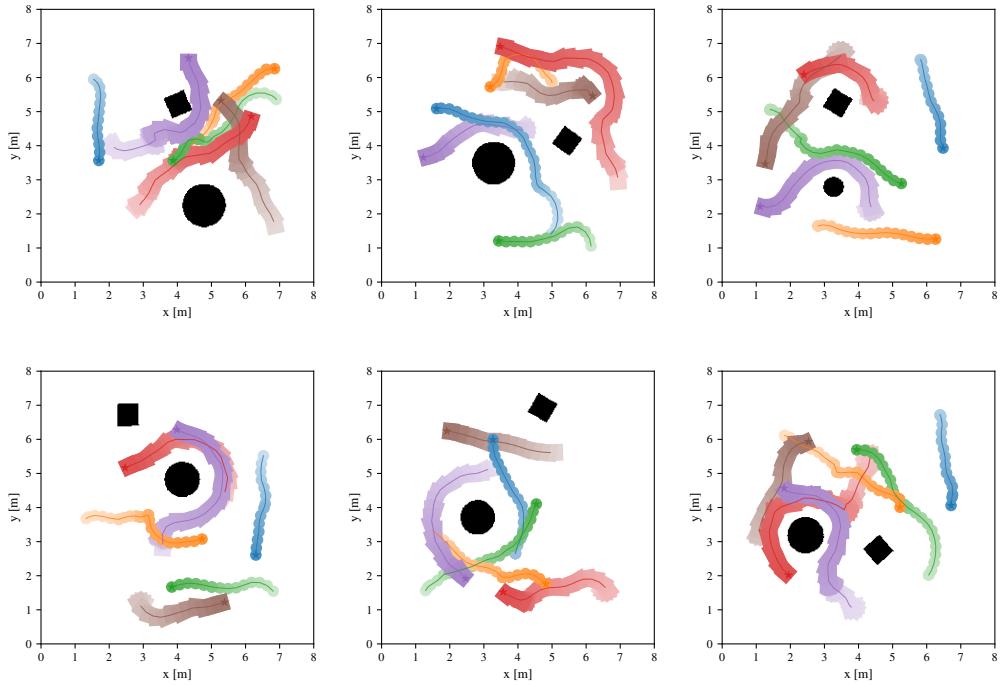


图 3.12 包含异质机器人的场景，其中包含三个矩形机器人，三个圆形机器人以及具有不同形状的障碍物。所有机器人都受相同的避障策略控制。

3.5.4 性能对比实验

在本小节中，我们评估避障策略的导航效率。我们首先介绍用于评估避障方法效果的一些指标。然后，我们使用这些指标比较不同的避障策略。

1. 测试指标和环境

我们引入了四个指标来评估不同的方法，指标定义如下：

- 成功率 π : 机器人到达目标点且没有任何碰撞的回合比率。
- 额外时间 \bar{t} : 每个机器人在没有任何碰撞的情况下成功到达目标点所需的平均时间减去每个机器人以最大速度直达目标点的平均时间的差值。
- 额外距离 \bar{d} : 每个机器人在没有任何碰撞的情况下成功达到其目标点所需的平均移动距离减去每个机器人以最大速度直驶其目标点的平均移动距离的差值。
- 平均线速度 \bar{v} : 导航期间每个机器人的平均线速度。

我们将比较四种策略的导航效率，即 NH-ORCA 策略、传感器级策略、基于地图的阶段 1 策略和基于地图的阶段 2 策略。所有这些指标都是根据每种情况下测试场景的 100 个不同环境的平均结果计算得出的。

现在，我们介绍本文中使用的测试场景。

- 圆形场景：与圆形训练场景相似的场景，只是所有机器人的初始位置均匀地放置在圆上。根据机器人的数量和圆形半径的不同，可以将这种场景分

为四种类型，即 6 个机器人半径为 2.5m 的场景，8 个机器人半径为 3m 的场景，10 个机器人半径为 3.5m 的场景和 12 个机器人半径为 3.5m 的场景。

- 交叉场景：两组机器人（每组四个机器人）相互交叉移动的场景。
- 交换场景：两组机器人（每组四个机器人）彼此相对并交换位置的场景。
- 随机场景：该场景与随机训练场景相似，不同之处在于该场景中仅包含 10 个机器人，不包含障碍物。

表 3.3 在含有不同数量的机器人不同大小的场景中，针对不同方法评估的性能指标（均值/标准差）。

场景(机器人, 范围)	方法	$\bar{\pi}$	\bar{t} (均值/标准差)	\bar{d} (均值/标准差)	\bar{v} (均值/标准差)
圆形场景 (6, 半径 2.5m)	NH-ORCA	0.969	2.6676/1.3981	0.2004/0.1160	0.4490/0.1537
	传感器级	1.000	2.0620/0.5576	0.8773/0.2269	0.5636/0.1328
	基于地图, 阶段 1	0.937	8.2528/6.4266	0.7861/0.4763	0.3328/0.2881
	基于地图, 阶段 2	1.000	2.0000/0.3502	0.8648/0.1447	0.5659/0.1283
圆形场景 (8, 半径 3m)	NH-ORCA	0.950	3.4988/1.9744	0.2057/0.1299	0.4479/0.1520
	传感器级	1.000	2.5400/0.5084	1.1992/0.1918	0.5687/0.1233
	基于地图, 阶段 1	0.914	10.3488/6.3236	0.9185/0.5446	0.3218/0.2880
	基于地图, 阶段 2	1.000	2.3170/0.2577	1.0204/0.1513	0.5730/0.1146
圆形场景 (10, 半径 3.5m)	NH-ORCA	0.892	4.2930/2.6132	0.2486/0.1983	0.4366/0.1546
	传感器级	1.000	3.3045/0.4784	1.5991/0.2145	0.5734/0.1142
	基于地图, 阶段 1	0.903	11.9304/9.2772	1.0635/0.6968	0.3212/0.2867
	基于地图, 阶段 2	1.000	2.5881/0.4650	1.1870/0.1710	0.5735/0.1114
圆形场景 (12, 半径 3.5m)	NH-ORCA	0.862	5.2137/3.4742	0.2817/0.2599	0.4078/0.1711
	传感器级	1.000	3.7290/0.5355	1.7884/0.2525	0.5699/0.1170
	基于地图, 阶段 1	0.873	15.7697/11.7475	1.0773/0.7475	0.2698/0.2871
	基于地图, 阶段 2	1.000	2.6133/0.4527	1.2170/0.1769	0.5745/0.1120
交叉场景 (8, 8 × 8m ²)	NH-ORCA	0.958	2.1283/1.5166	0.1883/0.2081	0.4851/0.1430
	传感器级	0.995	2.8238/1.2894	1.1174/0.5214	0.5419/0.1588
	基于地图, 阶段 1	0.950	4.0802/3.4952	1.0158/0.7322	0.4764/0.2278
	基于地图, 阶段 2	1.000	1.8315/1.2333	0.7873/0.4912	0.5608/0.1384
交换场景 (8, 8 × 8m ²)	NH-ORCA	0.906	2.2174/2.1307	0.2651/0.2228	0.4845/0.1648
	传感器级	1.000	2.7357/0.9494	1.1498/0.3479	0.5535/0.1419
	基于地图, 阶段 1	0.994	2.7272/2.2479	0.8017/0.5761	0.5206/0.1874
	基于地图, 阶段 2	1.000	2.0201/1.0430	0.9816/0.3660	0.5584/0.1424
随机场景 (10, 8 × 8m ²)	NH-ORCA	0.934	4.3181/3.1353	0.5697/0.6412	0.3760/0.1890
	传感器级	0.924	3.4519/3.4162	0.5417/0.5048	0.4017/0.2687
	基于地图, 阶段 1	0.955	3.1650/2.5632	0.5514/0.4643	0.4202/0.2590
	基于地图, 阶段 2	0.986	2.9009/2.4523	0.4531/0.3610	0.4460/0.2497

2. 测试结果

表3.3中总结了在不同测试场景中四种方法的性能指标，其中指标数值是根据每种场景随机生成的100个不同环境的平均结果统计得出的。我们可以看出基于地图的阶段2策略在额外时间和平均速度方面优于其他策略。在许多情况下，基于地图的阶段2策略和传感器级策略的成功率均为100%。但是，与基于地图的阶段2策略相比，传感器级策略在其他指标中表现不佳，这表明可以通过基于地图的方法进一步优化传感器级方法学习的策略。NH-ORCA策略是一种非合作策略，其中每个机器人贪婪地执行运动会导致较小的额外距离（如图3.13中第一行所示），但是其他指标的性能较差。对于基于地图的阶段1策略，我们可以看到它在迁移到圆形场景方面具有良好的性能，这表明该策略可以泛化到未见的场景，因为在第一训练阶段我们没有考虑圆形场景。需要注意的是，经过第二阶段的训练后，该策略的性能大大提高了，到达率等各项指标都有明显提升。最后我们分别在图3.13第二行和第三行的测试场景中列举了传感器级策略和基于地图的阶段2策略的实际测试轨迹。

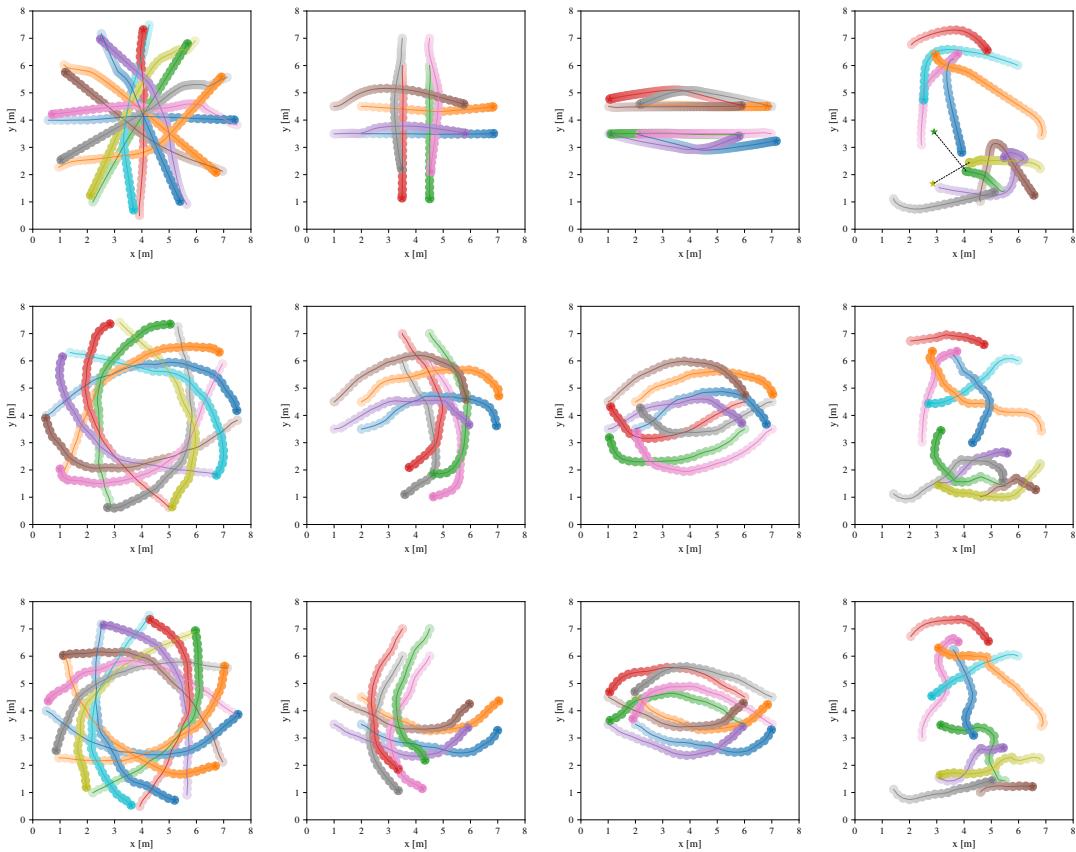


图3.13 执行不同避障策略的机器人轨迹图（第一行：NH-ORCA策略，第二行：传感器级策略，第三行：基于地图阶段2策略）。不同机器人的轨迹以不同的颜色区分，并且颜色透明性用于指示沿每个轨迹的时间状态。目标点用星形表示，第四张图中的黑色虚线连接机器人碰撞位置与目标点。

3.5.5 鲁棒性测试

除了泛化能力和导航效率外，学习的策略还对模型不确定性和输入噪声稳定且健壮。我们将在本节中通过多项实验来验证所学策略的鲁棒性。

1. 传感器噪音

传感器噪音和感知错误在机器人系统中很常见。为了验证不同避障方法抵御噪音的能力，我们逐渐增加了传感器级策略和基于地图的（阶段2）策略的激光感应距离数据的误差，并对NH-ORCA策略逐渐增加了周围机器人位置和速度的感知误差。图3.14描述了当高斯噪音方差增加时，在具有12个机器人的圆形测试场景和具有10个机器人的随机测试场景中，上述三种策略的成功率。结果表明，基于深度强化学习的策略（即基于地图的策略和传感器级别的策略）对噪音具有抵抗能力，而传感器噪音会极大地影响NH-ORCA策略的成功率。

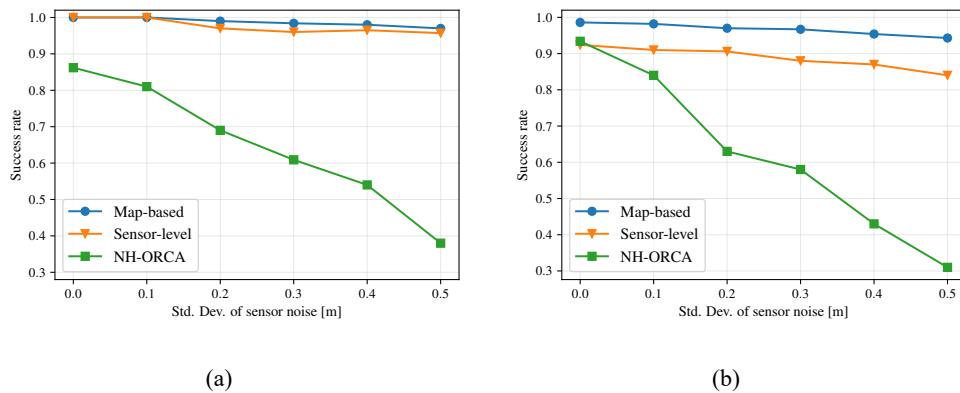


图3.14 (a) 当传感器噪音的标准偏差增加时，在具有12个机器人的圆形测试场景中，比较NH-ORCA策略、传感器级别策略和基于地图的策略的成功率。(b) 当传感器噪音的标准偏差增加时，在具有10个机器人的随机测试场景中，基于地图的策略、传感器级别的策略和NH-ORCA策略之间的成功率比较。

2. 传感器类型

如第3.1节所述，基于栅格地图的避障方法可以轻松地部署到具有不同传感器的机器人上。在这项实验中，我们通过将避障方法应用于仅装有深度摄像头的机器人来验证基于栅格地图的避障方法的兼容性。由于我们的自定义仿真器无法仿真3D传感器信息，因此我们选择更加逼真的3D仿真器Gazebo来测试模型的泛化能力。具体来说，我们在Gazebo仿真器中实现了带有深度摄像头的差动轮机器人，该摄像头具有120度水平FOV和640×480的分辨率。我们将深度相机生成的点云数据转换为相应的栅格图。我们首先根据地图的分辨率将点云数据下采样为稀疏的点云数据，去除异常值，然后从生成点云的鸟瞰图生成栅格图。图3.15展示了从点云数据生成此类栅格图的过程。使用生成的栅格图，我们可以构造相应的以自我为中心的局部栅格图，该栅格图稍后可用于基于地图的

避障策略。图3.16展示了在具有8个随机放置的障碍物环境中使用深度摄像头测试基于地图避障策略的效果。可以发现仅使用深度摄像头机器人就可以成功避开所有障碍物并到达相应目标点。我们认为我们提出的基于地图的深度强化学习避障方法不仅仅限于二维激光测距仪和深度相机，还可以推广到如2.2章节提到的其他多种测距传感器或间接提供测距信息的感知情况^[137]，甚至是不同种类和数量的传感器组合。

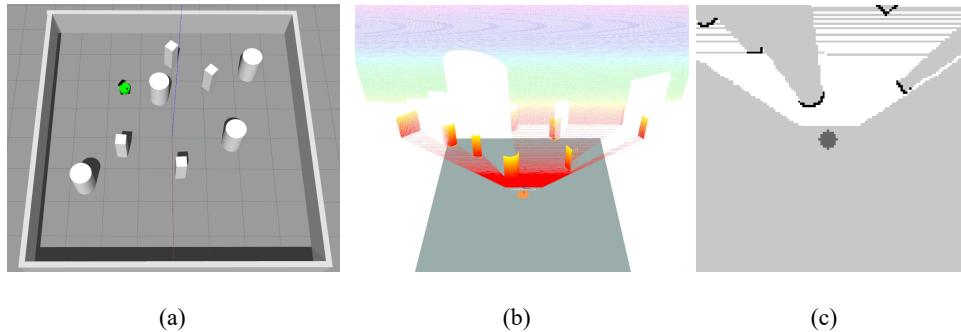


图 3.15 (a) 包含装备深度摄像头的机器人（绿色）的 Gazebo 环境；(b) 由 120 度 FOV 深度相机生成的点云数据；(c) 从点云数据生成的栅格图。

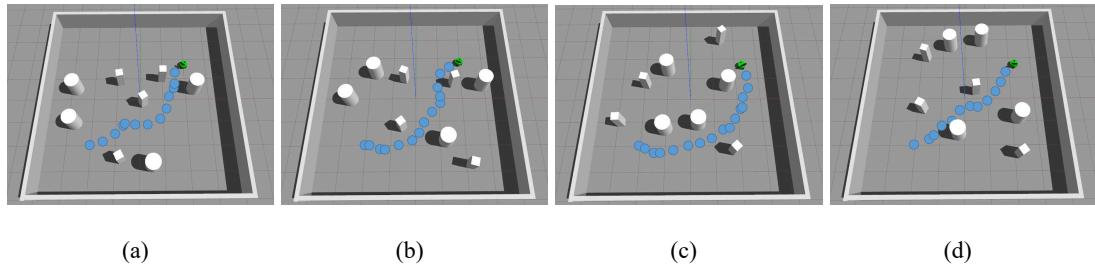


图 3.16 在具有 8 个随机放置障碍物的四个仿真环境 (a-d) 中，使用深度相机通过基于地图的学习策略生成的机器人轨迹示意图。

3. 视野大小

不同的传感器具有不同的视野大小，这也极大地影响了避障策略的性能。我们考虑了五种具有不同视野大小的激光传感器，并将基于地图的深度强化学习避障算法应用于各种传感器数据，以训练相应的基于地图的避障策略。图3.17(a)中展示了这些基于地图避障策略的学习曲线，图3.17(b)展示了这些策略在随机测试场景中的成功率。此外，我们还在视野大小受限的机器人上测试了传感器以 180 度视野大小训练的避障策略的迁移效果。结果表明，随着机器人传感器视野变大，机器人学到的避障策略效果更好。我们使用大视野传感器训练的策略迁移到小视野传感器时，避障成功率会降低。而且当机器人在训练和测试场景中具有相同的视野大小时，训练得到的避障策略性能会比使用大视野传感器训练的策略直接迁移到小视野传感器环境要好。

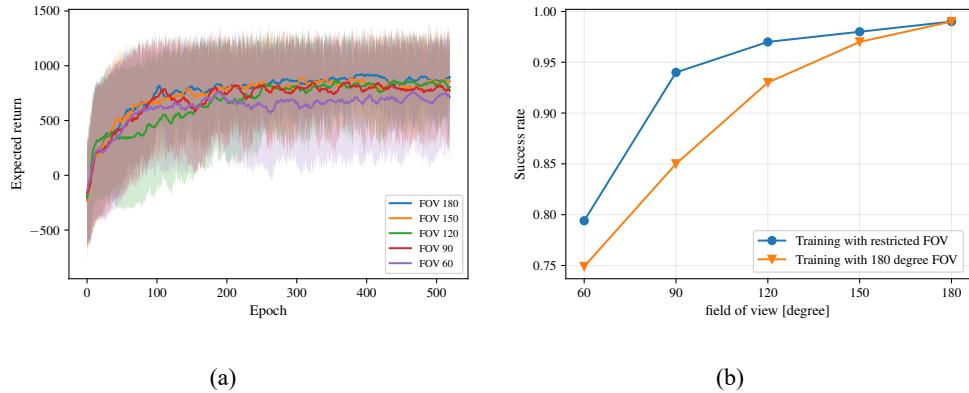


图 3.17 (a) 基于地图的避障策略对于具有不同视野大小 (FOV) 的传感器的期望回报。(b) 使用特定的受限 FOV 传感器训练的基于地图的策略与使用 180 度 FOV 传感器训练的基于地图的策略在随机测试场景中的成功率。

4. 局部极小问题

由于避障算法通常仅使用局部感知信息，机器人很容易陷入到困境区域（如 U 形环境），这称为局部极小问题^[138]。在本小节中，我们将测试我们的方法在一些复杂环境中是否具有探索未知区域并逃脱困境的能力。我们在如图3.18(a-b) 所示的复杂静态场景中训练了我们提出的基于栅格地图的强化学习避障算法，该训练场景包含大小和位置随机的两个矩形和四个圆形障碍物，机器人的起点和终点在每个回合也随机生成。训练过程曲线如图3.18(c) 所示，在该训练场景中我们达到了 92% 的成功到达率。然后我们在三种类型的测试场景中测试了我们的算法解决复杂未知静态环境的能力。首先是包含 16 个圆形障碍物和 16 个矩形障碍物的复杂随机环境，机器人成功到达率为 85%，如图3.19(a-g) 所示，我们列举了其中七种随机测试环境，我们可以看出机器人具有很强的探索未知环境并摆脱困境到达目标点的能力。最后我们在另外的螺旋场景和之字形场景做了测试，实验效果分别如图3.19(h) 和 (i) 所示。我们发现训练场景的多样性和代表性对学习到的避障能力影响很大，我们使用上文提到的多机器人避障模型在这三种测试场景中表现很差，基本不具备探索解决局部极小问题的能力，这是因为我们的多机器人训练场景很少包含需要机器人探索的复杂情况，我们猜测如果把三种测试场景也纳入训练场景，学习到的避障策略在三种测试场景中会有更好的表现。因此我们认为基于深度强化学习的避障算法应该考虑加入更多种类的代表性训练场景，提升避障模型应对各种复杂环境的能力。局部避障算法在实际使用过程中会配合全局路径规划提高其摆脱困境的能力，近期一些研究人员^[139-140] 开始探讨传统规划算法如何指导强化学习局部避障算法更好地到达目标点。我们认为导航系统其他模块的设计也可以一定程度上缓解局部避障算法引起的局部极小问题。

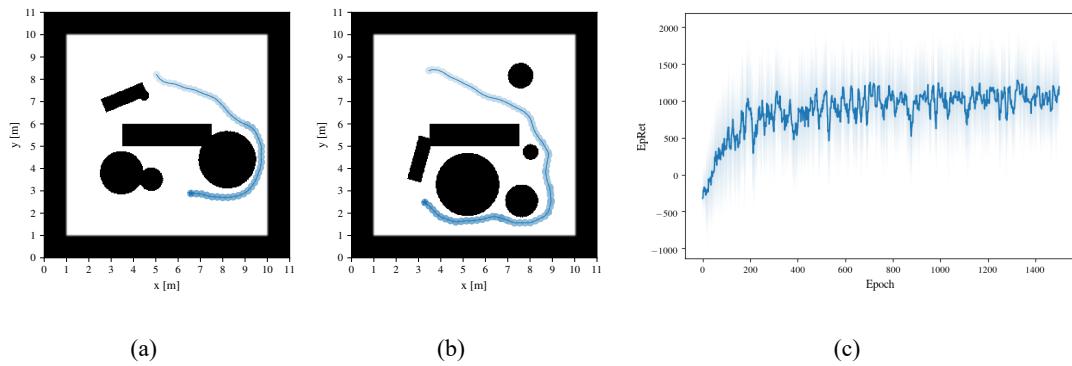


图 3.18 复杂静态训练环境 (a-b) 与训练过程曲线 (c)

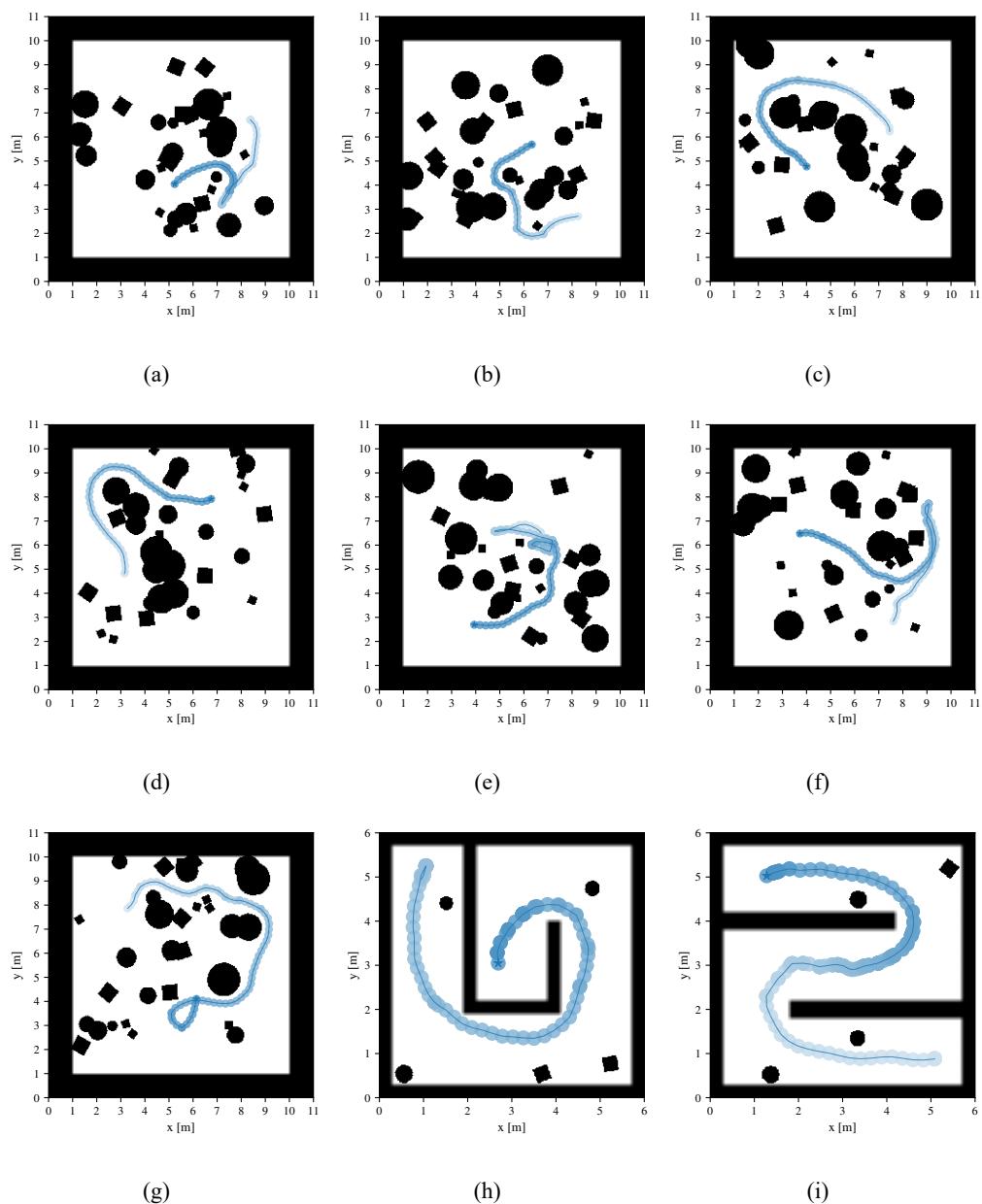


图 3.19 复杂静态测试环境 (a-g) 与螺旋测试场景 (h) 和之字形测试场景 (i)

3.6 真实环境实验

在本节中，我们将训练好的基于地图的避障策略部署到实体机器人上，并在现实世界中测试多机器人避障算法。如第3.1节中所述，基于地图的方法对嘈杂的传感器数据更鲁棒，不需要机器人的运动数据，并考虑了相关机器人的大小和形状，这使其更高效，更容易被部署到真正的机器人上。但是，将学习的避障策略从仿真器部署到现实世界还面临更多挑战。正如在文献^[116]中讨论的那样，除了嘈杂的传感器数据外，每个机器人的时钟都不相互同步，这导致难以控制的异步分布式系统。由于许多现实因素，例如机械细节、电机特性以及摩擦性质的差异，机器人无法通过相同的控制命令提供与仿真环境一致的行为。除了在第3.5节中讨论的在仿真环境中的泛化能力和鲁棒性之外，真实世界的实验还表明，基于地图的方法可以轻松地部署到真实机器人中，并在真实世界中表现良好。有关实验中机器人的具体表现，请参见<https://www.bilibili.com/video/BV12f4y1Q7cx/>页面上的演示视频。接下来，我们首先介绍真实机器人的硬件配置。然后，我们评估这些机器人在现实世界中多种场景下的性能。

3.6.1 实验机器人硬件配置

我们将基于地图的避障策略部署到四个机器人上，以在现实世界中测试多机器人避障策略。如图3.20所示，每个机器人都基于带有Kobuki底盘的TurtleBot 2搭建，并且机器人的形状都是半径为0.17m的圆形，每个机器人使用的激光传感器是Hokuyo UTM-30LX 2D激光或Hokuyo URG-04LX 2D激光（价格较低）。表3.4中列出了Hokuyo 2D激光传感器的详细技术信息，所有这些传感器都可以对机器人前方的周围环境进行180度视野观察。每个机器人搭载一台英伟达Jetson TX2作为其计算平台。具体来说，我们实现了这种机器人的四个版本，其中两个装有Hokuyo UTM-30LX 2D激光，另外两个使用Hokuyo URG-04LX 2D激光。

表 3.4 Hokuyo 二维激光传感器详细参数

产品型号	UTM-30LX	URG-04LX
测量范围	0.1 to 30m, 270°	0.02 to 5.6m, 240°
精度	0.1 to 10m : ±30mm, 10 to 30m : ±50mm	0.06 to 1m : ±30mm, 1 to 4.095m : ±3%
角分辨率	0.25°	0.36°
扫描时间	25ms	100ms

在实验中，机器人的相对局部目标是由基于状态估计的粒子滤波算法提供的。局部中心栅格图是从激光数据生成的，其大小固定为6.0×6.0 m，分辨率

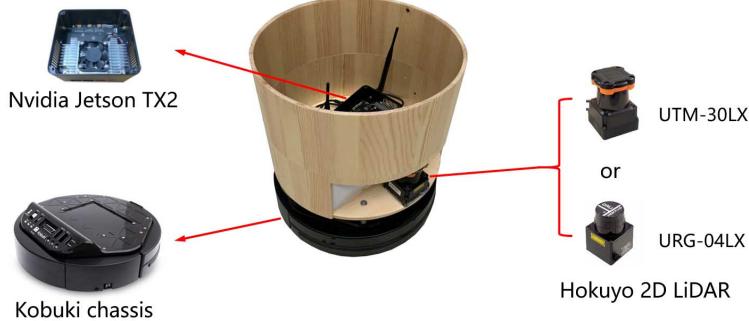


图 3.20 该机器人基于带有 Kobuki 底盘的 TurtleBot 2 搭建，使用 Hokuyo UTM-30LX 2D 激光或 Hokuyo URG-04LX 2D 激光和英伟达 Jetson TX2 计算平台。定制的圆形木壳被安装在移动平台的顶部，以使机器人之间能够相互检测到。

为 0.1 m。机器人木壳放置在机器人的移动平台的顶部，使机器人之间能够相互检测到。我们还使用纸箱和行李箱在测试中充当静态和动态障碍物。训练的策略网络的输出直接用于控制机器人，即，机器人线速度和角速度的范围分别为 $v_t \in [0, 0.6]$ （以米/秒为单位）和 $\omega_t \in [-0.9, 0.9]$ （以弧度每秒为单位）。

3.6.2 障碍物环境测试

在本小节中，我们介绍包含单个机器人的一系列实际障碍物测试场景，以评估由基于栅格地图的避障策略的实际性能。

我们首先介绍一个基础的静态场景，其中一些纸箱被放置为静态障碍物，以阻挡机器人从其起始位置到目标点，图3.21(a)展示了机器人在这种环境中的具体表现。在这种环境中，机器人可以成功寻找到可以通过的狭窄开口并成功绕开障碍物到达目标点。接下来，我们通过在开口的前后位置添加其他静态障碍物来扩展上述场景，以增加避障难度。图3.21(b)展示了这种扩展静态障碍物环境，在这种情况下，机器人仍然可以成功绕开狭窄开口前后放置的障碍物并顺利到达目标点。机器人行驶距离 4.396 米，完成时间为 22.65 秒。然后，我们通过添加一些突然出现的障碍物来实现动态障碍物场景，通过这类实验证明机器人对突发情况的实时反应能力。具体来说，在图3.21(c)所示的环境中，人为将纸板突然放置在机器人的移动路径上对机器人运动产生干扰。图3.21(d)展示了另一个动态障碍物环境，该环境是从图3.21(c)中的情况扩展而来的，其中行李箱在机器人的行进路径上快速移动，并且另一侧也突然出现了人为放置的纸板。在这两种环境中，机器人都能成功检测到动态障碍物并及时调整其运动以避免碰撞发生。需要说明的是，图中描绘的机器人轨迹不那么平滑的一部分原因是由于机器人定位有误差，记录的轨迹数据没有特别精确。在网页演示视频中可以找到有关这些静态和动态障碍物场景的更多细节。这些实体实验表明，我们的避障算法在实际静态和动态复杂障碍物环境中仍能表现良好。

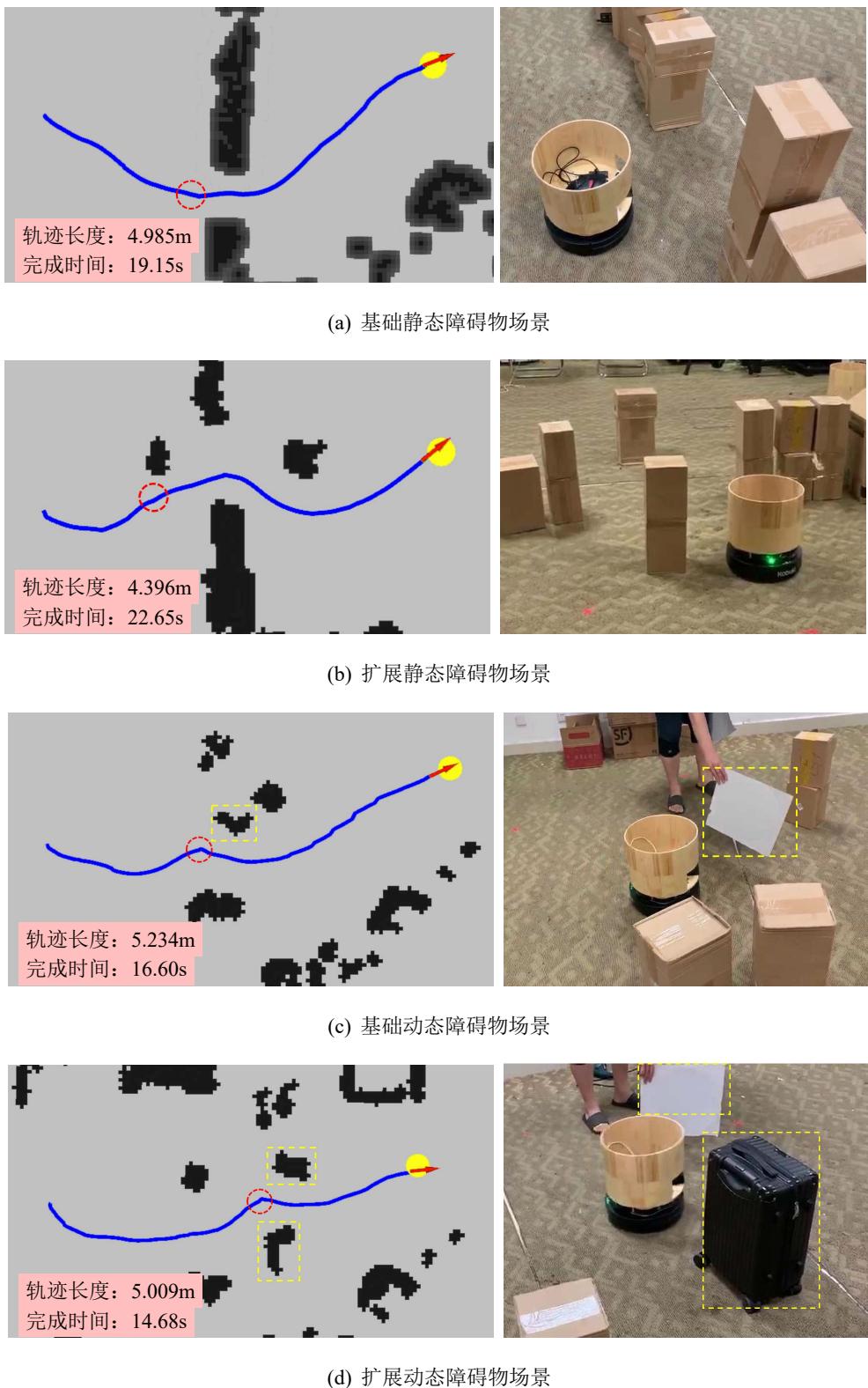


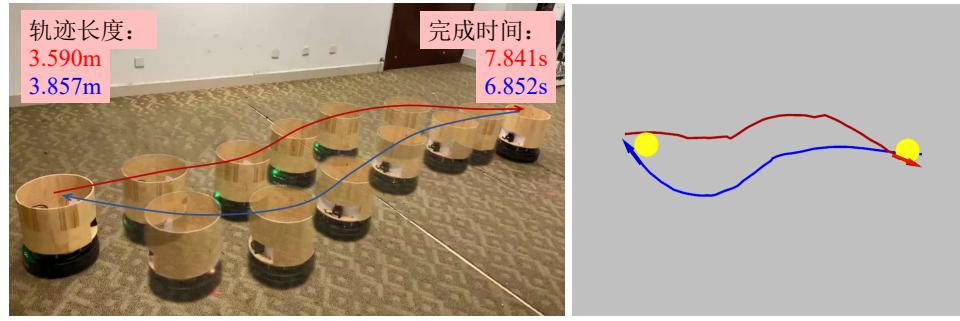
图 3.21 单机器人在静态和动态障碍物环境中的实体实验。左侧的子图显示了机器人的运动轨迹和导航地图，左下角标有相关具体指标。右侧的子图显示了现实世界中的场景，这些环境与相应的左图中机器人位于红色圆圈处的情况相对应。图中的黄色虚线框表示环境中相应的动态障碍物。

3.6.3 多机器人环境测试

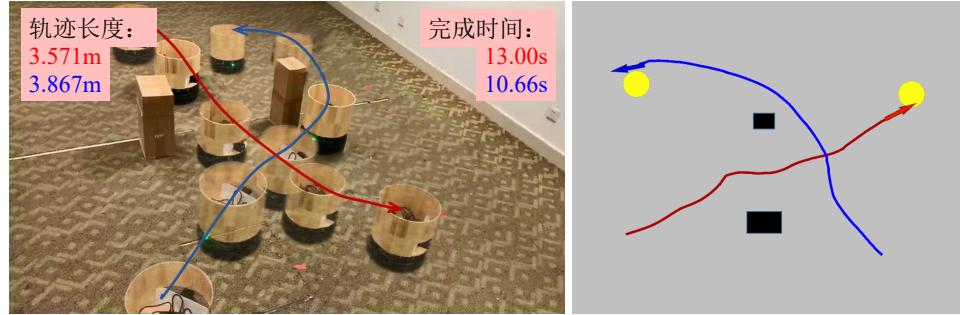
在本小节中，我们将介绍多机器人实体，测试场景中添加了静态障碍物和动态行人。我们首先介绍交换测试场景，环境中两个机器人相互靠近并交换位置。图3.22(a)展示了在这种测试场景中机器人的表现，其中两个机器人都成功到达了它们的目标点且没有发生碰撞。然后，我们通过随机放置一些静态障碍物来扩展两机器人交叉场景，以增加难度。图3.22(b)展示了新环境中的实验，两个机器人仍能成功到达其目标而不会发生碰撞。当机器人观察到另一个机器人正在经过障碍物时，它可以自动等待片刻并通过狭窄的障碍物。随后，我们通过要求行人穿过两个机器人的行进路径来进一步扩展场景。图3.22(c)显示其中两个机器人都自适应地降低了速度并等待行人通行，然后才成功到达各自目标点。需要注意的是，我们没有将行人引入避障策略的训练过程中，并且行人的形状和动态特征与其他机器人和静态障碍物完全不同。但是，在有行人的情况下，学习到的策略仍然可以把行人当成动态障碍物处理，只是机器人的避障距离较近，一定程度上会让行人不舒适。我们将两机器人场景扩展到三个机器人的场景，介绍了三机器人圆形场景，其中将三个机器人放置在一个圆上，并将它们的目标设置在圆上的相对位置。图3.22(d)描述了这种情况下机器人的具体表现，其中所有机器人都成功地到达了目标点而没有碰撞发生。最后，我们将测试实验扩展到涉及四个机器人的场景。图3.23描绘了在这些场景下四个机器人的具体表现，其中所有机器人仍能成功地到达其目标而不会发生碰撞。在图3.23(a)所示的四机器人交叉场景中，两组机器人彼此交叉穿行并到达各自的目标点。图3.23(b)展示了四机器人圆形场景的实验结果，其中将四个机器人放置在一个圆上，并将它们的目标设置在圆上的相对位置。为了进一步证明我们的模型在更复杂的环境中的泛化能力，我们在四机器人圆形场景中分别添加了三个静态障碍物或两个行人，实验结果如图3.23(c)和3.23(d)所示。在如此动态和拥挤的环境中，所有机器人都可以顺利到达各自的目标点而没有发生碰撞。四机器人场景测试的定量结果如表3.5所示，同时在演示视频中也可以找到有关这些多机器人场景的更多细节。这些实验表明，尽管现实世界中两个机器人的动态特性都与仿真器中的机器人不同，但学习到的基于地图的避障策略在这些情况下仍然可以较好地执行。

表 3.5 四机器人场景的测试结果（轨迹长度 [m] / 完成时间 [s]）

环境	机器人 1 (红)	机器人 2 (蓝)	机器人 3 (绿)	机器人 4 (黄)
四机器人交叉场景	2.846 / 5.800	2.835 / 7.606	3.029 / 7.802	3.149 / 9.804
四机器人圆形场景	3.716 / 9.120	3.064 / 12.56	3.054 / 10.97	3.200 / 11.33
有障碍物的四机器人圆形场景	4.890 / 12.51	5.265 / 17.28	5.577 / 15.80	4.689 / 25.31
有行人的四机器人圆形场景	4.543 / 11.29	4.200 / 9.549	3.726 / 11.63	4.346 / 11.54



(a) 两机器人交换场景



(b) 含有障碍物的两机器人交叉场景



(c) 含有行人的两机器人交叉场景



(d) 三机器人圆形场景

图 3.22 两到三个机器人复杂环境实体实验。左侧的子图显示了现实世界中机器人的轨迹，右侧的子图显示了机器人根据自身定位信息记录的轨迹。机器人记录的轨迹不那么平滑主要是由于机器人定位有误差，不同颜色的线表示不同机器人的轨迹。

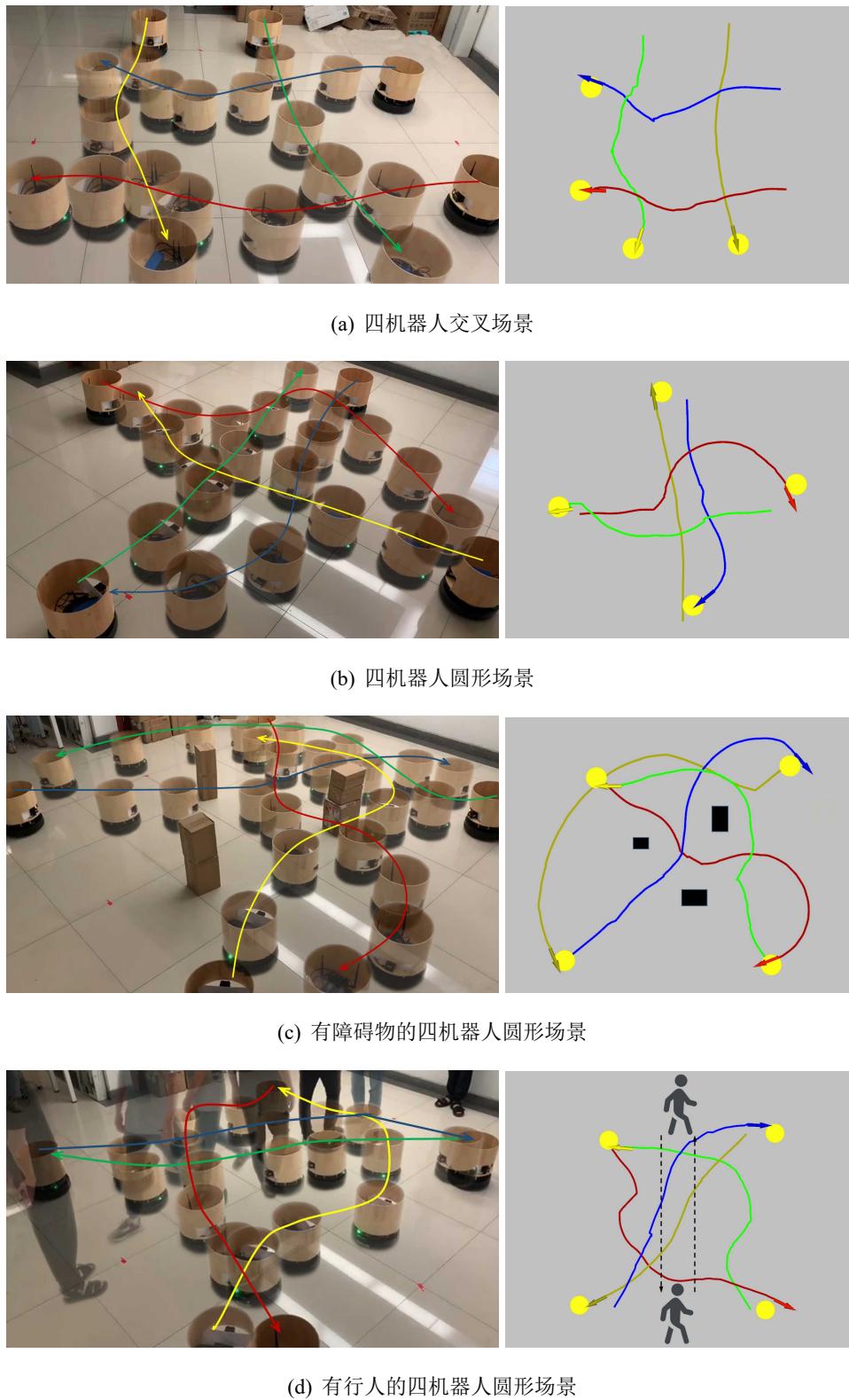


图 3.23 四机器人复杂环境实体实验。左侧的子图显示了现实世界中机器人的轨迹，右侧的子图显示了机器人根据自身定位信息记录的轨迹。机器人记录的轨迹不那么平滑主要是由于机器人定位有误差，不同颜色的线表示不同机器人的轨迹。



图 3.24 配备深度相机和 2D 激光的实验机器人

表 3.6 机器人与传感器标定结果

Kinect 内参					
	k_1	k_2	p_1	p_2	
RGB	0.1074	-0.2328	-0.0021	-0.0029	
IR	-0.1274	0.4823	0.0049	0.0021	
	f_x [px]	f_y [px]	c_x [px]	c_y [px]	
RGB	527.5	525.4	331.6	258.9	
IR	588.2	586.8	321.3	222.7	
	u' [px]	v' [px]			
IR-Depth 偏移	4	4			
机器人与多传感器外参					
	\mathbf{R}_r	\mathbf{R}_p	\mathbf{R}_y	\mathbf{t}_x [mm]	\mathbf{t}_y [mm]
RGB \leftrightarrow IR	0.00211	-0.00241	0.00332	-23.965	0.278
Kinect \leftrightarrow RGB	-1.57	0.0	-1.57	0.0	0.0
Base \leftrightarrow Kinect	0.00178	0.00216	0.00031	21.271	2.729
Base \leftrightarrow Laser	0.00327	-0.00531	-0.00391	121.655	3.882
					350.251
					149.628

3.6.4 多传感器融合避障实验

为了验证基于栅格地图的避障算法对兼容多传感器的能力，我们搭建了如图3.24所示的多传感器机器人，该机器人也是基于 TurtleBot 2 底盘，携带有一个 Hokuyo URG-04LX 激光和一个 Kinect 一代深度相机。首先我们使用第2章提出的机器人与多传感器标定方法标定整个机器人和传感器，标定结果如表3.6所示，包含 Kinect 传感器内参和机器人与多传感器的相关外参。机器人和传感器标定好后，我们运行基于多传感器融合的栅格图生成模块，如图3.25所示，子图（a）中描绘了我们的两个测试环境，其中右图中使用悬空障碍物堵住了机器人的最近出口，如果机器人从此通过会与悬空障碍物发生碰撞。子图（b）显示了 Rviz

可视化软件中的相应深度相机感知的点云变化，虚线框中表明深度相机可以感知到悬空障碍物。子图（c）展示了我们使用激光数据和深度相机点云数据融合生成的栅格图，需要注意的是右图中的最近出口被黑色障碍物堵住，这是深度相机感知的3D点云生成的结果。由于在两种测试环境中深度相机感知的变化，机器人表现出不同避障行为。如图3.25（a）所示，左图中机器人通过最近的障碍物出口达到目标点，而右图中机器人绕开有悬空障碍物的出口，选择绕开所有障碍物到达目标点。这项实验说明我们提出基于栅格地图的深度强化学习避障算法具有融合2D激光信息和3D深度相机点云信息（甚至是其他测距传感器的信息）进行避障的能力，而深度相机等可以提供三维立体信息的传感器对于应对复杂的现实环境具有重要意义，同时我们的标定方法也是使用多传感器在复杂环境导航的前提。

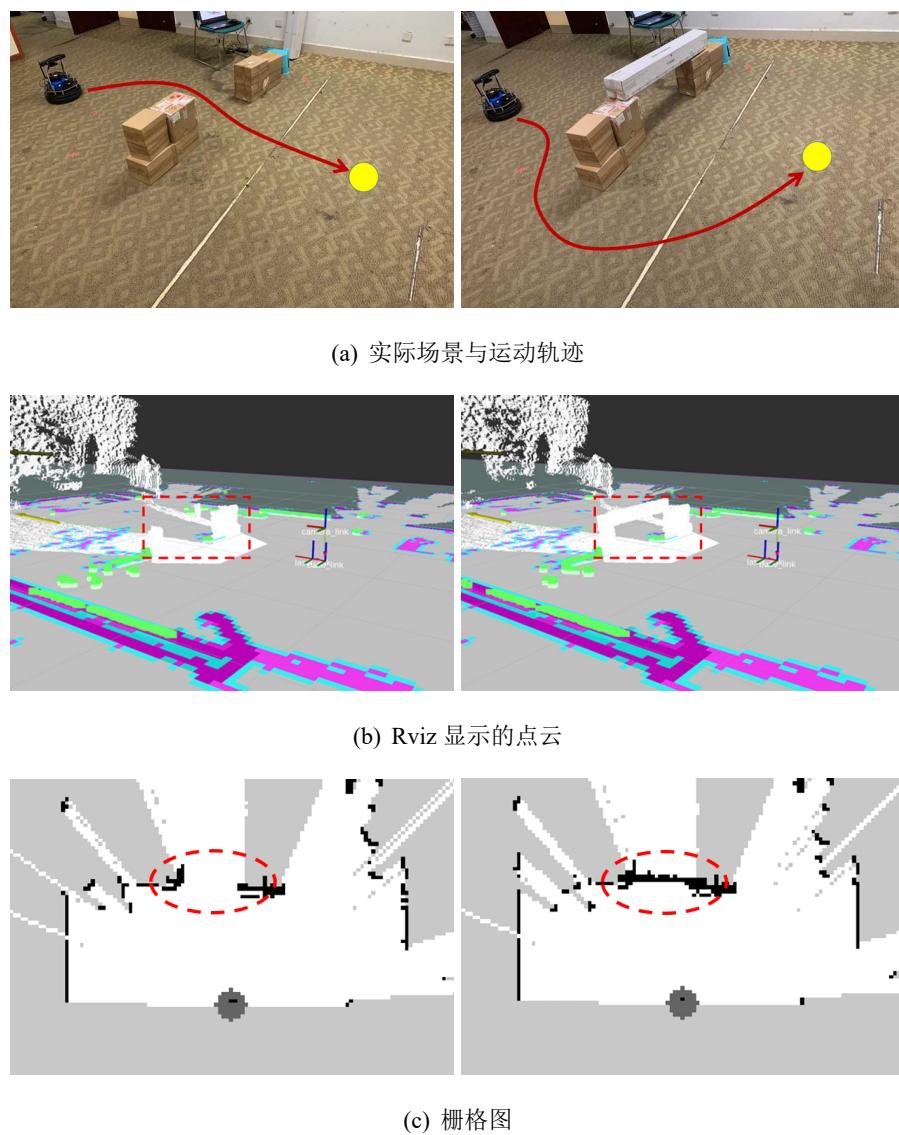


图3.25 多传感器融合3D避障实验，在有无悬空障碍物的场景中对比实验机器人感知和运动差别。

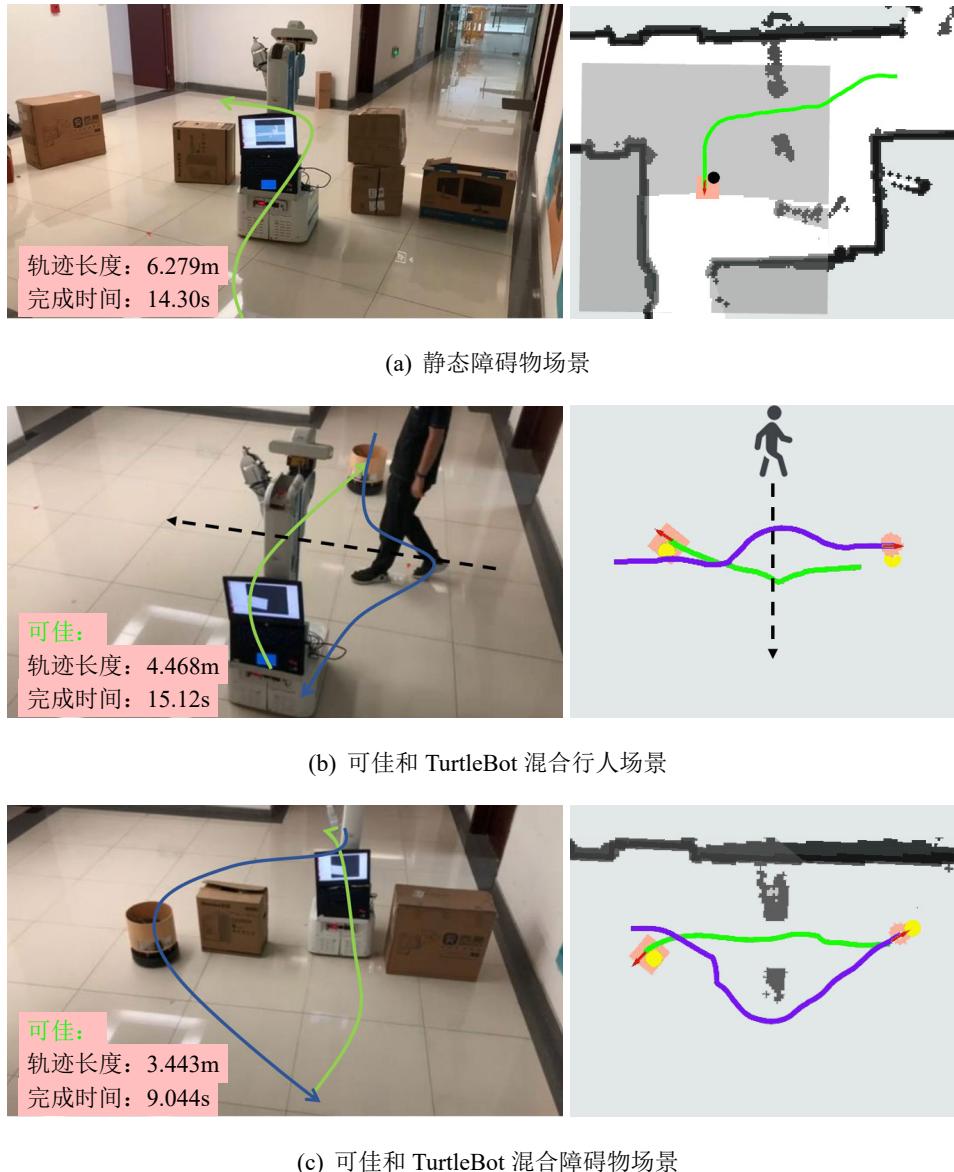


图 3.26 可佳机器人实体实验：(a) 简单静态障碍物场景；(b) 可佳机器人和 TurtleBot 机器人混合行人动态场景；(c) 可佳机器人和 TurtleBot 机器人混合静态障碍物场景。左侧的子图显示了现实世界中机器人的轨迹，右侧的子图显示了机器人根据自身定位信息记录的轨迹。

3.6.5 可佳机器人部署实验

为了验证我们提出的避障算法可以兼容不同机器人平台，我们在可佳机器人^[86]上进行了部署实验。如图3.26(a)所示，可佳机器人的底盘外形与 TurtleBot 机器人有很大不同，可佳机器人的底盘为较大的正方形，边长约 0.5 米，且运动中心位于前方 0.125 米处。我们使用第3.5.3节第3小节中提到的训练模型作为可佳机器人的避障策略。如图3.26所示，我们首先简单测试了可佳机器人的静态避障能力（图3.26(a))。然后我们测试了可佳机器人和 TurtleBot 机器人相互避障的能力，值得说明的是实验中的可佳机器人和 TurtleBot 机器人都是使用我们训练

的深度强化学习避障模型。如图3.26(b)所示，可佳机器人和 TurtleBot 机器人成功相互避开并且避开了动态行人到达各自的目标点。在图3.26(c)中，TurtleBot 机器人主动绕开就近的障碍物出口到达目标点，可佳机器人等待一段时间后选择就近的障碍物出口成功到达目标点，这一定程度上表现了不同机器人间合作避障的能力。尽管可佳机器人和 TurtleBot 机器人具有差别较大的避障外形和运动学模型，但我们提出的基于栅格地图的避障方法都能很好的适配，这说明我们的避障方法对不同硬件的兼容性且具有易于部署的特点。有关本实验中机器人的具体表现，请参见<https://www.bilibili.com/video/BV12v411C7sM/>页面上的演示视频。

3.7 长距离导航系统

本节主要介绍如何把基于深度强化学习的避障模块结合到完整导航系统来实现长距离导航。首先我们阐述该系统框架和组成模块，然后我们使用 TurtleBot 机器人进行了长距离实体导航实验，来验证整个导航系统的可靠性。最后我们演示了大规模上百米楼宇配送实验来说明我们提出的长距离导航系统的实际应用价值。

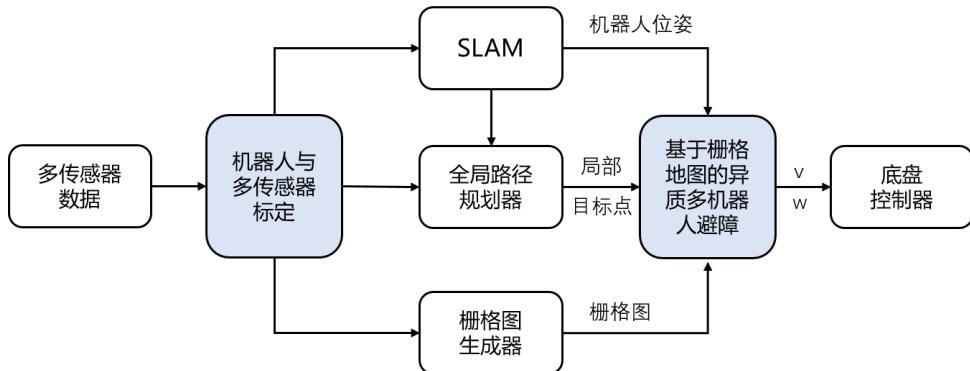


图 3.27 多传感器复杂环境长距离导航系统

3.7.1 系统框架

我们提出的多传感器复杂环境长距离导航系统包含七个模块。如图3.27中所示，机器人与多传感器标定模块用于融合多传感器数据信息。同时定位和建图（SLAM）模块基于传感器数据构建环境地图，并且可以同时估计机器人在地图中的位姿。当接收到目标位置时，全局路径规划器模块会根据来自 SLAM 模块的预先构建的地图，生成从当前位置到目标位置的路径或一系列本地目标点。为了应对动态和复杂的环境，在未知的杂乱环境中需要一个安全且强大的避障模块。除了路径规划模块的本地目标点和 SLAM 定位模块提供的机器人位姿外，我们

的基于栅格地图和深度强化学习的异质多机器人避障模块还需要周围的环境信息，该信息表示为由栅格图生成器模块生成的基于各种传感器融合数据的以自我为中心的栅格图。综上所述，我们基于深度强化学习的局部规划模块需要输入由 SLAM 模块生成的机器人位姿，由全局路径规划程序生成的本地目标的位置以及可融合多个传感器信息的地图生成器模块的栅格图，并输出机器人的控制命令：线速度 v 和角速度 w 。最后，输出速度命令由底盘控制器模块执行，该模块根据机器人的特定运动学将控制速度映射到轮式电机的指令。

3.7.2 长距离导航实验

学习到的多机器人避免碰撞策略用于组成完整的长距离机器人导航系统。我们使用基于深度强化学习的多机器人避障策略进行远距离导航实验，进一步验证了整个导航系统的有效性。具体来说，我们提出了一种用于远距离导航的分层架构^[133]，该架构将 A* 路径规划^[141]与我们提出的基于栅格地图的 DPPO 方法训练得到的基于深度强化学习的避障策略结合在一起，基于深度强化学习的局部运动规划器可以根据全局 A* 路径规划器提供的中间目标，将以自我为中心的局部栅格图直接映射到机器人的控制命令。在导航过程中，将每个局部目标位置选择为在全局计划路径上距机器人 3m 的局部目标点。该导航系统基于深度强化学习的多机器人避障策略，在长走廊（约 27 米）的环境中进行测试。我们使用开源 SLAM 算法 Cartographer^[142]预先构建环境的全局静态地图。全局静态地图用于基于粒子过滤器的定位和路径规划，给基于深度强化学习的避障模块提供了相对于机器人本地目标点的机器人实时姿态。图3.28通过导航地图和系统生成的轨迹描绘了这项长距离测试实验，机器人可以成功穿过长长的走廊，同时避开静态障碍物和动态行人到达远距离的目标点。

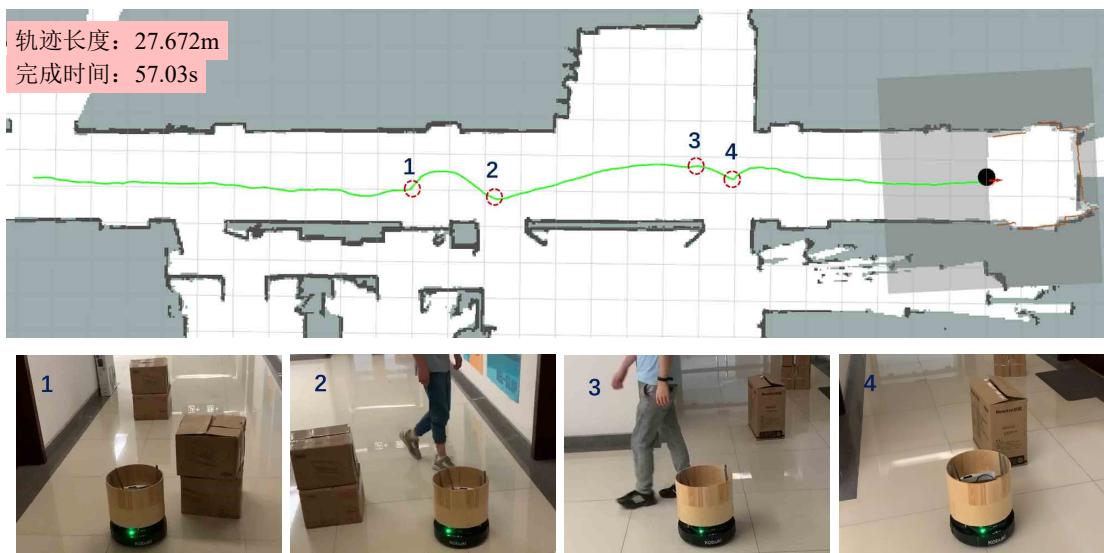


图 3.28 基于深度强化学习的远距离走廊导航实验

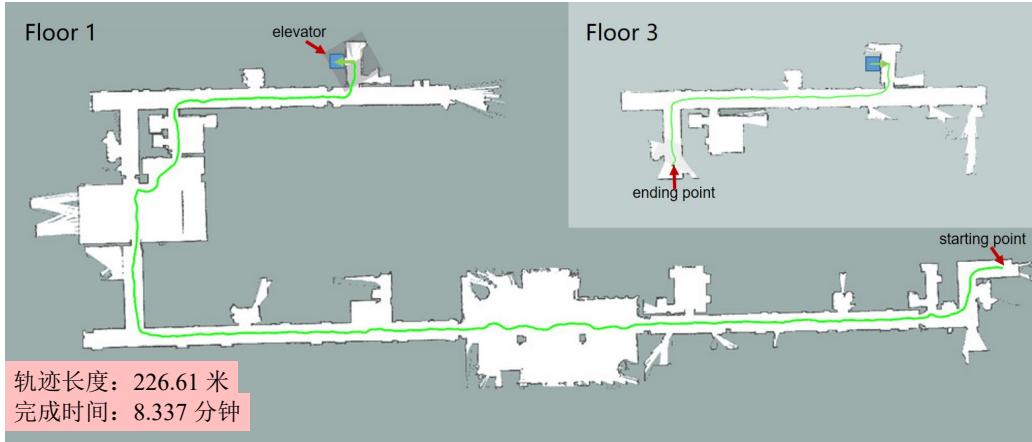


图 3.29 跨楼层大范围楼宇配送测试机器人轨迹图

3.7.3 楼宇配送场景实验

作为电商与自动化、物流与机器人的融合，末端配送机器人致力于解决“最后一公里”的交付问题。末端配送机器人主要依赖于自主移动导航技术，相对于用于智能仓储的移动搬运机器人，末端配送机器人的应用场景更加复杂。末端配送机器人的应用场景一般是开放的商业服务场景，场景更加复杂，需要避开各种复杂的障碍物和行人，甚至需要具有自动上下电梯的能力。末端配送机器人首推在相对封闭的室内场景落地应用，如写字楼、酒店和医院等环境，解决楼内一百米配送的问题。为了说明我们提出的长距离导航系统的实际应用价值，我们进行了一项在中科大西区科技实验楼东西楼楼宇间长距离（220米左右）的导航实验，来模拟楼宇配送机器人应用场景。如图3.29所示，机器人首先在东楼一楼起点出发，行进了约185米路程经过中央大厅到达西楼一楼电梯，搭乘电梯到达西楼三楼，继续行进42米左右路程到达终点。在本实验中，虽然机器人进出电梯时仍需要人员辅助开关电梯，但机器人能够安全避开楼宇中遇到的障碍物和行人并行进三百米远到达终点（如图3.30所示），这一定程度上说明了我们提出的导航系统的应用价值。

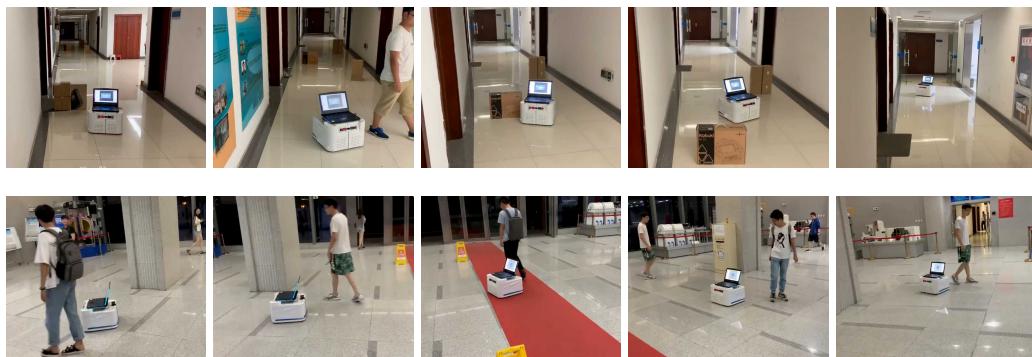


图 3.30 楼宇配送避障场景举例：走廊场景 (第一行) 和中央大厅 (第二行)。

3.8 本章小结

在本章中，我们提出了一种基于栅格地图的分布式近端策略优化方法，用于在分布式和无通信环境中避免多机器人碰撞。我们使用以机器人自我为中心的局部栅格图来表示周围的环境信息，包括其形状以及其他机器人的可观察外观以及障碍物，这些信息可以通过使用多个传感器或传感器融合轻松生成。然后，我们将分布式近端策略优化应用于训练卷积神经网络，该网络直接将三帧以自我为中心的局部网格图和机器人的相对局部目标位置映射为低级机器人控制命令。我们采用两阶段课程学习来提高学习效果。我们在多个仿真场景中评估学习到的避免碰撞策略，并将其与相关工作进行比较。实验结果表明，在成功率、额外时间以及平均线速度方面，我们的方法具有出色的性能。我们还从其他角度评估了我们的避障策略，包括泛化到未见的场景、导航效率以及对机器人形状和动力学的鲁棒性。然后，我们将训练后的模型部署到真实的机器人上，以评估其在具体实际场景中的性能，包括具有静态和动态障碍物的环境、多机器人和动态行人环境。这些实验表明，我们的方法高效且易于部署到真实的机器人上，并且在现实世界中表现良好。同时我们也测试了基于深度相机和激光信息融合的3D导航，实验中机器人成功避开了悬空的物体并到达目标点。为了验证我们提出的导航系统可以兼容不同机器人平台，我们在可佳机器人上进行了部署实验并在多种环境中进行了实体测试。最后，我们提出了多传感器复杂环境长距离导航系统框架，并成功将其部署并应用到实体机器人上。在实际测试环境中，机器人可以成功穿过长距离的走廊，同时避开其中的静态障碍物和动态行人，最后我们演示了大规模上百米楼宇配送实验来说明我们提出的长距离导航系统的实际应用价值。

第4章 动态行人环境下的机器人导航系统

本章我们主要介绍动态行人环境下基于多传感器信息融合的机器人导航系统，首先我们在4.1小节介绍相关研究背景和我们提出的动态行人环境下的机器人导航系统框架。在后续的小节中我们将分别介绍框架主要构成模块，其中包括4.2小节的基于多传感器信息融合的行人感知追踪模块、4.3小节的基于行人感知过滤的动态行人环境定位方法以及基于二维码激光联合主动全局重定位方法和4.4小节介绍的适用于行人环境的多传感器感知多地图通道多机器人深度强化学习导航避障模块。

4.1 动态行人导航研究背景

移动机器人在复杂动态的行人环境中有着许多广泛的重要应用，例如应用在商场、餐厅和医院里为人提供服务的辅助机器人。然而动态行人环境下的机器人导航系统存在两个主要问题：机器人冻结问题和导航定位丢失问题。机器人冻结问题的产生是因为机器人在高度动态的环境中倾向于保守的行为来最小化风险。当场景的动态复杂性增加时，由于环境不确定性的累积，机器人可能无法找到可行的前进路径，最终机器人将在原地停止或在两个方向之间摆动。机器人需要具有在高度动态行人面前的安全导航避障能力。直接建模行人行为的工作^[143-144]表明预测行人的行动是非常困难的。相比之下，使用众所周知的静态避障算法^[119,145-146]或者假定其他智能体行为^[97]，可以在现实环境中产生次优解。随着机器学习和深度强化学习领域的快速发展，近年来的研究工作开始探索神经网络在动态环境下的机器人导航，已经开发了允许通过原始传感器输入数据的端到端动态环境导航^[115]、基于行人位置信息输入的动态环境导航^[114,147]和我们提出的基于栅格地图信息输入的动态环境导航^[118]。虽然使用这些方法可以实现动态避障和导航到目标点，但它们在建模行人之间的交互和以行人可接受的方式导航时存在不足。此外，很难分析这些算法的端到端策略并确保安全运行。其他工作虽然明确地为行人建模来促进个体间的互动，但并不区分行人和静态障碍物^[148-149]。然而现实环境中存在形状各异的动态和静态的障碍物，而且当接近行人和接近障碍物或者其他机器人时，机器人应该表现出不同的避障方式。因此我们扩展了第3章提出的基于栅格地图信息融合的异质多机器人避障算法，我们在传感器栅格地图中添加了由多传感器行人感知追踪信息生成的行人地图通道，设计了新的远离行人的回报函数部分和基于社会力模型与速度障碍物法仿真的多策略行人训练环境。

当机器人在给定的地图上由于较大的定位不确定性或错误而无法准确定位时，就会出现导航定位丢失问题^[20]。例如在动态环境中，机器人可能混淆了已知的静态障碍物和未知的动态障碍物^[31]。因此 Ahmadi 等人^[24] 开发了基于 2D 激光人腿过滤的激光定位方案并证明在动态行人环境中提升了定位准确度。以前大多数解决导航定位丢失问题的方法都是被动方法^[31-33]，他们假设机器人的运动和传感器指向的方向不能被控制，有选择地集中利用传感器流来最小化定位的不确定性或误差。例如，通过使用不同的滤波器^[31] 或更复杂的动态场景建模^[32-33]。然而在高度动态的密集人群场景下，行人可能会遮挡定位所需的所有地标特征，因此机器人必须选择复杂的策略来确定机器人的运动方向和摄像机的指向方向。有些方法通过要求机器人看具有特殊属性的地方来恢复定位，例如具有高显著性的地方^[150] 或道路轨道^[151]。其他一些方法尝试寻找最优的行动，可以最小化关于定位质量的度量，包括熵^[8,34] 或关于机器人当前位置的假设数量^[35]。文献^[152] 中的海岸导航从行人相遇的概率和熵来计算信息的含量，但它假设行人在空间上是均匀分布的，信息的含量必须离线计算，该方法倾向于计算长度较长的轨迹和效率较低的轨迹。定位恢复问题也可以在更通用的部分可观测马尔科夫决策框架下表述和求解，该框架可以通过使用高斯置信空间近似来处理^[153-154]。在本工作中我们采用更加稳定鲁棒的二维码定位恢复区来帮助机器人重新恢复定位。与文献^[155] 类似，我们的框架包含两种机器人导航控制模式：目标点导航模式和恢复区导航模式。在具有挑战性的场景下导航时（例如在无特征的地方、当传感器视野被严重遮挡或当机器人被困在动态人群中的时候），机器人会主动在这两种模式之间在线切换，完成导航任务。

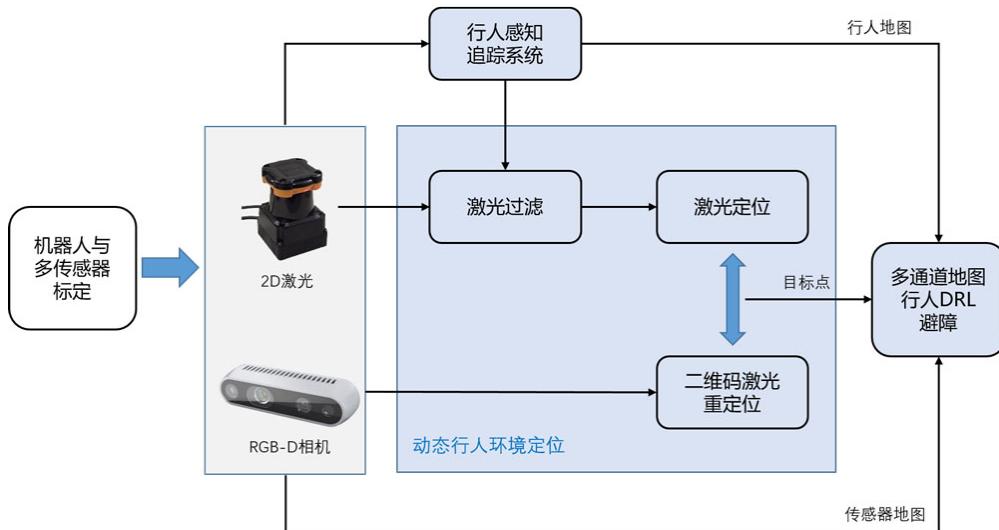


图 4.1 动态行人环境下的机器人导航系统框架

如图4.1所示，我们提出的动态行人环境下的机器人导航系统基于 2D 激光

和 RGB-D 相机信息融合实现，主要包括行人感知追踪系统、激光行人过滤、二维码激光重定位和基于多通道地图和深度强化学习的行人避障模块。基于多传感器融合的行人感知追踪系统可以实时感知行人的位置和速度信息，行人信息用于激光行人过滤和转换成用于行人避障的行人地图。同时为了让机器人在动态行人环境中长期稳定地定位，我们提出了基于二维码激光联合全局重定位的主动定位方法。当机器人定位出现一定偏差时，机器人主动前往二维码定位恢复校准区重定位，然后继续以普通激光定位工作。我们的基于多通道地图和深度强化学习的行人避障模块不仅输入由行人感知追踪系统生成的行人地图信息，还输入第3章使用的相对目标点和多传感器数据生成的栅格地图。需要注意的是，多传感器的使用需要提前应用第2章提出的机器人与多传感器标定方法。

4.2 基于多传感器融合的行人感知追踪系统

在机器人和计算机视觉领域，利用移动传感器平台进行第一人称视角的行人追踪研究已经有十多年的历史，针对不同的传感器模式研究者们提出了各种检测方法和追踪算法。随着在移动机器人平台上的实时检测和追踪的可用计算能力越来越强大，最近出现了融合多种传感器信息多种感知方法的行人追踪趋势。在机器人技术中，激光传感器通常用于覆盖大视场，特别是用于建图和导航。在这种情况下，检测人的方法是基于在激光扫描特征数据中寻找局部最小值的简单特定分类器^[156-157]。在视觉中，摄像头和 RGB-D 传感器的信息通常被用作行人检测管道的输入流。通常 HOG 特征被用来检测全身^[158-159]，而上身检测器更适合检测附近的人^[160]。近年来基于深度学习的行人识别方法被证明比传统基于特征提取的方法具有更强的鲁棒性与准确性。在这两个领域中，漏检和误报都需要用追踪算法来补偿，而追踪算法通常使用某种形式的数据关联方法。我

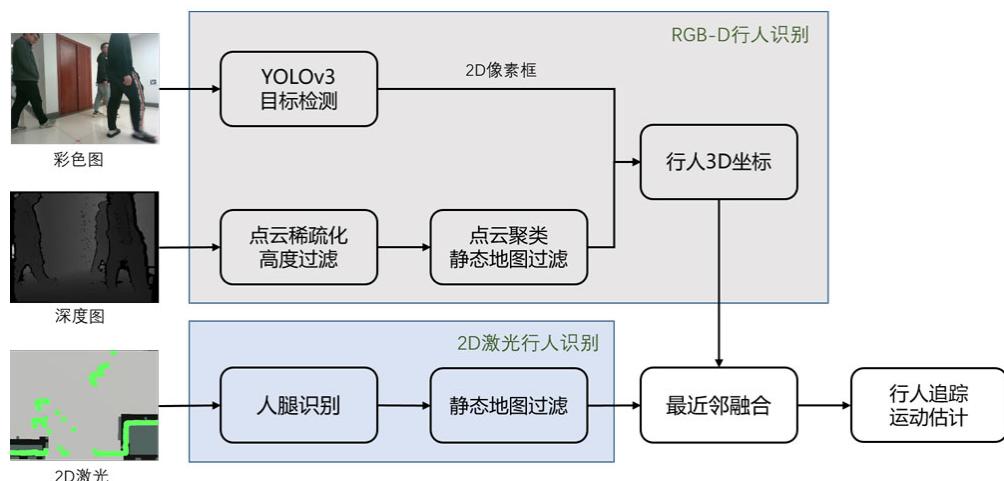


图 4.2 基于多传感器融合的行人感知追踪系统框架

们使用最近邻方法融合激光和 RGB-D 感知行人信息，然后使用扩展卡尔曼滤波(EKF)，纳入一个恒速度的行人运动模型来进行行人追踪与运动估计。

随着计算能力的提升，同时配备激光和 RGB-D 传感器的多传感器移动机器人平台越来越常见。这种组合使得机器人在高度拥挤和动态的操作领域应对挑战成为可能^[161-162]。图4.2描绘了我们提出的基于激光和深度相机的多传感器行人感知追踪系统，下面我们详细介绍我们系统的主要组成部分。

4.2.1 行人检测

行人检测包含基于 2D 激光的人腿识别和基于 RGB-D 相机的行人检测。本文中的基于 2D 激光的人腿识别使用文献^[157] 中提出的方法。激光扫描仪返回一个平面上的距离测量向量。首先根据一个固定的距离阈值对返回的扫描点进行聚类，这样在该阈值内的任何点都作为一个聚类分组在一起。这个阈值被选择得足够小，可以经常将一个人的两条腿分成两个不同的集群，但行人很少产生两个以上的集群。为了降低噪声，小于 3 个扫描点的聚类被丢弃。根据聚类的一组几何特征，聚类可以进一步划分为人类或非人类。几何特征包括点的数量、宽度、长度、标准差、离中位数的平均偏差、是否封闭、线性、圆度、最佳拟合圆半径、边界长度、边界的规律性、平均曲率、平均角度变化等，分类是使用随机森林分类器^[163] 来完成的，该分类器是在一组 1700 个正样本和 4500 个负样本上训练的。将激光扫描仪设置在有大量行人的开放区域，达到上述特征阈值的聚类为行人的结果，作为正训练样本。通过在没有人的环境中移动传感器获得满足上述特征阈值的所有聚类作为负训练样本。

对于基于 RGB-D 相机的行人识别部分，我们分别处理收到的 3D 点云信息和 2D 彩色图像。3D 点云首先进行稀疏化处理，然后根据高度过滤掉地面和超出行人身高的部分的点云信息，然后对剩余点云进行聚类，同时根据静态地图过滤掉位于障碍物里的点云聚类。基于 YOLO v3 目标检测程序^[164-165]，我们可以在彩色图像中识别出行人的 2D 像素范围（如图4.3(a) 所示），然后与处理后的点云聚类对应，最后由行人点云聚类信息计算出行心位置，即行人的三维空间位置坐标。

4.2.2 最近邻融合

对于基于多传感器融合的多行人追踪，我们的框架使用基于最近邻 (NN) 融合算法^[166] 的后端检测融合方案灵活地融合多种模式的检测。该算法具有性能和计算成本之间的合理折衷，在大多数需要追踪的实体集不太密集的情况下都能得到良好的结果^[167]。对于每个候选追踪，使用相关模型对观测结果进行预测。然后，在一个门选择程序之后，使用预测和真实观测之间的相似度度量填充一个

关联矩阵 $\mathbf{B}_{m \times n}$, 其中 m 是多种传感器观测的数量, n 是追踪数量。最后, $\mathbf{B}_{m \times n}$ 中相似性最高的元素被选择, 并且每个测量值 m 用于更新相关联的追踪 n 。由于 NN 算法只能一对一的关联, 即一个追踪只分配一个测量值, 因此对于激光和深度相机感知到的行人都符合相同追踪要求的取行人位置平均值。

我们采用了一种通用的门控方法, 排除了验证区域之外的所有测量值。该区域围绕预测观测 $\hat{\mathbf{x}}_i$ 根据 $d_{mn} \leq \lambda$ 的关系构建, 其中 λ 是人为定义的阈值, d_{mn} 是马氏距离:

$$d_{mn} = \sqrt{(\mathbf{x}_m - \hat{\mathbf{x}}_n)^T \mathbf{B}_{mn}^{-1} (\mathbf{x}_m - \hat{\mathbf{x}}_n)},$$

其中 \mathbf{B}_{mn} 是 $(\mathbf{x}_m - \hat{\mathbf{x}}_n)$ 的协方差矩阵。为了创建关联矩阵, 我们使用以下相似度量:

$$s_{mn} = \frac{1}{\sqrt{(2\pi)^2 |\mathbf{B}_{mn}|}} \exp\left(-\frac{d_{mn}^2}{2}\right),$$

可以看出, s_{mn} 值越大, \mathbf{x}_m 与 $\hat{\mathbf{x}}_n$ 的相似度越高。

4.2.3 行人追踪

估计行人速度和预测行人未来的轨迹是一个活跃的研究领域^[114,168-169]。与 Eppenberger 等人的工作^[170] 类似, 我们采用保守运动模型来估计动态行人的速度和短期未来路径。假设动态行人在水平平面上移动, 我们使用恒定的二维速度模型, 基于卡尔曼滤波 (KF) 估计其速度。卡尔曼滤波的观测输入 \mathbf{z}_i 是聚类行人 T_t^i 的形心 $\mathbf{c}_i = [c_x, c_y, c_z]_i^T$ 的 xy 平面的世界坐标 $\mathbf{z}_i = [c_x, c_y]_i^T$, 我们定义状态向量为 $\mathbf{x}_i = [x, y, \dot{x}, \dot{y}]_i^T$, 则系统转移模型和观测模型定义为:

$$\mathbf{x}_i[k+1] = \mathbf{A}[k] \cdot \mathbf{x}_i[k] + N(0, Q),$$

$$\mathbf{z}_i[k] = \mathbf{H}[k] \cdot \mathbf{x}_i[k] + N(0, R).$$

其中 Q 和 R 分别表示系统噪音和观测噪音, H 表示提取 \mathbf{x} 的前两维数据, A 定义为如下:

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

其中 T_s 表示两次更新的间隔时间。通过使用卡尔曼滤波, 我们可以处理短期行人遮挡问题。具体地说, 当行人 i 在追踪过程中丢失时, 我们继续保持卡尔曼滤波器执行并且计算所有新出现的行人 j 为丢失行人 i 的概率 $p(j = i) = N(\mathbf{c}_j | \mathbf{c}_i, C_i(\mathbf{x}_i))$, 其中 $C_i(\mathbf{x}_i)$ 为状态 \mathbf{x}_i 的估计方差。如果 $p(j = i)$ 超出一定阈值, 我们认为行人 j 为丢失的行人 i 。

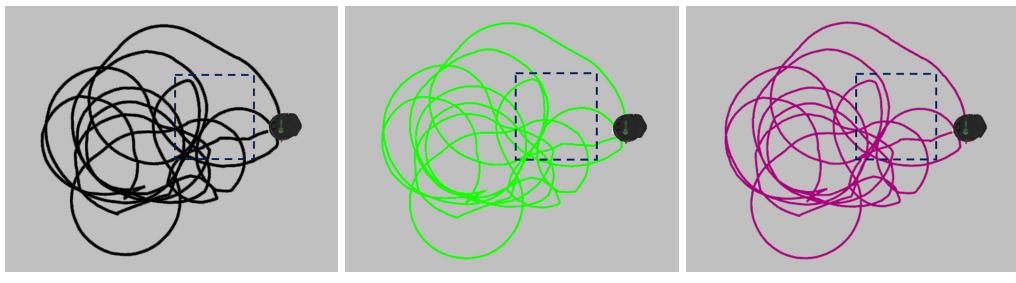
在行人追踪系统中，行人的检测坐标是基于他们的传感器特定的坐标系，然后立即转换成一个全局固定的框架（如基于世界地图的坐标系），不与机器人的自我运动。这就保证了被追踪行人的运动预测独立于机器人的自我运动。如图4.3所示，右图为左图场景的实际感知追踪效果，行人感知追踪可视化系统基于开源项目 SPENCER^[162] 实现。



图 4.3 (a) YOLO v3 的行人检测实例；(b) 多传感器融合的行人追踪效果实例

4.3 动态行人环境下的机器人鲁棒定位方法

当机器人在导航过程中产生的定位不确定性累积过大，导致机器人无法在给定的地图中准确定位时，就会出现导航定位丢失问题^[20]。例如在动态环境中，机器人可能混淆了已知的静态障碍物和未知的动态障碍物^[31]。如图4.4所示，我们首先根据上节介绍的多传感器行人追踪结果过滤掉相应的由于人腿遮挡产生的激光部分数据。我们在有四个动态行人的环境中做了机器人实际测试，不同算法的测试轨迹图如图4.4所示，结果说明过滤掉激光动态人腿部分可以一定程度上提高机器人定位准确度。但此方法在过渡拥挤的环境中可能导致过滤后的激光特征很少，因此不能完全保证机器人在密集人群环境中的定位鲁棒性。



(a) 真实机器人轨迹 (b) 行人过滤前的机器人轨迹 (c) 行人过滤后的机器人轨迹

图 4.4 激光人腿过滤前后机器人定位效果对比

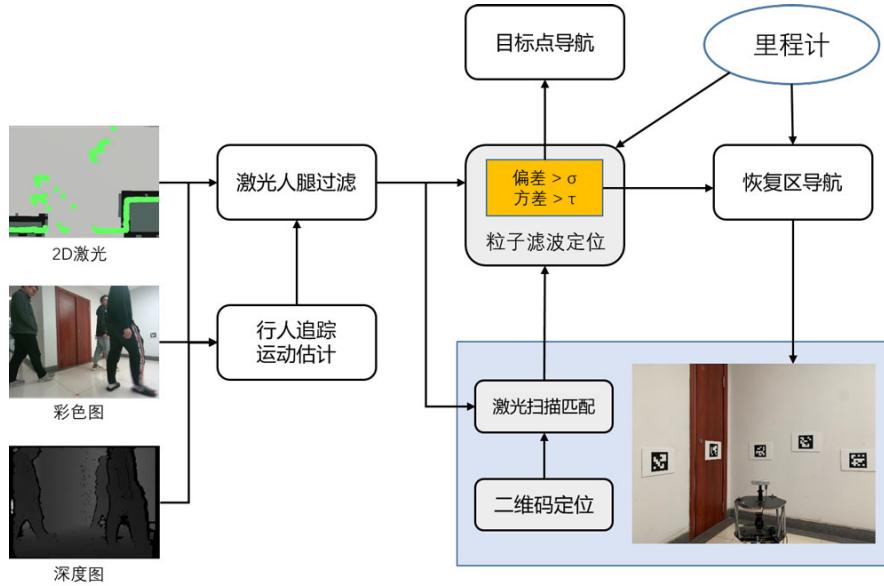


图 4.5 动态行人环境下的机器人鲁棒定位系统框架

现在大多数解决导航定位丢失问题的方法都是被动方法，然而在人群密集的高度动态场景中，定位所需的显著特征可能都被遮挡，因此机器人必须主动决定“移动到哪里”来解决遮挡问题，并“向哪里看”从丢失的定位中恢复^[8,34-35]。在本工作中我们采用更加稳定鲁棒的二维码定位恢复区来帮助机器人重新恢复定位。我们提出的动态行人环境下的机器人鲁棒定位系统框架如图4.5所示，与文献^[155]类似，我们的框架包含两种机器人导航控制模式：目标点导航模式和恢复区导航模式。在具有挑战性的场景下导航时（例如在无特征的地方、当传感器视野被严重遮挡、或当机器人被困在动态人群中的时候），机器人会主动在这两种模式之间在线切换，完成导航任务。恢复区导航触发指标包括里程计测量与 SLAM 定位的偏差和 SLAM 定位的协方差。我们认为，里程计定位系统可能会由于滑动和累积误差而不够准确，但不会瞬间漂移得太远太快。相反地，SLAM 定位系统一般是准确的，但如果在激光数据与全局地图的特征匹配过程中出现错误，可能会有明显的漂移。因此，这两种定位系统的输出之间的差异可以作为模式切换的信号。我们还利用 SLAM 定位输出的协方差来跟踪 SLAM 系统的不确定性。如果这两个值中的任何一个大于给定的阈值，系统将自动进入恢复区导航模式。如图4.6 (a) 所示，激光定位输出机器人位姿的前后两帧变化为 $\overset{M}{R_2} \mathbf{T}^{-1} \cdot \overset{M}{R_1} \mathbf{T}$ ，里程计输出机器人位姿的前后两帧变化为 $\overset{O_2}{R_2} \mathbf{T}^{-1} \cdot \overset{O_1}{R_1} \mathbf{T}$ ，两者的差别即为里程计测量与 SLAM 定位的偏差。当机器人进入恢复区导航模式时，里程计与地图之间的坐标转换固定为定位异常时刻上一帧的输出，即图4.6 (b) 中的 $\overset{M}{O_1} \mathbf{T}$ 。接下来，它将以里程计纯定位的方式移动到事先部署的二维码恢复区大致范围并使用二维码激光联合重定位，即第 t_i 时刻，机器人的位姿估计为 $\overset{M}{O_1} \mathbf{T} \cdot \overset{O_1}{R_i} \mathbf{T}$ 。值得注意的是，机器人开机到 t_1 时刻的里程计累积误差已经被 $\overset{M}{O_1} \mathbf{T}$ 纠正， t_1 时刻后才开始

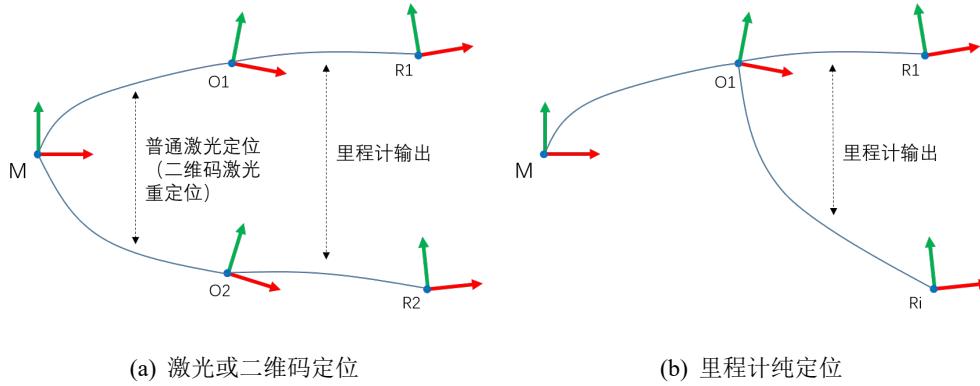


图 4.6 激光或二维码定位与里程计纯定位示意图

累积里程计误差。当恢复过程成功，即机器人定位的协方差小于给定阈值时，机器人将恢复到正常目标点导航模式，里程计纯定位累积误差再一次被重新清除。下面我们详细介绍系统的主要组成部分：二维码识别与二维码激光重定位算法。

4.3.1 二维码识别

当对相机定位的鲁棒性、精度和速度要求较高时，常用基准点标记系统进行相机定位和追踪。在最简单的情况下，点被用作基准标记点，如 LED、反光球等^[171-172]。然而，它们的主要缺点是需要一种方法来解决区分问题，也就是说，需要为每个标记元素分配一个唯一且一致的标识符。为了缓解这个问题，一个常见的解决方案是在每个标记中添加一个标识代码。这方面的例子有平面圆形标记^[173]，2d 条形码^[174]等。在所有提出的方法中，基于方形平面二维码标记的方法得到了广泛的应用。这些标记包括一个外部黑边框和一个内部代码（通常是二进制代码），唯一地识别每个标记。它们的主要优点是可以从单个标记来估计相机的姿态。ARToolKit^[175]是此类方法的开拓者之一。他们使用了带有自定义模式的标记，该模式通过模板匹配来识别。然而这种识别方法容易产生误差，对光照变化的鲁棒性不强。此外，该方法的灵敏度随着标记数的增加而降低。BinARyID^[176]是最早提出生成可定制标记代码方法的系统之一。他们没有使用一组预定义的代码，而是提出了一种为每个特定应用程序生成所需代码数量的方法。AprilTags^[177]在此基础上提出了错误检测和修正的方法，但他们的方法并不适合大量标记的部署。ArUco^[178]可能是目前最流行的二维码标记检测系统。该算法能够适应非均匀照明，鲁棒性强，能够对实现的二进制码进行错误检测和纠错。ArUco 中提出的标记检测和识别的主要步骤如图4.7所示。给定一帧输入图像 I (图4.7a)，由于设计的二维码标记有一个外部的黑色边界包围着，边界可以通过分割找到 (图4.7b)。然后采用等高线跟踪算法从阈值图像中获得等高线集 (图4.7c)，根据轮廓大小和形状过滤掉不相关背景元素 (图4.7d)。下一步是

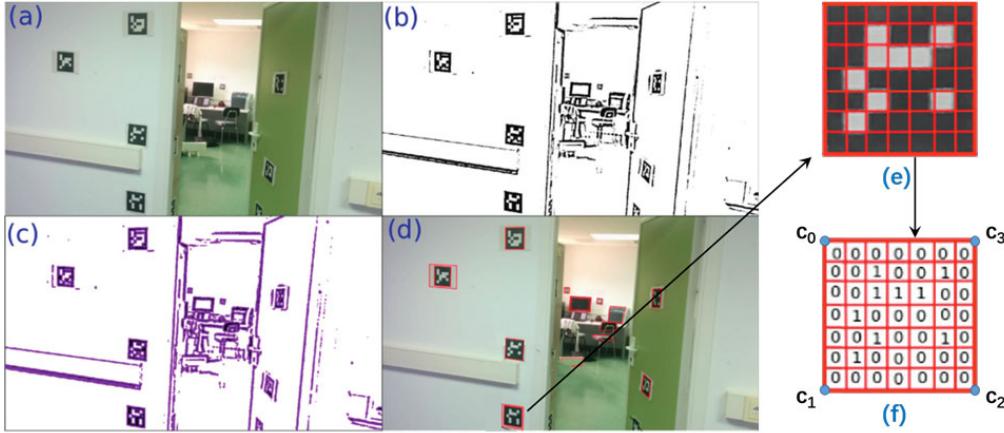


图 4.7 ArUco 二维码的检测与识别流程

分析其余轮廓的内部区域，以确定其中哪些是有效的标记。为此首先通过计算单应性矩阵去除透视投影，并对生成的标准图像（图4.7e）进行阈值处理。二值化的图像（图4.7f）被划分成一个个规则的网格，每个元素根据单元格中的大多数像素被分配一个二值。对于每个候选标记，都需要确定它是属于有效标记集，还是属于背景元素。为每个候选图像获得四种可能的标识符，对应于标准图像的四种可能的旋转。如果任何一个标识符属于有效的标记集，那么它将被接受。最后一步是用超像素精度估计角点的位置。为此该方法采用了二维码标记轮廓像素的线性回归。换句话说，它利用所有轮廓像素估计标记边的线条，并计算交点 $\mathbf{c} = (c_0, c_1, c_2, c_3)$ 。

最后，我们需要估计二维码识别的不确定性来用于精确的状态估计。首先我们根据估计的相机位姿 ${}^M \mathbf{T}_C$ ，识别的特征交点 \mathbf{c} 和二维码的边长 l 计算出相机的重投影误差 e 如下：

$$e = \text{repr}({}^M \mathbf{T}_C, \mathbf{c}, l),$$

然后二维码识别的误差 ϵ 可以用下式简单计算，

$$\epsilon = \frac{e \| {}^M \mathbf{T}_i \|}{l \| c_0 c_2 \|},$$

其中 $\| {}^M \mathbf{T}_i \|$ 表示相机到二维码的距离，相机离二维码越远，识别误差越大。重投影误差 e 越大，识别误差也越大。识别到的二维码图片大小 $\| c_0 c_2 \|$ 和二维码的边长 l 用于归一化。

4.3.2 二维码激光重定位

我们提出的二维码激光联合重定位方法如算法4.1所示，算法输入包括每帧二维码识别结果 $\mathcal{O} = (k, {}^M \mathbf{T}_C, \epsilon_o)$ ，其中 k 表示识别到的二维码编号， ${}^M \mathbf{T}_C$ 表示相机 C 相对于二维码坐标系 M 的位姿， ϵ_o 表示二维码识别的估计误差，需要注意的

算法 4.1 二维码激光建图与重定位算法

Input: 二维码识别结果 $\mathcal{O} = (k, {}_C^M \mathbf{T}, \varepsilon_o)$, 激光数据 ψ

Output: 机器人位姿 $\mathbf{T}_R = ({}^W_R \mathbf{T}, \varepsilon_r)$, 二维码地图 $\mathbf{M}_m = (k, {}_M^W \mathbf{T}, \varepsilon_m)$

```

1 初始化建立激光栅格地图  $\mathbf{M}_s$ , 标定相机外参  ${}_C^R \mathbf{T}$ 
2 if  $\mathbf{M}_m = \emptyset$  then
3    ${}_R^W \mathbf{T}, \varepsilon_s = \text{scan\_match}({}^W_R \mathbf{T}_s, \psi, \mathbf{M}_s)$ 
4    $(k', {}_C^{M'} \mathbf{T}, \varepsilon') = \text{find\_closest\_obs}(\mathcal{O})$ 
5    $\mathbf{M}_m \leftarrow (k', {}_R^W \mathbf{T} \cdot {}_C^R \mathbf{T} \cdot {}_C^{M'} \mathbf{T}^{-1}, h/\varepsilon_s + l\varepsilon')$ 
6    $\mathbf{T}_R \leftarrow ({}^W_R \mathbf{T}, h/\varepsilon_s)$ 
7 end
8 for  $(k_i, {}_C^{M_i} \mathbf{T}, \varepsilon_i) \in \mathcal{O}$  do
9   if  $k_i \in \mathbf{M}_m$  then
10     ${}_M^W \mathbf{T}, \varepsilon_m = \mathbf{M}_m(k_i)$ 
11     ${}_R^W \hat{\mathbf{T}} = {}_M^W \mathbf{T} \cdot {}_C^{M_i} \mathbf{T} \cdot {}_R^C \mathbf{T}$ 
12     $\varepsilon_s = \text{score}({}_R^W \hat{\mathbf{T}}, \psi, \mathbf{M}_s)$ 
13     $\mathbf{T}_R = \text{update}({}^W_R \mathbf{T}, \varepsilon_r, {}_R^W \hat{\mathbf{T}}, h/\varepsilon_s + l\varepsilon_i + \varepsilon_m)$ 
14  end
15 end
16  ${}_R^W \mathbf{T}, \varepsilon_s = \text{scan\_match}({}^W_R \mathbf{T}, \psi, \mathbf{M}_s)$ 
17  $\mathbf{T}_R \leftarrow ({}^W_R \mathbf{T}, h/\varepsilon_s)$ 
18 for  $(k_i, {}_C^{M_i} \mathbf{T}, \varepsilon_i) \in \mathcal{O}$  do
19   if  $k_i \in \mathbf{M}_m$  then
20     ${}_M^W \mathbf{T}, \varepsilon_m = \mathbf{M}_m(k_i)$ 
21     ${}_M^W \hat{\mathbf{T}} = {}_R^W \mathbf{T} \cdot {}_R^{M_i} \mathbf{T}^{-1}$ 
22     ${}_M^W \mathbf{T}, \varepsilon_m = \text{update}({}_M^W \mathbf{T}, \varepsilon_m, {}_M^W \hat{\mathbf{T}}, l\varepsilon_i + \varepsilon_m)$ 
23  end
24 else
25  for  $(k_j, {}_C^M \mathbf{T}_j, \varepsilon_j) \in \mathcal{O}$  do
26    if  $j \neq i$  and  $k_j \in \mathbf{M}_m$  then
27      ${}_M^W \mathbf{T}, \varepsilon_j = \mathbf{M}_m(k_j)$ 
28      ${}_M^W \mathbf{T} = {}_M^W \mathbf{T} \cdot {}_C^M \mathbf{T} \cdot {}_C^M \mathbf{T}^{-1}$ 
29      $\mathbf{M}_m \leftarrow (k_i, {}_M^W \mathbf{T}, l\varepsilon_i + \varepsilon_j)$ 
30   end
31 end
32 end
33 end

```

是每帧二维码识别结果 \mathcal{O} 可能包含多个二维码。算法的输入还包括当前激光扫描数据帧 ψ , 用于校准二维码估计的机器人位姿, 并把二维码和机器人位姿与激光地图对应, 以重定位机器人全局位姿。其中机器人位姿表示为 $\mathbf{T}_R = ({}^W_R \mathbf{T}, \epsilon_r)$, ${}^W_R \mathbf{T}$ 表示机器人 R 在全局地图坐标系 W 下的位姿, ϵ_r 表示机器人位姿的估计误差; 二维码地图表示为 $\mathbf{M}_m = (k, {}^W_M \mathbf{T}, \epsilon_m)$, 包含所有识别到的二维码 k , 及其相对于激光地图坐标系的位姿估计 $({}^W_M \mathbf{T}, \epsilon_m)$ 。

首先我们需要初始化建立激光栅格地图 \mathbf{M}_s 和标定相机相对于机器人的外参 ${}^C_R \mathbf{T}$ (l. 1), 当最开始二维码地图为空时, 我们需要根据激光定位结果 ${}^W_R \mathbf{T}_s$ 作为机器人位姿初值, 根据激光数据 ψ 和栅格地图 \mathbf{M}_s 使用算法4.2介绍的激光扫描匹配算法计算出机器人精确位姿 $({}^W_R \mathbf{T}, \epsilon_s)$, 激光扫描匹配算法将在后续介绍 (l. 3)。然后我们在二维码识别结果中选择最近的二维码作为二维码地图的起始位姿 (l. 4), 此二维码 k' 相对于激光全局地图 \mathbf{M}_s 的位姿计算为 ${}^W_R \mathbf{T} \cdot {}^R_C \mathbf{T} \cdot {}^{M'}_C \mathbf{T}^{-1}$, 同时其位姿估计误差更新为 $h/\epsilon_s + l\epsilon'$, 与激光扫描匹配分数 ϵ_s 成反比, 与二维码识别误差 ϵ' 成正比, 其中 h, l 为人为设置的参数 (l. 5)。机器人的全局位姿初始化为激光扫描匹配算法的结果 ${}^W_R \mathbf{T}, h/\epsilon_s$ (l. 6)。接下来算法主要分为两部分: 更新机器人位姿和更新二维码地图。算法4.1的 8 – 17 行表示更新机器人全局位姿, 18 – 33 行表示更新二维码在全局栅格地图下的位姿。而无论是更新机器人位姿还是二维码位姿, 归纳来说都是根据观察的位姿估计 $\mathbf{T}_2 = (\mathbf{q}_2, \mathbf{t}_2), \epsilon_2$; 去更新原始位姿 $\mathbf{T}_1 = (\mathbf{q}_1, \mathbf{t}_1), \epsilon_1$ 的过程 (定义为 $\text{update}(\mathbf{T}_1, \epsilon_1, \mathbf{T}_2, \epsilon_2)$ 函数)。在本算法中我们借鉴卡尔曼滤波中的更新思想, 首先我们计算卡尔曼更新系数 k 如下:

$$k = \frac{\epsilon_1}{\epsilon_1 + \epsilon_2},$$

然后更新后位姿的位移部分 \mathbf{t} 计算如下:

$$\mathbf{t} = \mathbf{t}_1 + k(\mathbf{t}_2 - \mathbf{t}_1),$$

更新后位姿的四元数部分 \mathbf{q} 计算如下:

$$\mathbf{q} = \text{slerp}(\mathbf{q}_1, \mathbf{q}_2, k),$$

其中 slerp 函数表示球面线性插值, 是四元数的一种线性插值运算, 主要用于在两个表示旋转的四元数之间平滑差值。新的位姿估计误差 ϵ 计算为:

$$\epsilon = 1/(2\pi(\mathcal{N}^2(0, \epsilon_1, \|\mathbf{t} - \mathbf{t}_1\|) + \mathcal{N}^2(\|\mathbf{t}_2 - \mathbf{t}_1\|, \epsilon_2, \|\mathbf{t} - \mathbf{t}_1\|))),$$

其中 $\mathcal{N}(a, b, c)$ 表示以 a 为均值, b 为方差的高斯分布在 c 点处的值。

在机器人位姿更新过程中, 我们首先根据已在二维码地图中的二维码识别信息估计出机器人的全局位姿 ${}^W_R \hat{\mathbf{T}}$ (ll. 9-11), 然后根据激光扫描匹配算法算出

此位姿下的得分 ε_s , 机器人的全局位姿估计 ${}^W_R \hat{\mathbf{T}}$ 的误差计算为 $h/\varepsilon_s + l\varepsilon_i + \varepsilon_m$, 最后根据上述 update 函数计算出新的机器人位姿 \mathbf{T}_R (ll. 12-13)。为了充分利用激光的高精度特点, 我们根据二维码估计机器人位姿 \mathbf{T}_R 为初值使用自适应的激光扫描匹配算法再次计算精确的机器人全局位姿 (ll. 16-17)。

最后我们根据上述估计的机器人全局位姿和二维码观察更新二维码在激光全局地图的位姿, 当观察到的二维码信息在原二维码地图中存在时, 我们可以根据当前二维码的观察更新原先保存的二维码全局位姿 (ll. 19-23)。如果观察到的二维码信息在原二维码地图中不存在时, 我们根据其他原先已经存在的二维码观察计算该二维码的全局位姿估计, 然后直接保存到二维码地图中 (ll. 24-32)。

算法 4.2 激光扫描匹配算法

Input: 机器人初始位姿 $\mathbf{T}_R = ({}^W_R \mathbf{T}, \varepsilon_r)$, 激光数据 ψ , 栅格地图 \mathbf{M}_s

Output: 校准后的机器人位姿 ${}^W_R \mathbf{T}$, 匹配分数 ε_s

```

1 标定激光外参  ${}^R_L \mathbf{T}$ 
2  $\varepsilon_s \leftarrow 0; \mu \leftarrow \infty$ 
3  $(\mathbf{X}, \mathbf{Y}, \mathbf{A}) = \text{search\_area}(\varepsilon_r)$ 
4 for  $(x, y, \theta) \in (\mathbf{X}, \mathbf{Y}, \mathbf{A})$  do
5    ${}^W_R \mathbf{T}' = T(x, y, \theta) \cdot {}^W_R \mathbf{T}$ 
6   for  $s \in \psi$  do
7      $(p_{hit}, p_{near}) = \text{scan\_point}({}^W_R \mathbf{T}' \cdot {}^R_L \mathbf{T}, s, \delta)$ 
8      $(m, n) = \text{map\_index}(\mathbf{M}_s, p_{hit})$ 
9      $(m', n') = \text{map\_index}(\mathbf{M}_s, p_{near})$ 
10    for  $(k_1, k_2) \in K$  do
11       $p'_{hit} = \text{world\_point}(\mathbf{M}_s, m + k_1, n + k_2)$ 
12       $p'_{near} = \text{world\_point}(\mathbf{M}_s, m' + k_1, n' + k_2)$ 
13      if  $\mathbf{M}_s(p'_{hit}) = obs$  and  $\mathbf{M}_s(p'_{near}) = free$  then
14         $\mu' = \|p_{hit} - p'_{hit}\|$ 
15        if  $\mu' < \mu$  then
16           $\mu = \mu'$ 
17        end
18      end
19    end
20     $\varepsilon_s = \varepsilon_s + \exp(-\frac{1}{\sigma\mu^2})$ 
21  end
22 end

```

最后我们简单介绍一下算法4.1中涉及到的激光扫描匹配算法4.2，该算法是根据激光的扫描数据 ψ 和栅格地图 \mathbf{M}_s ，在输入的初始机器人位姿附近需要最优的机器人位姿，最优的机器人位姿以激光数据和栅格地图的匹配程度判断（即匹配得分 ϵ_s ）。首先我们在输入的初始机器人位姿附近生成搜索区域（l. 3），对于区域中的每个机器人位姿，我们计算激光扫描点的障碍物点和空白点（ll. 7-9），如果和地图信息在一定范围内匹配，计算出当前得分（ll. 10-19），遍历所有激光扫描光束数据累加得分，输出得分最高的位姿和累计分数（l. 20）。

4.3.3 实体机器人测试

为了验证算法的有效性和精度，我们粘贴五张边长为 $0.4m$ 的二维码在墙壁上，使用机器人在二维码前自由移动，建立的二维码位姿地图如图 4.8 所示，同时我们可以清晰地看出，加入激光扫描匹配算法后可以在远处二维码识别误差很大的情况下仍然准确的定位。图 4.9 描绘了不同方法的机器人定位轨迹，根据图 (a) 和图 (b) 的对比我们可以看出激光和二维码联合重定位更接近使用激光的粒子滤波定位效果。其中图 (a) 或 (b) 中黑色线段是使用激光的粒子滤波定位效果，图 (a) 中蓝色线段为仅使用二维码的全局定位效果，图 (b) 中蓝色线段为激光和二维码联合重定位效果。同时定量计算定位精度的绝对轨迹误差 (ATE) 和相对位姿误差 (RPE) 指标结果如表4.1所示，我们可以准确地看出二维码和激光联合重定位精度更高。

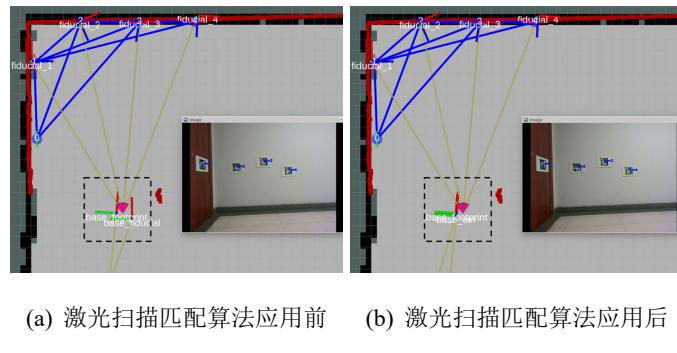


图 4.8 激光扫描匹配算法应用前后定位效果（黑色虚线框中坐标系重合程度）对比

表 4.1 二维码重定位准确度

	二维码重定位 [rmse/mean/std]	二维码和激光联合重定位 [rmse/mean/std]
ATE [m]	0.0522 / 0.0445 / 0.0274	0.0350 / 0.0293 / 0.0192
RPE.translation [m]	0.0749 / 0.0636 / 0.0396	0.0490 / 0.0414 / 0.0261
RPE.rotation [deg]	2.3287 / 1.8774 / 1.3777	2.2505 / 1.7149 / 1.4574

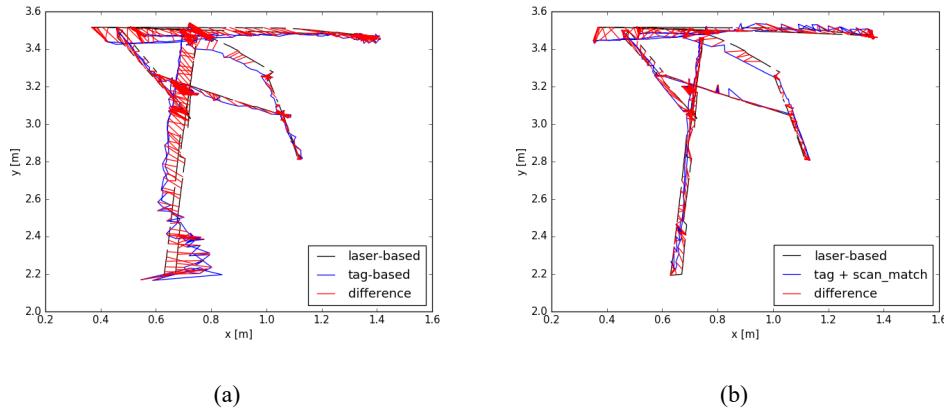


图 4.9 二维码定位轨迹图对比

4.4 动态行人环境下的强化学习避障算法

考虑到移动机器人领域的快速发展，可以预见的是，自主机器人将从主要在结构化工业环境中运行转向部署在非结构化和动态行人场景中。机器人需要具有在高度动态行人面前的安全导航能力。因此我们扩展了第3章提出的基于栅格地图信息融合的异质多机器人避障算法，如图4.10所示，我们在传感器栅格地图中添加了由多传感器行人感知追踪信息生成的行人地图通道，设计了新的远离行人的回报函数部分和基于社会力模型与速度障碍物法仿真的多策略行人训练环境，实验表明我们的方法进一步提升了机器人在不同行人策略的复杂障碍物环境中的避障成功率。下面我们详细介绍我们方法的主要部分。

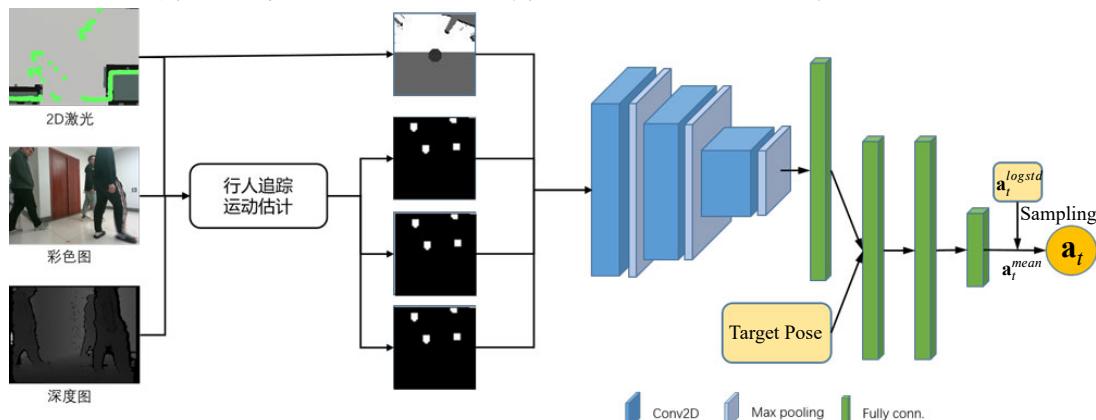


图 4.10 动态行人环境下的强化学习避障算法框架与策略网络

4.4.1 行人仿真

为了满足动态行人训练环境需求，我们需要在第3.5.2节提到的2D仿真器中添加动态行人。然而真实地模拟行走的人的互动是不容易的，可能导致计算昂贵的表示^[179]。在PedSim^[180]中，个体行人按照Helbing等人提出的社会力模型^[181]

(SFM) 移动。该模型结合了从目标、静态障碍物和其他行人产生的不同势场。社会力模型非常简单，可以在行人仿真的高速仿真器中使用。同时我们注意到有不少工作采用 ORCA 作为行人仿真的控制策略，为了增加行人行走策略的多样性，我们的仿真器同时支持添加基于 ORCA 策略控制的仿真行人和基于 PedSim 策略控制的仿真行人。另外受 Guldenring 等人的工作^[18]启发，在我们的 2D 仿真世界中，每个行人都被建模为两个圆圈，代表机器人激光扫描仪感知到的人腿。其根据行人控制策略 PedSim 或 ORCA 的计算速度和每个行人的位置，然后添加腿部摆动的运动。每条腿按照三角速度函数摆动。为了达到命令的全身速度 v ，腿部直线加速至最大速度 $4 \cdot v$ ，然后持续减速至 $0m/s$ ，在此期间另一条腿保持相同的姿势。由于真实的人类有不同的步行和运动行为，我们对步行模式的关键模型参数，如腿的速度，腿的大小和间距应用了域随机化。图4.11为行走模型模式示意图。

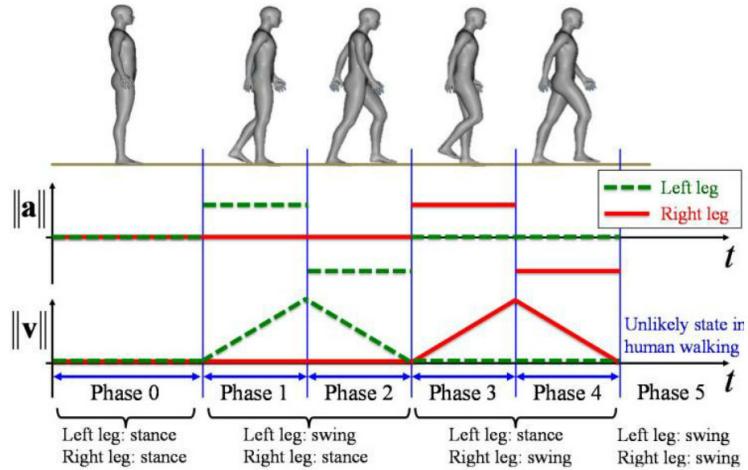


图 4.11 人类行走不同阶段的简单腿部运动模型^[18]

4.4.2 避障策略学习

我们使用3.4.2节介绍的分布式近端策略优化 (DPPO) 算法作为我们的强化学方法。深度学习部分是在 Tensorflow 框架中实现的，使用 Adam 优化器来优化策略网络。

1. 观察空间

观察空间由行人感知通道、传感器通道和机器人相对目标点三部分组成。行人感知通道生成一张三通道行人地图 \mathbf{M}_{ped} ，表示机器人周围行人的位置和速度。与上一章中的多机器人避障策略相比，我们的状态空间多了由感知到的行人位置和速度生成的三张行人地图通道信息。其中一个通道表示周围行人的位置，将行人所在的单位赋值为 1，另外两个通道表示行人的 x 轴和 y 轴速度信息。传感

器通道由一个以自我为中心的栅格地图 \mathbf{M}_{sens} 表示，该栅格地图对周围的障碍物信息进行编码。该地图可以很容易地表示各种传感器感知数据，如激光或深度相机。同时，上一章的工作也验证了这种表示法更有利于的网络训练和部署。目标点 $\mathbf{g}_i^t = (x_i^t, y_i^t, \alpha_i^t)$ 是一个元组，包括规划路径上的下一个相对局部目标点位置 (x_i^t, y_i^t) 和其相对方向 α_i^t 。

2. 动作空间

差分运动机器人的动作 a_i^t 由一个线速度 v_i^t 和一个角速度 ω_i^t 组成。不同于 DQN 只适用于离散动作空间，PPO 可用于离散的或连续的动作空间。因此，本文采用两种动作空间进行对比实验。当使用连续动作空间时，我们设 $v_i^t \in [0, 0.6]$, $\omega_i^t \in [-0.9, 0.9]$ 。在离散动作空间的情况下，我们设定线速度 $v_i^t \in [0.0, 0.2, 0.4, 0.6]$ 和角速度 $\omega_i^t \in [0.9, -0.6, -0.3, 0.0, 0.3, 0.6, 0.9]$ 。这些动作可以由我们实验中使用的差动机器人直接执行。我们设置 $v_i^t \geq 0$ ，即不允许机器人向后移动，因为我的机器人没有后视的传感器。

3. 回报函数

回报函数在深度强化学习任务中起着重要指导作用，所以回报函数的设计与实际任务有关。我们的目标是在保证不碰撞的情况下最小化机器人到达目标点的平均到达时间。因此，我们制定了以下奖赏函数: $r^t = r_{goal}^t + r_{safe}^t + r_{step}^t + r_{shaping}^t$ 。与上一章的多机器人避障工作相比，我们添加了 r_{safe}^t 部分。 r_{safe}^t 表示机器人遇到碰撞或危险时的惩罚。由于机器人出现在行人周围可能会让人不舒服，我们希望机器人离周围的行人尽可能在一米之外。当机器人靠近行人时，我们增加一个惩罚。我们定义 r_{safe}^t 为:

$$r_{safe}^t = \begin{cases} r_{col} & \text{if collision,} \\ -\xi (1 - d_{min}^t) & \text{if } d_{min}^t < 1, \\ 0 & \text{otherwise,} \end{cases}$$

其中 $r_{col} < 0$ 为碰撞惩罚， d_{min} 为机器人与最近行人的距离， ξ 为超参数。其他回报函数部分与上一章多机器人避障工作一致，即 r_{goal}^t 表示机器人到达目标点的奖励，

$$r_{goal}^t = \begin{cases} r_{arr} & \text{if goal reached,} \\ 0 & \text{otherwise,} \end{cases}$$

奖励塑造部分 $r_{shaping}^t$ 鼓励机器人朝目标点移动，

$$r_{shaping}^t = \eta (\|\mathbf{p}^{t-1} - \mathbf{p}_g\| - \|\mathbf{p}^t - \mathbf{p}_g\|),$$

在我们的实验中，我们设置 $r_{arr} = 500$, $r_{col} = -500$, $\xi = 50$, $\eta = 200$, $r_{step}^t = -5$ 。

4. 训练过程

如图4.12所示，我们设计了两种不同的训练场景，即随机场景和圆形场景。随机场景的环境包含两个机器人，四个行人，四个静态障碍物。行人、机器人、目标点和障碍物的位置是随机生成的。环境中的机器人将由策略网络驱动，行人由 ORCA 或者 SFM 驱动。需要注意的是，环境中两个机器人共享相同的避障策略，它们也需要避免彼此碰撞。正如上一章中讨论的，这有助于网络学习多机器人避障的能力，这也可以帮助机器人避开行人。在随机场景环境中进行训练，使策略网络能够避开静态和动态障碍，并与行人交互。圆形场景的环境包含两个机器人和四个行人，将机器人和行人随机放置在一个随机半径的圆上。同时，为了避免策略学习过拟合，我们在随机生成的起始和结束位置上添加了高斯噪声，这个场景增强了机器人和行人之间的交互。在实验中，我们使用四个环境同时训练和收集经验。其中前两种环境的策略采用社会力模型，后两种环境的行人策略采用 ORCA。具体地说，我们构建了两个随机场景环境，分别在 ORCA 和 SFM 策略下控制行人。我们还构建了 ORCA 和 SFM 分别驱动行人的圆形场景环境。然后我们根据这四种环境产生的经验训练网络。

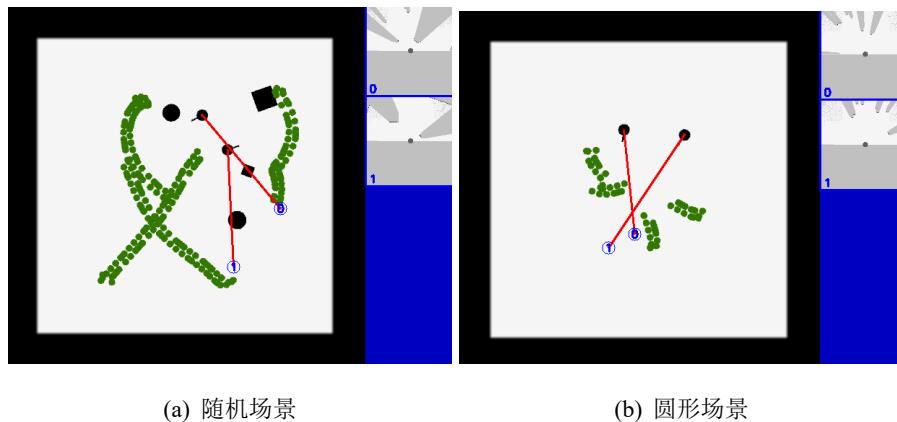


图 4.12 行人仿真训练场景

4.4.3 仿真实验

在本节中，我们首先描述实现的细节，包括用于训练的超参数、硬件和软件。然后，我们定量地评估了我们的人群避障策略在各种仿真场景下的性能，并与其他方法进行了比较。具体实验效果请参考在线视频：<https://www.bilibili.com/video/BV1Vb4y1D7R6>。

1. 训练结果

训练的超参数（表3.2）、硬件和软件与第3章多机器人避障策略基本一致。策略网络和价值网络都在 TensorFlow 中实现，并使用 Adam 优化器更新参数。训练硬件是一台配备 i9-9900k CPU 和 NVIDIA Titan RTX GPU 的计算机。我们实现

并对比了四种机器人运动控制策略：连续动作行人地图和传感器地图输入 PPO (PPO-PSC)、离散动作行人地图和传感器地图输入 PPO (PPO-PSD)、离散动作传感器地图输入 PPO (PPO-SD) 和 ORCA。图4.13展示了三种不同 PPO 策略方法训练过程的期望回报和成功率的变化对比。很明显可以看出行人地图的引入可以提高基于 PPO 的人群避障性能。

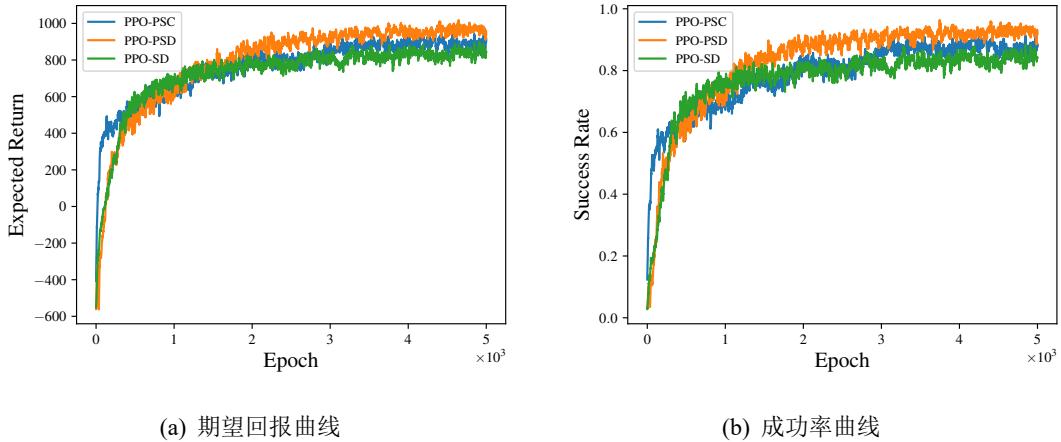


图 4.13 不同 PPO 算法训练过程的期望回报曲线和成功率曲线对比

2. 对比实验

现在我们比较四种不同的机器人避障方法的性能，即 ORCA、PPO-SD、PPO-PSC 和 PPO-PSD。我们使用成功率 $\bar{\pi}$ 、额外时间 \bar{t} 和平均角速度变化值 $\nabla\omega$ 来定量评价避障策略的好坏。我们使用了两个环境进行测试，一个是带有静态障碍物的随机环境，另一个是没有障碍物的圆形环境。随机环境随机放置一个机器人、四个行人和四个形状不同的障碍物。圆形环境与训练场景相似，不同的是这些机器人和行人的起始位置和目标位置都是均匀沿着圆形分布的。

表 4.2 不同方法的测试指标

环境	方法	$\bar{\pi}$ (ORCA / SFM)	\bar{t} (ORCA / SFM)	$\nabla\omega$ (ORCA / SFM)
随机场景	ORCA	0.464 / 0.318	2.97 / 1.53	—
	PPO-SD	0.856 / 0.840	3.60 / 3.45	0.63 / 0.63
	PPO-PSC	0.862 / 0.873	5.64 / 4.90	0.74 / 0.74
	PPO-PSD	0.936 / 0.968	4.83 / 4.99	0.57 / 0.56
圆形场景	ORCA	0.862 / 0.216	7.54 / 1.43	—
	PPO-SD	0.916 / 0.902	4.25 / 4.49	0.82 / 0.84
	PPO-PSC	0.948 / 0.924	5.18 / 6.22	1.02 / 0.97
	PPO-PSD	0.994 / 0.996	4.87 / 5.90	0.57 / 0.63

表4.2显示了这两个场景中这些方法的性能指标（每种场景包括两种行人策略 ORCA 和 SFM 测试），其中指标是根据每种场景下 500 个不同环境的平均结果计算的。可以看出，我们的离散动作行人地图和传感器地图输入 PPO 策略 (PPO-PSD) 在两种环境下的成功率都高于其他方法，ORCA 策略很容易被困在有静态障碍物的环境中。强化学习策略通常能很好地完成导航任务，而且我们同时使用行人地图和传感器地图比只使用传感器输入的策略更安全。由于我们的奖励功能不会对轨迹的平滑性施加限制，几种强化学习的角速度变化值 $\nabla\omega$ 往往大于 ORCA。注意表中 ORCA 的平均角速度变化值为空，因为经典 ORCA 算法只输出 x 和 y 方向的线速度而不输出角速度。有趣的是，我们本以为使用连续动作空间的机器人行走会更平稳，但在实验中，它的角速度变化比使用离散动作空间策略的角速度变化要高。在三种强化学习策略中，离散动作行人地图和传感器地图输入 PPO (PPO-PSD) 的角速度变化是最小的。

总而言之，基于 PPO 的人群避障方法允许机器人成功地与由 ORCA 或 SFM 驱动的行人互动。此外，行人地图的引入可以提高避障性能。在成功率方面，PPO-PSD 优于其他方法。图4.14所示为不同策略在包含行人和静态障碍物环境中的运动轨迹示例。其中，ORCA 在障碍物附近被障碍物和行人卡住，仅使用传感器信息作为输入的 PPO 与行人发生碰撞，使用行人地图和传感器地图作为输入的两种 PPO 方法成功完成了导航任务。实验结果表明，我们的 PPO-PSD 方法是有效的，优于其他方法。

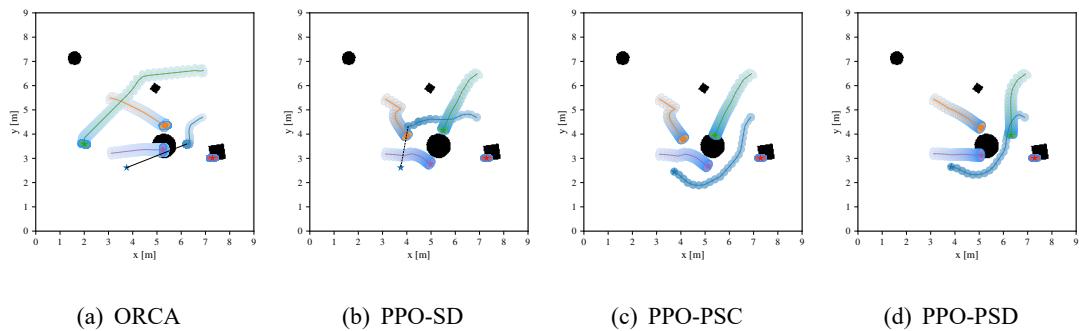


图 4.14 不同方法的机器人测试轨迹图

3. 不同行人策略实验

现有的人群导航方法大多假设行人遵循预先设定的避碰策略，并在相同的避碰策略下测试其性能，但很少在使用其他策略的行人环境中测试他们的表现。我们保持训练环境中其他配置不变，只使用社会力模型 SFM 或 ORCA 作为环境中行人的策略来训练出两个模型 (PPO-SFM 和 PPO-ORCA)。然后我们使用五种测试场景来测试三种模型 (包含我们的 SFM 和 ORCA 行人混合训练模型 PPO-multi)，这些场景包括分别使用社会力模型和 ORCA 作为环境中的行人策略

的随机场景 (SFM-random 和 ORCA-random) 和圆形场景 (SFM-circular 和 ORCA-circular), 以及一个有 5 个使用相同 PPO 策略多机器人的圆形场景 (PPO-circular)。不同模型在五种环境中的表现 (即每种方法在每种场景的 500 个不同环境中的平均成功率) 如图4.15所示。在训练过程中使用多种策略作为行人策略的模型 (PPO-multi) 在这五种环境中都是最优的。其中, 只使用社会力行人模型训练的策略 (PPO-SFM) 在使用社会力模型作为行人策略的测试环境中表现良好, 但在 ORCA 作为行人策略的两个环境中性能有所降低。只使用 ORCA 行人模型训练的策略 (PPO-ORCA) 在社会力行人模型环境 (SFM-random, SFM-circular) 中的实现情况略差于其他两个模型。同时, 在多机器人的圆形环境 (PPO-circular) 中, 使用单一策略作为环境中行人策略的两个模型 (PPO-SFM 和 PPO-ORCA) 表现较差。

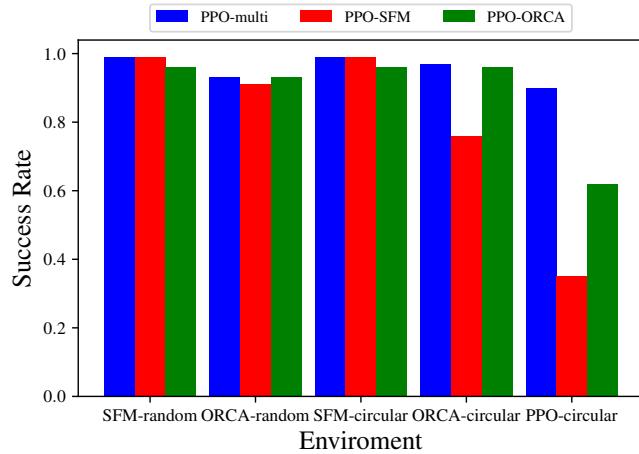


图 4.15 三种 PPO 策略在五种不同行人策略环境的成功率对比

4.4.4 实体实验

我们也将训练过的模型部署到真实的机器人上, 在现实世界中进行人群导航实体实验。如图4.16所示, 测试机器人是以 TurtleBot 2 为底盘搭建的, 同时使用 Hokuyo UTM-30LX 激光扫描测距仪作为二维激光传感器, 使用 RealSense D455 深度相机作为行人感知深度相机。行人感知追踪使用4.2节介绍的方法。该机器人使用两个 NVIDIA Jetson TX2 作为其计算平台, 其中一个用于行人感知计算, 另一个用于运行深度强化学习导航算法。在实验中, 基于状态估计器的粒子滤波提供了机器人的相对局部目标。基于激光数据构建以自我为中心的传感器地图, 行人追踪模块返回的行人速度和位置信息生成行人地图。传感器地图和行人地图都是固定尺寸 $6.0 \times 6.0\text{m}$, 分辨率 0.125m 。在测试中我们使用纸盒作为静态障碍物。我们在三种测试环境中进行了定性测试, 包括静态障碍物环境、动态行人环境和真实的走廊环境。在三种场景下, 机器人在大多数情况下都能

安全高效地完成导航任务。其中单人动态环境和多人动态环境的测试轨迹图如图4.17所示，走廊测试效果如图4.18所示，具体测试效果细节请参考我们的演示视频：<https://www.bilibili.com/video/BV1Vb4y1D7R6>。

这些实验表明，PPO-PSD 允许机器人根据多个传感器数据或传感器融合结果生成传感器地图和行人地图，并考虑了机器人的大小和形状，使其有效且易于部署到实际机器人中。

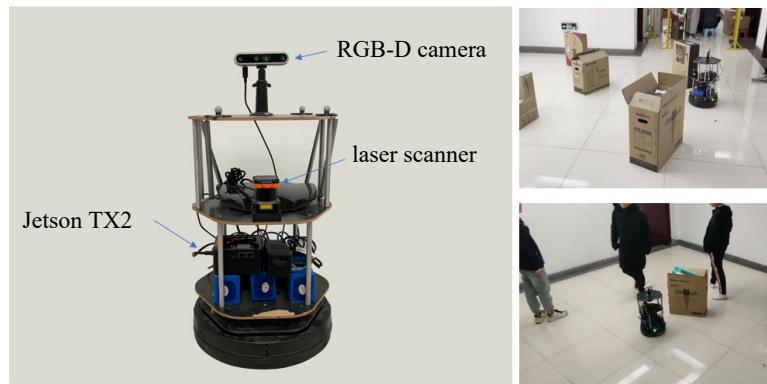


图 4.16 左图为配备 RealSense D455 深度相机和 Hokuyo UTM-30LX 激光的 TurtleBot 2 机器人；右上图为纸箱搭建的静态测试环境；右下图为行人测试环境

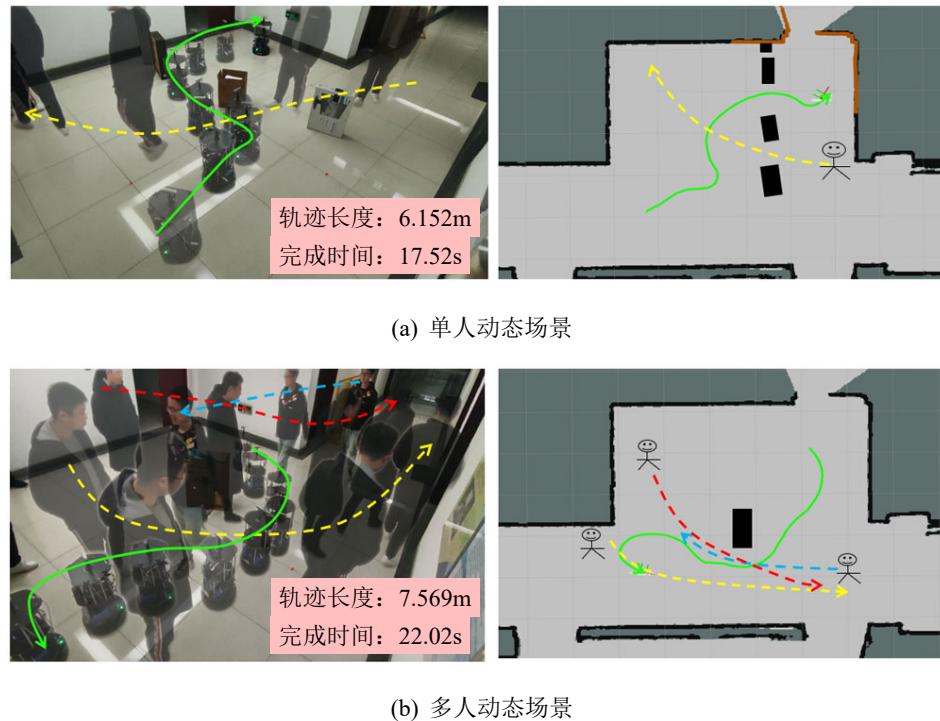


图 4.17 动态行人环境实体测试

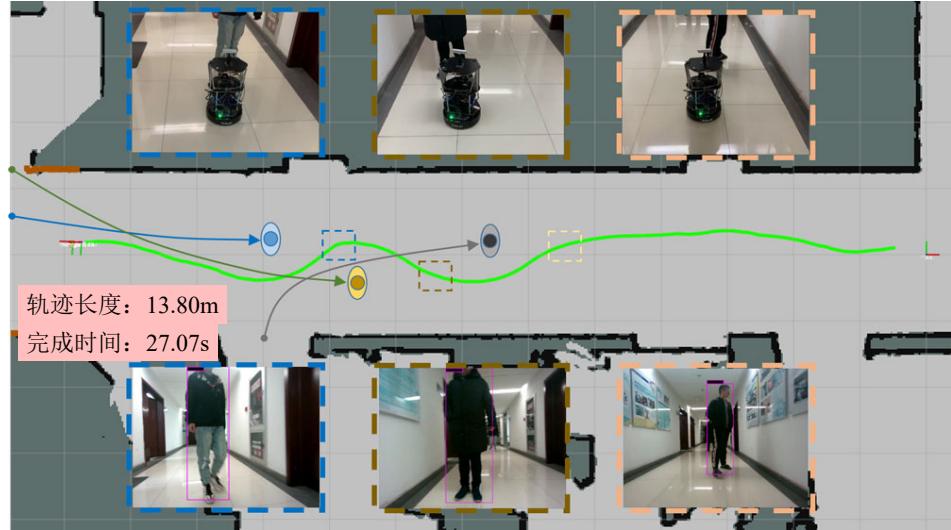


图 4.18 动态行人走廊场景实体测试

4.5 本章小结

目前的机器人导航系统应用于动态行人环境主要会面临两大难点，即导航定位丢失问题和机器人冻结问题。首先我们提出了多传感器行人感知追踪模块，我们的行人识别包含基于彩色图像的 YOLO v3 检测模块和基于 2D 激光的人腿感知模块，彩色图像的识别结果结合 3D 点云信息提取出行人的三维空间位置。然后我们使用最近邻融合 RGB-D 行人识别信息和 2D 激光行人识别信息，最后我们使用基于卡尔曼滤波的行人追踪模块追踪并计算行人的位置和速度信息。其次我们提出了解决动态环境下的长期稳定定位方法，我们首先根据感知的行人位置信息过滤掉激光被人腿遮挡的部分，实验结果显示对定位效果有一定改善作用。然而被动定位方法仍不能完全解决大量动态人腿遮挡激光问题，因此我们提出了基于二维码激光联合重定位的方法。当机器人定位信息和里程计信息差别较大或者定位方差较大时，机器人自主移动到二维码恢复区校准自身定位信息。我们的二维码激光联合重定位方法主要包含二维码检测识别、激光扫描匹配算法和基于卡尔曼滤波思想的状态估计方法。实验结果显示我们的重定位方法可以在定位完全丢失时仍能非常准确地重新定位机器人全局位置。最后我们在基于栅格地图的强化学习多机器人避障算法基础上，添加了由行人感知信息生成的行人地图输入，并设计了全新的基于社会力模型和 ORCA 的行人仿真器和训练场景。同时我们在回报函数中加入了机器人过于接近行人的惩罚，旨在使机器人移动中减少对行人的影响。我们在仿真器和实际机器人的多个场景下评估我们的方法。结果表明，我们的方法提高了机器人在多种行人策略环境下的避障成功率。

第 5 章 示范应用：成都大熊猫博物馆导览机器人

本章主要介绍大熊猫博物馆导览机器人（以下简称大熊猫导览机器人）系统，该机器人系统主要应用于博物馆环境中，给游客提供机器人导览服务。大熊猫导览机器人系统包含了前面 2-4 章介绍的主要机器人导航相关技术，并在多行为避障与人机交互方面做了一些新的探索，该导航系统已经在成都大熊猫繁育研究基地的大熊猫博物馆进行了大量实地场景测试。该导航系统的开发与测试依托于大熊猫导览机器人项目，该项目最终目标是让机器人更好地为游客提高信息导览等服务。本章我们主要介绍该项目的相关工作、应用挑战分析与解决方案设计、项目背景、需求分析、机器人软硬件系统设计、部署实施细节和实验结果。

5.1 导览机器人研究背景

过去几十年里在人们生活中出现了许多导览机器人，他们可以在各种场馆里穿梭，为游客提供导览服务。其中已知最早的博物馆导览机器人是 Wolfram 等人部署在德国一家博物馆的 Rhino 机器人^[182]，如图 5.1(a) 所示，Rhino 是德国波恩大学在 20 世纪 90 年代中期设计的室内导航机器人，其早期主要目标是能够在办公室环境内导航，从 A 点移动到 B 点，识别地板上的物体，并将它们扔进垃圾桶。Rhino 使用从 24 个声纳传感器和两个彩色摄像机接收到的数据建立栅格地图。后来在 1997 年 Rhino 进行了改进和加入了新特性，比如安全导航和吸引人的用户界面。Rhino 的新版本增加了激光传感器，可以提供更准确和可靠的环境数据。在与卡内基梅隆大学的合作下，Rhino 被改造成在德国博物馆进行为期六天的导游，一共接待游客 2000 多人。之后他们又开发了第二代导览机器人 Minerva^[183]，该机器人使用 DWA 算法作为避障模块并改进了人机交互功能，1998 年夏天，Minerva 在华盛顿特区的史密森尼美国国家历史博物馆进行了为期两周的专题展览服务。Minerva 的目标是通过引导参观者参观博物馆，并对一些展品进行简要的解释来娱乐和教育参观者。Rhino 和 Minerva 机器人部署复杂，需要专业人员一周的时间来手工建立高精的环境地图。同时它们不具备机器人全局定位异常时的恢复能力且基于反应式避障算法来处理动态行人环境避障问题，导航系统的长期稳定性仍没有绝对保障。但他们的工作使人们认识到移动机器人导航技术已经发展到一定水平，机器人甚至可以在人口稠密的空间中导航服务。从 2002 年 5 月 15 日到 10 月 20 日，服务机器人 Robox^[184]（图 5.1(b)）被部署到瑞士国家展览会 Expo.02，每天运行 12 个小时，行程超过 3315 公里，并

与 686,000 名游客互动。Robox 是一个完全自主的移动平台，具有独特的多模式交互能力和基于几何基元的环境地图格式，同时将基于图的全局规划与基于特征的定位相结合，并提出了一种结合动态窗口法和弹性带法的避障方法。但该机器人导航系统仍然平均 20-40 小时出现一次故障。为了提高游客对机器人的兴趣，有不少工作开始研发外形类人的导览机器人。机器人 HERMES^[185] 被部署在博物馆中连续运行了六个月并且每天工作 18 小时。Robovio 类人机器人^[186] 被部署在大阪科学博物馆进行为期两个月服务，文献显示博物馆参观者对在博物馆中看到机器人印象深刻，这些机器人有助于激发参观者对展品的兴趣。除了轮式机器人外，也有很多足式机器人被用于导览服务，如服务于博物馆的人形机器人 Pepper^[187] 和 ASIMO^[188] 等。近些年来随着传感器等机器人硬件和机器人算法的进步，导览机器人领域出现了更多鲁棒的应用。如图5.1(c)所示的 Kejia 机器人^[189] 在真实的国内大型购物商场里运行服务 40 多天，Sasaki 等人^[190] 在日本一家科技馆中无碰撞运行 120 小时。机器人 Lindsey^[191]（图5.1(d)）被部署在英国林肯收藏博物馆，为游客提供导游和信息服务。该机器人每天都在自动运行，在博物馆周围导航，并与公众互动，在其持续 7 个月的部署中，它已经旅行了近 300 公里，交付了超过 2300 次的导游服务。但机器人服务期间仍有硬件或软件故障，且需要机器人专家去现场协助解决。虽然已经有不少部署到真实博物馆提供导览服务工作的尝试，但完全解决机器人在开放环境中的长期自主 (Long-Term Autonomy, LTA) 问题^[192] 仍具有巨大挑战。

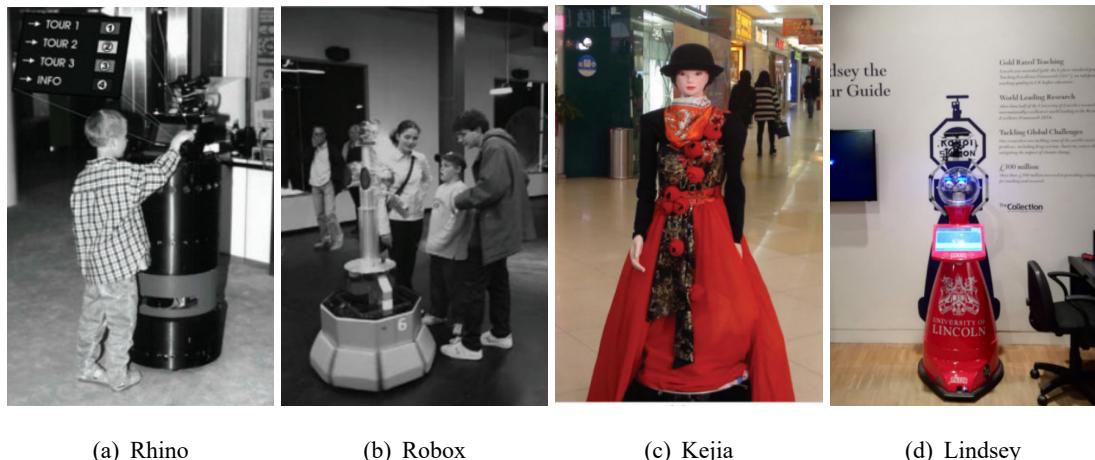


图 5.1 导览机器人实例

5.2 应用挑战分析与解决方案设计

近些年来，日常生活中已经出现了机器人充当博物馆导游、展览解说和迎宾员的现象。这种移动机器人的使用有可能使博物馆参观者享受到比人类导游提

供的更多内容形式的导览服务，机器人会使用不同语言和游客交互，可以按要求长时间运行，不需要像人类一样休息，可以减少人力需求。机器人的使用可以大大提高游客参观乐趣与体验^[193]。

然而，由于移动机器人在开放动态环境中长期稳定地自主移动导航仍然面临技术困难，导览机器人至今仍没有能够大规模应用。具体挑战包括：

- 动态环境定位问题：在当前的 SLAM 方法中，一个相当常见的假设是，当机器人移动时，世界保持不变（换句话说，地标是静态的）。这种静态世界的假设^[21-23]在小规模动态场景中运行短期定位是正确的，但在高度动态的行人环境中，激光和地图会发生错误的数据关联，从而导致定位发生错误。
- 定位异常感知与恢复：机器人定位系统出现异常时是很脆弱的，主要是因为几乎所有鲁棒的定位技术都基于迭代优化或滤波。一个理想的长期稳定定位解决方案应该是故障感知的，即系统需要知道即将发生的异常故障，并提供恢复机制来主动校准定位，现有的 SLAM 方法都不提供这些功能^[7]。
- 开放人群避障：传统的避障方法在密集动态人群中容易产生机器人冻结问题或频繁改变转向，这些避障方式会导致导航效率低下或者引发安全隐患。如何以行人友好方式高效地在密集动态人群中导航仍是目前未解决的问题^[194]。

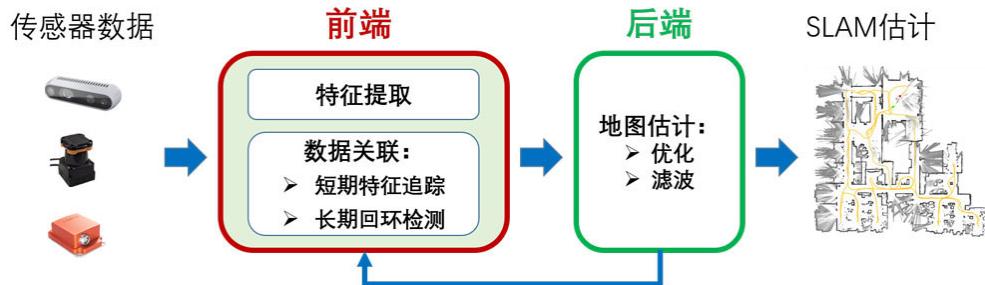


图 5.2 典型的 SLAM 框架图

SLAM 系统的典型架构（如图5.2所示）主要包括两个部分：前端和后端。前端将传感器数据抽象为可用于估计的模型，而后端对前端产生的抽象数据执行推理。前端的特征提取部分受传感器误差以及环境对传感器的影响，如光照对彩色相机的影响、轮子打滑对里程计的影响等。同时相似的环境结构也影响前端的数据关联部分，包括短期特征追踪和长期回环检测。前端输出信息的误差会影响后端的地图估计与定位，例如前端输出信息为基于优化算法后端的初始值，当优化初始值不佳时，可能使非线性优化过程陷入局部极小值，进而影响定位与建图精度。综上所述，机器人在开放环境中长期运行时很容易会发生定位异常，因此我们需要其他有效的解决开放环境 LTA 问题的具体实施方案。

当前智能机器人遇到的一个普遍性挑战是：单纯依靠现有算法本身，不能实

现完全自然条件下的应用（如完全的自动驾驶），这称为封闭性挑战^[195-196]。文献^[196]中作者指出符合强封闭性准则的工程项目中可应用暴力法、训练法或它们的集成，然而强封闭性准则中的失误非致命性和基础条件成熟性往往需要通过场景封闭化才能够满足。如果我们能在不影响游客体验和展出方需求的条件下对环境进行适当裁剪或改造使其封闭化，这将是解决长期自主导航（LTA）的可行方案。基于此准则，我们提出了基于二维码激光重定位的长期鲁棒定位方法和多行为导航避障策略。

在基于二维码激光重定位的长期鲁棒定位方法中，我们在环境中粘贴部署多处二维码恢复区来进行环境改造。这些二维码恢复区一般选择在机器人正常服务期间经常经过的区域，同时尽量不影响场馆的美观。由于我们在环境中人工加入了稳定的二维码视觉特征，会使得我们的定位算法更稳定，同时可以定期校准激光定位的长期累积误差。我们也考虑到如果二维码部署比较稀疏且开放环境中行人干扰较大时，仍有可能导致机器人定位瞬间发生较大误差。这种误差往往是由于当前环境过于复杂动态导致，而且由于不能限制游客与机器人交互，这种环境往往无法进行改造。在这种情况下，我们不再使用激光定位，而且把定位输出回退到上一正常帧并切换到里程计定位方式，里程计定位误差只和机器人硬件有关而与动态行人环境无关，这一意义上也是场景裁剪原理（以智能系统的可靠性、安全性为目标，对应用场景进行取舍，排除可能导致致命性失误或违反基础条件成熟性要求的情况），即把场景裁剪到不同定位方式的适用场景。如图5.3所示，机器人切换到里程计纯定位后，前往就近的二维码恢复区进行二维码激光重定位，当激光定位恢复正常后继续服务，需要注意的是，这里的里程计纯定位累积误差会被激光定位恢复正常后清除，因此不会从开机一直累计。

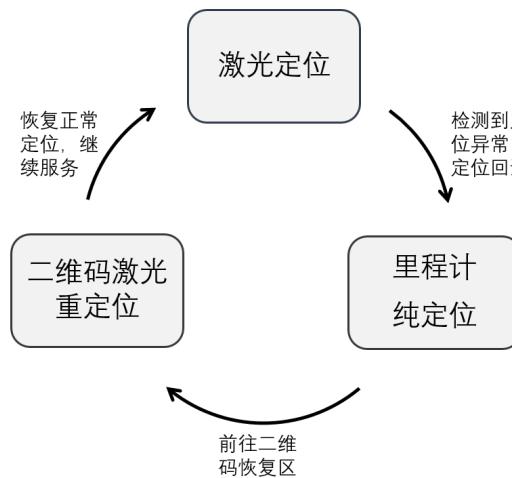


图 5.3 三种定位方式的切换策略

对封闭性准则的另一项重大挑战来自脆弱性。脆弱性的主要表现是：如果智能系统的输入不在训练好的人工神经网络的有效范围内，系统可产生错误的

输出。实际应用中无处不在的感知噪声和强化学习训练环境不具备代表性和多样性是脆弱性的一个主要原因。作者^[196]在强封闭性准则引入了失误非致命性要求，失误非致命性往往需要通过场景封闭化才能够满足。同样，在导航避障方面，我们也涉及到基于强封闭性准则的环境改造与裁剪。具体而言，我们实际部署的导航避障策略也包含三种行为：动态行人避障行为、停等避障行为和交互式语音提示。在不过分追求导航效率的应用条件下，停等避障行为可以避免动态行人避障行为动作的随意性，保证安全性。同时在人群特别拥挤或者行人注意力不集中的情况下，通过语音提示来进行交互式导航，让行人让路或提醒行人注意也是在主动改变环境，是人群导航安全有效的解决方式。

5.3 大熊猫导览机器人应用

本节我们详细介绍大熊猫导览机器人项目内容和大熊猫导览机器人软硬件系统，其中机器人软件系统包含了前面2-4章介绍的主要机器人导航相关技术。

5.3.1 项目背景与需求分析

为提升大熊猫繁育研究基地服务质量，增加游客旅游满意度，熊猫基地为新建的大熊猫博物馆研发定制大熊猫导览机器人。大熊猫导览机器人通过与游客自主交互，让游客在游览博物馆的同时了解场馆概况，通过与机器人互动获取科普知识，同时增添游览的趣味性与探索性。在其中的导航系统方面，我们主要考虑以下功能和需求：

- 巡航指路功能：机器人可以根据与游客语音交互内容，明确游客要到达的目的地，带领游客到指定场馆。当没有游客需要机器人服务时，机器人可在指定区域内自动巡航，主动寻找可以服务的对象，这一需求在导览机器人中普遍存在。大熊猫博物馆包含很多子场馆，游客需要根据地图寻找或多次询问工作人员，增加游客不便，而机器人根据规划最短路径带领游客很方便友好。
- 信息咨询服务：机器人可以给游客提供实时位置信息，帮助游客了解馆内所处位置。并且根据用户需要，为游客规划从指定起始点到目的点的游览路线。同时对于场馆内的功能区（存包处、卫生间等）机器人可以提示游客。机器人除了与游客语音交互外，还配备了显示器，可以显示博物馆全馆地图、最佳游览路线、与游客交互的知识问答内容等。
- 交互娱乐功能：机器人可以用语音和文字的方式进行基地总介绍和博物馆介绍，同时和游客娱乐聊天，回答游客问到的大熊猫相关科普知识，后期机器人也会配备手臂和面部动作，增强游客与机器人交流的趣味性。博物

馆引入机器人服务，可以提升游客的游览体验并且节省人力成本。同时机器人服务能够吸引更多游客的关注，给景区或博物馆带来更多的游客。

为了完成项目功能和需求，我们根据大熊猫场馆实际环境设计了大熊猫导览机器人软硬件系统。本项目的动机还包括：

- 在真实的场景中验证本文提出的机器人导航相关技术，在开放的博物馆场景中，我们可以得到大量真实的非专业人员的交互与反馈和项目实际部署的经验，从而更好地改进我们现有的方法。
- 越来越多的公共场所都有机器人部署服务的需要，通过本项目我们可以探索解决机器人在开放环境中的长期自主导航问题，让机器人导航技术早日更好地为人们提供日常服务。

5.3.2 机器人软硬件系统

如图5.4所示为我们当前机器人测试平台的硬件系统组成，需要说明的是，现有的模块为项目开发过程中导航必须模块，项目后期会升级机器人外形硬件和加入更多功能模块。我们的机器人底盘是两轮差速驱动的并包含四个支撑轮，前部安装有一个 Hokuyo UTM-30LX-EW 二维激光扫描测距仪和一个 XSENS MTi-7 惯性导航仪 (IMU)，机器人顶部装有一个 Realsense D455 RGB-D 相机用于感知追踪行人，同时还配备了用于发声的扬声器等外部设备，机器人计算单元包含一个 Intel NUC-10i7 和一个 NVIDIA Jetson AGX Xavier。



图 5.4 大熊猫导览机器人硬件组成

机器人软件系统基于机器人操作系统 ROS 搭建，如图5.5所示为大熊猫导览机器人软件系统架构，从下往上依次包含六层：

- 硬件层：主要是包括机器人硬件和传感器的驱动软件，激活与安装在平台



图 5.5 大熊猫导览机器人软件系统架构

上的不同设备(编码器、CAN卡、摄像头、激光等)的连接，以及与USB和以太网设备的连接。其中轮子编码器和惯性导航仪(IMU)融合成里程计数据，麦克风和扬声器用于收听和播放语音指令，显示器显示场馆地图信息和与游客问答交互信息。截止论文完成时间，声呐传感器和机器人手臂还没有加到硬件系统中。

- **控制层：** 主要包含机器人硬件的直接控制接口。包括底盘电机控制和手臂电机控制，机器人硬件的紧急急停。同时机器人还支持使用手柄无线遥控、网络的远程连接访问和一些底层的语音指令。
- **技能层：** 主要是指机器人的功能模块单元，分为视觉感知部分、移动导航部分和语音识别部分。机器人视觉感知部分主要是对游客的感知，包括行人的追踪、运动预测、视觉或激光行人检测和计划加入的人体姿态检测和情绪识别与情感交流。在机器人导航方面，包含机器人导航的相关基础功能模块，如建图、路径规划、局部避障、轨迹跟随和定位与重定位。最后是语音交互的基本功能，包括语音合成和语音识别，以及与本文导航系统不太相关的自然语音处理功能。
- **行为层：** 主要是指机器人的原子动作，该层的原子动作是基于机器人技能层的多个功能模块实现的。包含机器人的精确导航走点、我们提出的基于深度强化学习的移动导航、停等方式的导航行为、带领游客的导航方式和定义的一些手臂动作(如挥手)。

- 决策层：决策层是由一系列状态机组成的，状态机调用行为层和技能层的功能模块来完成复杂的特定任务，我们的机器人决策层目前包括多种导航行为的切换策略、定位异常检测与恢复和机器人电池电量管理。
- 应用层：应用层是面向项目实际应用开发的任务模块，目前我们的机器人应用层包括机器人长时间运行的集成规划、给游客提供服务的巡航指路、信息咨询和互动娱乐等功能。

5.4 部署过程与测试结果

5.4.1 测试环境介绍

成都大熊猫繁育研究基地年均接待游客近千万人次，成都大熊猫博物馆位于基地景区内，游客可持有效证件自由参观。该博物馆始建于1992年，2019年8月新馆开始布展建设，并于2021年3月3日对公众开放。我们的服务机器人部署在博物馆一楼大厅并主动移动导航为游客提供服务，一楼大厅总面积大约1000平方米，博物馆一楼实景图与平面图如图5.6所示。

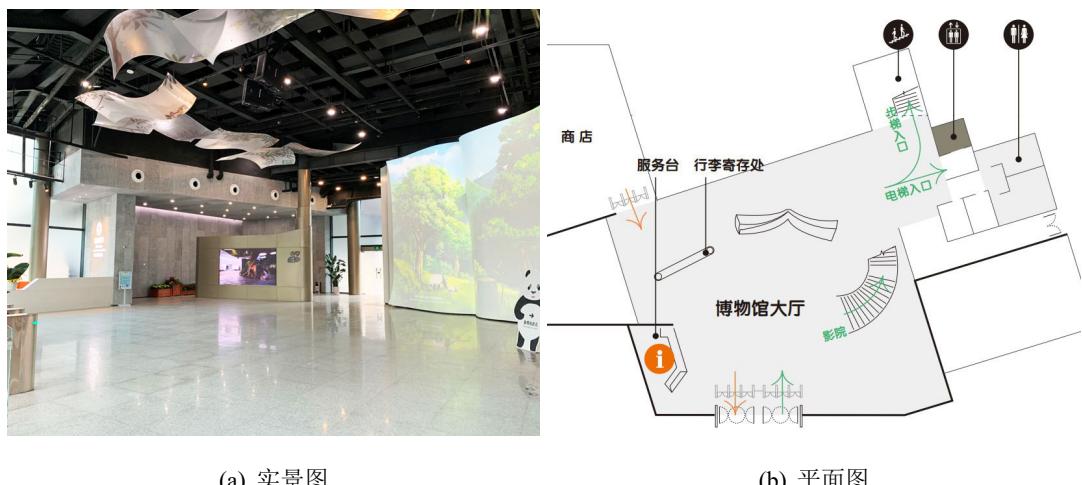


图 5.6 成都大熊猫博物馆大厅

对于机器人移动导航系统的部署，该测试环境有如下难点：

- 安全性非常重要。参观者可能对移动机器人没有任何经验，因此他们的行为会非常奇怪和令人惊讶。
- 环境中的墙面和柱子涂有反光材料，环境中包含大量透明与半透明玻璃，可能会使机器人激光定位或避障失败。
- 机器人长期运行或重启开机后，非机器人专业人员如何快速方便地进行全局定位。
- 场馆有大量游客，会大面积遮挡激光，可能使机器人激光定位失败。

- 我们需要在密集人群中确保安全有效的运动规划算法。同时机器人必须完成它的导游任务，即使有些情况下穿过人群非常困难。

5.4.2 机器人与多传感器标定

我们使用第2章提出的深度相机标定方法和通用批量标定框架来标定整个机器人和传感器，机器人与其各传感器外参标定结果如表5.1所示。由于我们通过

表 5.1 机器人与多传感器外参标定结果

坐标转换	\mathbf{R}_r	\mathbf{R}_p	\mathbf{R}_y	\mathbf{t}_x [mm]	\mathbf{t}_y [mm]	\mathbf{t}_z [mm]
Realsense \leftrightarrow Depth	-1.571	0.047	-1.571	11.361	18.112	13.429
Realsense \leftrightarrow Color	-1.576	0.028	-1.568	11.027	-41.052	13.294
Base \leftrightarrow Realsense	0.05423	0.095471	0.003576	220.129	1.832	218.932
Base \leftrightarrow Laser	0.10432	-0.03431	-0.43211	4.103	5.215	149.628

动作捕获系统可以得到机器人的实时位姿，因此我们可以计算出机器人实际运行期间的速度。如图5.7所示，我们发现机器人的里程计输出线速度比实际控制速度要小，具体差距随控制速度变化而变化。为了减小里程计误差，我们让机器

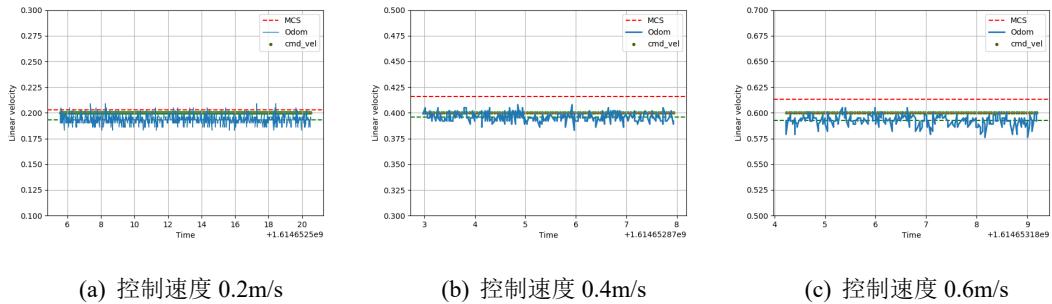


图 5.7 里程计线速度误差

人在动作捕获系统中以不同速度做匀速直线运动，收集大量里程计输出线速度和机器人实际线速度数据，然后我们使用一次函数拟合了里程计线速度 \dot{v} 与机器人实际线速度 v 的关系。修正后的机器人线速度 v 如下：

$$v = \begin{cases} \dot{v} & \text{if } \dot{v} < 0.05, \\ 1.01\dot{v} + 0.005 & \text{else.} \end{cases}$$

对于里程计输出的角速度 ω_o ，我们直接使用 IMU 估计的角速度输出 ω_i ，同时由于机器人静止时 IMU 一直存在微小偏移，因此最终机器人的角速度 ω 估计为：

$$\omega = \begin{cases} 0 & \text{if } \omega_o = 0, \\ \omega_i & \text{else.} \end{cases}$$

如图5.8所示，我们遥控机器人绕大熊猫场馆三圈，机器人行程约上百米，里程

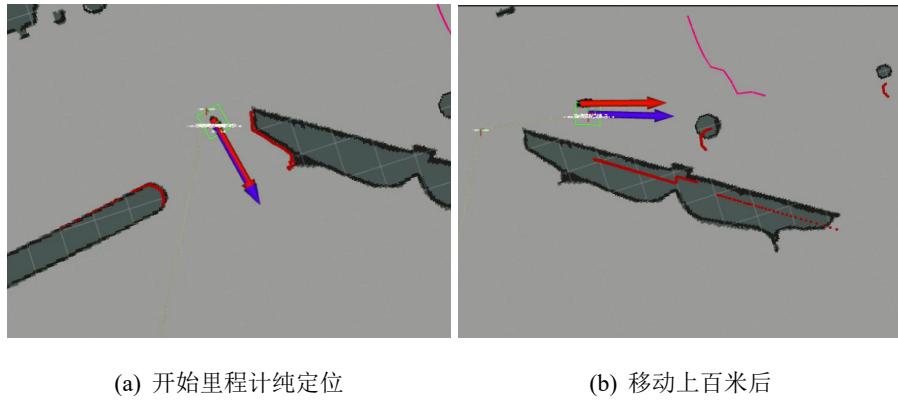


图 5.8 里程计纯定位前后误差（蓝色箭头为里程计纯定位位姿，红色箭头为激光定位位姿）

计位移偏移不到半米，角度偏差更微小。这说明我们校准后的机器人里程计非常准确，完全可以直接用于短期机器人定位。

5.4.3 地图构建

我们使用开源项目 cartographer^[142] 来进行激光建图，如图5.9 (a) 所示，我们手动遥控机器人绕场馆一周后建立场馆平面地图，从图中我们可以明显看出激光会部分穿透环境中的半透明玻璃和完全穿透环境中的全透明玻璃，同时环境中的柱子和闸机等涂有反光金属材料的物体在地图中不能完整清晰显示。建立的地图在导航中主要用于两方面：定位和规划。对于定位地图来说，我们原则上应最大程度上保证地图和实际场景相匹配。图5.9 (b) 是我们人为修改后的用于定位的地图，我们手动补全了环境中的半透明玻璃（主体建筑墙面）而忽略环境中的全透明玻璃，对于环境中表面涂有反光材料的物体，我们从多个视角建立子地图，然后手动把多视角子地图拼接融合成最终定位地图。同时为了让机器人安全地在环境中导航，我们在规划地图上设置了禁止区域，如图5.9 (c) 所示，我们在入口含有透明玻璃的闸机位置设置了一块禁止区，在楼梯间等低矮区域或其他非工作区也设置了机器人禁止区域。

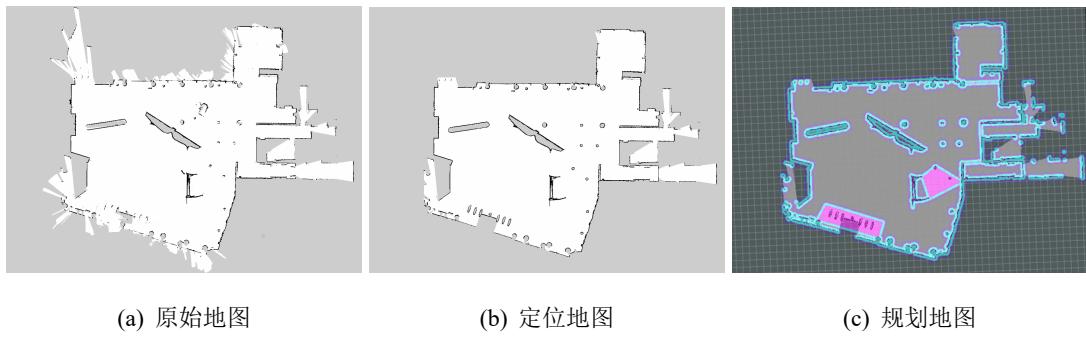


图 5.9 大熊猫场馆定位导航地图

5.4.4 二维码部署

为了不影响场馆的美观，如图5.10 (a) 所示，我们在行人很少出现的场馆大屏幕后方部署了一张二维码，实验表明即使只有一张二维码仍可以准确确定机器人全局位姿。如图5.10 (b) 所示，我们在环境设置 A、B、C 三个目标点，反复测试机器人导航并让机器人每回合在 C 点处进行一次二维码激光重定位，总测试时长达三个多小时。图5.10 (b) 显示了测试过程机器人的移动轨迹，机器人在每次经过图中二维码恢复区时校准一次机器人位姿，实验证明我们的二维码激光重定位性能是非常稳定的。另外，我们还部署测试了定位异常时机器人自动定位

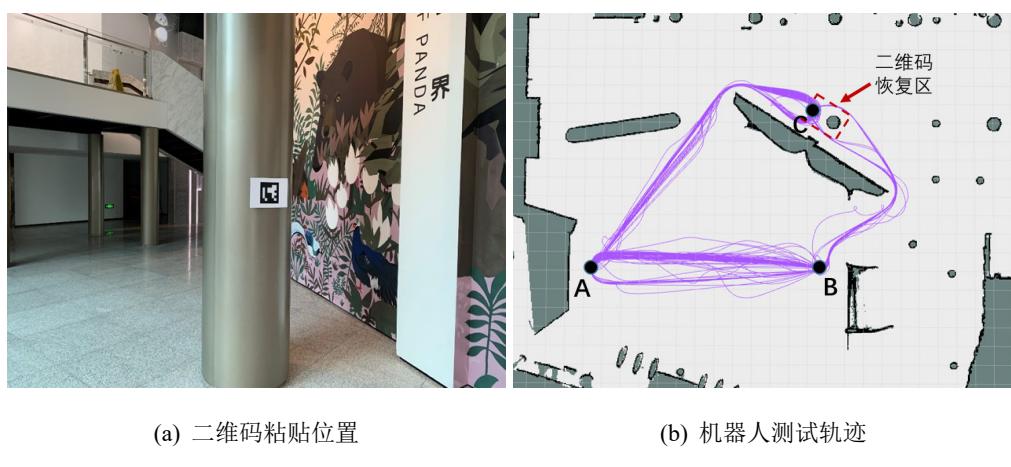


图 5.10 二维码重定位部署与测试

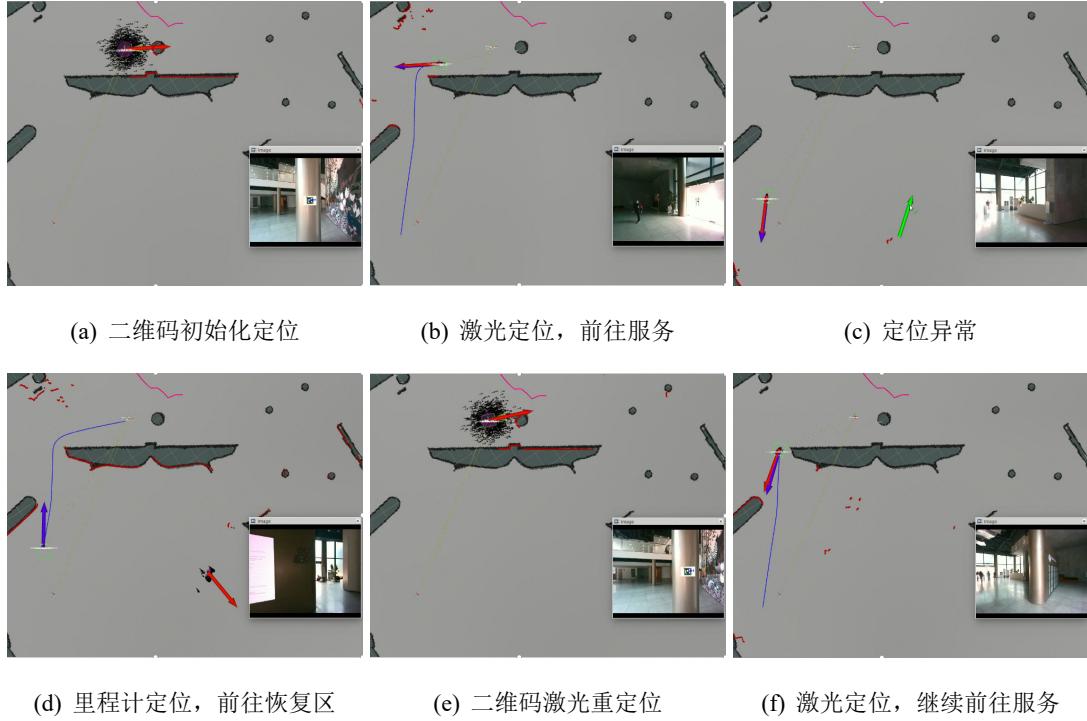


图 5.11 定位异常时二维码定位恢复策略测试

恢复的策略，如图5.3所示，我们设置了三种定位方式，包括激光普通定位、里程计纯定位和二维码激光重定位。机器人正常情况下处于激光定位状态，当检测到定位异常时，定位信息回退到上一帧正常的定位位置，并使用里程计的输出纯定位，根据里程计的定位信息导航前往二维码恢复区。在恢复区内根据二维码观察进行全局定位，然后把全局重定位信息用于初始化激光定位，机器人再次恢复到正常激光定位方式并继续投入工作。图5.11演示了整个定位恢复流程，在图5.11(c)中我们手动干扰机器人定位，但是机器人仍能自动监测到定位异常并根据里程计定位（图5.11(d)）成功前往二维码恢复区进行重定位（图5.11(e)）。有关实验中机器人的具体表现，请参见<https://www.bilibili.com/video/BV1q5411w7qX/>页面上的演示视频。

5.4.5 行人感知测试

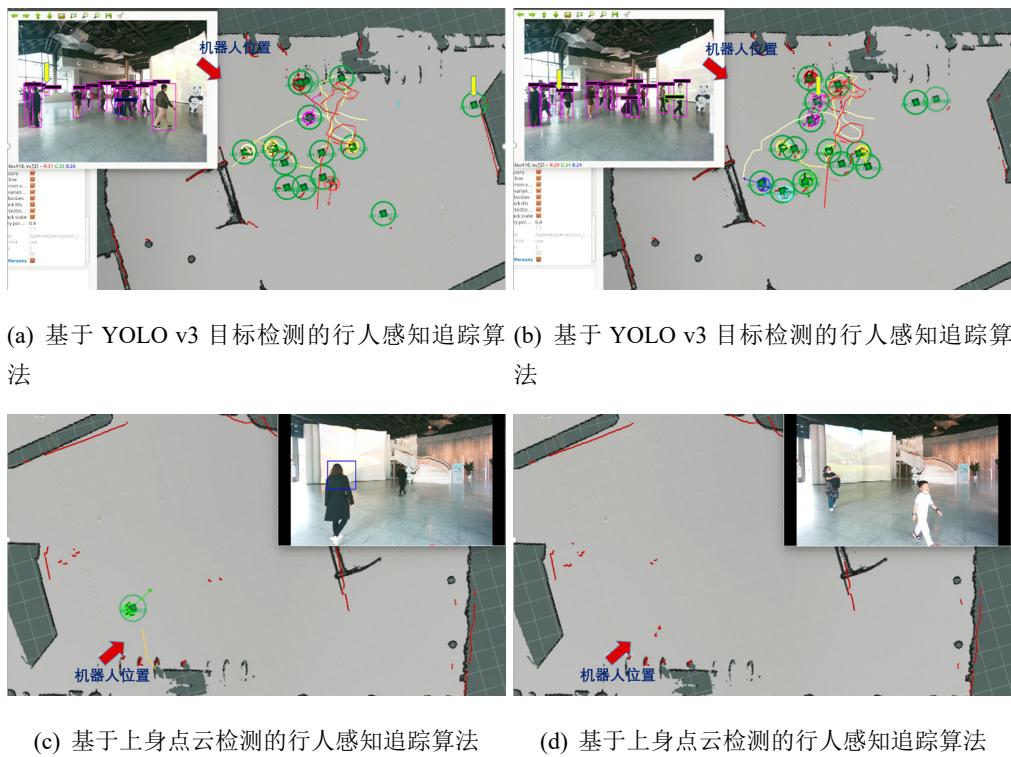


图 5.12 行人识别与追踪测试

我们在大熊猫博物馆实际场景中测试对比了两种行人感知追踪算法：基于 YOLO v3 目标检测的行人感知追踪算法和基于上身点云检测的行人感知追踪算法。如图5.12所示为两种算法的实际测试效果，我们发现基于 YOLO v3 目标检测的行人感知追踪算法可以识别很远距离的行人，但需要很多计算代价和 GPU 的支持，同时也很耗电。而基于上身点云检测的行人感知追踪算法只能识别到比较近距离的行人但仅需要 CPU 的计算支持。对比图5.12(b) 和 (d) 我们可以发现基于

YOLO v3 目标检测的行人感知追踪算法可以识别儿童或者蹲着的行人，但基于上身点云检测的行人感知追踪算法检测不到儿童或者蹲着的行人。而且从图5.12(a) 中我们也可以发现黄色箭头处的花瓶被基于 YOLO v3 目标检测的行人感知追踪算法误识别为行人，同时基于上身点云检测的行人感知追踪算法也存在漏检的情况。具体实验效果请参见<https://www.bilibili.com/video/BV1HB4y1P7xE/>页面上的演示视频。

5.4.6 密集人群导航测试

为了保证避障算法在密集人群中的安全性和高效性，我们使用三种避障行为：4.4节介绍的动态行人避障行为、停等避障行为和交互式语音提示，其中停等策略是基于 VFH^[197] 算法修改的，该策略会一直跟随全局规划的路径直至前方出现行人或者障碍物则停止。这两套避障策略的切换方式如图5.13所示，同时我们在某些遇到行人情况会主动语音提示行人注意。具体来说，导航正常情况使用动态行人避障，当机器人前方出现一个行人时，避障策略切换到停等策略并且机器人会说“请注意”，且三秒钟后或者前方没有行人时，避障策略会再回到动态行人避障。因为我们发现大部分时候行人只是路过机器人，等待一小会即可保持原路线继续前进，反复避障反而会让行人感觉不适，但同时我们并不想长期等待下去，因为配合语音提醒在前方只有一个人的情况下危险系数很低，因此我们最多等待三秒钟即开始避障绕开。当前方遇到的行人数目多于一人时，机器人会主动说“请让一让”并等待较长时间直至前方行人少于两人或者超时跳转。在

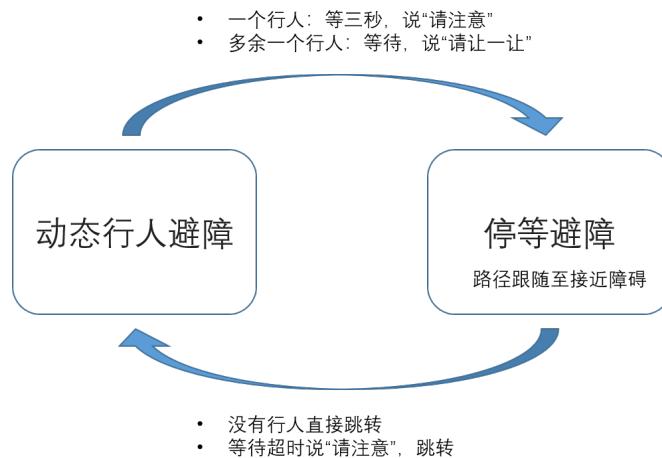


图 5.13 多行为人群避障算法的切换策略

图5.14中我们演示了当机器人遇到前方一个人和两个人的反应情况，在图5.14(a) 和 (b) 中，分别有一个行人在机器人前方经过，机器人短暂停等并说“请注意”后继续前进，和反应式避障 DWA 相比，机器人没有频繁改变转向且轨迹平滑。在图5.14(c) 中，机器人前方出现两个行人，机器人停等片刻后说“请让一让”，

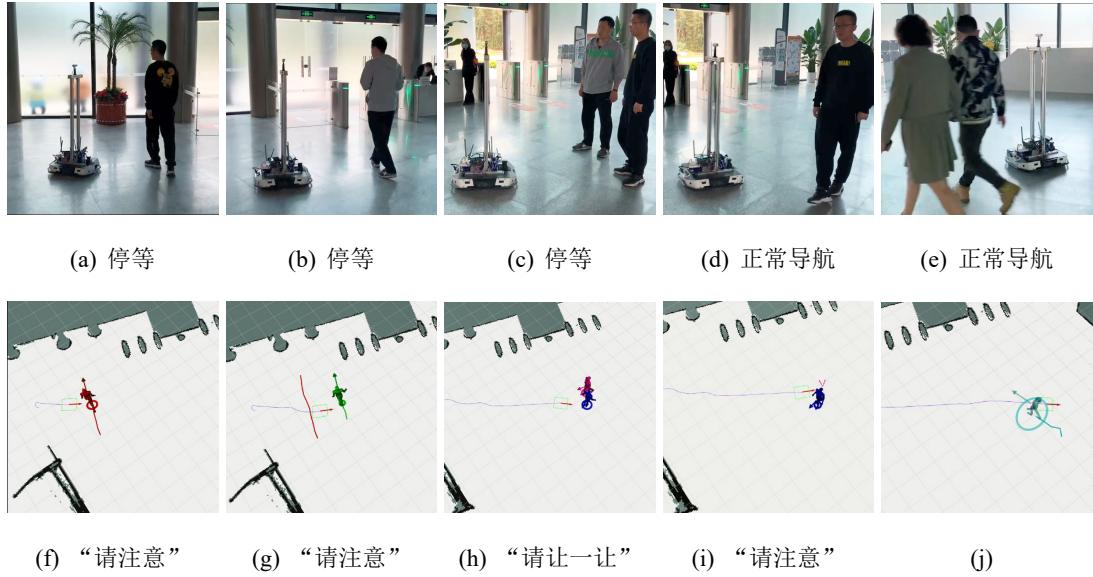


图 5.14 人群导航算法示范性测试

待其中一个行人让开后（图5.14(d)），机器人说“请注意”并继续前进。当遇到从机器人侧面经过的行人时，机器人正常导航前进（图5.14(e)）。图5.14(f)-(j) 分别为图5.14(a)-(e) 对应的机器人感知信息和机器人轨迹信息。

我们在高度复杂动态开放的大熊猫博物馆进行了大量实地测试，机器人顺利完成导航任务 3000 多次，其中不间断导航测试时长高达 7 个小时。具体机器人表现请参考我们上传的测试视频：<https://www.bilibili.com/video/BV13Z4y1A7br/>。图5.15为其中两天我们长时间测试机器人的轨迹图，其中 (b) 图描述了连续不间断测试七个小时的机器人轨迹。从机器人轨迹图我们可以看出，得益于我们多种避障行为的导航方式，机器人没有过分偏离全局路径且轨迹较为平滑。

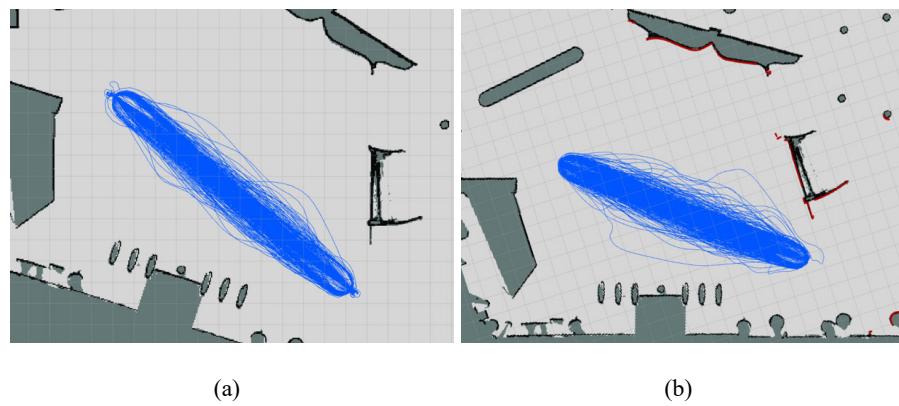


图 5.15 长时间机器人导航测试轨迹图

如图5.16所示，我们列举了机器人避开行人和小推车的场景。图5.16(a) 和 (b) 中行人故意挡在机器人面前，机器人说“请注意”并等待三秒钟，之后便安全避开行人继续前进。图5.16(c) 和 (d) 中机器人把小推车当做普通障碍物处理并成功

避开。我们在大熊猫博物馆做了大量实际开放测试，测试环境经常遇到大量动



图 5.16 机器人避让行人与小推车

态行人。如图5.17所示，我们列举了其中两段机器人遇到人群的实例，在第一段场景中（图5.17(a)-(e)），机器人遇到从它前方经过的行人会语音提示行人“请注意”，当机器人被一群人困住时，机器人主动说“请让一让”，一般可以有效脱离困境。在第二段场景中（图5.17(f)-(j)），机器人遇到了更加稠密的动态人群，但借助多行为导航方式，机器人仍能安全稳定地前往目标点。我们发现遇到行人主动语音说“请注意”或“请让一让”，通常情况下会让行人感觉机器人更智能，不会打扰行人。而且在某些行人环境中停等策略比简单反应式避障更高效，同时主动语音提醒对于拥挤环境运行是安全有效的策略。我们还发现由于机器人电



图 5.17 成都大熊猫博物馆的大量行人测试场景

机运行期间几乎没有声音，正在游览的游客很容易注意不到机器人的存在，从而可能存在潜在的危险。如图5.18所示，图(a)中行人正在和旁边的同伴交谈，很容易忽视旁边经过的机器人。图(b)中游客刚进入场馆，被周围环境吸引而忽视了侧面驶来的机器人。图(c)中一对情侣正在观看前方的大熊猫宣传视频，容易忽视后方行走的机器人。在这些情况下，机器人语音提示行人注意可以有效规避

潜在危险。大熊猫场馆还存在大量的儿童，这类游客行为具有更大的随意性与破



图 5.18 语音提示行人注意

坏性，而且儿童往往对机器人更感兴趣，更爱和机器人互动。如图5.19所示，我们列举了其中两段机器人遇到儿童的实例。儿童会故意阻挡机器人前进，而且我们的感知程序识别不到身高过低的儿童，因此机器人不会触发停等导航行为和语音提示。最终机器人随着儿童的运动不停改变运动朝向，试图脱离困境。更糟糕的是，场馆中还偶尔遇到调皮的儿童试图暴力干扰机器人正常运行。如图5.20(a)所示，一名儿童用力推机器人阻止其前进。甚至在图5.20(b)中一名儿童直接关闭了机器人急停开关。我们认为环境中儿童行为更加随意且对机器人的破坏性更大，未来我们在机器人软硬件设计方面应特殊考虑这种情况。



图 5.19 成都大熊猫博物馆的儿童测试场景

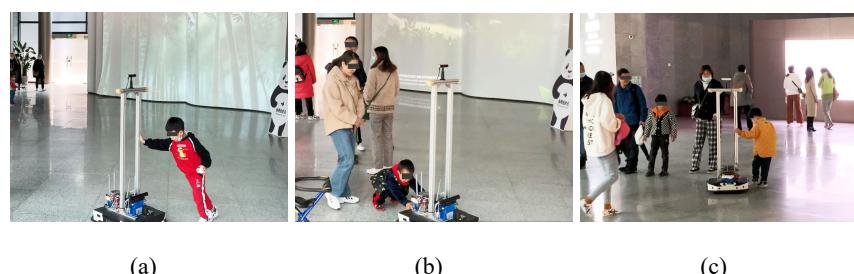


图 5.20 儿童对机器人的干扰

5.5 本章小结

本章我们详细介绍了大熊猫导览机器人项目和大熊猫导览机器人系统架构，大熊猫导览机器人系统包含了前面 2-4 章介绍的机器人与多传感器标定和动态行人导航系统。同时我们根据“封闭性准则”，提出了我们试图解决开放环境机器人长期自主导航的具体部署方案，包括基于二维码激光重定位的多定位方式切换策略和多行为导航避障策略。该导航系统已经在成都大熊猫繁育研究基地的大熊猫博物馆进行了部署和大量实地场景测试。我们详细介绍了实际部署过程中的机器人标定、地图构建、二维码部署测试、行人感知测试和密集人群导航测试。在这个高度开放复杂的实际动态人群环境中，机器人顺利完成导航任务 3000 多次，其中不间断导航测试时长高达 7 个小时。

第6章 总结与展望

6.1 工作总结

机器人导航是移动机器人的基础能力。随着人工智能、机器人技术和传感器技术的发展，越来越多的移动机器人被应用到开放的人群环境中。为了解决传统导航系统在开放动态人群环境下面临的困难与挑战，本文从机器人与多传感器标定、动态行人环境避障和动态环境定位等方面构建了一套完整的动态行人环境下的机器人导航系统。并且根据“封闭性准则”，本文提出了试图解决开放人群环境下的机器人长期自主导航的具体部署方案，成功在成都大熊猫研究繁育基地博物馆导览机器人上进行了大规模长时间测试。在这个高度复杂动态的密集人群环境中，机器人成功不间断导航测试时长高达7个小时。本文的具体工作总结如下：

(1) 提出了基于动作捕获系统的精确 RGB-D 相机标定方法。

我们完善了通用批量标定框架，提出了一种使用动作捕捉系统标定 RGB-D 传感器的可靠且准确的方法。在内参标定中，我们完整标定了深度图像和 IR 图像之间的像素偏移。在外参标定中，我们充分利用动作捕捉系统的特性来定制手眼标定的全局优化步骤，这进一步提高了外参标定的准确性。此外，采用非递归的新颖数据获取方法来获取深度图像中每个像素的真实值，并采用基于异方差高斯过程的单步深度校准方法来获取深度校准模型集的参数。所提出的标定方法比以前的相关工作获得了更好的结果，并且整个标定结果大大改善了基于深度相机传感器的视觉 SLAM 定位和建图效果。

(2) 提出了基于栅格地图融合信息和深度强化学习的多机器人避障算法。

我们提出了一种基于栅格地图的分布式近端策略优化方法，用于在分布式和无通信环境中避免多机器人碰撞。这项工作旨在说明基于栅格地图信息输入的强化学习避障算法的优势，且该多机器人避障算法是我们后续提出的基于深度强化学习的动态行人环境下避障算法基础，同时此多机器人避障算法也可以应用于多机器人无人仓储等实际场景。我们使用以机器人自我为中心的局部栅格图来表示周围的环境信息，包括机器人形状以及其他机器人和障碍物的可观察外观，这些信息可以通过使用多个传感器或传感器融合轻松生成。然后，我们将分布式近端策略优化应用于训练一个卷积神经网络，该网络直接将三帧以自我为中心的局部栅格图和机器人的相对局部目标位置映射为低级机器人控制命令。并且我们采用两阶段课程学习来提升学习效果。我们在多个仿真场景中评估学习到的避障策略，并将其与相关工作进行比较。实验结果表明，在成功率、额外时间以及平均线速度方面，我们的方法具有出色的性能。我们还从不同的角度

评估了我们学到的策略，包括泛化到看不见的场景、导航效率以及对机器人各种形状和动力学的鲁棒性。这些实验表明，基于地图的避障方法在许多指标上都优于其他方法。然后，我们将训练后的模型部署到真实的机器人上，以评估其在各种实际场景中的性能，包括具有静态和动态障碍物的环境、多机器人和动态行人环境。这些实验表明，我们的方法高效且易于部署到真实的机器人上，并且在现实世界中表现良好。

(3) 提出了动态行人环境下的包含行人感知、机器人避障与定位的导航系统。

我们首先提出了多传感器行人感知追踪模块，我们的行人识别包含基于彩色图像的 YOLO v3 检测模块和基于 2D 激光的人腿感知模块，然后使用基于卡尔曼滤波的行人追踪模块追踪并计算行人的位置和速度信息。当机器人在给定的地图上由于较大的定位不确定性或错误而无法准确定位时，就会出现导航丢失问题。在人群环境中定位失败主要是因为环境中存在大量动态的行人特征，因此我们首先根据感知的行人位置信息过滤掉激光被人腿遮挡的部分，实验结果显示对定位效果有一定改善作用。然而被动定位方法仍不能完全解决大量动态人腿遮挡问题，因此我们提出了基于二维码激光联合重定位的方法。当机器人定位信息和里程计信息差别较大或者定位方差较大时，机器人自主移动到二维码恢复区校准自身定位信息。实验结果显示我们的重定位方法可以在定位完全丢失时仍能非常准确地重新定位机器人全局位姿。机器人冻结问题的产生是因为机器人在高度动态的环境中倾向于保守的行为来最小化风险。在基于栅格地图的强化学习多机器人避障算法基础上，我们添加了由行人感知信息生成的行人地图输入，并设计了全新的基于社会力模型和 ORCA 的行人仿真场景。同时我们在回报函数中加入了机器人过于接近行人的惩罚，旨在使机器人移动中减少对行人的影响。我们在仿真器和实际机器人的多个场景下评估我们的方法，实验表明我们的方法进一步提升了机器人在不同行人策略的复杂环境中的避障成功率。

6.2 未来工作展望

本文提出了基于栅格地图和深度强化学习同时适用于复杂动态行人环境下的多传感器多机器人导航技术和方法，并通过大量的仿真和实体实验对相关技术进行了验证。然而我们的工作还有一些问题和不足，同时我们在实际部署中也产生了一些新的想法，现把下一步的研究方向列举如下：

1. 动态环境下定位算法的改进。现如今动态环境下的视觉 SLAM^[198] 和多传感器融合 SLAM^[199] 已是研究热点，我们认为动态环境下的定位技术是我

们整个导航系统能在复杂密集人群(如食堂和菜市场等环境)中导航的又一关键技术。

2. 增加与人避障的社会属性。通常在人类环境中工作的机器人使用把所有障碍物(包括人)都认为具有相似关联性的导航方式。为了避免让人感到不适,具有社会意识的机器人必须把人视为特殊的实体,并根据机器人的路径评估人们的舒适程度。这类导航行为我们称之为社会意识导航^[200-201],传统的解决方法需要人为考虑并定义社会规则,但有些规则是难以形式化描述的,基于数据驱动的方法可能会避开这个难点,因此我们认为基于深度强化学习的导航方式是解决此类复杂问题的未来研究方向。
3. 从仿真环境到现实世界的迁移。仿真器中的机器人任务学习有希望替代现实世界中代价高昂的实体机器人任务学习。不幸的是,在现实机器人上应用时,从仿真器中学到的策略通常比手工编写的策略执行得更差。其中很重要的原因是仿真器中智能体的运动转移函数和真实的差别比较大。在我们导航任务中,也就是机器人运动控制在仿真器和真实世界中不同。文献^[202]中作者在仿真器训练机器人执行动作时加入噪音,学出的避障策略更加好,但我们认为这样会使避障策略不精准。GAT 算法^[203]被提出解决这个问题,GAT 算法训练了真实世界的正向运动学方程和仿真世界的逆向运动学方程,从而使仿真器中的智能体动作和真实接近,但此类方法还未在机器人导航领域尝试应用。另一种解决思路是抽象表示网络输出代替速度输出,如使用轨迹点或者曲率等与机器人运动控制误差无关的控制信息。
4. 增加导航安全策略。基于学习的避障算法的一个挑战是缺乏对学习策略的安全性和完整性的保证,这被认为是在机器人技术中使用深度学习的主要问题之一。如何形式化地表述安全策略是需要深入研究的又一方向。^[204]
5. 交互式导航。传统的机器人导航着重于无碰撞地到达目标点,但是随着机器人越来越多地部署到家庭和办公室等非结构化和混乱的环境中,考虑机器人与环境的物理交互也是导航策略的一部分。例如在杂乱的房屋中导航时,机器人可能需要将物体推到一边或者打开门才能到达目的地,这种导航方式称为交互式导航^[205]。基于深度强化学习的导航是这类复杂问题很好的一个尝试方向。
6. 更多种廉价传感器方案。最后我们想尝试更多传感器类型,尤其是其中廉价的配置,如使用单目相机进行深度估计和避障等。进一步验证基于栅格地图输入的优势。

参 考 文 献

- [1] 陈羸峰. 大规模复杂场景下室内服务机器人导航的研究 [D]. 合肥: 中国科学技术大学, 2017.
- [2] HADDADIN S, SUPPA M, FUCHS S, et al. Towards the robotic co-worker [C]//Proceedings of the International Symposium of Robotics Research (ISRR). Lucerne, Switzerland: Springer, 2011: 261-282.
- [3] GROSS H M, MEYER S, SCHEIDIG A, et al. Mobile robot companion for walking training of stroke patients in clinical post-stroke rehabilitation [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Singapore: IEEE, 2017: 1028-1035.
- [4] SIEGWART R, NOURBAKHSH I R, SCARAMUZZA D. Introduction to autonomous mobile robots [M]. MIT press, 2011.
- [5] BARRERA A. Advances in robot navigation [M]. Mexico: BoD—Books on Demand, 2011.
- [6] ELFES A. Using occupancy grids for mobile robot perception and navigation [J]. Computer, 1989, 22(6): 46-57.
- [7] CADENA C, CARLONE L, CARRILLO H, et al. Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age [J]. IEEE Transactions on Robotics, 2016, 32(6): 1309-1332.
- [8] THRUN S. Probabilistic robotics [J]. Communications of the ACM, 2002, 45(3): 52-57.
- [9] TRAUTMAN P, KRAUSE A. Unfreezing the robot: Navigation in dense, interacting crowds [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Taipei, Taiwan: IEEE, 2010: 797-803.
- [10] GIUSTI A, GUZZI J, CIREŞAN D C, et al. A machine learning approach to visual perception of forest trails for mobile robots [J]. IEEE Robotics and Automation Letters, 2015, 1(2): 661-667.
- [11] GANDHI D, PINTO L, GUPTA A. Learning to fly by crashing [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vancouver, Canada: IEEE, 2017: 3948-3955.
- [12] PFEIFFER M, SCHAEUBLE M, NIETO J, et al. From perception to decision: A data-driven approach to end-to-end motion planning for autonomous ground robots [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Singapore: IEEE, 2017: 1527-1533.
- [13] SILVER D, SCHRITTWIESER J, SIMONYAN K, et al. Mastering the game of Go without human knowledge [J]. Nature, 2017, 550(7676): 354.

-
- [14] SILVER D, HUBERT T, SCHRITTWIESER J, et al. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play [J]. *Science*, 2018, 362(6419): 1140-1144.
 - [15] VINYALS O, BABUSCHKIN I, CZARNECKI W M, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning [J]. *Nature*, 2019, 575(7782): 350-354.
 - [16] BERNER C, BROCKMAN G, CHAN B, et al. Dota 2 with large scale deep reinforcement learning [J]. arXiv preprint arXiv:1912.06680, 2019.
 - [17] BAI Y, JIN C. Provable self-play algorithms for competitive reinforcement learning [J]. arXiv preprint arXiv:2002.04017, 2020.
 - [18] GULDENRING R, GÖRNER M, HENDRICH N, et al. Learning local planners for human-aware navigation in indoor environments [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Las Vegas, USA: IEEE, 2020: 6053-6060.
 - [19] FRAICHARD T, LEVESY V. From crowd simulation to robot navigation in crowds [J]. *IEEE Robotics and Automation Letters*, 2020, 5(2): 729-735.
 - [20] FOX D, BURGARD W, DELLAERT F, et al. Monte carlo localization: Efficient position estimation for mobile robots [C]//Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). Orlando, USA: AAAI press, 1999.
 - [21] WALCOTT-BRYANT A, KAESZ M, JOHANNSSON H, et al. Dynamic pose graph SLAM: Long-term mapping in low dynamic environments [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vilamoura, Portugal: IEEE, 2012: 1871-1878.
 - [22] XIAO L, WANG J, QIU X, et al. Dynamic-SLAM: Semantic monocular visual localization and mapping based on deep learning in dynamic environment [J]. *Robotics and Autonomous Systems*, 2019, 117: 1-16.
 - [23] HENEIN M, ZHANG J, MAHONY R, et al. Dynamic SLAM: The need for speed [C]// Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Paris, France: IEEE, 2020: 2123-2129.
 - [24] AHMADI E, MEGHDARI A, ALEMI M. A socially aware slam technique augmented by person tracking module [J]. *Journal of Intelligent & Robotic Systems*, 2019: 1-10.
 - [25] JEONG W, LEE K M. Cv-slam: A new ceiling vision-based slam technique [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2005: 3195-3200.
 - [26] KIM Y G, PARK T H. Localization of mobile robots from full detection of ceiling outlines [C]//Proceedings of the IEEE International Conference on Information and Automation

- (ICIA). IEEE, 2016: 1515-1520.
- [27] RIBACKI A, JORGE V A, MANTELLI M, et al. Vision-based global localization using ceiling space density [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018: 3502-3507.
- [28] ZHENG F, LIU Y H. Visual-odometric localization and mapping for ground vehicles using se (2)-xyz constraints [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2019: 3556-3562.
- [29] ZHANG H, ZHANG C, YANG W, et al. Localization and navigation using qr code for mobile robot in indoor environment [C]//Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO). IEEE, 2015: 2501-2506.
- [30] NAZEMZADEH P, FONTANELLI D, MACII D, et al. Indoor localization of mobile robots through qr code detection and dead reckoning data fusion [J]. IEEE/ASME Transactions on Mechatronics, 2017, 22(6): 2588-2599.
- [31] FOX D, BURGARD W, THRUN S. Markov localization for mobile robots in dynamic environments [J]. Journal of artificial intelligence research, 1999, 11: 391-427.
- [32] TIPALDI G D, MEYER-DELIUS D, BURGARD W. Lifelong localization in changing environments [J]. The International Journal of Robotics Research, 2013, 32(14): 1662-1678.
- [33] SUN D, GEISSE F, NEBEL B. Towards effective localization in dynamic environments [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Daejeon, South Korea: IEEE, 2016: 4517-4523.
- [34] BURGARD W, FOX D, THRUN S. Active mobile robot localization [C]//Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI). Nagoya, Japan: Morgan Kaufmann, 1997: 1346-1352.
- [35] LI A Q, XANTHIDIS M, O'KANE J M, et al. Active localization with dynamic obstacles [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Daejeon, South Korea: IEEE, 2016: 1902-1909.
- [36] ROY N, THRUN S. Online self-calibration for mobile robots [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA): volume 3. IEEE, 1999: 2292-2297.
- [37] DOMHOF J, KOOIJ J F, GAVRILA D M. An extrinsic calibration tool for radar, camera and lidar [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2019: 8107-8113.
- [38] MUELLER G R, WUENSCH H J. Continuous stereo camera calibration in urban scenarios [C]//Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC). Yokohama, Japan: IEEE, 2017: 1-6.

-
- [39] WU L, ZHU B. Binocular stereovision camera calibration [C]//Proceedings of the IEEE International Conference on Mechatronics and Automation (ICMA). Beijing, China: IEEE, 2015: 2638-2642.
 - [40] LING Y, SHEN S. High-precision online markerless stereo extrinsic calibration [C]// Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Daejeon, South Korea: IEEE, 2016: 1771-1778.
 - [41] SU R, ZHONG J, LI Q, et al. An automatic calibration system for binocular stereo imaging [C]//Proceedings of the IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference (IMCEC). Xi'an, China: IEEE, 2016: 896-900.
 - [42] DINH V Q, NGUYEN T P, JEON J W. Rectification using different types of cameras attached to a vehicle [J]. IEEE Transactions on Image Processing, 2018, 28(2): 815-826.
 - [43] VASCONCELOS F, BARRETO J P, NUNES U. A minimal solution for the extrinsic calibration of a camera and a laser-rangefinder [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 34(11): 2097-2107.
 - [44] PEREIRA M, SILVA D, SANTOS V, et al. Self calibration of multiple lidars and cameras on autonomous vehicles [J]. Robotics and Autonomous Systems, 2016, 83: 326-337.
 - [45] GEIGER A, MOOSMANN F, CAR Ö, et al. Automatic camera and range sensor calibration using a single shot [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). St. Paul, USA: IEEE, 2012: 3936-3943.
 - [46] ALMEIDA M, DIAS P, OLIVEIRA M, et al. 3d-2d laser range finder calibration using a conic based geometry shape [C]//Proceedings of the International Conference on Image Analysis and Recognition (ICIAR). Aveiro, Portugal: Springer, 2012: 312-319.
 - [47] GUINDEL C, BELTRÁN J, MARTÍN D, et al. Automatic extrinsic calibration for lidar-stereo vehicle sensor setups [C]//Proceedings of the IEEE International Conference on Intelligent Transportation Systems (ITSC). Yokohama, Japan: IEEE, 2017: 1-6.
 - [48] KWON Y C, JANG J W, CHOI O. Automatic sphere detection for extrinsic calibration of multiple RGBD cameras [C]//Proceedings of the International Conference on Control, Automation and Systems (ICCAS). PyeongChang, South Korea: IEEE, 2018: 1451-1454.
 - [49] KHAN A, ARAGON-CAMARASA G, SUN L, et al. On the calibration of active binocular and RGBD vision systems for dual-arm robots [C]//Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO). Qingdao, China: IEEE, 2016: 1960-1965.
 - [50] BASSO F, MENEGATTI E, PRETTO A. Robust intrinsic and extrinsic calibration of RGB-D cameras [J]. IEEE Transactions on Robotics, 2018, 34(5): 1315-1332.
 - [51] ZHANG Q, PLESS R. Extrinsic calibration of a camera and laser range finder (improves

- camera calibration) [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS): volume 3. Sendai, Japan: IEEE, 2004: 2301-2306.
- [52] LEVINSON J, THRUN S. Automatic online calibration of cameras and lasers. [C]// Proceedings of the Robotics: Science and Systems (RSS): volume 2. Berlin, Germany, 2013: 7.
- [53] GAO D, DUAN J, YANG X, et al. A method of spatial calibration for camera and radar [C]// Proceedings of the World Congress on Intelligent Control and Automation (WCICA). Jinan, China: IEEE, 2010: 6211-6215.
- [54] LIAO Y, LI G, JU Z, et al. Joint kinect and multiple external cameras simultaneous calibration [C]//Proceedings of the International Conference on Advanced Robotics and Mechatronics (ICARM). Hefei and Tai'an, China: IEEE, 2017: 305-310.
- [55] REHDER J, SIEGWART R, FURGALE P. A general approach to spatiotemporal calibration in multisensor systems [J]. IEEE Transactions on Robotics, 2016, 32(2): 383-398.
- [56] PRADEEP V, KONOLIGE K, BERGER E. Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach [C]//Proceedings of the International Symposium on Experimental Robotics (ISER). Marrakech and Essaouira, Morocco: Springer, 2014: 211-225.
- [57] OLIVEIRA M, CASTRO A, MADEIRA T, et al. A ROS framework for the extrinsic calibration of intelligent vehicles: A multi-sensor, multi-modal approach [J]. Robotics and Autonomous Systems, 2020, 131: 103558.
- [58] ZHENG K, CHEN Y, WU F, et al. A general batch-calibration framework of service robots [C]//Proceedings of the International Conference on Intelligent Robotics and Applications (ICIRA). Wuhan, China: Springer, 2017: 275-286.
- [59] ENDRES F, HESS J, STURM J, et al. 3-D mapping with an RGB-D camera [J]. IEEE Transactions on Robotics, 2014, 30(1): 177-187.
- [60] TIAN B, SHIM V A, YUAN M, et al. RGB-D based cognitive map building and navigation [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Tokyo, Japan: IEEE, 2013: 1562-1567.
- [61] BO L, REN X, FOX D. Learning hierarchical sparse features for RGB-(D) object recognition [J]. The International Journal of Robotics Research, 2014, 33(4): 581-599.
- [62] ALENYÀ G, FOIX S, TORRAS C. Using ToF and RGBD cameras for 3D robot perception and manipulation in human environments [J]. Intelligent Service Robotics, 2014, 7(4): 211-220.
- [63] Technical description of kinect calibration [EB/OL]. http://wiki.ros.org/%20kinect_calibration/technical/.
- [64] Ros freenect launch for kinect [EB/OL]. http://wiki.ros.org/freenect_launch.

-
- [65] HERRERA D, KANNALA J, HEIKKILÄ J. Joint depth and color camera calibration with distortion correction [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(10): 2058-2064.
 - [66] JUNG J, JEONG Y, PARK J, et al. A novel 2.5D pattern for extrinsic calibration of tof and camera fusion system [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). San Francisco, USA: IEEE, 2011: 3290-3296.
 - [67] STARANOWICZ A N, BROWN G R, MORBIDI F, et al. Practical and accurate calibration of RGB-D cameras using spheres [J]. *Computer Vision and Image Understanding*, 2015, 137: 102-114.
 - [68] SMISEK J, JANCOSEK M, PAJDLA T. 3D with Kinect [M]//Consumer depth cameras for computer vision. Springer, 2013: 3-25.
 - [69] ZHANG C, ZHANG Z. Calibration between depth and color sensors for commodity depth cameras [M]//Computer vision and machine learning with RGB-D sensors. Springer, 2014: 47-64.
 - [70] TEICHMAN A, MILLER S, THRUN S. Unsupervised intrinsic calibration of depth sensors via SLAM [C]//Proceedings of the Robotics: Science and Systems (RSS): volume 248. Berlin, Germany, 2013: 3.
 - [71] BASSO F, PRETTO A, MENEGATTI E. Unsupervised intrinsic and extrinsic calibration of a camera-depth sensor couple [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Hong Kong, China: IEEE, 2014: 6244-6249.
 - [72] STURM J, ENGELHARD N, ENDRES F, et al. A benchmark for the evaluation of RGB-D SLAM systems [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vilamoura-Algarve, Portugal: IEEE, 2012: 573-580.
 - [73] PEREZ-YUS A, FERNANDEZ-MORAL E, LOPEZ-NICOLAS G, et al. Extrinsic calibration of multiple RGB-D cameras from line observations [J]. *IEEE Robotics and Automation Letters*, 2018, 3(1): 273-280.
 - [74] KIM Y M, CHAN D, THEOBALT C, et al. Design and calibration of a multi-view TOF sensor fusion system [C]//Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Anchorage, USA: IEEE, 2008: 1-7.
 - [75] MEI C, RIVES P. Calibration between a central catadioptric camera and a laser range finder for robotic applications [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Orlando, USA: IEEE, 2006: 532-537.
 - [76] SCARAMUZZA D, HARATI A, SIEGWART R. Extrinsic self calibration of a camera and a 3D laser range finder from natural scenes [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). San Diego, USA: IEEE, 2007: 4164-

- 4169.
- [77] ALMOSALLAM I A, JARVIS M J, ROBERTS S J. GPZ: non-stationary sparse gaussian processes for heteroscedastic uncertainty estimation in photometric redshifts [J]. Monthly Notices of the Royal Astronomical Society, 2016, 462(1): 726-739.
- [78] ZHANG Z. A flexible new technique for camera calibration [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000, 22(11): 1330-1334.
- [79] XIANG W, CONLY C, MCMURROUGH C D, et al. A review and quantitative comparison of methods for kinect calibration [C]//Proceedings of the International Workshop on Sensor-based Activity Recognition and Interaction (iWOAR). Rostock, Germany: ACM, 2015: 3.
- [80] CANNY J. A computational approach to edge detection [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986, PMAI-8(6): 679-698.
- [81] YUEN H, PRINCEN J, ILLINGWORTH J, et al. Comparative study of Hough transform methods for circle finding [J]. Image and Vision Computing, 1990, 8(1): 71-77.
- [82] ZHAO Z. Hand-eye calibration using convex optimization [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Shanghai, China: IEEE, 2011: 2947-2952.
- [83] DIAMOND S, BOYD S. CVXPY: A Python-embedded modeling language for convex optimization [J]. Journal of Machine Learning Research, 2016, 17(83): 1-5.
- [84] SOMMER H, FORBES J R, SIEGWART R, et al. Continuous-time estimation of attitude using B-splines on lie groups [J]. Journal of Guidance, Control, and Dynamics, 2015, 39(2): 242-261.
- [85] FURRER F, FEHR M, NOVKOVIC T, et al. Evaluation of combined time-offset estimation and hand-eye calibration on robotic datasets [C]//Proceedings of the International Conference on Field and Service Robotics (FSR). Zurich, Switzerland: Springer, 2018: 145-159.
- [86] Wrighteagle 2019 team description paper [EB/OL]. <https://cgdsss.github.io/pdf/tdp19.pdf>.
- [87] WANG J, OLSON E. AprilTag 2: Efficient and robust fiducial detection [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Daejeon, South Korea: IEEE, 2016: 4193-4198.
- [88] WANG H, ZHANG C, SONG Y, et al. Master-followed multiple robots cooperation SLAM adapted to search and rescue environment [J]. International Journal of Control, Automation and Systems, 2018, 16(6): 2593-2608.
- [89] LI Z, BARENJI A V, JIANG J, et al. A mechanism for scheduling multi robot intelligent warehouse system face with dynamic demand [J]. Journal of Intelligent Manufacturing, 2020, 31(2): 469-480.
- [90] TRUONG X T, NGO T D. Toward socially aware robot navigation in dynamic and crowded

- environments: A proactive social motion model [J]. *IEEE Transactions on Automation Science and Engineering*, 2017, 14(4): 1743-1760.
- [91] SHALEV-SHWARTZ S, SHAMMAH S, SHASHUA A. Safe, multi-agent, reinforcement learning for autonomous driving [J]. arXiv preprint arXiv:1610.03295, 2016.
- [92] WANG Z, HU J, LV R, et al. Personalized privacy-preserving task allocation for mobile crowdsensing [J]. *IEEE Transactions on Mobile Computing*, 2018, 18(6): 1330-1341.
- [93] GE X, HAN Q L. Distributed formation control of networked multi-agent systems using a dynamic event-triggered communication mechanism [J]. *IEEE Transactions on Industrial Electronics*, 2017, 64(10): 8118-8127.
- [94] MICHAEL N, FINK J, KUMAR V. Cooperative manipulation and transportation with aerial robots [J]. *Autonomous Robots*, 2011, 30(1): 73-86.
- [95] ALONSO-MORA J, BAKER S, RUS D. Multi-robot formation control and object transport in dynamic environments via constrained optimization [J]. *The International Journal of Robotics Research*, 2017, 36(9): 1000-1021.
- [96] FIORINI P, SHILLER Z. Motion planning in dynamic environments using velocity obstacles [J]. *The International Journal of Robotics Research*, 1998, 17(7): 760-772.
- [97] VAN DEN BERG J, GUY S J, LIN M, et al. Reciprocal n-body collision avoidance [M]// *Robotics research*. Springer, 2011: 3-19.
- [98] SNAPE J, VAN DEN BERG J, GUY S J, et al. The hybrid reciprocal velocity obstacle [J]. *IEEE Transactions on Robotics*, 2011, 27(4): 696-706.
- [99] BAREISS D, VAN DEN BERG J. Generalized reciprocal collision avoidance [J]. *The International Journal of Robotics Research*, 2015, 34(12): 1501-1514.
- [100] HENNES D, CLAES D, MEEUSSEN W, et al. Multi-robot collision avoidance with localization uncertainty [C]//*Proceedings of the International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*. Valencia, Spain: IFAAMAS, 2012: 147-154.
- [101] CLAES D, HENNES D, TUYLS K, et al. Collision avoidance under bounded localization uncertainty [C]//*Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vilamoura, Portugal: IEEE, 2012: 1192-1198.
- [102] GODOY J E, KARAMOUZAS I, GUY S J, et al. Implicit coordination in crowded multi-agent navigation [C]//*Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*. Phoenix, USA: AAAI Press, 2016: 2487-2493.
- [103] ZHOU D, WANG Z, BANDYOPADHYAY S, et al. Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells [J]. *IEEE Robotics and Automation Letters*, 2017, 2(2): 1047-1054.
- [104] SNAPE J, VAN DEN BERG J, GUY S J, et al. Smooth and collision-free navigation for

- multiple robots under differential-drive constraints [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Taipei, Taiwan: IEEE, 2010: 4584-4589.
- [105] ALONSO-MORA J, BEARDSLEY P, SIEGWART R. Cooperative collision avoidance for nonholonomic robots [J]. IEEE Transactions on Robotics, 2018, 34(2): 404-420.
- [106] ALONSO-MORA J, BREITENMOSER A, RUFLI M, et al. Optimal reciprocal collision avoidance for multiple non-holonomic robots [M]//Distributed autonomous robotic systems. Springer, 2013: 203-216.
- [107] PHILLIPS M, LIKHACHEV M. Sipp: Safe interval path planning for dynamic environments [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Shanghai, China: IEEE, 2011: 5628-5635.
- [108] AOUDE G S, LUDERS B D, JOSEPH J M, et al. Probabilistically safe motion planning to avoid dynamic obstacles with uncertain motion patterns [J]. Autonomous Robots, 2013, 35 (1): 51-76.
- [109] TAI L, LI S, LIU M. A deep-network solution towards model-less obstacle avoidance [C]//Proceedings of the IEEE/RSJ international conference on intelligent robots and systems (IROS). Daejeon, South Korea: IEEE, 2016: 2759-2764.
- [110] GU S, HOLLY E, LILLICRAP T, et al. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Singapore: IEEE, 2017: 3389-3396.
- [111] LEVINE S, PASTOR P, KRIZHEVSKY A, et al. Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection [J]. The International Journal of Robotics Research, 2018, 37(4-5): 421-436.
- [112] XIE L, WANG S, ROSA S, et al. Learning with training wheels: speeding up training with a simple controller for deep reinforcement learning [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Brisbane, Australia: IEEE, 2018: 6276-6283.
- [113] CHEN Y F, LIU M, EVERETT M, et al. Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Singapore: IEEE, 2017: 285-292.
- [114] CHEN Y F, EVERETT M, LIU M, et al. Socially aware motion planning with deep reinforcement learning [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Vancouver, Canada: IEEE, 2017: 1343-1350.
- [115] LONG P, FAN T, LIAO X, et al. Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning [C]//Proceedings of the IEEE International Conference

- on Robotics and Automation (ICRA). Brisbane, Australia: IEEE, 2018: 6252-6259.
- [116] FAN T, LONG P, LIU W, et al. Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios [J]. The International Journal of Robotics Research, 2020: 0278364920916531.
- [117] CHEN G, CUI G, JIN Z, et al. Accurate intrinsic and extrinsic calibration of rgb-d cameras with gp-based depth correction [J]. IEEE Sensors Journal, 2018, 19(7): 2685-2694.
- [118] CHEN G, YAO S, MA J, et al. Distributed non-communicating multi-robot collision avoidance via map-based deep reinforcement learning [J]. Sensors, 2020, 20(17): 4836.
- [119] FOX D, BURGARD W, THRUN S. The dynamic window approach to collision avoidance [J]. IEEE Robotics & Automation Magazine, 1997, 4(1): 23-33.
- [120] VAN DEN BERG J, LIN M, MANOCHA D. Reciprocal velocity obstacles for real-time multi-agent navigation [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Pasadena, USA: IEEE, 2008: 1928-1935.
- [121] GUY S J, CHHUGANI J, KIM C, et al. Clearpath: highly parallel collision avoidance for multi-agent simulation [C]//Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA). New Orleans, Louisiana: ACM, 2009: 177-187.
- [122] KAELBLING L P, LITTMAN M L, CASSANDRA A R. Planning and acting in partially observable stochastic domains [J]. Artificial intelligence, 1998, 101(1-2): 99-134.
- [123] MORAVEC H P. Sensor fusion in certainty grids for mobile robots [J]. AI magazine, 1988, 9(2): 61-74.
- [124] LU D V, HERSHBERGER D, SMART W D. Layered costmaps for context-sensitive navigation [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Chicago, USA: IEEE, 2014: 709-715.
- [125] NG A Y, HARADA D, RUSSELL S. Policy invariance under reward transformations: Theory and application to reward shaping [C]//Proceedings of the International Conference on Machine Learning (ICML): volume 99. Bled, Slovenia: Morgan Kaufmann, 1999: 278-287.
- [126] SUTTON R S, BARTO A G. Reinforcement learning: An introduction [M]. MIT press, 2018.
- [127] SCHULMAN J, WOLSKI F, DHARIWAL P, et al. Proximal policy optimization algorithms [J]. arXiv preprint arXiv:1707.06347, 2017.
- [128] SCHULMAN J, MORITZ P, LEVINE S, et al. High-dimensional continuous control using generalized advantage estimation [J]. arXiv preprint arXiv:1506.02438, 2015.
- [129] KINGMA D P, BA J. Adam: A method for stochastic optimization [J]. arXiv preprint arXiv:1412.6980, 2014.
- [130] NAIR V, HINTON G E. Rectified linear units improve restricted boltzmann machines [C]//

- Proceedings of the International Conference on Machine Learning (ICML). Haifa, Israel: Omnipress, 2010.
- [131] BENGIO Y, LOURADOUR J, COLLOBERT R, et al. Curriculum learning [C]//Proceedings of the International Conference on Machine Learning (ICML). Montreal, Canada: ACM, 2009: 41-48.
- [132] ELMAN J L. Learning and development in neural networks: The importance of starting small [J]. *Cognition*, 1993, 48(1): 71-99.
- [133] CHEN G, PAN L, CHEN Y, et al. Robot navigation with map-based deep reinforcement learning [J]. arXiv preprint arXiv:2002.04349, 2020.
- [134] KOENIG N, HOWARD A. Design and use paradigms for gazebo, an open-source multi-robot simulator [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS): volume 3. Sendai, Japan: IEEE, 2004: 2149-2154.
- [135] VAUGHAN R. Massively multi-robot simulation in stage [J]. *Swarm intelligence*, 2008, 2 (2-4): 189-208.
- [136] BRESENHAM J E. Algorithm for computer control of a digital plotter [J]. *IBM Systems journal*, 1965, 4(1): 25-30.
- [137] XIE L, WANG S, MARKHAM A, et al. Towards monocular vision based obstacle avoidance through deep reinforcement learning [C]//Proceedings of the Robotics: Science and Systems Workshop: New Frontiers for Deep Learning in Robotics (RSS). Cambridge, USA, 2017.
- [138] KOREN Y, BORENSTEIN J, et al. Potential field methods and their inherent limitations for mobile robot navigation. [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA): volume 2. 1991: 1398-1404.
- [139] BRITO B, EVERETT M, HOW J P, et al. Where to go next: Learning a subgoal recommendation policy for navigation in dynamic environments [J]. *IEEE Robotics and Automation Letters*, 2021, 6(3): 4616-4623.
- [140] KÄSTNER L, BUIYAN T, ZHAO X, et al. Connecting deep-reinforcement-learning-based obstacle avoidance with conventional global planners using waypoint generators [J]. arXiv preprint arXiv:2104.03663, 2021.
- [141] DUCHOŃ F, BABINEC A, KAJAN M, et al. Path planning with modified a star algorithm for a mobile robot [J]. *Procedia Engineering*, 2014, 96: 59-69.
- [142] HESS W, KOHLER D, RAPP H, et al. Real-time loop closure in 2D LIDAR SLAM [C]// Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Stockholm, Sweden: IEEE, 2016: 1271-1278.
- [143] PELLEGRINI S, ESS A, SCHINDLER K, et al. You'll never walk alone: Modeling social behavior for multi-target tracking [C]//Proceedings of the IEEE International Conference on

- Computer Vision (ICCV). Kyoto, Japan: IEEE, 2009: 261-268.
- [144] VEMULA A, MUELLING K, OH J. Modeling cooperative navigation in dense human crowds [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Singapore: IEEE, 2017: 1685-1692.
- [145] BORENSTEIN J, KOREN Y. Real-time obstacle avoidance for fast mobile robots [J]. IEEE Transactions on Systems, Man, and Cybernetics, 1989, 19(5): 1179-1187.
- [146] COSÍO F A, CASTANEDA M P. Autonomous robot navigation using adaptive potential fields [J]. Mathematical and computer modelling, 2004, 40(9-10): 1141-1156.
- [147] EVERETT M, CHEN Y F, HOW J P. Collision avoidance in pedestrian-rich environments with deep reinforcement learning [J]. IEEE Access, 2021, 9: 10357-10377.
- [148] CHEN C, LIU Y, KREISS S, et al. Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning [C]//Proceedings of the International Conference on Robotics and Automation (ICRA). Montreal, Canada: IEEE, 2019: 6015-6022.
- [149] CHEN C, HU S, NIKDEL P, et al. Relational graph learning for crowd navigation [J]. arXiv preprint arXiv:1909.13165, 2019.
- [150] SIAGIAN C, ITTI L. Biologically inspired mobile robot vision localization [J]. IEEE Transactions on Robotics, 2009, 25(4): 861-873.
- [151] CHANG C K, SIAGIAN C, ITTI L. Beobot 2.0: Autonomous mobile robot localization and navigation in outdoor pedestrian environment [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Tokyo, Japan: IEEE, 2013: 2079-2079.
- [152] ROY N, BURGARD W, FOX D, et al. Coastal navigation-mobile robot navigation with uncertainty in dynamic environments [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Detroit, USA: IEEE, 1999: 35-40.
- [153] VAN DEN BERG J, PATIL S, ALTEROVITZ R. Motion planning under uncertainty using iterative local optimization in belief space [J]. The International Journal of Robotics Research, 2012, 31(11): 1263-1278.
- [154] PLATT R, TEDRAKE R, KAELBLING L, et al. Belief space planning assuming maximum likelihood observations [C]//Proceedings of the Robotics: Science and Systems (RSS). Zaragoza, Spain, 2010.
- [155] FAN T, CHENG X, PAN J, et al. Getting robots unfrozen and unlost in dense pedestrian crowds [J]. IEEE Robotics and Automation Letters, 2019, 4(2): 1178-1185.
- [156] ARRAS K O, MOZOS O M, BURGARD W. Using boosted features for the detection of people in 2d range data [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Roma, Italy: IEEE, 2007: 3402-3407.

-
- [157] LEIGH A, PINEAU J, OLMEDO N, et al. Person tracking and following with 2d laser scanners [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Seattle, USA: IEEE, 2015: 726-733.
 - [158] SUDOWE P, LEIBE B. Efficient use of geometric constraints for sliding-window object detection in video [C]//Proceedings of the International Conference on Computer Vision Systems (ICVS). Sophia Antipolis, France: Springer, 2011: 11-20.
 - [159] SPINELLO L, ARRAS K O. People detection in RGB-D data [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). San Francisco, USA: IEEE, 2011: 3838-3843.
 - [160] JAFARI O H, MITZEL D, LEIBE B. Real-time RGB-D based people detection and tracking for mobile robots and head-worn cameras [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Hong Kong, China: IEEE, 2014: 5636-5643.
 - [161] SPINELLO L, TRIEBEL R, SIEGWART R. Multiclass multimodal detection and tracking in urban environments [J]. *The International Journal of Robotics Research*, 2010, 29(12): 1498-1515.
 - [162] LINDER T, BREUERS S, LEIBE B, et al. On multi-modal people tracking from mobile platforms in very crowded and dynamic environments [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Stockholm, Sweden: IEEE, 2016: 5512-5519.
 - [163] BREIMAN L. Random forests [J]. *Machine learning*, 2001, 45(1): 5-32.
 - [164] REDMON J, DIVVALA S, GIRSHICK R, et al. You only look once: Unified, real-time object detection [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, USA: IEEE, 2016: 779-788.
 - [165] REDMON J, FARHADI A. Yolov3: An incremental improvement [J]. arXiv preprint arXiv:1804.02767, 2018.
 - [166] UHLMANN J. Introduction to the algorithmics of data association in multiple-target tracking [J]. *Handbook of Multisensor Data Fusion*, 2001: 3-1.
 - [167] BELLOTTO N, HU H. Multisensor-based human detection and tracking for mobile service robots [J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 2008, 39(1): 167-181.
 - [168] PFEIFFER M, PAOLO G, SOMMER H, et al. A data-driven model for interaction-aware pedestrian motion prediction in object cluttered environments [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Brisbane, Australia: IEEE, 2018: 5921-5928.
 - [169] WULFMEIER M, RAO D, WANG D Z, et al. Large-scale cost function learning for path

- planning using deep inverse reinforcement learning [J]. *The International Journal of Robotics Research*, 2017, 36(10): 1073-1087.
- [170] EPPENBERGER T, CESARI G, DYM CZYK M, et al. Leveraging stereo-camera data for real-time dynamic obstacle detection and tracking [J]. arXiv preprint arXiv:2007.10743, 2020.
- [171] DORFMÜLLER K, WIRTH H. Real-time hand and head tracking for virtual environments using infrared beacons [C]//Proceedings of the International Workshop on Modelling and Motion Capture Techniques for Virtual Environments (CAPTECH). Springer, 1998: 113-127.
- [172] RIBO M, PINZ A, FUHRMANN A L. A new optical tracking system for virtual and augmented reality applications [C]//Proceedings of the IEEE International Instrumentation and Measurement Technology Conference (I2MTC). Budapest, Hungary: IEEE, 2001: 1932-1936.
- [173] NAIMARK L, FOXLIN E. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker [C]//Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR). Darmstadt, Germany: IEEE, 2002: 27-36.
- [174] ROHS M, GFELLER B. Using camera-equipped mobile phones for interacting with real-world objects [C]//Proceedings of the International Conference on Advances in Grid and Pervasive Computing (GPC). 2004.
- [175] KATO H, BILLINGHURST M. Marker tracking and hmd calibration for a video-based augmented reality conferencing system [C]//Proceedings of the IEEE and ACM International Workshop on Augmented Reality (IWAR). IEEE, 1999: 85-94.
- [176] FLOHR D, FISCHER J. A Lightweight ID-Based Extension for Marker Tracking Systems [C]//FROEHLICH B, BLACH R, VAN LIERE R. Proceedings of the Eurographics Symposium on Virtual Environments. The Eurographics Association, 2007.
- [177] OLSON E. AprilTag: A robust and flexible visual fiducial system [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Shanghai, China: IEEE, 2011: 3400-3407.
- [178] GARRIDO-JURADO S, MUÑOZ-SALINAS R, MADRID-CUEVAS F J, et al. Automatic generation and detection of highly reliable fiducial markers under occlusion [J]. *Pattern Recognition*, 2014, 47(6): 2280-2292.
- [179] KRETZSCHMAR H, SPIES M, SPRUNK C, et al. Socially compliant mobile robot navigation via inverse reinforcement learning [J]. *The International Journal of Robotics Research*, 2016, 35(11): 1289-1307.

- [180] GLOOR C. PedSim: Pedestrian crowd simulation [EB/OL]. <http://pedsim.silmaril.org>.
- [181] HELBING D, MOLNAR P. Social force model for pedestrian dynamics [J]. Physical review E, 1995, 51(5): 4282.
- [182] BURGARD W, CREMERS A B, FOX D, et al. Experiences with an interactive museum tour-guide robot [J]. Artificial intelligence, 1999, 114(1-2): 3-55.
- [183] THRUN S, BEETZ M, BENNEWITZ M, et al. Probabilistic algorithms and the interactive museum tour-guide robot minerva [J]. The International Journal of Robotics Research, 2000, 19(11): 972-999.
- [184] SIEGWART R, ARRAS K O, BOUABDALLAH S, et al. Robox at Expo. 02: A large-scale installation of personal robots [J]. Robotics and Autonomous Systems, 2003, 42(3-4): 203-222.
- [185] BISCHOFF R, GRAEFE V. Demonstrating the humanoid robot HERMES at an exhibition: A long-term dependability test [C]//Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Lausanne, Switzerland: IEEE, 2002.
- [186] SHIOMI M, KANDA T, ISHIGURO H, et al. Interactive humanoid robots for a science museum [C]//Proceedings of the ACM SIGCHI/SIGART Conference on Human-robot Interaction (HRI). Salt Lake City, USA: ACM, 2006: 305-312.
- [187] ALLEGRA D, ALESSANDRO F, SANTORO C, et al. Experiences in using the pepper robotic platform for museum assistance applications [C]//Proceedings of the IEEE International Conference on Image Processing (ICIP). Athens, Greece: IEEE, 2018: 1033-1037.
- [188] IIO T, SATAKE S, KANDA T, et al. Human-like guide robot that proactively explains exhibits [J]. International Journal of Social Robotics, 2019: 1-18.
- [189] CHEN Y, WU F, SHUAI W, et al. Robots serve humans in public places—kejia robot as a shopping assistant [J]. International Journal of Advanced Robotic Systems, 2017, 14(3): 1729881417703569.
- [190] SASAKI Y, NITTA J. Long-term demonstration experiment of autonomous mobile robot in a science museum [C]//Proceedings of the IEEE International Symposium on Robotics and Intelligent Sensors (IRIS). Ottawa, Canada: IEEE, 2017: 304-310.
- [191] DEL DUCHETTO F, BAXTER P, HANHEIDE M. Lindsey the tour guide robot-usage patterns in a museum long-term deployment [C]//Proceedings of the IEEE International Conference on Robot and Human Interactive Communication (RO-MAN). New Delhi, India: IEEE, 2019: 1-8.
- [192] KUNZE L, HAWES N, DUCKETT T, et al. Artificial intelligence for long-term robot autonomy: A survey [J]. IEEE Robotics and Automation Letters, 2018, 3(4): 4023-4030.
- [193] PANG W C, WONG C Y, SEET G. Exploring the use of robots for museum settings and

- for learning heritage languages and cultures at the chinese heritage centre [J]. Presence: Teleoperators and Virtual Environments, 2018, 26(4): 420-435.
- [194] TRAUTMAN P, MA J, MURRAY R M, et al. Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation [J]. The International Journal of Robotics Research, 2015, 34(3): 335-356.
- [195] 陈小平. 封闭性场景: 人工智能的产业化路径 [J]. 文化纵横, 2020, 2: 34-42.
- [196] 陈小平. 人工智能中的封闭性和强封闭性——现有成果的能力边界, 应用条件和伦理风险 [J]. 智能系统学报, 2020, 15(1): 114-120.
- [197] ULRICH I, BORENSTEIN J. VFH*: Local obstacle avoidance with look-ahead verification [C]//Proceedings of the 2000 IEEE International Conference on Robotics and Automation (ICRA): volume 3. San Francisco, USA: IEEE, 2000: 2505-2511.
- [198] SAPUTRA M R U, MARKHAM A, TRIGONI N. Visual SLAM and structure from motion in dynamic environments: A survey [J]. ACM Computing Surveys, 2018, 51(2): 1-36.
- [199] BRESSON G, ALSAYED Z, YU L, et al. Simultaneous localization and mapping: A survey of current trends in autonomous driving [J]. IEEE Transactions on Intelligent Vehicles, 2017, 2(3): 194-220.
- [200] KRUSE T, PANDEY A K, ALAMI R, et al. Human-aware robot navigation: A survey [J]. Robotics and Autonomous Systems, 2013, 61(12): 1726-1743.
- [201] CHARALAMPOUS K, KOSTAVELIS I, GASTERATOS A. Recent trends in social aware robot navigation: A survey [J]. Robotics and Autonomous Systems, 2017, 93: 85-104.
- [202] CHOI J, PARK K, KIM M, et al. Deep reinforcement learning of navigation in a complex and crowded environment with a limited field of view [C]//Proceedings of the IEEE International Conference on Robotics and Automation (ICRA). Montreal, Canada: IEEE, 2019: 5993-6000.
- [203] HANNA J P, STONE P. Grounded action transformation for robot learning in simulation [C]//Proceedings of the AAAI Conference on Artificial Intelligence (AAAI). San Francisco, USA: AAAI press, 2017.
- [204] GARCIA J, FERNÁNDEZ F. A comprehensive survey on safe reinforcement learning [J]. Journal of Machine Learning Research, 2015, 16(1): 1437-1480.
- [205] XIA F, SHEN W B, LI C, et al. Interactive gibson benchmark: A benchmark for interactive navigation in cluttered environments [J]. IEEE Robotics and Automation Letters, 2020, 5(2): 713-720.

致 谢

光阴荏苒，在中国科学技术大学六年的硕博生活即将结束，我即将完成博士毕业论文。在此论文即将完成之际，我衷心地向所有关心、指导和帮助我的各位老师、同学、朋友和家人们送上我真诚的感谢！

首先我要特别感谢我敬爱的导师陈小平教授，在我硕博期间给我提供了许多帮助支持和参与一些重要的项目的机会，感谢您把我带入了机器人和人工智能这个令人兴奋的研究领域。您谆谆善诱、诲人不倦、德艺双馨，不仅在学识上为我树立了终生学习的榜样，而且在为人处世上为我树立了楷模。从 2015 年接纳我进入科大机器人实验室学习，2016 年和 2017 年参加 RoboCup@Home 国际大赛，再到 2019 年担任队长参加 IJCAI 养老机器人挑战赛，您给予了我充分的肯定与支持。陈老师宽广的胸襟、广阔的视野和严谨的研究态度给我留下了深刻的印象，必然使我受益终生。

其次，我要感谢一起准备并参加机器人比赛的队友们，包括帅威师兄、刘江川师兄、王希平师姐、周锋师兄、崔国伟同学、郑魁松同学、唐冰同学、张泽坤同学、晋忠孝师弟、张钊师弟、林楠师弟和廖志涵师弟，和你们一起拼搏的日子特别让我怀念。尤其感谢帅威师兄、刘江川师兄和王希平师姐，是你们带领我学习机器人技术并参加机器人比赛。感谢程敏博士、陈凯博士、赵哲博士、陈羸峰博士、唐可可博士和卢栋才博士，与你们短暂的相处也使我受益匪浅，感谢你们为实验室打下的坚实科研基础。感谢一路相伴的实验室博士生崔国伟同学、郑魁松同学、唐冰同学、姜皓同学、仵沛宸师弟、林楠师弟、曹荣昀师弟和殷实师弟，祝你们科研有成，早日完成学业。感谢指导帮助我科研的吉建民教授和吴锋教授，与你们交流拓宽了我的知识面，产生了新的思路，给了我科研很大的帮助。你们扎实的科研能力、踏实努力的作风、热心帮助他人的习惯给了我很大的影响。感谢庄庆昊师弟、潘李凡师弟、陈宇铵师弟、徐佩师弟、姚舜一师弟、马骏师弟、彭杰师弟、王志强师弟、陈树言师弟、王晶师妹和裘憲成师弟，与你们每周的组会讨论给了我很多新想法和帮助，衷心祝愿你们能够取得更大的学术成果。感谢相处多年已踏入职场的陈荣亚师兄、张钊师弟、鲁科烃师弟、晋忠孝师弟、叶子师弟、廖胜兰师妹、李煜煜师弟、刘庆周师弟和高杨师弟，祝你们前程似锦，事业有成。

感谢培育我的母校，中国科学技术大学，您给予了我宝贵的六年学习时光，提供了安静美好的校园环境和优秀的学习氛围，给了我完成人生中重大蜕变的机会。最后我要特别感谢二十多年来一直默默给予我支持关心爱护的我的父母和姐姐，他们给予了我坚强的后盾，他们的勤奋努力以及对待生活自强不息不屈

致 谢

不挠的精神深深影响了我。感谢身边朋友们的真诚关心，尤其是我的女朋友易颂佳同学，感谢你们对我的理解与包容，以及对我的付出和支持。

回首往事，历历在目。老师们的谆谆教导，同学们的热情帮助，家人们的默默支持，朋友们的关心鼓励都使我心生感激。即将毕业踏入人生的下一个阶段，无论前方将会遇到何种困难，我将怀着这份感激之情坚强勇敢地走下去！

陈广大

2021 年 3 月 6 日

在读期间发表的学术论文与取得的研究成果

已发表论文

1. **Guangda Chen**, Shunyi Yao, Jun Ma, Lifan Pan, Yu'an Chen, Pei Xu, Jianmin Ji, and Xiaoping Chen. "Distributed Non-Communicating Multi-Robot Collision Avoidance via Map-Based Deep Reinforcement Learning," *Sensors*, vol. 20, no. 17, p. 4836, Aug. 2020. (SCI, IF = 3.275)
2. Shunyi Yao*, **Guangda Chen***, Lifan Pan, Jun Ma, Jianmin Ji and Xiaoping Chen. "Multi-Robot Collision Avoidance with Map-based Deep Reinforcement Learning." *Proceedings of the 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2020. (CCF C)
3. **Guangda Chen**, Lifan Pan, Yu'an Chen, Pei Xu, Zhiqiang Wang, Peichen Wu, Jianmin Ji, and Xiaoping Chen. "Robot Navigation with Map-Based Deep Reinforcement Learning". *Proceedings of the 17th IEEE International Conference on Networking, Sensing and Control (ICNSC)*. IEEE, 2020. (**Best Student Paper Award**)
4. **Guangda Chen**, Guowei Cui, Zhongxiao Jin, Feng Wu, and Xiaoping Chen. "Accurate Intrinsic and Extrinsic Calibration of RGB-D Cameras With GP-Based Depth Correction," *IEEE Sensors Journal*, vol. 19, no. 7, pp. 2685-2694, April, 2019. (SCI, IF = 3.073)
5. Guowei Cui, **Guangda Chen**, Zekun Zhang, and Xiaoping Chen. "A Flexible Grasping Policy Based on Simple Robot-Camera Calibration and Pose Repeatability of Arm." *Proceedings of the International Conference on Intelligent Robotics and Applications (ICIRA)*. Springer, Cham, 2018.
6. Zhongxiao Jin, Guowei Cui, **Guangda Chen**, and Xiaoping Chen. "A Multi-view Images Generation Method for Object Recognition." *Proceedings of the International Conference on Intelligent Robotics and Applications (ICIRA)*. Springer, Cham, 2018.
7. Kuisong Zheng, **Guangda Chen**, Guowei Cui, Yingfeng Chen, Feng Wu, and Xiaoping Chen. "Performance Metrics for Coverage of Cleaning Robots with MoCap System." *Proceedings of the International Conference on Intelligent Robotics and Applications (ICIRA)*. Springer, Cham, 2017.

其他成果

专利

1. 陈广大, 姚舜一, 吉建民. 一种分布式多机器人的导航方法 [P]. 安徽省: CN112304314A, 2021-02-02.

竞赛

本人研究生期间参加过多项国内国际机器人竞赛并取得多项团队奖项:

1. 2015 年 9 月作为核心成员在中国贵阳参加全国机器人大赛暨 RoboCup 公开赛, 获第一名
2. 2016 年 7 月作为核心成员在德国莱比锡参加 RoboCup@Home 国际服务机器人大赛, 获第三名
3. 2016 年 10 月作为核心成员在中国北京参加世界机器人大赛 RoboCup 挑战赛, 获第一名
4. 2017 年 7 月作为核心成员在日本名古屋参加 RoboCup@Home 国际服务机器人大赛, 获最佳操作奖
5. 2019 年 8 月作为参赛队长在中国澳门参加 IJCAI-2019 养老机器人挑战赛, 获第一名