

室内移动机器人精确跟踪定位技术研究

作者姓名 孙煊溢

指导教师姓名、职称 张亮 副教授

申请学位类别 工学硕士

学校代码 10701
分 类 号 TN82

学 号 17031211602
密 级 秘密

西安电子科技大学

硕士学位论文

室内移动机器人精确跟踪定位技术研究

作者姓名：孙煊溢

一级学科：软件工程

二级学科（研究方向）：软件工程

学位类别：工学硕士

指导教师姓名、职称：张亮 副教授

学 院：计算机科学与技术学院

提交日期：2019年3月

Research on Accurate Tracking and Localization Technology of Indoor Mobile Robot

A Thesis submitted to
XIDIAN UNIVERSITY
in partial fulfillment of the requirements
for the degree of Master
in Software Engineering

By

Sun Xuanyi

Supervisor: Zhang Liang Title:Associate Professor

Martch 2019

西安电子科技大学 学位论文独创性（或创新性）声明

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文若有不实之处，本人承担一切法律责任。

本人签名：_____ 日 期：_____

西安电子科技大学 关于论文使用授权的说明

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权属于西安电子科技大学。学校有权保留送交论文的复印件，允许查阅、借阅论文；学校可以公布论文的全部或部分内容，允许采用影印、缩印或其它复制手段保存论文。同时本人保证，结合学位论文研究成果完成的论文、发明专利等成果，署名单位为西安电子科技大学。

保密的学位论文在____年解密后适用本授权书。

本人签名：_____ 导师签名：_____

日 期：_____ 日 期：_____

摘要

室内定位的发展使工作在室内的移动机器人能够提供更加智能化的服务。室内定位技术能够通过各式各样的传感器采集室内的丰富信息，充分利用这些信息构建算法模型，从而确定目标的位置。在多种类的信息中，因为室内环境的复杂特征，视觉信息相比其他信息具有更强的可辨识度。基于计算机视觉的室内定位技术与视觉信息关系密切，是本文研究的重点。在基于计算机视觉的定位中，相机是主要传感器，本文将扫地机器人作为研究对象，采用高分辨率的单目工业相机作为视觉图像的采集传感器，通过跟踪与定位结合的方式实现一个室内移动机器人的定位系统。本文的主要研究内容包含：

(1) 以个人计算机为平台实现一个面向室内移动机器人的跟踪定位系统，该系统使用单目工业相机完成图像的采集，借助特殊的标记物进行复杂特征的统一，并将目标跟踪与检测技术和位姿估计方法作为算法核心，在其中设计了跟踪器与检测器提取图像特征，通过位姿估计算法实现对扫地机器人的定位。系统分为四个模块，初始化模块实现对系统初始数据的获取，确保整个定位过程的结构化运作，并对外提供定位所需初始数据。特征检测模块实现对扫地机器人的跟踪与特征提取，关键特征的位置信息是定位算法的基本数据。机器人定位模块实现对相机位姿的离线估计和对扫地机器人位置的在线估计。可视化模块实现将连续时间的定位结果以路径的形式展现。

(2) 实现基于多台相机的定位优化方法，该方法针对一台相机定位扫地机器人时存在的异物遮挡、深度信息缺失问题，搭建多台相机的场景布局，同时设定基准坐标系统—多个与相机关联的坐标系。为满足多台相机同时定位的要求，方法中实现了多相机协同运作机制，以多线程的方式完成每台相机的定位任务，并设计了重跟踪算法，充分利用多台相机多个视角的优势保证了相机进行定位任务的稳定性。针对多台相机进行定位任务存在时间差的问题，方法中设计了一种时间同步算法，将经由不同时间、不同相机所得的定位结果进行整理融合，计算更加准确的机器人位置结果。

...

关键词：定位， 跟踪， 室内， 移动机器人， 多相机

ABSTRACT

The development of indoor positioning enables mobile robots working indoors to provide more intelligent services. Indoor positioning technology can pick a wealth of indoor information by a variety of sensors, and make full use of this information to build some algorithm models to determine the location of the target. Among various types of information, visual information is more recognizable than other information because of the complex features in indoor environment. The indoor positioning technology based on computer vision is closely related to visual information, which is the focus of this study. In computer vision-based positioning, the camera is regarded as the main sensor. In this paper, the sweeping robot is taken as the research object, and the high-resolution monocular industrial camera is used as the acquisition device of visual images. An positioning system of indoor mobile robot is realized by combining tracking and positioning. The main research content of this paper includes:

- (1) Implementing a tracking and positioning system for indoor mobile robots in a personal computer. The system uses the monocular industrial camera to complete image acquisition, unifies complex features with the help of special markers, and takes the technology of target tracking and detection and the method of pose estimation as the core of the algorithm. In this system, one tracker and one detector are designed to extract image features, and the position of the sweeping robot is realized by the pose estimation algorithm. The system is divided into four modules. The initialization module realizes the acquisition of the initial data of the system, ensures the structured operation of the entire positioning process and provides the initial data required for positioning externally. The feature detection module implements tracking and feature extraction of the sweeping robot and the position information of key features is the basic data for positioning. The robot positioning module realizes the offline estimation of the camera's pose and the online estimation of the sweeping robot's position. The visualization module realizes the display of continuous-time positioning results in the form of paths.
- (2) Implementing a positioning optimization method based on multiple cameras. This method solves problems of Foreign body occlusion and lack of depth information when the sweeping robot is being positioned by one camera. It builds a scene of multiple cameras, and sets a reference coordinate system to unify coordinate systems related to cameras. In order to meet the requirement of simultaneous positioning of multiple cameras, the method

implements a multi-camera cooperative operation mechanism. The mechanism completes the positioning task of each camera by multithreading. In this mechanism, a re-tracking algorithm is designed to make use of the advantages of multiple perspectives to ensure the stability of the camera's positioning task. In order to solve the problem of time difference in the positioning tasks of multiple cameras, a time synchronization algorithm is designed in the method. It collates and fuses the positioning results obtained by different times and different cameras to calculate more accurate results of robot position.

...

Keywords: XXX, XXX, XXX, XXX, XXX

插图索引

图 1.1 室外移动机器人	1
图 1.2 室内移动机器人	2
图 1.3 视觉相机	5
图 2.1 针孔相机模型	9
图 2.2 3D点投影形成示意图	10
图 2.3 目标跟踪实例	13
图 2.4 生成模型方法原理图	14
图 2.5 目标检测实例	15
图 2.6 传统目标检测方法的3个阶段	16
图 2.7 对极几何约束	17
图 3.1 移动机器人跟踪定位系统框架设计图	19
图 3.2 初始化模块关系图	19
图 3.3 本地数据文件结构	20
图 3.4 系统对象类结构图	21
图 3.5 特征检测模块关系图	22
图 3.6 跟踪器工作流程图	23
图 3.7 四月码图案	26
图 3.8 低通滤波前的图像	26
图 3.9 低通滤波后的图像	26
图 3.10 多线程执行线段检测方式	27
图 3.11 机器人定位模块流程图	29
图 3.12 以四月码标记物为基础的世界坐标系设置示意图	31
图 3.13 相机坐标系与世界坐标系位置关系示意图	32
图 3.14 坐标系之间转换关系示意图	33
图 3.15 可视化模块关系图	34
图 3.16 位姿数据封装结构体	35
图 3.17 位姿数据的发布与订阅结构图	35
图 3.18 机器人模型各环节与关节的关系图	36
图 3.19 机器人模型结构图	36
图 3.20 完整机器人模型	36
图 4.1 异物接近四月码的遮挡情况	39

图 4.2 异物接近相机镜头的遮挡情况	39
图 4.3 机器人大尺度变化移动示意图	40
图 4.4 图像中尺度过小的四月码	40
图 4.5 两台相机的布局图	42
图 4.6 四台相机的布局图	42
图 4.7 多相机场景中的基准坐标系设置	43
图 4.8 线程执行任务流程图	45
图 4.9 目标区域重构示意图	47
图 4.10 多线程定位任务的结果融合示意图	47
图 4.11 时间分段策略示意图	48
图 4.12 时间同步结构体组成示意图	48
图 4.13 图节点与边的定义	50
图 4.14 四台相机布局构建的图结构	51
图 5.1 海康威视工业相机	55
图 5.2 海康威视相机镜头	55
图 5.3 测试用三脚架	56
图 5.4 米家二代扫地机器人	56
图 5.5 棋盘格标定板	56
图 5.6 标记物实体	56
图 5.7 实验室大厅场景	57

表格索引

表 1.1 热点定位技术的比较 4

符号对照表

符号	符号名称
XXX	XXX
XXX	XXX
XXX	XXX

缩略语对照表

缩略语	英文全称	中文对照
GPS	Global Positioning System	全球定位系统
UWB	Ultra Wide Band	超宽带
TDOA	Time Difference of Arrival	到达时间差
SURF	Speeded Up Robust Features	加速的具有鲁棒性的特征
VO	Visual Odometry	视觉里程计
SLAM	Simultaneous Location and Mapping	同时定位与建图
ROS	Robot Operating System	机器人操作系统
IMU	Inertial Measurement Unit	惯性测量单位
KCF	Kernelized Correlation Filters	核相关滤波器
DSST	Discriminative Scale Space Tracker	可分辨尺度空间跟踪器

目录

摘要	I
ABSTRACT	III
插图索引	V
表格索引	VII
符号对照表	IX
缩略语对照表	XI
第一章 绪论	1
1.1 研究背景	1
1.2 国内外研究现状	2
1.3 研究内容	6
1.4 全文结构	7
第二章 相关技术及理论	9
2.1 坐标系转换	9
2.2 相机内参标定方法	11
2.3 目标跟踪与检测技术	13
2.3.1 目标跟踪技术	13
2.3.2 目标检测技术	15
2.4 位姿估计	16
2.4.1 常用位姿估计方法	17
2.5 本章小结	18
第三章 移动机器人跟踪定位系统设计	19
3.1 初始化模块	19
3.1.1 本地数据的管理及加载	20
3.1.2 系统对象类设计	21
3.2 特征检测模块	22
3.2.1 基于相关滤波的跟踪器设计	22
3.2.2 基于角点检测的检测器设计	25
3.3 机器人定位模块	29
3.3.1 基于3D-2D的位姿估计方法研究	29
3.3.2 相机外参标定方法	31
3.3.3 机器人位姿估计方法	32

3.4 可视化模块	34
3.4.1 位姿数据的封装与加载	35
3.4.2 机器人模型的制作与加载	36
3.5 本章小结	37
第四章 基于多相机的机器人定位优化方法设计	39
4.1 基于单相机的机器人定位局限性分析	39
4.2 基于多相机的机器人定位方法	41
4.2.1 定位场景的布局设置	41
4.2.2 坐标系统一	42
4.2.3 多相机协同运作机制	44
4.2.4 定位结果融合	47
4.3 基于多相机布局的精度优化策略	50
4.3.1 多相机布局的图构建	50
4.3.2 基于图的精度优化	51
4.4 本章小结	53
第五章 实验测试与分析评估	55
5.1 实验测试平台与测试环境说明	55
5.1.1 PC端测试平台	55
5.1.2 辅助测试工具说明	55
5.1.3 实际测试环境搭建	57
5.1.4 实验测试评估指标	57
5.2 基于多相机的定位系统功能测试与分析	58
5.2.1	58
5.2.2	58
5.3	58
5.3.1	58
5.3.2	58
5.4 本章小结	58
第六章 总结与展望	59
6.1 本文总结	59
6.2 未来展望	60
致谢	63
作者简介	65

第一章 绪论

1.1 研究背景

二十世纪中叶，随着工业技术、控制理论、计算机技术的快速发展，机器人技术开始萌芽^[2]。在二十世纪末，机器人不仅更多地被应用于工业领域，也进入了人们的生活，改善了人们的生活水平，与机器人有关的技术因此得到了大范围的应用。机器人提高了社会生产力，能够代替人类完成危险的工作，大大降低了人类遭遇危机的概率。在机器人中具有移动能力的类型被称为移动机器人，它们不同于固定位置工作的机械臂，在一定程度上对人类的移动能力进行模拟和扩展。

移动机器人在机器人中所占比重巨大，它凭借在空间中移动的能力，有效地扩大了机器人的工作范围和空间。常见的移动机器人有无人机、自动驾驶汽车、搬运机器人、扫地机器人等，其中无人机和自动驾驶汽车因为它们广泛的应用领域与不菲的商业价值而逐渐被商用化，搬运机器人则更多地被运用于工业场景中，协助货物的运输，扫地机器人作为消费级的机器人，在家庭中普及，在服务业大展身手。根据工作环境的不同，移动机器人可分为室外移动机器人和室内移动机器人。



(a) 无人机

(b) 自动驾驶汽车

(c) 无人地面车

图 1.1 室外移动机器人

无人机、自动驾驶汽车、无人地面车多工作在室外场景下，属于室外移动机器人。图1.1(a)中的大疆无人机 Inspire 2 是典型的室外移动机器人，它配备有前视和下视的双目视觉系统，能够对 30 米范围内的障碍物进行检测，同时在顶部配置红外感知系统，能有效感知上方 5 米范围内的物体。图1.1(b)中的特斯拉自动驾驶汽车 Tesla Model X 环绕车身配置 8 个相机，视野范围达 360 度，对周围环境的监测距离最远可达 250 米，传感器的配置弥补了视觉系统的不足，在保证监测范围的基础上增加了对软硬物体的探测。图1.1(c)为 CLEARPATH 研发的无人地面车 JACKAL，车外壳完全防水，高扭矩传动系统保证了足够的牵引力，能够在复杂的室外地形中无障碍地移动。



图 1.2 室内移动机器人

仿生服务机器人、仓储机器人、扫地机器人的工作场景多为室内，是典型的室内移动机器人。法国 Aldebaran Robotics 公司开发的人形寓教机器人 NAO，如图1.2(a)，具有类人的四肢，能够进行多种形式的表现，完成对儿童、老人的陪护，同时其高可动性被用于避障、抓取等研究。图1.2(b)为亚马逊的 kiva 仓储机器人，它只有 145 公斤的重量，却能搬起远超自身重量的货物，且具备稳定的导航系统，对商品的扫描准确率高达 99.99%。图1.2(c)中的米家扫地机器人是一种自主移动机器人，配置有多个传感器，能够在室内进行地图的实时绘制，从而规划路线完成智能化的清扫，如今已普遍应用于家庭中。

移动机器人完成相应任务的大前提是知晓自身的位置，即定位。对比室外移动机器人和室内移动机器人，前者的工作场景不是特别复杂，且如今已存在比较成熟的 GPS 定位技术表达机器人与环境的位置关系，机器人容易理解环境，而后者的工作环境更加繁杂，且无法运用 GPS 进行定位，但是室内环境的视觉信息比较丰富，环境特征更有标志性，通过计算机视觉对室内移动机器人进行精准定位具有更好的研究前景。

1.2 国内外研究现状

室外场景下的定位技术依赖于固定传感器或是基于 GPS 的传感器^[? ?]，而且 GPS 因其能提供最大的覆盖面积而成为使用最广泛的基于卫星的定位系统^[?]。但是以下问题的存在使得室外定位技术应用于室内场景存在较大的局限性：(1) 室内环境相对室外环境变得更加复杂；(2) 室内存在较强的信号干扰和反射，依靠信号进行的定位误差将会很大；(3) 室内定位对室内环境的依赖性极强；(4) 室内通信链路变得不可靠^[?]。

为解决上述问题，诸多室内定位技术被提出，如 WIFI 定位、超声波定位、射频识别^[?]定位、超宽带（Ultra Wide Band, UWB）定位、计算机视觉定位等，研究人员利用这些定位技术构建定位系统以达到定位的精度需求。

如今 WIFI 在室内应用广泛，且覆盖面大， WIFI 定位成为目前室内定位领域

的研究热点。借助接入点（Access Point, AP），Kotaru 等人在2015年提出 Spotfi 系统^[?]，达到了 40 厘米的较高精度，但是该系统因为 WIFI 信号的数量限制无法被用于实时场景。2016 年，Wooseong Kim 和 Sungwon Yang 提出基于 WIFI 定位的系统 Freloc^[?]，它通过收集 WIFI 位置信息自动进行地图构建，从而达到实时定位的目的，然而该系统的定位精度却远不如 Spotfi。

在超声波定位领域，定位技术更加成熟，在定位精度方面表现优异。早在 1997 年，Andrew Ward 等人提出了 BAT 系统^[?]，利用超声波信号进行室内定位，Harle Robert 与 Hopper Andy 对该系统进行了全面的实验评估工作^[?]，精度高达 4 厘米，然而定位精度过分依赖于传感器的摆放位置且系统的额外设备开销较大，使得该系统虽然具有先进性却没有应用于大部分的场景。在射频识别定位领域，Lionel M Ni 等人在 2003 年提出 LANDMARC 系统^[?]，是具有开拓性的一套定位系统，它要求被定位对象表面依附射频识别标记，通过标记检测器获取标记信号强度的信息并结合 k 最近邻算法计算标记的位置，定位结果的精度在 1 米左右。

为追求更高的定位精度，部分研究人员提出使用 UWB 技术进行定位。Krishnan 等人在 2007 年为机器人室内定位导航设计了一套基于 UWB 红外的定位系统^[?]，该系统设定 UWB 监测器的位置，将 UWB 发射器安装在机器人上，通过到达时间差（Time Difference of Arrival, TDOA）方法实现对机器人的室内定位和导航，它以 15 厘米的定位误差体现了 UWB 定位的优异性，但是其高昂的成本、额外的硬件设备要求以及不够大的定位范围限制了其被广泛使用。

大部分室内定位技术为了提高定位精度而牺牲了成本，使得精度较高的定位技术使用有所局限，一些研究人员从计算机视觉的角度出发，试图寻找定位技术精度与成本的制衡点。在 2011 年，Werner Martin 等人提出了 MoVIPS 系统^[?]，该系统分为标定与定位两部分，标定部分通过相机拍摄的图像创建特征数据库，定位部分中相机拍摄的一张图像会传输到服务器中，服务器使用加速的具有鲁棒性的特征（Speeded Up Robust Features, SURF）^[?]算法提取特征并与特征数据库进行匹配，这两部分结合完成了对物体的定位，定位的误差在 1.3 米左右。在 2009 年，Mulloni Alessandro 等人提出了基于条形标记物识别的定位系统 SignPost^[?]，它能够识别相机视野中的条形标记物并使用透视形变的方法计算物体与标记的相对位置，在条形标记物的帮助下，系统的定位精度能到达厘米级。微软研究院的 EasyLiving 系统^[?]于 2000 年被提出，这个较早提出的定位系统在平面的定位精度能够达到 10 cm，而该系统也成为了微软 Kinect^[?]生产线的先驱者。

表1.1中列出了上述提到的各个热点定位技术的优缺点对比。在规避或减弱环境光照变化带来的负面影响后，计算机视觉定位技术脱颖而出，以低成本高精度的特点被广泛采用。当定位的对象被限定为移动机器人时，许多室内定位技术便显得不

表 1.1 热点定位技术的比较

定位技术	优点	缺点
WiFi	使用成本较低，同时拥有较好的定位精度，覆盖面广，有很好的应用前景	依赖于AP，AP的变更对定位影响很大
超声波	精度很高，技术成熟	需要大量额外的设备，造成应用成本较高，且对设备的摆放位置有很高的要求，容易受到干扰
射频识别	通讯范围大，成本较低，非视距，非接触	定位有效范围较小，在没有辅助技术的情况下精度很低
UWB	定位精度极高，功耗低，抗干扰能力强	极高的应用成本，定位有效范围有限
计算机视觉	成本低，定位精度有时能到达厘米级，充分利用了室内环境的视觉信息	由于定位的基础数据来源于环境，对环境光照变化比较敏感，使得定位精度的波动较大

再实用，毕竟移动机器人可安装的传感器有限，成本开销也需要控制，因此计算机视觉定位成为了近几年室内定位的研究热门。

计算机视觉定位技术大多以相机为传感器做定位数据的获取，根据相机安装方式的不同，计算机视觉定位技术被分为两类：(1)“被动”方式，相机被设置在移动机器人上，移动机器人在环境中移动并通过相机去感知环境，最后对比数据库中的定位信息形成定位结果；(2)“主动”方式，一个或多个相机被固定安装在环境中，监视出现在视野中的移动机器人，借助机器人或标记物的特征对机器人进行定位^[?]。

视觉里程计（Visual Odometry，VO）^[?]是典型的“被动”方式，能够根据有序的图像序列实时计算机器人的位置。Kitt Bernd 等人基于 VO 提出使用双目相机采集图片序列并结合滤波技术的方法^[?]，实现以更低的计算开销获得更好的定位精度。融合了 VO，同时定位与建图（Simultaneous Location and Mapping，SLAM）技术^[?]获得了更好的实时性，视觉 SLAM 在机器人定位导航领域起到了巨大的作用。基于特征检测的 ORB-SLAM^[?]、基于直接法的 LSD-SLAM^[?]以及近几年刚被提出基于前两者优秀策略的 PL-SLAM^[?]在视觉定位方面都有很好的表现。微软公司的 EasyLiving 系统则是属于“主动”方式，由于这种“主动”方式依赖于机器人的特征，更多的研究工作便致力于机器人特征的提取以及机器人特征的统一。视觉标记是机器人特征统一很好的方式，标记依附在机器人表面，以特定规则呈现特征，使

得定位更加简单，比如 apriltag 标记^[? ?]。

在定位的过程中，大部分的低成本机器人本身不具有很强的计算能力，通常需要借助主机对采集的视觉信息进行处理，因此机器人与主机之间存在频繁的信息交互。为了提高交互的效率以及视觉定位技术的开发效率，机器人操作系统（Robot Operating System, ROS）^[?]于 2007 年被提出并在 2010 年正式开源。它不同于底层的操作系统，是一种次级操作系统，为开发者提供稳定的工具程序和库，简化开发的过程。ROS 不仅支持点对点设计，还兼容多种编程语言并且易于模块化，以丰富的工具包、无索取开源的特点成为机器人研究领域的开发利器。

相机作为计算机视觉定位技术的主要传感器，发挥着采集环境图像的巨大作用。根据相机的硬件配置类型与功能，视觉定位中的相机被分为单目相机、双目相机和深度相机，如图1.3所示。单目相机比较普遍，具有标定简单、成本低以及尺度不确定性的特点；双目相机如今被更多地应用到计算机视觉定位中，具有观测距离远、标定复杂的特点；深度相机相对前两者能够更加轻易地获取图像中的深度信息，但是成本高和稳定性差是它的缺陷。



图 1.3 视觉相机

国内对于室内机器人视觉定位技术的研究开始得比较晚，更多的是基于国外先进技术的改进与优化，缺少突出的理论成果，但仍有许多研究机构和公司已经形成了一些成果。上海交通大学自动化研究所提出一种利用天花板特征点对机器人进行视觉定位的方法^[?]，此后中科院自动化研究所同样利用天花板特征点提出基于直线和单特征点的视觉推算定位方法^[?]。香港科技大学基于视觉 SLAM，提出通过滑动

窗口优化完成视觉与惯性测量单位（Inertial Measurement Unit, IMU）的有效融合，进一步提高了“被动”方式定位的精度和稳定性^[?1]。

1.3 研究内容

近年来，自主移动机器人更多地被应用于室内场景，在各个方面为人类提供更为全面的服务，这得益于移动机器人日渐完善的自主性。机器人实现自主性的首要任务是对周围环境的感知，从而理解自身所处的位置，为下一步自主性行为提供准确的数据支持。室内定位技术能够协助机器人识别环境，研究针对移动机器人的室内定位技术对机器人的自主性形成有极其重要的意义。

本文围绕移动机器人的室内定位技术，提出如下研究内容：

(1) 将具有服务性质并且作为室内移动机器人典型的扫地机器人作为研究载体，依托于石头科技公司的硬件嵌入式扫地机器人，以位姿估计算法为核心，借助单目工业相机作为外部传感器，在PC平台上搭建一个室内移动机器人跟踪定位系统，利用标记物进行特征统一，设计跟踪器与检测器提取图像特征，实现针对扫地机器人的室内小规模定位。系统由四个模块组成，初始化模块获取系统初始数据，结构化整个定位过程的运作，同时也是其他模块的数据来源；特征检测模块对目标区域进行跟踪并提取区域中的关键角点特征，将角点特征的位置信息作为定位的基础数据供给其他模块；机器人定位模块执行对相机与扫地机器人的位姿估计；可视化模块将连续时间的定位结果以路径的形式更加直观地进行呈现。

(2) 针对一台相机定位扫地机器人时存在的异物遮挡、深度信息缺失等问题，本文提出基于多台相机的定位优化方法。该方法通过多台相机布局的场景搭建，弥补对扫地机器人移动中深度变化的感知，同时借助标记物设定基准坐标系达到坐标系统一的目的。此外，为了满足多台相机同时定位的要求本文设计了多相机协同运作机制，以多线程的方式执行从采集图像到完成定位的一系列任务，考虑多台相机参与机器人定位带来的协同性，提出了用于保持跟踪器对机器人跟踪的重跟踪算法。针对多台相机采集图像存在时间差的问题，本文设计了一种时间同步算法，该算法对复杂繁多的定位结果进行时间统一，在整理融合之后给出多相机场景下优化后的定位结果。

(3) 以多台相机参与定位的布局场景为基础，本文提出了围绕场景中相机与待优化定位点进行图构建的图优化方法，在为图中元素赋予定义后，引入重投影误差，通过分析图中一条边的误差优化问题，形成以全图为基础的整体优化问题，最终线性求解该优化问题得到对定位点坐标的修正坐标，从而降低多相机场景中积累的噪声误差。

1.4 全文结构

本文的结构安排如下：

第一章首先介绍了机器人的发展情况，同时列举了几种具有代表性的室内移动机器人和室外移动机器人，对比其工作环境，引出研究室内定位技术的重要意义，然后对室内定位技术的发展历史以及关键节点进行梳理，围绕室内移动机器人对计算机视觉定位领域进行了着重说明，最后介绍本文的研究对象并阐述了相关研究内容。

第二章首先描述了计算机视觉系统中四大坐标系之间的转换过程，引出转换过程中起关键作用的相机外参与内参，然后阐述了常用的相机内参标定方法原理，接着描述了在计算机视觉定位领域中具有重要作用的目标跟踪与检测技术，对它们的作用原理和技术分类进行了简单的说明。最后围绕计算机视觉定位领域中的关键方法——位姿估计方法，阐述了 2D-2D、3D-3D、2D-3D 三种情况下的位姿估计方法原理。

第三章描述了室内移动机器人跟踪定位系统的模块化设计，分别对初始化模块、特征检测模块、机器人定位模块和可视化模块这四个模块展开详细说明。初始化模块加载相机内参等本地数据并为其他模块提供计算数据，特征检测模块使用跟踪器与检测器对相机采集的图像中的关键特征角点进行提取，机器人定位模块结合角点特征的位置信息与相机内参数数据对相机与机器人的位姿进行估计，可视化模块直观呈现连续时间内的定位结果。

第四章首先分析了单相机场景下进行室内定位移动机器人的局限性，然后围绕这些局限性，描述了基于多相机的定位优化方法，其中突出了多台相机的布局方案、坐标系统一的策略、多相机协同运作的机制以及以时间同步为目的的定位结果融合方法这四个方面，其中在多相机协同运作机制的设计中考虑各台相机之间的信息交互，阐述了增加系统稳定性的重跟踪算法。最后详细说明了以多相机场景为基础构建图的过程以及围绕已构建的图对 3D 定位点进行优化的方法。

第五章首先介绍实验平台的构建，简要说明评价指标和软硬件实验平台，然后设计三个实验进行结果对比，分别是采用一个相机的定位实验、多相机参与的定位实验和融合图优化的定位实验。最后将三组结果数据与 ground truth 进行对比分析，得出结论。

第六章对全文进行概括性的总结，根据室内定位技术的发展前景以及当前的科学发展趋势提出未来的研究计划。

第二章 相关技术及理论

计算机视觉定位技术将相机作为传感器获取图像，并采用目标跟踪与检测算法提取图像中的特征信息，最后利用特征信息对机器人进行位姿估计。

本章分为四个部分：第一个部分描述针孔相机模型中主要坐标系之间的转换过程；第二个部分介绍相机内参标定方法，说明典型方法的具体原理和优缺点；第三个部分介绍目标跟踪与目标检测技术，描述主流方法的相关过程和特点；第四个部分对常见的位姿估计方法进行分析。

2.1 坐标系转换

在计算机视觉中，针孔相机模型是一个常用的几何模型，它通过小孔成像的原理描述了三维世界中坐标点映射到二维图像平面的过程。针孔相机模型中的主要坐标系包括世界坐标系、相机坐标系、图像坐标系和像素坐标系。

世界坐标系通常被定义为模型中的基准坐标系，以该坐标系为基础，物体在真实世界中的位置得以被数据化，在相机标定中世界坐标系通常被用于描述相机的实际位置。相机坐标系以相机的光心为坐标原点，其 xy 平面与成像平面平行，将空间中物体的位置以相机的角度进行表示。图像坐标系和像素坐标系描述了针孔相机模型中二维部分，其中图像坐标系以图像主点为坐标原点，形成的平面与成像平面重合，它描述了成像过程中物体从相机坐标系到图像坐标系的投影透视关系，而像素坐标系的坐标原点位于图像最左上角的像素点，描述了物体成像后的像素点在图像中的位置。上述坐标系的关系如图2.1所示，紫色坐标系是世界坐标系 O_w -x-y-z，绿色坐标系是相机坐标系 O_c -x-y-z，灰色坐标系是图像坐标系 O_i -x-y-z，蓝色坐标系是像素坐标系 O_p -x-y-z。

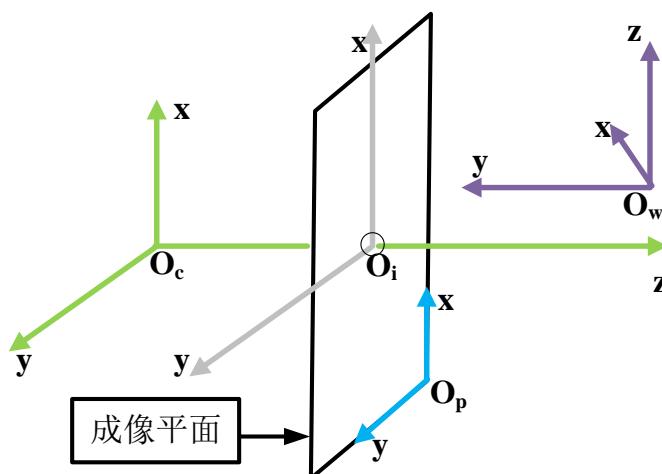


图 2.1 针孔相机模型

色坐标系是相机坐标系 O_c -x-y-z，灰色坐标系是图像坐标系 O_i -x-y，蓝色坐标系是像素坐标系 O_p -x-y。

世界坐标系与相机坐标系之间的转换是一种刚体变换，描述了相机在三维空间中的位置以及拍摄的角度。旋转与平移作用于世界坐标系中的三维坐标点 $P_w(X_w, Y_w, Z_w)$ ，将其转换为该点在相机坐标系下的坐标 $P_c(X_c, Y_c, Z_c)$ ，转换过程如公式2-1所示。

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = R \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + t \quad (2-1)$$

公式2-1中以齐次坐标的方式进行矩阵相乘， R 表示 3×3 旋转矩阵， t 表示 3×1 平移矩阵。它们的组合代表了相机的外参，描述了相对于世界坐标系，相机在三维空间中的位姿信息。

相机坐标系与图像坐标系之间的转换是一种投影透视的过程。相机坐标系的 xy 平面与成像平面之间的物理距离被称为焦距 f ，z 轴垂直成像平面，穿过图像主点，而相机坐标系下的三维点 P_c 与坐标原点的连线与图像平面相交，形成一个投影点 p ，如图2.2所示，该投影点的坐标是三维点在图像坐标系下的坐标 $p(x, y)$ 。图2.2中

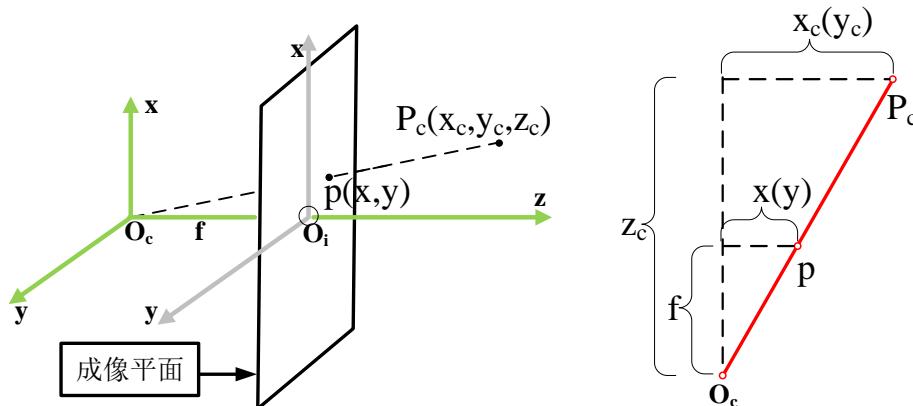


图 2.2 3D 点投影形成示意图

构成两组相似三角形关系，利用该关系构建等式，整理得：

$$Z_c \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \quad (2-2)$$

这个过程涉及了相机的焦距 f ，将三维的坐标转化为二维，是空间信息到平面信息的转变。

图像坐标系与像素坐标系之间的转换是缩放与平移的过程。图像坐标系通过物理单位描述投影点在图像中的位置，而像素坐标系则以像素数目为单位表示投影点

在图像中的位置。因此，每个像素点在图像平面 x 方向上的物理尺寸 dx 以及 y 方向上的物理尺寸 dy 是实现两个坐标系转换的关键，而这两者可通过相机内参标定获取的。公式2-3描述了二维坐标缩放与平移的过程。

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & c_x \\ 0 & \frac{1}{dy} & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (2-3)$$

其中 u 和 v 表示投影点在像素坐标系下的二维坐标，而 c_x 和 c_y 表示平移部分的标量，共同构成了图像的主点坐标。这个转换过程涉及了相机的像元尺寸 dx 与 dy 以及主点坐标 c_x 与 c_y ，它们分别表示了坐标系转换中的缩放部分与平移部分，是相机内参的一部分，实现了从物理尺寸到像素数目的转换。

结合公式2-1、公式2-2和公式2-3，可以得到一个比较明确的转换关系，如公式2-4所示：

$$Z_c \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} [R \ t] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2-4)$$

其中 f_x 等于 $\frac{f}{dx}$ ， f_y 等于 $\frac{f}{dy}$ 。由公式2-4可见，世界坐标系下的坐标经过相机内参矩阵 $\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$ 和相机外参矩阵 $[R \ t]$ 的作用转化为像素坐标系下的坐标。

2.2 相机内参标定方法

相机是计算机视觉定位中的重要传感器，针孔相机模型的构建对 3D 点的投影过程进行了清晰的解释。在针孔相机模型中涉及了一些相机本身的设备参数，如相机焦距 f ，相机的像元尺寸 dx 与 dy 以及成像平面的主点(c_x, c_y)，这些参数也被称为相机内参，在三维坐标点投影成为二维像素点的过程中有着重要的作用。相机内参标定方法的主要目的是对这些设备参数的估计求解，参数精度影响着投影后二维像素点的位置，因此相机内参标定方法围绕对相机内参的精度优化不断发展至今。

相机标定方法根据其标定的方式分为主动视觉相机标定法、相机自标定法和传统相机标定法。主动视觉相机标定法求解内参的过程并不复杂，多次求解所得的内参结果异常值较少，此外它无需标记物辅助标定，但是成本却是三类方法中最高的，常用的方法是使用定制系统控制相机做规定的运动进行标定。相机自标定法虽然有很强的灵活性，但是其在精度和稳定性方面的表现较差，比如分层逐步标定^[?]和基于 Kruppa 的自标定法^[?]。传统相机标定法依赖于棋盘、点阵等标记物，但是其泛用性高、精度可观、使用门槛低，使用较多的方法有 Tsai 两步法^[? ?]和张正友标定法^[?]，其中张正友标定法因其简单的辅助标定物和高精度被广泛使用。

张正友标定法是一种基于平面棋盘格的标定方法，它由以下几个部分组成：

(1) 两个平面之间的单应性。张氏标定法中涉及两个平面：棋盘格平面和图像平面。它们之间的投影映射关系由单应矩阵 H 表示，即棋盘格平面上 3D 点坐标在单应矩阵的作用下转化为图像平面上的 2D 点坐标，如公式2-5所示。

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = H \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2-5)$$

其中 s 表示尺度，与公式2-4中的 Z_c 意义相同。若将外参矩阵 $[R \ t]$ 按照列向量的形式写成 $[r_1 \ r_2 \ r_3 \ t]$ ，并使世界坐标系的 xy 平面与棋盘格标记物的平面重叠，那么对于棋盘格上的 3D 点，它们的 Z_w 全都为 0，因此可将 r_3 这列消去，结合公式2-4得到公式2-6：

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K [r_1 \ r_2 \ t] \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix} \quad (2-6)$$

于是3*3单应矩阵可表示为未知相机内参矩阵与简化相机外参矩阵的乘积，如公式2-7所示。

$$H = [h_1 \ h_2 \ h_3] = K [r_1 \ r_2 \ t] \quad (2-7)$$

其中 $[h_1 \ h_2 \ h_3]$ 是以列向量形式表示的单应矩阵。单应矩阵有 8 个未知量需要求解，利用已知的至少四对 3D 点与 2D 点的坐标可以求解出单应矩阵。

(2) 利用约束条件求解内参矩阵 K 。单应矩阵由内参矩阵与外参矩阵相乘得到，而内参矩阵与外参矩阵均为未知，无法正常求解。此时外参矩阵中旋转量 R 的约束条件成为求解内参的突破口， R 是一个正交矩阵，其列向量相互正交且模均为 1，根据约束条件将外参矩阵消去，得到如下等式：

$$\begin{aligned} h_1^T K^{-T} K^{-1} h_2 &= 0 \\ h_1^T K^{-T} K^{-1} h_1 &= h_2^T K^{-T} K^{-1} h_2 \end{aligned} \quad (2-8)$$

在内参矩阵中存在 5 个待求解的未知量，这 5 个未知量能够唯一求解的条件是至少已知三个单应矩阵。令矩阵 B 表示 $K^{-T} K^{-1}$ ，3*3矩阵 B 是一个对称矩阵，原本 B 中的 9 个未知量减少为 6 个，于是将 B 写成六维向量 b 的形式，如公式2-9所示：

$$B = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} \quad b = [B_{11} \ B_{12} \ B_{22} \ B_{13} \ B_{23} \ B_{33}]^T \quad (2-9)$$

结合公式2-8得到一个简化的等式：

$$Vb = 0 \quad (2-10)$$

式中 V 是一个 $2n*6$ 的已知矩阵，与单应矩阵相关。最后将三个单应矩阵结合组成 V ，通过奇异值分解求解 b ，从而得到唯一的内参矩阵 K 。

(3) 标定精度优化。标定方法中常采用最小化重投影误差的方法对内参进行优化，即将空间坐标按照当前估计的外参与内参投影到像素平面，根据图像检测算法检测得到的像素点和重投影的像素点之间的像素偏差构造代价函数，通过最小化代价函数进一步优化这些估计的参数，如公式2-11所示。

$$\sum_{i=1}^n \sum_{j=1}^m \|p_{ij} - \hat{p}(K, R_i, t_i, (P_w)_j)\|^2 \quad (2-11)$$

式中 i 与 j 分别表示相机位置的编号与一个相机位置中 3D 点的编号， p_{ij} 表示在第 i 号的相机位置上第 j 个 3D 点原本在图像中检测的像素位置， \hat{p} 表示该 3D 点重投影后的像素位置。这个优化问题其实是一个最小二乘问题，常使用列文伯格-马夸尔特方法迭代求解。

张正友标定法解决了传统标定法原本操作繁琐、标定物精度要求高的问题，同时相对自标定大大提升了标定的精度，因此成为了主流的相机内参标定方法。

2.3 目标跟踪与检测技术

在计算机视觉定位领域，图像是唯一的视觉信息来源，目标跟踪与检测技术能够将图像中的冗余信息过滤，提取关键特征信息。本节对目标跟踪技术与目标检测技术进行介绍。

2.3.1 目标跟踪技术

目标跟踪通常指单目标跟踪，在第一帧图像中获取输入的矩形框指定跟踪目标，并在后续的每一帧中找到目标的运动位置，如图2.3所示。定位技术借助目标跟踪方

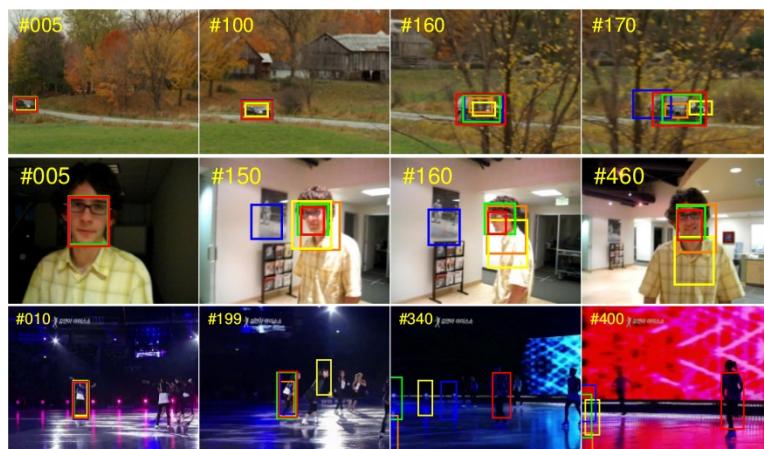


图 2.3 目标跟踪实例

法缩小图像处理的范围，提高定位效率与精度。目标跟踪的重点在于对目标物体的动态感知，包括目标的形状、移动速度等，因此目标跟踪过程中目标的形变、运动速度过快、模糊等问题极大地影响了跟踪的效果。此外环境光照的变化、背景的干扰等环境因素降低了跟踪的精度。为应对上述问题，诸多目标跟踪方法被提出，它们从实现方式上的不同被分为生成模型方法和判别模型方法。

生成模型方法能够提取图像中的目标特征并根据这些特征生成目标的外观模型，方法中把该模型当作匹配目标，在图像中搜索与其最匹配的区域，将这个区域作为跟踪结果，从而达到跟踪的目的，其运作原理如图2.4所示。均值漂移算法^[?]、卡尔

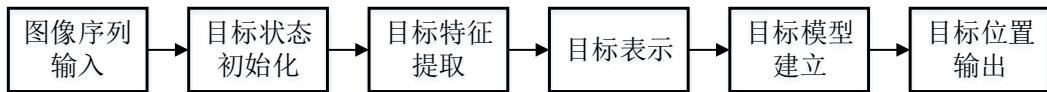


图 2.4 生成模型方法原理图

曼滤波算法^[?]和粒子滤波^[?]是生成模型方法中常用的算法，主要用于对目标区域的加速匹配，围绕这些算法，L1 Tracker^[?]、L1 APG^[?]和增量视觉跟踪方法^[?]等具有代表性的生成模型方法被提出。生成模型方法对于视角变化、目标被部分遮挡、目标尺度变化等情况有较好的处理效果，但是它无法充分利用目标的背景信息，因此在跟踪准确率的对比中判别模型方法更胜一筹。

判别模型方法将目标跟踪抽象为一个二分类问题，分类器的训练数据来自图像中目标与背景的特征信息。训练完成的分类器将目标从背景中分离出来，实现目标的跟踪。

基于相关滤波^[?]的目标跟踪方法是判别模型方法中的一条分支，它将第一帧图像作为训练样本，训练学习一个滤波器，滤波器与下一帧中的各个搜索窗口进行相关运算，通过相关值确定下一帧中目标的最佳位置。最初相关滤波在目标跟踪中的应用先进性在于把整个跟踪问题建立在频域中，通过快速傅里叶变换加速相关值的计算，MOSSE^[?]方法体现了这一先进性。面对大量的样本数据，稀疏采样策略会使样本之间存在很高的重叠率，产生的冗余数据限制了跟踪算法的性能，针对这一点，循环矩阵被应用于相关滤波之中，一个数据矩阵由数千个样本平移而成时，该矩阵呈现环状，如下所示：

$$\begin{bmatrix} x_1 & x_2 & x_3 & \cdots & x_n \\ x_n & x_1 & x_2 & \cdots & x_{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ x_2 & x_3 & x_4 & \cdots & x_1 \end{bmatrix} \quad (2-12)$$

循环矩阵具有良好的性质，能够通过傅里叶变换进行对角化从而减少计算量，CSK^[?]充分利用了这一性质，将跟踪算法性能提升到数百帧的运行速度。之后基于相

关滤波的目标跟踪方法又融合了多通道特征发展出核相关滤波（Kernelized Correlation Filters, KCF）^[2]跟踪器和可分辨尺度空间跟踪器（Discriminative Scale Space Tracker, DSST）^[2]。基于相关滤波器的目标跟踪方法在跟踪速度方面表现优异，通过在线学习保证了跟踪过程的实时性，如今已大规模地应用于工业中。

基于深度学习的目标跟踪方法是判别模型方法中的另一条分支，它将目标区域标记为正样本，背景区域为负样本，经过学习之后的分类器在每一帧图像中区分正负样本，鉴别出目标区域，基于卷积神经网络的 UPDT^[2]方法是该领域的典型。在目标跟踪的应用中，深度学习更好地提取了目标的特征，通过低层特征对目标进行精准的定位，通过高层特征中的语义信息处理目标形变、跟踪漂移等情况。深度学习过度依赖于网络的训练以及速度，大量的标注训练数据的提供虽然能够使深度模型的训练更加容易，但是目标跟踪问题缺少大量训练数据，仅仅提供第一帧的 bounding box，这使得模型的训练变得十分困难。DLT（Deep Learning Tracker）^[2]和 SO-DLT^[2]方法通过辅助图片数据预训练深度模型，并在在线跟踪时微调模型，克服了目标跟踪中训练数据缺失的难点。

2.3.2 目标检测技术

目标检测作为计算机视觉领域一个重要的研究方向，其主要目的是在一张图像中寻找目标物体，确定这些物体的位置与大小，甚至通过模型的训练确定物体的类别，如图2.5所示。目标物体在图像中的表现具有随机性，无论是目标的位置、尺度

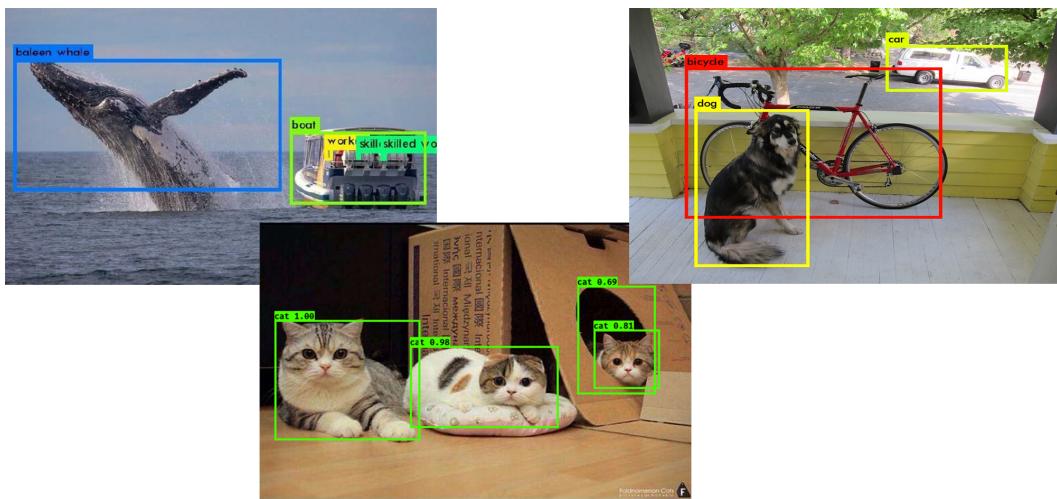


图 2.5 目标检测实例

还是形状，在图像中都是随机的，如何处理随机性成为目标检测中的核心问题。在视觉定位问题中借助目标检测技术获取图像中关键特征信息的位置，特征的位置信息能够辅助进行定位。然而目标检测同样面临着目标跟踪中的部分难点，目标物体的随机性、环境因素的干扰都对目标检测的准确度造成影响，这使得目标检测成为

机器视觉领域最具有挑战性的问题。目标检测根据实现方式被分为传统方法与深度学习方法，近几年基于深度学习的目标检测方法研究较多，传统方法则逐渐变少。

传统目标检测方法通常分为三部分，如图2.6所示，首先设置不同尺度、不同长

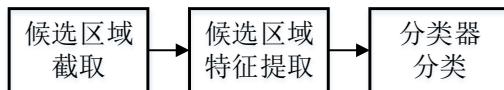


图 2.6 传统目标检测方法的3个阶段

宽比的窗口，利用滑动窗口策略对整幅图像进行遍历，通过大小不同的窗口截取图像的部分区域，然后在截取出来的图像区域中进行特征提取，进一步确定目标的相关特征，最后借助训练完毕的分类器对每一个区域中的物体进行分类。滑动窗口策略以一种穷举的思路进行区域选择，这使得目标检测的时间复杂度大幅上升，无法满足实时的要求。对目标区域的特征提取以手工特征的方式进行，比如方向梯度直方图（Histogram of Gradient, HOG）^[?]特征、SIFT^[?]特征、SURF 特征等，这种方式虽然使得传统目标检测方法在特定场景和较少类别的情况下取得比较好的效果，但是特征提取的稳定性却比较差。因此当光照等环境因素发生变化时，传统目标检测方法会出现大量的漏检与误检，效果呈现较差，而且检测速度也成为了传统目标检测方法的短板，深度学习的出现解决了这些问题。

基于深度学习的目标检测方法借助深度学习对大脑认知的模拟行为实现对大量数据特征的分析处理，具有突出的目标检测能力，成为了当前目标检测的主流方向^[?]。基于深度学习的目标检测方法根据检测模型的不同分为以 R-CNN^[?]为代表的围绕 Region Proposal 展开的二阶段目标检测算法与以 YOLO^[?]为代表的基于回归问题的一阶段目标检测算法。前者首先通过算法生成样本的候选框，然后使用卷积神经网络对样本进行分类，最后通过优化对检测框的位置进行调整，它在检测精度上占据优势，后者则不需要候选框的生成，而是将目标检测转化为回归问题，通过对目标物体进行回归获取目标的位置，它的检测精度虽然不及前者，但是由于放弃了候选框的生成，使得它的检测速度更快。

2.4 位姿估计

在移动机器人定位问题中，位姿估计是最核心的部分，相机标定与目标检测均服务于位姿估计环节。对移动机器人进行定位的目的是求解移动机器人在空间中的位置与方向角，而位姿估计方法能够结合相机采集到图像中的视觉信息与相机本身的内外参，推算估计出目标的位姿，即位置与方向。

2.4.1 常用位姿估计方法

在 VO 中，被用于对相机进行位姿估计的方法分为三种，它们分别是 2D-2D 方法、3D-2D 方法、3D-3D 方法。三种方法所需要的视觉信息各不相同，各自应用的场景也各不相同，但最终它们都能估计得到相机的位姿。

2D-2D 方法指位姿估计所需的数据来源由多组对应的 2D 点组成，这个方法应用于相机存在位移与旋转，能够以多个角度对一个场景进行拍摄的情况。一对 2D 点的组合由一个 3D 点投影在两个不同位置的相机成像平面中产生，因此在理想状态下一组匹配点与各自光心的连线最终交汇于被投影的 3D 点，产生了对极几何约束，如图2.7所示，图中 O_1 、 O_2 、 P 三点共同确定了一个平面，该平面被称为极平面，

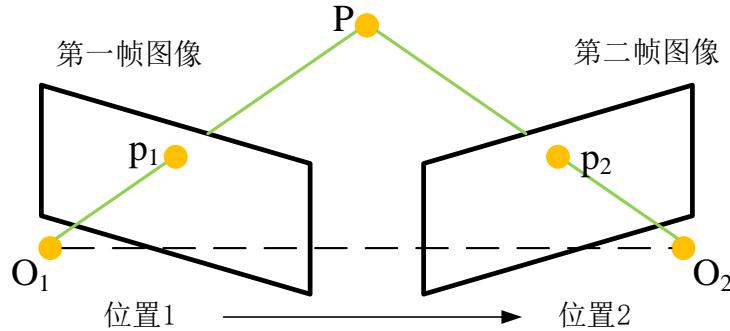


图 2.7 对极几何约束

O_1 与 O_2 的连线被称为基线。假设第一帧时的相机位姿已知，那么第二帧的相机位姿可通过求解相机第一帧到第二帧的位姿转换进行估计，于是有如下公式表示对极约束：

$$p_2^T K^{-T} t^\wedge R K^{-1} p_1 = 0 \quad (2-13)$$

式中 p_1 与 p_2 表示两个匹配点的像素位置， K 表示相机内参， R 与 t 表示待求解的位姿转换关系。公式2-13是对对极约束的描述， O_1 、 O_2 、 P 三点共面构成了对极约束的几何意义。为了进一步简化对极约束引入本质矩阵 E ，将本质矩阵表示为待求解的旋转矩阵 R 与平移向量 t 的外积，于是对相机位姿的估计问题被分解为求解本质矩阵以及根据本质矩阵求解 R 与 t 的两个子问题。 $3*3$ 的本质矩阵共有 5 个自由度，考虑本质矩阵的非线性性质与尺度等价性，采用八点法^[? ?]对本质矩阵进行估计求解。本质矩阵本身由两个待求解的矩阵相乘所得，因此对本质矩阵采用奇异值分解，分解得到四组矩阵解，以 P 点在两个相机中都具有正深度为依据，剔除其他三组解，最终估计出第二帧相机的位姿。在得到两帧相机的位姿之后，通过三角测量方法求解 3D 点位置，由于噪声的存在，估计得到的位姿无法使 O_1p_1 与 O_2p_2 两条直线相交得到 P ，因此对于三角测量方法更多地采用最小二乘解而不是零解，最小二

乘解指两条直线最接近时特征点的深度。2D-2D 的对极几何约束方法通过构建极平面解决问题，但是它通常需要八个或八个以上的匹配点对，在特征缺失的场景下将无法使用，此外它还存在着初始化、纯旋转和尺度的问题。

3D-3D 方法中不再出现相机模型，它所需的数据来源是多对匹配好的 3D 点，常用迭代最近点（Iterative Closest Point, ICP）^[?]进行求解，ICP 经常被用于激光 SLAM 以及 RGB-D SLAM 中。ICP 的求解分为线性代数方法和非线性优化方法，前者首先计算两组点的质心位置，根据质心位置计算每个点的去质心坐标，然后采用最小二乘法求解使旋转后点的去质心坐标与匹配点去质心坐标误差最小的旋转矩阵，如公式2-14所示：

$$\arg \min_R \frac{1}{2} \sum_{i=1}^n \|q_{1i} - Rq_{2i}\|^2 \quad (2-14)$$

式中 q_{1i} 与 q_{2i} 分别表示第 i 对点中第一个点与第二个点的去质心坐标， R 表示待求解的旋转矩阵。最后通过求解得到的旋转矩阵计算平移量 t ；后者则是以迭代的方式寻找最优值，同样构建最小二乘问题模型，用李代数^[?]表示位姿，得到目标函数并通过不断地迭代找到极小值，在唯一解的情况下将这个极小值作为全局最优值。3D-3D 方法的使用条件是已知特征点的深度，否则它无法进行位姿估计，也因为这一问题，使得 ICP 只能应用于容易获取深度的 RGB-D SLAM 以及激光 SLAM 中。

3D-2D 方法也被称为 n 点透视（Perspective-n-Point, PnP）方法，它描述了已知多个 3D 点空间位置以及它们的投影点坐标时对相机所在位姿进行估计的过程。相比较约束较多的对极几何方法和应用面较窄的 ICP，PnP 方法可以在很少的匹配点中获得较好的运动估计，是比较实用的一种位姿估计方法。仅使用 3 对点进行位姿估计的 P3P^[?]、直接线性变换^[?]、高效 PnP（Efficient PnP, EPnP）^[?]、未标定 PnP（Uncalibrated PnP, UPnP）^[?]都是常用的 PnP 方法，其中 EPnP 利用更多的点对信息，并用迭代的方式优化相机位姿以减少噪声的影响，同时能够快速进行位姿估计，因为这些优点 EPnP 被广泛采用。

2.5 本章小结

本章描述了在计算机视觉领域中四大坐标系的转换关系，详细说明了相机标定、目标跟踪、目标检测、位姿估计这些辅助定位移动机器人的视觉方法，阐述了主流方法的优缺点并描述了它们在定位技术中的主要作用。

第三章 移动机器人跟踪定位系统设计

本章设计了一套移动机器人跟踪定位系统，该系统主要服务环境为室内，共包含四个模块：第一个模块是初始化模块，该模块获取系统初始数据，为整个系统的运作结构化，并为其他模块提供必要的数据；第二个模块是特征检测模块，该模块设计了跟踪器与检测器，在工业相机采集的图像中分析提取图像特征，为机器人定位提供必需的图像特征位置信息；第三个模块是机器人定位模块，该模块根据3D点与对应2D点的位置信息，对相机位姿以及相机视野中的机器人位姿进行估计，并执行估计结果的持久化；第四个模块是可视化模块，该模块对定位结果进行展示，是系统交互性的体现。整体的系统架构如图3.1所示。

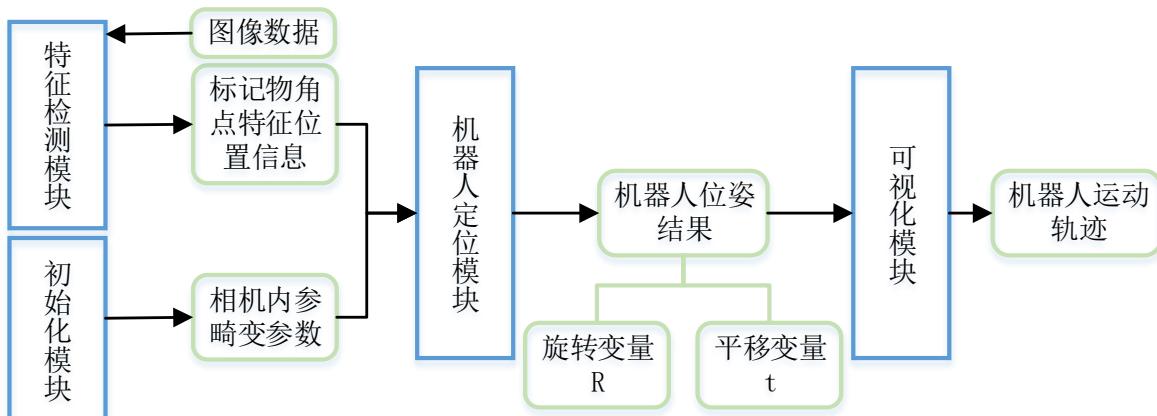


图 3.1 移动机器人跟踪定位系统框架设计图

3.1 初始话模块

初始化模块加载本地数据，感知工业相机与系统的连接，负责对管理整个系统的对象进行初始化，该模块为其他模块提供相机内参数据。本文为该模块设计了本地数据的管理与加载方式和系统对象的结构化。该模块的关系图如图3.2所示。

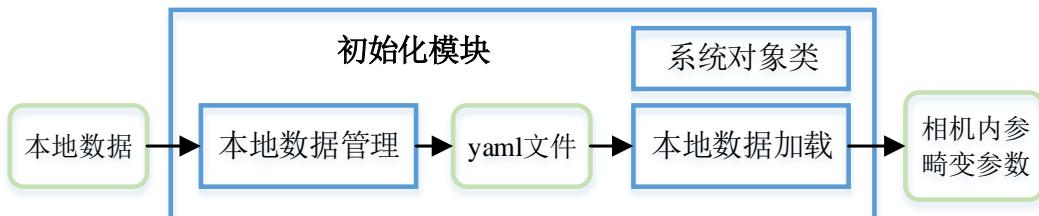


图 3.2 初始话模块关系图

3.1.1 本地数据的管理及加载

本地数据包括相机内参和畸变参数，由离线的相机内参标定流程获取得到，这些数据携带着重要的光学信息，是工业相机应用于计算机视觉领域的具体表现，对相机位姿以及机器人位姿的计算有着不可或缺的作用。系统中参与的工业相机设备不止一台，每一台设备对应一组内参与畸变参数，因此复数组相机光学数据的合理化管理以及数据的高效化加载是系统应当考虑的方面，本文采用 yaml 语法对本地数据进行管理。

本文将本地数据按照 yaml 语法进行组织，具体组织形式如下所示：

- (1) 在 yaml 文件起始位置，设置 yaml 文件特殊标识以及 yaml 语法版本，系统在解析该文件时通过该标识以及版本号确定具体的解析语法；
- (2) 设置 yaml 文件的创建时间 createTime 以及相机设备参数的组数 cameraNum，创建时间不会被系统使用，但是可以区分多个 yaml 文件，有着较好的交互性。相机设备参数的组数在被解析读取之后，成为系统中循环读取参数的次数；
- (3) 为体现良好的交互性，将一台相机设备的相机内参与畸变参数拆分保存，相机内参在 yaml 文件中命名为 cameraMatrix_i （其中 i 为设备编号，通常从0开始），畸变参数在yaml文件中命名为 distcoff_i ，安排同一台设备的相机内参与畸变参数在文件中的位置相邻。无论是内参数据还是畸变参数在系统中使用时都是矩阵的形式，因此本文在 cameraMatrix_i 和 distcoff_i 中以矩阵的格式组织数据，设置 rows 表示矩阵行数， cols 表示矩阵列数， dt 表示数据类型， data 中存储真正的数据。

结合上述关于组织形式的描述，其文件结构如图3.3所示。

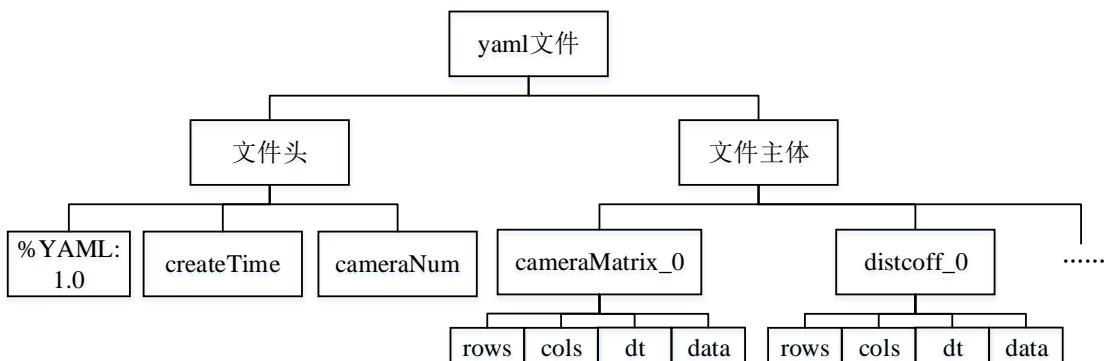


图 3.3 本地数据文件结构

系统在加载本地数据时，首先通过标签 cameraNum 获取文件中的设备数据的组数，然后根据组数循环加载 cameraMatrix_i 和 distcoff_i 下的数据 ($i=0,1,\dots,num-1$)，将加载成功的数据暂存在一块已开辟的内存中，最后关闭文件确保系统的安全性。

3.1.2 系统对象类设计

本文设计了一系列系统对象类结构，对整个系统的运作进行结构化的管理，使得数据的传输与交互更加合理，系统的执行效率更高。系统对象类由系统句柄类、系统辅助类、相机类以及特征检测类组成，系统句柄类对象对应整个系统流程，系统辅助类对象对应定位流程，相机类对象被系统句柄类对象所包含，对应重要数据的管理，特征检测类对象被系统句柄类所包含，对应特征检测流程。图3.6展示了系统对象类图，下面对四个类展开详细描述：

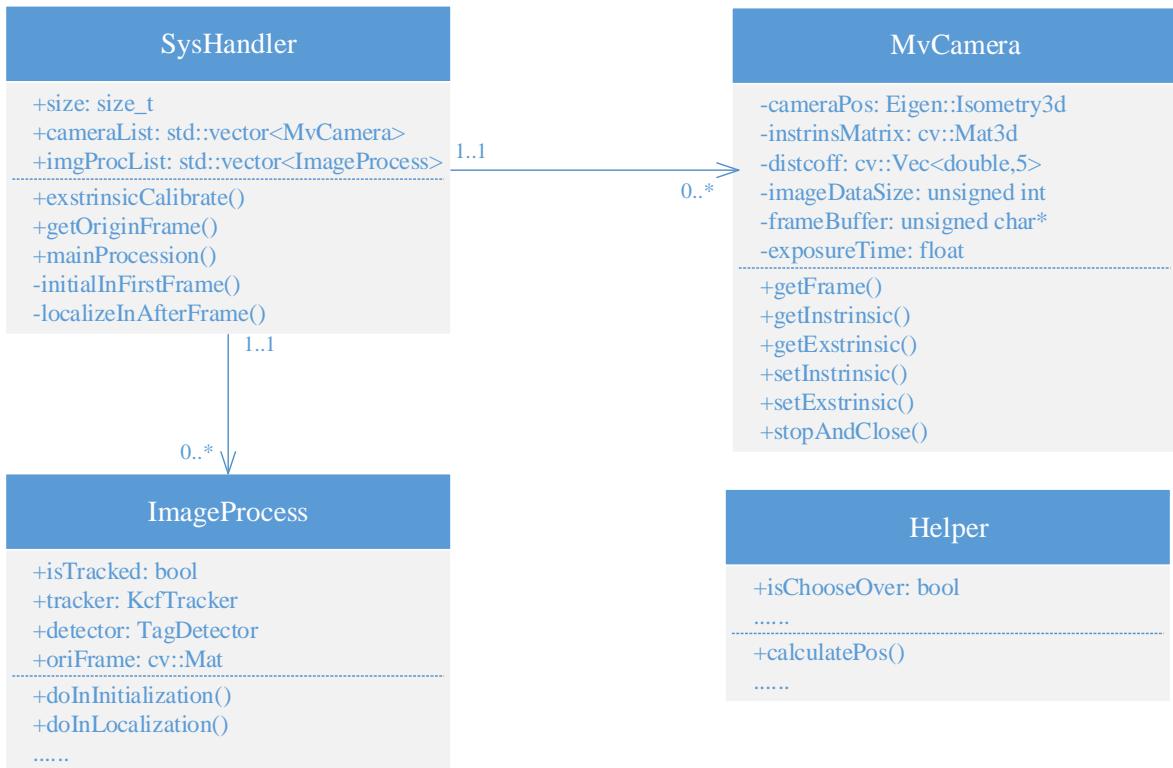


图 3.4 系统对象类结构图

(1) **SysHandler**。系统句柄类，该系统对象在整个系统中仅存在一个，其中包括相机类对象和特征检测类对象的列表、参与系统的相机设备数量以及可视化相关的变量。它管理整个系统流程的执行，维护相机类对象与特征检测类对象的对应关系，是系统的管家；

(2) **MvCamera**。相机类，由工业相机设备抽象而成，其中包括相机内参、畸变参数等标定得到的相机光学数据，经过测算的相机位姿，相机曝光时间、分辨率等硬件参数以及指向原始图像数据的指针。它负责系统对工业相机的感知与驱动，管理着相机本身与定位相关的数据，也是原始图像和系统之间的桥梁。

(3) **ImageProcess**。特征检测类，由特征检测过程抽象而成，其中包括一个跟踪器、一个检测器、表示上一帧是否处理成功的标志位以及由 Mat 类型封装的图像

数据。它主要负责对封装后的图像进行诸如跟踪、检测的处理行为，并在图像中展示处理结果，确保图像数据到图像特征的转变。

(4) Helper。系统辅助类，由定位过程抽象而成，在整个系统中仅存一个，其中不仅包括了对机器人位姿估计的行为，还存在优化位姿的各种参数，因此称其为辅助类。它的主要目的即从相机类对象中获取相机内外参和畸变参数，从特征检测类对象中获取图像特征信息，从而对机器人执行定位。

3.2 特征检测模块

特征检测模块在图像中进行目标跟踪和特征检测，将无用的像素信息滤去，获取关键特征在图像中的位置信息，该信息也作为此模块的输出，提供给机器人定位模块。本文为该模块设计了跟踪器以及检测器，以实现特征提取的功能。该模块的关系图如图3.5所示。

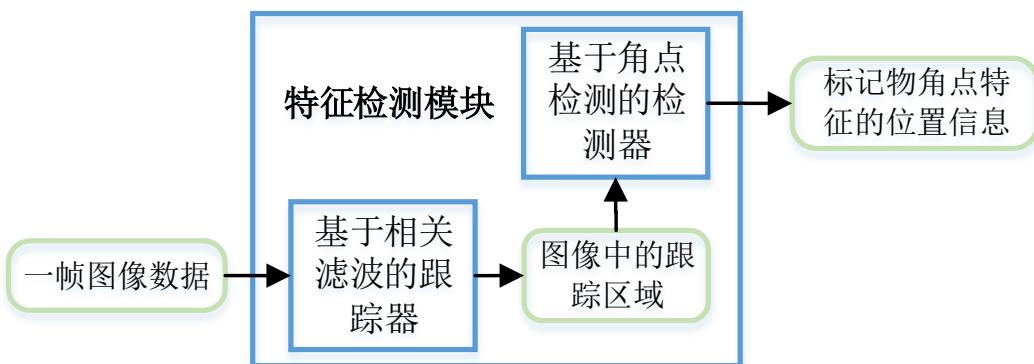


图 3.5 特征检测模块关系图

3.2.1 基于相关滤波的跟踪器设计

基于深度学习的目标跟踪方法能够产生准确率最高的跟踪结果，但是本系统在定位时追求实时，因此在对图像中的目标进行跟踪时，采用实时性较强的相关滤波方法，设计一个基于相关滤波的跟踪器，跟踪器的工作流程如图3.6所示，在整个流程中，样本的采集和分类器的设计是影响跟踪器跟踪效果的重要因素，下面围绕这两点说明基于相关滤波的跟踪器的设计思路。

基于相关滤波的目标跟踪问题本质是对正负样本的分类问题，其中正样本为目标，负样本为背景，所以将目标跟踪问题表述为线性回归函数，如公式3-1所示：

$$f(x_i) = \omega^T x_i \quad (3-1)$$

ω 表示权重系数，是优化的对象。该线性回归函数的损失函数表示为：

$$\min_{\omega} \sum_i (f(x_i) - y_i)^2 + \lambda \|\omega\|^2 \quad (3-2)$$

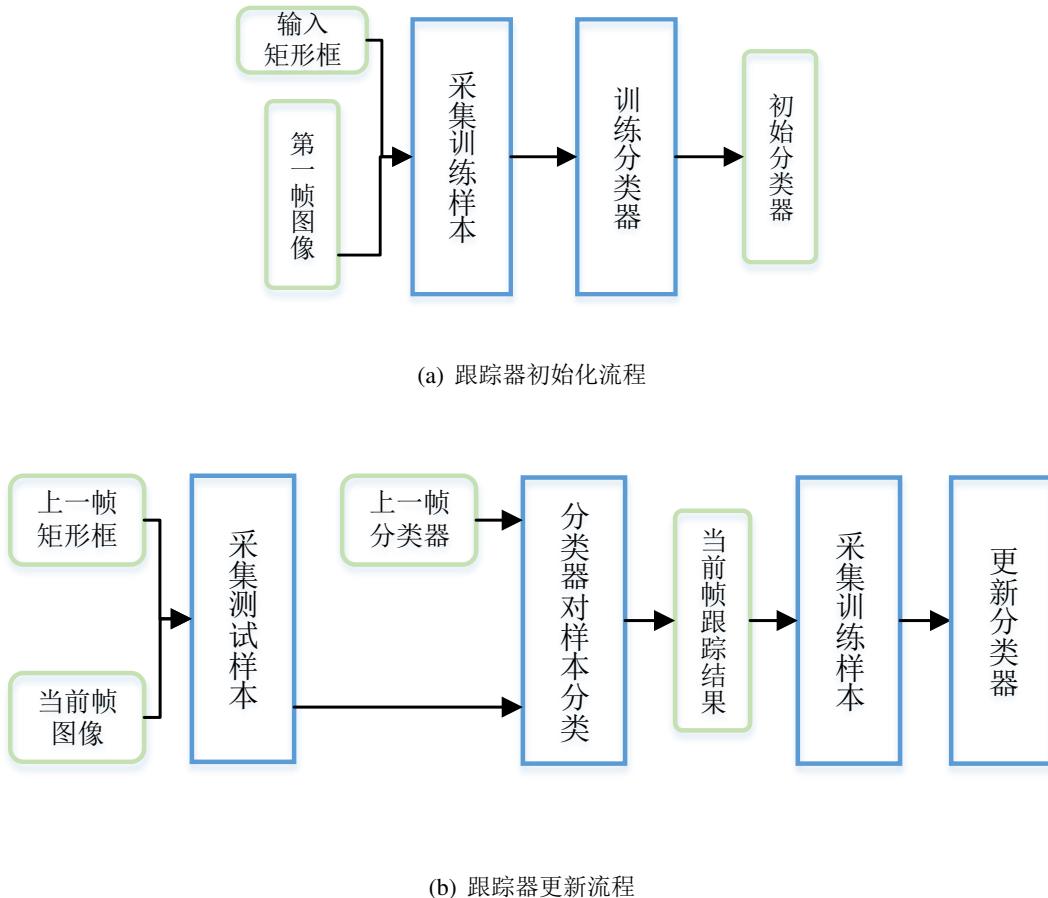


图 3.6 跟踪器工作流程图

训练样本集为 (x_i, y_i) , 其中 x_i 表示样本, y_i 表示回归的目标, λ 表示防止过拟合的正则系数。将公式3-2写成矩阵形式易于对 ω 进行求解, 如公式3-3所示。

$$\min_{\omega} ||X\omega - y||^2 + \lambda ||\omega||^2 \quad (3-3)$$

其中 X 是样本组成的矩阵， y 是回归目标组成的列向量。为了确定 ω 的最优值，在公式3-3中对 ω 求偏导，得到 ω 的封闭解：

$$\omega = (X^T X + \lambda I)^{-1} X^T y \quad (3-4)$$

考虑到目标机器人在视野中的移动不会非常快速，临近帧的目标区域往往分布在当前帧目标区域的周围，而不会突然大幅远离当前帧目标区域的位置，所以对当前帧目标区域的周围以相同的窗口大小进行采样，形成样本集。然而，周边区域的样本采集会引入新的背景特征，加重了更新分类器的负担，产生了更多的时间开销，而在下一帧的机器人跟踪中往往只关心机器人相对上一帧的移动信息，本文采用循环矩阵表征机器人的移动信息，将当前帧的目标区域循环位移产生的图像作为分类

器的训练样本，因此 X 本身是一个循环矩阵。循环矩阵在傅式空间中能够使用离散傅里叶矩阵进行对角化，如下所示：

$$X = F \text{diag}(\hat{x}) F^H \quad (3-5)$$

x 对应循环移位生成 X 的样本， \hat{x} 表示 x 经过傅里叶变换之后的值， F 是离散傅里叶矩阵，是一个常量矩阵。公式3-4中对权重系数 ω 的求解存在大量的求逆运算，当矩阵的规模变大时，求逆运算带来的计算开销会非常巨大，而循环矩阵的这一性质能够完全消除求逆运算，将公式3-4转换到复数域中并应用循环矩阵对角化的性质：

$$\begin{aligned} \omega &= (X^H X + \lambda)^{-1} X^H y \\ &= (F \text{diag}(\hat{x}^*) F^H F \text{diag}(\hat{x}) F^H + \lambda F^H)^{-1} F \text{diag}(\hat{x}^*) F^H y \\ &= (F \text{diag}(\hat{x}^* \odot \hat{x} + \lambda) F^H)^{-1} F \text{diag}(\hat{x}^*) F^H y \\ &= F \text{diag}\left(\frac{\hat{x}^*}{\hat{x}^* \odot \hat{x} + \lambda}\right) F^H y \end{aligned} \quad (3-6)$$

对公式3-6的等式两边作傅里叶变换最终化简为：

$$\hat{\omega} = \frac{\hat{x}^* \odot \hat{y}}{\hat{x}^* \odot \hat{x} + \lambda} \quad (3-7)$$

循环矩阵在傅式空间可对角化的性质将对权重系数 ω 的求解转换到傅式空间，在该空间中求解 ω 的傅里叶变换仅需要通过向量的点乘即可，减少了计算开销。

训练样本集在低维特征空间中通常是线性不可分的，也正因为这个原因，跟踪器在低维特征空间中的跟踪会存在跟踪丢失或者跟踪出错的情况。为了解决这个问题，在分类器的设计中引入核函数 $\kappa(i, j)$ ，通过非线性映射函数 φ 将原始训练样本数据映射到合适的高维特征空间，在高维特征空间中确保训练样本集的线性可分。在高维特征空间中，分类器的线性回归函数变为：

$$f(x_i) = \omega^T \varphi(x_i) \quad (3-8)$$

ω 的最优值可以表示为训练样本的线性组合，利用该性质在高维特征空间中引入新的权重系数 β ， β 与 ω 的关系可表示为：

$$\omega = \sum_{i=1}^n \beta_i x_i \quad (3-9)$$

对 β 的求解与低维空间求解 ω 时相同，利用岭回归方法构建损失函数，对式中的 β 进行偏导求解，得到 β 的封闭解：

$$\beta = (K + \lambda I)^{-1} y \quad (3-10)$$

K 是核空间的核矩阵，表示为两个矩阵 $\varphi(X)$ 和 $\varphi(X)^T$ 相乘的形式，核矩阵的元素由训练样本通过核函数转化而成，因此核矩阵也是一个循环矩阵，能够在傅式空间进行对角化，将对 β 求解时的求逆运算消除。转换的过程如公式3-11所示：

$$\beta = F \text{diag}\left(\frac{1}{\hat{k} + \lambda}\right) F^H y \quad \hat{\beta} = \frac{\hat{y}}{(\hat{k} + \lambda)^*} \quad (3-11)$$

最终分类器在傅式空间中权重系数已经得到，为进一步加快测试样本的分类速度，将整个分类过程转移到傅式空间中进行，如公式3-12所示。

$$\begin{aligned} f(z) &= (\beta^T \varphi(X) \varphi(Z)^T)^T \\ &= (\beta^T K')^T \\ &= (K')^T \beta \\ &= F \text{diag}(\hat{k}') F^H \beta \\ \hat{f} &= (\hat{k}')^* \odot \beta \end{aligned} \quad (3-12)$$

Z 由测试样本组成， K' 代表了测试样本与训练样本在核空间的核矩阵， \hat{k}' 对应循环移位生成 $(K')^T$ 的样本。...

在新的一帧图像中执行跟踪时，跟踪器首先采集图像中预测区域以及该区域循环位移形成的区域作为测试样本集，将这些测试样本数据分别代入公式3-12中得到每一个测试样本的响应，将响应最高的那一个作为当前帧的目标区域。此时将当前帧目标区域以及该区域循环位移形成的区域作为训练样本集，利用公式3-11对权重参数 β 进行在线更新，等待下一帧的到来。

该跟踪器融合了核函数与高维特征空间，利用循环矩阵的特性，能够进行实时跟踪并保证跟踪的准确性。使用该跟踪器将图像中的冗余信息过滤，最终把需要用于检测的部分由全图缩小至跟踪目标区域，极大地提升了定位系统的性能。

3.2.2 基于角点检测的检测器设计

由于移动机器人品种繁多，移动机器人的外表特征过于繁杂，这极大地影响了系统对机器人的认知。因此，本文采用在机器人表面依附标记物的方式达到特征统一的目的，同时令标记物的位姿无限接近机器人的位姿，在定位计算时会更加便利。

四月码（AprilTag）[?]是一种具有二维条形码风格的标记物，其图案以特殊的方式存储区别度较高的视觉信息，使得四月码更加容易被识别。图3.7中展示了三类四月码图案，Tag36h11表示该四月码所属的家族，其有效区域为6*6，最小汉明距离为11，汉明距离越小表示该四月码的校验信息越丰富，与环境的区别度越大。相比其他的二维条形码如QR码[?]，四月码的辅助定位效果更加出色，它虽然包含较



图 3.7 四月码图案

小的视觉信息负载，但是在分辨率低、光照不均匀或者旋转不规则的情况下依旧包含比较丰富的特征信息，辅助检测器进行检测。本文借助四月码设计了基于角点检测的特征检测器，通过对图像中四月码的检测获取四月码外围四个角点像素的位置信息，并将这些位置信息用于机器人定位。

四月码在图像中总是呈现不规则的封闭四边形形状，从这一特点出发，将四月码检测问题转换为图像区域中的四边形检测问题，检测器围绕这个问题，设计了四边形检测方法。从检测器检测速度的方面考虑，由四边形检测方法交付给四月码图案库作特征匹配的四边形应当控制数量并确保正确的四月码在其中，因此在四边形检测方法中过滤掉大部分的无用四边形区域是提升检测器性能的关键。

检测器在四边形检测方法中的工作主要分为以下几个方面：

(1) 图像预处理。考虑到在复杂环境中可能存在多个规模较小的四边形，这些四边形不包含正确的四月码特征信息，它们庞大的数量影响了检测器的检测效率，甚至在部分四边形区域中会存在与四月码相似的特征信息，引起检测器的误检与漏检，此外环境噪声会对图像质量造成巨大的负面影响，许多像素会因此发生变化，从而阻碍了正确线段的检测，由此检测器对图像进行低通滤波，本文采用了高斯低通滤波器，如公式3-13所示：

$$H(u, v) = e^{\frac{-D^2(u, v)}{2D_0^2}} \quad (3-13)$$

图??与图??分别展示了低通滤波前后的图像。高斯低通滤波器使用像素邻域的加权

图 3.8 低通滤波前的图像

图 3.9 低通滤波后的图像

均值代表像素点的像素值，确保滤波后图像不会失真并弱化四月码的特征，同时高斯函数傅里叶变换的单瓣性质使得滤波器能够过滤高频信号（比如噪声）并保留大部分所需的信号，因此经过低通滤波之后，图像降噪且像小规模四边形这种细纹理的区域将不会被检测到。

(2) 线段检测。线段是由多个梯度方向与强度相似的像素点组成，在这一部分，检测器的主要工作是计算图像中像素点的梯度方向与强度并以此为根据对像素点进行分类，组成同一条线段的像素点归为一类。各个像素点梯度方向与强度的计算可以通过图像降采样减轻计算负担，代价是降低了图像的分辨率，使得一些比较

小的四月码无法被检测出来。而像素点分类则无法避免地消耗大量时间，为解决这一问题，本文引入聚类算法辅助像素点进行分类。像素点聚类的过程如 **Algorithm 1** 所示。聚类算法是基于图的算法，在像素点分类这个问题中，每个像素表示图中节

Algorithm 1 像素点聚类算法

Input: 聚类 A , 聚类 B

Output: 聚类 A 与 B 能否合并的判断

```

1:  $D(A) \leftarrow dA_{max} - dA_{min}$ 
2:  $D(B) \leftarrow dB_{max} - dB_{min}$             $\triangleright D(\bullet)$ 由聚类中最大与最小梯度方向作差得到
3:  $M(A) \leftarrow mA_{max} - mA_{min}$ 
4:  $M(B) \leftarrow mB_{max} - mB_{min}$             $\triangleright M(\bullet)$ 由聚类中最大与最小梯度幅值作差得到
5:  $G_D \leftarrow 100, G_M \leftarrow 1200$            $\triangleright$ 控制参数, 使聚类内部的变化适度增加
6:  $D(A \cup B) \leftarrow \max(dA_{max}, dB_{max}) - \min(dA_{min}, dB_{min})$ 
7:  $M(A \cup B) \leftarrow \max(mA_{max}, mB_{max}) - \min(mA_{min}, mB_{min})$ 
8: if  $D(A \cup B) \leq \min(D(A), D(B)) + G_D/|A \cup B|$ 
9:   and  $M(A \cup B) \leq \min(M(A), M(B)) + G_M/|A \cup B|$  then
10:    将 $A$ 与 $B$ 合并
11:    return true
12: else
13:   return false
14: end if

```

点，相邻像素之间存在一条边，边的权值表示这两个像素的梯度方向差。最终通过边的权值与梯度方向差阈值的比较判定某两个像素点集合能否合并。在具体实现上采用了多线程执行像素点聚类的方式，图3.10展示了该方式，按照图像的高将图像

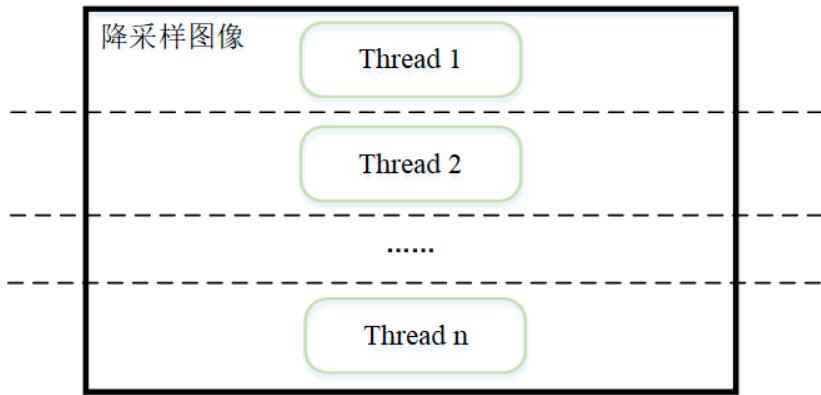


图 3.10 多线程执行线段检测方式

等分为几个区域，并在多个区域同时进行像素点的聚类，最后在各个区域的分割处对已存在的聚类再次进行聚类。

(3) 四边形区域检测。线段检测结束后，同一条线段中的像素点被聚集在同一类中，类的数量代表了候选线段的数量，需要在这些线段中寻找能够组成封闭四边形的线段。规定在逆时针方向上，一条线段的末端点和另一条线段的始端点之间

的距离小于阈值时，这两条线段被认为是有交点的。因为一个四边形的组成需要四条遵守上述规定的线段，所以采用深度为 4 的深度优先搜索算法检测四边形。深度优先搜索算法的过程如 **Algorithm 2** 所示。在深度搜索的第一层，将所有线段作为

Algorithm 2 深度优先搜索算法

Input: 节点数组 A , 数组大小 n

```

1: function DFS_SEARCH
2:   初始化  $hashmap$ 
3:    $count \leftarrow 0$ 
4:   for all  $node \in A$  do
5:     if  $hashmap[node]$  is false then
6:       DFS( $A, node, hashmap, count$ )
7:     end if
8:   end for
9: end function

10:
11: function DFS( $A, node, hashmap, count$ )
12:   if  $hashmap[node]$  is true or  $count = 4$  then
13:     return
14:   end if
15:    $hashmap[node] \leftarrow true$ 
16:    $count \leftarrow count + 1$ 
17:   for all  $adjnode \in A$  do
18:     if  $hashmap[adjnode]$  is false then
19:       DFS( $A, adjnode, hashmap, count$ )
20:     end if
21:   end for
22: end function

```

节点，并统计各条线段末端点的位置，在第二层中的搜索仅考虑那些始端点与上一层线段末端点位置足够接近的线段，三、四层的搜索也以这种方式进行，足够接近的定义表示为点与点之间的距离小于两倍的线段像素长度加 5 个额外的像素，如公式3-14所示：

$$distance \leq threshold = 2 * line_length + 5 \quad (3-14)$$

这一大阈值的选取大大降低了四月码被漏检的可能性，但也因此增加了不少候补四边形区域，用性能换取了检测的准确性。在具体实现上为了加速查找并过滤那些不遵守规定的线段，采用了二维查找表这种数据结构存储候选线段，被重复使用的线段也会被过滤，防止冗余搜索的进行，这一实现使得四边形区域检测在整个检测器工作过程中所占用的时间非常少。

四边形检测方法获得了一些候补四边形区域，四月码所表示的四边形区域也在其中，分别将这些四边形区域中的角点特征信息提取，与四月码图案库中的特征信

息进行匹配，最终根据匹配程度确认图像中四月码的位置，而四月码四个外围角点的位置早在深度优先搜索时便保存在二维查找表中。检测器将这四个角点特征的位置作为检测器的结果输出至其他模块，辅助机器人定位的进行。

3.3 机器人定位模块

机器人定位模块结合图像关键特征的位置信息、相机内参和畸变参数，使用 3D-2D 位姿估计方法计算相机和机器人的位姿，同时将这些位姿结果保存在本地，为后续数据的对比提供基础，是整个系统中最为重要的模块。本文为该模块设计了相机外参标定方法和机器人位姿估计方法，使系统在充分利用图像信息之后能够得到较为准确的定位结果。该模块的关系图如图3.11所示。

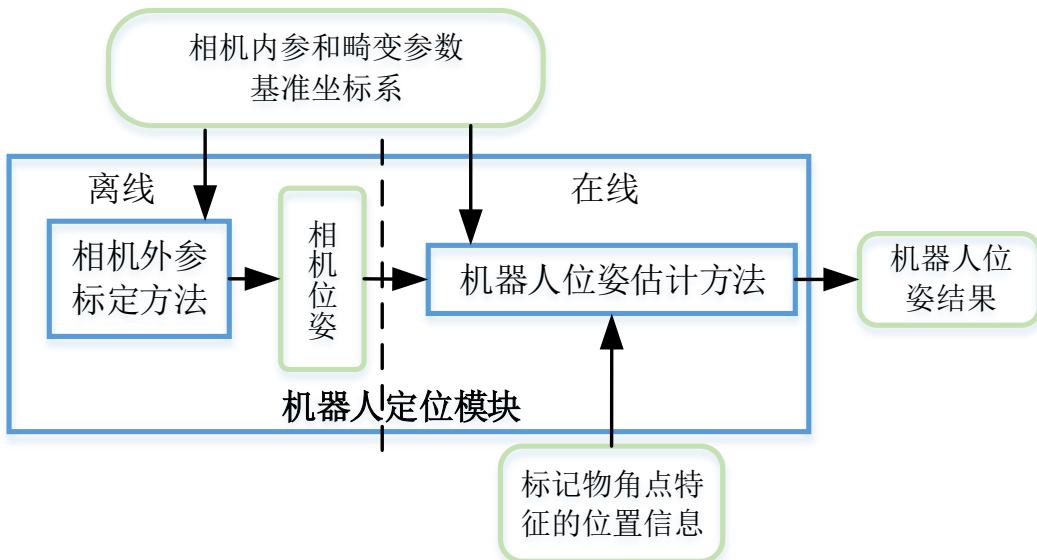


图 3.11 机器人定位模块流程图

3.3.1 基于3D-2D的位姿估计方法研究

EPnP 方法能够利用更多的 3D-2D 点对信息，并以迭代的方式优化相机位姿从而减少噪声的影响，同时能够快速进行位姿估计，是 3D-2D 位姿估计方法中最常用的方法。它的核心思想是选定一组虚构的控制点，将世界坐标系中的三维坐标表示为这些控制点的加权和。控制点被约束为 4 个不共面的点，优化仅围绕着这 4 个控制点进行，此外在求解的过程中考虑了 4 个奇异向量，所以相对其他 PnP 方法 EPnP 更加高效、准确。

EPnP 方法通过求解四个控制点在相机坐标系下的坐标进一步估计相机的位姿。其主要过程分为以下三步：

- (1) 选择控制点构成新的辅助坐标系。选择的控制点首先保证辅助坐标系唯

一，其次进一步增强系统的稳定性，因此方法中取所有已知 3D 点的质心位置作为第一个控制点，其他三个点的选择则分布在数据的主方向上。假设 c_1^w 是第一个控制点，采用主成分分析选择剩下的三个控制点，每一个 3D 点坐标与第一个控制点坐标作差后的结果构成一个 $n*3$ 的 A 矩阵，计算 $A^T A$ 的 3 个特征值 $\lambda_1, \lambda_2, \lambda_3$ ，它们对应的特征向量分别为 $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ 。于是剩下的三个控制点可选择为：

$$\begin{cases} c_2^w = c_1^w + \sqrt{\frac{\lambda_1}{n}} \mathbf{v}_1 \\ c_3^w = c_1^w + \sqrt{\frac{\lambda_2}{n}} \mathbf{v}_2 \\ c_4^w = c_1^w + \sqrt{\frac{\lambda_3}{n}} \mathbf{v}_3 \end{cases} \quad (3-15)$$

(2) 将已知 3D 点从世界坐标系转换到辅助坐标系。由于每一个 3D 点都可以被表示为四个控制点的加权和，则有如下关系：

$$p_i^w = \sum_{j=1}^4 \alpha_{ij} c_j^w \quad \sum_{j=1}^4 \alpha_{ij} = 1 \quad (3-16)$$

p_i^w 表示第 i 个 3D 点的坐标，向量 $[\alpha_{i1}, \alpha_{i2}, \alpha_{i3}, \alpha_{i4}]$ 表示 3D 点在以四个控制点为基的欧式空间中的坐标，向量中元素之和被约束为 1。由公式 2-1 可知，控制点在世界坐标系下的坐标可以借由相机外参的作用转换到相机坐标系下，对于一个 3D 点，其在相机坐标系下可表示为：

$$p_i^c = [R|t] \begin{bmatrix} p_i^w \\ 1 \end{bmatrix} = [R|t] \begin{bmatrix} \sum_{j=1}^4 \alpha_{ij} c_j^w \\ \sum_{j=1}^4 \alpha_{ij} \end{bmatrix} = \sum_{j=1}^4 \alpha_{ij} c_j^c \quad (3-17)$$

从式中可以看出即使在相机坐标系下 3D 点也能被表示为四个控制点的加权和，且参与运算的向量 $[\alpha_{i1}, \alpha_{i2}, \alpha_{i3}, \alpha_{i4}]$ 与世界坐标系中的一致，因此世界坐标系下计算得到的 α_{ij} 同样适用于相机坐标系。

(3) 求解控制点在相机坐标系下的坐标。第一步与第二步中利用了 3D-2D 的 3D 部分，得到了四个控制点在世界坐标系下的坐标以及每一个 3D 点的辅助坐标。这一步主要结合已知数据与 2D 部分对控制点在相机坐标系下的坐标进行求解。首先引入相机投影模型，将 3D 点在世界坐标系下的坐标投影至像素平面，并加入控制点元素，最终化简得到：

$$\begin{cases} \sum_{j=1}^4 (\alpha_{ij} f_x x_j^c + \alpha_{ij} (c_x - u_i) z_j^c) = 0 \\ \sum_{j=1}^4 (\alpha_{ij} f_y y_j^c + \alpha_{ij} (c_y - v_i) z_j^c) = 0 \end{cases} \quad (3-18)$$

其中 f_x, f_y, c_x, c_y 表示相机内参， u_i, v_i 表示 2D 点的坐标， x_j^c, y_j^c, z_j^c 表示待求解的四个控制点在相机坐标系下的坐标，共有 12 个待求目标。一个 2D 点可以

确定两个求解方程，当 n 个点全部参与进来，可以确定 $2n$ 个求解方程，于是对公式3-18整理得：

$$M\mathbf{x} = 0 \quad (3-19)$$

M 是一个 $2n*12$ 的矩阵， \mathbf{x} 中包含了 12 个待求目标。方法中没有直接对 M 进行 SVD 分解求解 \mathbf{x} ，而是对 $M^T M$ 进行 SVD 分解，将算法时间复杂度从 $O(n^3)$ 降为 $O(n)$ ，这也是 EPnP 高速的原因之一。最后针对相机坐标系下四个控制点的取值向量不同，考虑 4 个奇异向量，完成对四个控制点在相机坐标系下坐标的高精度估计。

EPnP 方法最终求得了控制点在相机坐标系下的坐标，与控制点在世界坐标系下的坐标形成 3D-3D 的关系，最终通过 ICP 方法估计相机的位姿。

3.3.2 相机外参标定方法

固定位置的单目工业相机因为缺少图像中的深度信息，无法对自身的位姿进行计算，因此本文借助四月码标记物，利用标记物平面到相机成像平面的匹配点，离线求解单目工业相机的固定位姿。

在进行外参标定之前，系统需要一个固定的基准坐标系，所有相机以及机器人的位姿都以该坐标系为参照物，即世界坐标系。本文将世界坐标系的原点设置为四月码标记物的中心点，将 x 轴与 y 轴组成的平面设置为四月码标记物的平面，x 轴与 y 轴在中心点相交，将四月码标记物划分为四个面积相等的正方形，z 轴垂直平面且正方向向上，如图3.12所示。为消除相机图像四角的畸变对四月码标记物造成形变的

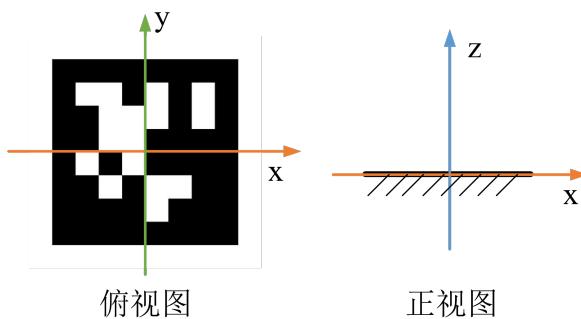


图 3.12 以四月码标记物为基础的世界坐标系设置示意图

不良影响，四月码标记物在图像中的位置接近相机视野的中心，且尺寸足够大。通过以上物理约束，本文减小了外参标定存在的误差。

在世界坐标系设置完成后，对相机外参标定过程就转变为求解世界坐标系与相机坐标系之间转换关系的过程。两坐标系的位置关系如图3.13所示。系统通过检测图像中的四月码标记物，获得其四个外围角点的像素坐标 $p_i(u_i, v_i)$ （其中 $i = 0, 1, 2, 3$ ），同时根据世界坐标系确定四个外围角点在三维世界下的3D坐标 $P_i(x_i, y_i, z_i)$

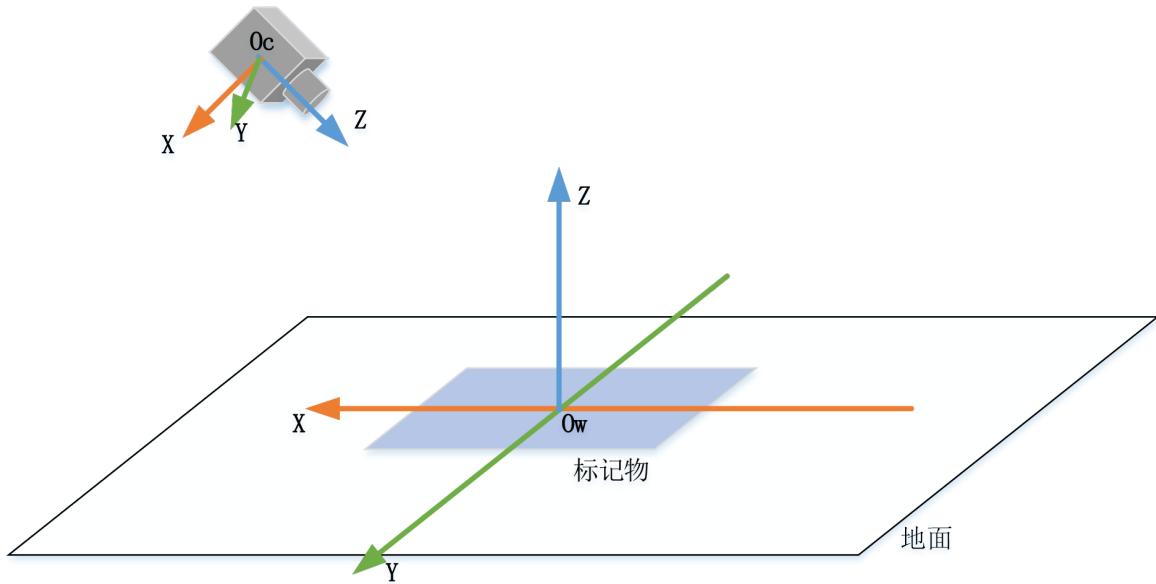


图 3.13 相机坐标系与世界坐标系位置关系示意图

(其中 $i = 0, 1, 2, 3$), 如公式3-20所示, 将四个点的像素坐标与3D坐标一一对应。

$$\begin{aligned}
 p_1(u_1, v_1) &\leftrightarrow P_1(x_1, y_1, 0) \\
 p_2(u_2, v_2) &\leftrightarrow P_2(x_2, y_2, 0) \\
 p_3(u_3, v_3) &\leftrightarrow P_3(x_3, y_3, 0) \\
 p_4(u_4, v_4) &\leftrightarrow P_4(x_4, y_4, 0)
 \end{aligned} \tag{3-20}$$

3D-2D 位姿估计方法 EPnP 算法接收这四组数据结合相机内参与畸变参数, 估计旋转矩阵 R_{CW} 以及平移向量 t_{CW} , 这两个结果表示相机外参, 即相机在三维世界中的位姿。相机外参对两坐标系转换关系如公式3-21所示:

$$P_C = R_{CW}P_W + t_{CW} \tag{3-21}$$

公式(3-21)中 P_C 表示3D点在相机坐标系下的坐标, P_W 表示3D点在世界坐标系下的坐标, R_{CW} 表示相机坐标系与世界坐标系之间的旋转矩阵, t_{CW} 表示相机坐标系与世界坐标系之间的平移向量。

3.3.3 机器人位姿估计方法

在每一帧图像中, 机器人估计计算方法接收四月码四个外围角点在整张图像中的像素坐标, 结合已经存在的相机内外参、畸变参数、世界坐标系, 对机器人当前帧的位姿进行计算。在上一节也提到过, 四月码统一了机器人的特征, 且代表了机器人的位姿, 系统计算出四月码的位姿, 即求解了机器人的位姿, 实现了初步定位。

本文引入物体坐标系的概念，物体坐标系表示附着在机器人上的四月码标记物形成的坐标系，该坐标系与世界坐标系相似，以四月码标记物的中心作为坐标系原点，本文再次采用 EPnP 位姿估计方法求解相机坐标系与物体坐标系之间的转换关系，如公式3-22所示：

$$P_C = R_{CM}P_M + t_{CM} \quad (3-22)$$

P_C 表示3D点在相机坐标系下的坐标， P_M 表示3D点在物体坐标系下的坐标， R_{CM} 表示相机坐标系与物体坐标系之间的旋转矩阵， t_{CM} 表示相机坐标系与物体坐标系之间的平移向量。

已知相机坐标系与世界坐标系、物理坐标系之间的位姿转换关系，本文采用链式位姿求解思路将世界坐标系与物理坐标系关联起来。三个坐标系的转换关系如图3.14所示。一个3D点在世界坐标系下的坐标与在相机坐标系下的坐标存在数值化

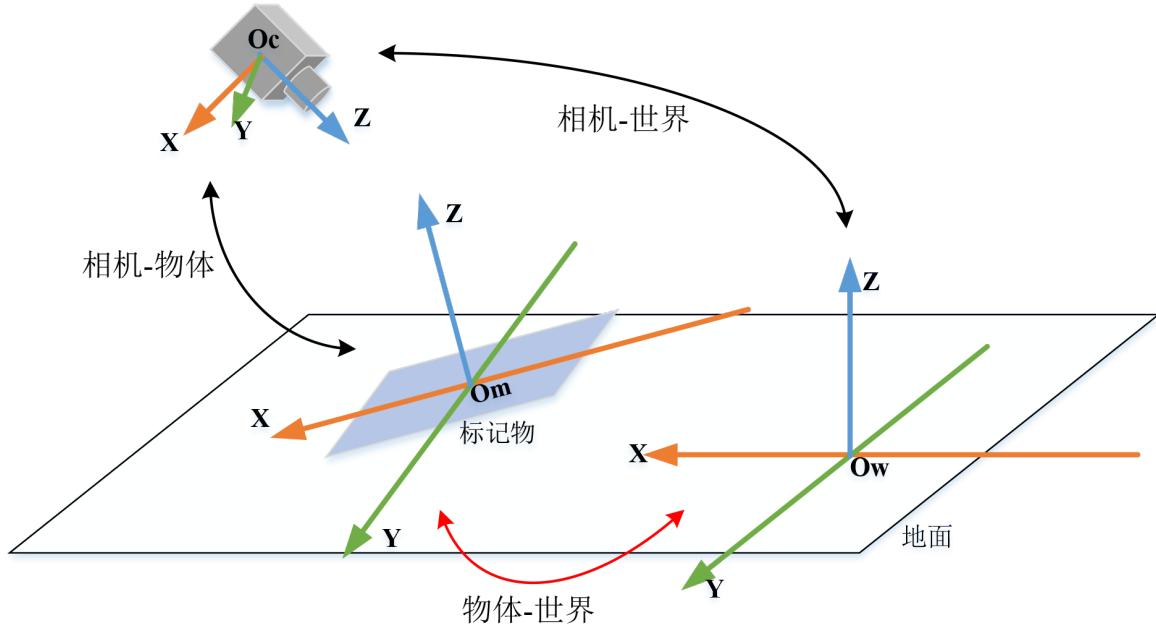


图 3.14 坐标系之间转换关系示意图

的转换关系，并且它在物体坐标系下的坐标与在相机坐标系下的坐标也存在数值化的转换关系，通过这两组转换关系，本文寻求该点在世界坐标系下的坐标与物体坐标系下坐标的转换关系。因此，本文结合公式3-21和公式3-22，得到公式3-23：

$$R_{CW}P_W + t_{CW} = R_{CM}P_M + t_{CM} \quad (3-23)$$

在公式3-23中， R_{CW} 、 t_{CW} 、 R_{CM} 、 t_{CM} 均为已知。因为旋转矩阵 R_{CM} 是正交矩阵，所以对公式3-23两侧进行整理得到公式3-24：

$$P_M = (R_{CM}^T R_{CW})P_W + R_{CM}^T(t_{CW} - t_{CM}) \quad (3-24)$$

在公式3-24中已经可以得到物体坐标系与世界坐标系之间的位姿转换关系，如公式3-25所示：

$$\begin{aligned} R_{MW} &= R_{CM}^T R_{CW} \\ t_{MW} &= R_{CM}^T (t_{CW} - t_{CM}) \end{aligned} \quad (3-25)$$

公式3-25中 R_{MW} 表示物体坐标系与世界坐标系之间的旋转矩阵， t_{MW} 表示物体坐标系与世界坐标系之间的平移向量。此时系统可以根据这些位姿信息求解物体坐标系中容易确定3D坐标的点在世界坐标系下的位置。本文采用物体坐标系原点在世界坐标系下的位置表示四月码的位姿，从而代表机器人的位姿，即如公式3-26所示：

$$P_M = (0, 0, 0) \quad (3-26)$$

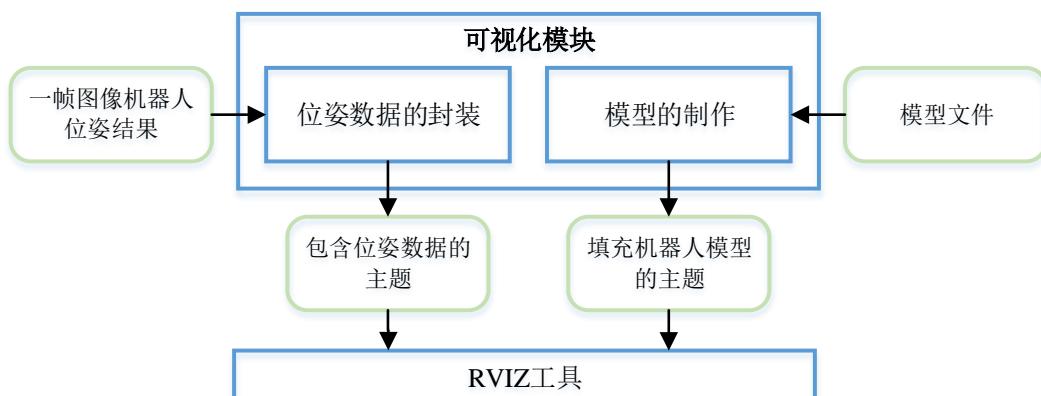
结合公式3-24，可以得到 P_W ，如公式3-27所示：

$$P_W = R_{CW}^{-1} (t_{CM} - t_{CW}) \quad (3-27)$$

公式3-27中的 P_W 表示四月码中心点在世界坐标系下的位置， t_{MW} 展示的是坐标系平移的过程， P_W 表示的是一个3D点的实际位置，显得更加直观。公式3-25中的 R_{MW} 表示结果位姿中的方向部分， P_W 表示结果位姿中的位置部分，后续对结果的展示主要以位置部分为主。

3.4 可视化模块

可视化模块接收每一帧图像对应的机器人位姿结果，借助 ROS 平台对原始的位姿结果数据进行封装与发布，同时解析模型文件加载机器人模型数据并进行发布。rviz 工具订阅发布的位姿数据与模型数据，形成可视化的定位路径与机器人模型，使整个场景更加真实直观。本文为该模块设计了位姿数据的封装方式以及模型的结构，该模块的关系图如图3.15所示。



3.4.1 位姿数据的封装与加载

本文设计了 path 结构体对位姿数据进行封装，其结构如图3.16所示。 path 结构

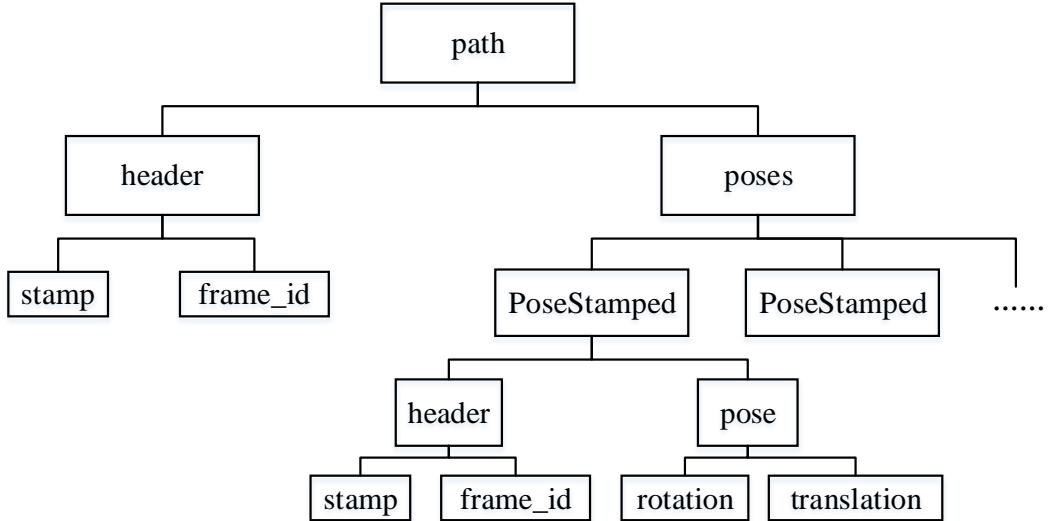


图 3.16 位姿数据封装结构体

体由头部 header 与位姿数组 poses 组成，头部中包含时间戳 stamp 与数据唯一标识 frame_id ， 时间戳指位姿数据开始发布时的时间，唯一标识在 rviz 中被用来与其他可视化数据进行区分。位姿数组 poses 中保存着多组 PoseStamped 类型的位姿变量，这些位姿变量中存储每一帧图像对应的位姿结果，由头部 header 与位姿结果 pose 构成， pose 中存储的位姿以平移向量的形式存储平移量 translation ，以四元数的形式存储旋转量 rotation 。

在将位姿数据封装入 path 结构体后， ROS 将位姿数据相关的主题进行发布， rviz 工具对这个主题进行订阅，如图3.17所示。图中...

待添加

图 3.17 位姿数据的发布与订阅结构图

3.4.2 机器人模型的制作与加载

可视化模块不仅对连续位姿结果形成的轨迹进行展示，还在场景中加载机器人模型进行展示，使可视化的场景更加直观。本文基于扫地机器人进行机器人模型的构建，采用统一机器人描述格式（Unified Robot Description Format, URDF）文件对扫地机器人的结构进行描述。

URDF 文件是一种用于描述机器人及其部分结构、关节、自由度等相关信息的 XML 格式文件，在文件中对机器人的描述分为环节与关节两个部分，其中环节表示机器人的各个组成部分，关节则描述了各个环节之间的连接关系。扫地机器人由一个机器人主体、主体上方的激光传感器以及主体下方起支撑与移动作用的三个轮子组成，机器人主体连接了激光传感器与三个轮子，因此扫地机器人在 URDF 文件中被描述为具有五个环节与四个关节的机器人模型，如图3.18和图3.19所示。
`base_link` 映射到机器人主体， `left_wheel`、`right_wheel` 和 `mid_wheel` 分别映射到机器

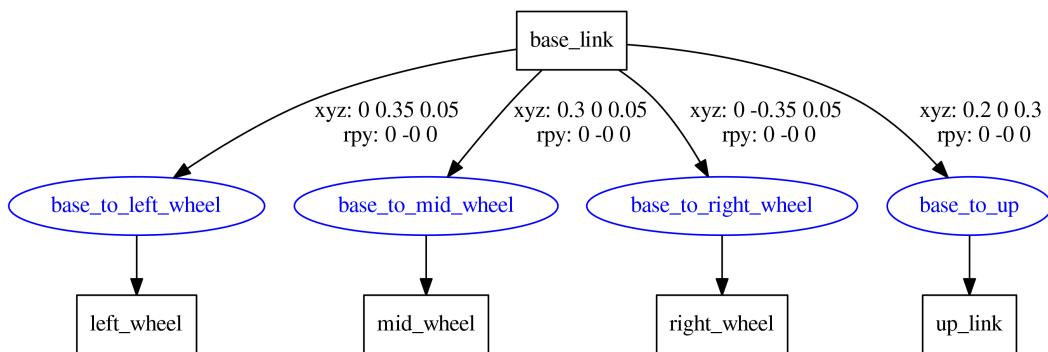


图 3.18 机器人模型各环节与关节的关系图

人左轮、右轮与前轮，`up_link` 映射到激光传感器。图3.18中椭圆部分表示四个关节，与图3.19中四个坐标系指向 `base_link` 坐标系的箭头一一对应，描述了其他四个环节相对机器人主体的坐标偏移与姿态变化。五个环节构成一个整体，随着 `base_link` 进行整体位移。在完成对机器人结构的基本描述后，依据扫地机器人的实际形状，在

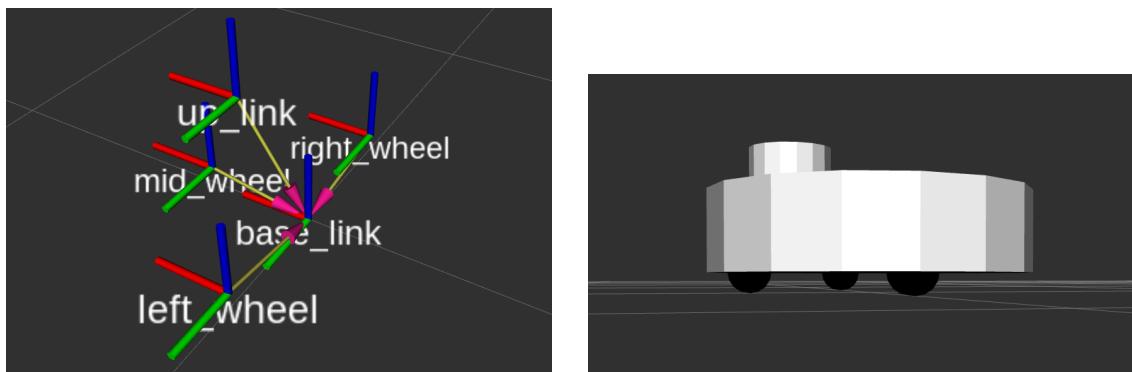


图 3.19 机器人模型结构图

图 3.20 完整机器人模型

URDF 文件中对各个环节的具体形状进行补充描述，本文使用圆柱体描述五个环节的形状，最终构成完整的机器人模型，如图3.20所示。

在实际的定位过程中，rviz 中的机器人模型根据发布的位姿数据进行移动，位姿数据描述了 base_link 环节代表的坐标系相对参考坐标系的平移与旋转，位姿数据的不断发布驱使机器人模型在 rviz 的三维世界中运动。

3.5 本章小结

本章介绍了移动机器人跟踪定位系统的设计，概述了系统的四个模块，然后针对各个模块的设计展开描述：初始化模块中采用 yaml 文件管理本地数据，构造系统对象管理整个系统流程；特征检测模块中设计基于相关滤波的跟踪器与基于角点检测的检测器，实现对图像特征的提取；机器人定位模块中设计了相机外参标定方法与机器人位姿估计方法，求解相机与机器人的位姿；可视化模块中借助 ROS 平台与 rviz 工具对轨迹路径和机器人模型进行可视化。

第四章 基于多相机的机器人定位优化方法设计

本章在移动机器人跟踪定位系统的基础上，分析单相机定位在一些特殊场景下的局限性，从这些局限性出发，设计基于多相机的定位优化方法，说明该方法中的场景设置方案、坐标系统一方法、多相机协同运作机制、位姿估计结果融合过程。同时以多相机定位场景为基础，设计基于图优化的定位精度优化策略。

4.1 基于单相机的机器人定位局限性分析

室内环境虽然具有较多的视觉特征信息，但是容易受到人为干预而改变了室内布局，这对固定位置进行定位的相机产生了巨大的负面影响。移动机器人在室内完成指定工作时有着较为复杂的移动模式，它在图像中的变化使得定位系统难以把握定位的精度。无论是定位环境还是定位目标，都对基于单相机的机器人定位造成了一定的局限性。

遮挡问题是视觉定位技术研究方向上的一大挑战。虽然在上一章提到使用四月码统一机器人的特征，代替机器人在环境中被定位，但是当四月码被异物遮挡时，图像中原本存在的四月码特征将被异物所取代。在基于单相机的定位中，跟踪器在遮挡发生的图像帧中所采集的测试样本并不包含上一帧目标区域中的相关特征，这使得分类器将这些测试样本全部归类为负样本，丢失了对目标区域的跟踪，而检测器本身基于跟踪器所提供的目标区域进行角点检测，目标区域丢失，即使是基于图像全域的四月码检测也因为异物的遮挡无法得到四边形区域。图4.1和图4.2展示了两种遮挡问题发生的情况。图4.1中虽然异物在图像中所占像素不多，但是却遮挡了包

待添加

图 4.1 异物接近四月码的遮挡情况

待添加

图 4.2 异物接近相机镜头的遮挡情况

含四月码在内的关键特征，即使在其他位置进行拍摄，也无法获取四月码外围角点

的位置信息。图4.2中虽然异物占据了图像中的大部分像素，但是异物相对四月码的物理距离较远，通过其他位置的拍摄能够捕捉到完整的四月码。室内环境中人为干预所产生的遮挡通常属于图4.2中的情况，基于固定位置的单相机机器人定位显然无法解决这种遮挡问题。

机器人在相机视野中的深度是估计机器人位姿的重要参数，公式2-4中，在已知深度 Z_c 的情况下可由像素点坐标推算该像素点坐标对应的 3D 点坐标，实现点的定位。因此，深度信息的准确计算对于机器人定位具有重要意义。在单目视觉 SLAM 中，仅凭单张图像无法获得图中像素点的深度信息，对 3D 点的深度计算需要通过三角测量进行，相机通过平移与旋转形成对同一个点的两个观察角度，假设 3D 点在位置 1 相机中的深度为 s_1 ，在位置 2 相机中的深度为 s_2 ，则有：

$$s_1 P_1 = s_2 R P_2 + t \quad (4-1)$$

式中的 P_1 与 P_2 表示两个特征点的归一化坐标， R 与 t 象征两个位置之间的平移旋转变换。 s_1 与 s_2 是求解的目标，通过消元分别求解这两个深度，如公式4-2所示。

$$\begin{aligned} s_1 P_1^\wedge P_1 &= s_2 P_1^\wedge R P_2 + P_1^\wedge t \\ 0 &= s_2 P_1^\wedge R P_2 + P_1^\wedge t \end{aligned} \quad (4-2)$$

式中利用向量叉乘的反对称矩阵消去 s_1 ，先进行 s_2 的求解，之后再对 s_1 求解。移动机器人定位系统使用单目工业相机，并固定相机的位置，当参与定位的相机仅有一个时，无法产生第二个视角进行三角测量，这使得基于单相机的机器人定位系统在定位中缺少深度信息，很难感知相机视野中机器人的尺度变化。当机器人带着四月码进行如图4.3所示的移动时，基于固定位置的单相机机器人定位将引入巨大的定位误差。

待添加

待添加

图 4.3 机器人大尺度变化移动示意图

图 4.4 图像中尺度过小的四月码

基于角点检测的检测器通过对封闭四边形的检测寻找图像中的四月码，但是为了尽量缩短线段检测所消耗的时间，检测器中对图像进行了降采样处理，降低了图

像的分辨率，许多小规模的四边形被过滤。当四月码随着机器人运动到距离相机很远的位置时，其在图像中的呈现如图4.4所示，检测器将它视为小规模的四边形而放弃对它的检测与验证。即使检测器能够检测到该四月码，四月码四个外围角点因为尺度问题聚集在一起，Epnp方法对于图像中过于集中的特征点会产生较大的估计误差。上述问题的存在使得基于单相机的机器人定位系统无法定位距离相机较远的机器人。

无论是遮挡问题、机器人在相机视野中的尺度变化还是对小规模四月码的定位，基于单相机的机器人定位系统都无法解决，在室内环境的机器人定位工作中存在一定的局限性，根据这些分析引入基于多相机的机器人定位方法，规避以上问题带来的异常定位。

4.2 基于多相机的机器人定位方法

由于单目相机在环境感知的过程中采集到的图像缺乏深度信息，所以在SLAM领域，许多研究驱使单目相机在地图中进行位移，通过在不同时刻不同位置采集图像弥补缺失的深度信息，在对机器人进行定位时，这种方法带来了巨大的计算消耗。本文围绕上一节单相机定位的局限性，提出基于多相机的机器人定位方法，进一步减少了机器人异常定位，也摆脱了SLAM需要额外空间缓存图片的限制。

4.2.1 定位场景的布局设置

在单相机定位机器人的场景中，由于参与定位的单目工业相机只有一台，没有特定的布局，更多地是对相机拍摄角度、相机自身参数的调整，而多相机定位机器人的场景中，为突破单相机定位的局限性，需要对定位场景的布局进行设置，规定多台相机的摆放方案，确保定位范围的最大化以及异常定位的最小化。

本文以两台相机和四台相机的情况为例，对两种情况下的场景布局方案进行说明，图4.5与图4.6分别展示了俯视视角下两台相机的布局与四台相机的布局。

图4.5中的蓝色区域表示1号相机对四月码有效检测的视野范围，橙色区域表示2号相机对四月码正常检测的视野范围，通过将两台相机正对设置，使两台相机的正常检测视野范围尽可能地重叠，四月码在重叠区域中可同时被两台相机有效检测，最终通过定位结果融合解决一台单目相机在定位时缺乏深度信息的问题。此外，当机器人带着四月码离开了1号相机的有效检测范围，使四月码在1号相机中的规模较小时，2号相机的设置可以使系统在一定范围内准确检测该四月码并估计其位姿。

图4.5中的两台相机布局虽然解决了深度信息缺失和四月码过小的问题，但它的定位范围依然比较小，布局中存在较多定位盲区，图4.6中四台相机的布局扩大了定

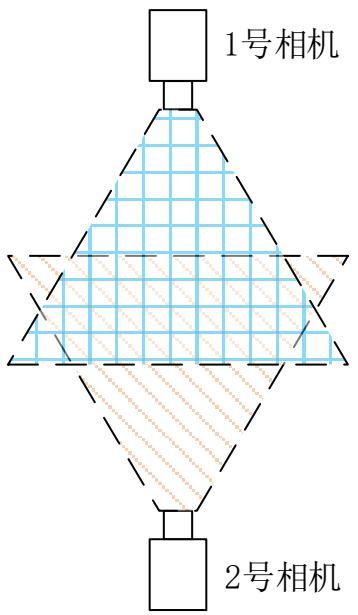


图 4.5 两台相机的布局图

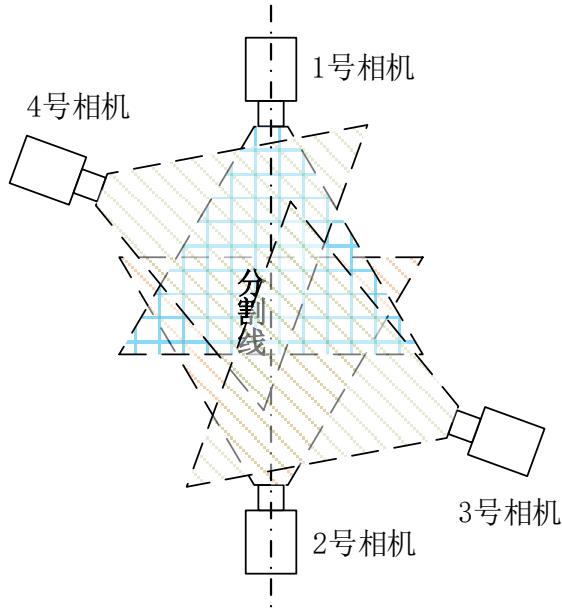


图 4.6 四台相机的布局图

位范围。原来的两台相机依旧保持正对并保证有效检测范围的大面积重叠，新增的两台相机分别设置于分割线的两侧，对 1 号相机与 2 号相机的视野盲区进行补充定位，它们之间不需要保证有效检测区域的大面积重叠，只需保证四台相机的有效检测区域有重叠部分。当机器人带着四月码离开了原来两台相机的横向定位范围，则新增的两台相机可以弥补它们的盲区，进行辅助的四月码检测与机器人定位。

4.2.2 坐标系统一

多台相机的引入使得多个固定的相机坐标系被添加至整个空间中，在多台相机定位的过程中，这些相机的位姿也是定位机器人的重要参数。在单相机定位的场景中，系统通过四月码标记物设置基准坐标系，利用标记物上的已知 3D 点与图像平面上的标记物角点建立 3D-2D 的点对关系计算相机坐标系与基准坐标系的转换关系，从而得到相机在基准坐标系下的位姿。而多相机定位的场景中，结合特殊布局，基准坐标系的设置也有一定的要求。为了使多台相机都能建立与基准坐标系的关系，代表基准坐标系的四月码应当设置在多台相机的共同有效视域中，即上一节提到的重叠区域，如图4.7所示。

为了求解各个相机在基准坐标系下的位姿，进行良好的坐标系统一，本文利用了标记物平面与图像成像平面之间的单应变换对各个相机进行位姿估计。单应矩阵将两个平面上的点进行关联，如图4-3所示，它们之间的关系可表示为：

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = s \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad (4-3)$$

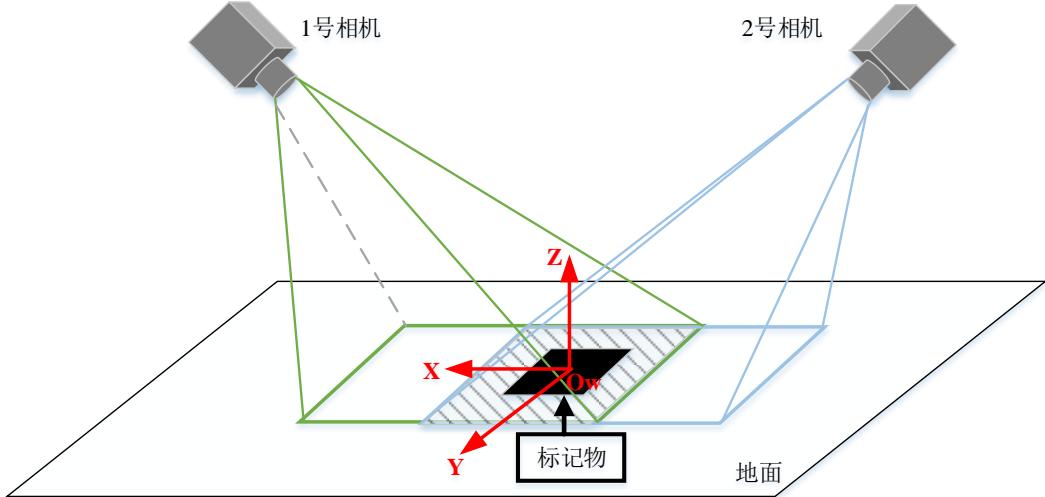


图 4.7 多相机场景中的基准坐标系设置

式中的 3×3 矩阵即为待求的单应矩阵。因为是两个平面之间的变换，没有固定的尺度，可以进行任意的缩放，因此根据公式4-3可得：

$$\begin{aligned} x &= \frac{h_{11}X + h_{12}Y + h_{13}}{h_{31}X + h_{32}Y + h_{33}} \\ y &= \frac{h_{21}X + h_{22}Y + h_{23}}{h_{31}X + h_{32}Y + h_{33}} \end{aligned} \quad (4-4)$$

对公式4-4进行整理得到：

$$\begin{aligned} Xh_{11} + Yh_{12} + h_{13} - xXh_{31} - xYh_{32} - xh_{33} &= 0 \\ Xh_{21} + Yh_{22} + h_{23} - yXh_{31} - yYh_{32} - yh_{33} &= 0 \end{aligned} \quad (4-5)$$

将公式4-5整理成矩阵相乘的形式，分离出已知的点对坐标与未知的单应矩阵元素，如公式4-6所示：

$$\begin{bmatrix} X_1 & Y_1 & 1 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 & -x_1 \\ 0 & 0 & 0 & X_1 & Y_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1 \\ \dots & & & & & & & & \end{bmatrix} \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \\ h_{33} \end{bmatrix} = 0 \quad (4-6)$$

虽然式中有 9 个未知数需要求解，但是约束 $\|H\| = 1$ 使得单应矩阵只有 8 个自由度，通过四组点对即可完成对单应矩阵的求解，而四月码四个外围角点也恰好可以作为点对数据组成已知量矩阵 A ，公式4-6简化为公式4-7。

$$Ah = 0 \quad (4-7)$$

这种形式的表达式可通过对 A 矩阵进行 SVD 分解从而求得单应矩阵的元素。公式4-3除了表示两个平面之间点对的关联关系，也表示了一个 3D 点到 2D 点的转换关系，因此单应矩阵是由相机内参矩阵与相机外参矩阵组合而成，如公式4-8所示。

$$[\mathbf{h}_1 \ \mathbf{h}_2 \ \mathbf{h}_3] = K [\mathbf{r}_1 \ \mathbf{r}_2 \ \mathbf{t}] \quad (4-8)$$

式中的 \mathbf{h}_i 表示 $[h_{i1} \ h_{i2} \ h_{i3}]^T$ 。在已知相机内参的情况下能够求解相机外参，而相机外参也表示了相机在基准坐标系下的位姿，如公式4-9所示。

$$\begin{aligned} \mathbf{r}_1 &= \lambda K^{-1} \mathbf{h}_1 & \mathbf{r}_2 &= \lambda K^{-1} \mathbf{h}_2 \\ \mathbf{t} &= \lambda K^{-1} \mathbf{h}_3 & \mathbf{r}_3 &= \mathbf{r}_1 \times \mathbf{r}_2 \\ \lambda &= \frac{1}{\|K^{-1} \mathbf{h}_1\|} = \frac{1}{\|K^{-1} \mathbf{h}_2\|} \end{aligned} \quad (4-9)$$

\mathbf{r}_3 根据旋转矩阵的性质由 \mathbf{r}_1 与 \mathbf{r}_2 叉乘所得，它们共同构成了相机位姿的旋转部分， λ 代表尺度参数，同样根据旋转矩阵的约束求得。

在完成了多相机场景的布局并设置好代表基准坐标系的四月码之后，系统结合每台相机采集的四月码图像，采用上述的相机位姿估计方法，计算基准坐标系到各个相机坐标系的转换关系。在多相机定位过程中，每台相机都能得到机器人在各自相机坐标系下的坐标，此时通过各台相机的对应位姿，将这些坐标全部转换到设置的基准坐标系中，实现坐标系的统一。

4.2.3 多相机协同运作机制

多相机协同运作机制引入多个规格相同的单目工业相机，在布局完成的场景中实现多台相机同时跟踪定位机器人的功能。该机制借助多线程实现，每一个线程承担一台相机从采集图像开始到定位结束之间的执行任务，通过这种形式既保证每一台相机的定位流程相对独立，又实现了多相机共同定位的过程。

由于跟踪器在初始化时需要人为输入感兴趣区域（Region of Interest, ROI），本文将每台相机代表的流程中均完成 ROI 的输入作为多相机协同运作开始的信号，这也将线程执行的任务分为两类，如图4.8(a)与图4.8(b)所示。

图4.8(a)展示的是等待 ROI 输入完毕并作第一帧图像定位的任务流程，在这个任务流程中，与各台相机所采集图像对应的 ROI 的输入是启动定位的关键，具体实现上采用了全局标志位 $global_flag$ 作为收集 ROI 输入情况的容器，其计算的过程如下所示：

$$global_flag = f_1 \& f_2 \& \dots \& f_n \quad (4-10)$$

f_i 表示第 i 号相机对应的流程中 ROI 的输入情况，0 表示未输入，1 表示已输入，只有当全部的 ROI 都输入的情况下，该全局标志位才能置 1 从而开始多相机协同定位

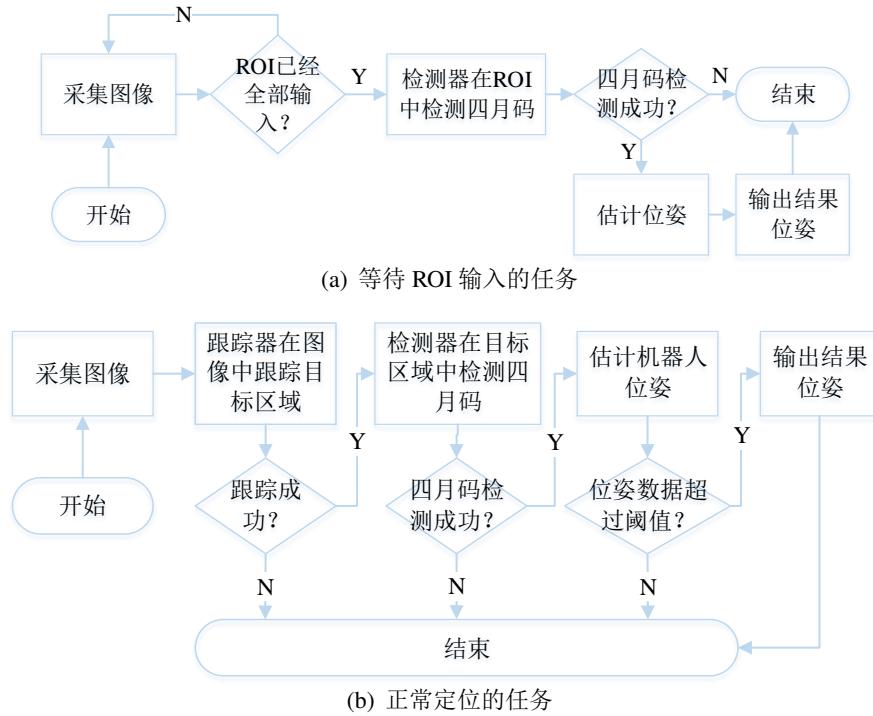


图 4.8 线程执行任务流程图

流程。图4.8(b)展示的是开始多相机协同运作之后的正常定位流程，在采集到图像之后，线程驱使跟踪器、检测器依次工作，提取图像中的目标区域与关键角点特征，最后对机器人位姿进行估计。

在多相机协同定位开始的第一帧图像中，检测器对 ROI 中的四月码进行检测，由于机器人初始设置的位置对于一些相机来说并不能很好地被观测，部分线程负责的流程中无法在 ROI 中检测到四月码。此外，在正常定位的任务流程中也会发生跟踪器跟踪失败、检测器无法检测到四月码的情况，这些都是机器人离开了相机的有效观测范围引起的。为了使相机代表的定位流程在机器人重新进入该相机有效观测范围时能够正常进行，本文设计了重跟踪算法，具体过程如 **Algorithm 3** 所示。

重跟踪算法充分体现了多相机工作的协同性。当机器人带着四月码离开了相机的有效观测范围，该流程转入重跟踪算法，执行一个时间间隔为 5 秒的定时任务。在该任务中首先查找当前时间下由其他相机流程估计的机器人位置，然后将代表该位置的 3D 点 P 通过相机外参转化至相机坐标系下的坐标 $P_c(X_c, Y_c, Z_c)$ ：

$$P_c = RP + t \quad (4-11)$$

结合公式2-4得到像素点的计算公式：

$$u = \frac{f_x X_c}{Z_c} + c_x \quad v = \frac{f_y Y_c}{Z_c} + c_y \quad (4-12)$$

Algorithm 3 重跟踪算法

Input: 定时计数 $time_count$, 相机焦距 f_x, f_y , 相机主点偏移 c_x, c_y , 相机坐标系下坐标 X_c, Y_c, Z_c , 图像 IMG

Output: 机器人是否重新进入相机有效观测范围的判断

```

1:  $u \leftarrow -1$ 
2:  $v \leftarrow -1$ 
3: if  $time\_count < 5$  then
4:   return false
5: else
6:    $time\_count \leftarrow 0$ 
7:    $u \leftarrow \frac{f_x X_c}{Z_c} + c_x$ 
8:    $v \leftarrow \frac{f_y Y_c}{Z_c} + c_y$ 
9:   if  $u \geq 0$  and  $u < img\_width$  then          ▷  $img\_width$ 表示图像宽度
10:    if  $v \geq 0$  and  $v < img\_height$  then         ▷  $img\_height$ 表示图像高度
11:      if 四月码在 $IMG$ 中被成功检测 is true then
12:        重构目标区域
13:        return true
14:      end if
15:    end if
16:  end if
17:  return false
18: end if

```

将像素点坐标 $p(u, v)$ 与图像边界进行比较, 判断点 P 是否成功重投影至该相机的像素平面上, 如公式4-13所示。

$$0 \leq u < img_width \quad 0 \leq v < img_height \quad (4-13)$$

若 u 与 v 的值满足公式4-13中的两个条件, 则说明三维点的重投影成功, 接着检测器以整个图像为目标区域对四月码进行检测, 最后以图中四月码的位置信息为基础重构跟踪器所需的目标区域。目标区域的重构如图4.9所示, 四月码的位置信息构成一个外接矩形, 本文将宽与高均比外接矩形长 200 像素的矩形区域作为重构的目标区域, 一方面为跟踪器中分类器的训练提供了更多的负样本, 另一方面严格控制目标区域的大小, 保证检测器执行四月码检测时的速度。

常规的多线程方法在需要使用线程时创建线程, 在线程结束时销毁线程, 这两个动作存在着一定的开销, 当创建与销毁过于频繁时会造成巨大的资源消耗。系统中每个线程的任务执行都从图像采集开始, 而图像采集在整个系统中是执行最为频繁的操作, 常规的多线程方法会因此产生足以影响系统性能的额外开销。线程池的引入很好地解决了这个问题, 在线程池初始化时在线程池中创建数量与相机数目相同的线程。这些线程的生命周期与线程池同步, 不会在线程池工作期间被销毁, 任务队列中没有任务时, 空闲线程并不占用计算资源, 而当任务队列中有任务排队时,

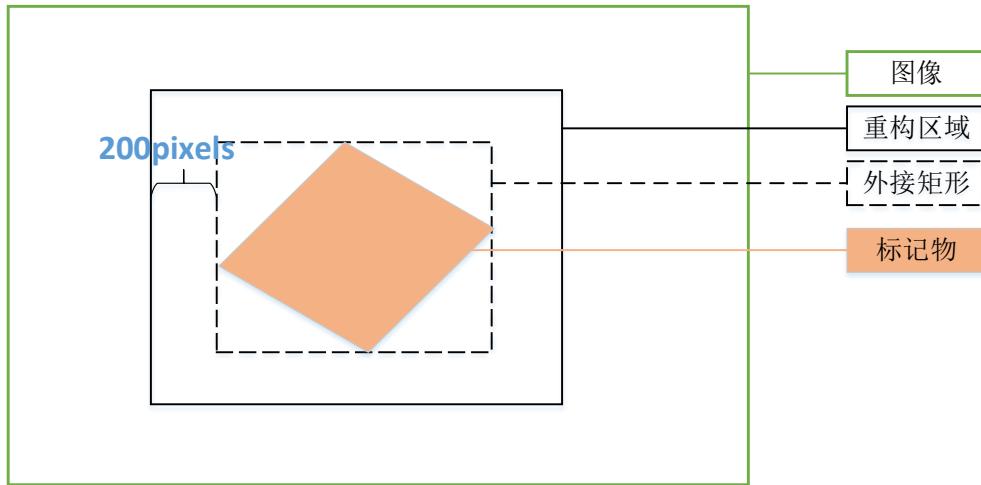


图 4.9 目标区域重构示意图

线程调度器会调度空闲线程去响应待处理的任务。

4.2.4 定位结果融合

多相机协同运作机制中的多线程任务在同一时刻产生了多个定位结果，这些结果中包含了各台相机从不同的角度观察机器人与四月码的深度信息。针对同一时刻的多个线程执行任务产生的定位结果，将它们进行融合得到更为精确的机器人位置，如图4.10所示，但是每个线程中采集的图像因为拍摄角度的不同而包含不同数量的

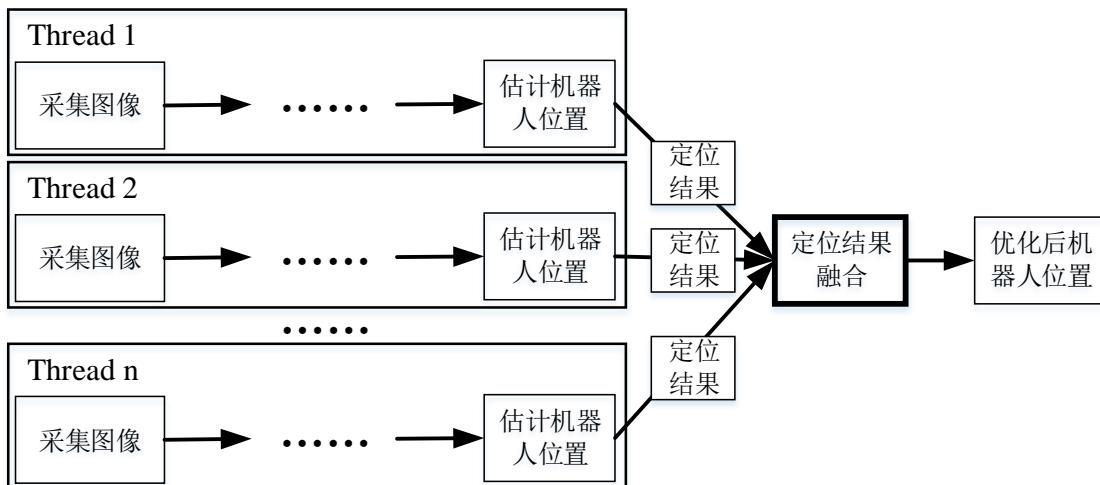


图 4.10 多线程定位任务的结果融合示意图

特征，不同线程中的跟踪器与检测器执行的时间因此存在差异，这产生了无法忽视的时间差问题，在结果融合中是需要解决的关键问题。

本文结合多相机布局与机器人移动方式设计了一种时间同步的算法，解决了时

间差问题，同时实现了定位结果的融合。算法采用了时间分段的策略，以多线程下各任务中的 ROI 输入完毕的时刻作为起始时刻，记作 t_{start} ，之后将定位任务中图像采集完毕的时刻作为当前时刻，记作 t_{cur} ，根据定位时机器人的移动速度选择一个合适的基准时间间隔和时间阈值，分别记作 \bar{t} 和 \tilde{t} ，基准时间间隔应当足够小，在这个间隔的时间内机器人没有发生大幅度的移动。它们之间的关系如图4.11所示，整个定

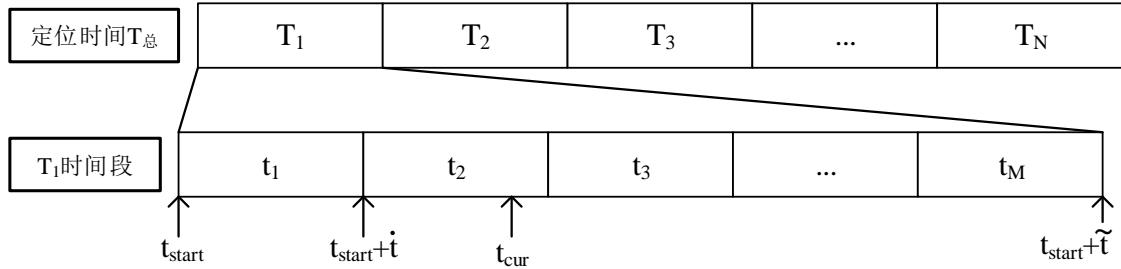


图 4.11 时间分段策略示意图

位流程消耗的时间以时间阈值 \tilde{t} 为基准等分为 N 份，而其中的每一份都以基准时间间隔 \bar{t} 为基准等分为 M 份，即

$$N = T_{\text{总}}/\tilde{t} \quad M = \tilde{t}/\bar{t} \quad (4-14)$$

围绕上述时间分段的策略设置两个相同的结构体协助进行时间同步，其组成如图4.12所示，该结构体与图4.11中的 T_1 时间段对应，头部中的 `start_time` 记录一个开始时间，

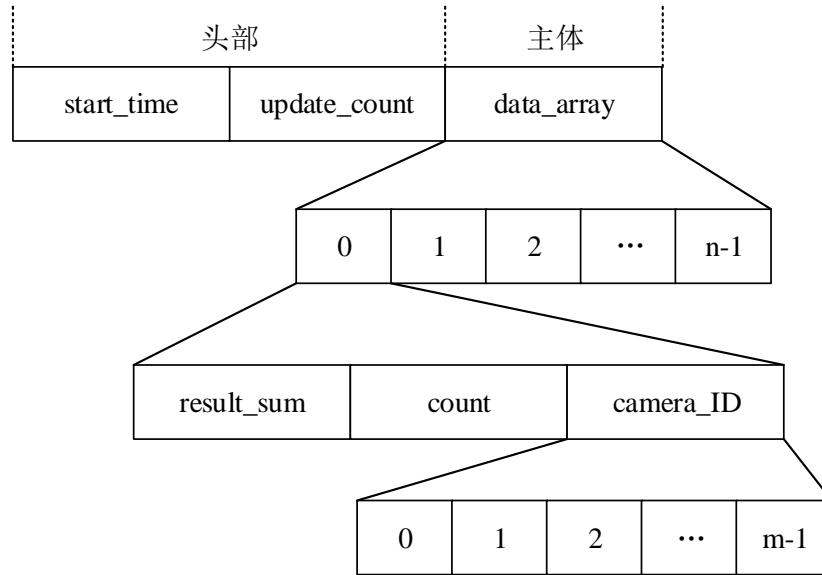


图 4.12 时间同步结构体组成示意图

标志着该结构体中统计的定位结果均在该时间之后被计算得到，`update_count` 记录了这个结构体中总的数据更新次数，主体是一个数据数组，数组大小 n 与公式4-14中的 M 对应，数组元素由累加的结果 `result_sum`、累加的次数 `count` 和参与融合的相机

编号数组组成。当当前时间 t_{cur} 被记录时，算法通过公式4-15计算出当前时间对应的数组下标*i*，

$$i = \lfloor \frac{t_{cur} - t_{start}}{\tilde{t}} \rfloor \quad (4-15)$$

并在对应下标的元素中融合 t_{cur} 时刻图像中得到的定位结果，图4.11中 t_{cur} 落在 t_2 段中，应当与该段包含的定位结果进行融合。以 t_{cur} 时刻采集的图像为样本能够得到合格的定位结果，将其记作 $result$ ，得到该结果的相机编号记作 id ，则融合的过程如**Algorithm 4**所示。随着时间的推移，当前时刻 t_{cur} 与开始时刻 t_{start} 的差值必然会超

Algorithm 4 数据融合

Input: 结构体中的数据数组 A ，数组大小 n ，当前时刻 cur_time ，待融合定位结果 $result$ ，参与融合的相机编号 id

```

1: function DATA-FUSION( $cur\_time, result, id$ )
2:    $i \leftarrow (cur\_time - start\_time)/\tilde{t} + 1$ 
3:   if  $i > n$  then
4:     return
5:   else
6:      $update\_count \leftarrow update\_count + 1$             $\triangleright update\_count$ 指 $A$ 的总更新次数
7:      $A[i].result\_sum \leftarrow A[i].result\_sum + result$ 
8:      $A[i].count \leftarrow A[i].count + 1$ 
9:      $A[i].camera\_ID[id] \leftarrow 1$             $\triangleright$  未参与该时间段数据融合的相机编号值为0
10:   end if
11: end function

```

过时间阈值 \tilde{t} ，到达下一个大时间段，此时算法将这个时刻待融合的定位结果暂存在第二个结构体中，该结构体中的 $start_time$ 是前一个结构体中 $start_time$ 与时间阈值之和，以同样的规则对这一结构体进行更新。若第一个结构体超过一段时间没有更新，则执行数据数组中的融合结果计算，数据数组中的每一个元素记录了累加结果与累加次数，通过均值计算对每个时间段的最终融合结果进行求解，如公式4-16所示：

$$\mathbf{P}_{fusion} = \frac{result_sum}{count} \quad (4-16)$$

对于数据数组中下标为*i*的元素求解得到的最终融合结果 \mathbf{P}_{fusion}^i ，以时刻 $t_{start} + (i + 1) * \tilde{t}$ 表示产生该结果的时间，如公式4-17所示。

$$t_{start} + (i + 1) * \tilde{t} \iff \mathbf{P}_{fusion}^i \quad (4-17)$$

完成了第一个结构体的定位结果融合与整理后，将最终的时刻、定位结果以及相关相机编号记录于文件中，并清空第一个结构体中的内容，使其发挥与之前第二个结构体相同的作用，暂存定位结果，防止溢出数据的丢失。

时间同步算法不仅融合了多台相机在同一微小时间段内的定位结果，还融合了一台相机在一个微小时间段中采集多幅图像产生的定位结果，实现了横向与纵向上的结果融合，对繁杂的定位结果进行了整理。

4.3 基于多相机布局的精度优化策略

4.3.1 多相机布局的图构建

多相机的机器人定位方法构建了多相机布局，以多相机协同工作的方式对机器人进行定位，最终的机器人定位结果通过结果融合方法缩小了定位误差。该优化方法主要针对因机器人深度变化而引起的定位误差，它无法处理场景噪声所产生的误差，噪声的存在导致了定位过程中误差的积累，是除了深度变化以外对定位结果影响较大的另一因素。

噪声带来的累积误差在定位过程中难以被处理，因此可以在定位结束后针对定位结果进行累积误差的优化。本文引入图结构对累积误差进行后端优化，多相机的布局以及待优化的3D点坐标共同构成了图中的元素。本文使用的图结构由图节点与二元边构成，如图4.13所示，一条二元边的两端节点分别被定义为3D点位置 \mathbf{P} 和参



图 4.13 图节点与边的定义

与定位该3D点的某一相机的相机位姿 $T(R, t)$ ，二元边被定义为3D点到该相机成像平面的投影过程，该投影过程如公式4-18所示。

$$\mathbf{p}' = K(\mathbf{R}\mathbf{P} + \mathbf{t}) \quad (4-18)$$

检测器本身在相机采集的图像中能够检测到该3D点对应的像素点 \mathbf{p} ，将检测所得的像素点作为观测值，重投影产生的像素点作为估计值，噪声的存在使估计值与观测值不会重合，它们之间的偏差成为衡量优化程度的参数，如公式4-19所示：

$$\mathbf{e} = \mathbf{p} - \mathbf{p}' = \mathbf{p} - K(\mathbf{R}\mathbf{P} + \mathbf{t}) \quad (4-19)$$

\mathbf{e} 是一个三维向量，不能非常直观地评估估计值与观测值之间的误差，所以本文采取它的平方形式 E 作为误差评估的标量，如公式4-20所示：

$$E = \mathbf{e}^T \mathbf{e} \quad (4-20)$$

这个标量越接近0，就说明重投影的像素点越接近检测得到的像素点。

基于以上定义，以四台相机的布局为例，相机与3D点构建的图结构如图4.14所示。图中方形图节点表示相机位姿，圆形图节点表示3D点位置，一个圆形图节点最多被四个方形图节点连接，这表示这个3D点处于四台相机的共同有效观测范围，由四台相机的定位结果融合而成。由于未被四台相机观测到的3D点不会作为图节点出现在图中，所以圆形图节点至少有一个方形图节点与其连接。

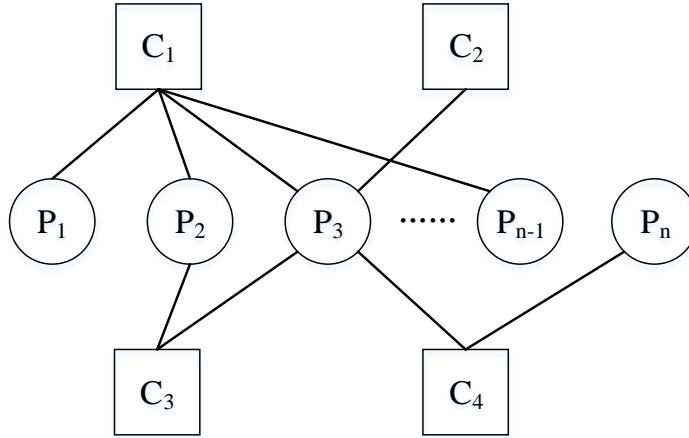


图 4.14 四台相机布局构建的图结构

4.3.2 基于图的精度优化

多相机的布局与待优化的机器人定位结果完成了图的构建，图中的节点与边得到了完整的定义，围绕构建的图展开对定位结果的优化。图中一条二元边代表的误差评估标量为 E ，将所有二元边代表的误差评估标量累加，形成一个总体的优化求解问题，如公式4-21所示。

$$f = \sum_{k=1}^n E_k = \sum_{k=1}^n \mathbf{e}_k^T \mathbf{e}_k \quad (4-21)$$

在这个优化问题中，待优化的变量是代表机器人位置的3D点坐标 \mathbf{P} ，将这个变量加入公式4-21得到：

$$f = \sum_{k=1}^n E_k(\mathbf{P}_k) = \sum_{k=1}^n \mathbf{e}_k(\mathbf{P}_k)^T \mathbf{e}_k(\mathbf{P}_k) \quad (4-22)$$

\mathbf{P}_k 表示第 k 条二元边对应的3D点坐标，寻找每条边对应的优化变量使得 f 的取值最小。

针对第 k 条边展开对优化变量 \mathbf{P}'_k 的求解，为初始值 $\tilde{\mathbf{P}}_k$ 提供一个微小的增量 $\Delta\mathbf{P}$ ，在这个增量的作用下， E_k 会变小，其形式也转变为：

$$E_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P}) = \mathbf{e}_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P})^T \mathbf{e}_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P}) \quad (4-23)$$

对优化变量的求解划分为对增量 $\Delta\mathbf{P}$ 的迭代求解。公式4-23中误差项由 $\mathbf{e}_k(\tilde{\mathbf{P}}_k)$ 变成了 $\mathbf{e}_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P})$ ，对误差项进行一阶展开：

$$\begin{aligned} \mathbf{e}_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P}) &\approx \mathbf{e}_k(\tilde{\mathbf{P}}_k) + \frac{d\mathbf{e}_k}{d\mathbf{P}} \Delta\mathbf{P} \\ &= \mathbf{e}_k(\tilde{\mathbf{P}}_k) + J_k \Delta\mathbf{P} \end{aligned} \quad (4-24)$$

公式4-24中的 J_k 表示误差项关于3D点坐标的导数，被称为雅可比矩阵。在一阶展开

后，误差评估标量也能够进一步展开：

$$\begin{aligned}
 E_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P}) &= \mathbf{e}_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P})^T \mathbf{e}_k(\tilde{\mathbf{P}}_k + \Delta\mathbf{P}) \\
 &\approx (\mathbf{e}_k(\tilde{\mathbf{P}}_k) + J_k \Delta\mathbf{P})^T (\mathbf{e}_k(\tilde{\mathbf{P}}_k) + J_k \Delta\mathbf{P}) \\
 &= \mathbf{e}_k(\tilde{\mathbf{P}}_k)^T \mathbf{e}_k(\tilde{\mathbf{P}}_k) + 2\mathbf{e}_k(\tilde{\mathbf{P}}_k)^T J_k \Delta\mathbf{P} + \Delta\mathbf{P}^T J_k^T J_k \Delta\mathbf{P}
 \end{aligned} \tag{4-25}$$

从公式4-25中可知增量 $\Delta\mathbf{P}$ 使得误差评估标量也产生了增量 ΔE_k ，表示形式如公式4-26所示：

$$\Delta E_k = 2\mathbf{e}_k(\tilde{\mathbf{P}}_k)^T J_k \Delta\mathbf{P} + \Delta\mathbf{P}^T J_k^T J_k \Delta\mathbf{P} \tag{4-26}$$

为了使误差评估标量变小，寻找一个特定增量 $\Delta\mathbf{P}$ 使得 ΔE_k 取得极小值，为此令 ΔE_k 对 $\Delta\mathbf{P}$ 的导数为0，以求解本次迭代的增量 $\Delta\mathbf{P}$ ，如公式4-27所示。

$$\frac{d\Delta E_k}{d\Delta\mathbf{P}} = 2\mathbf{e}_k(\tilde{\mathbf{P}}_k)^T J_k + 2J_k^T J_k \Delta\mathbf{P} = 0 \tag{4-27}$$

根据公式4-27得到关于增量 $\Delta\mathbf{P}$ 的等式，如下所示：

$$\begin{aligned}
 J_k^T J_k \Delta\mathbf{P} &= -\mathbf{e}_k(\tilde{\mathbf{P}}_k)^T J_k \\
 H_k \Delta\mathbf{P} &= -b_k
 \end{aligned} \tag{4-28}$$

式中 H_k 是海塞矩阵， b_k 表示了等号右侧的已知量。公式4-28表征了第 k 条边的优化求解，将所有边累加后形成一个整体求解的形式，如公式4-29所示。

$$H \Delta\mathbf{P} = -b \tag{4-29}$$

最终整体优化问题通过公式4-29进行线性求解，每次迭代得到的 $\Delta\mathbf{P}$ 用于修正当前此迭代的初始值，并将修正后的3D点坐标作为下次迭代的初始值，重复上述过程直到迭代结束，此时最终被修正的3D点坐标即为该优化问题的最终解，也是优化后使重投影误差最小的最优解，如公式4-30所示。

$$\mathbf{P}'_k = \mathbf{P}_k + \Delta\mathbf{P}^{(1)} + \Delta\mathbf{P}^{(2)} + \cdots + \Delta\mathbf{P}^{(m)} \tag{4-30}$$

m 表示进行优化问题求解的最终迭代次数， $\Delta\mathbf{P}^{(l)}$ 表示第 l 次迭代得到的增量。

多台相机定位的场景与SLAM中一台相机通过平移旋转对3D点进行定位的场景非常类似，因此多台相机的布局适合构建图结构，而图的构建使得原本只能逐一优化的重投影误差能够组合成一个整体，通过对整体优化问题的求解达到最小化每个3D点的重投影误差的目的，这在定位系统的后端精度优化处理中产生了较好的效果。

4.4 本章小结

本章首先对单相机场景下的机器人定位中存在的局限性进行分析，然后从局限性出发详细说明了多相机场景下的机器人定位优化方法，围绕多台相机的布局、坐标系的统一、多相机协同运作机制、定位结果的融合展开描述，最后设计了以多相机场景为基础的图构建方法与基于图的精度优化方法。

第五章 实验测试与分析评估

本章设计了四组对照实验，从性能与精度上对定位系统进行测试与评估...

5.1 实验测试平台与测试环境说明

本节对测试平台、辅助测试工具、实际测试环境以及实验测试评估指标进行说明。

5.1.1 PC端测试平台

PC 端测试平台承担了所有系统测试与算法测试，其使用 Ubuntu 16.04.4 系统，内核版本为 XXX，采用 ROS 的 kinetic 版本辅助数据转发，此外借助 rviz 工具作结果可视化。

定位系统涉及大量的图像处理算法，对测试平台的处理能力有一定的要求。PC 端测试平台的 CPU 为英特尔酷睿 I5-8250 四核心八线程处理器，主频为 1.6GHz，内存为 8G。

5.1.2 辅助测试工具说明

实验测试依托 PC 端测试平台，但仍需要其他硬件工具辅助测试。这些辅助测试工具主要有工业相机、测试用三脚架、待测试机器人、标定板和不同尺寸的标记物实体，接下来将分别对它们进行说明。



图 5.1 海康威视工业相机



图 5.2 海康威视相机镜头

工业相机的选型是海康威视的 MV-CA050-11UM，如图5.1所示，是 500 万像素的 CMOS 面阵工业相机，芯片尺寸为 2/3 寸，具有全局快门的特点。相机采集到的

图像是黑白的，图像分辨率为 2448*2048，图像数据通过 USB 3.0 传输至测试平台。工业相机镜头的选型是海康威视的 MVL-MF0824M-5MP，如图5.2所示，该镜头是 500 万像素的定焦镜头，焦距为 8 毫米，大小为 2/3 寸。

测试用三脚架为云腾 VT-6008，如图5.3所示，实验中使用 157cm 的最高高度，其配置目的是固定工业相机，并在实验测试中运用云台改变工业相机的拍摄角度。待测试机器人选用米家的二代扫地机器人，如图5.4所示，扫地机器人内置 Ubuntu 14.04.3 操作系统，内核为 3.4.39 armv7l，同时它内置 wifi 模块，在实验测试中通过该模块给机器人传输相应的移动指令。由于扫地机器人不作为运算平台，对扫地机器人内部的运算硬件不作过多的说明。



图 5.3 测试用三脚架



图 5.4 米家二代扫地机器人

标定板为自制 9*6 的 pvc 材质棋盘格，如图5.5所示。该标定板相比纸质标定板在测试中具有更加平整的表面，有利于标定精度的提升。标记物实体有不同的尺寸，如图5.6所示，大尺寸实体用于参考坐标系的设置，小尺寸实体依附于机器人表面，用于定位。

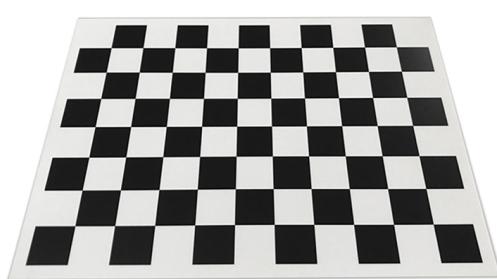


图 5.5 棋盘格标定板

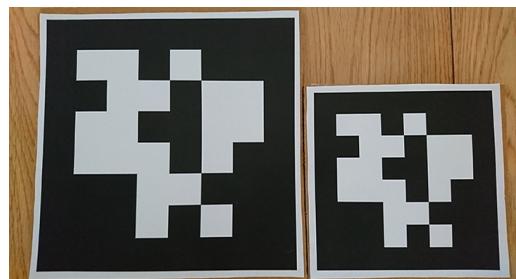


图 5.6 标记物实体

5.1.3 实际测试环境搭建

定位系统实际测试场景如图5.7所示。

待添加

图 5.7 实验室大厅场景

由于工业相机搭建高度的限制，实验室大厅的测试范围约为 2m*3m，该场景白天光线充足，特征信息丰富，与标记物特征的区分度较大，适合定位系统的精度测试，晚上光线较暗，环境特征与标记物特征的区分度变小，适合定位系统的稳定性测试。在两种不同条件的光照下测试能够模拟出室内光照不断改变的场景，对定位系统的应用具有很好的验证性。

5.1.4 实验测试评估指标

为验证定位系统的定位精度，本文将实际场景测量的位置信息作为真实值，将定位系统估计得到的机器人位置信息作为预测值，采用均方根误差（Root Mean Squared Error，RMSE）作为定位精度的评估，评估公式5-1如下：

$$e = \sqrt{\frac{1}{m} \sum_{i=1}^m (y_i - \hat{y}_i)^2} \quad (5-1)$$

式中 e 表示最终估计的均方根误差， m 表示参与精度评估的数据组数， y_i 表示真实值而 \hat{y}_i 表示预测值。[]中提到 RMSE 可以用来衡量预测值同真实值之间的偏差，值越大表示误差越大，估计的位置与实际位置的偏差也越大。

对于定位系统实时性与稳定性两大性能的评估，本文采取对 apriltag 检测所需时间的统计以及对遮挡后恢复所需时间的统计对性能进行评估。

5.2 基于多相机的定位系统功能测试与分析

5.2.1

5.2.2

5.3

5.3.1

5.3.2

5.4 本章小结

第六章 总结与展望

本章对室内移动机器人精准跟踪定位技术的研究与实现进行总结。从移动机器人在室内的应用日趋增多、室内定位技术相对室外定位技术发展前景更好这些现状分析对移动机器人进行室内精准定位的研究意义，从而引出本文的研究内容，设计实现了一套对室内移动机器人的定位系统，并对定位系统的性能以及定位方法的精度进行进一步的优化。最后设计多组对照实验对优化后的定位系统以及定位方法进行测试。

6.1 本文总结

二十世纪中叶到最近几年，工业生产、生活质量在机器人技术的快速发展下优质了许多，人们的生活水平得到了提高，工作方式也发生了变化。移动机器人在机器人中占多数，因其有效扩大工作范围的能力而被广泛地使用于室内和室外场景。移动机器人能够代替人类完成诸多工作，但是其达成目标的前提是知晓自身所在，对移动机器人的定位问题走上台面。室外移动机器人的定位主要依赖于 GPS，技术相对比较成熟，但是由于室内环境的复杂性以及强干扰性，使得 GPS 无法被用于室内，而室内具有较为丰富的视觉信息，因此，通过计算机视觉实现对室内移动机器人的定位具有重大意义。本文基于计算机视觉方法设计实现了一个定位系统，并基于此系统对定位性能和精度进行优化。

在对移动机器人进行室内定位时，许多成熟的计算机视觉方法可以进行辅助。本文对计算机视觉领域的重要背景——四大坐标系的转换进行了说明，然后以此为基础分别对相机标定方法中的张正友标定法，目标跟踪方法中的 KCF 跟踪，标签检测方法中的 apriltag 检测方法以及位姿估计方法中的 epnp 算法进行原理的阐述和优点分析，与此同时描述了这些方法在定位室内移动机器人时的重要站位。

在分析了一个定位系统所需的主要模块后，设计实现了对室内移动机器人的定位系统。本文将定位系统进行模块化，每个模块都有其在流程中的独特作用和设计思想，且模块之间存在数据的交互。定位系统被划分为四个模块，分别是初始化模块、图像处理模块、机器人定位模块、可视化模块。初始化模块负责获取系统的初始数据，包括存储在本地的相机内参等，同时它也承担了对整个系统对象的初始化工作，并计算参与定位相机的外参。图像处理模块封装了跟踪器与检测器，主要负责对图像中机器人的跟踪以及对机器人特征的检测。机器人定位模块实现了链式的位姿估计方法，依赖其他模块提供的数据对机器人进行定位，并保存定位的结果。

可视化模块借助 ROS 平台以及 rviz 工具展示对机器人定位的结果，同时加载机器人模型、相机模型，使整个场景更加直观。在数据交互方面，初始化模块为其他模块提供了相机内参与相机外参以及采集的原始图像数据，图像处理模块通过图像的特征提取为机器人定位模块提供位姿估计所需的视觉信息，机器人定位模块利用初始化模块与图像处理模块提供的数据输出定位结果给可视化模块。

基于实现的定位系统，设计定位性能以及定位精度的优化方法。本文首先针对定位性能进行优化，考虑定位的实时性，从图像处理耗时多这个问题出发，将图像处理模块中跟踪器与检测器的再验证机制转变为先跟踪后检测的优化机制，减少检测所需时间开销，同时考虑定位的稳定性，从环境负面因素影响造成机器人跟丢这个问题出发，设计重跟踪流程，用先检测特征后重构跟踪 ROI 的方式解决机器人跟丢问题。然后针对定位精度进行优化，一方面考虑单相机参与定位的局限性，单相机由于位置固定会产生深度信息缺失的问题，在机器人远离相机时会产生较大的定位误差，引入多相机参与定位的机制，从多个角度补全缺失的深度信息，采用对各相机所得位姿和求均值的方式优化定位精度，另一方面考虑多相机定位的构造场景，多相机固定位置的情形类似于 SLAM 中单相机移动估计位姿的方式，因此采用图优化方法，先对多相机情景进行建图，然后利用最小化重投影误差对相机位姿以及机器人位姿进行进一步优化。

对实现的定位系统设置对照实验，依托扫地机器人在实际场景中进行测试。本文在实验前首先制定机器人固定的移动方案并对实际场景的尺度进行测量。之后首先在实际场景中进行单相机无优化条件的实验，从数据看无论是定位的速度、抗遮挡能力以及精度都有提升空间。然后在实际场景中进行单相机性能优化条件的实验，记录定位在图像处理模块所花的时间，并进行遮挡实验验证优化后的稳定性。接着进行多相机性能优化条件以及基于图优化改进条件的实验，记录最终的定位结果。最后对比了各组实验后的定位结果数据，分析了基于图优化定位方法的优越性。

6.2 未来展望

本文的主要工作是对室内移动机器人精准定位技术的研究，通过实现定位系统，对定位系统的性能进行优化，借助多相机工作机制和图优化提高定位的精度，展示了在室内环境对机器人进行精准定位的可行性。移动机器人在室内环境的应用不断，对移动机器人的室内定位就依然是计算机视觉领域研究的热门，未来的研究可以从以下几个方面进行：

(1) 脱离标签的高速视觉定位。采用“主动”方式进行视觉定位的系统中大多采用了标签统一特征，加速检测识别机器人在图像中的位置，但是标签在很多场景下显得冗余，脱离标签的定位方法是以后研究的主要方向；

(2) 结合其他信号的精准视觉定位。目前的纯视觉定位方法受限于环境光强的影响无法对机器人进行稳定的高精度定位，未来可以加入比如 wifi 信号等其他信息，辅助进行视觉定位。

致谢

本论文是在导师的悉心指导下完成的，从论文的选题到论文的撰写，无不渗透着导师的心血，……值此论文完稿之际，谨对导师的辛勤培育以及谆谆教诲表示最衷心的感谢！

作者简介

1. 基本情况

孙煊溢，男，浙江台州人，1995年10月出生，西安电子科技大学计算机科学与技术学院软件工程专业2017级硕士研究生。

2. 教育背景

2013.09~2017.07，西安电子科技大学，本科，专业：软件工程

2017.09~ ，西安电子科技大学，硕士研究生，专业：软件工程

3. 攻读硕士学位期间的研究成果

3.1 发表学术论文

3.2 申请（授权）专利

3.3 参与科研项目及获奖

[1] XXX项目，项目名称，起止时间，完成情况，作者贡献。

[2] XXX, XXX, XXX等. 科研项目名称. 陕西省科技进步三等奖, 获奖日期.

[3] ...

