

硕士学位论文

基于视觉的多机器人室内协同 SLAM  
算法的研究与实现

**RESEARCH AND IMPLEMENTATION OF  
VISION-BASED MULTI-ROBOT INDOOR  
COOPERATIVE SLAM ALGORITHM**

叶必鹏

哈尔滨工业大学

2018 年 6 月

国内图书分类号：TP242

学习代码：10213

国际图书分类号：621

密级：公开

## 工程硕士学位论文

# 基于视觉的多机器人室内协同 SLAM 算法的研究与实现

硕 士 研 究 生：叶必鹏

导 师：夏红伟 研究员

申 请 学 位：工程硕士

学 科：控制工程

所 在 单 位：航天学院

答 辩 日 期：2018 年 6 月

授予学位单位：哈尔滨工业大学

Classified Index: TP242  
U.D.C: 621

Dissertation for the Master's Degree in Engineering

**RESEARCH AND IMPLEMENTATION OF  
VISION-BASED MULTI-ROBOT INDOOR  
COOPERATIVE SLAM ALGORITHM**

<b>Candidate:</b>	Ye Bipeng
<b>Supervisor:</b>	Prof.Xia Hongwei
<b>Academic Degree Applied for:</b>	Master of Engineering
<b>Speciality:</b>	Control Engineering
<b>Affiliation:</b>	School of Astronautics
<b>Date of Defence:</b>	June,2018
<b>Degree-Conferring-Institution:</b>	Harbin Institute of Technology

## 摘 要

随着智能移动机器人技术的快速发展和需求，同时定位与建图（Simultaneous Localization and Mapping, SLAM）技术越来越被重视和关注，目前单机器人 SLAM，尤其是基于视觉相机的单机器人 SLAM 已经在学界和应用上取得诸多重大进展，但当前研究人员主要的研究方向还是更多的聚集在单机上，对于多机协同 SLAM 的研究还相对较少，本文根据未来工厂多机协同辅助作业的需求，对基于视觉的多机协同 SLAM 算法进行了初步的研究和实现实验。

本文首先对多机协同 SLAM 中的单机算法基础进行了研究，介绍 SLAM 问题的数学描述和本文传感器的选择，并对 SLAM 算法的前后端理论进行了调研和综述，综合比较课题需求后选取了 ORB-SLAM 算法作为实现基础，并对该算法的实现框架进行了分析介绍。关于基于此算法的多机器人协同 SLAM 问题主要研究热点包括三个方面，一是机器人系统的任务分配和多机通信，二是多机（或是相机）位姿的数据关联，三是依赖数据关联的地图拼接（也有研究人员将地图拼接和数据关联并称为地图拼接）。本文对多机算法的研究主要集中于地图拼接部分，着重分析了出现地图拼接的两种情况，第一种是基于相机相对观测的地图拼接算法，针对此种情况，本文采用迭代最近点（Iterative Closest Point, ICP）算法进行相对位姿求解并对其进行捆集调整（Bundle Adjustment, BA）优化得到高精度的相对变换矩阵，进而进行地图点云的坐标转换实现地图拼接，第二种情况是基于场景辨识的地图拼接算法，此种情况下，机器人未相遇但它们存在相似的路过场景，此时需进行场景识别来判断是否可以进行地图融合，本文采用视觉词袋技术进行场景辨识，通过导入图像词典树并将保留的图像帧向量化和二进制化后依据词典树单词信息进行分类存储和索引编号，并随时对当前图像帧与处理后进行相似图片索引来进行场景辨识，在匹配成功相似图像帧后本文采用 PnP（Perspective-n-Point）算法结合 BA 优化过程求解此场景下的多机相对位姿变换关系，并通过对此时刻未在该场景下的另一机器人进行路径累加求逆将其所构建的局部地图转换到此场景下坐标系中，最后依据相对位姿关系进行地图拼接并给出了具体的算法实现流程框架。

本文算法实验部分包括单机算法的数据集测试，验证了 ORB-SLAM 算法的跟踪实时性性能优越，通过对此算法的实验测试，得到了所构建的稀疏点云地图，由于其稀疏性，该算法具有进行实时地图拼接的可行性，针对多机算法实现实验，

本文基于 ROS (Robot Operating System) 系统设计实现了一种简单可行的通信方案, 利用一台机器人即作为服务端也作为客户端, 另一台机器人只作为客户端通过 wifi (Wireless Fidelity) 连接通信, 客户端通过话题随时发布自身观测和状态信息以及必要时发布自身构建的环境地图, 供给服务端订阅和使用, 本实验的目的是验证多点云地图拼接的效果, 由于稀疏点云的稀疏性是相对的, 因此本实验主要是在小范围内进行地图拼接, 并对其进行评价, 依此作为大范围扩展的基础, 实验结果表明拼接效果大致符合预期目标。

**关键词:** SLAM; 多机器人; 视觉相机; BA 优化; ORB 特征; 地图拼接

## Abstract

With the rapid development and demand of intelligent mobile robot technology, Simultaneous Localization and Mapping (SLAM) technology has been paid more and more attention. At present, single robot SLAM, especially the single robot SLAM based on visual camera, has made great progress in the academic field and application. But the main research direction of the current researchers is still more concentrated on single robot SLAM. The research on multi robots cooperative SLAM is relatively rare. According to the demand of multi robots cooperative auxiliary operation in future factory, this paper make a preliminary research and experiment on the multi robots cooperative SLAM algorithm based on vision.

This paper firstly studies the basis of single robot algorithm in multi robots cooperative SLAM, introduces the mathematical description of the SLAM problem and the selection of the sensor, and makes a survey and summary of the front and back theory of the SLAM algorithm. After the comprehensive comparison of the requirements, the ORB-SLAM algorithm is selected as the foundation of the implementation, and the implementation framework of the algorithm is also introduced. The main research hotspots of multi robot cooperative SLAM include three aspects. The first is task allocation and multi robots communication. The second is data association of multi robots (or cameras) position. The third is map splicing depending on data association(some researchers add data association into map stitching). In this paper, the research of multi robots algorithm mainly focuses on the map splicing, especially on the analysis of two cases of map splicing. The first is a map mosaic algorithm based on the relative observation of the cameras . In this case, the ICP algorithm is used to solve the phase position and to optimize it by BA in this paper. High precision relative transform matrix could be gotten, and then coordinate transformation of map point cloud is carried out to realize map splicing. The second situation is map mosaic algorithm based on scene identification. In this case, the robots do not meet each other but are passing through similar scene. At this time, it needs scene recognition to judge whether the map can be fused. The visual bag of word technology is used to identify the scene. By importing the image dictionary tree, the reserved image frames are classified and numbered according to the dictionary word information, and the same image index is

used to identify the scene at any time, and the similar image is successfully matched. After matching the similar image frames successfully, we use the PnP algorithm and the BA optimization process to solve the multi robots relative position and pose transformation relationship in this scene, and convert the local map which it has constructed to the coordinate system in this scene, and then the map splicing is carried out according to the relative position and attitude relationship. The specific algorithm implementation process framework is given.

The experimental part of this paper includes the data set test of the single robot algorithm, which proves that the tracking performance of the ORB-SLAM algorithm is superior. Through the experimental test of the algorithm, the sparse point cloud map is constructed. Because of its sparsity, the algorithm has the feasibility of carrying out the real-time map splicing, and the experiment is implemented for the multi robots algorithm. For implementation experiment of multi robots algorithm, a simple and feasible communication scheme is implemented based on ROS system. One robot is used as a server and a client. The other is only the client and they connect through the WiFi. The client releases its own observation and state information at any time through the topic and releases its own built environment map when necessary. The purpose of this experiment is to verify the effect of multipoint cloud map splicing, because the sparsity of the sparse point cloud is relative, so the experiment is mainly to make map splicing in a small range and evaluate it, which is the basis of large scale expansion, and the experimental results show that the stitching effect is great and could meet the expected goals.

**Keywords:** SLAM, Multi Robots, Vision Camera, BA Optimization, ORB Feature, Map Stitching

## 目录

摘 要 .....	I
Abstract.....	III
第 1 章 绪 论 .....	1
1.1 课题研究的来源、背景及意义.....	1
1.2 国内外研究现状.....	3
1.3 主要研究内容及章节安排.....	7
第 2 章 基于视觉的 SLAM 算法理论 .....	9
2.1 引言.....	9
2.2 SLAM 问题的基本原理.....	9
2.3 基于外部传感器的 SLAM 算法分类.....	10
2.3.1 传感器分类 .....	10
2.3.2 小孔成像原理 .....	12
2.4 基于视觉的 SLAM 算法原理.....	13
2.4.1 前端视觉里程计 .....	14
2.4.2 后端数据处理 .....	19
2.4.3 环形闭合检测 .....	21
2.5 ORB-SLAM 算法原理分析 .....	22
2.5.1 ORB-SLAM 算法框架 .....	22
2.5.2 跟踪实现 .....	23
2.5.3 局部建图实现 .....	26
2.5.4 环形闭合检测实现 .....	27
2.6 本章小结.....	28
第 3 章 多机器人通信与地图拼接 .....	29
3.1 引言.....	29
3.2 多机器人系统.....	29
3.3 基于 ROS 的节点通信设计 .....	30
3.3.1 ROS 操作系统 .....	30
3.3.2 基于 ROS 的多机节点通信设计 .....	31
3.4 基于相机相对观测的地图拼接算法.....	32
3.4.1 相机坐标模型 .....	32



3.4.2 相机相对观测 .....	32
3.4.3 地图拼接 .....	35
3.5 基于场景辨识的地图拼接算法.....	36
3.5.1 场景辨识 .....	36
3.5.2 相对位姿计算 .....	37
3.5.3 地图融合 .....	38
3.6 本章小结.....	39
第 4 章 实验与结果分析 .....	40
4.1 引言 .....	40
4.2 硬件平台搭建.....	40
4.3 实验及分析.....	41
4.3.1 相机标定 .....	41
4.3.2 特征提取与匹配 .....	41
4.3.3 ORB-SLAM 数据集测试 .....	42
4.3.5 地图拼接实验 .....	45
4.4 本章小结.....	48
结 论 .....	49
参考文献 .....	51
攻读硕士学位期间发表的学术论文 .....	55
哈尔滨工业大学学位论文原创性声明和使用权限 .....	56
致 谢 .....	57

# 第 1 章 绪 论

## 1.1 课题研究的来源、背景及意义

20 世纪的两世界大战为人类带来了毁灭性的灾难，尤其是第二次战争，不仅对地球生态造成了巨大破坏，世界主要参战国的人口数量也减少了众多，但同时在战争的需求下，各国的疯狂投入也使得众多传统技术和生产方式得到大幅改进，在此基础上又诞生了很多新的技术和概念，如雷达的产生和应用，电子计算机的诞生和惊人发展等。战后，得益于电子机械、计算机、控制论等科技理论的发展，同时巨大的人力资源缺口也使得当时世界主要国家急需寻找一种可以替代人从事简单生产活动的替代品，在此推动下，1954 年美国诞生了世界上第一种正真的可编程机械手类机器人，之后机器人进入快速发展阶段，不断有新型机器人问世，如 1959 年第一台商业化工业机器人 Unimate，1965 年的带有声纳、光电装置的能够自校正的 Beast，1973 年的第一台由小型计算机控制的机器人 T3，到 1997

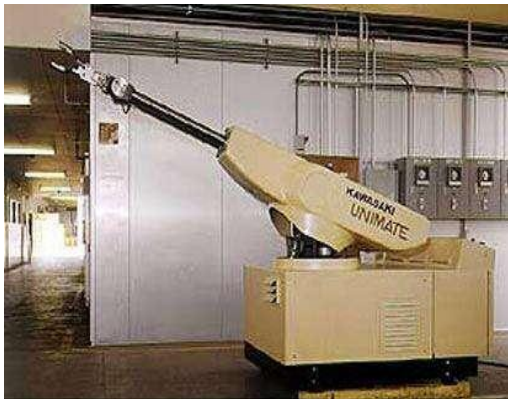


图 1-1 (a) Unimate



图 1-1 (b) Beast



图 1-1 (c) T3



图 1-1 (d) Sojourner

图 1-1 机器人的发展

年第一台真正用于科学工作的自主移动式火星车 Sojourner 等。可以看到机器人的发展逐渐呈现多元化，特定功能性以及智能化。

目前机器人从应用环境来分可分为工业机器人和特种机器人两类，工业机器人主要以工业生产中的机械手和机械臂为主，特种机器人主要是针对特定环境、特定应用需求等条件而专门设计实现的机器人类。其中自主移动机器人作为特种机器人的代表初产生于上世纪 60 年代，这是一种相对高度智能化的机器人定义，能够实现诸多自主功能，主要有感知环境定位、实时的决策与规划、自身控制与执行等。由于它的高自主性，以及不断提升的性能，其被希望应用于越来越多的场合，并随其设计者和使用者的特点而各具特色。如今自主移动机器人单个独立运行已经得到了广泛的应用，如现代工厂中广泛使用的 AGV（Automated Guided Vehicle）小车，或是餐馆的送餐机器人，在这些机器的辅助下不管是生产效率还



图 1-2 (a) AGV



图 1-2 (b) 送餐机器人

图 1-2 移动机器人

是服务效率都得到了很大提升。但同时随着不断新增的工程环境复杂度和更广泛的的需求，单个机器人独立作业的方式已经难以满足，加之相关理论的发展和完善，如多智能体系统（MAS, Multi-agent System）、分布式人工智能（DAI, Distributed Artificial Intelligence）等，机器人研究者们随之提出了多机器人系统（MRS, Multiple Robots System），并已取得诸多进展，如当下火热的机器人足球大赛、仓储物流配送等。多机器人系统可通过相互间的协同配合来完成相应的任务需求，因此可以完成更加复杂的工作，具有很强的灵活性和适应性，适用于大规模环境，对于多机中的个体来说，由于只需要完成相应的子任务，其个体功能可做到简单易实现，便于设计。

自主移动机器人的自主性首先体现在机器人能否在未知初始环境下完成自身的环境感知和定位，同时建立环境地图以供之后的导航与规划任务需求，因此机器人的同时定位与地图构建（Simultaneous Localization and Mapping, SLAM）已成



为移动机器人的关键技术之一。同时随着传感器技术的发展，应用于 SLAM 问题



图 1-3 (a) 足球机器



图 1-3 (b) 物流机器

图 1-3 多移动机器人系统

题上的选择也变得广泛，声纳与激光雷达较早的被应用，但由于它们所采集的信息丰富度较低，加之相对高昂的价格，应用范围受限，很难被大规模推广。近些年来，随着计算机视觉算法的发展与完善，结合相对低廉的视觉传感器，基于视觉的 SLAM 算法的研究已成为当前的热门研究方向之一。相比于单机器人视觉 SLAM，多机器人还需解决的问题包括多机通信，多机数据关联以及地图融合等问题，这些都是当前多机研究所面临的最为急迫的问题，因此对于多机器人视觉 SLAM 算法的研究具有很高的理论和应用价值，同时结合现有的软硬件条件进行多机协同 SLAM 算法的实现也成为当前的热门研究方向之一。本课题正是基于这样的应用前景，以及室内工厂多机辅助作业的需求而提出的。

## 1.2 国内外研究现状

SLAM 问题公开提出于 1988 年，Smith Self 和 Cheeseman 讨论了如何解决环境完全未知下移动机器人的定位导航问题。1995 年 Durrant-Whyte 首次给出了 SLAM 问题的理论框架，并讨论了如何对该问题收敛性进行检验<sup>[1]</sup>。SLAM 问题的研究从其产生到现在已经历 30 年的发展，尤其是进入新世纪以来，随着计算机性能的快速大幅度提升，以及相应传感器硬件的更新换代，使得众多 SLAM 算法可以从理论走向实现。以激光雷达和视觉相机作为外部信息采集单元的 SLAM 算法是当前最为主流的两个研究方向，单个自主移动机器人的视觉 SLAM 算法目前已经取得了很多成功的实验室实现案例，其中以激光雷达作为外部唯一传感器的 SLAM 系统有 2007 年开源的 gmapping<sup>[2]</sup>，这是由德国 Freiburg University 的 Giorgio Grisetti 等人基于粒子滤波而提出的一种 SLAM 算法<sup>[3]</sup>，可以实时的建立二维环境地图，但需结合其它定位算法实现实时定位，如自适应蒙特卡洛定位算法(Adaptive

Monte Carlo Localization, AMCL)<sup>[4]</sup>。2010 年,针对于大规模环境下的构图效率问题, Konolige K 等人提出了 KartoSLAM<sup>[5]</sup>,其后端方案选用的是基于位姿图优化的方法,利用了信息矩阵稀疏特性进行解耦求解。之后 Kohlbrecher S 等人针对三维空间的位置估计问题提出了的 HectorSLAM<sup>[6]</sup>,在硬件上该算法仅需使用高频激光雷达进行环境探索,由于能够建立三维空间地图使得该算法可以应用于不平坦区域移动机器人的探索和无人机 SLAM 上。目前最引人注目的室内激光雷达 SLAM 项目当属 Google 于 2016 年的 ICRA (IEEE International Conference on Robotics and Automation) 会议上公布的 Cartographer<sup>[7]</sup>,该项目几乎能够实现一套完整的并支持众多平台和多种传感器的 SLAM 系统,并且相对其它开源项目所需占用的计算和存储资源更少,显著的提高了建图效率,这也使其在 SLAM 研究领域掀起了一阵轰动。以视觉相机作为环境感知传感器的 SLAM 系统发展与兴起主要得益于视觉理论与硬件的突破性进展和计算机性能的大幅提升,算法上,视觉 SLAM 可分为视觉前端和处理后端,前端构造视觉里程计,后端对前端信息处理并完善优化定位和建图。以后端处理方法分类 SLAM 算法可分为两类:基于滤波的方法和基于图优化的方法,滤波器法的代表最著名的当属 EKF-SLAM (Extended Kalman Filter SLAM)<sup>[8]</sup>和 FastSLAM (Rao-Blackwellized Particle Filters SLAM)<sup>[9]</sup>。而基于图优化的 SLAM<sup>[10]</sup>理论则是在其信息矩阵的稀疏性<sup>[11]</sup>被发现后逐渐被广泛研究并成为了主流的视觉 SLAM 后端方案<sup>[12]</sup>,图优化法的突出优点就是对系统不在有强线性和高斯性的要求,同时优化的目标是全部时刻的状态变量,这使得该算法能够得到高精度的估计值,应用面更加广泛,由于易加入环形闭合检测过程,建图的一致性和可控性将更高,所以当前的视觉 SLAM 算法研究大都会使用基于图优化的后段方案。第一个实现实时视觉 SLAM 的开源项目当属 2007 年 MonoSLAM<sup>[13]</sup>,该项目将单目相机与 EKF (Extended Kalman Filter) 后端相结合,能够实时的跟踪稀疏特征点,紧接着该年由 Klein 等人发布了另一单目开源项目 PTAM (Parallel Tracking and Mapping)<sup>[14]</sup>,一种基于捆集调整 (BA, bundle adjustment) 的实时 SLAM 算法,之后该种后端方案成为趋势,2013 年 Engel 等人提出了基于单目相机的直接法实现半稠密视觉里程计<sup>[15]</sup>,随后发布了可大范围构建半稠密地图的 LSD-SLAM<sup>[16]</sup>,能够实现稠密建图的算法有 RGB-D SLAM<sup>[17]</sup>,其采用了一种可直接获取深度信息的新型相机 RGB-D 相机,2015 年由 Raúl Mur-Artal 等人提出的基于稀疏法的 ORB-SLAM<sup>[18]</sup>则是 PTAM 的众多继承者中最具代表性的一位,可支持多种相机模式,由于使用了三线程并行运行,效率更高。

相对于单个机器人 SLAM 问题上述已有不少解决方案,多机器人 SLAM 问题的研究还算是一个比较新的领域,目前,协同 SLAM 的研究主要方法大都是对某

种单机算法的扩展，研究人员们的兴趣主要聚焦于协同定位的精度问题、地图拼接的一致性和冗余问题、算法的时间和空间成本问题以及环境的适应性问题等问题，并已取得诸多研究成果。大体上多机器人 SLAM 问题的解决方案目前主要基于两种方式：（1）集中式，由单个中央模块管理负责评估所有机器人的运动轨迹，并结合所有机器人的观测同时建立联合地图。（2）分布式，整个系统中每个个体成员根据自己的轨迹观测建立各自的局部地图，同时每个成员都可通过融合机器人团队成员的局部地图生成一个全局地图并发布。多机协同 SLAM 的研究现状如下

文献[19]提出了一种基于分布式系统的方法，所用的后端方案为基于滤波的方案，思路是每个成员机器人在构建自身的局部地图时持续地尝试在其它成员构建的局部地图中定位。该方法可以应用于机器人成员间彼此初始位置未知的情况，数据关联采用了 rendez-vous 技术，但是受粒子滤波器粒子数的影响使得多机局部地图拼接的计算复杂度很高。

文献[20]提出采用一个中央管理模块使用 EKF 估计所有机器人成员的全局位置和观测得到的 2D 路标形成的状态向量并建立统一的全局地图。在该方法中，机器人成员间的相对初始位置、数据关联的结果均假设为已知，另外假设所用多机器人也假设为理想运行。

文献[21]提出了通过 CLSF 过滤器（Constrained Local Submap Filter）为每个成员机器人创建一个全局坐标系下的子地图。成员机器人实时更新当前位姿和各自子地图中的路标，并周期性地将子地图路标信息传送给全局地图，该方法的计算复杂度相对有所降低。

文献[22]描述了一种二维地图的表示法，并在多机器人协作 SLAM 中应用了这种表示法。作者从流形的角度出发将所建地图从二维平面转化为一个高维空间的平面。利用流形表示的自我一致性优点，当机器人在环境中沿闭环运行时，流形地图避免了二维地图的交叉覆盖的问题。

文献[23]计算了相对变换矩阵并将其和地图融合库（MMBs, Map Merging Bases）用于基于 RBPF 的多机器人地图融合。文中作者提出一种基于高斯过程的概率地图融合法（PMM, Probabilistic Map Merging），并在仿真和真实实验环境中验证了其能够降低地图融合时的误差。

文献[24]提出了一种基于占用网格的多机器人地图融合算法，在无 GPS 的环境下完成多机协同 SLAM。该方法通过将单个地图转化到 Hough 空间并抽象表示然后利用 Hough 峰来实现地图融合，该方法在基准数据集和多机器人平台真实实验中得到验证可行。

文献[25]设计了一种基于子地图的 SLAM 系统,它依赖于惯性测量和立体视觉来创建多机器人稠密 3D 地图。对于在线的位姿和地图估计,其通过增量式图 SLAM 将基于关键帧的局部参考滤波器的结果结合起来。根据滤波器的不确定性估计和独立性假设,文中提出了一种新的图形拓扑结构,将多个机器人的局部滤波器估计解耦成 SLAM 图,并评估了其在室内、室外和混合场景下对两个不同机器人的优势,还进行了两个扩展实验中的多机器人设置,以评估全 SLAM 系统,包括视觉机器人检测和子地图匹配作为机器人间的环形闭合约束。

文献[26]介绍了一个实时的多机器人协作 SLAM 的软件框架,而不是建立一个完整的 SLAM 系统,该框架能够为现有的(单机器人)SLAM 系统提供协作建图。该框架可将从其多个机器人获得的局部位姿图聚合成一个全局位姿图,然后反馈给机器人,以提高其建图和定位效果,姿态图的合并纯粹是基于视觉的,不需要定义良好的初始机器人位置或环境标记。为了处理网络延迟,还提出了一种避免使用互斥(从而避免修改现有 SLAM 系统)的局部图一致性的图像校正方案,并提出了一种简单的图像特征滤波方法,该方法利用关联深度图像对不适合于场景识别的图像特征进行滤波。

文献[27]中每个机器人检测地标和其他机器人,并通过扩展卡尔曼滤波器对其位置进行估计,为了获得较好的估计精度,将最优信息融合技术应用于多机器人 SLAM 问题,该技术通过加权平均的扩展卡尔曼滤波器的状态估计,使估计误差协方差最小化。

文献[28]将基于八叉树的占用网格应用到多机器人 3D 建图的问题中。八叉树能够有效地大范围环境建图并能保证足够的分辨率。文中定义和验证一个系统的,它使用具有配准技术的来自环境中常被建图部分的地图数据进行拼接,所定义的系统可以用于移植到移动机器人的完整解决方案中。

文献[29]提出了一种基于分布式系统方案。利用两台移动机器人分别各自执行基于 FastSLAM 的改进算法生成局部 3D 网格地图,记录两机器人的位置估计信息和环境观测数据。并在单机 SLAM 完成之后,在其中一台机器人建成的局部地图中使用自适应蒙特卡罗定位算法定位其同伴相对于它的位置,利用相对变换矩阵在基于重叠处进行两地图的拼接融合,生成完整的探索环境地图。

国内在 SLAM 问题领域研究起步较晚,第一篇正式介绍 SLAM 的文献发表于 2004 年<sup>[30]</sup>,2006 年鞠玉江<sup>[31]</sup>在国内首次利用了视觉方法实现了 SLAM 问题,之后 SLAM 问题的研究越来越被国内科研人员所重视和投入,有关多机器人协作 SLAM 的研究也很快引起了 SLAM 领域研究人员的兴趣。苑晶等人<sup>[32]</sup>将局部子地图方法结合到 SLAM 问题中提出了一种多机器人主动 SLAM 算法,哈尔滨工业大

学的钟进<sup>[33]</sup>实现了基于视觉的小范围环境下两机器人协同定位与建图，张宝先<sup>[34]</sup>通过迭代最近点(ICP, Iterative Closest Point)和自适应蒙特卡洛定位算法(AMCL)来实现基于异构传感器的地图融合和自身定位，利用多个异构传感器建立多特性地图，秦野<sup>[35]</sup>提出了一种基于视觉和激光融合的方法来实现多机协作 SLAM，利用视觉辅助激光进行环形检测，利用视觉场景识别算法进行多机间的数据关联，通过通信建立连接实现多机协同，苑全德<sup>[36]</sup>实现了一个完整的多机器人协作 SLAM 系统，包括多机系统选取、任务分配、数据关联、协同定位和数据关联，提出了一种基于 Mean Shift 聚类算法改进的自然路标提取算法，并基于此联合局部地图更新提出了一种的 NL-SLAM(Natural landmark and Local map based SLAM)算法，最后提出了一种改进自 Fast SLAM 的团队成员共享路标信息的多机协同算法 MR-vSLAM (Multi-robot visual SLAM)，在该算法中单个机器人将其它成员的观测信息视为自身传感器观测的延伸，进而实现全局建图，该算法在大规模未知环境下的环境探索具有很大的优势。

综上所述，这些研究人员们围绕着多机器人协同 SLAM 问题进行了大量的研究和尝试。针对协同 SLAM 问题的研究，有的研究并提出了相关的多移动机器人协同定位算法或是对基于已有算法的改进，有的给出了限定条件下的地图拼接算法和数据关联算法的尝试或是提升性能，有的侧重于地图表示方式(2D/3D 地图或是栅格/点云地图)，还有的侧重于协同 SLAM 的实现，这些研究都取得了一定的成果。

### 1.3 主要研究内容及章节安排

本文主要研究室内环境下基于视觉相机的多机器人协同 SLAM 算法和实现，协同 SLAM 是目前 SLAM 领域的热门研究方向之一，其难点主要包括多机任务分配，通信和地图拼接，本文主要针对其关键技术之一地图拼接做了一些研究和尝试，基于 ROS 系统设计了一种简单可行的多节点通信方案，并最后进行了实验验证。本文研究的地图拼接内容主要包括多机位姿信息关联和基于此的地图拼接，本文分析了两种主要的可进行地图拼接的场景条件，给出了此种条件下地图拼接实现的方案和流程。本文研究了基于 ORB-SLAM 算法的地图拼接，拼接的地图为三维稀疏点云地图，该方法的优势是可以快速构建地图，并且由于构建地是三维点云地图，环境信息更加丰富便于后期机器人的扩展利用。本文结构安排如下

第一章主要进行了课题的调研，介绍了多机协同 vSLAM 的发展历程，研究了国内外该领域的研究现状，给出了本课题的来源背景，并介绍了一下本课题的主要研究内容。



第二章介绍了单机 SLAM 问题的数学表示,SLAM 的分类,对基于视觉 SLAM 的前后端方案进行了研究介绍,重点研究了基于光滑的 SLAM 算法理论,最后经分析选取了 ORB-SLAM 算法作为多机协同 SLAM 的单机实现基础。

第三章设计了一种可实现的 ROS 多节点通信方案,之后重点研究了两种多机地图拼接的适用场景及相应算法,并给出了理论分析和实现方案。

第四章介绍了硬件平台的搭建,实验部分内容包括特征提取算法实验和比较,ORB-SLAM 算法的数据集测试,目的是通过与高精度的真实值比较,验证算法的实现性能,之后是单机算法实验主要是建图的效果评估,包括实时建图效率和场景复现效果,最后是多机协同 SLAM 算法实现实验与分析,着重分析地图拼接的效果并对其拼接的一致性进行了分析。

## 第 2 章 基于视觉的 SLAM 算法理论

### 2.1 引言

本章将对 SLAM 问题的基本原理做出阐述，并将着重研究基于视觉的 SLAM 算法，包括视觉相机的应用原理、前端的视觉里程计构建、后端的系统状态估计，之后将对当前若干种主流的视觉 SLAM 算法进行分析，并选取一种较为合适的方案作为多移动机器人协同 SLAM (CSLAM, Collaborative SLAM) 算法室内实现的基础进行拓展。

### 2.2 SLAM 问题的基本原理

同时定位与建图 (SLAM) 问题的提出是为了为机器人提供精确实时的地图供给机器人进行精确的移动和自主导航，SLAM 问题可定义为在机器人没有环境先验信息的情况下，利用自身所携带的运动和环境感知传感器得到的检测和观测信息进行全局参考位置估计，同时实时地建立一种可供使用的全局环境地图。

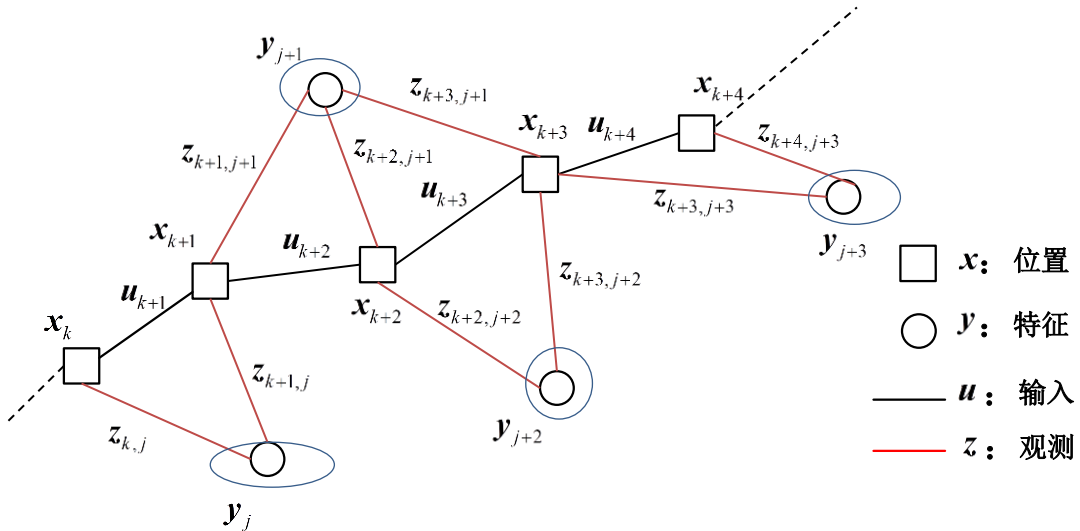


图 2-1 SLAM 问题基本表示

如图 2-1，SLAM 问题的基本数学表示可这样描述，对于某个移动机器人或是其它类型的机器人，当它处于某个未知环境且无法获取自身的先验信息，主要是自身的初始位置信息时，需要依赖自身设备建立起周围环境地图，并依赖此图进行自身的相对定位。假设某一时刻  $k$ ，机器人的相对位置估计值是  $x_k$ ，当前可观测到的路标特征是  $y_j$ ，观测值为  $z_{k,j}$ ，在控制输入  $u_{k+1}$  的作用下，机器人在下一采样

时刻到达位置  $x_{k+1}$ ，同时观测到  $y_j$  和路标特征  $y_{j+1}$ ，观测数据分别为  $z_{k+1,j}$  和  $z_{k+1,j+1}$ ，则  $k+1$  时刻机器人的运动方程和观测方程可表示为

$$x_{k+1} = f(x_k, u_{k+1}) + w_{k+1} \quad (2-1)$$

$$z_{k+1,j} = h(y_j, x_{k+1}) + v_{k+1,j} \quad (2-2)$$

其中  $w_k$ 、 $v_{k+1,j}$  分别为过程噪声和观测噪声（图中椭圆所示范围），机器人在这种运动探索模式下，从初始位置开始每隔等长时间采样一次获取当时机器人的环境观测数据以及自身的运动参数并记录下此时刻起的控制输入，经过一段时间  $t$  的运动，机器人将产生一系列状态离散值，状态向量的集合可表示为

$$X^t = (x_0, \dots, x_k, \dots, x_t, y_1, \dots, y_j, \dots, y_m) \quad (2-3)$$

其中， $x_k$ ， $y_j$  分别为各采样时刻的机器人位置估计值和环境中被观察到的路标特征，同时路标特征的观测值集合可表示为

$$Z^t = (Z_0, \dots, Z_k, \dots, Z_t)^T \quad (2-4)$$

其中， $Z_k$  是  $k$  时刻机器人对路标特征观测值的集合。由 SLAM 问题的描述可知，在上述方程组中，每一时刻的机器人位置  $x_k$  与新增路标  $y_j$  是未知的，因此需要建立某种方法，利用每一采样时刻的已知量对当前时刻的状态未知量进行估计，并需要考虑这些估计量的不确定性，将估计误差控制在可接受的范围，对位置的估计构成了 SLAM 算法中的定位技术，对环境路标特征的估计构成了 SLAM 算中的建图核心，机器人需要利用已建立好的环境地图进行定位导航，同时也需要依据自身的位置信息进行地图构建的扩展，也就是说定位与建图需要同时进行，即同时定位与地图构建（SLAM）。

## 2.3 基于外部传感器的 SLAM 算法分类

### 2.3.1 传感器分类

SLAM 算法的研究发展到如今，其主要外部环境传感器的选择也呈现出多样化，主要包括声纳、激光和视觉相机三类。早期研究人员主要利用声纳进行环境探测，这既是由于算法本身的不完善以及相关传感器数据处理技术的不成熟使得对传感器数据采集与处理的要求不高，同时声纳传感器也具有诸多优点，其使用简单，技术要求相对与激光雷达较低，因此价格也较为便宜，容易获取，便于推广，再加上声纳的距离探测原理较为简单，易于实现稳定性较好，所以声纳也是

第一种被利用在 SLAM 算法实验和实现的环境感知传感器。



图 2-2 (a) 声纳



图 2-2 (b) 激光雷达

图 2-2 传感器

声纳传感器虽具有诸多优点，但缺点也较为明显，由于超声波的传递路径较为分散，其接收信息的不确定性较大，测距的范围无法精确，且信息量较为单一，这使得用于 SLAM 时的定位精度需额外弥补，建图的不确定性也较高。之后激光雷达的应用改善了这些性能，激光雷达同样是基于这种类似于声纳的直接测距原理，相比于声纳系统激光雷达使用激光作为外部发射和接收信号，这使得整个测距过程更为快速，并且激光雷达的测距误差更小，精度提升幅度较大，测量的不确定性也得到降低，激光雷达采集到的环境信息是以分散点云的形式表示出来，点云的描述信息内容丰富，它们具有较为精确的角度和距离等信息，这使得以激光雷达作为环境传感器的 SLAM 算法可以直接利用点云状态信息的绝对不变性对相邻时刻激光雷达采集到的点云数据进行描述匹配，进而可以计算前后时刻机器人相对运动（主要是旋转和平移运动），以实现自身的定位。

视觉传感器的应用主要是得益于计算机视觉算法的快速发展与实现以及计算机硬件性能的大幅提升，使得对于海量信息的实时处理变为了可能，视觉相机相对于激光雷达来说，对周围环境的信息采集更为丰富，利用这些丰富的信息所建立的环境地图描述性更好，且能够实现对环境的三维重构，同时视觉相机属于无源传感器，具有不发射的特点，对周围环境物体空间上的变动适应性较强，但受光照条件变化的影响较大，另外视觉相机的价格相比激光雷达较为低廉，便于推广，应用前景广泛，这也使得当前国内外对于 SLAM 算法的研究和实现上主要集中在基于视觉相机的硬件基础上。目前已有诸多典型的开源案例<sup>[8][9][52][14][16][17][18]</sup>，SLAM 上用视觉相机目前主要包括单目相机、多目相机（主要是双目相机）和 RGB-D 相机（如微软的 Kinect、Intel 的 RealSense），如图 2-3。

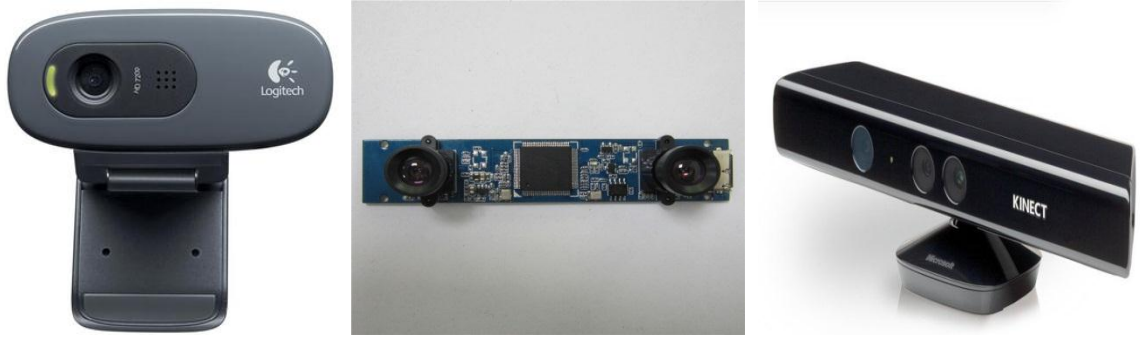


图 2-3 视觉相机

### 2.3.2 小孔成像原理

一般相机的成像过程都满足小孔成像原理，其成像示意图可如图 2-4 表示：

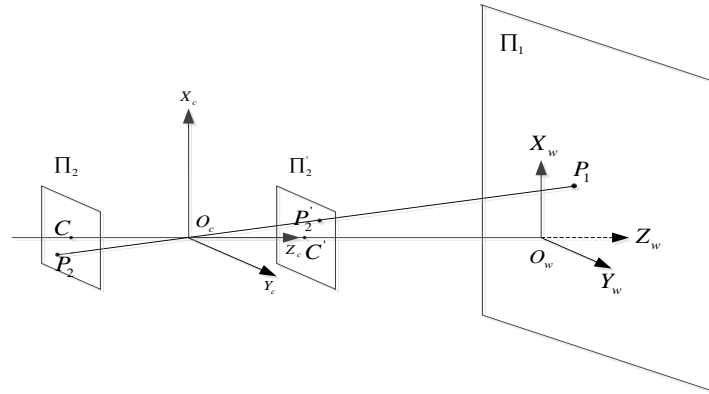


图 2-4 成像示意图

对于某一普通相机摄像时，若外部所有可视景物光线只能通过相机的光心到达相机的成像平面并形成倒立缩小的成像平面点，这种相机即属于基于小孔模型的相机。图 2-4 中  $P_1$  为小孔相机视野内某一点，坐标系  $O_c - X_c Y_c Z_c$  为以光心  $O_c$  为中心所建立的相机笛卡尔坐标系可称为光轴坐标系，坐标系  $O_w - X_w Y_w Z_w$  为一与点  $P_1$  关于  $O_w - X_w Y_w$  面共面且平行于光轴坐标系的另一笛卡尔坐标系，其相当于光轴坐标系沿光轴方向平移距离  $z_1$  而得到，可设  $P_1$  位于光轴坐标系下的坐标  $(x_1, y_1, z_1)$ ，平面  $\Pi_2$  和  $\Pi_2'$  分别为相机的成像平面和等效成像平面， $P_1$  点透过光心  $O_c$  将会在成像平面  $\Pi_2$  上形成投影点  $P_2$ ，将其等效到等效平面即为点  $P_2'$ ，其在光轴坐标系下的坐标可设为  $(x_2, y_2, z_2)$ ，又  $z_2$  值为距离  $O_c C'$  的值即相机焦距  $f$ ，由相似三角形关系可得如下等式

$$\begin{cases} \frac{x_2}{x_1} = \frac{z_2}{z_1} = \frac{f}{z_1} \\ \frac{y_2}{y_1} = \frac{z_2}{z_1} = \frac{f}{z_1} \end{cases} \quad (2-5)$$

上述内容只涉及到外部环境某点到成像平面点的变换关系，之后成像点还将通过尺度变换变换为图像中的像素点（本过程的将不做详细的原理性介绍，只给出相关定义，详细介绍可参看书籍[37]），这整个过程所得的变换关系可称为相机的内参模型，在经过焦距归一化处理后其矩阵形式可表示为

$$M_{in} = \begin{bmatrix} k_x & k_s & u_0 \\ 0 & k_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2-6)$$

式中  $k_x$ ， $k_y$ ——分别为成像平面到像素平面放大系数在  $X$  轴和  $Y$  轴的分量；

$k_s$ ——上述  $X$  轴和  $Y$  轴的耦合系数（由于两轴基本上满足正交关系，此系数长取值为零）；

$u_0$ ， $v_0$ ——光轴线与成像面的交点变换到像素平面的坐标。

另外在实际相机成像的过程中还需考虑相机镜头的畸变系数，需要对相机该系数进行补偿校正（本文选取全局优化方法将畸变引起的误差进行分散化处理，具体内容将在本章后半段介绍）。

## 2.4 基于视觉的 SLAM 算法原理

基于视觉相机的 SLAM 算法经过这个世纪以来的发展已经形成了成熟的研究框架体系，其完整过程框架可表示如下图：

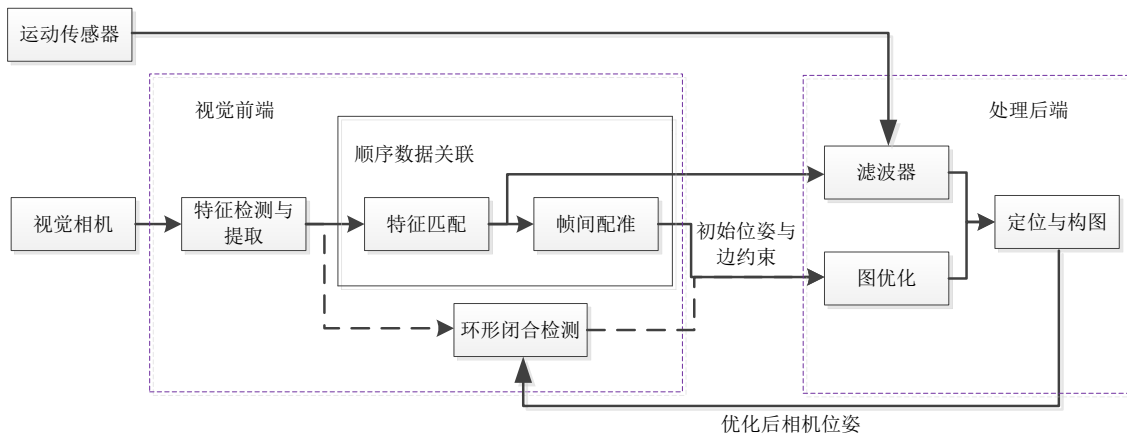


图 2-5 SLAM 算法框架

由图 2-5 可将视觉 SLAM 过程分为三部分，分别是运动图像信息采集、视觉

前端和后端约束问题求解实现 SLAM，视觉 SLAM 算法主要是指后两部分，下面将分别研究比较。

### 2.4.1 前端视觉里程计

视觉前端的主要是利用图像信息构建视觉里程计和进行环形闭合检测<sup>[38]</sup>，可细分为三部分：特征检测与提取、顺序数据关联和环形闭合检测，由于环形闭合检测过程涉及到后端数据的利用，将放在后面介绍。目前视觉里程计按对处理图像方式的不同可分为直接法视觉里程计和特征法视觉里程计，直接法产生于光流，会直接根据图像帧的像素灰度信息来计算相机前后时刻的相对运动，需要有一个灰度不变的强假设条件，对于某一前后时刻的相邻两帧中的相同空间点的投影做最小化亮度测量误差处理，这一步需要事先获取景物的深度信息，进而计算出相机的相对运动变化矩阵。特征法需要提取图像特征点，并采用描述子进行描述区分，之后进行帧间的特征匹配，利用代数方法或最小化重投影误差（BA，Bundle Adjustment<sup>[39]</sup>）实现视觉里程计。

两种方法相比，直接法避开了特征提取与描述的过程，也就避开了特征缺失后的重定位问题，直接利用灰度信息进行匹配，能够实现稠密构图，但是由于需附加一个很强的外部假设条件，使得算法的光照鲁棒性变差，无法计算灰度变化很小的图像区域，容易出现灰度误匹，同时需要计算和匹配整张图像，算法的时间成本也不会很低。特征法也被称为间接法，需要提取图像中具有代表性的特征点，并进行描述子描述，这些特征点通常满足如下性质：

- 所代表的空间点具有固定不变的特性，也就是观测可重复性；
- 区分度可辨识，描述子描述能力强，便于数据关联；
- 数量充足且高效，对空间物体的轮廓复现均匀，但不稠密；
- 局部代表性，每一特征仅与其所在位置相关联。

特征法的特征点提取与描述需要占用比较多的计算时间，但考虑到特征点所具有的特性，其相比于直接法对环境的适应性更强，由于描述子的存在，帧间匹配效率会更高，便于重定位与场景识别。且特征法对外部传感器性能要求不高，可以无需直接获取景物深度值，实现上较为便利，这也使得特征法成为当前 SLAM 算法应用的主流，本文将主要研究基于特征的视觉里程计实现，同时考虑到大规模使用的成本性问题，拟采用单目相机作为视觉传感器。

#### 特征提取：

自特征法提出以来，不断出现了多种特征提取方案，它们各具优势，可分别应用于不同场景需求，目前主流的应用方案有如下几种：

1) SIFT 特征<sup>[40]</sup> SIFT 特征 (Scale-invariant Feature Transform) 于 1999 年提出, 2004 年成熟。SIFT 特征提取于图像中物体的局部外观上具有尺度和旋转不变性的兴趣点, 这些兴趣点具有很高的分辨性和稳定性, 对环境的微小改变适应性很强, 且能够大量产生, 但计算复杂读较高。

2) SURF 特征<sup>[41]</sup> SURF 特征 (Speeded Up Robust Features) 产生于 2006 年, 与 SIFT 特征提取目标一致, 寻找这种尺度不变的兴趣点进行识别和描述, 采用方形滤波代替高斯滤波, 通过构造并计算海森(Hesseian)矩阵行列式值的极值点来检测和提取特征点, 计算寻找特征点主方向, 构造 64 维 (SIFT 特征描述子 128 维) 特征描述子。相比于 SIFT 特帧计算时间成本成倍降低, 但对图像灰度梯度依赖性更强, 尺度不变性降低。

3) ORB 特征<sup>[42]</sup> ORB 特征 (Oriented FAST and Rotated BRIEF) 提出于 2011 年, 提取方法源自于 FAST 特征 (Features from Accelerated Segment Test) <sup>[43]</sup>, 由于增加了特征点主方向, 使其具有一定的旋转不变性, 过程中也加入了一定的尺度不变性, 同时继承了 FAST 特征的高速提取。描述子源于 BRIEF 特征描述(Binary Robust Independent Elementary Features) <sup>[44]</sup>, 通过为 BRIEF 增加旋转因子, 增强了该种描述符的旋转不变性, 再加上二进制数描述的快速优势使得整个 ORB 特征提取描述过程比前述两种算法效率成数量级倍增。下面介绍一下该种算法提取与描述的过程。

oFAST 特征提取:

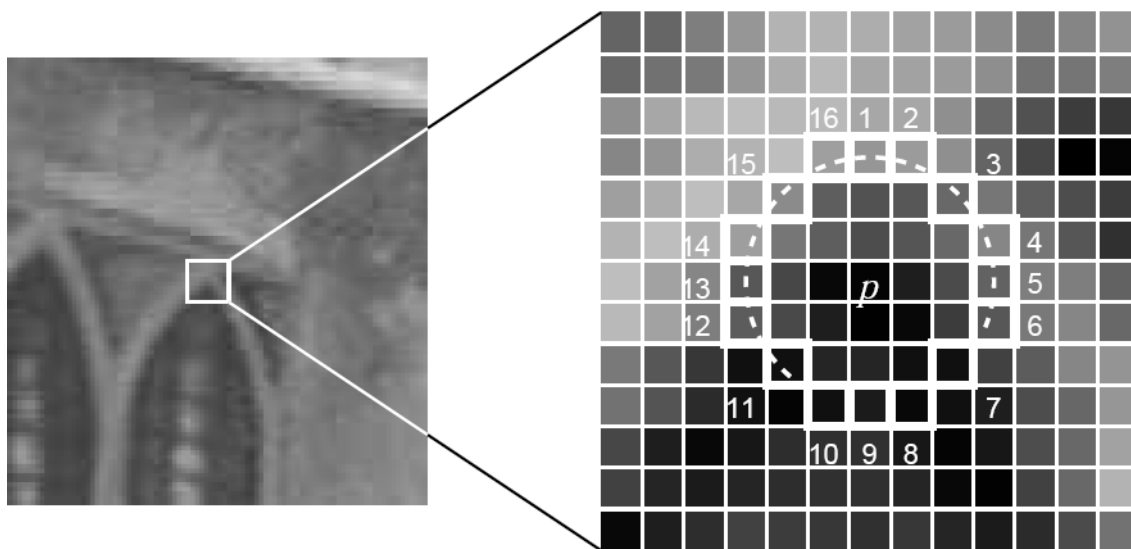


图 2-6 FAST 角点检测<sup>[43]</sup>

1) 提取候选点 任选灰度图像中一点  $p$ , 围绕其构建一个 3 像素的圆, 并将圆周上像素依次编号(1, 2, ...), 检测  $p$  点灰度值  $I_p$  并与编号灰度值  $P_x (x \in 1, 2, \dots)$  进行比较判断是否有连续超过 12 个像素灰度值一致小于(大于)  $p$  点一个阈值  $t$ ,



有则选取  $p$  点为候选点，重复该过程得到大量候选点；

2) 筛选候选点 利用非极大值抑制去除局部较密集候选点，为每一个候选点计算其分数函数  $V$  ( $V = \max(\sum_{x \in S_{\text{bright}}} |I_{p \rightarrow x} - I_p| - t, \sum_{x \in S_{\text{dark}}} |I_p - I_{p \rightarrow x}| - t)$ )，在临近的候选

点中保留分数值较大的点，删除其余点，产生 FAST 特征点；

3) 加入尺度不变性 建立 8 层图像金字塔，并加入比例因子，在每一层通过 harris 滤波之后产生 FAST 特征点；

4) 加入旋转不变性 利用强度质心法确定 FAST 特征点的主方向，首先定义一个特征点块矩

$$m_{pq} = \sum_{x,y \in r} x^p y^q I(x,y) \quad (2-7)$$

式中  $r$  ——特征点块半径；

$p, q$  ——列表值 ( $\in \{0,1\}$ )；

$I(x,y)$  ——图像灰度。

利用矩计算特征点块质心

$$C = \left( \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right) \quad (2-8)$$

连接质心与特征角点，定义从角点到质心的方向为特征点主方向，用  $\theta$  表示

$$\theta = \arctan\left(\frac{m_{01}}{m_{10}}\right) \quad (2-9)$$

rBRIEF 特征描述：

1) 取尺度空间上的特征点  $S \times S$  像素领域，随机选取  $n$  ( $n$  可取 256) 对像素点  $p_i, q_i (i=1,2,...,n)$ ，定义像素对集合为

$$T_n = \left[ \begin{pmatrix} x_{p_1} & x_{q_1} \\ y_{p_1} & y_{q_1} \end{pmatrix}, \begin{pmatrix} x_{p_2} & x_{q_2} \\ y_{p_2} & y_{q_2} \end{pmatrix}, \dots, \begin{pmatrix} x_{p_n} & x_{q_n} \\ y_{p_n} & y_{q_n} \end{pmatrix} \right] \quad (2-10)$$

2) 加入旋转角度  $\theta$ ，转换点对集

$$T'_n = R_\theta T_n \quad (2-11)$$

3) 比较  $T'_n$  中像素点对灰度值，生成  $n$  位二进制描述符，规则是

$$i = \begin{cases} 1, & I'(p_i) > I'(q_i) \\ 0, & I'(p_i) < I'(q_i) \end{cases} \quad (2-12)$$

基于实时性的要求，同时考虑到室内环境的相对固定性，本文将采用 ORB 特

征提取方案，特征提取与匹配实验将于第五章说明。

#### 顺序数据关联：

该过程可包含两部分，特征匹配与帧间配准。特征匹配主要是建立相邻图像帧间的对应相同空间点特征的对应关系，可以使用改自于暴力匹配法的快速最近邻算法（FLANN, Fast Library for Approximate Nearest Neighbors），计算描述子间的相似特性（距离）进行匹配。对于匹配失败或是误配的特征点对可利用随机抽样一致性算法（RANSAC, random sample consensus）<sup>[45]</sup>滤除，通过构造抽样集和模型，寻找包含内点的一致集来去除剩余的外点（误匹配对）。

帧间配准是通过计算相邻帧上已经配对成功特征点对之间空间变换关系（主要是旋转和平移变换），来初步确定相邻帧的相对运动。由于本文主要研究基于单目相机的 SLAM 算法实现，故只对用于单目相机的帧间配准方案进行分析介绍，多目相机和 RGB-D 相机由于可以间接或直接获取景物深度信息，使用迭代最近点算法（ICP, Iterative Closest Point）可方便高效的进行运动关系求解，且计算精度高。对于单目相机连续帧间配准一般需要用对极几何约束和 PnP 求解来计算，利用对极几何约束关系可用来进行两帧间的相对运动初始化，同时利用三角化原理运动分析得到初始的特征点对应的地图点的三维坐标，即完成地图构建的初始化，之后可进行连续的 PnP 问题求解，计算连续相机相对运动。PnP 问题可表示如下图（示例为 P3P）

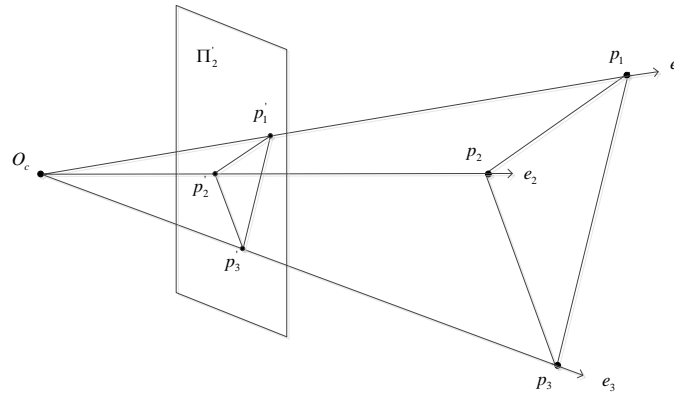


图 2-7 P3P 问题示意

PnP 问题可以描述为，对于已经匹配成功（至少存在三对特征匹配点对）的两帧图像（特征点图像像素坐标可直接获取），其匹配的点对在所构建的地图中可视且三维信息已知，相机内参矩阵已知，由这些已知量求解两帧间的相对运动信息。目前求解方法可分为两种，一是代数求解法，二是误差优化法，当匹配成功的特征点对较少时可用代数方法（如 P3P 法）求解，但是考虑到不管是稠密法还是稀疏法，特征匹配的点对相对而言都是大量的，此时采用代数方法很容易陷入局部

最优，无法得到正确的帧间运动，因此可增加优化过程进行求解（BA）。

BA（或称捆集优化）通过将已知空间坐标的地图点重投影到发生相对运动的图像帧中并与真正的像素投影点计算距离差，并对这些差的平方集合进行优化，优化变量的一部分即是相对运动矩阵，最后达到最小化重投影误差的目的，示意图如下

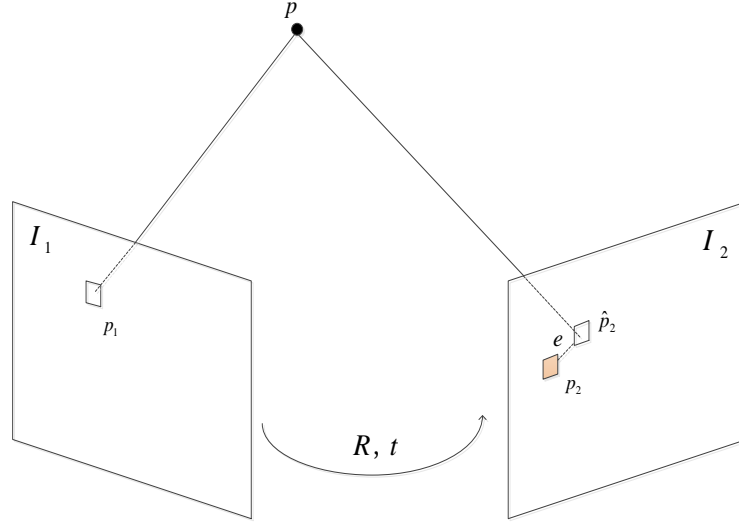


图 2-8 重投影

$I_1, I_2$  分别是前后两帧图像， $p_1, p_2$  分别是地图点  $p$  在两帧图像上的真实投影像素， $\hat{p}_2$  为重投影像素点， $e$  是像素坐标误差值， $R, t$  为帧间运动参数，则理想情况下有如下关系

$$\begin{cases} s_1 \begin{bmatrix} x_1 \\ 1 \end{bmatrix} = M_{in} X \\ s_2 \begin{bmatrix} x_2 \\ 1 \end{bmatrix} = M_{in} (RX + t) \end{cases} \quad (2-13)$$

式中  $x_1, x_2$  ——分别为  $p_1, p_2$  图像点的像素坐标；

$s_1, s_2$  ——分别为  $p$  点在两帧中的深度值；

$M_{in}$  ——相机内参。

对于每一匹配成功的特征点对都满足上式关系，但考虑到观测误差的存在，上式关系并不能严格满足，需优化处理，假设匹配成功  $N$  个点对，可构造对目标函数的优化

$$\min J = \min_{R, t, X} \sum_{j=1}^N \left\| s_2^j x_2^j - M_{in} (RX_j + t) \right\|^2 \quad (2-14)$$

对上式进行求解，可得最优相对运动解和优化后的地图点坐标，求解方法可选择

共轭梯度法、高斯-牛顿法或是 L-M (Levenberg-Marquardt) 法等, 利用 P3P 法求的值作为初始迭代值。

通过以上两步的不断重复计算实现视觉里程计的构建。

### 2.4.2 后端数据处理

SLAM 的后端是指对前端构造的视觉里程计进行降噪处理以得到更加精确的相机位姿变化, 同时依据位姿的精确变换关系优化地图空间特征点的位置信息, 使其更精确且能够达到全局一致, 目前主要的后端算法分为两种, 一种是基于滤波的, 一种是基于平滑的 (图优化)。基于滤波的方法, 考虑是将整个 SLAM 过程看做为一个马尔可夫随机过程, 相机当前时刻的状态值只与前一时刻的状态相关, 不考虑更早时刻的影响, 滤波过程只利用到前端的特征匹配结果, 利用内部运动传感器获取自身的运动方程, 通过运动方程预测当前时刻的状态以代替帧间配准的过程, 再根据求取观测误差极小值的方式进行状态修正和地图更新或是利用概率逼近的方式进行前述过程, 典型的基于滤波的方法有 EKF-SLAM (基于扩展卡尔曼滤波) 和 FAST-SLAM (基于粒子滤波), 这种基于滤波的递归贝叶斯状态估计的方法由于采取的是这种增量式建图过程, 对于每一步估计的误差都会保留到状态中, 长时间大范围的运行之后将会产生大量的误差累计, 严重影响估计精度造成建图结果不一致。而基于平滑的方法, 是对整个 SLAM 过程的估计误差进行优化处理, 在全局一致性建图和大规模探索过程中更具优势, 这也是当前最为被关注和研究的后端处理算法, 本文将主要研究基于后端图优化的 SLAM 算法实现。

基于图优化的 SLAM 主要包括前端的构建图和后端的基于图约束的误差优化, 前端构建的图主要组成部分为节点和边, 节点包括相机的位姿节点 (pose) 和地图路标特征节点 (point), 边 (edge) 为观测约束和相机运动约束, 以因子图的形式可表示为

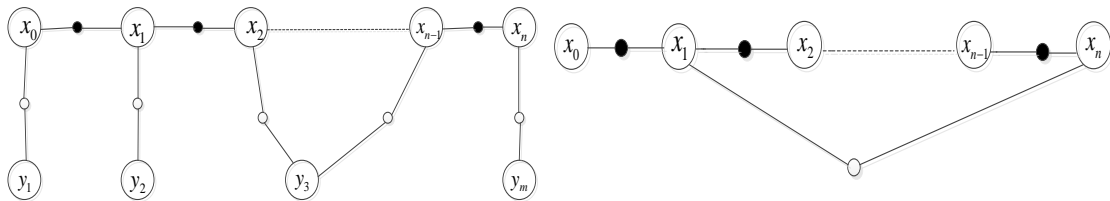


图 2-9 两种因子图表示

图中左图为包含所有节点和边的表示法, 右图为包含相机位姿节点和相机相对运动约束的边即位姿约束边的表示法, 其中黑色小圆点代表里程计因子, 白色小圆点代表路标特征因子, 右图由于不含路标节点和观测约束只有相机位姿节点和它们之间的边故又称位姿图 (PoseGraph), 对于左图, 考虑利用所有边约束优化

所有相机位姿节点坐标和路标节点坐标，这种 GraphSLAM 又称为 Full-SLAM 问题，为了方便描述可将相机状态空间模型改为如下形式

$$w_k = x_k - f(x_{k-1}, u_k) \quad (2-15)$$

$$v_{k,j} = z_{k,j} - h(y_j, x_k) \quad (2-16)$$

设  $w_k$  服从  $w_k = e_k \sim N(0, \Sigma_k)$ ， $v_{k,j}$  服从  $v_{k,j} = e_{k,j} \sim N(0, \Sigma_{k,j})$ ，则 Full-SLAM 过程对

目标函数的优化表示为

$$\min_{x,y} J = \min \left( x_0^T \Sigma_0^{-1} x_0 + \sum_{k=1}^n e_k^T \Sigma_k^{-1} e_k + \sum_{k=1}^n \sum_{j=1}^m e_{k,j}^T \Sigma_{k,j}^{-1} e_{k,j} \right) \quad (2-17)$$

基于位姿图的优化过程未有对大量地图点进行直接优化，时间成本大为降低，实时性更好，其对目标函数的优化表示为

$$\min_x J = \min_x \left( x_0^T \Sigma_0^{-1} x_0 + \sum_{k=1}^n e_k^T \Sigma_k^{-1} e_k \right) \quad (2-18)$$

上述两种基于图的优化方法各具优势，但考虑到实时性的条件，本文选择基于位姿图的优化方法，对公式(2-18)的求解可利用沿梯度负方向下降最快的原理，设  $J$  中某一时刻  $k$  对应的优化目标为

$$E(x_k) = e_k^T \Sigma_k^{-1} e_k \quad (2-19)$$

则

$$E(x_k + \Delta x) = e_k^T (x_k + \Delta x) \Sigma_k^{-1} e_k (x_k + \Delta x) \quad (2-20)$$

$$\Delta E = E(x_k + \Delta x) - E(x_k) \quad (2-21)$$

将  $e_k(x_k + \Delta x)$  在  $x_k$  处线性化，并保留一次项舍去高阶项，得

$$e_k(x_k + \Delta x_k) = e_k(x_k) + J \Delta x_k \quad (2-22)$$

式中  $J$  此处意为雅克比矩阵，且有

$$J = \frac{\partial e_x(x_k)}{\partial x_k} \quad (2-23)$$

将式(2-22)(2-20)带入式(2-21)中得

$$\Delta E = 2e_k^T \Sigma_k^{-1} J_k \Delta x_k + \Delta x_k^T J_k^T \Sigma_k^{-1} J_k \Delta x_k \quad (2-24)$$

优化  $E(x_k)$  即为使  $E(x_k)$  取最小值，则令  $\frac{\partial \Delta E}{\partial \Delta x_k} = 0$ ，得

$$J_k^T \Sigma_k^{-1} J_k \Delta x_k = -J_k^T \Sigma_k^{-1} e_k \quad (2-24)$$

取  $H_k = J_k^T \Sigma_k^{-1} J_k$ ， $b_k = J_k^T \Sigma_k^{-1} e_k$ ，上式可变为

$$H_k \Delta x_k = -b_k \quad (2-25)$$

利用上式可求取  $\Delta x_k$ ，全部目标优化求解式集合为

$$\sum_{k=0}^n H_k \Delta x_k = -\sum_{k=0}^n b_k \quad (2-26)$$

上述整个过程将求解非线性最小二乘问题转化为线性矩阵方程求解问题，利用高斯-牛顿法或是 L-M 法可求的最优位姿解，进而更新地图路标特征。

### 2.4.3 环形闭合检测

环形闭合检测过程可以利用图象帧/关键帧特征信息和已经优化后的相机位姿高效的判断运动过程的局部或是全局一致性，并可以更正检测到的不一致，在 SLAM 过程中加入该流程将极大的提升 SLAM 的准确性，修正长时间估计所产生的位姿累计漂移误差，或者称为累计相对运动误差并依次修正地图误差，基于平滑的 SLAM 方法是对全部时刻的变量进行优化，加入闭环检测流程之后，当相机经过一段时间  $t$  到达某一相似环境时，检测到存在闭环不一致，如下图：

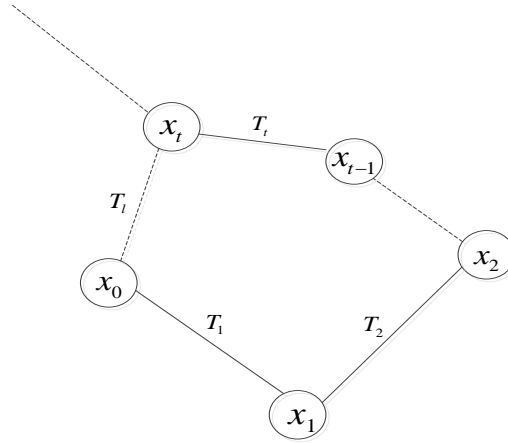


图 2-10 一致性检测

即  $T_1 T_2 \cdots T_t T_t \neq 1$  ( $T$  为相机相对运动矩阵)，当前时刻的优化过程中会加入检测到的一致性约束，即  $T_1 T_2 \cdots T_t T_t = 1$ ，计算求解后得到满足一致性结果的位姿相对运动，并重新更新地图路标特征的位置。相比于基于平滑的方法，基于滤波的算法因其只对最近时刻的两帧图像信息进行相对估计，很难利用上检测到的闭环信息进行全局或局部优化。

基于特征法的环形闭合检测主要是包括场景识别过程和闭环优化融合，本文主要研究基于特征词袋(BOW)的环形闭合检测实现。首先根据词袋(bag of word)模型(一种典型的图像检索模型)的创建思路，利用典型的 K 均值聚类法，离线训练出一个容量巨大包含各种图像特征信息的 K 个独立不相关且单个共性明显的单词的词袋(二进制描述)，并可根据需要以这 K 个单词作为词根依次向下进行相

同类型的聚类形成  $w$  层词典树 (Vocabulary tree)，将其作为图像帧分类存储、索引和判断相似度的基础依据，之后在 SLAM 的过程中对每一个图像帧依据其特征及描述子的特性，在词袋中进行聚类分配，并编上索引号进行分类存储（如建立反向索引系统），场景识别过程发生在 SLAM 的整个过程，通过将当前图像帧用词袋向量表示，利用反向索引系统并结合求解汉明距离 (Hamming distance) 找到最为相似的场景图像帧，之后进行帧间特征匹配（如暴力匹配）配准进行相对变换关系求解，上述即为 SLAM 的场景识别过程典型一种，流程图如下表示

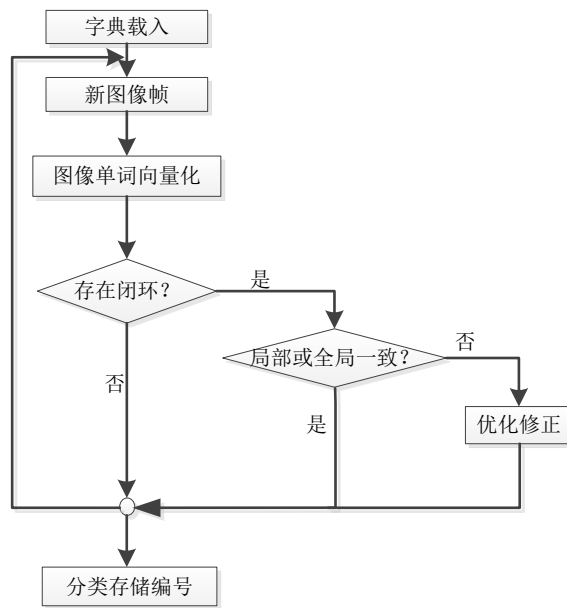


图 2-11 闭环流程

之后对检测到的闭环进行一致性判断和优化修正传递。经过上述对基于视觉相机的 SLAM 算法的分析和比较，本文选取基于当前最为主流的 ORB-SLAM 算法为基础进行多机拓展，该算法满足上述分析选取条件，且实时性好。

## 2.5 ORB-SLAM 算法原理分析

### 2.5.1 ORB-SLAM 算法框架

算法框架示意如图 2-12，该 SLAM 算法全程始终使用 ORB 特征算法作为对图像帧的处理方法，并在整个 SLAM 过程中沿用该特征，这样做能够使得避免特征重检测带来的多余时间成本和空间成本，同时 ORB 特征也有着一定的旋转和光照不变性，基本满足 SLAM 需求，它的极快速性使得 ORB-SLAM 算法具有巨大的实时性优势，ORB-SLAM 算法的一大巧妙性创新是首次实现加入了环形闭合检测线程的三线程工作方式，每一线程结构流程清晰，便于扩展。

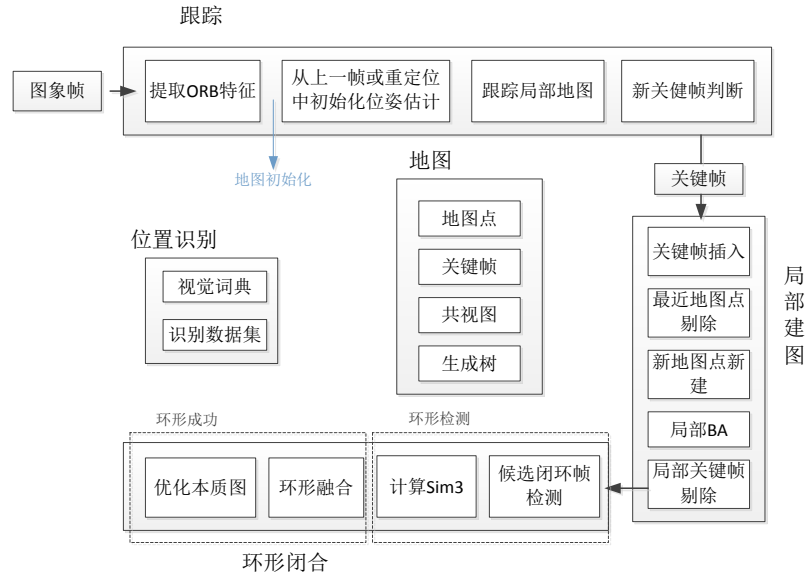


图 2-12 ORB-SLAM 算法框架

ORB-SLAM 算法框架主要包括跟踪、局部地图构建和环形闭合检测三个相互关联的线程以及用于闭环检测的场景识别与全局地图维持模块，三个线程均由对关键帧的不同处理相关联。

### 2.5.2 跟踪实现

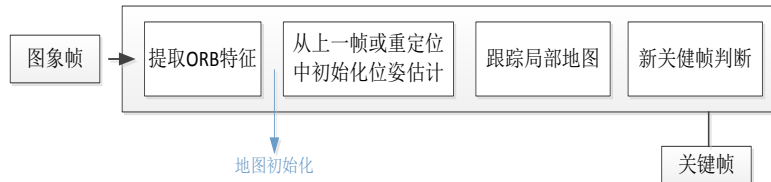


图 2-13 跟踪线程框架

跟踪线程承担定位相机在产生每一帧图像时的位置，同时决定何时插入下一张新的关键帧。首先对当前帧进行 ORB 特征提取描述，并将最邻近一张关键帧与当前帧进行 ORB 特征匹配（暴力匹配），同时仅使用运动 BA 来优化此刻相机的位姿，如果跟踪发生丢失目标（可能由于出现突发性的干扰或噪声），场景识别单元将进行一次全局重定位，一旦有一个初始化的相机位姿和 ORB 特征匹配的估计结果，一个局部可视地图就会被重新获得，它通过利用被系统维护的相邻关键帧之间的共视图来实现，之后搜索与局部地图点匹配的像素点需利用重投影技术，再根据所有的匹配点来重新对相机此刻位姿进行 BA 优化。最后该线程利用关键帧判定条件（如设定插入关键帧时间阈值），决定当前帧是否可作为新的关键帧插入到局部建图线程。

上图给出了图象帧跟踪定位的执行框架，具体过程如下：



1) **ORB 特征提取** 如前述图像帧提取方法, 此处取尺度因子 1.2, 尺度空间层数为 8, FAST 角点提取数量控制在 1000 到 2000 之间, 且以划分网格内数量均匀分布, 网格数量依据所设阈值进行控制, 之后增加 FAST 角点的尺度不变性并增加描述子描述。

2) **地图初始化** 即初始化地图点云, 为之后的点云融合作为依据, 算法步骤包括两步:

(1) 查找初始对应关系:

将当前帧  $f_c$  与参考帧  $f_r$  进行 ORB 特征匹配, 要求匹配数量足够, 否则重置参考帧, 重新执行该步骤。

(2) 对两个模型并行计算:

在并行线程中计算一个单映射  $H_{cr}$  和一个基本矩阵  $F_{cr}$ :

$$x_c = H_{cr} x_r \quad x_c^T F_{cr} x_r = 0 \quad (2-27)$$

统一两个模型循环迭代的次数和用于迭代的匹配特征点对, 并在每次迭代时分别计算一个 SM 分值:

$$S_M = \sum_i \left( \rho_M \left( d_{cr}^2 \left( x_c^i, x_r^i, M \right) \right) + \rho_M \left( d_{rc}^2 \left( x_c^i, x_r^i, M \right) \right) \right) \quad (2-28)$$

$$\rho_M \left( d^2 \right) = \begin{cases} \Gamma - d^2 & \text{if } d^2 < T_M \\ 0 & \text{if } d^2 \geq T_M \end{cases}$$

式中  $d_{cr}^2, d_{rc}^2$  ——为匹配帧间的相互传递误差;

$T_M$  ——为判断匹配成功点的阈值。

(3) 选择模型:

当场景为平面时, 选择单映射作为重建的模型, 其优势是仅对二维图像处理即可获得正确的初始化同时当发生低视差时判断为无初始化条件, 初始化停止; 当场景为三维面时, 基于视差条件满足, 此时选择基本矩阵模型, 可用公式表示如下

$$R_H = \frac{S_H}{S_H + S_F} \quad (2-29)$$

式中  $R_H$  ——为判断模型选择条件, 阈值为 0.45。

(4) 运动和从运动中恢复结构

针对两种不同模型的选择, 通过相应的计算获取运动状态信息。单映射时, 根据不同的运动假设, 选择是否初始化以及何种初始化方案, 如果不具备初始化条件则返回第一步。当选择基本矩阵模型时, 利用校正矩阵和本质矩阵进行转换:

$$E_{rc} = K^T F_{rc} K \quad (2-30)$$

### (5) 捆集调整 (BA)

上述步骤完成后, 对所有变量进行全局 BA 优化, 得到初始化重构地图。

3) 初始化位姿估计 通过与相邻帧成功匹配, 利用运动模型可以估算相机当前时刻的相对位置, 如果匹配失败也即跟踪丢失, 需要进行相机的重定位, 此时利用构建的词袋模型以及其中分类存储的之前图象帧, 利用索引技术对当前帧的二进制向量寻找候选匹配帧, 之后计算它们之间的汉明距离确定是否为匹配成功帧, 对于成功匹配到的帧, 在通过 RANSAC 算法去除掉误匹点后利用 PnP 法求解此刻相机的相对位姿, 并进行 BA 优化完成相机位姿的初始化估计。

4) 跟踪局部地图 利用位姿估计结果建立地图与关键帧间的对应关系, 并进行对应关系扩展 (图像特征点与地图点云与之间)。由于地图点云数量的庞大复杂, 考虑只映射局部地图, 其包含一组与当前关键帧拥有共有地图点云且与相邻组  $K_2$  图像内容关联的关键帧  $K_1$ ,  $K_1$  中还包含一参考关键帧  $K_{ref}$ , 它与当前帧相似性最高。在当前帧中搜索对应于  $K_1, K_2$  中的点云, 步骤如下:

(1) 重投影对应于当前帧中的地图特征路标点  $x$ , 去除重投影中的外点;

(2) 计算夹角 (地图点可视方向平均值  $n$  与当前视图线  $v$  之间), 去掉这样的点: 满足  $v \cdot n < \cos(60^\circ)$ ;

(3) 计算距离  $d$  (地图点到相机光心之间), 去掉这样的点, 判断规则为

$$d \notin [d_{\min}, d_{\max}] \quad (2-31)$$

式中  $d_{\min}, d_{\max}$  ——为尺度空间中距离的上下界。

(4) 计算尺度 (图象帧中), 通过计算  $d/d_{\min}$  得到;

(5) 比较描述子  $D$  与 ORB 特征 ( $D$  代表地图特征路标的代表描述子, ORB 特征为图象帧中额未成功匹配点), 在预测尺度, 地图点投影点  $x$  附近, 关联具有最佳匹配的地图点。

最后对相机的位姿进行 BA 优化, 优化变量还包括图象帧中所有对应的地图点。

5) 新关键帧判断 针对当前帧, 利用设定好的限定条件判断, 并快速插入到另一线程, 由于这一过程使得跟踪的鲁棒性更强, 避免了不必要的图象帧 (冗余图象帧) 的处理, 筛选规则有如下四条:

(1) 距最近一次重定位之间有超过 20 帧图像通过;

(2) 下一线程无工作, 或是据最近一次关键帧插入时已产生超过 20 帧的图像;

(3) 至少有 50 个地图点被当前图像帧捕捉;

(4) 当前图像帧捕捉到的地图点数量少于帧  $K_{ref}$  捕捉的 90%。

### 2.5.3 局部建图实现



图 2-14 局部建图框架

该线程先是将判断的关键帧插入到序列并进行关键帧中的点云预处理建立匹配关系，之后对判断无效或是低观测率的点云进行剔除，再是对新关键帧或过往关键帧中的未匹配和建图点云进行三角化恢复以及一定的判断准则生成新的地图点，在对前述过程所涉及到的所有变量进行捆集优化同时删除无效观测，最后是删除被判断的冗余关键帧，将关键帧的数量维持在一定范围，步骤：

1) 关键帧入队 更新内容重叠的视图，添加节点  $K_i$  更新共点边，随后更新  $K_i$  在生成树上的链接，计算  $K_i$  的词袋向量，用于匹配特征点进而三角化恢复新的点。

2) 地图点云筛选 地图中的地图点需要特征明显，需多于三个关键帧时刻可被跟踪，且不受错误观测干扰，地图点判定条件：

(1) 地图点需一组帧可观测，超过 25% 的成员图像能够实际共视该点。

(2) 创建的地图点可被观测的关键帧数必须大于 3。

3) 新地图点产生 三角化共视图中位于  $K_c$  之间特征向量，可产生新的地图点。对  $K_i$  中未匹配的特征点进行匹配。将不满足对极几何约束的匹配点对删除。ORB 在随后的三角化获得新的地图点之后检查两视图的图像特性。如果一个地图点 3 个以上关键帧观测到，即可以将其映射到其它共视的关键帧。

4) 局部捆集优化 该过程用来优化本线程之前过程所涉及到的变量。随后弃用无效偶然观测。

5) 局部关键帧删除 局部地图构建将检测冗余的关键帧并将其删除。规则是删除  $K_c$  中所有 90% 的地图点可以至少被其他 3 个关键帧共视到的关键帧。

## 2.5.4 环形闭合检测实现

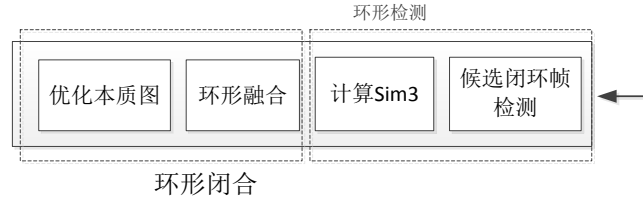


图 2-15 环形闭合检测框架

环形闭合检测实现的是对相机环形闭合运动时所得关键帧的相似判断和关键帧间的数据关联。该过程先是检测环形闭合检测的间隔周期是否达到，可以以中间关键帧的数量来判断（如，阈值设为 10），判断有效时，继续进行候选闭环帧检测，条件是当前帧与相邻帧间的相关判断计算低于某一阈值，之后是搜寻非相邻相似帧和与当前帧间的  $\text{Sim3}$  求解优化，得到此时刻相对于该帧的当前帧的相对位置，进行到这一步时关闭关键帧插入入口，并结合此关键帧特征点进行全局地图拼接、优化及更新。步骤可如下表示

1) 候选闭环帧检测 依据词袋向量值对关键帧  $K_i$  进行相似帧筛选，剔除低于相似度阈值  $s_{\min}$  和与其相连的关键帧。

2) 计算  $\text{Sim3}$  在单目 SLAM 过程中，所构建的地图可以进行空间 6 自由度翻转平移运动和尺度伸缩。因此，为了实现一个闭环，需要计算闭环处帧间相似变换矩阵，这即可显示出环形路径中的累计误差，亦可作为完成环形闭合运动的几何验证。

首先对对应于可能形成闭环的帧间地图点对应的 ORB 特征点建立对应关系，此时，当前关键帧与每个环形候选帧之间会存在一个三维空间转换关系。之后对匹配点对执行 RANSAC 迭代，再寻找相似转换矩阵。一旦经过之前 RANSAC 迭代之后还能存在相当多的成功匹配点并得到相似转换  $S_{ij}$ ，即对这些点对和相对关系进行优化，并搜索更多的匹配关系，再进一步利用这些被认定的正确匹配关系进行再次优化，直到认定环形候选关键帧  $K_j$  选取正确。

3) 环形融合 环形校正首先是对重叠的地图特征点拼接覆盖并在共视图中插入新的依附于环形闭合约束的边。一开始，经过形成环形闭合的始末帧间匹配配准后校正  $K_i$  的位姿  $T_{iw}$ ，再利用这种方式校正所有与  $K_i$  相邻的关键帧位姿，这样达到环形闭合边保持一致。所有的地图路标点位于环形关键帧中的和它们的邻近地图路标点被投影到帧  $K_i$  中，它们的邻近点和匹配点被搜索于以投影为中心的小范围区域内。之后对它们进行融合，包括在共视图中更新关键帧的边，这些边依赖于环形闭合约束。

4) 优化本质图 为了有效成功地实现闭环,在基于本质图的基础上执行位姿图的优化,这样能够把闭环运动过程中估计累计的误差分散到闭环位姿图中。在相似变换矩阵上进行优化并进一步校正尺度偏差。之后,更新地图点的三维信息。

## 2.6 本章小结

本章主要对 SLAM 问题做了简要的说明,之后介绍了 SLAM 外部环境传感器的分类,选择了基于视觉相机的 SLAM 算法分析和研究,介绍了当前最为流行的若干种特征提取算法,并进行了实验比较(实验结果见第五章),考虑到实际需求,ORB 算法成为被选方案,之后着重分析和研究了基于后端图优化并结合环形闭合检测的 SLAM 方案,最后本文选取了 ORB-SLAM 作为多机算法中的单机实现基础,并对该算法的实现框架流程做了细致的分析介绍。

## 第3章 多机器人通信与地图拼接

### 3.1 引言

本章承接上一章，依据上一章选取得单机方案，着重研究多机器人在初始时刻未知室内相对空间位置情况下如何通过相机间图象帧的特征关联进行相对位姿变换估计以及局部地图到全局地图的构建，主要包括两类可能，一是在机器人相互在对方相机视野下时的识别判断，并进行相对位姿变换求解，然后依此相对位姿关系对所构建的当前时刻单机局部地图进行拼接，中间过程还包括如何对地图重复区域的拼接去冗余；另一类是多个机器人在同时独立地进行 SLAM 过程时，虽没有相互视野内可见，但有经过相同识别场景时的相对位置判断，主要是基于图像特征的数据关联，实现相同场景观测时的相对位姿的求解，并进行时间路径累加到场景重合时刻下的机器人间的相对位姿关系，进而如同第一类情况进行地图拼接；本章还将对多机器人系统进行介绍，主要是不同系统之间的实现思路以及优缺点，并结合实际实验室条件约束选取方案实现。

### 3.2 多机器人系统

多机器人系统是实现多机器人间任务分配，调度和多机协同作业的基础，根据任务的执行需要或是任务的规模容量再或是环境条件约束等因素需选择不同的系统实现多机器人的任务功能，目前按照系统的控制实现方式可将多机器人系统分为三类，集中式、分散式和混合式。

1) 集中式 集中式系统主要由中央控制接收单元和下层的机器人执行单元构成，上层负责任务分解、分配、调度以及接受下层单元的即时信息反馈，做好协调工作；下层通过接受上层的任务信息以及调度安排完成自身控制执行的初始化，协同其他成员完成指定任务，并随时与上层通信报告所需状态信息，以及接受实时指令。集中式系统的优势是所有任务控制指令都由上层派发，下层机器人只需完成自身任务，这样方便任务的统一规划，且单个机器人相对构造简单，可以降低成本，且执行效率高，但是也很受制于通行环境的约束，上层单元的可靠性和计算性能也需得到保证。

2) 分布式 分布式系统的特点是没有中控单元，成员间通过通信进行必要状态信息交互，主要包括空间位置信息，健康状况以及各自功能信息等，并依此信息对系统任务进行分解，分配和协同执行，由于增加了单个机器人的决策功能，相

应机器人的软硬件结构更为复杂，单个成本较高，但由于这种任务协调机制，使得系统的稳定性更强，几乎不受单个个体的约束，且可根据不同的条件需求扩充或是减少系统成员，系统灵活可调。

3) 混合式 混合式系统即有上层控制接收单元也有下层成员间的信息交互，每个机器人既可以作为独立个体执行各自任务，也可以作为任务的发起者召集临近成员协同完成某项集体任务并报备中央控制单元。混合式系统具备上述两种系统的优势既保持了一致性，又不缺失灵活，但同样其成本也也是最高。

就多机器人 SLAM 问题本身的描述来说，机器人协同对一片区域进行环境探索，控制探索区域指令可由中央单元进行下达，机器人相互之间空间位置未知，当机器人出现相遇（机器人相机视野内可见）识别或是路过相同场地环境通过视觉技术可获得相互间空间位置，因此多机器人 SLAM 问题在室内应用时，由于相对位置较近，可选用集中式控制方案。由于本文的研究主要集中于该问题的协同 SLAM 上，对多机器人的协同控制拟采用遥控的方案，并结合实验室现有条件，选取了基于 ROS 系统的集中式相机节点控制通信方案。

### 3.3 基于 ROS 的节点通信设计

#### 3.3.1 ROS 操作系统

ROS (Robot Operating System) 系统是一种次级的用于机器人软件开发的开源操作系统，它集合了多种软件工具、开源库和连接约定用于辅助编写，其基于不同的 Linux 内核对应着不同的版本，可以支持多种编程语言独立开发，如 C++、python 等，为世界的开发者提供了一个开发、交流和共享平台，聚集了大量的共享开源功能包。ROS 提供了硬件抽象，设备驱动，各类函数库，可视化工具，信息传递机制，和软件包的管理等诸多功能。其核心是节点 (node，即进程)，一种可执行程序，具有分布式特性，节点间互不干扰也可联合通信（存在多种通信方式，话题、服务和动作）。

ROS 系统可分为三层：文件系统层，计算图形层和社区层。文件系统层主要由功能包(package)集构成，每一个包都定义了可执行的节点，依赖的库，数据集和配置文件等，同时还支持用户定义所需的消息数据类型和服务数据类型等。计算图形层是处理数据点对点的网络，计算图形层创建了 ROS 的运行网络，其结构成员包括节点(node)，消息(messages)，主题(topic)，服务(services)，节点管理器(master)，参数服务器(parameter server)以及有记录和回放功用的 bags 机制。其中节点是执行任务的程序，节点可以发布消息到话题上供给其他成员订阅，同时节

点和话题也在节点管理器上进行了信息注册，用于节点之间的通信交流，其中管理器只提供查询信息。社区层是 ROS 网络上进行代码发布的一种表现形式，主要是用来分享软件 and 知识，这为开发人员们提供了极大的交流共享便利。

本课题所使用的移动机器人 TurtleBot2，ROS 同样提供控制支持。

### 3.3.2 基于 ROS 的多机节点通信设计

多机通信是完成多机器人协同 SLAM 关键技术之一，对于多个机器人之间没有任何先验知识的情况（如，成员相对位置未知），此时机器人之间需通过必要的通信分享各自的信息，以便于建立成员间的联系。传统的依赖 ROS 系统的多机通信框架是通过一个节点管理器和参数管理器管理多个机器人之间的通信交流。

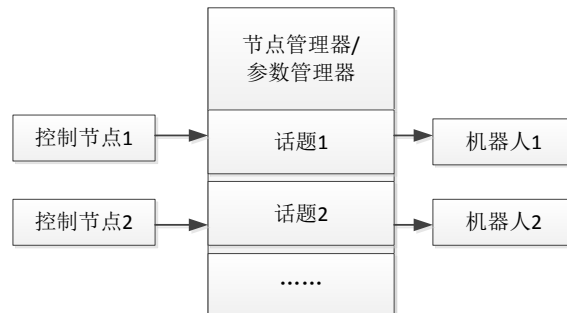


图 3-1 ROS 传统多机通信框架

其结构可如上图表示，针对本文设计要求（本文使用视觉信息和视觉词袋技术进行多机数据关联），把机器人 1 作为服务端（同时也是客户端），用来建立和存储多机器人团队视觉词袋库，其它机器人只作为客户端，将图象帧信息、机器人位姿信息和各自构建的局部地图数据打包装入 ROS 袋文件中，通过 TCP/IP 网络协议以话题的形式发送消息包，以供服务端随时订阅，服务端维护各个机器人之间的位姿信息，利用视觉信息把各自机器人之间的联系建立起来，并将拼接结果通过话题发布给被用于拼接的机器人订阅，示意图如下所示。

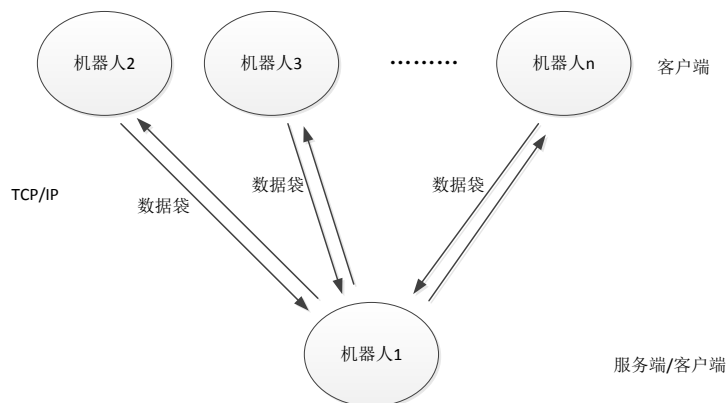


图 3-2 本文通信原理框架



针对本文实际实验情况节点使用方式为：两个 ROS 节点用于单目相机驱动程序，两个客户端节点用于各自的 SLAM 过程并用于话题消息的发布和订阅，服务端节点用于图像词袋库存储和检测并完成地图拼接以及消息的订阅与发布。

### 3.4 基于相机相对观测的地图拼接算法

#### 3.4.1 相机坐标模型

对于多机器人视觉系统（如多个机器人各自携带位置固定的相机）而言相机相对所乘载的机器人来说位置固定，它们之间的空间位置关系固定，且初始时刻已经确定，因此对于移动机器人空间模型建立可等价于相机坐标模型的建立。空间中任一机器人相机位姿信息可由一三维坐标向量和一归一化四元数来表示，其可表示为 $(r_i, q_i)$ （ $i$ 为机器人编号），三维向量表示相机相对前一时刻的相对空间位置信息，四元数表示相机的相对空间方向信息，对于整个多机器人团队来说，每个相机在初始时刻都将建立自身的局部地图坐标系（依照相机成像原理选用笛卡尔坐标系，以下简称为局部坐标系），团队的全局坐标系可由团队中某一机器人相机局部坐标系确定，如将机器人 1 的相机局部坐标系确立为团队坐标系。对于某个机器人而言，其局部运动状态可表示如下图 3-4（a），右图 3-4（b）表示的是在

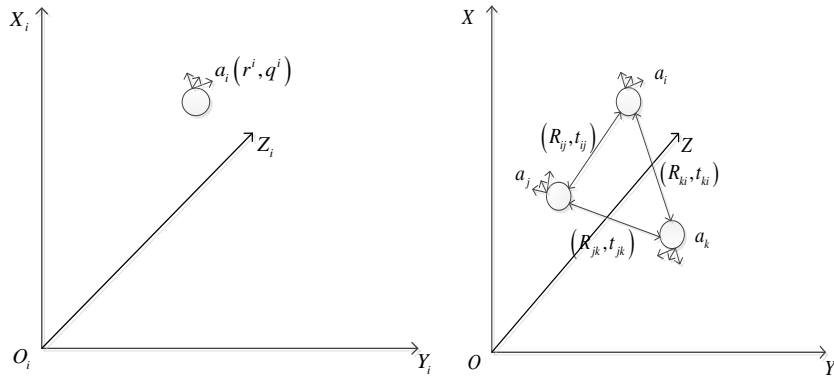


图 3-3（a）单相机坐标系

图 3-3（b）多相机坐标系

图 3-3 相机坐标系

团队坐标下多个相机间的位姿变换关系， $(R, t)$  分别表示相机间的相对旋转和平移，下标顺序为相对运动顺序。

#### 3.4.2 相机相对观测

当相机群某一成员  $i$  处于独立环境观测并进行 SLAM 过程时，在某一时刻  $t$ ，视野内观测到另一成员  $j$  并成功识别它时，两者间进行通信或是通过中央控制单元

（服务端）进行信息交互，上层单元对相机  $i$  和  $j$  当前时刻以及临近时刻的关键帧图像进行 ORB 特征匹配，并利用 RANSAC 算法去除匹配不一致点，之后根据成功匹配的特征进行相对位姿求解，本文采用 ICP 算法求解，ICP 算法是针对已获取各自坐标系下三维信息的匹配点对间的相机相对位姿的优化求解算法，为了便于计算，将采用齐次坐标代替解算。对于相机  $i$  在  $k$  时刻的相对（前一时刻）位姿可表示为  $(r_k^i, q_k^i)$ ，其中

$$\begin{cases} r_k^i = (x_k^i, y_k^i, z_k^i)^T \\ q_k^i = (q_{0,k}^i, q_{1,k}^i, q_{2,k}^i, q_{3,k}^i) \end{cases} \quad (3-1)$$

可将四元数  $q_k^i$  转化为相机  $i$  从前一时刻坐标系到当前坐标系的旋转矩阵形式，旋转轴依次顺序为  $Z-Y-X$ ，旋转表示如下

$$C_k^i = \begin{bmatrix} q_{0,k}^{i^2} + q_{1,k}^{i^2} - q_{2,k}^{i^2} - q_{3,k}^{i^2} & 2(q_{1,k}^i q_{2,k}^i - q_{0,k}^i q_{3,k}^i) & 2(q_{1,k}^i q_{3,k}^i + q_{0,k}^i q_{2,k}^i) \\ 2(q_{1,k}^i q_{2,k}^i + q_{0,k}^i q_{3,k}^i) & q_{0,k}^{i^2} - q_{1,k}^{i^2} + q_{2,k}^{i^2} - q_{3,k}^{i^2} & 2(q_{0,k}^i q_{1,k}^i - q_{2,k}^i q_{3,k}^i) \\ 2(q_{1,k}^i q_{3,k}^i - q_{0,k}^i q_{2,k}^i) & 2(q_{0,k}^i q_{1,k}^i + q_{2,k}^i q_{3,k}^i) & q_{0,k}^{i^2} - q_{1,k}^{i^2} - q_{2,k}^{i^2} + q_{3,k}^{i^2} \end{bmatrix} \quad (3-2)$$

将相机  $i$  在  $k$  时刻的相对位姿以齐次坐标表示，即

$$P_k^i = \begin{bmatrix} C_k^i & r_k^i \\ 0 & 1 \end{bmatrix} \quad (3-3)$$

对于此时刻 SLAM 过程所创建和维持的局部地图已转化为当前时刻坐标系下的局部地图，将提取匹配成功点对对应在各当前地图下的三维信息作为 ICP 算法的输入，ICP 算法的目标是对所有匹配成功的三维点对求取最优的转换矩阵  $T$ ，也即使得所有点对经转换后与配对点误差平方和最小，该算法需要一开启矩阵  $T_0$ ，之后的每次迭代对迭代结果选取一组距离最近的新点对组，并依此算取新的迭代矩阵，依次进行下去。在计算迭代矩阵时，将地图点三维坐标转换为齐次坐标表示，此时相对位姿变换矩阵可表示成

$$T_l = \begin{bmatrix} R_{ij} & t_{ij} \\ 0 & 1 \end{bmatrix} \quad (3-4)$$

式中  $l$ ——迭代次数。

利用相机内参模型获取配对地图点当前时刻各自坐标系下三维信息，对于配对成功的点对满足如下关系

$$\begin{bmatrix} p \\ 1 \end{bmatrix} = T_l \begin{bmatrix} p' \\ 1 \end{bmatrix} \quad (3-5)$$

式中  $p, p'$ ——为配对点对的三维坐标。

$T_i$  中的未知量共有 12 个，故最少需要 4 对匹配点对才可精确计算该变换矩阵值，在上述用于迭代计算的新点对组成员数量应不小于 4（一般为了避免匹配过程的偶然性，也即错误匹配，匹配点对的数量一般要求大于 15），ICP 算法流程图如下：

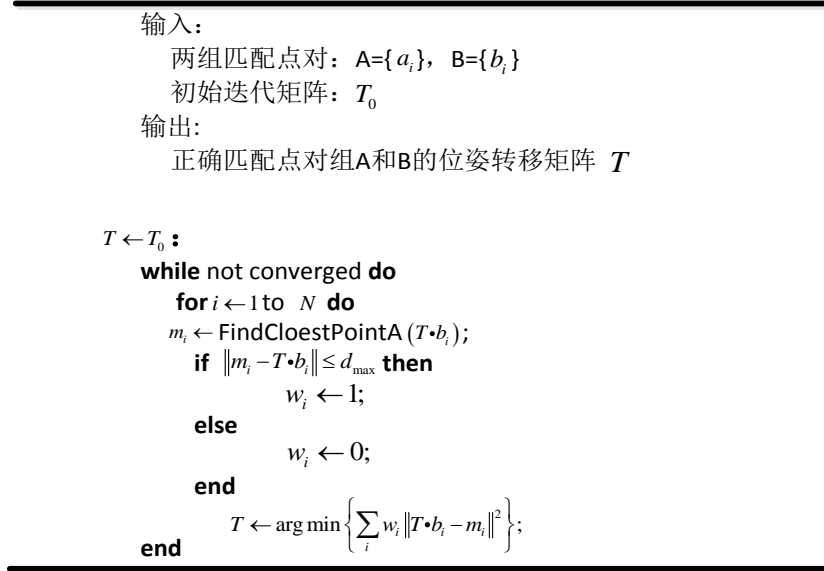


图 3-4 ICP 算法流程

由 ICP 算法可求得优化后的相机相对位姿变换关系  $T$ ，考虑到 ICP 算法仍然只是在寻找匹配点对中最优的求取转换矩阵的某组匹配点，并没有优化构建地图中的路标误差，这可能导致下一步地图拼接出现整体偏移或是冗余点过多，针对这种可能，本文利用 BA 算法对其进行进一步优化，将由 ICP 算法得到优化后矩阵作为 BA 过程的初始迭代矩阵，首先，将齐次矩阵  $T$  还原成形式  $(R, t)$ ，对式(2-13)做出修改

$$\begin{cases} s_1 \begin{bmatrix} x_1 \\ 1 \end{bmatrix} = M_{in}^i X \\ s_2 \begin{bmatrix} x_2 \\ 1 \end{bmatrix} = M_{in}^j (RX + t) \end{cases} \quad (3-6)$$

式中  $M_{in}^i, M_{in}^j$ ——分别为相机  $i, j$  的内参；

$s_1, s_2$ ——分别为匹配成功的某地图点对应的两相机坐标系下的深度值；

$x_1, x_2$ ——分别变为该地图点对应两相机像素坐标值。

BA 的优化目标过程可表示如下：

$$\arg \min_{R,t,X} \sum_{n=1}^N \left( \left\| s_1^n \begin{bmatrix} x_1^n \\ 1 \end{bmatrix} - M_{in}^i X_n \right\|^2 + \left\| s_2^n \begin{bmatrix} x_2^n \\ 1 \end{bmatrix} - M_{in}^j (RX_n + t) \right\|^2 \right) \quad (3-7)$$

式中  $N$  ——为两相机图象帧成功匹配的特征点对应的地图点。

可采用 L-M 法进行迭代计算，先对目标函数进行高斯—牛顿法求解得出迭代公式如下：

$$J^T J \delta x = -J^T \varepsilon \quad (3-8)$$

式中  $J$  ——为目标函数对所有变化量的 Jacobi 矩阵；

$\delta x$  ——为变化量的增量；

$\varepsilon$  ——为变量增量的同阶无穷小。

考虑到无法保证每次迭代时的误差变化方向，需对这种方向做出规定调整，此时加入最速下降法下降因子  $\lambda$  进行调整，公式 (3-8) 变为

$$(J^T J + \lambda \text{diag}(J^T J)) \delta x = -J^T \varepsilon \quad (3-9)$$

调整  $\lambda$  值使得迭代始终向着  $\varepsilon$  的减小方向的进行，该线性矩阵方程的信息矩阵具有稀疏特性，求解过程利用 Cholesky 分解可明显的减少高维矩阵巨大的计算量，最终求得 BA 值作为地图拼接的转换矩阵（齐次表示）。

### 3.4.3 地图拼接

基于相机相对观测的地图拼接过程发生在相机相遇时，拼接的地图以当前时刻某一相机 SLAM 过程地图坐标系为基准（如选取相机  $i$  所构建的地图），对于基于相机  $j$  所构建的地图路标特征，将其全部转移到相机  $i$  的地图中，转化拼接过程的数学表达为

$$X_n^{j'} = RX_n^j + t \quad (3-10)$$

式中  $n$  ——为成功匹配地图点标号。

转化完成后继续将该时刻局部拼接地图转化到相机  $i$  的地图局部坐标系下（齐次表示）

$$\begin{bmatrix} X_m^i \\ 1 \end{bmatrix} = P_1^i P_2^i \dots P_k^i \begin{bmatrix} X_m^{i'} \\ 1 \end{bmatrix} \quad (3-11)$$

至此完成此种情况下的地图拼接过程。

### 3.5 基于场景辨识的地图拼接算法

#### 3.5.1 场景辨识

机器人团队对未知环境的协同探索更多情况下相互之间很难发生相遇情况，但是由于机器人在行进过程中存在自身的闭环探索，同时在协同过程中必定会存在探索场景重叠，也就是说机器人之间一定会有经过相同场景的机会，对于此种情况下的机器人数据信息关联，并以此为依据进行局部地图拼接构建全局地图的现实意义会更加明显。针对这种情况下的多机器人协同 SLAM 的联系如何建立，本文采用视觉 SLAM 中常用的视觉词袋方案进行辨识，首先在协同 SLAM 的初始阶段载入视觉词典，视觉词典可由本文第二章所描述的进行训练得到，但限于条件限制，本文利用开源视觉词典包载入，即有单个机器人的词典载入也有上层控制单元的词典载入（上层单元可以是远程控制端也可以是团队中某一机器人），上层单元及时接收团队成员当前时刻的图象帧信息以及局部地图更新信息，并随时将对应机器人上传的图象帧进行 K 均值聚类生成二进制词袋向量，在词典树下编号和分类存储，图象帧的编号内容包含对应机器人在团队中的编号值，索引方式采用反向索引，为每一单词在图象帧中出现的可能赋予权重，示意图如下。

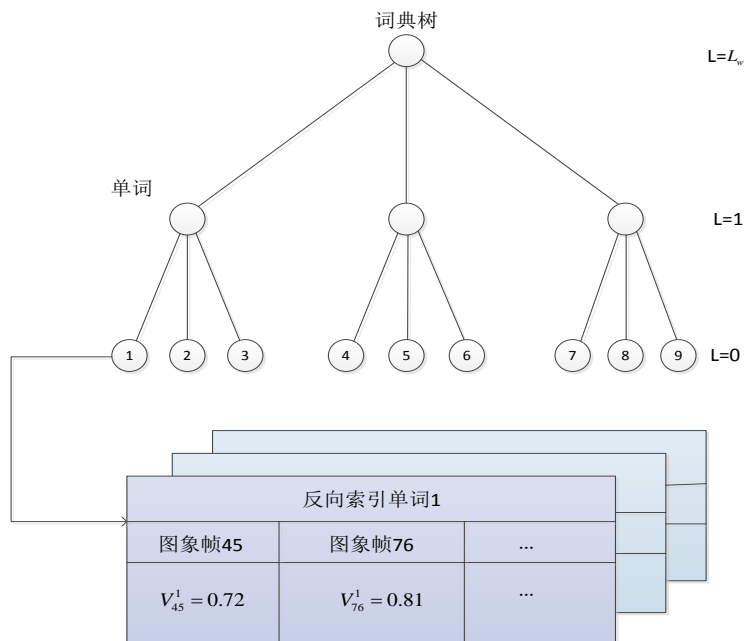


图 3-5 词典树与反向单词表

反向索引依据出现的单词找到对应的图象帧集，上层单元在完成图象帧存储的同时，也在对当前图象帧完成搜索相似帧的任务（主要是利用到索引技术和求取最小汉明距离的手段），当搜索到结果时，通过计算当前帧与候选相似帧集中的每一帧的汉明距离，进一步筛选相似帧集，将当前帧与汉明筛选图象帧集进行帧

间特征匹配(暴力匹配),找到匹配成功的图象帧集,并连接到对应编号的机器人,建立位姿相关联系,之后进行相对位姿求解过程。整个上层单元关联过程的流程表示如下

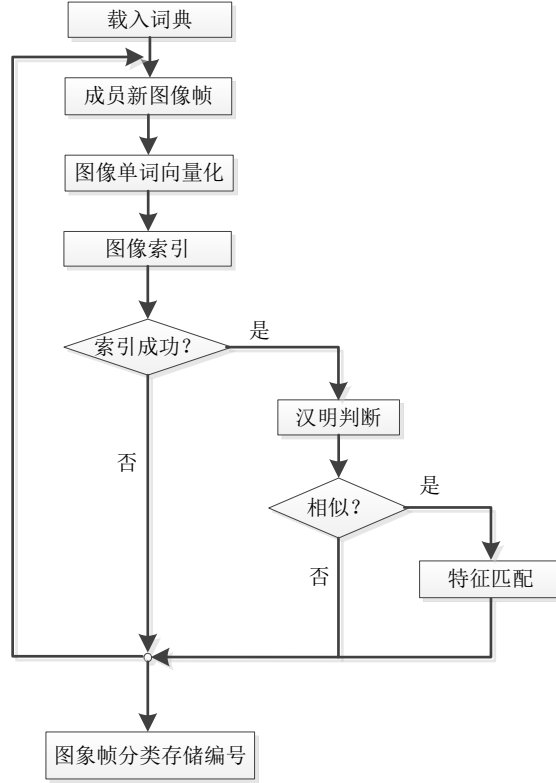


图 3-6 识别流程

### 3.5.2 相对位姿计算

本部分旨在研究如何进行位姿求解计算,对于整个团队该方法均能使用,为了便于分析,本部分只选取团对中两个机器人间的关系求解,并用相机间关系替代分析,显然此种情况下一台相机识别到相似场景时,另一相机并未在此处,此时刻建立的联系只是相机 1 与过去某一时刻相机 2 之间的联系,在此后的一段时间相机 2 又探索到了不同的环境场景,因此需要计算这段时间相机 2 的 SLAM 过程。假设在某一时刻 $t$ ,相机 1 到达相机 2 于之前 $k$ 时刻到达的某一处场景 A,相机 2 在 $t$ 时刻到达 B 地,设相机 2 此时刻自身的相对(前一时刻)位姿为 $(r_t^2, q_t^2)$ ,由 $k$ 时刻到 $t$ 时刻相机 2 的相对运动关系集合表示如下

$$\{(R_{k+1}^2, t_{k+1}^2), (R_{k+2}^2, t_{k+2}^2), \dots, (R_t^2, t_t^2)\} \quad (3-12)$$

齐次表示为

$$\left\{ \begin{bmatrix} R_{k+1}^2 & t_{k+1}^2 \\ 0 & 1 \end{bmatrix}, \begin{bmatrix} R_{k+2}^2 & t_{k+2}^2 \\ 0 & 1 \end{bmatrix}, \dots, \begin{bmatrix} R_t^2 & t_t^2 \\ 0 & 1 \end{bmatrix} \right\} \quad (3-13)$$

可令

$$T_i^2 = \begin{bmatrix} R_i^2 & t_i^2 \\ 0 & 1 \end{bmatrix}, i = k+1, k+2, \dots, t \quad (3-14)$$

相机 2 这段时间的运动累计可表示为齐次矩阵运算

$$T_{k \rightarrow t}^2 = T_{k+1}^2 T_{k+2}^2 \dots T_t^2 \quad (3-15)$$

则可求得相机 2 在  $k$  时刻时的局部坐标下位姿的齐次表示为

$$\begin{bmatrix} C_k^2 & r_k^2 \\ 0 & 1 \end{bmatrix} = T_{k \rightarrow t}^{2^{-1}} \begin{bmatrix} C_t^2 & r_t^2 \\ 0 & 1 \end{bmatrix} \quad (3-16)$$

在此过程中将基于相机 2 构建的局部地图从  $t$  时刻坐标转移到  $k$  时刻坐标系下，为之后的地图拼接做准备（地图拼接过程将在基于相机 1 的局部地图中完成），上述过程完成了相机关联时刻的场景统一。考虑到发生场景辨识时刻的真实情况，在计算两相机  $k$  时刻的相对位姿关系时，本文利用 P3P 方法对其进行求初始解，并结合 BA 优化提高解的准确度，得到优化后的解  $(R, t)$ 。

### 3.5.3 地图融合

地图融合的过程可能发生在任意时刻任意机器人（相机）之间，本文仍以上述相机 1 与相机 2 来作为分析对象。经上述过程计算出了相对位姿关系，并统一了两相机的坐标时刻，地图融合过程发生在相机 1（假设相机 1 为上层单元）地图的局部坐标中，将基于相机 2 构建的经转换后的局部地图利用相对位姿关系  $(R, t)$  与地图 1（基于相机 1 的地图）进行融合，地图特征点融合的齐次表达式表示如下

$$\begin{bmatrix} p_j^{2'} \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_j^2 \\ 1 \end{bmatrix}, j = 1, 2, \dots, m^2 \quad (3-17)$$

式中  $m^2$  ——为地图 2 中的地图特征数量。

融合后的地图 1 处于  $k$  时刻坐标系下，将其转移到基于相机 1 地图的局部坐标系下（齐次变换）

$$\begin{bmatrix} p_j^1 \\ 1 \end{bmatrix} = P_0^1 P_2^1 \dots P_k^1 \begin{bmatrix} p_j^{1'} \\ 1 \end{bmatrix}, j = 1, 2, \dots, m^1 + m^2 \quad (3-18)$$

式中  $P_0^1 P_2^1 \dots P_k^1$  ——为相机 1 以相对坐标累加表示的局部坐标的矩阵表示（可见上节）；

$m^1 + m^2$  ——为融合后的地图 1 特征数量。

经过以上过程最终完成从各自局部地图到全局地图的创建。本文选取相机 1

(机器人 1) 为上层单元 (服务端), 并考虑基于此种场景情况下的地图拼接算法流程框图表示如下

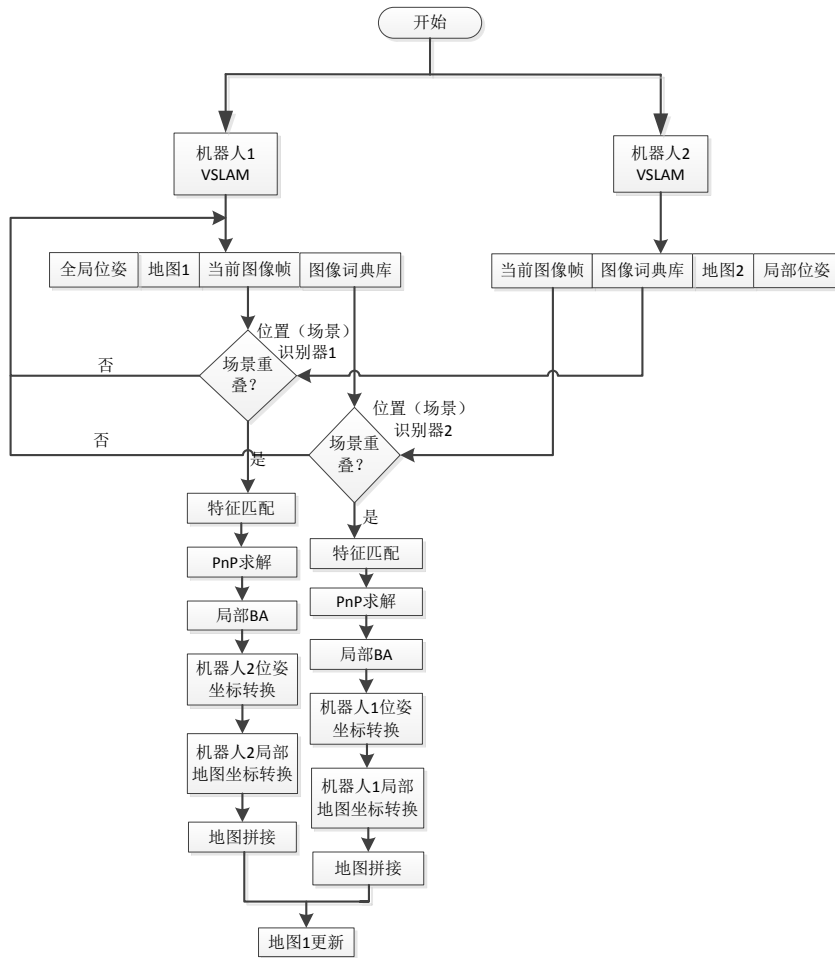


图 3-7 地图拼接框架

### 3.6 本章小结

本章先是介绍了多机器人系统按控制方式的分类, 给出了室内多机器人协同 SALM 过程的适用方案, 在分析了本文研究重点之后并结合现有条件约束, 选取了遥控的方案, 利用 ROS 平台进行多相机的协同 SLAM, 并基于此平台设计了多机节点通信规则。之后对协同 SLAM 的两种实现地图拼接的情况作出分析, 中间包括多机间的位姿图像信息数据关联和地图拼接过程中的地图时刻对齐, 并着重说明了第二种情况的现实意义, 最后给出了此种情况下到的地图拼接流程框架。



## 第4章 实验与结果分析

### 4.1 引言

本章是对前两章内容的实验验证分析，内容包括进行场地试验的硬件平台搭建，场地试验环境介绍，几种特征提取匹配实验及结果分析，ORB-SLAM 算法的数据集测试实验和误差分析，ORB-SLAM 算法的场地试验主要分析其建图性能，之后本章的重点是基于此算法的多机地图融合实验，并对实验结果进行评估，主要是针对地图拼接的一致性和拼接后环境轮廓复现效果的分析。

### 4.2 硬件平台搭建

根据实验目标和实验室条件约束，本文搭建了以一台 Turtlebot2.0 机器人搭载单目相机和手持单目相机组成的协同 vSLAM 系统，通过 ROS 系统利用上位机（笔记本电脑）通过 USB 连接的方式对机器人节点进行运动控制，两台笔记本之间通过 wifi 进行通信，本文算法适用于机器人运动状态未知，仅通过相机观测的图像信息进行相机（机器人）定位与建图，因此本实验的重点是保持相机的连续小幅运动性，这样可以避免经常性的跟踪丢失与重定位，本实验中需要对 Turtlebot2.0 机器人的运动幅度和角度进行限制。本文单目相机选取的是罗技 c525 单目摄像头，该摄像头是一款 RGB 摄像头，动态分辨率达  $1280 \times 720$ ，最大帧率为 30FPS，支持 USB2.0 接口，适用于笔记本。



图 4-1 Turtlebot2.0 与罗技 c525

本实验的机器人运动用一台笔记本电脑控制，两台笔记本中一台装 ubuntu14.04 系统并安装了 ROS Indigo，另一台装有 ubuntu16.04 和 ROS Kinetic，

同时两台电脑分别控制两个罗技摄像头，装有 ROS Kintetic 的电脑既作为客户端也作为服务端，另一台电脑只作为客户端，两者通过 wifi 进行通信。

### 4.3 实验及分析

#### 4.3.1 相机标定

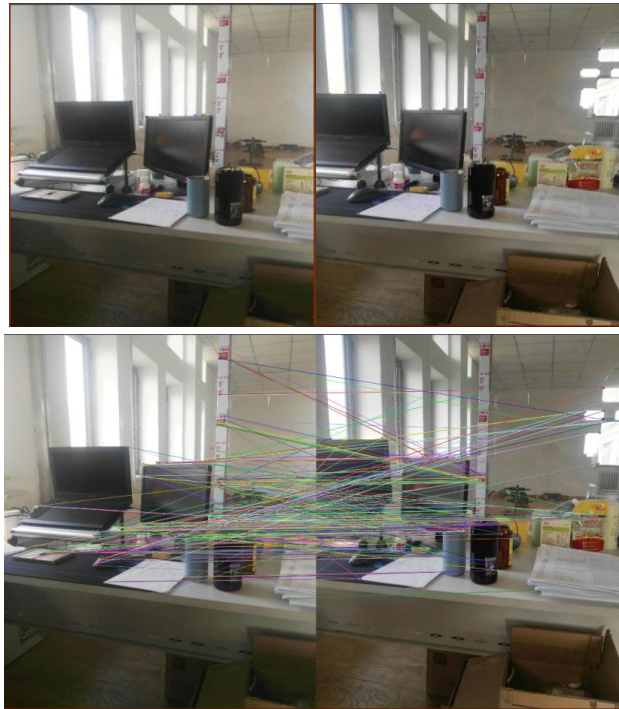
本文直接利用 matlab 标定工具进行单目标定，得到相机的内参矩阵如下（具体内参模型见式 2-6）。

$$M_{in}^1 = \begin{bmatrix} 1629.20740 & 0 & 639.50000 \\ 0 & 1747.48510 & 511.50000 \\ 0 & 0 & 1 \end{bmatrix} \quad (4-1)$$

$$M_{in}^2 = \begin{bmatrix} 1679.4286 & 0 & 674.50270 \\ 0 & 1793.3854 & 531.63824 \\ 0 & 0 & 1 \end{bmatrix} \quad (4-2)$$

#### 4.3.2 特征提取与匹配

本文对三种特征提取算法（SIFT、SURF、ORB）利用 OPENCV 库函数进行了实验，针对每一种算法分别在不同室内光照条件、带有不同角度和远近距离下对同一小范围场景进行了多次实验，比较了几种方法在尺度、旋转和光照情况下的特征数量、位置的变化大小和匹配的成功率（定性分析）。



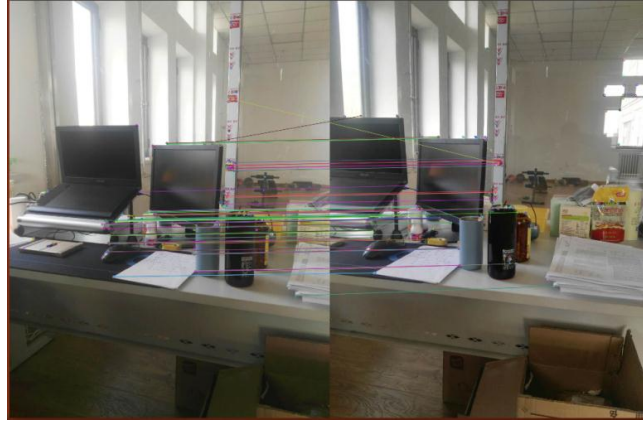


图 4-2 ORB 特征提取、匹配与 RANSAC 一致性检测

以上是某一次实验（ORB 特征算法）结果图，经过对实验结果的分析，得到如下结果

表 4-1 不同特征提取算法性能比较

种类	旋转不变	尺度不变	光照不变	重复性	时间效率	综合评定
SIFT	+++	+++	+	++	+	良
SURF	++	++	+	++	++	良
ORB	++	+	+	++	+++	优

本文是基于室内环境下的研究，光照情况变化并不强烈，三种算法都能满足实现要求，但 ORB 性能的时效性最好，在大规模运用情况下最具优势，且其他性能也基本能满足实现要求，故本文选取 ORB 特征提取算法作为处理图像的方案。

### 4.3.3 ORB-SLAM 数据集测试

本文主要研究基于单目相机的室内协同 SLAM 算法与实现，结合当前最为被广泛应用的几大数据集的不同采集方式和采集环境，本文选取 TUM 数据集进行测试，并将结果与数据集所提供的高精度真实测量值进行比较分析，评价过程包括各个坐标下相机运动轨迹与真实轨迹的比较即估计值与真实值的相似程度，由于数据集提供了高精度真实值，故本文采用均方根误差（RMSE，Root-mean-square Error）进行评价，公式如下

$$X_{RMSE} = \sqrt{\frac{\sum_{i=1}^n (X_{obs,i} - X_{model,i})^2}{n}} \quad (4-3)$$

式中  $X_{obs,i}$  ——为  $i$  时刻相机位置坐标的估计值；

$X_{model,i}$  ——为  $i$  时刻相机位置坐标的真实值。

用于测试的子集为 fr1/room，其采集环境为狭小的房间，相机在房间中心区域进行了一次回环运动，对房间的四周进行了图像采集，其对应时间戳上运动轨迹的估计值与真实测量值经 matlab 处理后表示如下

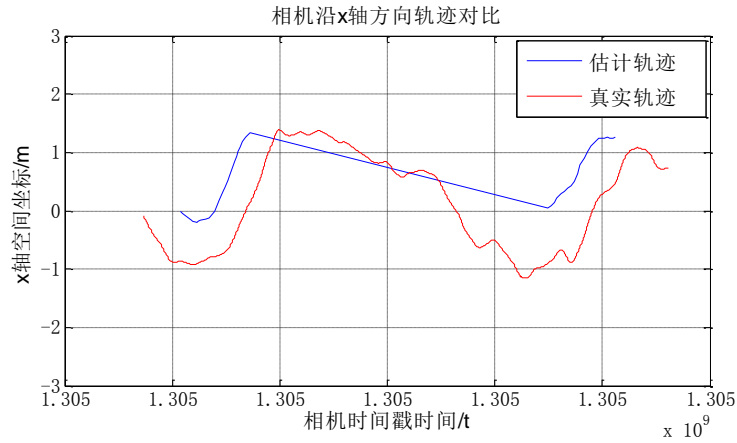


图 4-3 (a) x 轴轨迹对比

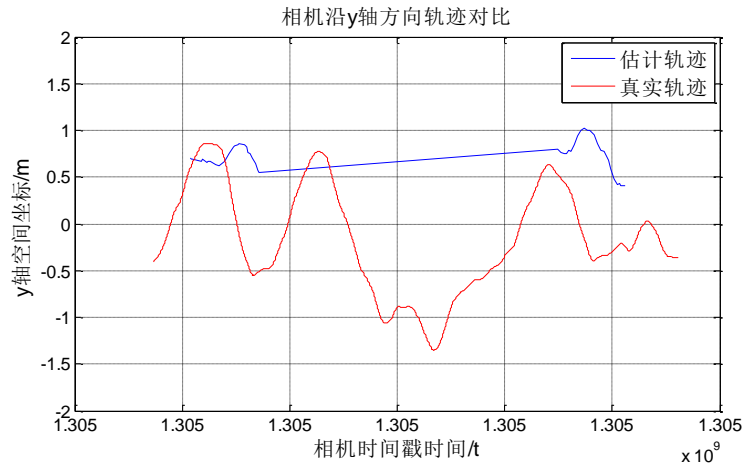


图 4-3 (b) y 轴轨迹对比

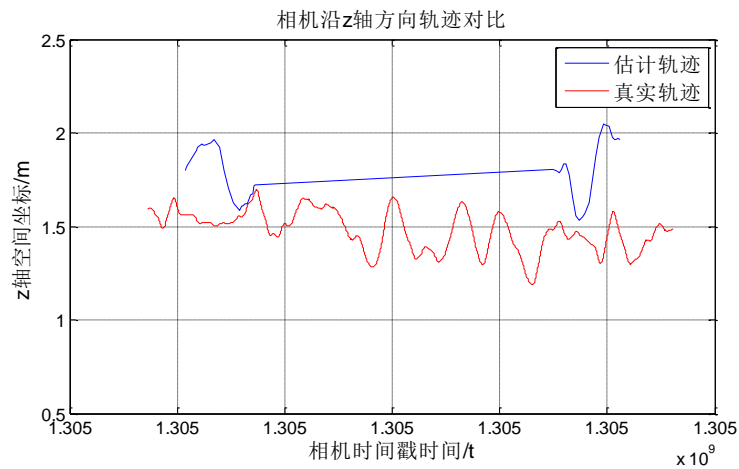


图 4-3 (c) z 轴轨迹对比

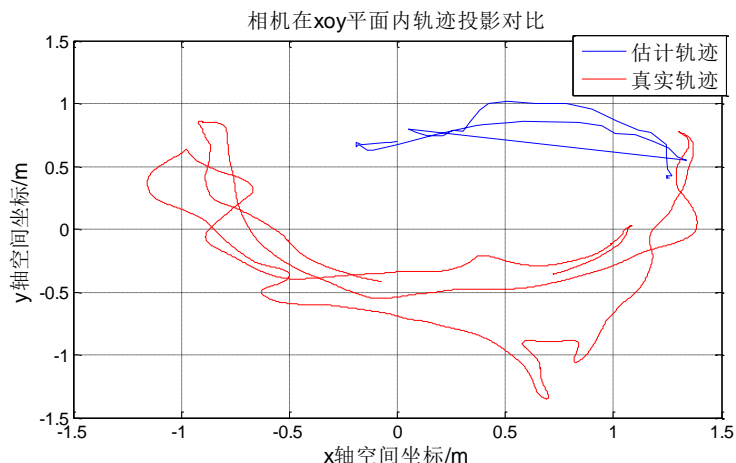


图 4-3 (d) xoy 平面轨迹对比

测试结果的评价指标结果如下表

表 4-2 数据集测试结果

地图点数	平均跟踪 时间/s	测试帧率 /fps	沿 x 轴 RMSE/m	沿 y 轴 RMSE/m	沿 z 轴 RMSE/m	跟踪成功 率
165	0.0736	5	1.0615	0.8224	0.3372	39.21%

由轨迹对比图和测试结果表可以看出，该算法基于单目的实现轨迹跟踪的实时性较好，可以实时跟踪，但是跟踪误差较明显，且由曲线图可以看出，该数据集测试下，出现了较长时间的跟踪失败。通过对测试过程的分析判断，沿 x 轴和 y 轴方向跟踪均方根误差比沿 z 轴方向大是因为相机主要进行的是 xoy 平面内的运动在高度上运动幅度较小，由于基于单目的 SLAM 实验实现缺乏尺度特性，其估计运动轨迹（如图 4-4 (b)）形状上与真值相似但与真实轨迹之间偏差较大，另外测试数据集中图象帧场景变化幅度较大且某一部分环境结构性相对较弱导致算法前端计算效率无法跟上，以及提取特征失败致使 SLAM 过程出现跟踪丢失。

经过分析，为了保证后续实验的有效进行，在实验过程中应保持相机在旋转运动时小幅进行，保证相机视角环境缓慢变化，在移动过程中应保持低速稳定运动且能够持续观测小幅变化的场景，考虑到光照变化也有可能造成跟踪丢失，在实验的过程中应调整好相机的曝光时间，避免突发性的白屏。

#### 4.3.4 ORB-SLAM 实验测试

本实验过程如下图，在经过某一结构性较强的物体时（本实验在途中放入了若干空纸盒）能够大致构建出场景轮廓，具有一定的场景复现效果，同时由运行过程信息显示可看出，新建的地图点云数量有限，对于地图拼接过程来说运算量相较于稠密点云地图负担较轻，故本文依此稀疏地图用于地图拼接实现，但同

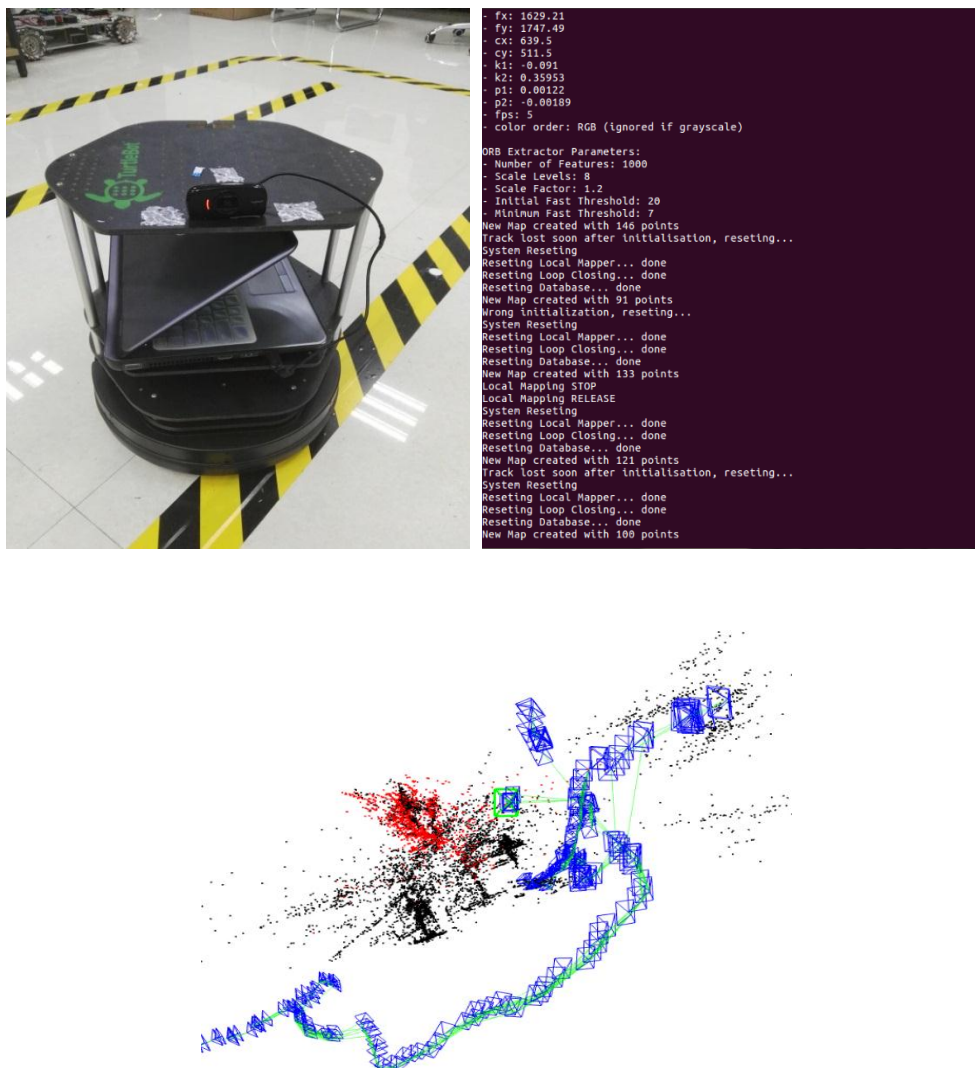


图 4-5 实验设备、过程及构建的地图

时可以看到单目的跟踪成功率依然受运动过程较大影响，一种可行的调整是在跟踪过程中当机器人稳定移动时保持相机能够转动，使其缓慢的从一处场景移动到下一处场景。

#### 4.3.5 地图拼接实验

本实验的目的是对地图拼接的效果做出评价，由于拼接是对稀疏点云的拼接，而不是栅格地图拼接，为了能够清晰的看到拼接的点云效果图，本文先对算法方案进行数据集测试，测试数据集选择 TUM 数据集，采用的数据集为办公桌的前后两端，先将数据集进行分割，分为桌前桌后两部分，并在边缘处保留足够重叠部分。本部分测试在一台笔记本上完成，为前述的装有 ROS Kinetic 的笔记本。创建的地图如下：





图 4-6 地图 1、地图 2 和拼接地图

地图 1 和地图 2 分别为两局部地图，点云拼接发生在两处重叠区域，地图 1 中检测到重叠（如图中的上方方框），服务端开始进行相对位姿求解和优化，当到达过道边缘时辨识到更多的重叠场景，增加了优化的优化变量，误差得到更多的均分，相对位姿解的精度得到提高，并随后完成了地图 2 到地图 1 的点云转换拼接。

可以看出拼接地图的拼接效果较为明显，对于环境中具有典型形状特征的人或物体轮廓的复现较为明显，拼接处衔接性较好，未出现较为明显的裂缝或是大量重叠点云，地图拼接的一致性较好，基本实现了算法的目标要求。

经数据集测试显示拼接效果较好，接着实验室测试，场景选为两隔断所围成的一个环形空间（如下图 4-7），其中环形一侧控制机器人缓慢行进并进行 SLAM 过程，另一面手持相机如同机器人一般，缓慢移动相机，直到检测到场景重叠部分，再在用于控制手持相机节点的笔记本上完成地图拼接过程，效果如下图 4-8：

如同数据集测试一般，在图 1 中完成了地图拼接，拼接重叠处发生在图 4-8 方框中，为了便于观察整体拼接效果，已将拼接点云图适当放大，局部物体轮廓缩小后会有一些不清晰，但仍能观看到一些大致轮廓，如右侧桌面上的书籍等。由拼接点云图可以看出拼接处衔接性较好，虽有点云重叠现象发生，但并不突出，



图 4-7 实验场景图

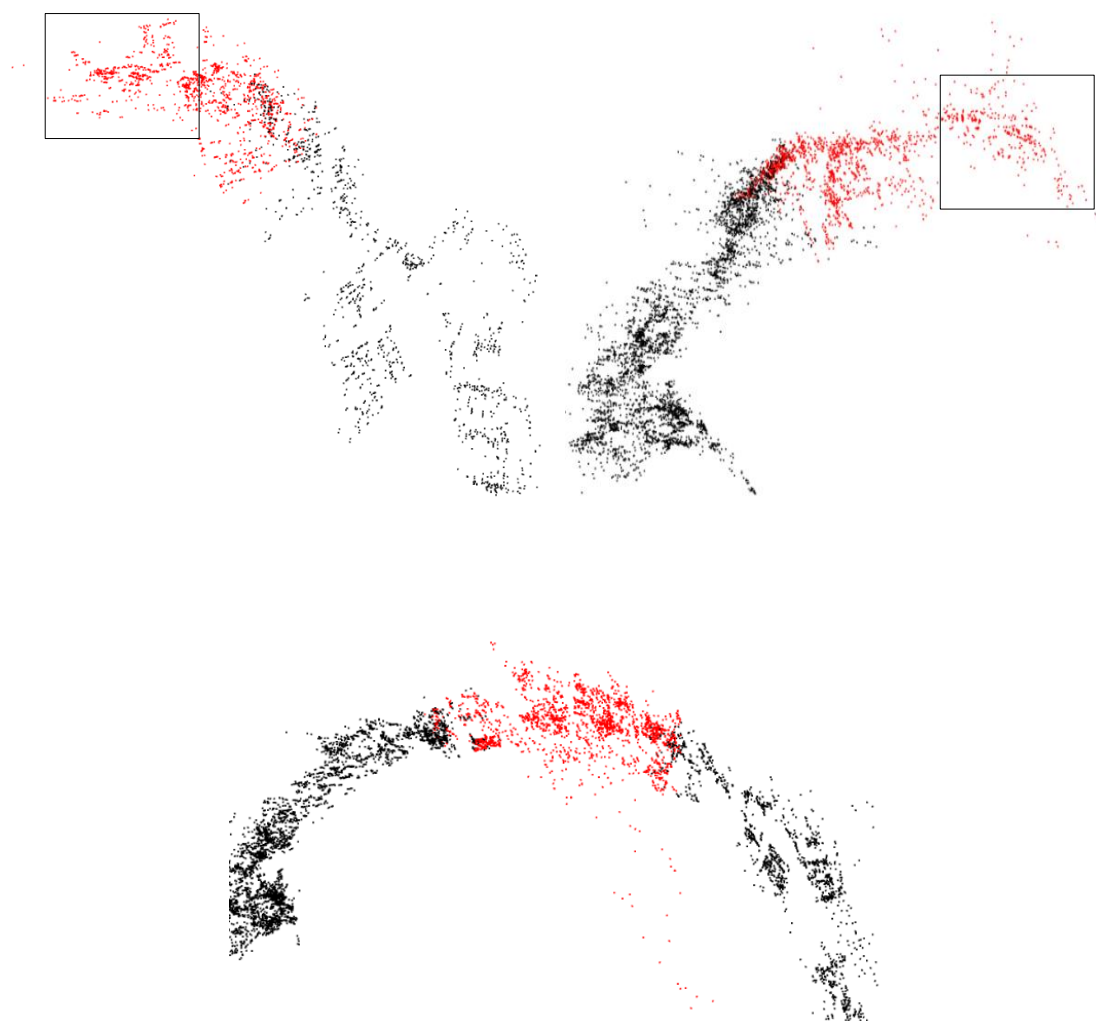


图 4-8 点云图 1、点云图 2 和拼接点云图

并未影响整体地图创建效果，这进一步验证了此算法方案可行性以及拥有较好的地图拼接一致性。但仔细观看时，仍能注意到目前还不能进行更进一步的任意景物的特性判断，只能对某些规则物体进行轮廓判断辨识，同时还可以发现，拼接点云图中拼接处以外仍存在一些冗余点，这可能是由于进行相对位姿求解时，某



些地图点坐标未能被优化处理，这些点处的累计误差未能得到进一步的重分配，从而在拼接时出现偏移。针对此问题，可以将局部 BA 优化扩展到全局 BA，不过这会加大计算量，可以通过使用高性能的上层单元进行此过程是一个合理的选择。

#### 4.4 本章小结

本章依据课题需求搭建了实验平台，首先对实验所用相机进行了标定，对三种典型特征提取算法进行了实验比对，结合课题实时性需求选取了 ORB 特征算法进行实验实现，对本文单机算法基础进行了数据集测试和实验，验证了单机算法的实时跟踪性能，并对算法的建图效果进行了分析，该算法具有不错的场景结构性复现，在地图拼接实验中，本文提出的方案可以较为有效的建立全局一致性地图，这为大规模稀疏地图的构建提供了一种可能。

## 结 论

本文的研究内容为对基于视觉的室内多机器人协同 SLAM 算法的研究和实现,在调研了目前 SLAM 发展的现状以及对 SLAM 问题的细致的描述和单机算法众多方案的介绍后,本文基于课题目标和实验室条件约束,选取了 ORB-SLAM 算法作为多机算法的单机基础,并依此向多机延伸,之后对多机的关键技术通信与地图拼接进行了研究,给出了一种可行的通信方案,对地图拼接的具体执行情况做出了分析和方案设计,最后进行了相关实验验证,本文主要工作可表示如下:

1) 介绍了 SLAM 问题的数学描述,对 SLAM 问题的基于外部传感器分类进行了比较,并选取了单目相机作为本课题研究的外部环境传感器,之后对 SLAM 算法的众多方案进行了研究,主要分为基于滤波的 SLAM 算法和基于平滑的 SLAM 算法,并重点研究了基于平滑方法的前后端理论,通过比较本文选取了 ORB-SLAM 作为单机研究基础,ORB-SLAM 算法主要分为三个线程,且三线程并行工作,包括位姿/地图跟踪、局部建图以及环形闭合检测,线程一完成对图像帧位姿的跟踪并判断何时插入新的关键帧,线程二主要对关键帧进行处理,完成局部地图的更新和并利用 BA 对局部地图进行优化,并删除旧的重叠度高的关键帧,线程三同样需要处理关键帧,目的是进行场景重识别,完成回环检测,并进行回环优化和传递。

2) 介绍了对多机器人系统的分类,选取了遥控的方式对机器人进行控制,并利用 ROS 系统对多相机节点进行控制,基于 ROS 系统设计了一种可行的多机通信方案,对于本科题,将某一机器人作为服务端其它机器人作为客户端通过 wifi 进行通信。之后详细研究了多机器人地图拼接的环境可能和拼接算法,主要分为基于相对观测(相机视野内可见)的多机位姿数据关联和基于此的地图拼接,以及基于场景辨识的地图拼接,其过程也分为数据关联和拼接两部,此种情况下场景识别利用的是视觉词袋技术,通过对图像词典库的索引配对找到与当前图像帧最为相似的候选场景帧并进行帧间匹配,利用 PnP 算法求解相对位姿并用局部 BA 进行优化,之后是多机拼接前的位姿坐标对齐和地图拼接,最后给出了此种情况下的算法流程框架,相较于第一种场景环境,此种情况更为普遍且有包含关系,因此本文主要对此种情况进行算法实现。

3) 搭建了硬件实验平台,对单机方案进行了实验验证,通过数据集测试,验证了单机算法的跟踪实时性,通过场地实验测试了算法的建图效果,评估了其作

为多地图拼接的可行性，本文构建的地图为稀疏地图，可以实现拼接的计算量需求，最后是多机地图拼接实验，本文的方案经验证可以较好的实现全局一致性地图的构建并能复现结构较为明显的环境特征，这为大规模稀疏地图的构建提供了一种可能。本文中仍存在问题有待解决和进一步研究：

1) 单目相机虽能进行实时跟踪，但容易出现跟踪失败且跟踪误差较大，这主要是单目相机无法直接获取景物深度信息造成的额外计算量并且缺乏尺度信息，可以利用双目相机取代单目进行实现。

2) 虽然 ORB 特征提取的快速性较好，但受限于特征点数的庞大数量，这仍对算法的实时性造成了影响，可以利用高性能 GPU 对算法进行加速。

## 参考文献

- [1] Leonard J J, Durrant-Whyte H. Mobile Robot Localization by Tracking Geometric Beacons[J]. IEEE Transactions on Robotics and Automation, 1991, 7(3):376-382.
- [2] Grisetti G, Stachniss C, Burgard W. Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling[C].IEEE International Conference on Robotics and Automation. IEEE, 2005:2432-2437.
- [3] Grisetti G, Stachniss C, Burgard W. Improved techniques for grid mapping with rao-blackwellized particle filters[J]. IEEE transactions on Robotics, 2007, 23 (1) : 34-36.
- [4] Fox D. Adapting the Sample Size in Particle Filters Through KLD- sampling[J]. International Journal of Robotics Research, 2003, 22(12): 985-1003.
- [5] Konolige K, Grisetti G, Kummerle R, et al. Efficient sparse pose adjustment for 2D mapping[C].IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2010: 22-29.
- [6] Kohlbrecher S, Stryk O V, Meyer J, et al. A flexible and scalable slam system with full 3d motion estimation[C].IEEE International Symposium on Safety, Security and Rescue Robotics. IEEE, 2011:155-160.
- [7] Hess W, Kohler D, Rapp H, et al. Real-time loop closure in 2D LIDAR SLAM [C].IEEE International Conference on Robotics and Automation. IEEE, 2016: 1271-1278.
- [8] Durrant-Whyte H, Bailey T. Simultaneous Localization and Mapping: Part I[J]. IEEE Robotics & Automation Magazine, 2017, 13(2):99-110.
- [9] Montemerlo M, Thrun S, Koller D, et al. FastSLAM: A Factored Solution to the Simultaneous on Localization Artificial and Mapping Problem[C].Eighteenth national conference on Artificial Intelligence. American Association for Artificial Intelligence, 2003:593-598.
- [10]Lu F, Milios E. Globally Consistent Range Scan Alignment for Environment Mapping[J]. Autonomous Robots, 1997, 4(4):333-349.
- [11]Dellaert F, Kaess M. Square root SAM: simultaneous localization and mapping via square root informationsmoothing[J]. International Journal of Robotics Research,

- 2006, 25( 12) : 1181-1203.
- [12]王忠立, 赵杰, 蔡鹤皋. 大规模环境下基于图优化 SLAM 的后端优化方法[J]. 哈尔滨工业大学学报, 2015, 47(7): 20-25.
- [13]Davison A J, Reid I D, Molton N D, et al. MonoSLAM: real-time single camera SLAM[J]. IEEE Transactions on Pattern Analysis&Machine Intelligence 2010, 29 (6):1052-1067.
- [14]Murray D , Klein G. Parallel tracking and mapping for small AR workspaces[C]. IEEE and ACM International Symposium on Mixed and Augmented Reality. IEEE, 2008: 1-10.
- [15]Engel J, Sturm J, Cremers D. Semi-dense Visual Odometry for a Monocular Camera[C].IEEE International Conference on Computer Vision. IEEE, 2013:1449-1456.
- [16]Engel J, Schöps T, Cremers D. LSD-SLAM: Large-Scale Direct Monocular SLAM [M].Computer Vision—ECCV 2014. Springer International Publishing, 2014:834-849.
- [17]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.
- [18]Mur-Artal R, Montiel J M M, Tardós J D. ORB-SLAM: A Versatile and Accurate Monocular SLAM System[J]. IEEE Transactions on Robotics, 2015, 31(5):1147-1163.
- [19]Fox D, Ko J, Konolige K, et al. A Hierarchical Bayesian Approach to the Revisiting Problem in Mobile Robot Map Building[M].Robotics Research. The Eleventh International Symposium. Springer Berlin Heidelberg, 2005:551-558.
- [20]Fenwick J W, Newman P M, Leonard J J. Cooperative Concurrent Mapping and Localization[M]. 2002.
- [21]Williams S B , Dissanayake G , Durrant-Whyte H. Towards Multi-vehicle Simultaneous Localisation and Mapping[C].IEEE International Conference on Robotics and Automation. Proceedings. ICRA. Washington, DC 2002:27-430.
- [22]Howard A, Sukhatme G, Mataric M J. Multirobot Simultaneous Localization and Mapping Using Manifold Representations[J]. Proceedings of the IEEE, 2006 , 94(7):1360-1369.
- [23]Lee H C, Lee S H, Choi M H, et al. Probabilistic Map Merging for Multi-robot RBPF-SLAM with Unknown Initial Poses[J]. Robotica, 2012, 30(02):205-220.

- [24]Saeedi S, Paull L, Trentini M, et al. Map merging for multiple robots using Hough peak matching[J]. Robotics and Autonomous Systems, 2014, 62(10):1408-1424.
- [25]Schuster MJ, Brand C, et al. Multi-Robot 6D Graph SLAM Connecting Decoupled Local Reference Filters[C].IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2015:5093-5100.
- [26]Isaac Deutsch, Ming Liu, Roland Siegwart. A Framework for Multi-Robot Pose Graph SLAM[C].IEEE International Conference on Real-time Computing and Robotics (IEEE RCAR). IEEE, 2016:567-572.
- [27]Sasaoka T, Kimoto I, et al. Multi-robot SLAM via Information Fusion Extended Kalman Filters[M]. Ifac Papersonline, 2016,49(22):303-308.
- [28]Jessup J, Givigi SN, Beaulieu A. Robust and Efficient Multirobot 3-D Mapping Merging With Octree-Based Occupancy Grids[J]. IEEE System Journal, 2017, 11(3): 1723-1732.
- [29]Baoxiao Zhang, Jun Liu, Haoyao Chen. AMCL based map fusion for multi-robot SLAM with heterogenous sensors[C].IEEE International Conference on Information and Automation (ICIA). IEEE, 2013:822-827.
- [30]罗荣华, 洪炳镨. 基于信息融合的同时定位与地图创建研究[J]. 哈尔滨工业大学学报, 2004, 36(05):566-569.
- [31]鞠玉江. 基于单目视觉的家庭机器人全局定位研究[D]. 哈尔滨工业大学, 2006.
- [32]苑晶, 黄亚楼, 陶通等. 基于局部子地图方法的多机器人主动同时定位与地图创建[J]. 机器人, 2009, 31(2):97-103.
- [33]钟进. 基于路径图优化的多机器人同步定位与制图研究[D]. 哈尔滨工业大学, 2014.
- [34]张宝先. 基于异构传感器的多机器人 SLAM 地图融合研究[D]. 哈尔滨工业大学, 2013.
- [35]秦野. 大尺度环境的多机器人视觉激光同步定位与制图研究[D]. 哈尔滨工业大学, 2016.
- [36]苑全德. 基于视觉的多机器人协作 SLAM 研究[D]. 哈尔滨工业大学, 2016.
- [37]徐德. 机器人视觉测量与控制[M]. 谭民, 李原, 第 3 版, 北京: 国防工业出版社, 2016: 035-040.
- [38]Cadena C, Carlone L, Carrillo H, et al. Simultaneous Localization And Mapping: Present, Future, and the Robust-Perception Age[J]. IEEE Transactions on Robotics. 2016, 32(6):1309-1322.

- [39]Triggs B, McLauchlan P F, Hartley R I, et al. Bundleadjustment—a modern synthesis[C].Vision Algorithms: Theory and Practice. Springer Berlin Heidelberg, 1999: 298-372.
- [40]Lowe D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision, 2004, 60(60):91-110.
- [41]Bay H, Tuytelaars T, Gool L V. SURF: Speeded Up Robust Features[J]. Computer Vision & Image Understanding, 2006, 110(3):404-417.
- [42]Rublee E, Rabaud V, Konolige K, et al. ORB: An Efficient Alternative to SIFT or SURF[C].IEEE International Conference on Computer Vision. 2011 , 58(11):2564-2571.
- [43]Rosten E, Drummond T. Machine Learning for High-Speed Corner Detection[C]. European Conference on Computer Vision. Springer Berlin Heidelberg , 2006:430-443.
- [44]Calonder M, Lepetit V, Strecha C, et al. BRIEF: Binary Robust Independent Elementary Features[C].European Conference on Computer Vision. Springer Verlag, 2010:778-792.
- [45]Nister D. Preemptive RANSAC for live structure and motion estimation[J]. Machine Vision Applications, 2005, 16(5):321-329.

## 攻读硕士学位期间发表的学术论文



## 哈尔滨工业大学学位论文原创性声明和使用权限

### 学位论文原创性声明

本人郑重声明：此处所提交的学位论文《基于视觉的多机器人室内协同 SLAM 算法的研究和实现》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：叶必鹏 日期：2018 年 6 月 26 日

### 学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

(1) 学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；(2) 学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；(3) 研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：叶必鹏 日期：2018 年 6 月 26 日

导师签名：夏红伟 日期：2018 年 6 月 26 日

## 致 谢

转眼间，两年的硕士学习和生活时间就要即将结束，还没收拾好行装整理好心情，就要跟母校说声再见，心中实在是感慨万千，在此我愿向所有给与我学习、生活等各个方面关心和帮助的老师、同学、师兄弟以及学校的其他工作人员表达我真诚的谢意。

首先要感谢的人是我的导师夏红伟研究员，夏老师严谨的工作态度，深厚的专业学识，广阔的科学视野以及敏锐的学术洞察力给我留下了深刻的影响并激励我朝着这个方向前进，在夏老师的悉心指导和热心帮助下，我的硕士生涯虽有坎坷但都顺利的通过了，尤其是硕士课题的研究，夏老师给了我很多深刻的意见，并在其帮助下完成了论文的撰写，我将无以为报，只有踏实努力的工作来报答老师的关心和指导。

感谢王常虹老师，钟佳朋师兄，李远伟师兄，窦赫暄师兄对我课题的指导以及生活上和未来方向上的指导和帮助，他们渊博的学识和丰富地生活经验给了我极大的帮助，为我树立了科研和生活的榜样，鼓励着我继续前进。

感谢我的室友刘文之、房东飞、张宇飞和赵文一，他们在生活和学习上给了我很大的帮助，使我能够伴随着愉快和放心继续前进，感谢惯导那些和我一起走过来这段时间的兄弟姐妹们，你们给了我信心和勇气面对挑战，我衷心的祝福你们能够有个好的未来。

感谢我的父母一直以来的养育和支持，你们的关心和支持是我最大的动力，我将继续努力，不负你们的希望。

最后感谢学校给我这样一个机会，使我既学到了知识又遇到了你们，祝愿哈工大能够继续高歌猛进，谢谢大家。