学校代码： 10663
学号：15010210811

贵 州 师 范 大 学

硕 士 学 位 论 文

基于 SLAM 的虚拟现实空间定位系统的研究与实现

Research and Implementation of Virtual Reality Spatial Positioning System Based on SLAM

专 业 名 称：计算机科学与技术

专 业 代 码：081200

研 究 方 向：计算机视觉

申 请 人 姓 名：刘家豪

导 师 姓 名：刘志杰

二零一八年五月二十八日

摘要

SLAM（Simultaneous Localization And Mapping）技术被认为是实现真正全自主移动机器人的关键，其中以相机作为传感器的视觉 SLAM 技术相比于激光雷达 SLAM 技术来说，对空间要求更低，能够有效解决移动机器人在纹理丰富的动态环境中的定位和构图问题，同时视觉 SLAM 具有低成本、消除累积误差以及重定位等优点。本文将视觉 SLAM 与虚拟现实技术结合起来，设计并实现了基于视觉 SLAM 的虚拟现实空间定位系统。本文主要工作内容如下：

首先从成本、响应时间以及适用环境等方面，分析比较了基于 RGB-D 传感器的视觉 SLAM 中几种深度测量技术之间的优缺点。一方面，采用飞行时间法技术的 RGB-D 传感器计算速度更快，计算精度更高。另一方面，此类传感器的成本低、有效范围大，应用更加广泛。综合比较后，本文选用 Kinect2.0 传感器完成对图像数据的采集工作。

然后对视觉 SLAM 前端基于 RGB-D 的视觉里程计算法进行了深入研究，详细阐述了特征点法以及直接法两种主流视觉里程计算法的优缺点，针对两种主流算法各自适用的不同环境条件，提出了基于图像特征的视觉里程计自适应算法，此算法能够根据图像特征的状态来选择适用的视觉里程计算法估计相机运动。

最后结合视觉 SLAM 以及虚拟现实技术，设计了基于视觉 SLAM 的虚拟现实空间定位系统，同时将基于图像特征的视觉里程计自适应算法应用到系统中端相机位姿的实时优化过程中，该系统实现了现实空间

与虚拟空间相机的同步定位。随后搭建相关的实验平台，验证了算法的实时性和鲁棒性，展示了系统的实验结果。

关键词：视觉 SLAM；虚拟现实；RGB-D；Kinect2.0；视觉里程计；

ABSTRACT

SLAM(Simultaneous Localization And Mapping) technology is considered to be the key to achieving real autonomous mobile robot, in which the visual SLAM technology with the camera as the sensor is lower in space requirement than in the laser radar SLAM technology, it can effectively solve the location and composition problem of mobile robot in the rich texture of dynamic environment, and visual SLAM has the advantages of low cost, elimination of cumulative error and repositioning. In this paper, visual SLAM and virtual reality technology are combined to design and implement a virtual reality spatial location system based on visual SLAM. The main contents of this paper are as follows:

Firstly, the advantages and disadvantages of several depth measurement techniques in visual SLAM based on RGB-D sensors are analyzed and compared from the aspects of cost, response time and applicable environment. On the one hand, the RGB-D sensor using time-of-flight technology has faster computation speed and higher accuracy. On the other hand, such sensors have low cost, large effective range and wide application. After comprehensive comparison, this paper selected Kinect2.0 sensor to complete the collection of image data.

Secondly, the visual odometry algorithm based on RGB-D for

SLAM front-end of visual SLAM is studied in detail, the advantages and disadvantages of the feature point algorithm and the direct algorithm of two mainstream visual odometry algorithms are described in detail, according to the different environment conditions applicable to each of the two mainstream algorithms, the visual odometry adaptive algorithm based on the image features is proposed, this algorithm can select suitable visual odometry algorithm to estimate the camera motion according to the state of image features.

Finally, combining visual SLAM and virtual reality technology, a virtual reality spatial positioning system based on visual SLAM is designed, and the visual odometry adaptive algorithm based on image features is applied to the real-time optimization of the mid-range camera pose. The system realizes the simultaneous positioning of real space and virtual space cameras. Afterwards, the relevant experimental platform was built to verify the real-time and robustness of the algorithm and demonstrate the experimental results of the system.

Keywords: Visual SLAM; Virtual Reality; RGB-D; Kinect2.0; Visual Odometry;

目录

第一章 绪论

1.1 选题的背景及意义

随着计算机科技的不断进步，计算机学科的迅速发展，移动机器人领域以及虚拟现实（Virtual Reality，VR）领域得到了丰富的理论支持和技术支持，尤其是近些年，传感器技术的持续更新换代为二者不断注入新鲜的血液。移动机器人在社会安全防护、生产生活、医疗服务、救援抢险及地理探测等许多行业领域中发挥着重要的作用[18]。如下图 1-1 所示，为移动机器人的应用示例。

（1）如图 1-1（a）所示，为“灵蜥”系列排爆机器人，它是国家“863”计划支持下的研究成果，目前已研制出 A 型、H 型等多种型号的机器人，分别应用于不同的排爆及探测场景，该系列机器人可以大范围应用于武警、公安系统，大量减少了人员伤亡的损失。

（2）如图 1-1（b）所示，为深圳市大疆创新科技有限公司设计研发的专业影视航拍无人机，适合高端电影、视频的拍摄取景。

（a）排爆机器人“灵蜥”

（b）航拍无人机

图 1-1 移动机器人的应用示例

移动机器人如何在未知的、复杂的现实环境中，实现高度智能化、精准化的自身定位以及构建实时的、完整的环境地图，是移动机器人领域正在研究解决的实际问题。人类可以通过眼睛去感知客观世界，理解自身周围的环境，判断自身的位置，计算机也可以通过相机等图像采集工具记录环境的信息，经过相关算法的处理后，理解图像的内容[51]。在这样的背景下，SLAM（Simultaneous Localization And Mapping）技术应运而生，SLAM 中文解释为“同时定位与地图构建”[10]，它在移动机器人领域占有重要位置，同时它也是该领域重要的基本问题之一。移动机器人

在没有实际环境先验知识的情况下，要进行精准的实时定位、实现自主导航、构建环境模型以及目标识别和追踪，需要依靠 SLAM 技术的支持。

人类社会持续的发展，伴随着人工智能和机器学习技术的兴起，虚拟现实技术再一次成为热点讨论话题。虚拟现实包含了“虚拟世界”和“现实世界”两层含义，它是计算机图形学领域中的前沿学科，严格来说虚拟现实是一种交互式的计算机仿真系统，创建和体验虚拟世界是虚拟现实技术的核心内容。虚拟现实作为一种新的人际交互方式，旨在提高人的认知能力，用户要对虚拟环境中的物体进行操纵，从视觉、听觉、嗅觉以及触觉等方面与虚拟环境进行多感知交互，并相互影响，需要借助相应的 VR 专业设备，从这个角度来说，虚拟世界就是特殊的现实世界[12]。现如今，虚拟现实技术在各行各业都有实际的应用，例如：军事训练、考古发现、娱乐活动、游戏开发、教育培训等方面，由于虚拟现实技术的加入，这些领域的工作模式以及交互模式正在逐渐改变。

本文研究的基于 SLAM 的虚拟现实空间定位系统结合了 SLAM 和虚拟现实技术，不仅解决了 VR 在使用过程中无法移动的技术限制，同时又增加了实时定位等功能，该系统能根据用户在实际环境中的自身定位，实现用户在虚拟环境中的同步定位，使得用户能在虚拟环境中“动”起来，让人的感知与虚拟环境的交互关系上升到一个新的层次，具有一定的应用价值，总而言之，将 SLAM 技术和虚拟现实技术相结合是富有前瞻性和挑战性的课题。

1.2 SLAM 概述及国内外研究现状

1.2.1 SLAM 概述

移动机器人在人类生活各方面的地位与日俱增，要实现移动机器人完全自主移动，这个过程要解决的首要问题之一，就是未知环境中持续更新的地图构建问题，怎样实现地图的实时更新，怎样判断自身的相对位置，是移动机器人学科无数科研人员和学者追求的目标。自从 1986 年 SLAM 诞生至今[54]，它一直都是移动机器人领域共同关注、共同讨论的热点问题，SLAM 是指：移动机器人携带相应的传感器，在不断运动的过程中估算自身的位置，同步建立环境的模型[34][45]。SLAM 技术旨在解决移动机器人在复杂化、非结构化实际环境中的智能化移动问题以及增量式地图的构建问题。从 SLAM 的定义可知，移动机器人在未知的环境中运动，它要解决两个基本问题，一是“自身定位问题”，二是“地图构建问题”，这两个问题的结果之间相互依赖，相互影响，只有知道了具体环境地图信息，才能计算自身的位置信息，同样的，只有在获知自身位置信息的基础上，才能构建未知环境的地图，这里所指的移动机器人的位置也就是自身携带的传感器的位置。所以，SLAM 技术对于移动

机器人领域来说，是一个值得深入研究的问题。如下图 1-2 所示，为移动机器人的 SLAM 过程。

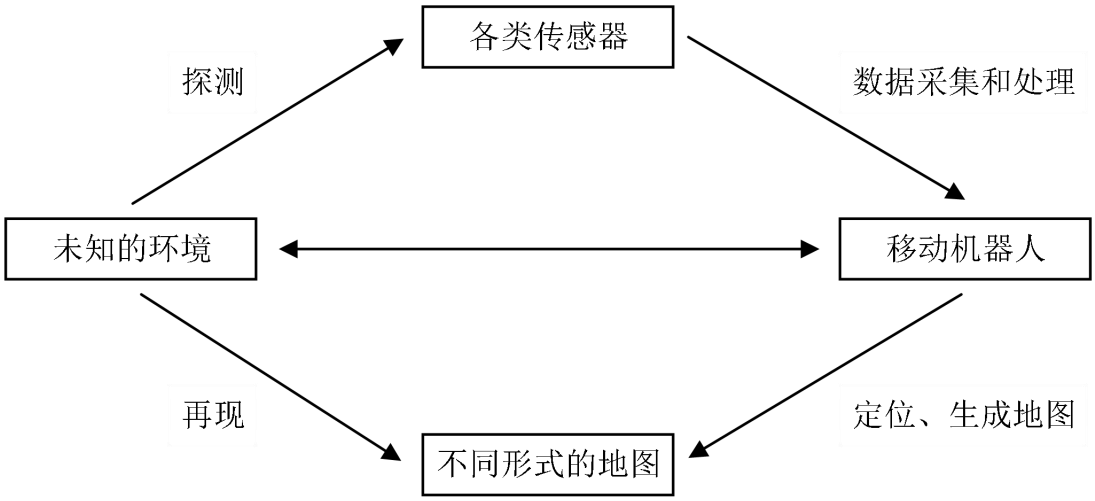


图 1-2 移动机器人的 SLAM 过程

1.2.2 国内外研究现状

SLAM 技术经过三十年的持续研究，逐渐在移动机器人领域取得了关键性的地位，国内外均有不少学者和优秀研究团队对其进行了大量的理论及实践研究工作，取得了许多重要成果和进展。

（1）国内研究现状

相比于国外来说，国内对于移动机器人领域的研究起步较晚，成果相对来说较少，尤其是在 SLAM 算法研究成果方面还是要大幅度落后于德国、美国等该方面起步较早的国家。

国防科技大学的季秀才博士通过从递增建图与定位、循环闭合以及多机器人 SLAM 中的地图合并这三类 SLAM 数据关联的子问题出发，根据 SLAM 数据关联的特点，将关联树的图搜索过程应用到具体的建模过程中，结合最小二乘的 SLAM 状态估计方法，提出了一种基于关联树的数据递增式完全 SLAM 算法，该算法可以在修正过去错误数据关联的基础上获得良好的状态估计[6]。

袁志千提出一种改进神经网络支持的基于 EKF 的 SLAM 算法，有效的减少了双目视觉条件下误匹配造成地图绘制的不准确性，并且通过实验证明了该算法的可用性，但是该算法计算量偏大，并不能满足实时性的要求[21]。

张文玲等人提出了一种基于强跟踪滤波器的自适应 SLAM 算法，该算法将无迹卡尔曼滤波（UKF）与强跟踪滤波器（STF）相结合，解决了由于 UKF 缺乏自适应调整能力而导致的低精度状态估计问题，具体原理是：将无迹卡尔曼滤波提取出的

每一个采样点，使用强跟踪滤波器实时更新，有效的减少了噪声对状态估计的影响，体现了基于强跟踪滤波器的自适应 SLAM 算法较强的鲁棒性以及较好的适应性[26]。

同时，国内高校等相关研究机构也展开了 SLAM 相关的理论和实践研究，如浙江大学、中南大学、哈尔滨工业大学、南开大学等，取得了一些有效的成果。

（2）国外研究现状

由于国外相对较早的将 SLAM 应用于移动机器人的定位运动估计问题中，有许多的研究成果，同时也涌现出了大量优秀的研究团队，如以 Nuechter 为首的德国不来梅国际大学研究团队，以 Newma 为首的英国牛津大学研究团队，以 Thrun 为首的美国斯坦福大学研究团队。

Henry 等人在文献[38]中最早提出采用 RGB-D 传感器对室内三维环境进行重建等方法，他们使用 SIFT 算法进行特征提取，再结合 RGB-D 传感器获取的图像深度信息，利用 RANSAC（Random Sample Consensus，随机抽样一致）[44]算法对特征点进行匹配，然后求解刚体的运动变换，将此基础上得到的初始值再结合 ICP（Iterative Closest Point，迭代最近点）算法[49]进行位姿估计，求解相机运动。

Newcombe 等人在文献[47]提出的 KinectFusion 算法依靠 GPU 工具构建环境的 3D 模型，算法的速度达到了 30HZ，基本满足了实时性的处理标准。同时，在 Newcombe 等人提出的系统中，在持续改变的环境光照条件下，他们仅使用一个价格低廉的深度相机以及一些普通的图形硬件设备，就能实时构造相对复杂的室内环境中任意场景的地图。该系统将 Kinect 传感器获取的深度图像数据流与全局曲面跟踪模型结合到一起进行分析比较，展示了系统构图的有限漂移和算法的高准确性，同时还能够进行实时的重建工作。但是 KinectFusion 算法也存在一些不足，如偏移噪声过大，缺少回环检测环节导致鲁棒性不强等。

1.3 虚拟现实概述及国内外研究现状

1.3.1 虚拟现实概述

虚拟现实（Virtual Reality，VR）属于计算机图形学的范畴，是一种新型的 3D 交互方式，用户通过专业的 VR 设备体验虚拟仿真空间，这种虚拟仿真空间是建立在人的感知认识上的，通过“看”、“听”、“摸”、“闻”等行为反馈的结果，模拟人对现实世界的观察来与虚拟世界进行交互，想象、交互、沉浸是虚拟现实的三大基本特征。虚拟现实技术最早诞生在上个世纪 40 年代的美国，最初由美国军方开发研究，主要集中应用于航空、航天等军事活动的模拟训练中，如飞机驾驶员、宇航员等日常训练任务中，随着世界的紧张局势得到缓解以及冷战的结束，美国缩减了

部分军用开支，虚拟现实慢慢地转变为民用技术，随着越来越多的机构及学者研究虚拟现实技术，VR 的基本内容逐渐增加，其影响也扩大到其他许多领域中，如：医疗、教育、交通等，改变了相关行业的传统交互方式[9][20][27]。随着互联网以及移动互联网的普及，虚拟现实技术走进了人们的生活中，虚拟现实产品的面世给人们的生活增加了许多乐趣。如下图 1-3 所示，为虚拟现实技术原理图。

图 1-3 虚拟现实技术原理图

1.3.2 国内外研究现状

虚拟现实技术的产生较早，但是当时的应用范围小，硬件设备相对落后，限制了虚拟现实技术的发展，随着计算机技术，尤其是传感器技术的更新换代，使得虚拟现实技术的发展速度显著加快，技术不断革新。如下图 1-4 所示，为虚拟现实技术示例。

图 1-4 虚拟现实技术示例

（1） 国内研究现状

我国虚拟现实技术的相关研究相比于起步比较早的一些发达国家，无论是从理论基础还是研究成果来说，都是有一定差距的，但是随着国家对计算机科学的重视，以及社会各界人士对虚拟现实技术的关注，依靠国内许多研究机构和高校、无数研究学者的努力，越来越多的公司投资虚拟现实产业，国内在虚拟现实技术方面还是取得了一些不错的研究成果[1]，2016 年更是被称之为“VR 元年”。

北京航空航天大学计算机系在虚拟现实技术方面是国内高校的“领头羊”，相比于其它高校来说，它在虚拟现实技术方面研究较早、成果颇丰，也是最具权威的单位之一，其自主研发的虚拟现实应用平台能为飞行员的训练提供演示环境以及实时的三维动态数据库，同时，其在虚拟现实技术方面的硬件设备、有关算法、实现方法等方面也有较多的研究成果。

哈尔滨工业大学计算机系主要研究人脸图像、表情、手势、头势以及语音在虚拟现实技术中的同步表达问题，目前已经解决了表情和唇动的技术难题。西安交通大学信息工程研究所致力于研究立体显示技术。另外，浙江大学、国防科技大学、上海交通大学等高校也做了很多虚拟现实技术的研究工作和尝试。

北京优联威迅科技发展有限责任公司成功研发实现了国内第一套动作捕捉系统，该公司研发的数据手套、虚拟环境的力反馈等系统也是国内领先的。

（2） 国外研究现状

在美国，虚拟现实技术取得了令世界瞩目的研究成果，美国宇航局（NASA）和美国国防部在上世界 80 年代进行了一系列的虚拟现实技术研究工作，先后建立了航空 VR 训练系统、卫星维护 VR 训练系统、空间站 VR 训练系统以及全国教育 VR 系统等[19]。

在日本，虚拟现实技术在游戏方面的应用全世界领先，同时，它有大规模的 VR 知识库。日本东京大学在远程控制、虚拟全息系统的研发方面处于世界领先地位。2004 年，日本奈良尖端技术研究生院开发了一种嗅觉模拟器，当在虚拟环境中拿起水果闻时，会闻到水果的香味，这是虚拟现实技术与嗅觉感知结合的一项重大突破。

在欧洲，德国将虚拟现实技术用于产品设计和产品演示等传统产业方面，降低成本，规避风险，同时还将虚拟现实技术用于工人培训。荷兰着力于研究完全沉侵式虚拟现实系统的研发。英国将虚拟现实技术用于科学可视化计算和建筑行业中。

1.4 本文的主要工作及组织结构

本文的主要工作是将 SLAM 技术与虚拟现实技术相结合，SLAM 技术是移动机器人实现自主移动的关键性技术，视觉 SLAM 是 SLAM 的一个重要的分支，视觉

SLAM 中特征提取与匹配等相关的前端视觉里程计工作更加和计算机视觉关联紧密，它通过相机传感器采集目标场景的图像序列从而实现自身的位姿估计，实现了相机在现实空间中的定位，同样的也可以推理到虚拟空间中去。本文从深度测量技术出发，介绍了相关 RGB-D 传感器的工作原理，并采用 Kinect2.0 传感器进行图像信息的获取工作，建立相应的图片信息库数据集。

在视觉 SLAM 框架中，视觉里程计主要用于相机运动估计以及局部地图构建两项内容，本文从视觉里程计的基本定义出发，介绍了视觉 SLAM 前端视觉里程计定位估计的相关算法，并讨论了两种主流视觉里程计定位估计算法的优缺点。在此基础上，结合两种主流视觉里程计定位估计算法的特点，提出了基于图像特征的视觉里程计自适应算法，详细阐述了算法的主要思想、算法流程、实现步骤以及实验验证，并将此算法应用到本文设计的基于视觉 SLAM 的虚拟现实空间定位系统中，验证了算法的有效性和稳定性。

本文主要分为五个章节：

第一章：绪论，主要介绍选题的背景及意义，对 SLAM 技术和虚拟现实技术进行了具体阐述，并对两者在国内以及国外的研究现状进行了举例分析，最后介绍了本文的主要工作以及组织结构。

第二章：基于 RGB-D 传感器的视觉 SLAM 技术介绍。详细阐述了深度测量技术的原理和分类，同时详细介绍了 Kinect1.0 和 Kinect2.0 两种 RGB-D 传感器的工作原理以及组成结构，并展示了相关的数据采集示例，介绍了经典的视觉 SLAM 框架，同时将视觉 SLAM 问题用数学公式做了具体的表述。

第三章：基于 RGB-D 的视觉里程计算法。从视觉里程计的基本定义出发，介绍了特征点法以及基于像素信息的直接法两种主流的 RGB-D 视觉里程计定位估计算法，并对它们的优缺点进行了具体讨论。

第四章：基于视觉 SLAM 的虚拟现实空间定位系统设计与实验。首先，将系统设计分为前端、中端、后端三个主要功能模块。然后，针对特征点法和直接法两种主流的视觉里程计定位估计算法各自适用的不同特征环境，提出了基于图像特征的视觉里程计自适应算法，并且将该算法应用到系统中端相机位姿的实时优化过程中去。最后，搭建相关的实验平台，验证了算法的有效性和稳定性，同时展示了系统的实验结果。

第五章：总结与展望。对本文的所有工作内容进行了总结与分析，提出了相关的不足之处，并且准备在下一步研究工作中进行改善。

第二章 基于 RGB-D 传感器的视觉 SLAM 技术介绍

2.1 引言

移动机器人在未知环境中的运动缺少先验知识，得不到正确的引导，它必须通过传感器采集实时的、精确的环境信息，在此基础上做相应的计算，将自身的位置估计融合到环境地图中，更加准确的描述自身所处的实际环境[25]。

视觉 SLAM 是 SLAM 技术中一个极其重要的分支，它将相机作为移动机器人的传感器观察现实世界，通过相机拍摄的一系列连续变化的图像作为依据，经过相应的图像数据处理、分析、比较、计算等过程，得到相机的运动，即移动机器人的运动，同时了解并且记录周围的环境情况[14]。

RGB-D 传感器采集的图像数据包含了基于普通三色通道的彩色图像，同时生成与彩色图像相对应的深度信息图像，深度测量技术为 RGB-D 传感器的这种工作方式提供了很好的支持，深度测量技术的进步，使得 RGB-D 传感器衍生出了更多的种类，扩展了更多的功能，推动了基于 RGB-D 传感器的视觉 SLAM 研究与创新。

2.2 深度测量技术

2.2.1 深度测量技术分类

如下图 2-1 所示，为深度测量技术的具体分类。

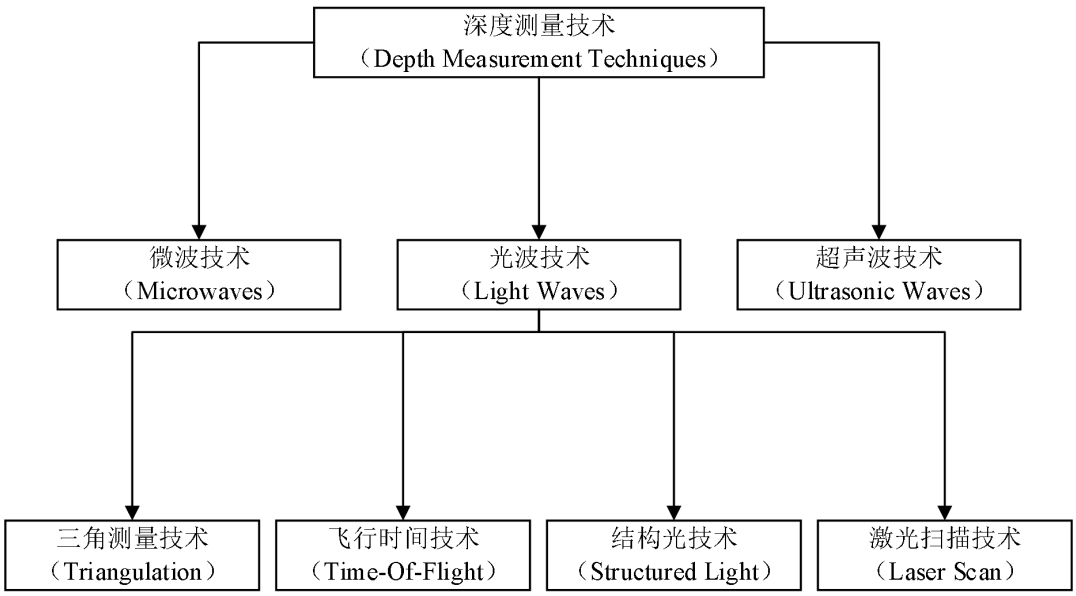


图 2-1 深度测量技术的具体分类

深度测量技术应用广泛，发展迅速[4][17]，目前深度测量技术主要分为三大类：微波技术、光波技术、超声波技术。其中光波技术分为四种应用：①三角测量技术，如双目相机；②飞行时间技术，如 Kinect2.0；③结构光技术，如 Kinect1.0；④激光扫描技术，如三维激光扫描仪；

2.2.2 深度测量各方案之间的比较

由于深度测量各个方案之间采用的方法不同，所以各自的实际应用表现也不相同。如下表 2-1 所示，为深度测量各方案之间的比较。

表 2-1 深度测量各方案之间的比较

	双目立体视觉方案	结构光法方案	飞行时间法方案
基础原理	双相机的图像处理	单相机和投影条纹斑点编码	红外光反射时间差
软件复杂度	较高	较高	较低
材料成本	较低	较高/适中	适中
响应时间	适中	较慢	较快
弱光环境	较差	较好（取决于光源）	较好
室外环境	较好	较差	不适用
有效精度	厘米（cm）	微米（um）~厘米（cm）	毫米（mm）-厘米（cm）
有效范围	中距离	超短距离（cm）~中距离（4~6m）	短距离（<1m）~长距离（<40m）
缺点	不适合昏暗环境	容易受光照影响	受限于环境条件
主要产品	LeapMotion，uSens，微动等	Intel（Omek），奥比中光，苹果（Iphone X）	Nimble，ThisVR，Kinect2.0 等

根据表 2-1 的数据分析可知，双目立体视觉方案只在材料成本上稍微占据优势。结构光法方案的响应时间较慢，同时在室外环境的表现较差。飞行时间法方案除了不适用于室外环境，其它方面都表现优秀，除了使用时不需要定焦之外，在响应时间和弱光环境方面也占据绝对优势。

2.3 RGB-D 传感器介绍

近几年，随着 Optrima、微软等厂商在 RGB-D 相机研发方面的投入开发，众多 RGB-D 相机产品出现在大众的视野里，其中，最引人关注的有微软分别在 2010 年以及 2014 年推出的 Kinect1.0 和 Kinect2.0 两款产品。RGB-D 相机这种新型的传感器相比于传统的相机来说，它能够采集到包含颜色信息的图像，也能获取每一帧上像素的深度信息图像[29]。Kinect1.0 和 Kinect2.0 都是作为 Xbox 360（微软的第二代

家用游戏主机）体感周边的外设推出的，它们是一种集动态捕捉，影音辨识等功能于一身的 3D 体感相机。

2.3.1 Kinect1.0 介绍

Kinect1.0 的主体结构包括：

（1）带内置马达的电机底座：调整俯仰的角度。

（2）RGB（彩色）摄像头：负责采集可视范围内的彩色图像信息。

（3）两个 Depth（深度）传感器：左侧和右侧的深度传感器分别为红外线发射器和红外线接收器，通过发射和接受红外线，从而能对整个室内环境进行立体定位。

如下图 2-2 所示，为 Kinect1.0 的结构图。

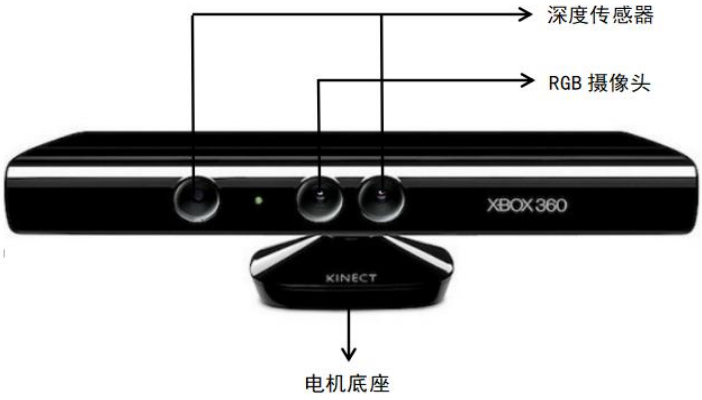


图 2-2 Kinect1.0 的结构图

如下图 2-3 所示，为 Kinect1.0 相机原理示意图。

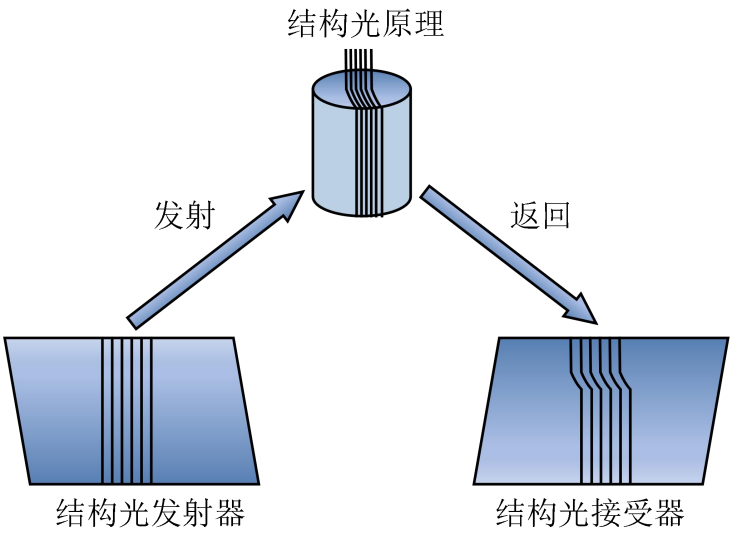


图 2-3 Kinect1.0 相机原理示意图

Kinect1.0 采用的是光编码（Lighting Coding）技术来获得环境中物体的深度信息，光编码技术利用光源照明给测量的目标空间编码，虽然还是属于结构光技术，但是它跟传统的结构光测量技术又有所区别，它的光源打出去的是一个具有三维纵深的编码，也就是说，整个三维空间中每一处的光源散斑图案是不同的，从而把整个空间都做了标记，这样就得到了空间中物体的具体位置信息[3]。

2.3.2 Kinect2.0 介绍

Kinect2.0 的主体结构包括：

（1）RGB（彩色）摄像头：负责采集可视范围内的彩色图像信息。

（2）Depth（深度）传感器：根据投射出的红外线脉冲反射回来的时间来取得整个环境中物体的深度信息。

（3）红外光线发射器：主动向场景中发射红外光线，在强光、弱光、甚至无光源的环境下都能探测到物体。

（4）四元线性麦克风阵列：并排四个线性麦克风采集声音，不仅能过滤大部分环境噪音，还能定位声源的具体方向。

如下图 2-4 所示，为 Kinect2.0 的外观以及内部结构图。

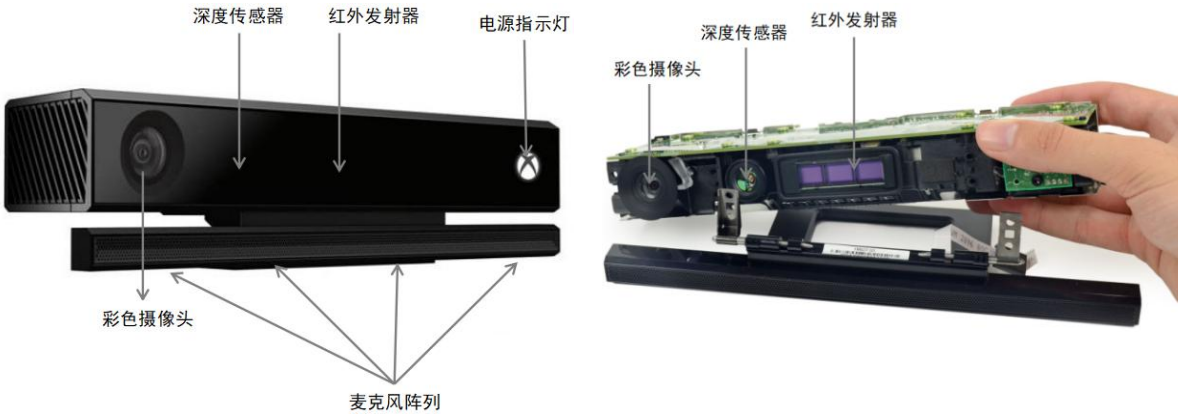


图 2-4 Kinect2.0 的外观以及内部结构图

Kinect2.0 采用的是 TOF（Time Of Flight，飞行时间法）技术来获得环境中物体的深度信息，可以理解为飞行时差测距，TOF 属于双向测距技术。TOF 技术将红外光作为光源，采用主动光探测方式，发出一道调制后的、强弱随时间变化的近红外光线，将光线发射和反射之间的相位差作为依据，计算场景中物体的深度信息，最后结合彩色摄像头采集到的场景中物体的颜色图像信息，得到物体的 3D 模型[11]。

如下图 2-5 所示，为 Kinect2.0 相机原理示意图。

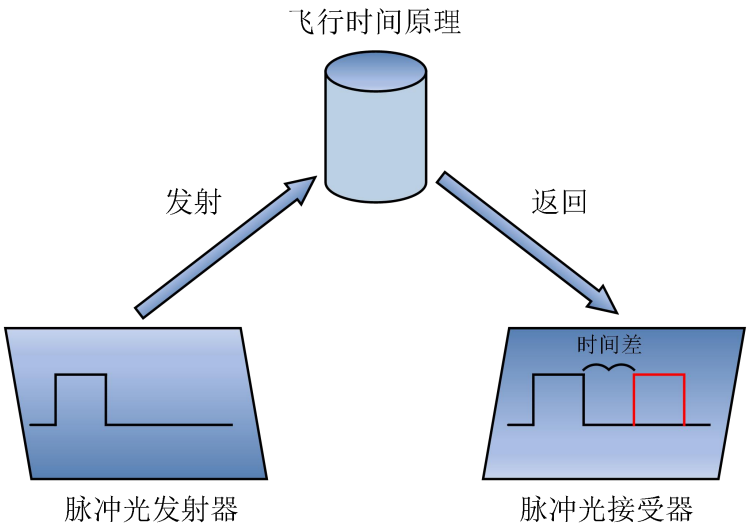


图 2-5 Kinect2.0 相机原理示意图

2.3.3 Kinect1.0、Kinect2.0 传感器配置比较

Kinect1.0 和 Kinect2.0 的主体结构不尽相同，两者在图像深度信息的采集原理和方式上不一样，Kinect2.0 不管是在硬件上，还是在软件上都要强于 Kinect1.0，它是 Kinect1.0 的升级版，下面主要从两个方面进行说明：

（1）分辨率方面。不管是颜色图像分辨率还是深度图像分辨率，Kinect2.0 都要高于 Kinect1.0，在颜色图像分辨率上，Kinect2.0 达到了 1080p 级别，可以说是进步非常大了。

（2）检测范围方面。由于 Kinect2.0 水平和垂直角度的增大，使得 Kinect2.0 的有效监测范围稍大于 Kinect1.0 的有效检测范围，范围的增加能拍摄到场景中更多的物体，得到更加丰富的图像深度信息。

如下表 2-2 所示，为 Kinect1.0、Kinect2.0 的传感器配置比较。

表 2-2 Kinect1.0、Kinect2.0 的传感器配置比较

		Kinect1.0	Kinect2.0
颜色（Color）	分辨率（Resolution）	640×480	1920×1080
	fps	30fps	30fps
深度（Depth）	分辨率（Resolution）	320×240	512×424
	fps	30fps	30fps
人物数量（Player）		6 人	6 人
检测范围（Range Of Detection）		0.8m~4.0m	0.5m~4.5m
角度（Angle）	水平（Horizontal）	57 度	70 度
	垂直（Vertical）	43 度	60 度

2.3.4 Kinect2.0 的优势及图像数据采集示例

Kinect2.0 采用的 TOF 技术与 3D 激光传感器的原理基本类似，3D 激光传感器是逐点扫描，TOF 技术则是通过探测光脉冲的飞行时间来得到目标物体的距离，同时得到整幅图像的深度信息。

（1）Kinect2.0 的优势。在图像信息采集方面，由表 2-2 可知 Kinect2.0 比 Kinect1.0 优秀。Kinect2.0 与传统的双目立体相机比较也具有很多优势，如下表 2-3 所示，为 Kinect2.0 与传统的双目立体相机之间的优缺点比较。

表 2-3 Kinect2.0 与传统的双目立体相机之间的优缺点比较

	传统的双目立体相机	Kinect2.0
体积、重量	往往个头大，且偏重	与一般相机相差无几
计算速度	较慢	较快
计算精度	受环境特征影响相对较差	非常准确

Kinect2.0 在体积、重量方面比传统的双目立体相机轻便。Kinect2.0 能实时快速的得到图像的深度信息，而双目立体相机则需要复杂的算法进行支撑，处理速度比较慢。Kinect2.0 能准确的进行三维探测，而双目立体相机的计算精度受到环境物体特征的影响，如：物体表面的灰度等，同时 Kinect2.0 的计算精度不随距离的改变而变化。综上所述，本文从实用性和有效性出发，选择 Kinect2.0 作为图像数据采集的传感器，为后续实验提供精度准确的图像数据，减少相对误差。

（2）图像数据采集示例。如下图 2-6 所示，为使用 Kinect2.0 传感器对实验室的室内环境进行图像数据采集，得到的彩色图像以及相对应的深度图像示例。

（a）彩色图像　　（b）深度图像

图 2-6 Kinect2.0 图像数据采集示例

2.4 视觉 SLAM 概述

首先，在视觉 SLAM 的整体框架中，传感器获取到的图像数据信息质量好坏，直接影响到前端视觉里程计的定位估计工作。假如由于传感器自身的原因，导致获取到的图像数据信息噪声超过了一定范围，那么视觉里程计构建的局部地图必定会与现实场景偏差增大，最终可能导致定位估计失败[8]。其次，传感器获取图像数据的实时性、精确性也会影响到视觉里程计的相关工作，对于实际的要求来说，RGB-D 传感器的正常工作除了有自身的某些特性之外，应该不受特定环境或者物体特征的限制[16]。最后，视觉里程计得到的初始值直接决定了整个视觉 SLAM 体系结果的可用性，在实际的应用过程中，可以利用传感器的优势提高视觉里程计的工作效率，使得整个视觉 SLAM 框架平稳的运转。

2.4.1 经典视觉 SLAM 框架

通过视觉 SLAM 的定义，我们知道它并不只是某种特定的算法，也不是说只要我们将图像数据输入就会有相应的定位和地图信息输出，它理论上应该是一套完整的算法框架，用来解决一系列的问题，经过研究者们过去十几年的努力、不断的工作，视觉 SLAM 已经形成一套经典的框架结构。如下图 2-7 所示，为经典视觉 SLAM 框架的流程图。

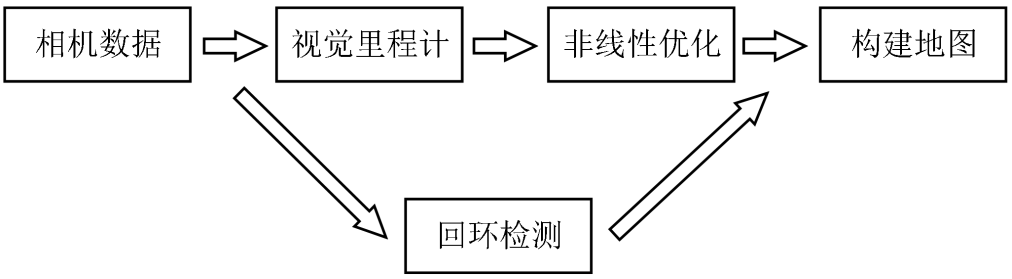


图 2-7 经典视觉 SLAM 框架的流程图

经典视觉 SLAM 框架流程包括：

（1）图像数据获取：在视觉 SLAM 中一般指读取相机获取的多张图片或者图片流的信息以及预处理过程。

（2）视觉里程计（Visual Odometry）：又称为视觉前端，负责为后端非线性优化提供相机运动的估计值以及局部地图模型等优化后的初始值。

（3）回环检测（Loop Closing）：又称为闭环检测，负责将机器人的历史位置信息与实时位置信息进行检测判断，如果位置信息相同，则提交给后端非线性优化进行处理。

（4）非线性优化（Optimization）：又称为噪声优化后端，负责将前端视觉里程计提供的初始值以及回环检测的信息进一步优化，最终得到全局统一的运动轨迹和初始地图。

（5）构建地图（Mapping）：负责构建任务具体要求的完整地图。

2.4.2 视觉 SLAM 问题的数学表述

视觉 SLAM 中通过相机的运动数据得到移动机器人的位姿变化，相机的图像观测数据包含了环境地图的信息，那么视觉 SLAM 问题可以用两个通用的基本数学方程进行描述，如下式 2-1 所示：

$$\begin{cases} x_k = f(x_{k-1}, u_k, r_k)，\text{运动方程} \\ m_{k,n} = h(y_n, x_k, q_{k,n})，\text{观测方程} \end{cases} \tag{2-1}$$

公式中离散时刻 t 取值 $1 \sim k$，通用抽象函数 f，h，运动方程中，x 表示移动机器人的位置，各时刻的位置取 $x_1 \sim x_k$，$x_1 \sim x_k$ 构成机器人的运动轨迹，u_k 为输入，即传感器的读数，r_k 表示噪声。观测方程中，路标数量取值 $1 \sim n$，$m_{k,n}$ 表示观测的具体数据，y_n 表示在离散时刻 k 及位置 x_k 处观察到的某一个路标点，$q_{k,n}$ 表示此观测数据的噪声。通过这两个方程可知，由于要解决定位估计（求解 x）以及地图构建（估计 y）两个问题，那么视觉 SLAM 问题可以描述为一个状态估计问题：如何在图像数据存在噪声干扰的前提下，估计求解内部的状态变量。

2.5 本章小结

本章首先对深度测量技术的相关知识进行了详细介绍，阐述了深度测量技术的分类以及深度测量技术各方案之间的优缺点比较。然后对 Kinect 系列传感器的主体结构进行了区别介绍，从多方面对 Kinect1.0、Kinect2.0 的传感器配置进行了比较，说明了 Kinect2.0 多方面的优势。接着从 Kinect2.0 采用 TOF（飞行时间法）技术作为深度信息获取方式的相关优势出发，将 Kinect2.0 与传统的双目立体视觉相机作多方面比较，为了兼顾传感器的实用性和有效性，选择将 Kinect2.0 作为本文后续的改进算法以及实验阶段的图像数据采集传感器，并将 Kinect2.0 采集到的实验室室内环境的颜色图像和深度图像作为示例。最后，从视觉 SLAM 的定义出发，介绍了经典的视觉 SLAM 框架，并将视觉 SLAM 问题用抽象、统一的数学公式表达出来。

第三章 基于 RGB-D 的视觉里程计算法

3.1 引言

人类经过大量的视觉训练能直观的了解客观世界的情况，但是在计算机中，图像只是一个个数值矩阵而已，计算机理解客观世界的难度就像人类理解这些数值矩阵一样大。所以计算机如何通过图像信息确定移动机器人或相机的运动是计算机视觉领域一直以来的难题。

在视觉 SLAM 中，目标场景中的空间点投影到相机的成像平面上形成了一个个像素点，获取相机与空间几何点之间的关系才能定量地、相对准确地估计相机的运动。在 RGB-D 传感器直接能获得物体在场景中深度信息的基础上，视觉里程计要有效解决“相机运动估计”问题以及“局部地图构建”问题，选择合适的定位估计算法是视觉里程计为视觉 SLAM 后端非线性优化提供高质量初始值的前提。但是各类视觉里程计估计算法有其各自独特的优势以及相应的局限性，在不同的环境下，实现的效果各有差异，所以结合、扩展、优化这些算法是关键所在。

3.2 视觉里程计详细概述

3.2.1 视觉里程计的基本定义

（1）里程计的含义。里程计一般意义上包括硬件和算法两部分，它的目标是测量一个运动物体的轨迹信息，常见的实现手段有很多，例如在汽车的轮胎上安装带有计数功能的码盘，记录轮胎的转动数据，从而求得汽车的运动估计。也可以测量并记录汽车的速度、加速度等数据，利用时间积分来求得汽车相应的位移，从而完成汽车的运动估计。

（2）里程计的特性。大多数情况下，它只关心目标在局部时间上的运动，比如某相邻两个时刻。

（3）视觉里程计的定义。它是指根据相机传感器拍摄到的图像作为依据来估计相机的运动，在视觉 SLAM 中，它还负责构建局部地图的工作。视觉里程计具有里程计的特性，它只关心相邻两帧图像之间的关系，以此估计相机运动和构建场景的空间结构[33][48][50]。

（4）视觉里程计存在的问题。视觉里程计解决定位问题的方法，是把相邻时刻图像间的相机运动联系起来，从而估计移动机器人的运动轨迹。视觉里程计解决构图问题的方法，是将各个时刻的相机位置记录下来，以此作为依据，求出每一个

像素与空间点之间的相对应关系来构建地图。由于里程计的工作特性，视觉里程计估计的只是相邻时间相机的运动，而每一次估计都带有一定的误差，这些误差会随时间传递，逐渐累加，最终导致估计的运动轨迹出现偏移，这就是视觉里程计存在的累积漂移问题（Accumulating Drift）[39]。误差是不可避免的，但是高质量、高效率的视觉里程计算法可以减少相对误差，使得累积漂移问题的影响最大程度的减小。同时，视觉 SLAM 后端优化技术和回环检测技术就是为了解决这个问题而存在的，后端优化校正运动轨迹，回环检测则判断先前位置与实时位置的相似性。

3.2.2 视觉里程计的相关算法

（1）在视觉 SLAM 中，图像特征指的是路标（Landmark），简单来说，路标就是图像中具有代表性的点，当相机的视角进行少量偏移的时候，路标不会发生变化，路标的定位问题是相机位姿估计问题的基础。在计算机中，数字图像是以灰度值矩阵的形式存放的，但是光照条件、物体形变等多种因素会在多种程度上影响灰度值的稳定，当相机的视角偏移或者场景某些条件发生改变时，仅仅用灰度值去判断图像中哪些地方是同一个点是不太可靠的，所以对图像提取特征点是必要的。特征点包括关键点（Key-Point）和描述子（Descriptor）两部分内容。关键点是指图像中特征点的位置，也可能包含方向、大小等其它信息，描述子的主要内容是该关键点周围像素的信息，通常要判断两个特征点是否为同样的特征点，只需要知道两个特征点的描述子在向量空间上的位置是否相近。直接法没有特征匹配的过程，它在基于灰度不变假设的前提下，只考虑像素的亮度信息来估计相机的运动。灰度不变假设是指，在各个图像中，同一个空间点的像素灰度值是不发生变化的。

（2）在经典视觉 SLAM 框架中，后端优化的初始值来源于视觉前端视觉里程计提供的相机运动估计值以及局部地图模型。根据特征提取的必要性，特征点法以及基于像素信息的直接法都能用于视觉里程计的实现。在基于 RGB-D 传感器的视觉里程计算法中，特征点提取算法常用的有：SIFT（尺度不变特征变换，Scale-Invariant Feature Transform）算法[43]、SURF（加速健壮特征，Speed Up Robust Feature）算法[30]、ORB（Oriented FAST and Rotated BRIEF）算法[53]等等。相对于直接法来说，特征点法的解决方案比较成熟，其一直以来被视为视觉里程计的主要实现方法。近几年，得益于在一些开源项目（如：SVO[37]、LSD-SLAM[35]等）中的使用，基于像素信息的直接法逐渐在视觉里程计算法中扮演重要角色。

3.3 基于 RGB-D 的视觉里程计估计算法

3.3.1 特征点法

（1） 算法流程。

如下图 3-1 所示，为基于特征点法的 RGB-D 视觉里程计估计算法的流程图。

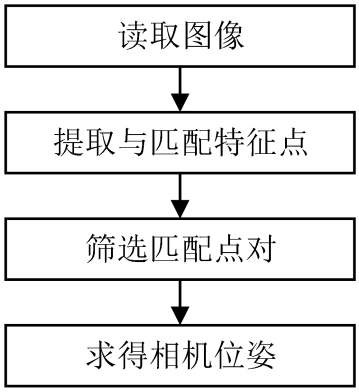


图 3-1 基于特征点法的 RGB-D 视觉里程计估计算法的流程图

（2） 图像特征的提取与分析。

1.基于加速分割测试（FAST，Features From Accelerated Segment Test）算法[52]。

FAST 关键点提取原理：对于图像每一个像素执行以下相同的操作，选取图像中亮度为 I_p 的某一个像素点 p 为中心，然后从 p 周围半径为 3 的圆上选取 16 个像素点，设定一个阈值 T，如果这 16 个像素点中有 N 个的亮度小于 $I_p - T$，或者大于 $I_p + T$，那么 p 就是一个特征点，根据 N 的取值，一般可分为 FAST-9，FAST-11，FAST-12，其中 FAST-12 最常用。如下图 3-2 所示，为 FAST 关键点提取原理。

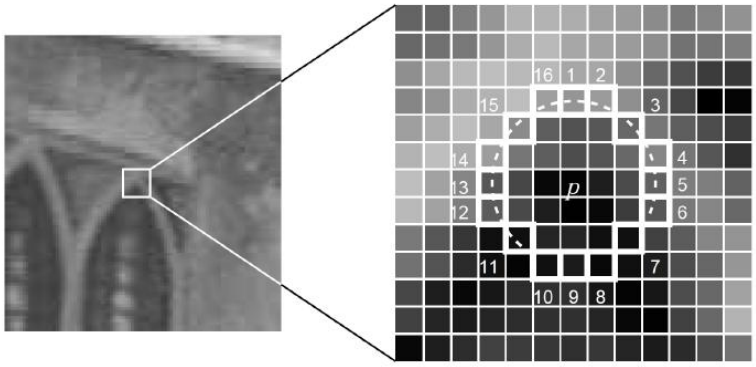


图 3-2 FAST 关键点提取原理

2.尺度不变特征变化（Scale-Invariant Feature Transform，SIFT）算法。

SIFT 算法它将层叠滤波的方法运用到特征点的提取中，SIFT 算法解决了相机在运动的过程中，场景中物体的位置以及尺寸会随着相机视角的改变而变化的问题，同时，SIFT 利用一个连续的尺度变换空间以及搜索稳定特征的方法，判断图像中那些随旋转不发生改变的位置。最终，相关的学者研究发现，在多种前提假设相对合理的条件下，高斯函数可能是唯一满足这些条件的尺度变换空间。

假设可变尺度高斯方程为：$G(x,y,\delta)$，图像方程为：$I(x,y)$，图像的尺度变换空间方程为：$S(x,y,\delta)$，那么 $S(x,y,\delta)$ 可以用可变尺度高斯方程和图像方程的卷积表示，如下式所示，其中 $*$ 为卷积的运算符号。

$$S(x,y,\delta)=G(x,y,\delta)*I(x,y) \tag{3-1}$$

可变尺度高斯方程：$G(x,y,\delta)=\frac{1}{2\pi\delta^2}e^{-(x^2+y^2)/2\delta^2}$ （3-2）

SIFT 算法的提出者 Lowe 将它分成了以下四个步骤：

a）检测特征点的具体位置。首先利用高斯差分函数中相邻尺度的差异计算尺度变换空间，然后将尺度变换空间中相应的极值检测图像中稳定关键点的位置。设需要构建的高斯差分函数尺度空间为：$H(x,y,\delta)$，通过计算相邻尺度的差异因子 k 来构建，那么高斯差分函数尺度空间用下式表示为：

$$H(x,y,\delta)=(G(x,y,k\delta)-G(x,y,\delta)*I(x,y))=S(x,y,k\delta)-S(x,y,\delta) \tag{3-3}$$

$\delta^2\nabla^2G$ 为高斯差分函数尺度空间 $H(x,y,\delta)$ 尺度归一化后的高斯拉普拉斯近似。

$H(x,y,\delta)$ 与 $\delta^2\nabla^2G$ 的关系用下式表示：

$$\frac{\partial G}{\partial\delta}=\delta\nabla^2G \tag{3-4}$$

由此可以知道，$\partial G/\partial\delta$ 的有限差分逼近可以得到 ∇^2G，那么 $\delta\nabla^2G$ 可以近似表示为下式：

$$\delta\nabla^2G=\frac{\partial G}{\partial\delta}\approx\frac{G(x,y,k\delta)-G(x,y,\delta)}{k\delta-\delta} \tag{3-5}$$

最后得到下式：

$$G(x,y,k\delta)-G(x,y,\delta)\approx(k-1)\delta^2\nabla^2G \tag{3-6}$$

其中 $(k-1)$ 属于常数，对尺度变换空间极值的检测没有影响，k 一般取 $\sqrt{2}$ 。

b）确定关键点的位置。比较图像相邻像素的信息可以提取出候选的关键点，

候选关键点的位置和尺度可以根据附近相邻的数据进行二次函数拟合求得，在此基础上，排除掉某些低对比度的点以及边缘不稳定的点，确定了中央样本点附近的尺度和位置，最后通过拟合样本点和三元二次函数定位最大的插入位置。具体过程如下所示：

将 $H(x,y,\delta)$ 做泰勒展开：

$$H(x)=H+\frac{\partial H^T}{\partial x}x+\frac{1}{2}x^T\frac{\partial^2 H}{\partial x^2}x \tag{3-7}$$

令 $H(x)$ 导数的 $H'(x)$ 为 0 得到极值 $\hat{x}$ 的位置，其中 $x=(x,y,\delta)^T$，如下式所示：

$$\hat{x}=-\frac{\partial H^T}{\partial x}\left(\frac{\partial^2 H}{\partial x^2}\right)^{-1} \tag{3-8}$$

将公式 3-8 带入公式 3-7 中得下式：

$$H\left(\hat{x}\right)=H+\frac{1}{2}\frac{\partial H^T}{\partial x}\hat{x} \tag{3-9}$$

候选极值点对噪声敏感的标准是 $\left|H\left(\hat{x}\right)\right|$<=0.03，这些候选点因为噪声变得不稳定，予以剔除。

c）关键点方向和幅值的确定。正常情况下，每个筛选后的关键点有各自的方向，即关键点描述子的方向，图像的旋转不变性正是以此为基础的。以下介绍一种效果较好的算法，首先图像内容包括：尺度空间表示为 $S(x,y)$，梯度幅值表示为 $\mathrm{m}(x,y)$，梯度方向表示为 $\theta(x,y)$，然后将 $3\times1.5\times\delta$ 作为邻域间窗口的大小，最后将窗口内满足要求的候选关键点全部进行采集，并计算这些候选关键点的梯度幅值和梯度方向，结果分别为公式 3-10，3-11 所示;

梯度幅值：$\mathrm{m}(x,y)=\sqrt{(S(x+1,y)-S(x-1,y))^2+(S(x,y+1)-S(x,y-1))^2}$ （3-10）

梯度方向：$\theta(x,y)=\tan^{-1}\left[\frac{S(x,y+1)-S(x,y-1)}{S(x+1,y)-S(x-1,y)}\right]$ （3-11）

d）计算描述子。描述子的计算就是特征点建立特征向量的过程，每个关键点的特征向量为 128 维，具体内容包括：每个关键点分为 4×4 一共 16 个分点去描述，每个分点含有 8 个方向向量的相关信息，这样就产生了 128 个数据。

3.加速健壮特征算法（Speed Up Robust Feature，SURF）算法。

SURF 算法的执行速度大概在 SIFT 算法的 3 倍左右，它是将 SIFT 进行优化后的算法，同时也是一种稳健的局部特征点提取和分析描述的算法，它具体包括以下几个步骤：

a）检测特征点。在 SURF 算法中，利用的是 Hessian 近似矩阵的响应来衡量某一点是否是特征点，对于图像 I 中某一点 $x=(x,y)$，如下列公式 3-12 中，$H(x,\delta)$表示的是点 $x=(x,y)$在尺度系数 δ 下的 Hessian 矩阵：

$$H(x,\delta)=\begin{bmatrix} L_{xx}(x,\delta) & L_{xy}(x,\delta) \\ L_{xy}(x,\delta) & L_{yy}(x,\delta) \end{bmatrix} \tag{3-12}$$

公式 3-12 中，Hessian 矩阵中包含的四项分别代表高斯二阶微分 L_{xx}，L_{xy}，L_{yy} 和图像 I 对应点之间进行卷积而得到的结果，例如：高斯二阶偏导数 $\frac{\partial^2}{\partial x^2}$ 在 x 处与图像 I 的卷积为 $L_{xx}(x,\delta)$。高斯滤波器对高斯二阶微分采取离散化、截断化，能够在不同尺度的情况下使用，但是当图像进行降采样的时候，不会出现新的特征，所以考虑使用盒滤波器代替高斯滤波器，将对应点与高斯二阶微分进行卷积的方法近似成与盒滤波器进行卷积。如下图 3-3 所示，为盒滤波器示例图。

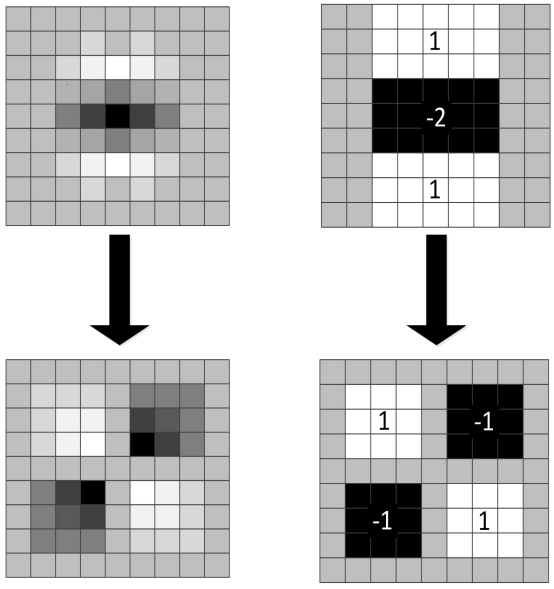


图 3-3 盒滤波器示例图

采用C_{xx}，C_{xy}，C_{yy}表示近似滤波器计算得到的卷积，近似计算 Hessian 矩阵对应的行列式$\det(H_{ap})$，其中 j 为调节参数，与尺度因子δ有关，下式为具体计算公式：

$$\det(H_{app}) = C_{xx}C_{yy} - (jC_{xy})^2 \qquad (3\text{-}13)$$

b）构建尺度空间。SURF 算法中不同图像组中，图像尺寸是一致的，但盒滤波器的模板尺寸逐渐增大，所以将相同尺寸的滤波器运用到同一组中的不同层上。

c）定位特征点。SURF 算法中，初步筛选定位关键点的方法是：将二维图像空间以及尺度变换空间邻域内的 26 个点和经过 Hessian 矩阵处理后的图像单个像素点进行比较。最终稳定的特征点剔除掉了定位错误的关键点。如下图 3-4 所示，为 SURF 算法定位特征点的示意图。

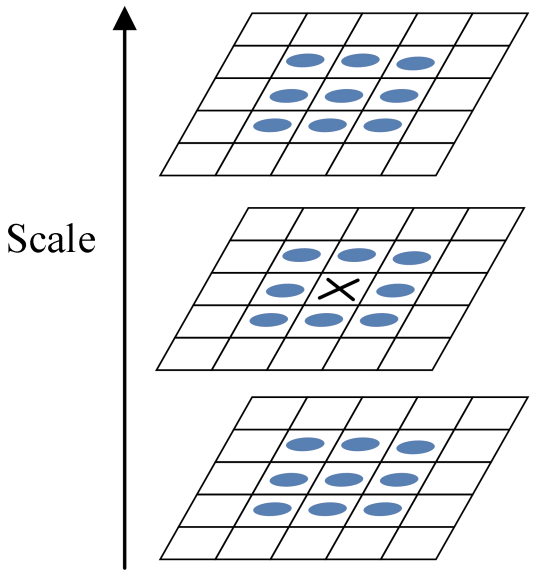


图 3-4 SURF 算法定位特征点的示意图

d）确定特征点主方向的分配。首先为每一个特征点确定一个方向，方向是能够复制的，这样保证了旋转的不变性。然后，SURF 算法统计特征点对应圆形邻域内的 Haar 小波特征响应，包括统计圆形邻域中 60 度扇形区域内所有点 x，y 坐标方向 Haar 小波特征之和，x 和 y 分别表示水平、垂直方向，接着扇形区域以 0.2 弧度大小的间隔进行旋转后，继续统计扇形区域内所有点 x，y 坐标方向的 Haar 小波特征之和，最终将此过程中累加值最大的扇形区域的方向作为特征点的主方向，该过程的示意如下图 3-5 如下：

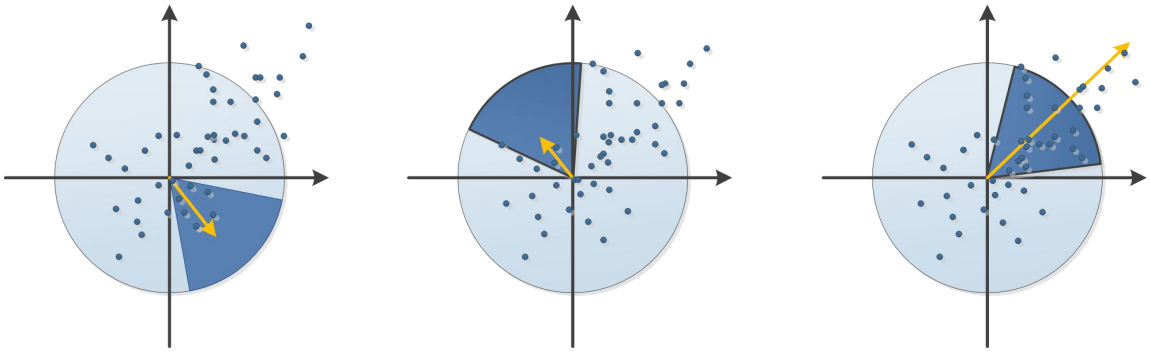


图 3-5 确定特征点主方向过程的示意图

e）生成特征点的描述子。在特征点的周围取一个矩形图像块，大小为4×4，方向是特征点的主方向，在选取的每一个子区域图像块中，统计 25 个像素水平、垂直坐标方向上的 Haar 小波特征，该 Haar 小波特征有：水平方向值之和、水平方向绝对值之和、垂直方向之和、垂直方向绝对值之和四个方向。把四个方向作为每个子区域块的特征向量，总共有 $4\times4\times4=64$ 维特征向量，这 64 维特征向量就是 SURF 算法特征的描述子。该过程如下图 3-6 所示：

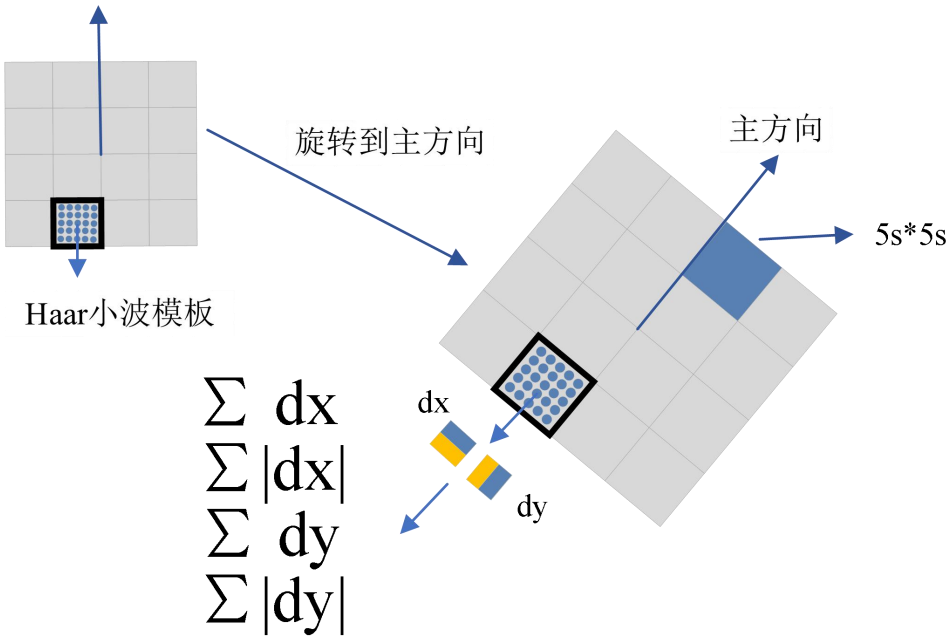


图 3-6 特征点描述子生成过程的示意图

4.ORB（Oriented FAST and Rotated BRIEF）算法。

Rublee 等人在 2011 年提出了 ORB 算法，ORB 特征提取方法包括“Oriented

FAST”关键点和二进制描述子 rBRIEF（Rotation Aware Binary Robust Independent Elementary Feature）。

a）“Oriented FAST”关键点。FAST 关键点速度方面的优势在于它只涉及像素亮度差异的比较，但提取出的特征点数量很大，具有不确定性，又没有方向信息，也不满足尺度变化，ORB 特征提取算法对 FAST 关键点提取进行了改进和优化[7]。

ORB 特征利用 FAST 特征点检测方法检测特征点，采用 Harris 角点的度量方法，从检测出的所有特征点中挑选出 Harris 角点响应值最大的 N 个特征点。同时为了解决 FAST 关键点没有尺度不变性这个缺点，ORB 算法通过构建图像高斯金字塔的方法，在每一层金字塔图像上检测角点，实现了尺度不变性。最后，为了解决 FAST 关键点没有方向这个缺点，ORB 算法利用灰度质心法（Intensity Centroid）添加、计算了特征点的主方向，图像块权重的中心称之为质心，灰度质心法具体计算如下：首先定义某一图像块 A 的矩：

$$\mathrm{m}_{pq} = \sum_{x,y \in A} x^p y^q I(x,y) \quad p,q = \{0,1\} \tag{3-14}$$

其中 $I(x,y)$ 为点 (x,y) 处的灰度值，然后利用矩找到图像块 A 的质心：

$$Z = (\frac{m10}{m00}, \frac{m01}{m00}) \tag{3-15}$$

特征点的方向就是图像块 A 的几何中心 O 与质心 Z 连接的方向向量：

$$\overrightarrow{OZ} = \arctan(m_{01}/m_{10}) = \arctan\left(\frac{\sum_{x,y} yI(x,y)}{\sum_{x,y} xI(x,y)}\right) \tag{3-16}$$

ORB 特征算法为 FAST 特征算法增加了特征点的主方向以及尺度不变性，从而实现了特征点的旋转不变性，这种旋转不变性需要确保 $x,y \in [-r,r]$，r 为相邻圆形区域的半径。

b）rBRIEF 描述子。BRIEF 描述子是一种识别率很好的二进制特征描述子，效率很高，适用于实时的图像匹配，它使用的是随机选点的比较，用 1 和 0 分别描述了关键点附近某两个像素的大小关系，所以 BRIEF 描述子 N 维描述向量的基本元素是 1 和 0。

改进旋转不变性：BRIEF 描述子本身是没有旋转不变性的，Steered BRIEF 给 BRIEF 添加了旋转不变性，以下是具体过程：任意一个特征点周围的 $2n$ 个点（n 个点对）生成一个长度为 n 的二进制码串 BRIEF 描述子，这 $2n$ 个点 (x_i, y_i)，$\mathrm{i} = 1,2,...,2n-1,2n$ 组成一个矩阵 S 如下式所示：

$$S=\begin{pmatrix} x_1 & x_2 & \cdots & x_{2n} \\ y_1 & y_2 & \cdots & y_{2n} \end{pmatrix} \tag{3-17}$$

S用θ表示邻域方向和R_θ表示对应的旋转矩阵进行校正得到S_θ，如下式所示：

$$S_\theta = R_\theta S \tag{3-18}$$

其中：

$$R_\theta = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \tag{3-19}$$

θ为特征点求得的主方向。

改进 Steered BRIEF 描述子的相关性：描述子的相关性也就是可区分性，这个性质非常影响特征匹配的好坏。BRIEF 是二进制串的描述子，只含有 0 和 1，二进制串的均值在 0.5 左右时，方差比较大，这里的 0.5 表示 0 和 1 的数量相差不多。但是 Steered BRIEF 相对于改进的 BRIEF 来说，描述子二进制串的均值在 0.5 以上，每个描述符的方差比较小，可区分性不强，特征点的描述子之间具有很大的相关性，rBRIEF 能解决 Steered BRIEF 这个问题，以下为具体过程：

ORB 论文中采用统计学习的方法重新选择点对集合，首先选取$300k$个特征点作为测试集合，考虑测试集中每个点周围31×31的邻域，为了使得特征值更加不受噪声的影响，在对图像进行高斯平滑处理后，将邻域中某一点5×5邻域灰度的平均值替代某个点对的值来比较点对的大小。那么31×31的邻域里共有$(31-5+1)^2=729$个子窗口，有$N=265256$取点对的方法，从中选择 256 中取法保证它们的相关性最小，选取步骤为：首先在N种方法中比较点对大小，得到一个$300k\times N$的二进制矩阵F，F中的每一列的二进制序列是$300k$个点按照某种取法得到的，然后以均值 0.5 的标准距离，对F的列向量取平均值并且重新排序得到矩阵T，接着将T矩阵的第一列向量放到R中，取得T矩阵的下一列向量和R中所有的列向量计算相关性得到相关系数，假如相关系数小于预先设定的值，那么将T这一列加入R中，重复该对比操作，直到R中的向量数量为 256 即可，这就是“贪婪算法”，那么以上就是 rBRIEF。

c）如下图 3-7 所示，为完整的 ORB 特征算法流程图：

图 3-7 完整的 ORB 特征算法流程图

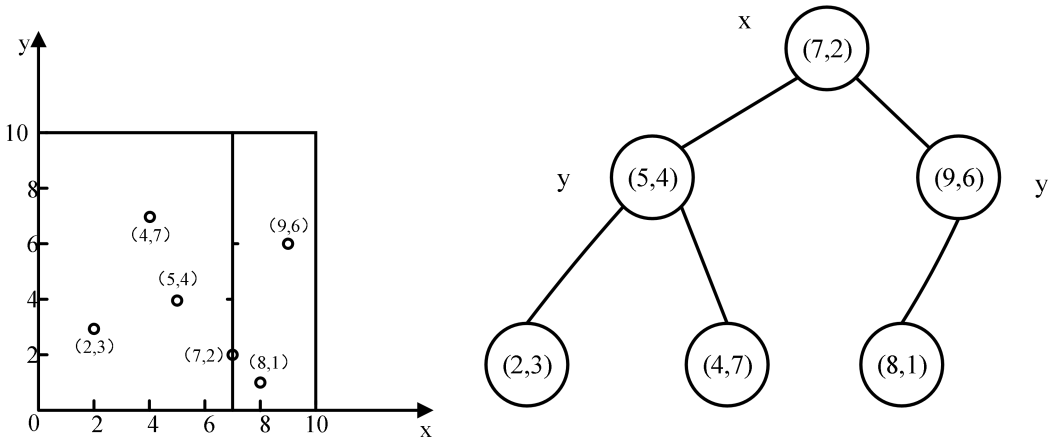
（3） 图像特征匹配

暴力匹配（Brute-Force Matcher）是最简单的特征匹配方法，它将每一个特征点和下一时刻的图像提取到的所有特征点进行比较，测量描述的距离，然后进行排序，最后取最近的一个作为匹配点。很显然，当图像特征点的数量庞大时，暴力匹配的运算时间和运算量会非常大。此时，快速近似最近邻（Fast Library For Approximate Nearest Neighbors，FLANN）算法[46]比较适合。

2009 年，Lowe 等人提出了基于 K 均值树和 KD-Tree（K-Demension Tree）[31][42]搜索操作的 FLANN 算法，在n维实数的向量空间R_n中，FLANN 算法利用欧式距离找到与实例特征点相邻的点，假设现有两个特征点分别为x和y，那么x，y之间的欧式距离$d(x,y)$为公式 3-20 所示：

$$d(x,,y)=\sqrt{(x_1-y_1)^2+(x_2-y_2)^2+.....+(x_n-y_n)^2}=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2} \tag{3-20}$$

假设 KD-Tree 在向量空间R_n中存在数据点，这个数据点将 KD-Tree 划分为几个特定的部分，通过对 KD-Tree 进行递归、且由上至下的搜索，得到数据点最近的欧式距离。以一个简单直观的实例说明，现有 6 个二维数据点(2,3)，(5,4)，(9,6)，(4,7)，(8,1)，(7,2)，如下图 3-8 所示，为 KD-Tree 的结构示例图。



（a）将空间分为左右空间　　（b）实例生成的 K-D 树

图 3-8 KD-Tree 的结构示例图

对于两帧图像之间的特征匹配来说，图像中提取出的每一个特征点包括关键点和描述子两部分，关键点的主要内容包括特征点的位置、方向、大小等信息，特征点周围的像素信息是特征点描述子的主要内容，那么要判断两个特征点之间是否相似，或者两个特征点之间的相似程度如何，可以通过描述子的距离来度量。典型的欧式距离适用于浮点型的描述子，本文选用 ORB 算法作为特征提取的方法，ORB 特征点中的 BRIEF 描述子是用二进制串的形式表达的，所以用欧式距离度量很明显不适用，在这种情况下往往使用汉明距离（Hamming Distance）来度量描述子的距离。汉明距离表示的是两个字符串之间不同位数的个数，例如：现在有两个字符串：100010，111001，从字符串第一个字符往后依次比较，不相同的位数有 4 个，那么这两个字符串的汉明距离为 4。

3.3.2 直接法

（1）算法流程。如下图 3-9 所示，为基于直接法的 RGB-D 视觉里程计估计算法的流程图。

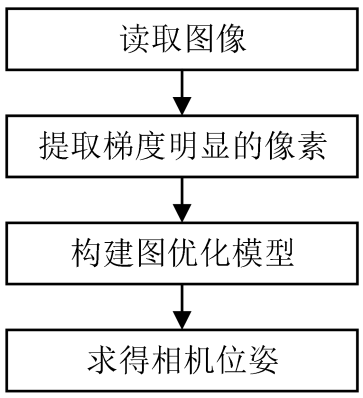


图 3-9 基于直接法的 RGB-D 视觉里程计估计算法的流程图

（2）提取梯度明显的像素。基于灰度不变假设的条件下，直接法构造了一个“最小化不同图像中同一个像素的光度误差”的优化问题来求解相机位姿。本文采用的是半稠密的直接法，即在像素的提取过程中，不考虑像素梯度变化小的像素点，只提取像素梯度明显的像素点。直接法可以在明暗条件发生变化的环境中，即存在对应像素梯度的场景中，根据像素的亮度信息来估计相机的运动。在基于直接法估计相机运动的过程中，直接法可以完全舍弃关键点和描述子的计算过程，这样既节省了特征点的计算时间，也不会出现特征缺失的情况，这就是直接法比特征点法在图像视觉特征缺乏的环境中鲁棒性更加好的原因。

3.4 特征点法与直接法的优缺点讨论

（1）在基于特征点法估计相机运动的过程中，虽然特征点法目前在视觉里程计算法中占据主流地位，但是它还存在以下缺点：

a）计算时间长。特征点法中关键点的提取过程和描述子的计算过程比较消耗时间，假如以 30 毫秒/帧的速度运行整个视觉 SLAM，那么这个过程中超过一半的时间都将会用在特征点的计算上。

b）图像某些信息会丢失。由于特征点法排除了除特征点之外的所有图像信息，所以特征点法不能构造半稠密或者稠密的地图。而在同一张图像中，像素的数量要远远大于特征点的数量，这样某些可能有用的图像信息就会被丢弃。

c）难以面对视觉特征缺失的场景。相机运动过程中有时会在缺少明显纹理特征的场景中工作，例如纯色的墙面环境，稀疏线条的地板环境等，这些场景中特征点的数量少之又少，特征点法可能会缺失足够的匹配点，导致算法失效。

（2）直接法也存在以下缺点：

a）非凸性质。图像的像素梯度直接决定了直接法的梯度走向，而直接法是依靠梯度搜索来降低目标函数像素点的灰度值，从而求得相机位姿的。图像一般都是非凸函数，不能保证沿着梯度走时，灰度误差会不断的下降。所以直接法要求相机的运动足够小，这样才能保证图像梯度不会有很强的非凸性。

b）单一像素的区分度太小。图像中某单个像素点存在大量与它相似的其它点，所以只能通过图像块或者计算像素之间复杂的相关性来解决，也就是多数代替整体的方式。

3.5 本章小结

本章为后文提供一些基础的理论知识，为后文的算法设计和系统设计打下了理

论基础。首先详细的概述了视觉里程计的基本定义、特性以及存在的一些问题，然后介绍了视觉里程计相关的一些算法。接着从原理、流程等方面，阐述了特征点法和直接法两种主流的视觉里程计算法，特别详细介绍了 FAST、SIFT、SURF、ORB 特征算法和半稠密直接法的原理。最后深入讨论了特征点法和直接法的优缺点。

第四章 基于视觉 SLAM 的虚拟现实空间定位系统设计与实验

4.1 引言

视觉 SLAM 的传感器是相机，本文选用 Kinect2.0 深度传感器采集现实空间场景中的颜色、深度信息。本文提出的基于图像特征的视觉里程计自适应算法能够很好的解决特征点法和直接法在不同环境特征中单一使用造成的特征丢失、运动估计失败等问题，例如：当场景中的图像视觉特征发生大幅度减退时，特征点法由于提取不到足够的特征点而造成相机位姿估计偏差较大，甚至失败，此时选用基于像素信息的直接法能稳定的估计相机位姿，所以该算法能有效解决现实空间中相机的位姿问题，即定位问题。

本章设计的基于视觉 SLAM 的虚拟现实空间定位系统是前面几个章节知识的延伸以及综合运用，虚拟现实空间的定位问题总体来说也就是相机视角在虚拟空间的变化以及相机在虚拟空间中的移动。而相机视角在虚拟空间的变化和移动实际上是通过现实空间中相机的移动来实现的，现实空间中的相机定位直接反应在虚拟空间中，所以现实空间中相机的位姿估计问题求解结果是虚拟现实空间定位系统的关键所在。

基于视觉 SLAM 的虚拟现实空间定位系统要实现的功能包括：

（1）利用 Kinect2.0 深度传感器实现现实空间中图像数据的实时采集。

（2）根据采集到的图像数据，利用视觉里程计定位估计算法估计相机的运动，得到相机位姿的估计结果。

（3）根据相机位姿的估计结果，确定相机在现实空间中的位置。

（4）将相机在现实空间中的位置传输到本地服务处理端。

（5）本地服务处理端将现实空间中相机的位置传输到虚拟空间中，确定虚拟空间相机的朝向和位置，实现现实空间与虚拟空间相机的同步定位。

4.2 系统的设计和实现流程

本文设计的基于视觉 SLAM 的虚拟现实空间定位系统分为三个模块，即系统前端、系统中端、系统后端。系统前端的主要工作是现实空间中图像数据的采集，系统中端根据系统前端的图像数据进行相机位姿的优化，系统后端实现现实空间和虚拟空间相机的同步定位。

（1）如下图 4-1 所示，为系统的整体框架设计图。

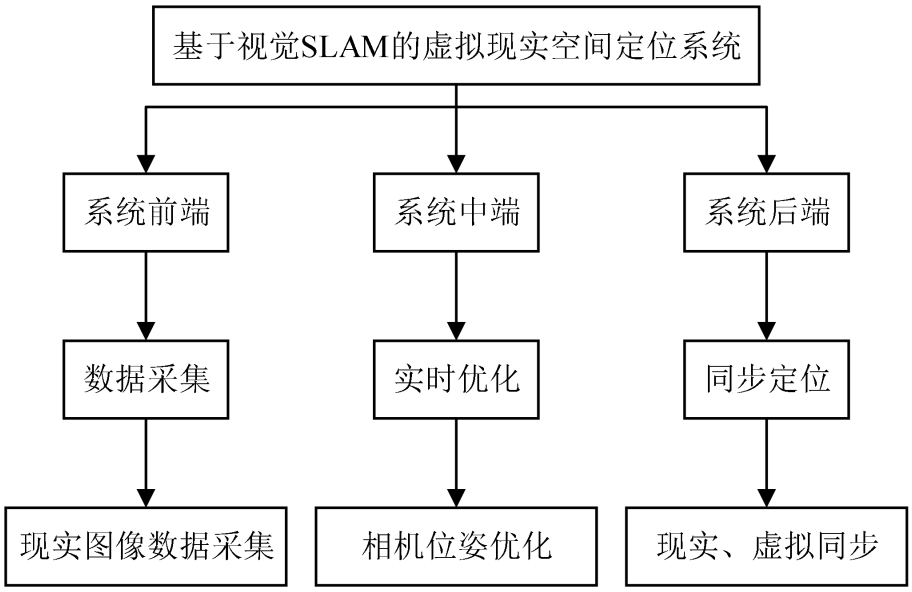


图 4-1 系统的整体框架设计图

4.2.1 前端数据采集

在现实空间中，目标场景中摆放的物体有不同的明暗程度，物体的外观轮廓各有差异，甚至纹理、线条都是各不相同的。Kinect2.0 能够根据采集到的目标场景的深度信息，获得物体在场景中的摆放位置，也就是目标物体的三维坐标信息。场景中物体的三维坐标信息到像素坐标信息的过程要经过一系列的坐标系转换，下面从针孔相机模型出发，介绍世界坐标系，相机坐标系，像素坐标系之间的转换关系。

（1）针孔相机模型。如下图 4-2 所示，为针孔相机模型图。

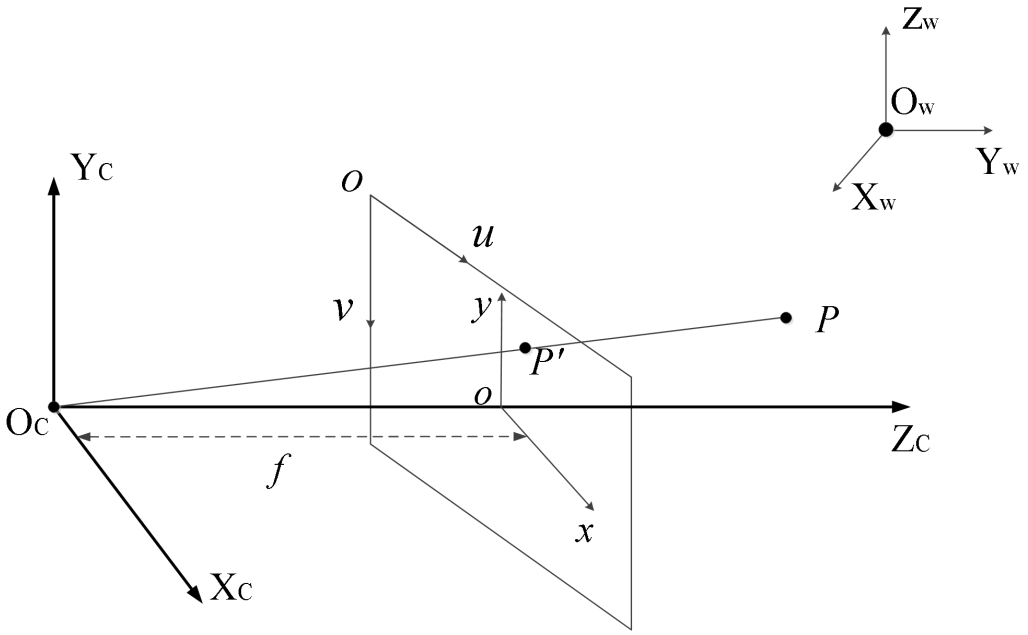


图 4-2 针孔相机模型图

相机的成像过程可以由简单的针孔相机模型来解释，设相机坐标系为$O_c-X_cY_cZ_c$，O_c为相机的光心，即针孔，其中Z_c轴的方向指向相机的前方。设相机坐标系有一空间点$P(X_cY_cZ_c)$，$P(X_cY_cZ_c)$过光心O_c后投影在物理成像平面$o-xy$上，成像点为$p'(x,y)$，o是光轴与物理成像平面的交点，O_c-o之间的长度为焦距f，根据相似三角形的关系可知：

$$\frac{Z_c}{f}=\frac{X_c}{x}=\frac{Y_c}{y} \tag{4-1}$$

整理可得：

$$\begin{cases} x=f\dfrac{X_c}{Z_c} \\ y=f\dfrac{Y_c}{Z_c} \end{cases} \tag{4-2}$$

公式 4-2 描述了空间点P和它所成像之间的空间关系。

（2）相机的内参数和外参数

内参数：像素坐标系（也称图像坐标系）和图像物理坐标系之间的转换关系，它是与相机自身特性相关的参数，例如相机的焦距，像素大小等。

外参数：世界坐标系和相机坐标系之间的转换关系。它是在世界坐标系中的参数，例如相机的位置，旋转方向等。

（3）相机坐标系到图像物理坐标系之间的转换

由针孔相机模型可知，相机坐标系为O_c-$X_cY_cZ_c$，O_c为光心，Z_c轴与光轴平行，X_c轴和Y_c轴分别平行于x，y轴方向。光轴与物理成像平面的交点为图像物理坐标系$o-xy$的坐标原点。公式 4-2 即为相机坐标系到图像物理坐标系的转换关系。

（4）图像物理坐标系与像素坐标系之间的转换关系

设像素坐标系$O-uv$，O为坐标原点，u轴，v轴分别平行于x轴以及y轴，那么图像物理坐标系和像素坐标系之间，相差了一个缩放和一个原点的平移。假设像素坐标在u轴缩放了α倍，在v轴缩放了β倍，原点平移了$[c_x,c_y]^T$，可以得到成像

点 $p'(x,y)$ 与像素坐标 $[u,v]^T$ 之间的关系为下式所示，

$$\begin{cases} u = \alpha x + c_x \\ v = \beta y + c_y \end{cases} \tag{4-3}$$

带入公式 4-2，将 αf 和 βf 分别用 f_x，f_y 表示，f_x，f_y 单位为像素，得到下式，

$$\begin{cases} u = f_x \dfrac{X_c}{Z_c} + c_x \\ v = f_y \dfrac{Y_c}{Z_c} + c_y \end{cases} \tag{4-4}$$

用矩阵形式表示为下式，

$$Z\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} \triangleq KP \tag{4-5}$$

其中 K 为内参数矩阵 $\begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}$，相机在出厂之后内参一般是固定不变的。

（5）相机坐标系和世界坐标系之间的转换关系

相机坐标系和世界坐标系之间的转换关系也就是通常所说的相机的外参数，它会随着相机的运动发生改变，也就是视觉 SLAM 中求解的相机位姿，它代表着相机的轨迹。设世界坐标系为 $O_x - X_w Y_w Z_w$，在相机的运动过程中，相机坐标系 $O_c - X_c Y_c Z_c$ 与世界坐标系之间是不一定完全重合的，它们之间存在着一个刚体变换过程，包括一个旋转矩阵 R 和一个平移向量 t，也就是外参数，如下式 4-6 所示，为相机坐标系和世界坐标系之间刚体变换过程的公式，

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = R\begin{pmatrix} X_w \\ Y_w \\ Z_w \end{pmatrix} + t \tag{4-6}$$

其中 R 为 3×3 的正交矩阵，t 是一个三维平移向量，用齐次坐标表示可得下式

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{pmatrix} = \begin{pmatrix} R_{3\times3} & t_{3\times1} \\ 0 & 1 \end{pmatrix}\begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} = M\begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \tag{4-7}$$

其中 $M=\begin{pmatrix} R_{3\times3} & t_{3\times1} \\ 0 & 1 \end{pmatrix}$，即外参数矩阵。

根据公式 4-6 和公式 4-7 可得下式，

$$Z_c\begin{pmatrix} u \\ v \\ 1 \end{pmatrix}=\begin{pmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} R_{3\times3} & t_{3\times1} \\ 0 & 1 \end{pmatrix}\begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}=KTP_w \tag{4-8}$$

对于一个空间点 P 经过外参以及内参后，可以唯一确定它的图像点的位置，也就是它的像素位置。反过来，知道了相机的位姿（即旋转矩阵 R 和平移向量 t），也就可以求出空间点 P 的三维坐标。

（6）前端数据的采集将 Kinect2.0 深度传感器作为场景图像接收采集器，前端是目标场景三维数据采集的过程，包括数据采集、图像处理、视觉捕捉三部分[28]。在基于视觉 SLAM 的虚拟现实空间定位系统中，现实空间场景中所有的图像数据采集都是通过 Kinect2.0 获取的，但是在实际情况当中，Kinect2.0 直接获取的深度数据一般都会出现许多噪音，所以要对采集到的数据进行降噪处理。由 Kinect2.0 采集到的三维目标物体的坐标信息可以推算出深度数据点的相邻关系，在此基础上，将采集到的深度数据建立一张网格表，其中网格信息的生成需要加上相应的约束，设定一个阈值来确定同一条边上的两个顶点的深度变化范围，这里阈值取顶点深度最小值和最大值之间的差，再乘上一个 0.1。接着去除没有连接边的孤立顶点，这些孤立顶点包括一些杂点，还有成像质量较差的边界点。最后为了提高网格表的质量，使用双边滤波算法[36]对网格进行去噪，完成对整个深度图像数据的预处理过程。

4.2.2 中端实时优化

前端数据的采集和处理，获得了场景中物体的颜色和深度信息，当相机在现实空间中移动时，要实时得到相机在现实空间中的具体位置，就需要知道相机的相对位姿。在基于视觉 SLAM 的虚拟现实空间定位系统中，系统中端主要负责相机在现实空间中的位姿优化问题。由于相机在现实空间中运动采集图像数据的过程中，场景的视觉特征有时会发生明显的变化，所以此时本文提出的基于图像特征的视觉里程计自适应算法能发挥自身的优势，既能适应场景特征的变化，也能相对稳定、准确的估计相机的运动。

4.2.3 基于图像特征的视觉里程计自适应算法的设计

在前几个章节理论知识和研究的基础上，经过认真、深入的思考后发现，暂时

不存在一种适用于各种视觉特征环境的视觉里程计定位估计算法，不同的视觉特征场景可以采用不同的视觉里程计定位估计算法。特征点法和直接法两种视觉里程计定位估计算法的原理不同，在各自适用的场景下有各自独特的优势，本文结合两种算法的优点，提出了基于图像特征的视觉里程计自适应算法。

（1）算法的主要思想。

通过分析特征点法和直接法的优缺点可知，在图像视觉特征丰富的场景中，由于特征明显，特征点法能够提取到足够的、有效的特征点，此时特征点法相对来说更加具有优势。而在图像视觉特征缺乏的场景中，特征点法存在极大的不稳定性，甚至会因为特征点不足而导致算法失效，而直接法却能正常工作，得到可靠的结果，所以此种情形下，直接法体现了较强的鲁棒性[23]。

为了使视觉里程计在优化相机位姿、估计相机运动方面不受图像视觉特征的限制，本文提出了基于图像特征的视觉里程计自适应算法的设计思想，它从特征点法和直接法在不同图像视觉特征条件下的优势出发，结合了两者各自的特点。它的主要内容包括：

1.判断目标场景当中图像视觉特征的状态（丰富或者缺乏）。

2.若图像视觉特征丰富则使用特征点法来估计相机运动。

3.若图像视觉特征缺乏则使用直接法来估计相机运动。

（2）算法的流程。

如下图 4-3 所示，为基于图像特征的视觉里程计自适应算法的流程图。

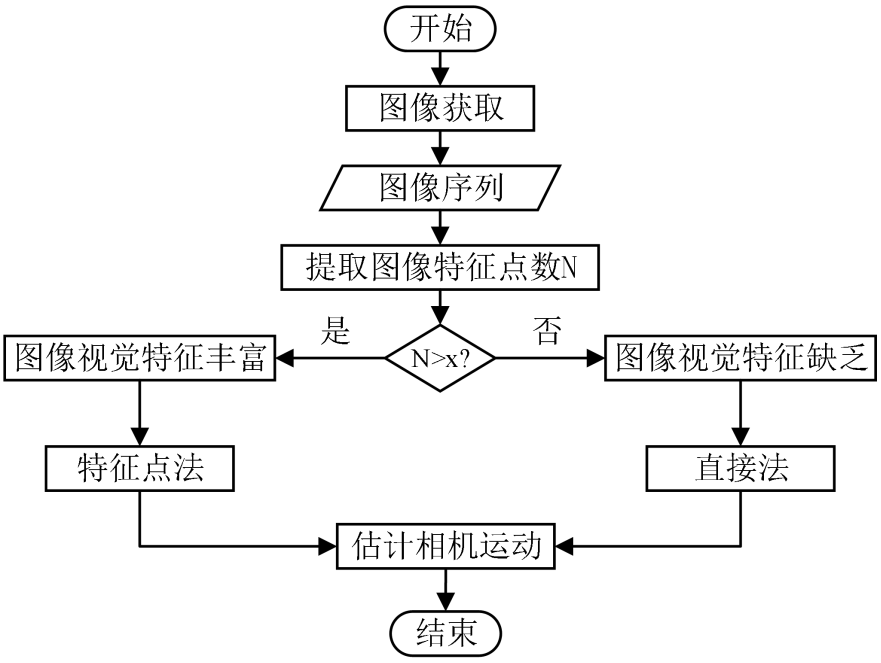


图 4-3 基于图像特征的视觉里程计自适应算法的流程图

算法的主要流程包括：

1.采用 Kinect2.0 采集目标场景的图像数据，获得相应的图像序列数据集。

2.对图像序列数据集进行 ORB 有效特征提取，记录特征点的数量。

3.判断特征点的数量 N 与图像特征参考值 x 之间的大小关系，若特征点数 $N > x$，则认为图像视觉特征丰富，反之若特征点数 $N < x$，那么认为图像视觉特征缺乏。

4.若图像视觉特征丰富，则采用基于特征点法的视觉里程计算法估计相机运动。若图像视觉特征缺乏，则采用基于像素信息的直接法估计相机运动。

（3）算法的实现

1.图像的采集以及预处理。图像序列的集合由 Kinect2.0 传感器采集目标场景后得到，图像序列中的每一张 RGB-D 图像包含颜色图像以及对应的深度图像，在构建高斯金字塔之前，需要将颜色图像转换为灰度图像，高斯金字塔的构建使得图像特征点的检测能在不同的尺度变换空间中进行，高斯金字塔中每一组图像对应相应的尺度空间[15]，例如目标场景中图像结构较小的对应金字塔的低尺度的特征点，这样保证了图像中特征点重复出现，或者特征点对应的尺度发生变化也能正常工作。

2.提取特征点。基于图像特征的视觉里程计自适应算法中特征点法部分采用了 ORB 特征提取算法，ORB 特征相对于 SIFT 特征以及 SURF 特征来说，FAST 关键点不具有方向性的限制在 ORB 特征中得到了有效的解决，二进制描述子 BRIEF 加入后，用汉明距离对其进行度量，不仅得到了准确的匹配点对，也使得整个图像特征提取过程节约了大量的时间，对于实时的图像特征来说，ORB 算法在时间上占有绝对优势，所以 ORB 特征提取算法非常具有代表性。

3.根据图像视觉特征确定视觉里程计自适应算法。Kinect2.0 传感器在目标场景中移动采集图像数据序列的过程中，目标场景的图像视觉特征并不是一成不变的，有时图像视觉特征会突然发生骤降，例如由图像视觉特征相对丰富的室内环境移动到两边都是白墙的空荡走廊，也有时图像视觉特征会突然猛增，例如在室内环境中，当相机传感器的视角发生变化，由视觉特征缺乏的木地板环境转到物体远近摆放且颜色丰富的环境。图像视觉特征的数量 N 代表每一帧图像提取到的特征点数量。基于图像特征的视觉里程计自适应算法根据提取到的特征点数量 N 与视觉特征参考值 x 大小关系，来确定采用特征点法还是直接法来估计相机运动，经过多次实验对比研究，最终确定 x 的值为 300，也就是说当图像的特征点数量大于 300，那么认为图像视觉特征丰富，采用特征点法进行相机定位估计，当图像的特征点数量小于 300 时，认为图像视觉特征缺乏，采用基于像素信息的直接法估计相机运动。

4.估计相机运动。

a）当提取到的图像特征点数量$N>300$时，利用基于 ORB 算法的特征点法求得相机位姿。通过对 Kinect2.0 传感器采集的图像进行特征点的提取、分析、匹配后，得到了许多组匹对好的点，这些点是带有深度信息的$3D$ 点，此时估计相机的运动也就是$3D-3D$的位姿估计问题，具体过程如下：

假设有两帧已经匹配的 RGB-D 图像，其中有一组配对好的$3D$ 点P、P'，分别为$P=\{p_1,p_2,...,p_n\}$，$P'=\{p'_1,p'_2,...,p'_n\}$，求出$P$到$P'$之间的变换关系就能估计相机的运动，$P$到$P'$之间可以通过一次旋转$R$和一次平移$t$得到，$P$、$P'$、$R$、$t$之间满足以下公式：

$$1\le i\le n,\forall i,p_i=Rp'_i+t \tag{4-9}$$

因为两组$3D$ 点之间的变换关系和相机模型并没有直接联系，要求解R和t，可以利用迭代最近点（Iterative Closest Point，ICP）算法来求解$3D-3D$的位姿估计问题。ICP 分为线性代数求解法和非线性优化求解法，下面主要介绍线性代数法中的奇异值分解算法（Singular Value Decomposition，SVD）来求解R和t。首先用e_i表示第i对点的误差项，如下式所示：

$$e_i=p_i-\left(Rp'_i+t\right) \tag{4-10}$$

我们的目标是要通过求解最小化误差项e_i的平方和（用J表示）来求得R和t，这里需要构建一个最小二乘问题，如下式所示：

$$\min_{R,t}J=\frac{1}{2}\sum_{i=1}^{n}\left\|\left(p_i-\left(Rp'_i+t\right)\right)\right\|_2^2 \tag{4-11}$$

要求得J，可以先将P、P'两组点的质心分别用$p=\frac{1}{n}\sum_{i=1}^{n}p_i$，$p'=\frac{1}{n}\sum_{i=1}^{n}\left(p'_i\right)$表示，那么误差函数的分解过程为如下所示：

$$\begin{aligned}
\min_{R,t}J&=\frac{1}{2}\sum_{i=1}^{n}\left\|\left(p_i-\left(Rp'_i+t\right)\right)\right\|^2\\
&=\frac{1}{2}\sum_{i=1}^{n}\left\|p_i-Rp'_i-t-p+Rp'+p-Rp'\right\|^2\\
&=\frac{1}{2}\sum_{i=1}^{n}\left\|(p_i-p-R(p'_i-p'))+(p-Rp'-t)\right\|^2\\
&=\frac{1}{2}\sum_{i=1}^{n}\left(\left\|p_i-p-R\left(p'_i-p'\right)\right\|^2+\left\|p-Rp'-t\right\|^2+2\left(p_i-p-R\left(p'_i-p'\right)\right)^T\left(p-Rp'-t\right)\right)
\end{aligned}$$

其中$\left(p_i-p-R\left(p_i'-p'\right)\right)$求和之后为 0，得到下式：

$$\min_{R,t}J=\frac{1}{2}\sum_{i=1}^{n}\left\|p_i-p-R\left(p_i'-p'\right)\right\|^2+\left\|p-Rp'-t\right\|^2 \tag{4-12}$$

在上式中，加号左边项只与R有关，加号右边项有R和t，与两个质心p和p'有关，所以只要知道R，令加号右边项为 0 就能求出t。ICP 算法包括以下三个步骤：

①求解质心p和p'，得到单个点的去质心坐标，如下式所示：

$$q_i=p_i-p,q_i'=p_i'-p' \tag{4-13}$$

②计算旋转矩阵R，如下式所示：

$$R^*=\arg\min_R\frac{1}{2}\sum_{i=1}^{n}\left\|q_i-Rq_i'\right\|^2=\frac{1}{2}\sum_{i=1}^{n}q_i^Tq_i+q_i'^TR^TRq_i'-2q_i^TRq_i' \tag{4-14}$$

③根据R求出t，如下式所示：

$$t^*=p-Rp' \tag{4-15}$$

在公式 4-14 中，与旋转矩阵R有关的项为：$-2q_i'^TRq_i'$，那么目标函数变为：

$$\sum_{i=1}^{n}-q_i^TRq_i'=\sum_{i=1}^{n}-tr\left(Rq_i'q_i^T\right)=-tr\left(R\sum_{i=1}^{n}q_i'q_i^T\right) \tag{4-16}$$

要求出最优的R，首先定义矩阵W，并对W进行 SVD 分解，其中W为3×3的矩阵，Σ为对角元素从大到小排列的奇异值对角矩阵，U、V为对角矩阵：

$$W=\sum_{i=1}^{n}q_iq_i'^T=U\Sigma V^T \tag{4-17}$$

当W满铁时，得到R：

$$R=UV^T \tag{4-18}$$

将R带入公式 4-15，求得t。

b）当提取到的图像特征点数量$N<300$时，利用基于半稠密的直接法求得相机位姿。直接法的推导：假设有一空间点P，设P在第一帧和第二帧图像当中的像素位置分别为p_1、p_2，将前一时刻相机坐标系作为参照系，前一时刻到后一时刻之间的旋转和平移分别为R，t，对应的李代数为ξ[22]，相机的内参为K，如下图 4-4 所示，为直接法的示意图。

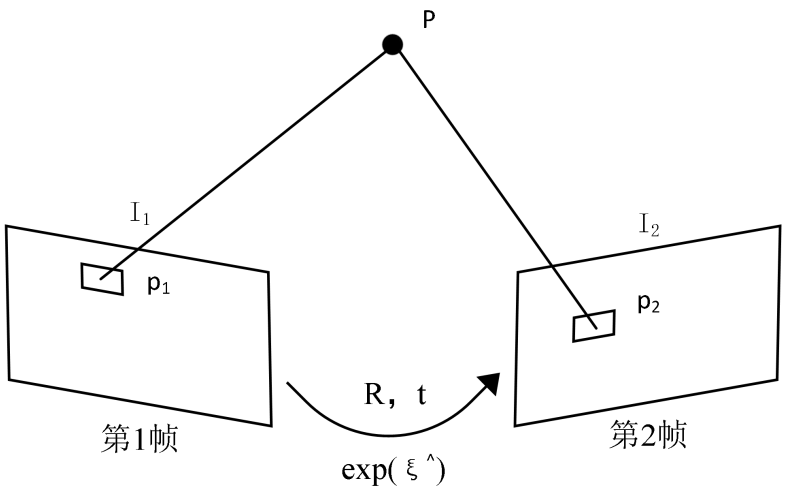


图 4-4 直接法示意图

其中投影方程为：

$$p_1=\begin{bmatrix}u\\v\\1\end{bmatrix}_1=\frac{1}{Z_1}KP \tag{4-19}$$

$$p_2=\begin{bmatrix}u\\v\\1\end{bmatrix}_2=\frac{1}{Z_2}K\left(RP+t\right)=\frac{1}{Z_2}K\left(\exp\left(\xi^{\wedge}\right)P\right)_{1:3} \tag{4-20}$$

其中 Z_1、Z_2 为深度，由于受光照环境、噪声等因素的影响，p_1 和 p_2 之间会存在亮度误差，即光度误差。因为直接法没有特征匹配，要找到 p_1 最相似的 p_2，直接法的求解可以抽象成求解最小化像素的光度误差。设标量 e 为两像素之间的光度误差，对于 N 个点来说，相机位姿的估计问题变为：

$$\begin{cases}\min\limits_{\xi} J(\xi)=\sum\limits_{i=1}^{N}e_i^Te_i\\ e_i=I_1(p_1,i)-I_2(p_2,i)\end{cases} \tag{4-21}$$

其中优化变量为李代数 ξ，表示相机的位姿，根据李代数的左扰动模型，可以分析误差 e 和相机位姿 ξ 的导数关系，如下式所示：

$$\begin{aligned} e(\xi \oplus \delta\xi) &= I_1\left(\frac{1}{Z_1}KP\right) - I_2\left(\frac{1}{Z_2}K\exp(\delta\xi^{\wedge})\exp(\xi^{\wedge})P\right) \\ &= I_1\left(\frac{1}{Z_1}KP - I_2\left(\frac{1}{Z_2}K\exp(\xi^{\wedge})P + u\right)\right) \\ &= e(\xi) - \frac{\partial I_2}{\partial u}\frac{\partial u}{\partial q}\frac{\partial q}{\partial \delta\xi}\delta\xi \end{aligned} \tag{4-22}$$

其中 $q = \delta\xi^{\wedge}\exp(\xi^{\wedge})P$ 是扰动分量在下一时刻相机坐标系下的坐标，记为 $q = [X, Y, Z]^T$，$u = \frac{1}{Z_2}Kq$ 为它的像素坐标，$\frac{\partial I_2}{\partial u}$ 是在 u 处的像素梯度，三维点的导数如下式所示：

$$\frac{\partial u}{\partial q} = \begin{bmatrix} \frac{\partial u}{\partial X} & \frac{\partial u}{\partial Y} & \frac{\partial u}{\partial Z} \\ \frac{\partial v}{\partial X} & \frac{\partial v}{\partial Y} & \frac{\partial v}{\partial Z} \end{bmatrix} = \begin{bmatrix} \frac{f_x}{Z} & 0 & -\frac{f_x X}{Z^2} \\ 0 & \frac{f_y}{Z} & -\frac{f_y Y}{Z^2} \end{bmatrix} \tag{4-23}$$

三维点对于变换的导数为 $\frac{\partial q}{\partial \delta\xi} = [I, -q^{\wedge}]$，合并公式 4-22 和公式 4-23 得下式：

$$\frac{\partial u}{\partial \delta\xi} = \begin{bmatrix} \frac{f_x}{Z} & 0 & -\frac{f_x X}{Z^2} & -\frac{f_x XY}{Z^2} & f_x + \frac{f_x X^2}{Z^2} & -\frac{f_x Y}{Z} \\ 0 & \frac{f_y}{Z} & -\frac{f_y Y}{Z^2} & -f_y - \frac{f_y Y^2}{Z^2} & \frac{f_y XY}{Z^2} & \frac{f_y X}{Z} \end{bmatrix} \tag{4-24}$$

得到误差相对于李代数的雅可比矩阵为：

$$J = -\frac{\partial I_2}{\partial u}\frac{\partial u}{\partial \delta\xi} \tag{4-25}$$

接着采用列文伯格-马夸尔特方法计算增量，迭代求解，最后得到优化时间以及相机位姿。

本文选用的直接法是半稠密（Semi-Dense）直接法，即丢弃像素梯度不明显的像素，只采用有明显梯度的像素，由上面推导我们可以知道，直接法的求解是一个优化问题，那么可以将这个优化问题转化为图优化的方法来解，利用 g2o[32][40][41]优化库帮助求解，将优化变量也就是相机位姿作为图优化的顶点，将单个像素的光度误差作为图优化边来构建优化问题。

4.2.4 后端同步定位

在本文基于视觉 SLAM 的虚拟现实空间定位系统中，后端的同步定位包括现实

空间与虚拟空间相机的同步定位，当相机在现实空间中运动时，相机观察到的现实空间场景随着相机在现实空间中的三维坐标的变化而变化，同时，虚拟空间的场景与现实空间的场景同时发生变化，只是各自空间相机观察到的内容不同而已。

（1）现实空间与虚拟空间相机的同步定位，即相机在现实空间中移动一定距离时，那么同样会反应在虚拟空间中。要实现现实空间与虚拟空间之间的移动状态通信，首先必须要确定相机在现实空间中的移动，通过前端图像数据的采集、中端相机位姿的实时优化，可以得到相机在现实空间中的移动状态，相机三维坐标信息的变化，也就是相机在连续时间内位姿变换的结果。在视觉 SLAM 前端视觉里程计中，相机位姿变化的结果一般包括一个旋转矩阵 R 和一个平移向量 t，即一个朝向信息，一个位置信息，相机位姿的这两个值由系统中端相机位姿的实时优化可以得到，后端同步定位的任务就是将相机位姿变换的结果传送给虚拟空间，然后虚拟空间中的相机位姿根据现实空间中相机位姿变换的结果进行相应的调整，这就实现了现实空间与虚拟空间之间的相机位姿同步定位。

如下图 4-5 所示，为现实空间中相机位姿变换结果传送给虚拟空间的过程。

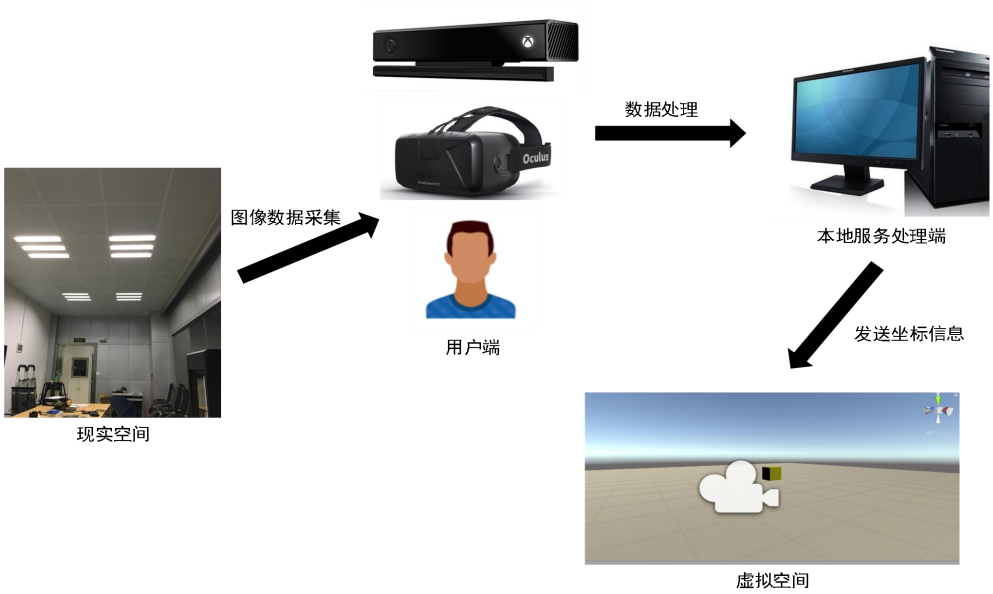


图 4-5 现实空间中相机位姿变换结果传送给虚拟空间的过程

（2）由上图可知，本地服务处理端接收到现实空间中的相机位姿变换结果，然后传送给虚拟空间，这个过程涉及到了本地进程间的数据通信。Linux 系统进程之间的数据通信机制一共有 6 种[5]，分别为管道（Pipe）、信号量（Semophore）、消息队列（Message Queue）、信号（Signal）、共享内存（Shared Memory）、套接字（Socket），如下表 4-1 所示，为 Linux 系统进程之间数据通信机制的特点。

表 4-1 Linux 系统进程之间数据通信机制的特点

通信方式	特点
管道（Pipe）	只支持单项数据流，双向通信需要创建两个管道，而且只能用于亲缘关系进程之间的通信。
信号量（Semophore）	是一个计数器，控制多个进程对共享资源的访问
消息队列（Message Queue）	具有同步机制，提供有格式的字节流
信号（Signal）	通信携带的信息极少，不适合携带数据的通信
共享内存（Shared Memory）	提供一段内存由多个进程共同使用，是最快的进程间的通信方式
套接字（Socket）	用于本地单机或者跨网络通信

（3）现实空间服务端与虚拟空间客户端之间的通信。本文选用的是套接字（Socket）的进程间通信方式[13]，由表 4-1 可以知道，套接字这种通信机制，可以用于本地单机，也可跨网络进行，凭借着这种机制的特点，客户端和服务端这两个要进行通信的进程之间相关的开发工作可以在本地单机上进行，建立双向的通信，Socket 进程通信和网络通信使用的是统一套接口，只是地址结构与某些参数不同。

如下图 4-6 所示，为 Linux 系统下 Socket 进行本地进程间通信的主要流程。

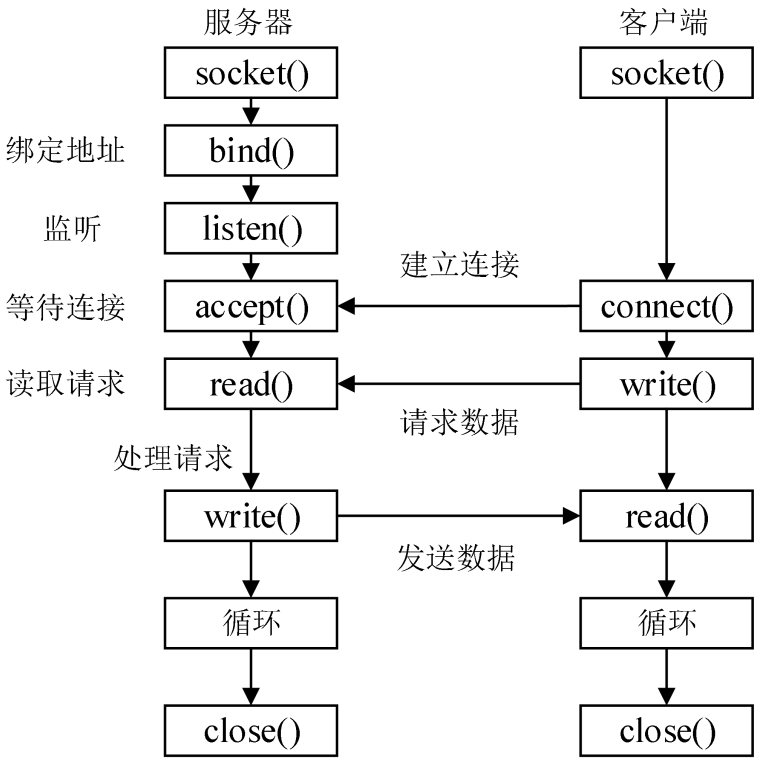


图 4-6 Linux 系统下 Socket 进行本地进程间通信的主要流程

（4）位姿变换结果的封装和序列化。将现实空间服务端中相机位姿变换结果的朝向信息和位置信息用两个 Float 类型的数组进行封装，然后将其序列化成二进制 Byte 类型的数组传输到虚拟空间客户端，虚拟空间中设置了两个 Camera，接收到的是一样的数据，有着同样的位姿变换。首先需要在服务端与客户端建立 2 个 Socket 类，一个用于发送数据，一个用于接收数据，客户端需要向服务端发起连接，服务端接收连接消息后开始向客户端发送数据，客户端接收到数据后将其反序列化后再进行封装。

（5）如图 4-7 所示，为基于视觉 SLAM 的虚拟现实空间定位系统具体实现的流程图。

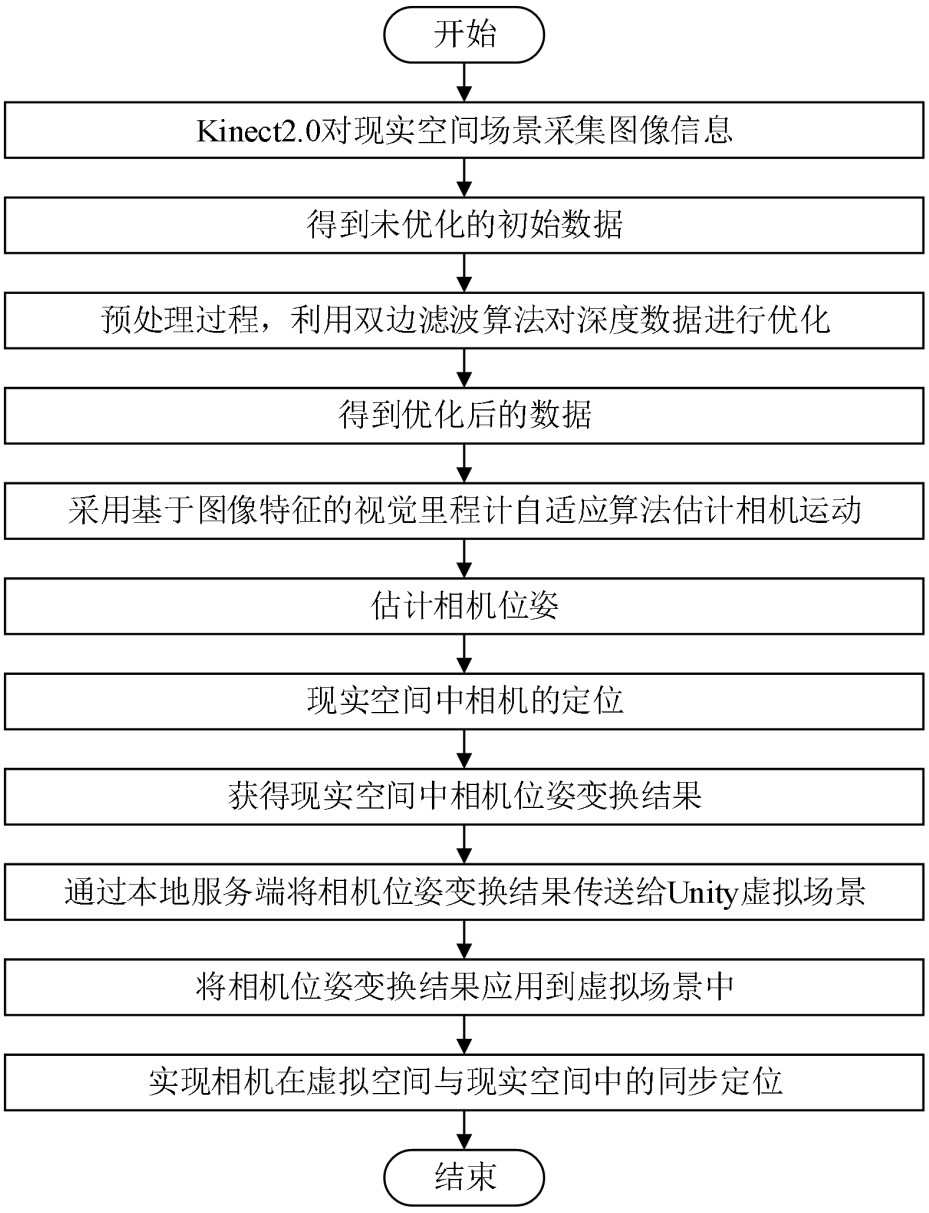


图 4-7 基于视觉 SLAM 的虚拟现实空间定位系统具体实现的流程图

4.3 系统的平台搭建

系统开发平台的搭建包括以下内容：

（1）Kinect2.0 驱动平台由机器人操作系统和 libfreenect2 驱动两部分组成。

1.机器人操作系统（Robot Operating System，ROS）：ROS 是一个开源的机器人操作系统，有类似于操作系统的功能，支持 C++、Python 等多种编程语言的开发。ROS 主要分为两个部分：文件系统层，计算图层[2][24]。

文件系统层主要包括 ROS 中的一些源文件，如下图 4-8 所示，为文件系统层包括的内容。

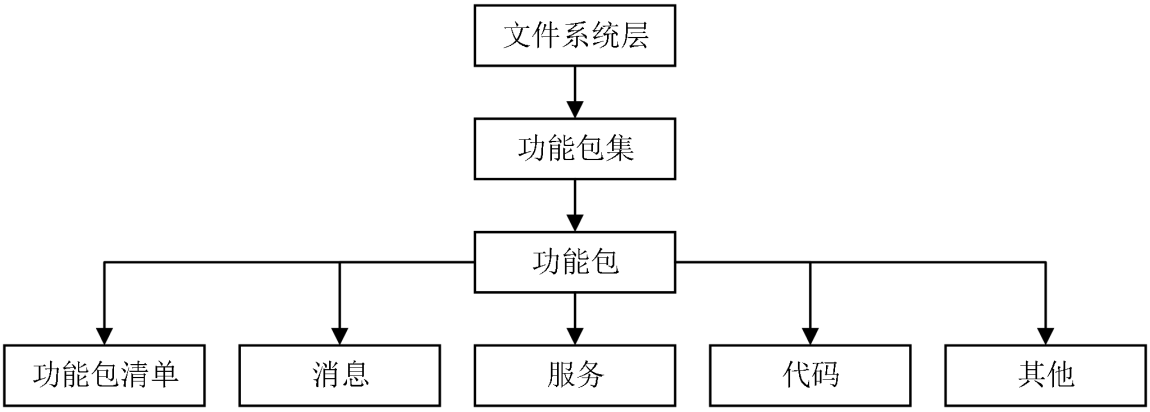


图 4-8 文件系统层

计算图层是 ROS 系统用来处理数据的点对点的通信网络。如下图 4-9 所示，为计算图层包括的内容。

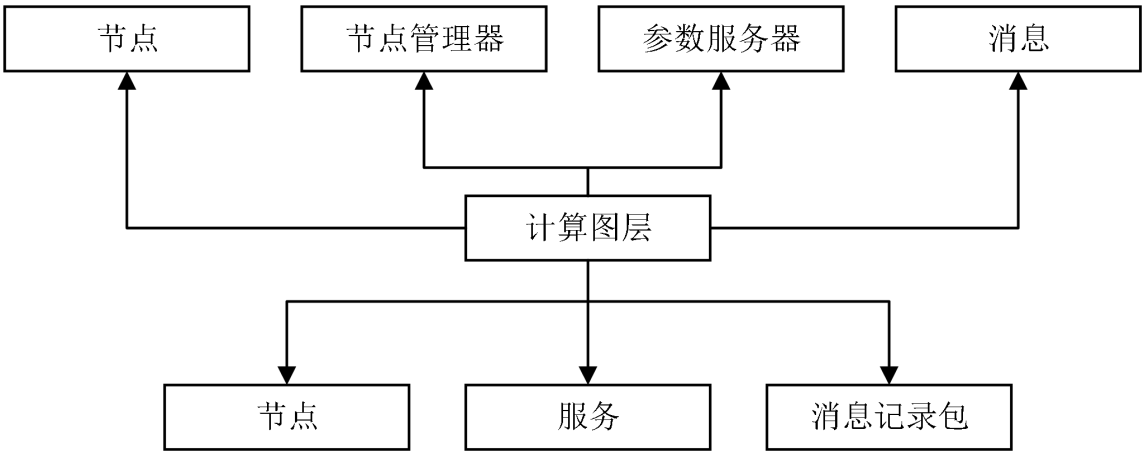


图 4-9 计算图层

2.libfreenect2 驱动：libfreenect2 是 Kinect2.0 的一个跨平台开源驱动，只支持 USB3.0，libfreenect2 开源驱动支持 RGB 图像、红外 IR、深度图像的获取，同时支持 RGB 和深度图像的校准，还支持 GPU 加速。在 ROS 系统中使用 Kinect2.0 时，需要用到 iai_kinect2 驱动，iai_kinect2 驱动负责 Kinect2.0 和 ROS 系统之间联系，本文要使用的 iai_kinect2 工具库包括 Kinect2_calibration，Kinect2_registration，Kinect2_bridge，Kinect2_viewer 四个。如下表 4-2 所示，为四个工具库的具体作用。

表 4-2 iai_kinect2 工具库

工具库名称	具体作用
Kinect2_calibration	完成 Kinect2.0 的红外传感器、RGB 传感器的校正工作以及深度值的测量工作
Kinect2_registration	主要用于深度数据的配准
Kinect2_bridge	连接 ROS 和 libfreenect2 驱动
Kinect2_viewer	负责图像数据集的可视化工作

（2）Unity3D：Unity3D 是 Unity Technologies 开发的一个 3D 游戏引擎，是一个多平台的游戏开发工具，用户可以用 Unity3D 创建三维动画，三维游戏等 3D 互动内容，本文使用 Unity3D 创建实验所需要的虚拟空间场景并发布到 Linux 上。如下图 4-10 所示，为 Unity3D 创建的虚拟空间场景示例。

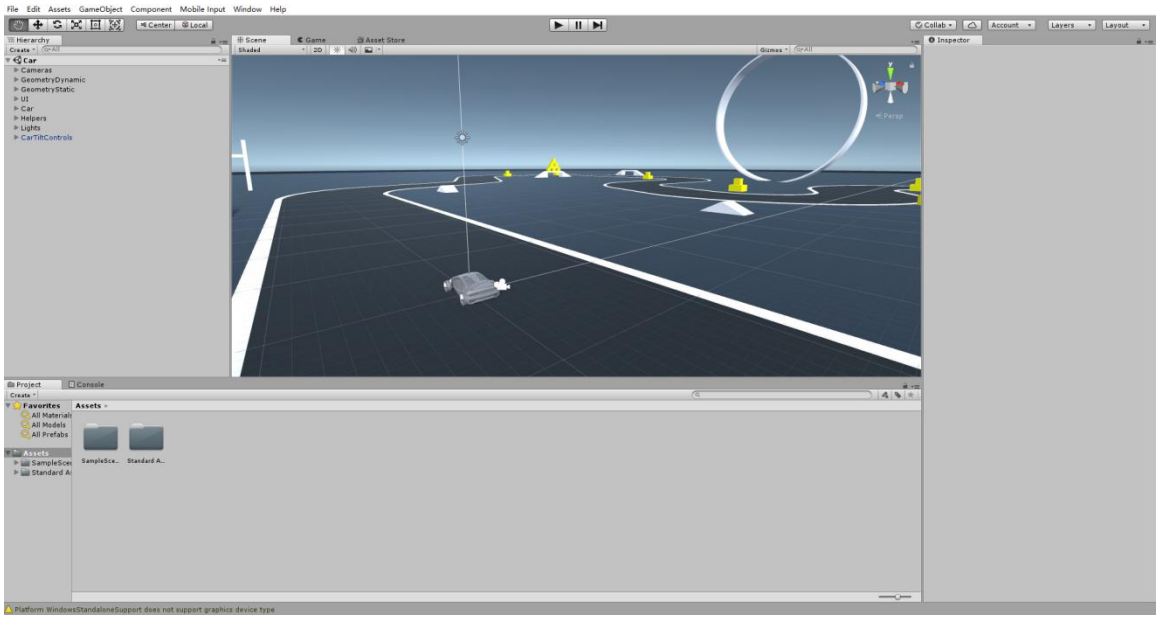


图 4-10 Unity3D 创建的虚拟空间场景示例

4.4 实验过程与结果分析

4.4.1 算法的实验验证

实验是验证一个算法是否合理，是否适用于实际环境的有效手段，完整、严谨的实验过程是保证算法得到有效结果的前提，对于视觉 SLAM 的视觉里程计定位估计算法来说，不同的硬件环境会有不同的结果，比如算法的时间等，同时运行算法需安装必要的软件工具库。为了验证基于图像特征的视觉里程计自适应算法的有效性，本小节搭建了相应的硬件以及软件平台，分别在不同图像视觉特征的室内环境中进行实验，并对实验结果进行了分析。将 Kinect2.0 深度传感器采集的图像序列作为实验数据，实验地点选在贵州师范大学贵州省信息与计算科学重点实验室。

（1）实验装置。如下图 4-11 所示，为算法验证实验的装置图。

图 4-11 算法验证实验的装置图

（2）软硬件平台参数。如下表 4-3 所示，为算法验证实验的硬件平台以及软件平台的具体参数。

表 4-3 算法验证实验的硬件平台以及软件平台的具体参数

硬件平台（PC 机及传感器设备）	软件平台
一台 Microsoft Xbox One Kinect2.0 传感器	Ubuntu14.04 操作系统
Intel 酷睿 I5 6200U CPU	KDevelop 4.6 集成开发环境，开发语言为 C++
8GB 内存	OpenCV、g2o 库等

（3）实验步骤。

首先，选择三组图像特征具有显著差异的环境进行图像数据集的采集工作，得到三组不同环境的图像序列数据集。

然后，在这三组图像序列数据集中，选择每一组的前 500 帧图像数据进行实验。

最后，分别采用基于 ORB 算法的特征点法、基于像素信息的半稠密直接法以及基于图像特征的自适应算法，对选择的三组实验数据进行位姿估计，分别得到三组数据的优化时间均值以及特征提取成功、失败的帧数。

1.实验验证的第一组环境选择了光学条件良好的室内环境，此环境的图像视觉特征复杂，颜色丰富，非常适用于做本次研究的实验环境。如下图 4-12 所示，为光学条件良好的室内环境图像序列数据集中某一帧的颜色图像以及深度图像。

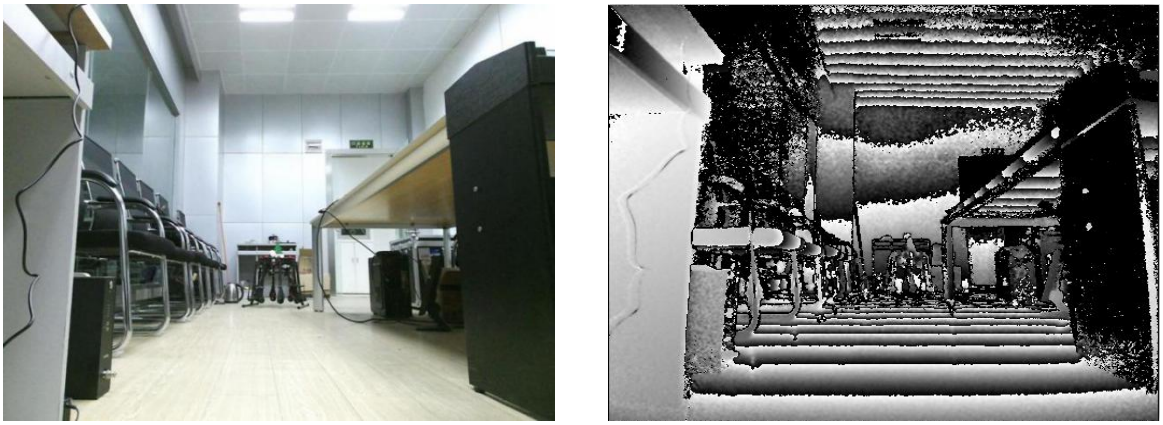


图 4-12 光学条件良好的室内环境图像序列数据集中某一帧的颜色图像以及深度图像

2.实验验证的第二组环境选择了光学条件良好的木地板环境，此环境图像视觉特征简单，颜色单一，如下图 4-13 所示，为光学条件良好的木地板环境图像序列数据集中某一帧的颜色图像以及深度图像。

图 4-13 光学条件良好的木地板环境图像序列数据集中某一帧的颜色图像以及深度图像

3.实验验证的第三组环境选择了光学条件恶劣的室内环境，受光照条件的影响，该环境图像视觉特征缺乏，且分布不均匀，如下图 4-14 所示，为光学条件恶劣的室内环境图像序列数据集中某一帧的颜色图像以及深度图像。

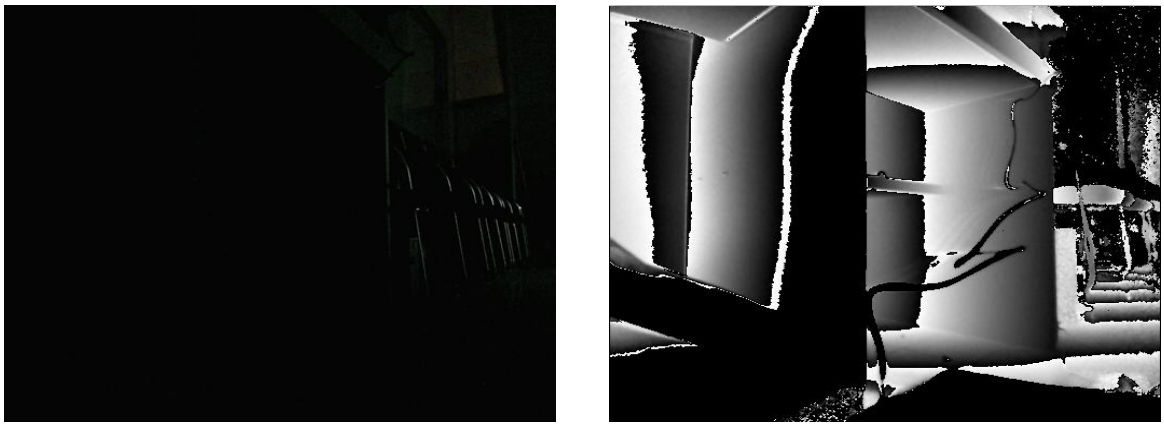
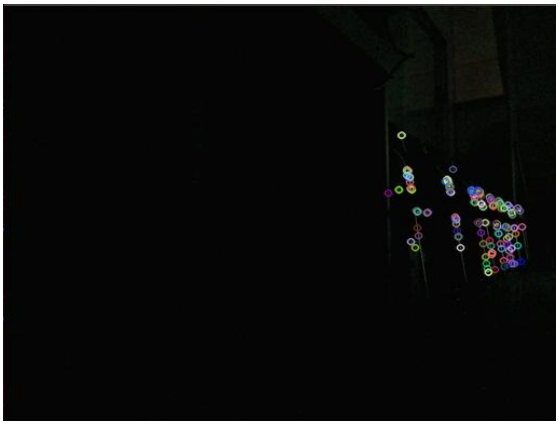


图 4-14 光学条件恶劣的室内环境图像序列数据集中某一帧的颜色图像以及深度图像

（4）算法验证。如下图 4-15 所示，为三组数据的各一帧图像进行 ORB 特征提取的结果示例。

（a）第一组环境 ORB 特征点提取　　（b）第二组环境 ORB 特征点提取



（c）第三组环境 ORB 特征点提取

图 4-15 ORB 特征提取的结果示例

第一组、第二组、第三组环境所采集到的图像序列数据集，分别采用基于 ORB 算法的特征点法、基于像素信息的半稠密直接法以及基于图像特征的自适应算法进行位姿估计。

（5）实验结果和分析。

1.采用基于 ORB 算法的特征点法进行位姿估计的实验结果，如表 4-4 所示。

表 4-4 基于 ORB 算法的实验结果

算法类别	组号	环境条件	优化时间均值（单位/秒）	特征提取成功的帧数	特征提取失败的帧数
基于 ORB 算法的特征点法	第一组	光学条件良好的室内环境	0.0053	421	79
	第二组	光学条件良好的木地板环境	0.0021	356	144
	第三组	光学条件恶劣的室内环境	0.0011	102	398

2.采用基于像素信息的半稠密直接法进行位姿估计的实验结果，如表 4-5 所示。

表 4-5 基于像素信息的半稠密直接法的实验结果

算法类别	组号	环境条件	优化时间均值（单位/秒）	成功的帧数	失败的帧数
基于像素信息的半稠密直接法	第一组	光学条件良好的室内环境	0.0368	436	64
	第二组	光学条件良好的木地板环境	0.0186	412	88
	第三组	光学条件恶劣的室内环境	0.0099	368	132

3.采用基于图像特征的自适应算法进行位姿估计的实验结果，如表 4-6 所示。

表 4-6 基于图像特征的自适应算法的实验结果

算法类别	组号	环境条件	优化时间均值（单位/秒）	特征提取成功的帧数	特征提取失败的帧数
基于图像特征的自适应算法	第一组	光学条件良好的室内环境	0.0102	491	9
	第二组	光学条件良好的木地板环境	0.0093	472	28
	第三组	光学条件恶劣的室内环境	0.0030	408	92

4.结果分析。通过比较表 4-4，4-5，4-6 中不同图像视觉特征的三组环境图像序列数据集实验结果可知，在面对不同图像数据视觉特征环境的情况下，可以发现本文提出的基于图像视觉特征的视觉里程计自适应算法，优化时间上符合实时性的要求，同时在稳定性方面更加具有优势。在图像视觉特征丰富的环境中，能够根据提取到的特征点数量，选择合适的视觉里程计算法进行位姿估计，特征提取成功的帧数最多，说明本文提出的优化改进算法实用性更好。在图像视觉特征缺乏的环境中，基于图像视觉特征的视觉里程计自适应算法仍然可以进行正确的位姿估计，特征提取成功的数量也比较可观，说明本文的优化改进算法具有良好的鲁棒性。

4.4.2 系统的实验结果展示

本小节搭建相应的硬件以及软件平台，将基于视觉 SLAM 的虚拟现实空间定位系统的实验结果展示出来，实验地点选在贵州师范大学贵州省信息与计算科学重点实验室。

（1）实验装置。如图 4-16 所示，为系统实验的装置图：

图 4-16 系统实验的装置图

（2）软硬件平台参数。如下表 4-7 所示，为实验硬件和软件平台的具体参数。

表 4-7 实验硬件和软件平台的具体参数

硬件平台（PC 机及传感器设备）	软件平台
一台 Microsoft Xbox One Kinect2.0 传感器	Ubuntu14.04 64 位操作系统
一台 Oculus Rift Development Kit 2	Unity 3D Personal 版本
PC 机配置： Intel 酷睿 I5 6200U CPU 8GB 内存 NVIDIA Geforce 940M 独立显卡	ROS 系统 Indigo 版本，开发语言为 C++
	Kinect libfreenect2 开源驱动
	OpenCV、g2o 库等

（3）实验选择的现实空间场景和创建的虚拟空间环境。利用 Unity3D 搭建一个简单的室内虚拟空间场景并发布到 Linux 中，为了使用户从视觉上感觉到在虚拟场景中的实时变换，在虚拟场景中放入了一些虚拟物体作为参照物，如电视机、沙发、电脑、衣柜、桌子等。如下图 4-17，4-18 所示，分别为实验选择的现实空间场景和实验创建的虚拟空间场景示例。

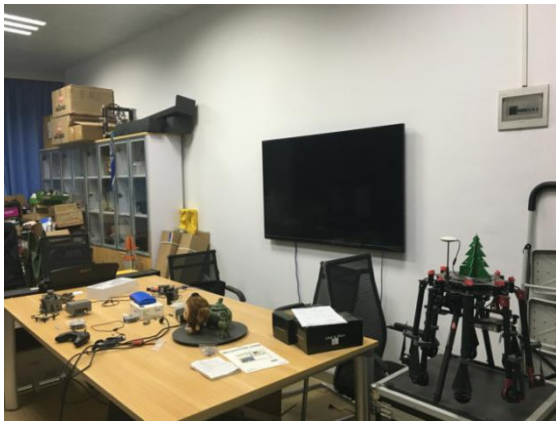


图 4-17 实验选择的现实空间场景示例

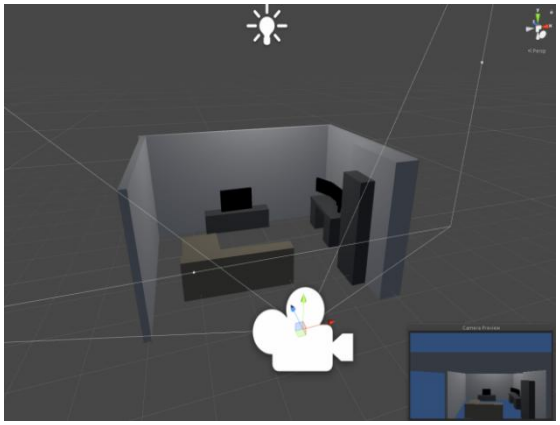


图 4-18 实验创建的虚拟空间场景示例

（4）用户带上实验装置在现实空间中移动，会从视觉上实时的感受到虚拟场景的变化，比如虚拟场景中摆放的物体离自己的远近变化，自己在虚拟空间中的位置变化。为了体现实验的对比性以及同步性，分别在现实空间和虚拟空间录制的实验过程视频流中，间隔相同时间各自取一帧图像作为对比，如图 4-19，4-20，4-21，4-22，4-23 所示，为五组实验结果对比展示。

图 4-19 第一组实验结果对比展示

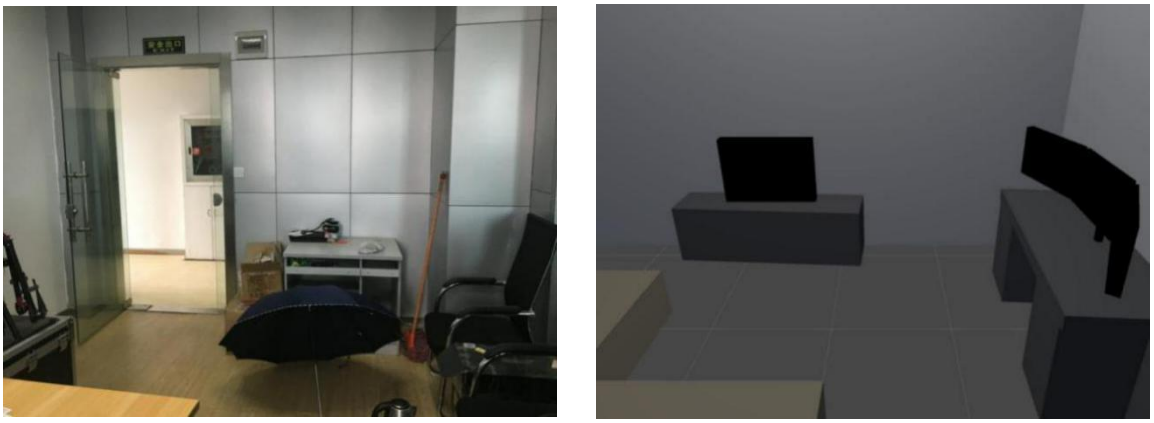


图 4-20 第二组实验结果对比展示

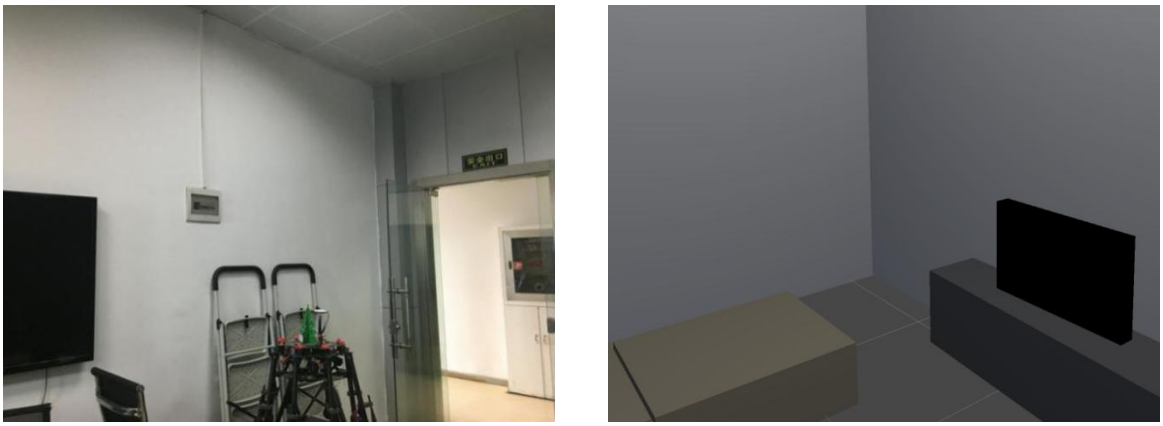


图 4-21 第三组实验结果对比展示

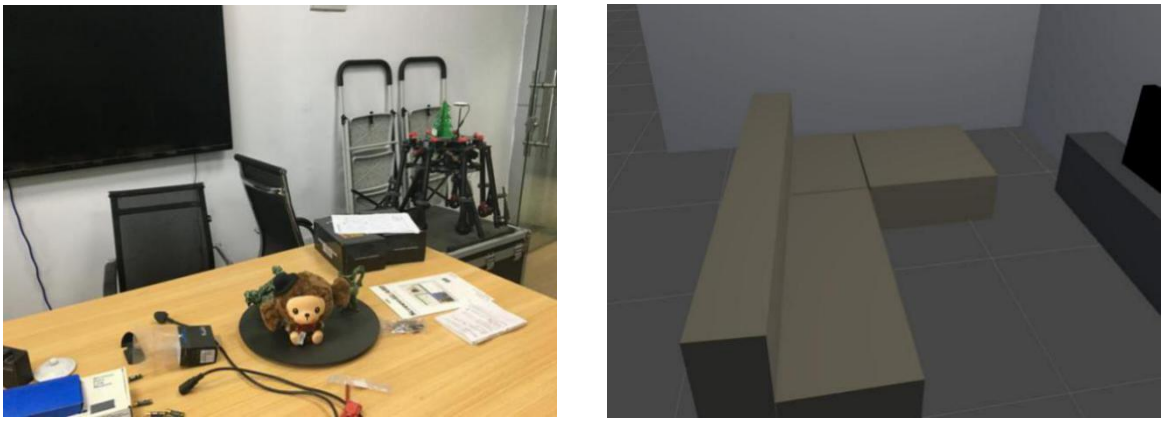


图 4-22 第四组实验结果对比展示

图 4-23 第五组实验结果对比展示

4.5 本章小结

本章的主要内容是基于视觉 SLAM 的虚拟现实空间定位系统的设计与实验，系统的最终目的是实现现实空间和虚拟空间场景中相机的同步定位，也就是场景的同步变化，让用户从视觉上实时的感受到虚拟空间场景的变化。

首先，系统的设计主要包括三个模块，前端数据采集，中端实时优化，后端同步定位。系统前端通过 Kinect2.0 采集现实场景中的图像数据，经过去噪处理后，系统中端将优化后的数据用于相机的位姿估计，这里本文结合了特征点法和直接法两种算法的特点，提出了基于图像特征的视觉里程计自适应算法来估计相机位姿，将得到的现实空间中相机位姿的变换结果通过 Linux 本地进程间的通信机制 Socket 传送到虚拟空间中，然后虚拟空间的相机位姿根据现实空间中的相机位姿结果（包括相机的朝向信息以及位置信息）进行调整，实现了相机在现实空间和虚拟空间中的同步定位。最后，通过实验验证了基于图像特征的视觉里程计自适应算法有更好的实时性和稳定性，同时通过几组实验数据对比，展示了基于视觉 SLAM 的虚拟现实空间定位系统的实验结果。

第五章 总结与展望

5.1 本文工作的总结与分析

SLAM 能够解决在没有环境先验条件下移动机器人的自主移动问题，视觉 SLAM 将相机作为传感器来收集移动机器人所处环境的空间信息来构建地图，移动机器人通过环境地图确定自身所处的位置，进行路线规划。虚拟现实技术借助外部传感器实现了沉浸式的交互体验，通过硕士研究生阶段的深入学习以及相关实践，对 SLAM 技术以及虚拟现实技术有了系统的了解，并掌握了相关的实现方法，在此基础上，对计算机视觉有了更进一步的认识。本文的主要工作如下：

收集相关资料，阅读大量 SLAM 技术以及虚拟现实技术的中外文献，为论文的研究奠定了理论基础。在此过程中，深入了解了 SLAM 技术和虚拟现实技术的国内外研究现状，对两者的难点和热点深入研究，确定了两者结合的可行性，明确了选题的背景及意义。

视觉 SLAM 是 SLAM 技术一个重要的分支，通过了解深度测量技术的主要原理以及相关分类，介绍了 Kinect 系列 RGB-D 传感器的工作原理，选择 Kinect2.0 深度传感器作为本文采集现实空间图像数据的工具，深入研究了基于 RGB-D 传感器的视觉 SLAM 技术，同时用数学公式抽象表述出视觉 SLAM 要解决的两个基本问题。

本文深入研究了视觉 SLAM 中两种主流的视觉里程计估计算法，即特征点法和基于像素信息的直接法，了解到两种方法的工作原理以及适用的场景各不相同，详细讨论了两种算法的优缺点，特征点法通过提取图像的特征以及匹配特征点对来估计相机位姿，但是无法面对图像特征缺失的环境，而直接法在特征缺失的环境鲁棒性好。

通过深入了解特征点法和直接法的优缺点，本文提出了基于图像特征的视觉里程计自适应算法，该算法可以根据图像视觉特征的状态进行正确的相机位姿估计，同时对该算法进行了具体的实验验证，实验结果表明该算法符合实时性的要求，同时该算法具有更好的鲁棒性。

本文结合了视觉 SLAM 以及虚拟现实技术，设计了基于视觉 SLAM 的虚拟现实空间定位系统，并且将本文提出的基于图像特征的视觉里程计自适应算法应用到系统中端相机位姿的实时优化部分，随后搭建了系统的实验平台，详细展示了系统的实验结果，用户能够通过此系统体验现实空间和虚拟空间的同步交互。

5.2 工作展望

通过硕士研究生期间的研究与项目经历，锻炼了自己的动手能力以及科研能力，但是由于自身能力水平的限制，加上研究时间有限，对视觉 SLAM 算法以及虚拟现实技术的研究还是比较浅显的，以后更加要严格要求自己，发挥更大的潜力。通过对本文认真的分析总结，还有一些问题需要进一步研究和探讨：

（1）在本文提出的基于图像特征的视觉里程计自适应算法中，图像视觉状态参考值 x 的值，虽然是经过大量的实验得出的，但是接下来需要进行多次校准实验，根据实验结果进一步深入的研究来确定 x 的最优值，提高算法的可靠性。

（2）本文主要研究的是视觉 SLAM 前端视觉里程计定位估计算法与虚拟现实技术的结合，在今后的研究工作中，可以考虑结合视觉 SLAM 框架中的其它部分，做一个完整、精确的虚拟现实空间定位系统，进一步提高系统的交互性和娱乐性。

致谢

又是一个阳光明媚的清晨，我坐在实验室座位上，一切都还是那么安静，转过头看了看窗外，还是一幅鸟语花香的场景，这是我生活了近三年的校园，脑海中满满的都是回忆，伴随着键盘的敲打声，我的硕士研究生生活也已经要接近尾声了，非常感激一路上关心我、帮助我、理解我的人，让我未来的生活充满了信心。

首先要感谢我的导师刘志杰教授，这三年一路走来，每一次的进步，每一次的研究成果都离不开刘老师的细心指导，他谦虚和蔼，每一次跟他的谈话都会让我受益匪浅，每一次我都会有新的思考，有些冥思苦想解决不了的问题，经过他的指点能让我茅塞顿开。感谢谢晓尧教授，谢老师一丝不苟、兢兢业业的教学科研态度，一直是我学习的榜样，也一直激励着我不断勇往直前，克服各种难题。感谢项目组的刘嵩老师、于徐红老师、但文红老师深深的教诲，让我从实际项目中锻炼自己，提高自己的动手能力。感谢我的两位师兄郭子选和王拓，你们给我树立了榜样，我万分感激。在此，我还要感谢实验室的所有老师，谢谢你们三年来的教导以及对我学习、生活的关注，你们创造的优秀的学术氛围深深地影响了我，希望你们工作顺利，生活美满！

其次我要感谢我的家人，谢谢你们一直以来的支持和鼓励，特别要感谢我的妈妈、姐姐、外公，你们的无私付出让我健康的成长，你们不断的鼓励是我坚强的后盾，希望你们青春永驻、身体健康！

然后我还要感谢的同学们，包括孙建忠、朱丹、王鹏、曹烁等，大家从五湖四海来到这个陌生的城市，我们相互包容、相互帮助，维系着我们共同的友谊，希望多年以后我们还是意气风发的模样。感谢我的师弟师妹们，包括赵圆圆师妹、吴招娣师妹、付子燨师弟等在学习生活当中的帮助。

一眨眼，三年了，仿佛我还是当时的懵懂少年，这三年有过快乐、有过痛苦、有过失望、有过收获，这些都是我成长的印记，都是我人生的故事，不管怎样，对生活、对未来还是要充满希望，希望这篇论文不会是我学术生涯的终点，继续努力、继续奋斗！

最后，要感谢各位老师在百忙之中抽出时间来参与我论文的审阅工作以及论文的答辩工作，我对各位老师表示由衷的感谢！

参考文献

[1] 陈浩磊,邹湘军,陈燕,陈燕,刘天湖. 虚拟现实技术的最新发展与展望[J].中国科技论文在线,2011,6(01):1-5+14.

[2] 陈炜灿. 基于 ROS 的机器人三维仿真平台设计与研究[D].东北大学,2014.

[3] 陈晓明,蒋乐天,应忍冬. 基于 Kinect 深度信息的实时三维重建和滤波算法研究[J].计算机应用研究,2013,30(04):1216-1218.

[4] 洪启松. 基于三维视觉技术的物体深度测量系统的研究[D].华南理工大学,2010.

[5] 黄茹. 浅析 Linux 环境下的进程间通信机制[J].科技信息,2014(14):96-97.

[6] 季秀才. 机器人同步定位与建图中数据关联问题研究[D].国防科学技术大学,2008.

[7] 李小红,谢成明,贾易臻,张国富. 基于 ORB 特征的快速目标检测算法[J].电子测量与仪器学报,2013,27(05):455-460.

[8] 李宇波,朱效洲,卢惠民,张辉. 视觉里程计技术综述[J].计算机应用研究,2012,29(08):2801-2805+2810.

[9] 廖爱国. 虚拟现实平台的研究与开发[D].同济大学,2006.

[10] 刘浩敏,章国锋,鲍虎军. 基于单目视觉的同时定位与地图构建方法综述[J].计算机辅助设计与图形学学报,2016,28(06):855-868.

[11] 刘雷杰. 基于 KinectV2.0 的真实感动画生成方法研究[D].天津大学,2016.

[12] 吕文婷. 虚拟世界就是特殊的现实世界[D]. 上海师范大学,2017.

[13] 欧军,吴清秀,裴云,张洪. 基于 socket 的网络通信技术研究[J].网络安全技术与应用,2011(07):19-21.

[14] 权美香,朴松昊,李国. 视觉 SLAM 综述[J].智能系统学报,2016,11(06):768-776.

[15] 屠礼芬, 仲思东, 彭祺,等. 基于高斯金字塔的运动目标检测[J]. 中南大学学报(自然科学版), 2013, 44(7):2778-2786.

[16] 王亚龙,张奇志,周亚丽. 基于 RGB-D 相机的室内环境 3D 地图创建[J].计算机应用研究,2015,32(08):2533-2537.

[17] 伍春洪,游福成,杨扬. 一种基于 3 维全景图像技术的深度测量方法[J].中国图象图形学报,2006(04):563-569.

[18] 徐则中. 移动机器人的同时定位和地图构建[D]. 浙江大学,2004.

[19] 许微. 虚拟现实技术的国内外研究现状与发展[J]. 现代商贸工业,2009,21(02):279-280.

[20] 杨江涛. 虚拟现实技术的国内外研究现状与发展[J].信息通信,2015(01):138.

[21] 袁志千. 基于双目视觉的移动机器人 SLAM 算法研究[D]. 天津理工大学,2016.

[22] 翟美新. 基于李群李代数的机器人运动特性分析与研究[D]. 南京理工大学,2015.

[23] 张国良, 姚二亮, 林志林,等. 融合直接法与特征法的快速双目 SLAM 算法[J]. 机器人, 2017, 39(6):879-888.

[24] 张鹏. 基于 ROS 的全向移动机器人系统设计与实现[D].中国科学技术大学,2017.

[25] 张松. 基于 RGB-D 传感器的地面移动机器人目标检测与跟踪[D].中北大学,2017.

[26] 张文玲,朱明清,陈宗海. 基于强跟踪 UKF 的自适应 SLAM 算法[J].机器人,2010,32(02):190-195.

[27] 赵沁平. 虚拟现实综述[J].中国科学（F 辑:信息科学）,2009,39(01):2-46.

[28] 周文. 基于 RGB-D 相机的三维人体重建方法研究[D].中国科学技术大学,2015.

[29] 朱笑笑,曹其新,杨扬,陈培华. 基于 RGB-D 传感器的 3D 室内环境地图实时创建[J].计算机工程与设计,2014,35(01):203-207.

[30] BAY H, TUYTELAARS T, GOOL L V. SURF:Speeded Up Robust Features[J]. Computer Vision & Image Understanding,2006,110(3):404-417.

[31] Byungjoon Chang,Woong Seo,Insung Ihm. On the Efficient Implementation of a Real-Time Kd-Tree Construction Algorithm[M].Springer Singapore:2015-06-15.

[32] Chi-Hoon Lee,Osmar R. Zaïane,Ho-Hyun Park,Jiayuan Huang,Russell Greiner. Clustering high dimensional data: A graph-based relaxed optimization approach[J]. Information Sciences,2008,178(23).

[33] Chung-Hsun Sun,Ying-Jen Chen,Yin-Tien Wang,Sheng-Kai Huang. Sequentially switched fuzzy-model-based control for wheeled mobile robot with visual odometry[J]. Applied Mathematical Modelling,2016.

[34] Davison Andrew J,Reid Ian D,Molton Nicholas D,Stasse Olivier. MonoSLAM: real-time single camera SLAM.[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence,2007,29(6):1052-1067.

[35] ENGEL J, SCHOPS T, CREMERS D. LSD-SLAM:Large-Scale Direct Monocular SLAM[M]// Computer Vision–ECCV 2014. Berlin German:Springer International Publishing,2014:834-849.

[36] Fleishman S, Drori I, Cohen-Or D. Bilateral mesh denoising[C]// Acm Siggraph. ACM, 2003:950-953.

[37] FORSTER C, PIZZOLI M, SCARAMUZZA D. SVO: Fast semi-direct monocular visual odometry[C]// IEEE International Conference on Robotics & Automation. New Jersey USA:IEEE,2014:15-22.

[38] Henry P, Krainin M, Herbst E, et al. RGB-D mapping: Using depth cameras for dense 3-D modeling of indoor environments[M]//KHATIB O, KUMAR V, PAPPAS G J. Experimental Robotics. Berlin Heidelberg: Springer, 2014: 647-663.

[39] Ji Zhang,Michael Kaess,Sanjiv Singh. A real-time method for depth enhanced visual

odometry[J]. Autonomous Robots,2017,41(1).

[40] Jinxia Zhang,Krista A. Ehinger,Haikun Wei,Kanjian Zhang,Jingyu Yang. A novel graph-based optimization framework for salient object detection[J]. Pattern Recognition,2017,64.

[41] Kümmerle R, Grisetti G, Strasdat H, et al. G2o: A general framework for graph optimization[C]// IEEE International Conference on Robotics and Automation. IEEE, 2011:3607-3613.

[42] Kun Zhou,Qiming Hou,Rui Wang,Baining Guo. Real-time KD-tree construction on graphics hardware[J]. ACM Transactions on Graphics (TOG),2008,27(5).

[43] LOWE D G. Distinctive Image Features from Scale-Invariant Keypoints[J]. International Journal of Computer Vision,2004,60(2):91-110.

[44] Martin A. Fischler,Robert C. Bolles. Random sample consensus[J]. Communications of the ACM,1981,24(6) : 381-395.

[45] Matthies L, Shafer S. Error modeling in stereo navigation[M]// Autonomous robot vehicles. Springer-Verlag New York, Inc. 1990:239-248.

[46] Muja M, David G. Lowe. Fast approximate nearest neighbors with automatic algorithm configuration[J]. Proc. VISAPP, 2009, 2009.

[47] Newcombe R A, Izadi S, Hilliges O, et al. KinectFusion: Real-time dense surface mapping and tracking[C]. Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on. IEEE, 2011: 127-136.

[48] Niclas Zeller,Franz Quint,Uwe Stilla. Depth estimation and camera calibration of a focused plenoptic camera for visual odometry[J]. ISPRS Journal of Photogrammetry and Remote Sensing,2016,118.

[49] P. J. Besl and N. D. McKay. A method for registration of 3-D shapes[J]. IEEE Transsactions on Pattern Analysis and Machine Intelligence,1992,14(2):239-256.

[50] Qian Sun,Ming Diao,Yibing Li,Ya Zhang. An improved binocular visual odometry algorithm based on the Random Sample Consensus in visual navigation systems[J]. Industrial Robot: An International Journal,2017,44(4).

[51] Richard Hartley, Andrew Zisserman. Multiple view geometry in computer vision[J]. Cambridge University Press. 2002, 37(1):85-86.

[52] Rosten E, Drummond T. Machine Learning for High-Speed Corner Detection[C]// European Conference on Computer Vision. Springer-Verlag, 2006:430-443.

[53] RUBLEE E, RABAUD V, KONOLIGE K,et al. ORB:An efficient alternative to SIFT or SURF[C]// IEEE International Conference on Computer Vision. New Jersey USA:IEEE,2012,58(11):2564-2571.

[54] Smith R.C, Cheeseman P. On the Representation and Estimation of Spatial Uncertainty. The International Journal of Robotics Research. 1986,5(4):56-68.

攻读学位期间发表的学术论文及参与项目

硕士期间发表的学术论文：

（1）刘家豪，刘志杰，刘嵩 基于 RGB-D 视觉里程计估计算法的研究[J]. 重庆科技学院学报（自然科学版）;2018,01:88-93

硕士期间参与的项目：

贵州省科技厅攻关项目：海龙屯申报世界文化遗产关键性技术研究的后期图像处理工作。

贵州科技馆大数据展区 3D 打印展品的设计与开发。

贵州师范大学学位论文原创性声明

本人郑重声明：所呈交的学位论文，是本人在导师指导下进行研究工作所取得的研究成果。除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人创作的、已公开发表或者没有公开发表的作品的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。本学位论文原创性声明的法律责任由本人承担。

学位论文作者签名：　　年　　月　　日

贵州师范大学学位论文使用授权书

本学位论文作者完全了解贵州师范大学关于保存、使用学位论文的管理办法及规定，即学校有权保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅和借阅，接受社会监督。贵州师范大学可以将本学位论文的全部或部分内容编入或允许第三方机构编入有关数据库进行检索和传播，可采用影印、缩印、数字化或其它复制手段保存或汇编本学位论文。

本学位论文属于：

□保密，在　　年解密后适用本授权（保密申请见附件）。

□不保密。

（请在以上相应方框内打“ √ ”）

学位论文作者签名：　　导师签名：

签字日期：　　年　月　日　　签字日期：　　年　月　日