分类号	
学校代码	10700
学号	2210321239

而易强工大学 硕士学位论文 (专业学位)

基于视觉与惯导的ROV组合导航方法研究

杨启航

专业类别:	电子信息硕士		
工程领域:	控制工程		
指导教师:	张晓晖 教授		
企业导师:	谢卫 高级工程师		
申请日期:	2024年6月		

论文题目: 基于视觉与惯导的 ROV 组合导航方法研究

学科名称:控制工程

研究生:杨启航

指导教师:张晓晖 教授



#### 摘要

有缆遥控式水下机器人(Remotely Operated Vehicle, ROV)的导航定位技术在海洋探测 工程中起着至关重要的作用,是 ROV 准确完成海底地形勘测、深海资源勘探、水下考古、 生物调查、水下管道监测等任务的基本前提。虽然 ROV 已经普遍应用到实际工程中,但 是目前水下导航定位方法大多成本较高或严重依赖外界信息,在鲁棒性、稳定性和精确性 等方面制约了海洋探测技术的发展。因此本文面向海洋探测工程中 ROV 更高性能水下导 航定位算法的需求,基于多状态约束卡尔曼滤波器(Multi-State Constraint Kalman Filter, MSCKF),将惯性测量单元(Inertial Measurement Unit, IMU)与视觉传感器相结合,借助图 像信息弥补 IMU 产生的累计误差,从惯性测量值与图像预处理、数据融合技术和位姿修 正三个角度进行了深入的理论分析,利用数据集和实物测试验证了导航定位算法的准确 性和鲁棒性。本文具体研究内容如下:

(1) 详细阐述了视觉/惯导融合定位的研究现状和发展历程,强调了本文的研究意义。 继而介绍了系统传感器模型,对惯导定位的解算过程进行介绍,对误差来源进行分析与建模,并利用水下图像增强算法对视觉信息进行优化进而提取图像特征。

(2) 对基于滤波的视觉与惯导融合算法 MSCKF 进行研究,在图像处理部分进行特征 点检测、匹配跟踪与误匹配剔除,采用紧耦合的方案将图像特征与 IMU 数据进行融合, 详尽推导了基于滤波的视觉与惯导融合算法的理论和公式,选取了系统状态向量并基于 系统的运动模型和观测模型进行滤波器的设计,深入探讨了包含系统状态预测、状态增广 以及滤波器观测更新在内的位姿解算方案,最终得到机体的位置与姿态信息。

(3) 针对系统运动估计存在的累积误差,根据两种不同情况进行纠正:在 ROV 经过 历史位置的情况下,进行水下回环检测与校正;在 ROV 不经过历史位置的情况下,则利 用已知标志点进行外环位姿修正。通过词袋模型建立了水下视觉字典进行相似度计算来 评估两幅图像是否构成回环,给出了回环候选帧筛选策略并采用几何约束对改进的水下 回环识别进行研究,最后阐述了标志点检测的过程,对机体位姿进行修正。

(4) 通过 KITTI 数据集仿真,与滑动窗口滤波算法进行了对比分析,仿真结果表明了 本文研究的定位算法具有较高的准确性和实时性,且能够较好地检测到视觉回环。利用已 知标志点进行位姿修正,仿真误差减小了 23.07%,有效降低了位姿估计的累积误差。基

T

于 BlueROV 开源软硬件搭建了实验平台,通过实验方案设计分别在室内水箱和室外水池 开展水下实验,证明本文方法的有效性。最终结果表明,本文算法能有效地提升系统位姿 估计的精度。

关键词:水下导航定位;视觉惯导融合;多状态约束卡尔曼滤波器;位姿修正

Abstract

# Title: RESEARCH ON ROV COMBINED NAVIGATION METHOD BASED ON VISION AND INERTIAL NAVIGATION

Abstract

Major: Control Engineering

Name: Qihang YANG

Sig	natu	ire:	
		· · · ·	

Signature:

Supervisor: Prof. Xiaohui ZHANG

The navigation and positioning technology of Remotely Operated Vehicle (ROV) plays a vital role in ocean exploration engineering. It is the basic premise for accurate submarine terrain survey, deep sea resource exploration, underwater archaeology, biological investigation, and underwater pipeline monitoring. Although ROV has been widely used in practical engineering, most of the current underwater navigation and positioning methods have high cost or rely heavily on external information, which restricts the development of ocean exploration technology in terms of robustness, stability and accuracy. Therefore, this paper is based on Multi-State Constraint Kalman Filter (MSCKF) to meet the requirements of ROV underwater navigation and positioning algorithms with higher performance in ocean exploration engineering. Combining Inertial Measurement Unit (IMU) with vision sensor, this paper uses image information to compensate for accumulated errors caused by IMU. Theoretical analysis is conducted from three aspects: inertial measurement value and image preprocessing, data fusion technology, and pose correction. The accuracy and robustness of the navigation and positioning algorithm are verified by data sets and physical tests. The specific research content of this paper is as follows:

(1) The research status and development course of vision/inertial navigation fusion positioning are elaborated, and the research significance of this paper is emphasized. Then the sensor model of the system is introduced, the solution process of inertial navigation positioning is introduced, error sources are analyzed and modeled, and the underwater image enhancement algorithm is used to optimize visual information and extract image features.

(2) Study the filter-based vision and inertial navigation fusion algorithm MSCKF, carry out feature point detection, matching tracking and mismatching elimination in the image processing part, adopt a tight coupling scheme to fuse image features with IMU data, and derive the theory and formula of filter-based vision and inertial navigation fusion algorithm in detail. The filter is

designed based on the motion model and observation model of the system, and the pose solution including system state prediction, state augmentation and filter observation update is deeply discussed. Finally, the position and attitude information of the body are obtained.

(3) The accumulated errors in the system motion estimation are corrected according to two different situations: underwater loop detection and correction when the ROV passes through the historical position; When the ROV does not pass the historical position, the external loop position correction is performed using the known marker points. An underwater visual dictionary was established by the bag of words model to evaluate whether the two images constituted a loop by similarity calculation, and a loop candidate frame screening strategy was given and the improved underwater loop recognition was studied by geometric constraints. Finally, the process of mark point detection was described to correct the body pose.

(4) Through the simulation of KITTI data set, the comparison and analysis with the sliding window filter (SWF) algorithm are carried out. The simulation results show that the positioning algorithm studied in this paper has high accuracy and real-time performance, and can detect the visual loop well. The simulation error is reduced by 23.07%, and the cumulative error of pose estimation is reduced effectively. An experimental platform was built based on BlueROV open source software and hardware, and underwater experiments were carried out in indoor water tank and outdoor pool respectively through experimental scheme design to prove the effectively improve the above methods. The final results show that the proposed algorithm can effectively improve the accuracy of pose estimation.

**Key words:** Underwater navigation and positioning; Visual inertial navigation fusion; Multi-state constraint kalman filter; Pose correction

目 录

1 绪论	1
1.1 研究背景与意义	1
1.2 国内外研究现状	2
1.2.1 ROV 及其导航定位系统发展现状	2
1.2.2 水下导航定位技术	5
1.2.3 视觉与惯导信息融合研究现状	7
1.3 研究内容及主要工作	8
2 ROV 导航定位系统的传感器模型建立与误差分析	11
2.1 ROV 坐标系与运动学模型的建立	11
2.2 惯性导航系统原理	14
2.2.1 SINS 姿态解算	14
2.2.2 SINS 位置和速度解算	16
2.3 惯性导航误差分析	17
2.3.1 惯性导航定位仿真	17
2.3.2 IMU 测量误差分析	18
2.4 相机成像与图像预处理	20
2.4.1 相机成像模型	20
2.4.2 相机标定理论	22
2.4.3 水下图像增强	24
2.5 本章小结	26
3 视觉与惯导的信息融合算法研究	27
3.1 视觉图像处理	27
3.1.1 FAST 特征点检测	27
3.1.2 基于图像金字塔的 LK 特征点跟踪	28
3.1.3 特征点误匹配剔除	31
3.2 滤波器设计	32
3.2.1 扩展卡尔曼滤波器	32
3.2.2 系统状态向量	34
3.3 IMU 状态预测	35
3.3.1 状态预测模型	35
3.3.2 状态增广	37
3.4 相机状态更新	39
3.4.1 状态观测模型	39
3.4.2 滤波器更新	40

3.5 组合导航定位算法仿真分析	41
3.6 本章小结	46
4 组合导航定位系统的位姿修正方法研究	47
4.1 回环检测	47
4.1.1 回环检测简介	47
4.1.2 水下视觉字典	48
4.1.3 相似度计算	49
4.2 水下回环识别与改进研究	50
4.2.1 回环候选帧筛选	50
4.2.2 几何约束及改进	50
4.3 基于视觉回环的位姿修正	52
4.4 基于已知标志点的位姿修正	53
4.4.1 标志点检测	53
4.4.2 机体位姿修正	55
4.5 本章小结	57
5 水下试验与结果分析	59
5.1 引言	59
5.2 实验平台搭建	59
5.2.1 实验平台	59
5.2.2 传感器数据获取	60
5.2.3 相机-IMU标定	62
5.3 水下定位试验	63
5.3.1 小范围静水试验	63
5.3.2 室外水池试验	65
5.4 本章小结	69
6 总结与展望	71
6.1 论文总结	71
6.2 未来展望	72
参考文献	73

# 1 绪论

### 1.1 研究背景与意义

随着能源需求的不断增长和全球气候变化的挑战,陆域经济发展面临瓶颈,人类社会逐渐步入泛海洋时代,海洋的战略地位日益凸显<sup>[1]</sup>。地球上海洋总面积约为3.6亿平方千米,约占地球表面积的71%。丰富的可再生资源、生物资源以及矿产资源蕴藏在这片广阔的领域,这些自然资源不仅数量巨大,而且分布广泛。从《中国海洋经济统计公报》(2013—2022年)可以看出,我国的海洋经济生产总值约占国内生产总值(GDP)的9%,仍有发展的潜力和空间<sup>[2]</sup>。海洋环境所蕴藏的丰富资源以及其在全球能源安全、气候变化等重大问题中的战略地位,使得海洋成为各国竞相研究和开发的重点领域。

党中央适时地做出了建设海洋强国的战略部署,相继提出了"提高海洋资源开发能力, 发展海洋经济,保护海洋生态环境,坚决维护国家海洋权益,建设海洋强国"、"坚持陆海 统筹,加快建设海洋强国"、"发展海洋经济,保护海洋生态环境,加快建设海洋强国", 为新时代海洋事业发展、海洋强国建设提供了行动指南<sup>[3]</sup>。近几年发布的海洋战略和政策 文件,都旨在促进海洋经济的高质量发展,加强海洋生态保护,推动海洋科技创新,以及 保障国家海洋利益安全。

在广阔的海洋中,由于水下环境的特殊性质,如低可见度、高压强等,使得人类无法 单靠自身力量完成水下的科学考察和资源勘探<sup>[4-6]</sup>。然而我国海洋产业的核心技术受制于 人,海洋关键技术装备自给率相对较低,自主创新能力有待提高,这在一定程度上制约了 我国海洋产业的高速发展<sup>[7]</sup>。水下机器人(Unmanned Underwater Vehicle, UUV)是深海探索 的关键装备之一,可以代替人类进行水下观测、巡视与作业等任务。UUV 可以简单分为 无缆自主式水下机器人(Autonomous Underwater Vehicle, AUV)和有缆遥控式水下机器人 (Remotely Operated Vehicle, ROV)。ROV 与水面母船或岸基系统通过脐带电缆或细缆进行 信息交互,能够在水下数百米处长时间工作,同时具有易于回收、结构可靠等优势,因此 目前 ROV 应用范围非常广泛<sup>[8]</sup>。

在 ROV 进行水下作业时,导航定位技术是确保水下机器人能够可靠、准确地执行水 下任务的关键信息保障和技术前提。在获取水下机器人自身位置或坐标的基础之上,可进 一步考虑地图构建、路径规划、避障等其他导航任务,海底管道检测等更高层次的应用级 任务更依赖于 ROV 准确的定位。由于水下环境的复杂性和不确定性,导航定位技术面临 着诸多挑战,成为了水下机器人研究和开发中的热点和难点问题之一。在陆地上表现良好 的 GPS 信号进入水下环境后会受到严重的衰减和干扰,无法使用其获取准确的定位。长 基线、短基线、超短基线等传统的水声定位系统需要提前布设水声基元或者需要母船支持, 价格昂贵且安装和维护较为困难,不适合进行紧急水下任务,这也限制了其在水下机器人 定位领域的应用范围和普及程度<sup>[9]</sup>。声呐也存在分辨率低、成本相对高的缺点。因此,研 究水下机器人导航定位技术具有非常重要的工程价值和意义。

惯性导航是一种无需依赖外部信息源的自主导航技术,利用惯性测量单元(Inertial Measurement Unit, IMU)来实现机体的导航定位。IMU 内部集成了加速度计和陀螺仪等传感器,能够实时测量并记录机体的运动参数,解算得到机体的姿态、速度和位置信息。惯性导航具有自主性和抗干扰性等优点,但也存在一些局限性,由于定位误差会随着时间的推移而逐渐积累,无法进行长期和高精度的导航定位。因此在实际应用中,惯性导航通常需要与其他导航方式结合使用,以实现更加精确和可靠的导航定位。

视觉导航通过摄像机拍摄周围环境的图像序列来感知外界环境变化并推算位姿变化。 由于摄像机拍摄的图像信息丰富,使得视觉导航备受关注。若只依靠视觉传感器进行定位, 定位结果的准确性可能会因为多种因素受到影响,在环境纹理较少、机器人快速移动、光 照剧烈变化等情况下,会导致无法从连续拍摄的照片提取足够的特征,从而影响定位精度。

随着技术的发展和广泛的应用,单一的导航系统已经无法满足多种复杂环境下的导航需求。为了提高定位的准确性和可靠性,通常会采用多种互补的传感器来获得多源信息。 IMU 具有较高的采样频率且不依赖于外部环境信息,即使在高速运动或环境条件不佳的 情况下,它也能提供相对稳定的定位数据。视觉传感器可以提供丰富的环境信息,将 IMU 与视觉传感器相结合,可以实现属性互补,获得更加准确和可靠的位置和姿态信息,提高 导航系统的鲁棒性和适应性。基于滤波的视觉/惯导信息融合算法具有计算量小、实时性 高的优点,能够在陆地和空中提供高精度的定位,然而水下各种复杂因素给这些算法带来 挑战。文献[10]将多状态约束卡尔曼滤波器(Multi-State Constraint Kalman Filter, MSCKF)的 视觉/惯导信息融合算法首次用于水下,能够较好地应用于水下环境。但是 MSCKF 在水 下的应用过程中存在逐渐累积的误差,若未对位姿估计的误差及时消除,可能会导致定位 结果发散的情况。因此研究如何消除定位过程中的累积误差具有实际的意义。

视觉/惯导组合导航定位技术能够在多种环境条件下实现高精度的定位效果,而且相 机和 IMU 等设备的成本相对较低,使得该技术具有很大的应用潜力。水下环境复杂多变, 对导航系统的要求更高,因此这种低成本、高精度的导航技术受到了水下机器人研究者的 广泛关注。研究基于视觉与惯导的组合导航定位技术并对水下定位过程中的累积误差消 除具有重大的现实意义和应用前景,随着技术的不断发展和完善,该技术有望在未来得到 更为广泛的推广和应用。

# 1.2 国内外研究现状

#### 1.2.1 ROV 及其导航定位系统发展现状

为了合理开发和利用海洋的空间和资源,ROV 利用先进的传感器、导航系统和作业 工具,可以在人类难以承受的水下环境中长时间工作,并收集大量有价值的数据和信息。 这些信息不仅有助于科学家们更深入地了解海洋生态系统和海底地质结构,还为资源勘 探和水下考古等领域提供了重要的技术支持。从研究内容来看,ROV 有水下感知技术、 导航定位技术、运动控制技术和仿真技术等热点问题。其中现有的水下导航技术还存在很 多缺点,精确、高效、可靠的导航定位技术是 ROV 主要研究方向之一。国内外对 ROV 的 导航定位问题展开了大量研究并进行了相关实际工程应用。

国外研发 ROV 的时间相对较早,被普遍认可的世界第一台 ROV 是 1953 年由美国研制成功的 "Poodle",开启了 ROV 的研究篇章,是 ROV 发展史上的重要开端。为了满足军事侦察和远洋考察的需要,一些发达国家纷纷紧随其后开启了对 ROV 的研究。1960 年美国海军研发出了世界上第一台实用化的 ROV,并将其命名为 "CURV-I",如图 1-1 所示。"CURV-I"于 1966 年在西班牙海域 868 米深的海底处成功打捞了一枚氢弹,这次行动让其一举成名,成为了世界范围内无人水下技术的典范<sup>[11]</sup>。



图 1-1 世界上最早的实用型 ROV-CURV Fig.1-1 The World's First Practical ROV-CURV

约翰霍普金斯大学的 Giancarlo Troni 等人在 JHU-ROV 上搭载了深度计、陀螺仪、加速度计和多普勒测速仪(Doppler Velocity Log, DVL)进行了组合任务,并完成了低成本航姿参考系统的实验评估<sup>[12]</sup>, JHU-ROV 实物图如图 1-2 所示。



图 1-2 JHU-ROV 实物图 Fig.1-2 JHU-ROV Physical Picture

澳大利亚海事大学的 Khoa Duy Le 等人设计了一种航向观测器,利用惯性测量单元 和磁力计的数据,通过双卡尔曼滤波消除和补偿外部加速度和滚转/俯仰扰动,并在 AMC ROV 验证了该导航系统的可行性<sup>[13]</sup>。近些年来,更多轻量化设计的 ROV 受到从业者的 青睐。2012 年,Open ROV 公司开发的 ROV 整机质量只有 2.5kg,并且硬件和软件都是

开源的。以其创新的开源设计、轻量化的机身和强大的功能,进一步推动了 ROV 技术的 进步。

在美国、英国、法国、俄罗斯和日本等地,根据其独特的海洋环境和技术优势,这些 海洋强国都将不同型号的 ROV 广泛应用于海洋科学研究、资源开发、海洋安全、环境保 护等多个领域。与国外 ROV 的进展相比,我国在这一领域的研究启动相对较晚<sup>[14,15]</sup>。然 而,近些年随着海洋强国战略的提出和对海洋资源的持续关注,我国接连推出了多项规划 和指导方针,为 ROV 的进一步发展和实际应用提供了有益的建议。

上世纪 80 年代立足于军事需求,我国开始了 ROV 的研制与开发,与国外相比起步 较晚,在取得一定成果后开始向商业用途拓展。国内深海 ROV 以中国科学院沈阳自动化 研究所和上海交通大学两家单位研制为主。经过多年的探索,1985 年沈阳自动化所成功 研发了我国的首台 ROV "海人一号",如图 1-3 所示。该 ROV 能够在水下完成简单的作 业任务。



图 1-3 海人一号 ROV Fig.1-3 HR-01 ROV

在十一五期间的国家高技术研究发展计划项目支持下,上海交通大学研制的"海马" 号 ROV 于 2014 年完成了 4500m 海上试验,是国内首套 4500m 级 ROV,如图 1-4 所示。 "海马"号的导航系统装配了声呐设备、高度计、深度计以及罗经,实现了装备系统的国 产化,具有重要的战略意义<sup>[16]</sup>。



图 1-4 "海马"号 4500 米级 ROV Fig.1-4 "Seahorse" 4500 Meters ROV

江苏科技大学曾庆军等人设计了一套包括深度传感器、磁罗盘、加速度计和陀螺仪的 微惯性导航系统<sup>[17]</sup>,并将其应用在 MC-ROV 上进行水下实验,实验结果表明姿态角中横 滚角和俯仰角的误差约为 0.5°、航向角误差约为 1°。国内从事水下机器人研究的公司较 少,天津深之蓝公司生产的各系列 ROV 已在"南水北调"项目中,成功进行了水下管道 检测等工作。

随着科技的飞速发展和海洋工程需求的不断增长,各种水下导航定位技术广泛应用 于 ROV,ROV 产业已经具备相当的规模,全球范围内有数百种不同型号的 ROV 在各种 应用场景中发挥着重要作用。为了满足广泛应用的需求,ROV 不仅需要拥有长时间和长 距离的导航能力,还必须能够提供姿态和速度等相关信息。高精度导航系统被认为是 ROV 核心技术的一部分,面临着水下环境复杂多变以及对其高可靠性、高稳定性需求的双重挑 战,因此实现精准导航成为了一项极具挑战性的任务。

#### 1.2.2 水下导航定位技术

无线电信号在水中迅速衰减,导致陆地和空中使用范围最为广泛的高精度全球导航 卫星系统<sup>[18]</sup>(Global Navigation Satellite System, GNSS)不适用于水下环境。特殊的水下环境 给水下导航带来挑战,目前主要的水下导航定位技术有:航位推算导航、惯性导航、地球 物理导航、视觉导航、水声定位导航以及多航行器协作导航<sup>[19-21]</sup>。

(1) 航位推算导航

航位推算导航最早于 16 世纪提出, Cotter 将推位导航定义为"从一个已知的坐标位 置开始,推算下一时刻的坐标位置的导航过程"<sup>[22]</sup>。机体一般配备速度、航向及深度传感 器等,UUV 的推位导航系统常用罗经实时测量航向信息,并利用 DVL 测量机体坐标系下 的速度信息,进而对速度进行航向分解,在固定时间间隔内对速度进行一次积分得到 UUV 的实时平面坐标信息。刘沛佳等人提出一种低成本而有较高精度的 DVL 自主导航方法, 应用在中小型自主水下机器人"海为一号"上表现出较强的实用性<sup>[23]</sup>。

(2) 惯性导航

惯性导航系统(Inertial Navigation System, INS)根据惯性测量元件安装方法可以分为平 台式和捷联式。INS 利用 IMU 测量机体自身的加速度和角速度,通过积分运算得到位置 与姿态信息,短时间内可以达到高精度的导航定位,本质上也属于航位推算。IMU 的测 量结果含有随机游走噪声和零偏,一般假设随机游走的噪声服从正态分布,另外通过标定 可以对零偏进行补偿。惯导可以在不与外界通信的条件下连续提供位置、速度、姿态、航 向信息,具有完备性、隐蔽性和自主性的特点。但惯性导航的定位误差随时间二次方增加, 惯性单元的精度越高,其成本肯定也越高,昂贵的 IMU 一般可以持续使用更长时间,而 价格低廉的 IMU 在使用一段时间后会出现显著的数据偏移现象。因此惯性导航不仅需要 绝对初始位置,且误差累计还会迅速增大。在水下航行一段时间后需要消除累积误差,因 为电磁波在水下传播衰减严重,若要用水面之上的天文导航、卫星导航、无线电导航等方 法修正误差,UUV 就必须浮出水面,不利于 UUV 的水下隐蔽性作业,也会造成动力损

5

失。因此应用于 UUV 的惯性导航多与其他水下导航设备组合使用<sup>[24]</sup>。

(3) 地球物理导航

海洋地球物理导航作为地球固有物理属性的导航技术,具有无时间累积误差、隐蔽性强、自主无源等优点。常用有重力匹配导航<sup>[25]</sup>、地磁匹配导航和地形匹配导航<sup>[26]</sup>,通常作为惯性辅助导航技术。地球物理导航需要事先根据海洋地球物理特征分布特点,将UUV 航行附近的数据信息特征进行存储,UUV 通过搭载的传感器收集航行轨迹周围的物理信息与计算机存储的信息进行匹配,可以实时获取位置信息并修正惯导的累计误差,实现水下导航与定位。

(4) 水声定位

水声定位系统由应答器和声波发射器组成,它通过测定声波的传播相位差和时间差 来对目标进行精确定位。根据接收基阵的长度将水声定位系统分为3类。超短基线(Ultra Short Base Line, USBL)定位系统<sup>[27]</sup>的基点置于船底,利用声信号相位差确定目标方向、声 信号传播时间测算距离。系统集成度高,操作便捷,适用于小范围精准定位。然而目标的 绝对位置测量精度与其外部设备的精度密切相关,而且系统安装完成后需要进行准确的 校准<sup>[28]</sup>。长基线(Lone Base Line, LBL)具有较高的定位精度,但其定位范围相对有限,通 常被用于高精度的局部定位。短基线(Short Base Line, SBL)系统基点间距超10米,精度介 于 USBL 与 LBL 之间,但安装后需进行严格校准以确保准确性。自上世纪50年代起,国 外对水声定位技术的研究已逾60载,技术日趋成熟,多家公司如英国 Sonardyne 己推出 高性能的 LBL 产品如 Compatt6 和 Fusion6G 系列<sup>[29]</sup>。随着我国的长期研发努力,与发达 国家的水声定位技术差距已显著缩小。现今,如江苏中海达海洋信息技术有限公司等国内 企业也推出了如 iTrack 系列长基线等水声定位产品。但由于基站覆盖范围有限,水声定 位系统需要布设辅助基站在水底或水面船只上,这限制了其在军事行动和远航科研任务 中的应用。

(5) SLAM 定位

相比于水声定位复杂的布阵操作,同步定位与地图创建<sup>[30,31]</sup>(Simultaneous Localization and Mapping, SLAM)利用机体搭载的相机和声呐<sup>[32]</sup>等传感器实现定位并构建周围环境的 地图。在浑浊水下环境中,声呐能发挥作用,但高昂的价格和有限的分辨率限制了其广泛 应用;而在清澈水域中,摄像头则能提供更为详尽丰富的视觉信息。随着技术的迅猛进步, 相机以其高性价比、高分辨率和低延迟的特点,为实时处理图像数据提供了强大支撑。在 视觉定位技术的后端核心,视觉里程计(Visual Odometer, VO)扮演着举足轻重的角色。VO 系统使用单目、双目和 RGB 相机捕获的图像流来估算移动物体的位置和姿态。经过数十 年的深入探索,国内外对于 VO 的研究已经积累了深厚的理论基础,并成功构建了一套完 整且成熟的系统解决方案。VO 技术的优势不仅体现在其相对较低的成本上,而且由于相 机的存在,即使在没有 GPS 信号的情况下也能准确估计位姿。此外,与 IMU 等设备相比, VO 没有明显的漂移问题,这也使得 VO 技术的未来发展受到了广泛的关注。近年来,随 着集成闭环技术的 VO 系统的涌现, SLAM 与 VO 之间的分界线变得越来越不明确。SLAM 系统的核心目标是利用移动设备上的传感器来创建地图,并实时地计算机体在该地图上 的位置和姿态,而 VO 系统则更多地关注于计算机体自身的运动特性,而非地图的构建。

(6) 多航行器协同导航

随着大范围的水下目标探测、全方位的海洋环境调查等海洋研究、海洋开发的不断深入与复杂化,单 UUV 往往难以完成复杂度更高的任务,为此多 UUV 协同导航定位系统 应运而生。多 UUV 协同导航多用于 AUV,多 AUV 协作的一个基本条件就是要全部或者 部分获取每个 AUV 平台的位置、速度、姿态等状态信息。根据领航者数量的差异,多 AUV 协同导航定位系统通常被分类为多领航者和单领航者系统。通常而言,采用多个领航者进 行协同导航定位,能够显著增强低精度 AUV 在导航和定位方面的精确性<sup>[33,34]</sup>,协同导航 定位系统的优势往往是通过协同导航定位算法实现的。

随着导航任务复杂性的不断提升,单一的导航技术已无法满足需求,因此 UUV 导航 技术的研究焦点逐渐转向了组合导航技术。在 UUV 水下组合导航中,通常采用以 INS 为 核心,辅助以声学导航、地球物理导航及视觉导航等技术中的一种或多种。这一综合应用 策略旨在通过高精度的导航技术,有效纠正 INS 的累积误差,同时确保 UUV 保持高度的 隐蔽性和自主性。

#### 1.2.3 视觉与惯导信息融合研究现状

在众多以 INS 为主的水下组合导航定位技术中,视觉与惯导的组合方式以不需要提前创建地图、不需要高昂的传感器等优势脱颖而出,近些年国内外学者对此热点开展了广泛研究。在惯性与视觉信息融合的定位技术研究中,根据不同的后端处理方法可划分为两大主要类别:一类是采用基于滤波的策略对这两种数据进行整合,另一类则是采用优化技术来整合 IMU 和视觉传感器的数据。在基于滤波的方法中,均采用卡尔曼滤波对 IMU 测量值进行状态向量预测,并利用相机的图像信息对状态向量更新。系统状态是通过递推得到的,当前时刻的状态量只与前一时刻有关。同时前一时刻的误差也会积累到当前时刻,但优点是系统的计算量较小。基于优化的后端处理策略是保存之前所有时刻的状态,并构建一个包含惯性测量参数的积分误差和重投影误差的联合函数作为约束条件,然后计算出状态的最优解。这种方法的缺点是在非线性最优化求解过程中计算量巨大,实时性较差。

基于滤波的视觉/惯导组合导航系统数据融合算法最常用的是卡尔曼滤波(Kalman Filter, KF)及其扩展算法。常规的 EKF-SLAM 算法将特征点直接加入到状态向量中,在一些简单的小场景中效果比较好,面对复杂场景时提取的环境中特征点巨幅增加,状态向量的维数也会急速增加。针对 EKF-SLAM 维数太多而导致计算量极大的问题, Mourikis 等人提出了 MSCKF 算法<sup>[35]</sup>,利用特征点间的关系对连续相邻的图像帧形成约束,在滑动窗口内将相机的姿态加入状态向量中,旧相机状态会随着计算的进程逐渐删除,状态向量的维度大大减小,计算效率有效提高。2013 年,李明阳等人<sup>[36]</sup>在 MSCKF 方法的基础上进行了进一步的改进,并首次提出了通过估计雅克比来解决滤波器估计不一致性问题的新

策略。MSCKF 在水下环境表现出较好的效果,另外还有双目版本的 S-MSCKF 算法<sup>[37]</sup>。 2020年,黄国权团队<sup>[38]</sup>开源了一种基于 MSCKF 框架的 Open VINS 算法,该算法不仅支 持在线的相机内外参数标定,还融入了 SLAM 的局部地图功能,从而显著提升了定位精 度。除了该算法以外,SSF 算法基于 EKF 的时延补偿框架对多个传感器进行数据融合<sup>[39]</sup>。 MSF-EKF 算法将噪声和扰动引起的误差加入到误差状态向量来建立误差运动模型,理论 上可以处理无限数量不同类型传感器具有延时的相对或绝对测量值<sup>[40]</sup>。ROVIO(Robust Visual Inertial Odometry)<sup>[41]</sup>算法将地标位置估计划分为方位向量和距离,以直接法进行前 端处理使系统初始化没有延迟,并通过 IEKF<sup>[42]</sup>来将数据融合在一起。该算法同时兼顾了 精度和计算效率,对于纹理不够清晰的场景和在运动剧烈时表现出了较高的鲁棒性。

随着计算机资源与性能的提升,基于优化的方案也迎来快速发展。VINS-MONO<sup>[43,44]</sup> 算法将单目摄像机与 IMU 数据进行融合,通过对 IMU 预积分值与相机的观测值进行非线 性优化,提升了输出状态估计值的准确度。该算法可以在线标定参数,并且包含回环检测 和校正功能,还可以比较高效的保存和重用地图,因此是一个可靠完整的通用系统;另外 可以支持多种传感器的融合,兼容性较强。ICE-BA<sup>[45,46]</sup>算法是一种具有增量性、一致性 且有效的捆集调整算法,其在滑动窗口算法的基础上进行了局部的 BA(Bundle Adjustment)<sup>[47]</sup>优化,并在所有关键帧上进行并行全局的 BA 优化,能够实时输出相机的位 姿和局部更新的地图点。Maplab<sup>[48,49]</sup>算法提供了包括地图合并、视觉/惯导批量处理优化、 环路闭合、三维重建等研究工具,是一个开放可行的视觉/惯导 SLAM 框架。

另外根据数据融合的角度可以进一步划分为松耦合<sup>[50,51]</sup>和紧耦合两种。松耦合系统通 过视觉与 IMU 分别对机体的位姿进行独立的估计,然后将两个位姿结果利用算法进行一 定程度的耦合,最后得到优化后的位姿信息<sup>[52,53]</sup>。这种方式未将两个传感器数据集中到同 一状态向量内,系统关系简单且较为独立,耦合度低。若其中一个传感器出现故障,其最 终还是能得到一个位姿估计,因此可以拓展其他传感器进行信息融合,但另一方面也导致 了松耦合对位姿估计结果的精度不够高。在紧耦合的方法中,直接将视觉信息和 IMU 测 量数据进行融合来估计相机的位置和姿态,联合优化得到的位姿估计结果精度更高,同时 增加了数据处理计算的成本。

#### 1.3 研究内容及主要工作

基于视觉与惯导组合的 ROV 水下导航定位系统总体框架如图 1-5 所示。系统前端有 IMU 传感器和相机传感器。其中 IMU 传感器用于获取机体的位置和姿态信息,而相机传 感器可以检测到图像中的特征点,并利用这些特征点来估计机体的运动信息。为了将这两 种信息进行融合,使用了基于 MSCKF 算法框架的位姿解算方案估计机体的位置和姿态信 息。并对长时间运动造成的累计误差进行位姿修正,最后生成轨迹图。



1 绪论

图 1-5 系统整体框图 Fig.1-5 Overall Block Diagram of the System

论文主要内容安排如下:

第一章是绪论部分,首先明确了本研究工作的背景和重要性,以ROV水下作业需求为切入点,深入探讨了导航定位在ROV系统中的核心作用,以及当前水下导航定位系统的发展态势。同时详细阐述了视觉与惯导组合导航技术的显著优势及其在实际应用中的必要性。接下来概述了与视觉/惯导融合定位有关的当前研究状况和历史发展,探讨了目前 MSCKF 的信息融合方法及其存在的缺陷,引出了进行位姿修正提高定位算法精度的意义。最终,对本文的各个章节内容进行了简洁的概述。

第二章首先建立了系统坐标系和 ROV 运动学模型,对惯导定位的相关理论进行介绍 并对误差来源进行分析与建模,从而引出与视觉融合的必要性。接下来对相机成像模型进 行了详细介绍,由于水下的散射、吸收和光线的折射等因素导致的图像质量下降,所以在 图像特征处理之前需进行水下图像增强。

第三章,视觉与惯导的信息融合算法研究。本章主要对 MSCKF 算法进行研究,采用 紧耦合的方案将图像特征与 IMU 数据进行融合,详尽推导了基于滤波的视觉与惯导融合 算法的理论和公式,根据位姿解算方案最终得到机体位置与姿态,并通过仿真与 SWF 算 法进行了对比分析。

第四章,组合导航定位系统的位姿修正方法研究。针对长时间运动产生的累计误差进行位姿修正。在 ROV 不经过历史位置的情况下,利用已知标志点进行外环位姿修正,对于 ROV 经过历史位置的情况,则进行水下回环检测与校正。最后通过数据集验证了修正 方法的可行性。

第五章主要聚焦于实验平台的搭建及水下实验的开展。首先详细介绍了 ROV 实验平台的组成,设计了水下的试验方案。在实验平台进行了多次水下实验,并对实验数据进行了细致的分析,以期得出准确可靠的结论。

第六章,总结展望。总结了实现 ROV 水下组合导航定位系统过程中所做的工作,展望了本文方法后续改进的方向。

# 2 ROV导航定位系统的传感器模型建立与误差分析

为了进一步探索满足 ROV 实际需求并且不依赖外界信息的高精度水下导航定位方法, 本文主要研究视觉与惯导的导航定位技术,仅依赖单一的导航方法很难达到导航系统在 准确性、实时响应和可靠性等多个方面的标准,结合视觉导航和惯性导航构成组合导航定 位系统能够属性互补,成为实现 ROV 高可靠性和高性能导航的一条有效途径。ROV 视觉 /惯导组合导航定位系统前端不仅涵盖了视觉和 IMU 两种传感器数据的采集,还包括了对 获取的水下图像信息和惯导测量数据进行预处理的过程。结合 ROV 水下导航定位的工作 需求,本章根据两种导航技术的原理及特点,首先对两个传感器的工作机制和误差来源进 行了深入分析,并且建立了相应的误差模型,这为接下来的组合导航定位系统中传感器数 据融合奠定了基础。

#### 2.1 ROV 坐标系与运动学模型的建立

为了更直观地描述 ROV 在水下运动时的各种状态信息,如位置、姿态、加速度和速度等,首先需要建立 ROV 运动的参考坐标系统。由于仅依赖单一坐标系来全面描述 ROV 的平移、旋转及翻滚等复杂运动动作颇具挑战,因此构建了如图 2-1 所示的坐标系统。该系统由定坐标系 {*G*}和动坐标系 {*O*}共同协作,以提供更为精确和全面的参数来描述 ROV 的运动状态,两个坐标系都遵循右手定则。



图 2-1 ROV 坐标系示意图 Fig.2-1 ROV Coordinate System Diagram

定坐标系 *G*-*ξηζ* 也称大地坐标系,一般将原点固定在母船或岸上的观察者位置处。 *Gξ* 轴指向正北,*Gη* 轴在同一平面指向正东,*Gζ* 轴正方向为地心方向。

动坐标系 *O*-*xyz*,又称机体坐标系,是直接建立在 ROV 本体上的,它与 ROV 的运动保持同步。该坐标系的原点与 ROV 的重心相重合,其中,*Ox* 轴指向 ROV 的艏部(即

前进方向), Oy 轴指向 ROV 的右舷(即右侧方向), 而 Oz 轴则垂直向下, 指向 ROV 的底部。

ROV 在水中所做的六自由度刚体运动分别为按 xyz 三个轴向进行平移和旋转,将这 六种运动方式分别定义为前进、横摆、浮沉、滚转、俯仰和偏航。ROV 运动的姿态参数 用机体坐标系相对于定系的姿态角 ( $\phi, \theta, \psi$ )表示,位置参数用机体坐标系原点 O 在定系上 的坐标值 (x, y, z)表示。滚转角 $\phi$ 、俯仰角 $\theta$ 、偏航角 $\psi$ 分别表示 ROV 绕 Ox 轴、Oy 轴、 Oz 轴转过的角度, $\phi$ 向右倾为正方向, $\theta$ 向尾倾为正方向, $\psi$ 向右转为正方向。ROV 机 体中心相对于大地坐标系以线速度 $v_1$ 移动、以角速度 $v_2$ 转动。在机体坐标系中,线速度的 投影可以分为纵向、横向以及垂向移动速度,表示为(u, v, w);角速度的投影可以分为横 倾、纵倾以及偏航角速度,可以将其表示为(p, q, r)。通过以上定义得到 ROV 运动参数的 向量表达式,其中线速度向量表示为 $v_1 = [u \ v \ w]^T$ ,角速度向量表示为 $v_2 = [p \ q \ r]^T$ ; 位置向量表示为 $\eta_1 = [x \ y \ z]^T$ ,姿态向量表示为 $\eta_2 = [\phi \ \theta \ \psi]^T$ 。表 2-1 列出了本文所 使用的符号及其定义。

自由度	运动方式	线速度和角速度	位置和方向
1	前进	U	x
2	横摆	v	У
3	浮沉	W	Z
4	滚转	р	$\phi$
5	俯仰	q	heta
6	偏航	r	$\psi$

表 2-1 ROV 物理符号说明表 Tab.2-1 ROV Physical Symbol Description Table

接下来建立动坐标系和定坐标系之间的转换关系,以便更准确地描述 ROV 的运动状态。当机体坐标系的原点O与大地坐标系的原点G重合时,经过三次旋转,大地坐标系就能与机体坐标系完全对齐,三次旋转示意图如图 2-2 所示。



Tig.2-2 Three Rotations from the Earth Traine to the Body Traine

第一次绕 $O\zeta$ 轴旋转偏航角 $\psi$ :  $O\xi \rightarrow Ox_1$ 、 $O\eta \rightarrow Oy_1$ ,旋转矩阵为:

$$\boldsymbol{C}_{1}(\boldsymbol{\psi}) = \begin{bmatrix} \cos \boldsymbol{\psi} & \sin \boldsymbol{\psi} & \boldsymbol{0} \\ -\sin \boldsymbol{\psi} & \cos \boldsymbol{\psi} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{1} \end{bmatrix}$$
(2-1)

第二次绕 $Oy_1$ 轴旋转俯仰角 $\theta$ :  $Ox_1 \rightarrow Ox \cup O\zeta \rightarrow Oz_1$ , 旋转矩阵为:

$$\boldsymbol{C}_{2}(\boldsymbol{\theta}) = \begin{bmatrix} \cos \boldsymbol{\theta} & 0 & -\sin \boldsymbol{\theta} \\ 0 & 1 & 0 \\ \sin \boldsymbol{\theta} & 0 & \cos \boldsymbol{\theta} \end{bmatrix}$$
(2-2)

第三次绕Ox轴旋转滚转角 $\phi$ :  $Oy_1 \rightarrow Oy$ 、 $Oz_1 \rightarrow Oz$ , 旋转矩阵为:

$$\boldsymbol{C}_{3}(\boldsymbol{\phi}) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \boldsymbol{\phi} & \sin \boldsymbol{\phi} \\ 0 & -\sin \boldsymbol{\phi} & \cos \boldsymbol{\phi} \end{bmatrix}$$
(2-3)

将三个旋转矩阵相乘,即可得到动系与定系的转换矩阵 $J_1$ ,则使用欧拉角描述 ROV 线速度与位置之间的关系为:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \mathbf{J}_{1} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$
(2-4)

式中:

$$\mathbf{J}_{1} = \mathbf{C}_{1}\mathbf{C}_{2}\mathbf{C}_{3} = \begin{bmatrix} \cos\theta\cos\psi & -\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi & \sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi \\ \cos\theta\sin\psi & \cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi & -\sin\phi\cos\psi + \cos\phi\sin\theta\sin\psi \\ -\sin\theta & \sin\phi\cos\theta & \cos\phi\cos\theta \end{bmatrix}$$

在转动过程中,通过深入探究欧拉角(φ,θ,ψ)的变化率与角速度(p,q,r)在定坐标系 与动坐标系之间的关联,可以推导出以下数学公式来描述它们之间的关系:

$$\begin{cases} \dot{\phi} = p + q(\sin\phi\tan\theta) + r(\cos\phi\tan\theta) \\ \dot{\theta} = q(\cos\phi) + r(-\sin\phi) \\ \dot{\psi} = q(\sin\phi/\cos\theta) + r(\cos\phi/\cos\theta) \end{cases}$$
(2-5)

使用欧拉角描述 ROV 角速度与姿态角之间的关系为:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \mathbf{J}_2 \begin{bmatrix} p \\ q \\ r \end{bmatrix}$$
(2-6)

式中:

$$\mathbf{J}_{2} = \begin{bmatrix} 1 & \sin\phi \tan\theta & \cos\phi \tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix}$$

通过**J**<sub>1</sub>、**J**<sub>2</sub>两个矩阵可以实现机体坐标系和大地坐标系间速度的相互转换,结合公式(2-4)和(2-6),能够精确描述 ROV 的运动学方程:

$$\begin{bmatrix} \dot{\boldsymbol{\eta}}_1 \\ \dot{\boldsymbol{\eta}}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{J}_1 & \mathbf{0}_{3\times 3} \\ \mathbf{0}_{3\times 3} & \mathbf{J}_2 \end{bmatrix} \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \end{bmatrix}$$
(2-7)

通过应用运动学方程,能够在操作过程中精准预测 ROV 的运动轨迹和实时姿态,从 而为 ROV 的导航和控制提供至关重要的信息支持。

#### 2.2 惯性导航系统原理

惯性导航系统根据惯性测量元件 IMU 的安装方法可以划分为平台式和捷联式这两个 主要类别。平台式惯导系统(Platform Inertial Navigation System, PINS)将 IMU 所固定的稳 定平台的坐标系作为基准来测定机体的运动参数,省去了坐标系转换的过程,优点是计算 简单、系统精度高,不足之处是体型大结构复杂,可靠性相对较差,不利于在轻量化的移 动设备上使用,并且生产成本高。捷联式惯导系统(Strapdown Inertial Navigation System, SINS)在机体上安装 IMU,避免了复杂的机电系统,其优势在于体积紧凑、成本经济、易 于维护,但劣势在于对部件的高要求和坐标转换所需的大量计算,需要通过系统算法完成 坐标系转换过程从而解算得到机体量测物理量,以实现高精度的导航定位。

虽然 SINS 在惯性器件和计算量上的要求远高于 PINS,但随着技术进步和理论完善, SINS 的优势逐渐显现。本文采用 SINS 实现 ROV 导航定位,SINS 的核心组件包括 IMU 和相应的惯导算法。IMU 通过内置的加速度计和陀螺仪分别测量空间中三个正交轴向的 加速度和角速度,绝大多数场景下认为两者在同一坐标系 {*I*},然后运用算法对 IMU 获 取的数据进行解算得到机体的状态信息,系统流程如图 2-3 所示。



Fig.2-3 Flowchart of SINS Solution

#### 2.2.1 SINS 姿态解算

SINS 通过陀螺仪输出的角速度对系统姿态信息进行更新。欧拉角法存在奇异性问题的缺陷,在姿态表示过程中会出现"万向锁"现象<sup>[54]</sup>;方向余弦的姿态表示法计算量过大, 直接对微积分方程进行求解影响系统计算效率;相较于其他方法,四元数法的计算量较小 效率更高,因其显著的优势成为目前应用最广泛的姿态表示方法之一,更多地应用于 INS 等实际工程中。 哈密尔顿于 1843 年首次提出四元数的概念,利用四元数表示空间中三维坐标系的旋转<sup>[55]</sup>。四元数的定义为:

$$\overline{\boldsymbol{q}} = \boldsymbol{q}_0 + \boldsymbol{q} = \boldsymbol{q}_0 + \boldsymbol{q}_1 \boldsymbol{i} + \boldsymbol{q}_2 \boldsymbol{j} + \boldsymbol{q}_3 \boldsymbol{k}$$
(2-8)

上式中, 虚数单位i、j、k分别与空间坐标系中三个轴向上的单位矢量相对应, 四元数 在这三个轴向上的投影分别表示为实数 $q_1$ 、 $q_2$ 、 $q_3$ 。 $q_0$ 为四元数的实数部分, q为矢量。

假设角速度在单位时间内恒定不变,姿态四元数采用零阶积分,当惯性坐标系{*I*}相 对于地面参考系{*G*}移动时,可以通过计算差商的极限得到变化率或描述其关系的相应四 元数的导数。根据四元数性质进行推导得到:

$${}_{G}^{I}\dot{\bar{\boldsymbol{q}}}(t) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} \left( {}_{G}^{I}\dot{\bar{\boldsymbol{q}}}(t+\Delta t) - {}_{G}^{I}\dot{\bar{\boldsymbol{q}}}(t) \right) = \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}_{m}) {}_{G}^{I}\dot{\bar{\boldsymbol{q}}}(t) = \frac{1}{2} \begin{bmatrix} -\lfloor\boldsymbol{\omega}_{m} \times \rfloor & \boldsymbol{\omega}_{m} \\ -\boldsymbol{\omega}_{m}^{T} & 0 \end{bmatrix} {}_{G}^{I}\dot{\bar{\boldsymbol{q}}}(t) \quad (2-9)$$

式中:

$$\boldsymbol{\omega}_{m}$$
 —— {*I*} 系中表示的角速度测量值,  $\boldsymbol{\omega}_{m} = \begin{bmatrix} \omega_{x} & \omega_{y} & \omega_{z} \end{bmatrix}^{T}$ ;  
 $\lfloor \boldsymbol{\omega}_{m} \times \rfloor$  ——  $\boldsymbol{\omega}_{m}$  的反对称矩阵,  $\lfloor \boldsymbol{\omega}_{m} \times \rfloor = \begin{bmatrix} 0 & -\omega_{z} & \omega_{y} \\ \omega_{z} & 0 & -\omega_{x} \\ -\omega_{y} & \omega_{x} & 0 \end{bmatrix}^{\circ}$ .

采用毕卡法,根据角速度是否趋近于0,使用不同的数学公式来更新四元数:

$${}_{G}{}^{I}\hat{\overline{q}}(t+\Delta t) = \begin{cases} \left(\cos\left(\frac{|\boldsymbol{\omega}_{m}|}{2}\right)\cdot \boldsymbol{I}_{4\times4} + \frac{1}{|\boldsymbol{\omega}_{m}|}\sin\left(\frac{|\boldsymbol{\omega}_{m}|}{2}\right)\cdot\boldsymbol{\Omega}(\boldsymbol{\omega}_{m})\right){}_{G}{}^{I}\hat{\overline{q}}(t) & |\boldsymbol{\omega}| > 10^{-5} \\ \left(\boldsymbol{I}_{4\times4} - \frac{\Delta t}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_{m})\right){}_{G}{}^{I}\hat{\overline{q}}(t) & |\boldsymbol{\omega}| \le 10^{-5} \end{cases}$$
(2-10)

当利用四元数 q 来进行转动表示时,应该满足方程:

$$q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$$
 (2-11)

但是由于计算时存在一定的误差,所以在计算时必须对 $\bar{q}$ 进行归一化处理:

$$\bar{\boldsymbol{q}} = \frac{q_0 + q_1 \boldsymbol{i} + q_2 \boldsymbol{j} + q_3 \boldsymbol{k}}{\sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}}$$
(2-12)

基于向量投影的相对关系,选择任意向量 R,假定 {I} 系通过绕转轴 q 转动可到达 {G} 系,转动前后的  $R_I \propto R_G$  投影分别为  $R_I = x_I i + y_I j + z_I k \propto R_G = x_G i + y_G j + z_G k$ 。

基于转动的四元数表示法,可以得到:

$$\boldsymbol{R}_{G} = \boldsymbol{\bar{q}} \boldsymbol{R}_{I} \boldsymbol{\bar{q}}^{-1}$$
(2-13)

式中转轴q为转动四元数q中对应的向量部分。

将向量 $R_I$ 和 $R_G$ 分别用四元数的形式进行表示:

$$\begin{cases} \boldsymbol{R}_{I} = 0 + x_{I}\boldsymbol{i} + y_{I}\boldsymbol{j} + z_{I}\boldsymbol{k} \\ \boldsymbol{R}_{G} = 0 + x_{G}\boldsymbol{i} + y_{G}\boldsymbol{j} + z_{G}\boldsymbol{k} \end{cases}$$
(2-14)

将式(2-8)和(2-14)代入式(2-13)得:

$$x_{G}\boldsymbol{i} + y_{G}\boldsymbol{j} + z_{G}\boldsymbol{k} = (q_{0} + q_{1}\boldsymbol{i} + q_{2}\boldsymbol{j} + q_{3}\boldsymbol{k})(x_{I}\boldsymbol{i} + y_{I}\boldsymbol{j} + z_{I}\boldsymbol{k})(q_{0} - q_{1}\boldsymbol{i} - q_{2}\boldsymbol{j} - q_{3}\boldsymbol{k}) \quad (2-15)$$

根据四元数乘法对式(2-15)进行展开并写成矩阵的形式为:

$$\begin{bmatrix} x_G \\ y_G \\ z_G \end{bmatrix} = C_I^G \begin{bmatrix} x_I \\ y_I \\ z_I \end{bmatrix}$$
(2-16)

通过四元数表示的捷联矩阵可以表示为:

$$\boldsymbol{C}_{I}^{G} = \begin{bmatrix} q_{0}^{2} + q_{1}^{2} - q_{2}^{2} - q_{3}^{2} & 2(q_{1}q_{2} - q_{0}q_{3}) & 2(q_{1}q_{3} + q_{0}q_{2}) \\ 2(q_{1}q_{2} + q_{0}q_{3}) & q_{0}^{2} - q_{1}^{2} + q_{2}^{2} - q_{3}^{2} & 2(q_{2}q_{3} - q_{0}q_{1}) \\ 2(q_{1}q_{3} - q_{0}q_{2}) & 2(q_{2}q_{3} + q_{0}q_{1}) & q_{0}^{2} - q_{1}^{2} - q_{2}^{2} + q_{3}^{2} \end{bmatrix}$$
(2-17)

捷联矩阵 $C_I^G$ 和 2.1 节的求得的转换矩阵 $\mathbf{J}_1$ 有对应关系,则机体姿态角可通过下式进行求解:

$$\begin{cases} \phi = \arctan\left[\frac{2(q_2q_3 + q_0q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2}\right] \\ \theta = \arcsin\left[2(q_0q_2 - q_1q_3)\right] \\ \psi = \arctan\left[\frac{2(q_1q_2 + q_0q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2}\right] \end{cases}$$
(2-18)

#### 2.2.2 SINS 位置和速度解算

采用四阶的龙格库塔法(Runge-Kutta Methods)对 SINS 的速度和位置信息进行递推, 具体计算过程如下:

$$\begin{cases} \dot{\hat{\boldsymbol{p}}}(t) = \hat{\boldsymbol{v}}(t) \\ \dot{\hat{\boldsymbol{v}}}(t) = \hat{\boldsymbol{C}}_{I}^{G}(t)\boldsymbol{a}_{m} + \boldsymbol{g} \end{cases}$$
(2-19)

式中:

 $a_m \longrightarrow \{I\}$ 系中表示的加速度测量值, $a_m = \begin{bmatrix} a_x & a_y & a_z \end{bmatrix}^T$ ;

g——重力加速度项,  $g = \begin{bmatrix} 0 & 0 & 9.8 \end{bmatrix}^{T}$ 。

对下一时刻速度具体计算过程如下:

$$\hat{\mathbf{v}}(t+\Delta t) = \hat{\mathbf{v}}(t) + \frac{\Delta t}{6} (\mathbf{k}_{v1} + 2\mathbf{k}_{v2} + 2\mathbf{k}_{v3} + \mathbf{k}_{v4})$$
(2-20)

对下一时刻位置具体计算过程如下:

$$\hat{p}(t+\Delta t) = \hat{p}(t) + \frac{\Delta t}{6} (k_{p1} + 2k_{p2} + 2k_{p3} + k_{p4})$$
(2-21)

速度和位置预测公式中的各项龙格库塔系数k 分别为:

$$\begin{cases} \boldsymbol{k}_{v1} = \hat{\boldsymbol{C}}_{I}^{G}(t)\boldsymbol{a}_{m} + \boldsymbol{g} \\ \boldsymbol{k}_{v2} = \hat{\boldsymbol{C}}_{I}^{G}\left(t + \frac{\Delta t}{2}\right)\boldsymbol{a}_{m} + \boldsymbol{g} \\ \boldsymbol{k}_{v3} = \hat{\boldsymbol{C}}_{I}^{G}\left(t + \frac{\Delta t}{2}\right)\boldsymbol{a}_{m} + \boldsymbol{g} \\ \boldsymbol{k}_{v4} = \hat{\boldsymbol{C}}_{I}^{G}(t + \Delta t)\boldsymbol{a}_{m} + \boldsymbol{g} \\ \boldsymbol{k}_{v4} = \hat{\boldsymbol{C}}_{I}^{G}(t + \Delta t)\boldsymbol{a}_{m} + \boldsymbol{g} \\ \begin{cases} \boldsymbol{k}_{p1} = \hat{\boldsymbol{v}}(t) \\ \boldsymbol{k}_{p2} = \hat{\boldsymbol{v}}(t) + \boldsymbol{k}_{v1}\frac{\Delta t}{2} \\ \boldsymbol{k}_{p3} = \hat{\boldsymbol{v}}(t) + \boldsymbol{k}_{v2}\frac{\Delta t}{2} \\ \boldsymbol{k}_{n4} = \hat{\boldsymbol{v}}(t) + \boldsymbol{k}_{v3}\Delta t \end{cases}$$
(2-23)

### 2.3 惯性导航误差分析

#### 2.3.1 惯性导航定位仿真

从 IMU 传感器获取到加速度和角速度数据后,根据以上 SINS 位姿估计原理进行惯 性导航仿真分析,机体初始位置为(0,0,0)、姿态为(0,0,0),历经 15.7999s 的运动之后, SINS 解算得到的终点位置为(63.4569m, 11.4794m, -0.9993m)、姿态为(0.0099°, -0.0137°, -0.2098°),实际的终点位置为(61.5846m, 9.0336m, -0.5768m)、姿态为(0.0076°, 0.0038°, -0.2271°)。运动轨迹如图 2-4 所示,红色曲线是 SINS 解算得到的轨迹,绿色曲线是真实的 三维运动轨迹。SINS 解算的六自由度位姿误差如图 2-5 所示,位置平均均方根误差 (Average Root Mean Square Error, ARMSE)为0.965215m,姿态估计的 ARMSE 为0.020822°, 最终位置在三个方向的误差总和为 3.109005m,最终姿态在三个方向的误差总和为 0.024738°。



图 2-4 SINS 解算轨迹与真实轨迹对比 Fig.2-4 Comparison of SINS Calculated Trajectory with Real Trajectory



Fig.2-5 Six Degrees of Freedom Error of Pose Computed by SINS

以上仿真结果可以看出,与真实位姿相比惯性导航位姿估计结果存在误差,并随着运动时间增加累积误差越来越大,导致导航定位结果的可靠性大大降低,这进一步说明了利用视觉信息进行补偿修正的必要性。接下来对惯性导航误差来源进行分析建模,为惯导与视觉的信息融合奠定基础。

#### 2.3.2 IMU 测量误差分析

为了尽可能精确地获得机体在空间内的位姿估计,首要之务是收集实时加速度和角速度等关键数据信息。这些信息不仅有助于了解机体的运动状态,还能为后续的姿态分析和位置确定提供重要的依据。IMU 传感器,作为一种常见的测量工具,其结构简单而高效,由3个单轴的陀螺仪与3个单轴加速度计组成。这些组件协同工作,使得 IMU 能够实时地捕捉并输出物体在空间内的加速度和角速度数据。IMU 在机动性和实时性方面表现出色,但在测量过程中难免会混入一些杂乱的信息。这些信息可能来源于多种因素,其中最为常见的是高斯白噪声,这会干扰传感器的测量数据,导致所得结果存在误差。此外传感器本身在使用过程中也会因为各种因素(如温度变化、机械振动等)而产生零偏,同样会引入误差从而影响到测量结果的准确性。

这些误差和零偏对惯性系统的估计精度有着不可忽视的影响。在对姿态角和位移进 行估计时,可能会使估计结果与实际值之间出现明显的偏离。因此建立 IMU 的测量模型 必须充分考虑这些误差因素。通过采取适当的方法尽可能地减小误差,提高测量数据的准 确性和可靠性。这样才能更准确地获得机体在空间内的位姿估计,为后续的分析和决策提 供有力的支持。下面介绍 IMU 测量数据过程中存在的确定性误差和随机误差,并建立测 量模型。

(1) 确定性误差

理想状态下, IMU 外部作用为 0 时, 输出结果也应该为 0。而在实际情况中, 由于元器件的物理特性, 即使在 IMU 静止时, 也会产生一个输出 *b*, 这是确定性的偏置, 也称

零偏。偏置通常在系统里面作为状态量去估计,长时间缓慢变化,短时间内认为不变。偏置的导数是高斯白噪声记为 $\varepsilon_b(t)$ ,即:

$$\dot{b}(t) = \varepsilon_b(t) \tag{2-24}$$

(2) 随机误差

IMU 传感器得到的数据会有一个时刻间互不干扰的高斯白噪声,记为 $\varepsilon(t)$ 。实际情况下, IMU 采集的数据是离散的,因此需要把高斯白噪声也进行离散化采样。下面介绍连续误差与离散误差的表达式,通过连续时间下白噪声的标准差计算离散的标准差。设 $N_i$ 为白噪声序列中的第i个随机变量,表示离散的噪声。每个 $N_i$ 都有相同的分布,期望 $E(N_i) = E(N) = 0$ ,方差 $Var(N_i) = Var(N) = \sigma^2$ 。当i = j时, $Cov(N_i, N_j) = 0$ 。计算高斯白噪声的积分:

$$\int_{0}^{t} \varepsilon(\tau) d\tau = \Delta t \sum_{i=1}^{n} N_{i}$$
(2-25)

上式中n为从传感器接收的样本数,  $\Delta t$ 为 IMU 传感器采样时间。

使用标准公式:

$$E(aX+bY) = aE(X)+bE(Y)$$
(2-26)

$$Var(aX+bY) = a^{2}Var(X) + b^{2}Var(Y) + 2abCov(X,Y)$$
(2-27)

式中a和b是常数, X和Y是随机变量, 可以得出:

$$E\left(\int_{0}^{t} \varepsilon(\tau) d\tau\right) = \Delta t \cdot n \cdot E(N) = 0$$
(2-28)

$$Var\left(\int_{0}^{t} \varepsilon(\tau) d\tau\right) = \left(\Delta t\right)^{2} \cdot n \cdot Var(N) = \Delta t \cdot t \cdot \sigma^{2}$$
(2-29)

连续标准差为 $\sigma_c(t) = \sigma \sqrt{\Delta t \cdot t}$ 。当取t = ls时, $\sigma_c$ 是一个已知的标定结果,所以误差的离散标准差相当于在标定结果基础上除以 $\sqrt{\Delta t}$ ,即:

$$\sigma = \frac{\sigma_c}{\sqrt{\Delta t}} \tag{2-30}$$

接下来对偏置随时间变化的过程进行建模,白噪声经过一次积分后便形成了随机游 走过程。由上面的推导可知,白噪声离散下的标准差等于连续情况下结果除以 $\sqrt{\Delta t}$ 。不同 的是,此处的标准差不管是离散的还是连续的,都是偏置的导数。实际计算要的是偏置随 机游走,在离散情况下还要乘以 $\Delta t$ ,所以最终看起来从标定结果 $\sigma_{bc}$ 到离散的偏置随机游 走 $\sigma_b$ 相当于乘以 $\sqrt{\Delta t}$ 。离散偏置随机游走标准差计算如下:

$$\sigma_b = \frac{\sigma_{bc}}{\sqrt{\Delta t}} \Delta t = \sigma_{bc} \sqrt{\Delta t}$$
(2-31)

(3) IMU 测量模型

把相应的噪声考虑进去对 IMU 进行参数化建模, IMU 的噪声模型可由下式表示:

$$\begin{cases} \boldsymbol{\omega}_{m}(t) = \boldsymbol{\omega}(t) + \boldsymbol{b}_{g}(t) + \boldsymbol{\varepsilon}_{g}(t) \\ \boldsymbol{a}_{m}(t) = \boldsymbol{C}_{I}^{G}(t)(\boldsymbol{a}(t) - \boldsymbol{g}) + \boldsymbol{b}_{a}(t) + \boldsymbol{\varepsilon}_{a}(t) \end{cases}$$
(2-32)

式中 $\boldsymbol{b}_a$ 为加速度计偏差, $\boldsymbol{b}_a$ 为陀螺仪偏差, $\boldsymbol{\varepsilon}_a$ 为加速度计白噪声, $\boldsymbol{\varepsilon}_a$ 为陀螺仪白噪声。

建立好的 IMU 测量模型将用于融合算法中,一般通过标定对固定偏置的确定性误差进行补偿,而随系统运行不断变化的随机误差必须通过在线矫正,以此得到加速度计和陀螺仪的修正值进行姿态预测。

# 2.4 相机成像与图像预处理

#### 2.4.1 相机成像模型

在视觉图像领域,任何图像可以看做是由众多像素组成的二维矩阵,每个像素储存了 对应位置的亮度和色彩信息。任何在三维空间中由相机观察到的坐标点,利用坐标系之间 的转换关系可以计算出该坐标点在相机平面上的像素点投影。针孔相机的理论模型主要 基于小孔成像的光学原理,光线穿过小孔投影在相机的成像平面上形成物体的倒像<sup>[56]</sup>。基 于小孔成像原理,可以建立摄像机成像模型,如图 2-6 所示。



图 2-6 相机成像模型 Fig.2-6 Camera Imaging Model

三维世界中的空间信息通过相机拍摄为二维图像,在从三维路标点到二维像素点的 坐标转换过程中,涉及到世界、相机、图像以及像素四个坐标系。C为相机坐标系下相机 的光心,C'点位于物理成像平面的中心,CC'为相机的光轴,摄像机的焦距f为C到成像 平面的距离。假设空间中一点P在相机上的坐标为(x,y,z),其对应的成像平面内的坐标 为(x',y')。根据透视投影关系,小孔所得的物像是翻转的。为了便于理解,将成像平面一 比一对称地挪到相机前面,将空间中的点以及相机位置放于同一边,这样可以不用考虑正 负号的影响。于是可以得到:

$$\begin{cases} x' = f \frac{x}{z} \\ y' = f \frac{y}{z} \end{cases}$$
(2-33)

然而像素平面内坐标与相机坐标系下坐标的变换关系并不完全按上式计算。成像平面和像素平面间有缩放和原点平移的差距,如图 2-7 所示。通常情况像素平面和成像平面的原点并不统一,另外在成像平面内坐标是用米等单位定义的,但像素坐标一般用像素(pixel)作为单位表示。



图 2-7 像素平面与成像平面 Fig.2-7 Pixel Plane and Imaging Plane

假设 X' 轴和U 轴存在 $\alpha$  倍的缩放关系, Y' 轴和V 轴存在  $\beta$  倍的缩放关系, C' 在像素平面的坐标为( $u_0, v_0$ ), 于是投影点在像素平面的坐标为:

$$\begin{cases} u = \alpha x' + u_0 \\ v = \beta y' + v_0 \end{cases}$$
(2-34)

将式(2-33)代入式(2-34),用 $f_x$ 代替 $\alpha f$ ,用 $f_y$ 代替 $\beta f$ 。式中 $\alpha$ 、 $\beta$ 以"像素/米" 为单位,整理后的公式为:

$$\begin{cases} u = f_x \frac{x}{z} + u_0 \\ v = f_y \frac{y}{z} + v_0 \end{cases}$$
(2-35)

上式中f以米为单位,  $f_x$ 、 $f_y$ 和 $u_0$ 、 $v_0$ 的单位都是像素。

那么在齐次坐标下可以将式(2-35)改写为:

$$z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \triangleq KP$$
(2-36)

用 $f_x$ 、 $f_y$ 、 $u_0$ 、 $v_0$ 表示相机的内参数,通常认为是固定不变的,共同构成的矩阵**K** 

被称作相机的内部参数矩阵。有了参数矩阵和距离z,就可以完成像素、相机和图像这三个坐标系之间的转换。世界坐标系上的坐标点可以利用旋转矩阵 **R** 与平移矢量t 转换到相 机坐标系上:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{R} \begin{bmatrix} x_w \\ y_w \\ z_w \end{bmatrix} + \mathbf{t}$$
(2-37)

代入式(2-36)可得,任意空间点P可通过式(2-38)从世界坐标系转换到像素坐标系:

$$z\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{R}_{3\times3} & \mathbf{t}_{3\times1} \end{bmatrix} \begin{vmatrix} x_w \\ y_w \\ z_w \\ 1 \end{vmatrix} \triangleq \mathbf{KTP}_w$$
(2-38)

式中:

 $P_w$ ——世界坐标系下的齐次坐标;

T——外参矩阵,由旋转矩阵R与平移向量t构成。

#### 2.4.2 相机标定理论

相机标定的准确性对于获取精确的相机参数至关重要,其精确度将直接影响定位系统的计算效率和位姿解算的准确性。张氏相机标定法介于传统的标定法与自动标定法两者之间,它具有操作简便、所需设备简单、标定准确度较高以及角点变化明显等优势。正因如此,该方法在计算机视觉领域得到了广泛应用,缺点是需要在标定过程中人为多次转动和平移平面模板。在本文中,采用了张氏标定法,通过在同一平面内拍摄多幅不同姿态、不同视角和不同部位的标准模板图片,即可计算得到相机的内外参数<sup>[57]</sup>。内参的主要作用在于调整相机坐标系与图像坐标系之间因单位不统一而产生的线性变换差异,而外参则负责实现坐标点在相机坐标系与世界坐标系之间的准确转换。

根据公式(2-38)将z=0平面上点 $P_w(x_w, y_w, 0)$ 通过投影变换映射到像素坐标系下,可以得到该点的像素坐标 $p_w = (u, v)$ 与世界坐标之间的对应关系:

$$z\begin{bmatrix} u\\v\\1\end{bmatrix} = K\begin{bmatrix} r_1 & r_2 & r_3 & t\end{bmatrix} \begin{vmatrix} x_w\\y_w\\0\\1\end{bmatrix} = K\begin{bmatrix} r_1 & r_2 & t\end{bmatrix} \begin{bmatrix} x_w\\y_w\\1\end{bmatrix}$$
(2-39)

记单应性矩阵  $H = \lambda K[r_1 \ r_2 \ t] = [h_1 \ h_2 \ h_3]$ ,式中 $r_1$ 、 $r_2$ 是旋转矩阵 R的分量,由旋转矩阵的正交性可知:  $r_1^T r_2 = 0$ ,  $||r_1|| = ||r_2|| = 1$ ,根据这一性质可以进一步推导出以下约束条件:

$$\begin{cases} \boldsymbol{h}_1^T \boldsymbol{K}^{-T} \boldsymbol{K}^{-1} \boldsymbol{h}_2 = 0 \\ \boldsymbol{h}_1^T \boldsymbol{K}^{-T} \boldsymbol{K}^{-1} \boldsymbol{h}_1 = \boldsymbol{h}_2^T \boldsymbol{K}^{-T} \boldsymbol{K}^{-1} \boldsymbol{h}_2 \end{cases}$$
(2-40)

为求解方便, 记 $B = \lambda K^{-T} K^{-1}$ ,  $\lambda$ 表示任意比例因子, 将B 展开可以得到:

$$\boldsymbol{B} = \lambda \boldsymbol{K}^{-T} \boldsymbol{K}^{-1} = \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} = \lambda \begin{bmatrix} \frac{1}{f_x^2} & 0 & \frac{-u_0}{f_x^2} \\ 0 & \frac{1}{f_y^2} & \frac{-v_0}{f_y^2} \\ \frac{-u_0}{f_x^2} & \frac{-v_0}{f_y^2} & \frac{u_0}{f_x^2} + \frac{v_0}{f_y^2} + 1 \end{bmatrix}$$
(2-41)

上式为对称矩阵,选择其中的6个因子构建新的矩阵b:

 $\boldsymbol{b} = \begin{bmatrix} B_{11} & B_{12} & B_{22} & B_{13} & B_{23} & B_{33} \end{bmatrix}^T$ (2-42)

根据上面的约束条件(2-40)可以得出:

$$\boldsymbol{h}_i^T \boldsymbol{B} \boldsymbol{h}_j = \boldsymbol{v}_{ij}^T \boldsymbol{b} = 0 \tag{2-43}$$

$$\begin{bmatrix} \mathbf{v}_{12}^T \\ \left(\mathbf{v}_{11} - \mathbf{v}_{22}\right)^T \end{bmatrix} \mathbf{b} = \mathbf{0}_{2\times 1}$$
(2-44)

 $\pm \exists t \neq \mathbf{v}_{ij} = \begin{bmatrix} h_{i1}h_{j1} & h_{i1}h_{j2} + h_{i2}h_{j1} & h_{i2}h_{j2} & h_{i3}h_{j1} + h_{i1}h_{j3} & h_{i3}h_{j2} + h_{i2}h_{j3} & h_{i3}h_{j3} \end{bmatrix} \circ$ 

拍摄每张图片可以产生两个约束方程,由于相机的内参矩阵涉及5个未知元素,所以 只需拍摄3张图像,就能通过解这些方程来确定内参系数。内参矩阵中元素如下:

$$f_{x} = \sqrt{\frac{\lambda}{B_{11}}}$$

$$f_{y} = \sqrt{\frac{\lambda B_{11}}{\left(B_{11}B_{22} - B_{12}^{2}\right)}}$$

$$u_{0} = -\frac{B_{13}f_{x}^{2}}{\lambda}$$

$$v_{0} = \frac{\left(B_{12}B_{13} - B_{11}B_{23}\right)}{\left(B_{11}B_{22} - B_{12}^{2}\right)}$$

$$\lambda = \frac{B_{33} - \left[B_{13}^{2} + v_{0}\left(B_{12}B_{13} - B_{11}B_{23}\right)\right]}{B_{11}}$$
(2-45)

在单应性矩阵中引入内参系数后,可以得到以下的外参系数:

$$\begin{cases} \boldsymbol{r}_{1} = \frac{1}{\lambda} \boldsymbol{K}^{-1} \boldsymbol{h}_{1} &, \quad \boldsymbol{r}_{2} = \frac{1}{\lambda} \boldsymbol{K}^{-1} \boldsymbol{h}_{2} \\ \boldsymbol{r}_{3} = \boldsymbol{r}_{1} \times \boldsymbol{r}_{2} &, \quad \boldsymbol{t} = \frac{1}{\lambda} \boldsymbol{K}^{-1} \boldsymbol{h}_{3} \end{cases}$$
(2-46)

由此得到了每一张图像对应的单应性矩阵 H,也计算出了相机的内参矩阵 K,带入 上式即可获得每张图像的外参系数旋转矩阵  $R_{3\times 3} = [r_1 \ r_2 \ r_3]$ 和平移矢量 t。

#### 2.4.3 水下图像增强

由于水下环境的光线较为微弱,导致摄像机拍摄的低照度图像呈现出较高的模糊度 以及较低的对比度,利用原水下图像进行特征点提取效果不佳,对后续研究有很大影响。 在视觉定位系统之前需要增强水下图像的细节信息,本节研究图像增强算法来进行水下 视觉信息预处理。直方图均衡化(Histogram Equalization, HE)算法基于像素统计信息进行 图像增强,通过重新分配图像像素的灰度值,将图像的直方图进行均匀分布,从而提升图 像的对比度。传统的 HE 算法会导致图像出现对比度过度增强和噪声放大的问题,调整后 更亮/暗的区域反而提取不到更多特征。限制对比度自适应直方图均衡化(CLAHE)<sup>[58]</sup>将图 像进行分割局部应用 HE,通过实施对比度限制并结合双线性插值技术,在增强图像对比 度的同时,也显著提升了图像中细节的可见性和清晰度,解决了传统 HE 的缺陷。下面介 绍 CLAHE 的具体过程。

(1) 将待处理的图像分割成多个大小相等的不重叠子块,这些区域被称为局部直方图 均衡化单元,每个子块含有的像素数为 N;

(2) 对于每个局部直方图均衡化单元,应用传统的 HE 算法使像素分布更加均匀,以此提升图像的对比度。具体包括以下几个步骤:计算每个区域内像素的灰度直方图;根据 直方图计算对应的累积分布函数(Cumulative Distribution Function, CDF);根据 CDF 重新 分配像素的灰度值,以实现直方图的均衡化;根据分配的灰度值更新图像中对应区域的像 素值。对于灰度图,像素值分布为 0-255, *n<sub>x</sub>*为像素值为*x*的点数,则像素值为*x*的点增 强后的像素值为:

$$y(x) = 255 \sum_{i=0}^{x} \frac{n_x}{N}$$
(2-47)

(3)为了避免过度增强,对于每个局部直方图均衡化单元图像像素的灰度值进行了限制,将直方图在阈值 T 处进行截断,对于超出截取阈值的部分重新均匀分布在整个灰度范围上,整个直方图上升高度 L,以此来保证直方图总面积不变、概率密度的积分仍然为 1, CLAHE 原理示意图如图 2-8 所示。通过调整截取阈值,CLAHE 约束了局部对比度的增强幅度,能够实现差异化的增强效果,尽可能保留了图像的局部细节;





(4) 为了缓解分割区域之间的边缘效应,引入双线性插值进行改进,如图 2-9 所示为 插值示意图。图中黑色小方块处仍然采用原始分割子块的变换函数进行计算,在蓝色阴影 区域的像素上,通过与其相邻的四个黑色小方块处的变换函数进行双线性插值计算来获 取其值。而对于边缘的绿色阴影部分,则采用与其相邻的两个黑色小方块处的变换函数进 行线性插值计算。至于红色阴影的四个角点处,则直接应用它们所处子块的变换函数来得 到其值;



Fig.2-9 Interpolation Diagram

(5) 将经 CLAHE 处理后的各个局部均衡化单元拼接起来,得到最终的增强图像。

以上是完整的 CLAHE 算法流程,水下图像增强后对比度有一定程度的提升。水下图像增强前后对比如图 2-10 所示。



(a) 增强前图像







由上图可以看出,原水下图像模糊程度很高,在进行水下图像增强之前,所提取的特征点数量往往非常有限。然而通过应用预处理方法,图像清晰度显著提高,从而大大增加 了特征点的提取数量。这为后续的视觉信息提取和回环检测提供了重要帮助,可以提升水 下导航定位系统的性能。

#### 2.5 本章小结

本章首先建立了三维空间中的坐标系统和 ROV 运动学模型,为理解 ROV 运动规律 和位姿表示方式提供了基础。接下来深入阐述了系统传感器相关的理论知识,特别是捷联 式惯性导航的姿态、位置和速度解算方法。通过仿真定量分析了误差来源并建立了 IMU 测量模型。最后从相机与图像的关系出发,建立了相机成像模型,实现了坐标点在不同坐 标系之间的转换,并推导了相机标定解算内外参数的流程。在此基础上,本章还探讨了 CLAHE 算法在水下图像增强中的应用,有效提升了图像质量。这些理论知识为后续视觉 /惯导融合算法的研究提供了理论支撑,对于算法的实现和优化至关重要,也为后续研究 工作提供了指导和帮助。

# 3 视觉与惯导的信息融合算法研究

视觉与惯导信息融合的导航定位技术自最初被提出至今,从视觉信息和惯导数据融合的视角出发可以划分为松耦合和紧耦合两种方案,如图 3-1 所示为两种方案的示意图。 松耦合的方法是将视觉传感器数据和 IMU 的测量数据分别用于独立计算机体的运动状态, 再将这两种不同来源的状态信息进行合理的融合或耦合,以获取经过优化处理的、更为准确的位姿信息。紧耦合的方法联合优化相机和 IMU 数据,最终通过处理融合后的数据获取到准确的位姿信息。由于紧耦合的方法采用了联合处理并优化不同传感器数据的策略, 与松耦合相比,这种方式所需的计算时间较长,但是精度上有较为突出的性能。



Fig.3-1 Loose Coupling and Tight Coupling

上一章已经对 IMU 与视觉传感器的相关原理进行了详细介绍并建立了相应的数学模型,并对获取到的传感器数据进行了预处理。本章的重点是要进行传感器数据融合,由于 松耦合的处理方案数据间的耦合程度较低,因此本文采用紧耦合的方案融合视觉与惯导 的信息。本章首先需要从增强后的水下图像中提取特征点,基于图像处理获得的特征点数 据,融合 IMU 测量结果来估计机体的运动状态,并深入探讨相关的理论和数学模型,最 后利用数据集进行仿真分析。

# 3.1 视觉图像处理

#### 3.1.1 FAST 特征点检测

特征点检测算法有很多,各种相关改进算法也层出不穷,在水下定位系统中,特征提取只是一部分,若此处利用效果非常好的 SIFT、SURF 等算法,会使得实时性变差降低系统性能。Features from accelerated segment test 算法通过比较目标像素与其周围像素的亮度值,来迅速确定该像素是否可作为特征点,这是 2006 年 E.Rosten 等人提出的一种快速有效的特征点检测算法<sup>[59]</sup>,也是目前检测速度最快的算法之一<sup>[60]</sup>。中文名为"基于加速分割测试的特征",简称 FAST。因其筛选方法的简洁性和更高的计算效率,经常被应用于需要

实时处理的应用场景中。FAST 特征点检测示意图如图 3-2 所示。



图 3-2 FAST 特征点检测示意图 Fig.3-2 FAST Feature Point Detection Diagram

具体的检测步骤如下:

(1) 在灰度处理后的水下图像中任选点 p, 令其灰度值为  $I_p$ ;

(2) 根据图 3-2 所示作圆,分别对圆周上分布的像素点标记为 p1、 p2…p16;

(3) 定义一个阈值*t*,检查1和9像素点,如果*Ip*<sub>1</sub>、*Ip*<sub>9</sub>都在[*Ip*-*t*,*Ip*+*t*]之间,则*p* 不可能是特征点,返回步骤(1)判断下一个点;否则进行下一步;

(4) 如果 *p* 被视为一个候选点,那么需要计算 *p*1、 *p*5、 *p*9、 *p*13与中心 *p* 之间的像 素差异。如果它们的绝对数值中有三个或更多超出了设定的阈值,那么可以将其视为候选 点,并进行后续的评估;否则直接判断下一个点;

(5) 如果 p 仍然被视为一个候选点,那么需要计算其与 p1 至 p16这 16 个点之间的像 素差异,如果其中至少有 N 个像素超出了设定的阈值,那么 p 就被认为是特征点;否则 丢弃;

(6) 每帧图片逐个像素经过以上步骤检测之后再进行非极大值抑制处理,首先通过  $3 \times 3$  的滑动窗口检查区域内是否存在多个特征点,需要分别计算各 FAST 特征点的得分值  $s = \sum |Ip - t|$ ,保留窗口内得分值最大的 FAST 特征点;在滑动窗口中,如果只存在一个 特征点,那么就应该直接保留这个特征点。

由上可以看出,FAST 算法实现起来简单,这也是其速度快的原因。步骤(3)、(4)通过 简单的预处理排除了灰度图片中大多数非特征点的像素。步骤(5)中 N 通常取 9、11、12 等,称作 FAST-N。为了防止检测到的特征点过于集中造成计算资源浪费,步骤 f 利用非 极大值抑制策略进行了稀疏化处理,在滑动窗口内只保留了响应值较高的特征点有助于 后续分析,至此完成了整个水下图像的特征点检测任务。

#### 3.1.2 基于图像金字塔的 LK 特征点跟踪

在视觉图像处理的流程中,一旦从当前帧中提取出特征点,接下来的关键步骤是与相

邻帧的特征点进行匹配,从而建立起这些特征点在连续图像帧之间的对应关系,实现图像间的特征点关联,方便对相机运动姿态的估计和位姿优化。稠密光流和稀疏光流是最常用的特征匹配或跟踪方法。稀疏光流法仅求解图片中某些像素点的运动,因此计算速度快实时性高,比较适合对实时性有要求较高的水下移动平台,但也存在精度较低的缺陷。而稠密光流法<sup>[61]</sup>需要求解整幅图片全部像素点的运动。本文采用 LK 光流法<sup>[62]</sup>追踪特征点的移动过程以表征相机的运动状态,以达到 ROV 位姿估计的目的。LK 光流法对相邻帧图片的特征点做差分运算进行稀疏光流估计,本节将介绍 LK 光流的基本原理。

LK 光流法有三个假设条件:

(1) 灰度恒定: 同一像素点的灰度值在任意一帧图片中保持恒定不变;

(2) 微小运动: 在相机观测过程中像素点的位置不会发生很大的改变,基于此可以利用相邻帧之间的特征点位置变动导致的灰度值改变,计算出灰度值关于位置的偏导数;

(3) 空间一致:在相机观测过程中,由于图像的稳定性和连续性,相邻像素点的相对 位置关系保持不变,即它们依然相邻。基于这一特性,可以利用邻域内的*n*个像素点来构 建出*n*个方程,并通过联立这些方程来求解速度信息。

LK 光流法的示意图如图 3-3 所示, 接下来对 LK 光流法研究的数学模型进行推导。



图 3-3 LK 光流法示意图 Fig.3-3 LK Optical Flow Method Diagram

将图像中像素的灰度值定义为关于时间和坐标的函数,具体地说,在时间t,坐标为 (x,y)的像素点,其灰度值被表示为I(x,y,t);后一帧时间为t+dt,该像素点的位置为 (x+dx,y+dy),灰度值为I(x+dx,y+dy,t+dt)。根据灰度恒定假设可以列出如下等式:

$$I(x+dx, y+dy, t+dt) = I(x, y, t)$$
(3-1)

根据微小运动假设,将式(3-1)泰勒展开忽略高阶项:

$$I(x+dx, y+dy, t+dt) \approx I(x, y, t) + \frac{\partial I}{\partial x}dx + \frac{\partial I}{\partial y}dy + \frac{\partial I}{\partial t}dt$$
(3-2)

结合式(3-1)和(3-2),可以得到:

$$\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dx}{dt} = -\frac{\partial I}{\partial t}$$
(3-3)
# 式中:

- dx/dt ——像素在 x 轴方向的运动速度,记为 u; dy/dt ——像素在 y 轴方向的运动速度,记为 v;  $\partial I/\partial x$  ——图像关于 x 轴方向的梯度,记为  $I_x$ ;  $\partial I/\partial y$  ——图像关于 y 轴方向的梯度,记为  $I_y$ ;
- $\partial I / \partial t$ ——当前帧图像在该像素点处关于时间的梯度,记为 $I_t$ 。

# $I_x$ 、 $I_y$ 和 $I_t$ 可通过以下式子得到:

$$\begin{cases} I_x = \frac{1}{2}I(x+1, y, t) - I(x-1, y, t) \\ I_t = I(x, y, t) - I(x, y, t+dt) \\ I_y = \frac{1}{2}I(x, y+1, t) - I(x, y-1, t) \end{cases}$$
(3-4)

式(3-3)可写为如下形式:

$$I_x u + I_y v = -I_t \tag{3-5}$$

上式含有两个未知数速度*u、v*,根据空间一致假设,在图像的局部邻域内,像素的运动是一致的。因此,对于一个*r×r*的方格内的*n*个像素点,可以根据它们的运动一致性构建出*n*个方程:

$$\begin{bmatrix} I_{xk} & I_{yk} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -I_{tk} , \quad k = 1, 2, \cdots, n$$
(3-6)

将式(3-6)写成如下的形式:

$$A\begin{bmatrix} u\\v \end{bmatrix} = -b \tag{3-7}$$

式中:

$$\mathbf{A} = \begin{bmatrix} I_{x1} & I_{y1} \\ \vdots & \vdots \\ I_{xn} & I_{yn} \end{bmatrix} , \quad \mathbf{b} = \begin{bmatrix} I_{t1} \\ \vdots \\ I_{tm} \end{bmatrix}$$

利用最小二乘法进行求解得到:

$$\begin{bmatrix} u \\ v \end{bmatrix} = -(A^T A)^{-1} A^T b$$
(3-8)

当A<sup>T</sup>A可逆时,上述方程在最小二乘意义下具有唯一解。在实际应用中,为了求解光流,通常会采用迭代的方法来进行数值计算,每次迭代都会利用式(3-8)得到前后两帧图像 之间的运动速度。

在 LK 光流法的应用中, 其基于的假设条件通常相当严格。然而, 在实际操作中, 当

系统经历高速运动时,像素会产生显著的位移,这种情况下,原先严格的假设条件可能不 再成立,从而导致标准的 LK 光流法难以获得精确的结果。因此研究图像金字塔的 LK 光 流优化方法,该方法通过降采样技术对图片进行不同级别的缩放来构建金字塔,而金字塔 的最底部则是原始图像,如图 3-4 所示。构建金字塔模型后会缩小图像尺寸,这意味着在 两帧图像之间的时间间隔保持不变的情况下,图像的运动速度也会相应地降低。通过分层 的方法,首先在最顶层运用标准 LK 光流法求解运动矢量,然后将这个初始估计的运动矢 量传递给下一层,直至在最底层获得最终的精确解。这种方法使得 LK 光流法能够应对大 的运动,从而适用于快速移动的场景。





### 3.1.3 特征点误匹配剔除

通过对相机捕获的图像数据进行处理可以成功完成特征检测与匹配,但在已经完成 匹配所获取到的多组特征点对中,误匹配的情况是非常常见的。错误地匹配特征点会对后 续结果产生直接的负面影响,所以需要剔除掉错误匹配的特征点。Random Sample Consensus 算法是由Fischler 等人提出的,其含义是指随机抽样的一致性,简称 RANSAC。 RANSAC 算法<sup>[63]</sup>是通过数学计算得到基于含有异常数据的样本数据集所建立模型的参数, 并据此生成有效样本数据的一种计算方法。可以理解为在匹配的数据集中随机选取样本, 并找到具有一致性的数据。RANSAC 算法通过将观测数据与参数化模型结合起来,不断 地循环测试数据,目标是找到一个3×3的最优单应性矩阵 **H**,并将需要的结果保留下来, 常用于移除错误匹配的特征点对。数学模型可用下式进行表示:

$$s\begin{bmatrix} x'\\y'\\1\end{bmatrix} = H\begin{bmatrix} x\\y\\1\end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13}\\h_{21} & h_{22} & h_{23}\\h_{31} & h_{32} & h_{33}\end{bmatrix} \begin{bmatrix} x\\y\\1\end{bmatrix}$$
(3-9)

式中:

$$(x, y)$$
——目标图像角点位置;

(x', y') ——场景图像角点位置;

s----尺度参数。

通过 RANSAC 算法确定的最佳单应性矩阵 H,必须确保 H 中匹配的特征点数量是 最多的。一般情况下取 h<sub>33</sub> =1,那么至少需要 4 个样本导出的 8 个线性方程对归一化后 H 中所剩的 8 个不确定元素进行求解,才能得到单应性矩阵 H,并且要求 4 个样本不能共 线。本小节采用 RANSAC 算法并通过以下流程对上文的特征点对进行错误剔除:首先从 上一小节匹配的样本数据集任意抽出 4 组特征点对,接下来计算该模型的单应性矩阵,并 使用此模型对所有数据进行了测试;经过不断的计算和迭代,找到特征点匹配数量最多的 H,即为最佳的参数模型。

## 3.2 滤波器设计

视觉图像处理完成之后,利用紧耦合的策略将特征信息与 IMU 测量结果进行数据融合推断系统状态,这样就形成了一个状态估计问题。处理状态估计问题的方法一般有两类: 一类是当前时刻的估计状态,然后用新的数据来更新这个状态,这种方式称为增量或渐进 滤波器;还有一类是将所有的数据都保存起来,再使用批量的方法来实现最优化,这种方 式相较于传统的滤波器更为有效,但也存在计算量大实时性不高的缺点。

本章主要研究基于滤波的系统状态估计问题,通过结合图像和惯性导航信息,采用设 计的卡尔曼滤波器来实现系统状态的最优预测与更新。相较于非线性优化方案,本方法展 现出更优越的实时性能。

#### 3.2.1 扩展卡尔曼滤波器

作为一种效率极高的最优估计算法,卡尔曼滤波得到了广泛的应用<sup>[64]</sup>。该算法通过构 建状态方程,并利用线性系统的输入和输出数据来观测,进而计算出状态向量的最优估计 值。鉴于系统输入中不可避免地存在噪声和干扰,寻找最优估计的过程实质上可以视为一 种滤波操作,用于减少这些不利因素对系统状态估计的影响。KF的实施流程主要可以划 分为预测和更新两个阶段。在当前时刻系统状态基础上,预测阶段利用状态方程计算下一 时刻的协方差矩阵以及状态向量;完成第一阶段后再进入状态更新阶段,接下来基于观测 方程得到卡尔曼滤波增益,并完成对预测结果的更新,实现了卡尔曼滤波的目标。

卡尔曼滤波器作为状态估计的一种重要方法,有着其使用范围的局限性,即卡尔曼滤波器仅适用于线性系统中。在系统采用线性高斯模型的情况下,滤波器可以提供最优估计。 但在实际系统中会出现二次方等各种程度的非线性问题,这些复杂因素给滤波带来了困 难,因此对非线性系统进行有效处理就变得十分重要。扩展卡尔曼滤波器(Extended Kalman Filter, EKF)<sup>[65]</sup>利用线性化技术将问题转化为近似线性系统。这种可行的策略是对非线性函 数进行泰勒展开,并简化该过程,仅保留一阶项而忽略二阶及更高阶的项。通过这种方式, 得到一个近似线性化的模型,随后可以利用 KF 对这个模型进行状态估计。在本文研究的 惯性/视觉数据融合状态估计问题中,运动方程和观测方程大部分情形下都是非线性函数。 所以本小节将从数学模型的角度来解释 EKF 的大致流程。

在非线性系统中,系统状态方程和测量方程由以下形式给出:

$$\begin{cases} \boldsymbol{x}_{k} = f\left(\boldsymbol{x}_{k-1}, \boldsymbol{u}_{k}\right) + \boldsymbol{w}_{k} \\ \boldsymbol{z}_{k} = h\left(\boldsymbol{x}_{k}\right) + \boldsymbol{v}_{k} \end{cases}$$
(3-10)

式中:

 $x_k$ ——系统状态;

$$u_k$$
——输入量;

- $w_k$ ——运动噪声;
- $z_k$ ——实际观测值与预测之间的残差;
- $\mathbf{v}_k$ ——观测噪声。

对于 $w_k$ 和 $v_k$ 均服从正态分布,可以表示为 $w_k \sim N(0, \mathbf{R})$ 、 $v_k \sim N(0, \mathbf{Q})$ ,其中 $\mathbf{R}$ 、 $\mathbf{Q}$ 为协方差矩阵。 $\hat{\mathbf{P}}_{k-1}$ 和 $\hat{\mathbf{x}}_{k-1}$ 表示k-1时刻协方差矩阵和系统状态向量的估计值。将式(3-10)进行泰勒展开并且保留一阶项,可以分别得到:

$$\begin{cases} \boldsymbol{x}_{k} \approx f\left(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_{k}\right) + \frac{\partial f}{\partial \boldsymbol{x}_{k-1}} \Big|_{\hat{\boldsymbol{x}}_{k-1}} \left(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}\right) + \boldsymbol{w}_{k} \\ z_{k} \approx h\left(\boldsymbol{x}_{k/k-1}\right) + \frac{\partial h}{\partial \boldsymbol{x}_{k}} \Big|_{\boldsymbol{x}_{k/k-1}} \left(\boldsymbol{x}_{k} - \boldsymbol{x}_{k/k-1}\right) + \boldsymbol{n}_{k} \end{cases}$$
(3-11)

基于卡尔曼滤波器预测公式,能够预测系统状态向量和协方差矩阵:

$$\boldsymbol{x}_{k/k-1} = f\left(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{u}_{k}\right) \tag{3-12}$$

$$\boldsymbol{P}_{k/k-1} = \boldsymbol{F} \hat{\boldsymbol{P}}_{k-1} \boldsymbol{F}^{T} + \boldsymbol{R}_{k}$$
(3-13)

上式中 $F = \frac{\partial f}{\partial x_{k-1}}\Big|_{\hat{x}_{k-1}}$ ,  $H = \frac{\partial h}{\partial x_k}\Big|_{x_{k/k-1}}$ 。

接下来再计算系统的卡尔曼增益,由于预测结果和观测结果与真实的状态存在不同 程度的偏差,卡尔曼增益包含两个信息。第一个信息相当于预测值和观测值之间的比例因 子,将预测误差与观测误差进行比较,系统更加偏向于信任误差较小的一方。第二个信息 起到了转换的作用,卡尔曼增益将更新量投影到了状态上。卡尔曼增益*K*<sub>k</sub>的计算公式如 下:

$$\boldsymbol{K}_{k} = \boldsymbol{P}_{k/k-1} \boldsymbol{H}^{T} \left( \boldsymbol{H} \boldsymbol{P}_{k/k-1} \boldsymbol{H}^{T} + \boldsymbol{Q}_{k} \right)^{-1}$$
(3-14)

得到 K<sub>k</sub>后,系统的状态和协方差矩阵由以下公式计算更新:

$$\hat{\boldsymbol{x}}_{k} = \hat{\boldsymbol{x}}_{k/k-1} + \boldsymbol{K}_{k} \left( \boldsymbol{z}_{k} - h(\hat{\boldsymbol{x}}_{k/k-1}) \right)$$
(3-15)

$$\hat{\boldsymbol{P}}_{k} = (\boldsymbol{I} - \boldsymbol{K}_{k}\boldsymbol{H})\hat{\boldsymbol{P}}_{k/k-1}$$
(3-16)

滤波过程如图 3-5 所示,至此 EKF 已经成功地预测和更新了非线性系统中的系统状态。依赖于模型的高斯噪声假设,EFK 在数据融合上有很广泛的应用,相对于优化的方法,基于 EKF 的方法精度虽然较低但是计算效率更高。由于本章的研究对象是非线性系统,因此对于本文研究的位姿估计部分采用 EKF 来对系统状态信息进行预测和更新。



图 3-5 滤波过程 Fig.3-5 Filtering Process

## 3.2.2 系统状态向量

为了进行 IMU 与视觉传感器数据融合,以实现水下机器人位姿的估计,本文选取的 系统状态向量为:

$$\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_{I}^{T} & \boldsymbol{X}_{C}^{T} \end{bmatrix}^{T}, \quad \boldsymbol{X} \in \boldsymbol{R}_{(23+7N)\times 1}$$
(3-17)

式中:

$$X_{I}$$
——IMU 相关状态量,  $X_{I} = \begin{bmatrix} {}^{I}_{G}q^{T} & b_{s}^{T} & {}^{G}v_{I}^{T} & b_{a}^{T} & {}^{I}_{G}q^{T} & {}^{I}_{D}v_{c}^{T} \end{bmatrix}^{T}$ ;  
 $X_{c}$ ——N 个相机位姿,  $X_{c} = \begin{bmatrix} X_{c_{1}}^{T} & \cdots & X_{c_{N}}^{T} \end{bmatrix}^{T}$ ,  $X_{c_{i}} = \begin{bmatrix} {}^{C}_{G}q^{T} & {}^{G}p_{c_{i}}^{T} \end{bmatrix}^{T}$ ;  
 ${}^{I}_{G}q$ ——{I} 系相对于{G} 系的姿态转换四元数;  
 ${}^{G}p_{I}$ ——{I} 系相对于{G} 系的应置;  
 ${}^{6}v_{I}$ ——{I] 系相对于{G} 系的速度;  
 $b_{s}$ ——陀螺仪的三轴零偏;  
 ${}^{L}_{g}$ ——相机坐标系{C} 到{I} 系的姿态信息;  
 ${}^{I}p_{c}$ ——{C} 系到{I} 系的位置信息;  
 ${}^{C}_{G}q$ ——相对于{G} 系的第*i*个相机姿态;  
 ${}^{G}p_{c_{i}}$ ——相对于{G} 系的第*i*个相机位置。

为确保滤波器的稳定性并避免协方差矩阵的奇异性,将状态量描述为误差的形态:即

一个量*x*的估计的误差为 $\tilde{x} = x - \hat{x}$ , *x*为状态量的真实值,  $\hat{x}$ 为状态量的估计值。对于四元数状态量采用了不同的误差定义, 如果 $\hat{q}$ 是四元数 $\bar{q}$ 的估计值, 则方向误差由误差四元数 $\delta \bar{q}$ 描述, 它由关系 $\bar{q} = \delta \bar{q} \otimes \hat{q}$ 定义, 符号  $\otimes$ 表示四元数乘法。误差四元数为:

$$\delta \overline{\boldsymbol{q}} \simeq \begin{bmatrix} \frac{1}{2} \delta \boldsymbol{\theta}^T & 1 \end{bmatrix}^T \tag{3-18}$$

微小的旋转 õq 直观地描述了真实姿态与估计姿态之间近乎重合的偏差。由于姿态具 有三个自由度,使用 õθ 这种微小的旋转来表示姿态误差,是一种最为精简和直接的方式。

IMU 有关误差状态量表示为:

$$\boldsymbol{X}_{I} = \begin{bmatrix} {}^{I}_{G} \boldsymbol{\partial} \boldsymbol{\theta}^{T} & \boldsymbol{b}_{g}^{T} & {}^{G}_{g} \tilde{\boldsymbol{v}}_{I}^{T} & \boldsymbol{b}_{a}^{T} & {}^{G}_{g} \tilde{\boldsymbol{p}}_{I}^{T} & {}^{I}_{C} \boldsymbol{\partial} \boldsymbol{\theta}^{T} & {}^{I}_{g} \tilde{\boldsymbol{p}}_{C}^{T} \end{bmatrix}^{I}$$
(3-19)

第*i*个相机位姿的误差表示为  $\tilde{X}_{C_i} = \begin{bmatrix} C_i \\ G \\ G \end{bmatrix}^T \quad G \\ \tilde{p}_{C_i}^T \end{bmatrix}^T$ 。

整个系统的误差状态向量为 $\boldsymbol{X} = \begin{bmatrix} \boldsymbol{X}_{I}^{T} & c_{I} \delta \boldsymbol{\theta}^{T} & {}^{G} \tilde{\boldsymbol{p}}_{C_{I}}^{T} & \cdots & c_{N} \delta \boldsymbol{\theta}^{T} & {}^{G} \tilde{\boldsymbol{p}}_{C_{N}}^{T} \end{bmatrix}^{T}$ 。

# 3.3 IMU 状态预测

#### 3.3.1 状态预测模型

(1) 连续时间系统建模

对于 IMU 的误差状态向量 X<sub>1</sub> 中的各个状态分别计算如下:

$$\begin{cases} {}_{G}{}^{I}\dot{\hat{\boldsymbol{g}}} = \frac{1}{2}\boldsymbol{\Omega}\left(\hat{\boldsymbol{\omega}}\right){}_{G}{}^{I}\hat{\boldsymbol{q}} \\ \dot{\hat{\boldsymbol{b}}}_{g} = \boldsymbol{0}_{3\times 1} \\ {}^{G}\dot{\hat{\boldsymbol{v}}}_{I} = \boldsymbol{C}\left({}_{G}{}^{I}\hat{\boldsymbol{q}}\right)^{T}\hat{\boldsymbol{a}} + {}^{G}\boldsymbol{g} \\ \dot{\hat{\boldsymbol{b}}}_{a} = \boldsymbol{0}_{3\times 1} \\ {}^{G}\dot{\hat{\boldsymbol{p}}}_{I} = {}^{G}\hat{\boldsymbol{v}}_{I} \\ {}_{C}{}^{I}\dot{\hat{\boldsymbol{q}}} = \boldsymbol{0}_{3\times 1} \\ {}^{I}\dot{\hat{\boldsymbol{p}}}_{C} = \boldsymbol{0}_{3\times 1} \end{cases}$$
(3-20)

式中:

 $C({}_{G}^{I}\hat{q})$ ——{G}系到{I}的姿态转换矩阵;

 $\boldsymbol{\omega}_{m}$ ——陀螺仪的测量值;

 $a_m$ ——加速度计的测量值;

 $\hat{\boldsymbol{\omega}}$  ——IMU 角速度估计值,  $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\boldsymbol{b}}_g$ ;

 $\hat{a}$  ——IMU 加速度估计值,  $\hat{a} = a_m - \hat{b}_a$ .

IMU 误差状态的线性化连续时间模型为:

$$\tilde{\boldsymbol{X}}_{I} = \boldsymbol{F} \boldsymbol{X}_{I} + \boldsymbol{G} \boldsymbol{n}_{I}, \quad \boldsymbol{X}_{I} \in \boldsymbol{R}_{21 \times 1}$$
(3-21)

式中:

- F---状态转移矩阵;
- G——噪声转移矩阵;
- $\boldsymbol{n}_{I}$  ——系统噪声,  $\boldsymbol{n}_{I} = \begin{bmatrix} \boldsymbol{n}_{g}^{T} & \boldsymbol{n}_{wg}^{T} & \boldsymbol{n}_{a}^{T} & \boldsymbol{n}_{wa}^{T} \end{bmatrix}^{T} \in R_{12 \times 1};$

 $n_g$  —— 陀螺仪测量值的噪声数据;

 $n_a$ ——加速度计测量值的噪声数据;

- $n_{wg}$  ——陀螺仪测量偏差的随机游走数据, $n_{wg} = \dot{b}_{g}$ ;
- $\boldsymbol{n}_{wa}$  ——加速度计测量偏差的随机游走数据:  $\boldsymbol{n}_{wa} = \dot{\boldsymbol{b}}_a$ 。

*n*<sub>1</sub>中的各项数据均为零均值高斯白噪声,且各个时刻是相互独立的,所以连续时间系统噪声协方差矩阵为:

$$\boldsymbol{Q}_{12\times12} = E \Big[ \boldsymbol{n}_{I} (t + \Delta t) \boldsymbol{n}_{I}^{T} (t) \Big] = \begin{bmatrix} \sigma_{g}^{2} \cdot \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \sigma_{\omega g}^{2} \cdot \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \sigma_{a}^{2} \cdot \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \sigma_{\omega a}^{2} \cdot \boldsymbol{I}_{3\times3} \Big]$$

根据公式(3-18)和(3-19)推出的 IMU 运动模型,可以得到矩阵 F 和 G 的具体形式为:

$$F_{21\times21} = \begin{bmatrix} -\lfloor \hat{\boldsymbol{\omega}} \times \rfloor & -\boldsymbol{I}_{3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ -\boldsymbol{C} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{q} \end{pmatrix}^{T} \lfloor \boldsymbol{\hat{\boldsymbol{a}}} \times \rfloor & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times$$

$$\boldsymbol{G}_{21\times12} = \begin{bmatrix} -\boldsymbol{I}_{3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & -\boldsymbol{C} \begin{pmatrix} \boldsymbol{I} & \boldsymbol{\hat{q}} \end{pmatrix}^{T} & \boldsymbol{0}_{3\times3} \\ \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times3} \end{bmatrix}$$

(2) 离散时间系统建模

由于 IMU 测量值为离散量, IMU 以周期T 对信号 $\boldsymbol{\omega}_m$ 和 $\boldsymbol{a}_m$ 进行采样,这些测量结果 用于状态传播。IMU 姿态 $_{G}^{I}\hat{\boldsymbol{q}}(t)$ 、位置 $^{G}\hat{\boldsymbol{p}}_{I}$ 与速度 $^{G}\hat{\boldsymbol{v}}_{I}$ 由 2.2.1 小节和 2.2.2 小节递推得到。 根据连续时间下的系统模型,可以推导得到状态方程(3-21)的离散形式如下:

$$\boldsymbol{X}_{I}(t+\Delta t) = \boldsymbol{\Phi}(t+\Delta t,t) \boldsymbol{X}_{I}(t) + \int_{t}^{t+\Delta t} \boldsymbol{\Phi}(t+\Delta t,\tau) \boldsymbol{G}(\tau) \boldsymbol{n}(\tau) d\tau \qquad (3-22)$$

利用泰勒公式展开并进一步做离散化,可以得到k时刻的噪声协方差矩阵 $Q_k$ 以及 IMU 从k时刻到k+1时刻的状态转移矩阵 $\Phi_k$ 如下:

$$\boldsymbol{\Phi}_{k} = \boldsymbol{\Phi}\left(t + \Delta t, t\right) = e^{\int_{t}^{t + \Delta t} \boldsymbol{F}(\tau) d\tau} = \boldsymbol{I} + \boldsymbol{F} \Delta t + \frac{1}{2!} \left(\boldsymbol{F} \Delta t\right)^{2} + \dots$$

$$\boldsymbol{Q}_{k} = \int_{t}^{t+\Delta t} \boldsymbol{\Phi}(t+\Delta t,\tau) \boldsymbol{G}(\tau) \boldsymbol{Q} \boldsymbol{G}^{T}(\tau) \boldsymbol{\Phi}^{T}(t+\Delta t,\tau) d\tau \approx \boldsymbol{\Phi}_{k} \boldsymbol{G} \boldsymbol{Q} \boldsymbol{G}^{T} \boldsymbol{\Phi}_{k}^{T} \Delta t$$

有了噪声协方差矩阵 $Q_k$ 和状态转移矩阵 $\Phi_k$ ,就可以计算在k时刻到k+1时刻的 IMU 状态协方差矩阵:

$$\boldsymbol{P}_{\boldsymbol{H}_{k+1|k}} = \boldsymbol{\Phi}_{\boldsymbol{k}} \boldsymbol{P}_{\boldsymbol{H}_{k|k}} \boldsymbol{\Phi}_{\boldsymbol{k}}^{T} + \boldsymbol{Q}_{\boldsymbol{k}}$$
(3-23)

描述系统状态时需要协方差来衡量状态的稳定性。当k--1的状态依据 IMU 递推到了 k时刻,对应状态的协方差一定是变大的,因为 IMU 是有误差的,引入了新的误差,协 方差自然变大,需要根据上一时刻的协方差结合 IMU 误差来更新新的协方差。式中  $P_{I_{lsk}}$ 表示k时刻 IMU 状态的协方差, $Q_k$ 表示k-1到k时刻 IMU 的误差对状态影响的协方差, 这个值是根据 IMU 的读数的噪声方差(标定而来)推到了状态方差。实际的协方差矩阵并 不是只有 IMU 部分,还有相机部分,所以关于相机部分还需要单独讨论。实际的协方差 矩阵如下:

$$\boldsymbol{P}_{k|k} = \begin{bmatrix} \boldsymbol{P}_{II_{k|k}} & \boldsymbol{P}_{IC_{k|k}} \\ \boldsymbol{P}_{IC_{k|k}}^T & \boldsymbol{P}_{CC_{k|k}} \end{bmatrix}$$
(3-24)

式中:

**P**<sub>II<sub>10</sub></sub> ——21×21的 IMU 协方差矩阵;

 $P_{CC_{kk}}$  ——  $6N \times 6N$  的相机位姿协方差矩阵;

**P**<sub>ICu</sub> ——IMU 与相机位姿的协方差相关矩阵。

递推阶段关于相机状态部分是不变的,此时k时刻到k+1时刻的协方差矩阵为:

$$\boldsymbol{P}_{k+1|k} = \begin{bmatrix} \boldsymbol{P}_{II_{k+1|k}} & \boldsymbol{P}_{IC_{k+1|k}} \\ \boldsymbol{P}_{IC_{k+1|k}}^{T} & \boldsymbol{P}_{CC_{k+1|k}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Phi}_{k} & \boldsymbol{\theta}_{3\times3} \\ \boldsymbol{\theta}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} \begin{bmatrix} \boldsymbol{P}_{II_{k|k}} & \boldsymbol{P}_{IC_{k|k}} \\ \boldsymbol{P}_{IC_{k|k}}^{T} & \boldsymbol{P}_{CC_{k|k}} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Phi}_{k}^{T} & \boldsymbol{\theta}_{3\times3} \\ \boldsymbol{\theta}_{3\times3} & \boldsymbol{I}_{3\times3} \end{bmatrix} + \begin{bmatrix} \boldsymbol{Q}_{k} & \boldsymbol{\theta}_{3\times3} \\ \boldsymbol{\theta}_{3\times3} & \boldsymbol{\theta}_{3\times3} \end{bmatrix}$$
(3-25)

### 3.3.2 状态增广

系统每次收到最新的相机位置和姿态信息时,滤波器会自动更新协方差矩阵和相机 的状态量。在记录新图像时,摄像机的姿态估计由 IMU 姿态估计计算,如下式所示:

$$\begin{cases} {}^{C}_{G}\hat{\bar{\boldsymbol{q}}} = {}^{C}_{I}\bar{\boldsymbol{q}} \otimes {}^{I}_{G}\hat{\bar{\boldsymbol{q}}} \\ {}^{G}_{G}\hat{\boldsymbol{p}}_{C} = {}^{G}\hat{\boldsymbol{p}}_{I} + \left(\boldsymbol{C}\left({}^{I}_{G}\hat{\bar{\boldsymbol{q}}}\right)\right)^{T}{}^{I}\boldsymbol{p}_{C} \end{cases}$$
(3-26)

式中 ${}^{c}_{I}\bar{q}$ 和 ${}^{l}p_{c}$ 分别是表示相机坐标系相对于 $\{I\}$ 系的旋转四元数和位置变换,两者都是已知的, C表示将四元数转换成对应的姿态转换矩阵。

将摄像机姿态估计追加到状态向量上,对 EKF 的协方差矩阵进行相应的增宽,状态 量增广后的协方差矩阵:

$$\boldsymbol{P}_{k|k} \leftarrow \begin{bmatrix} \boldsymbol{I}_{21+6N} \\ \boldsymbol{J} \end{bmatrix} \boldsymbol{P}_{k|k} \begin{bmatrix} \boldsymbol{I}_{21+6N} \\ \boldsymbol{J} \end{bmatrix}^{T} = \begin{bmatrix} \boldsymbol{P}_{k|k} & \boldsymbol{P}_{k|k} \boldsymbol{J}^{T} \\ \boldsymbol{J} \boldsymbol{P}_{k|k} & \boldsymbol{J} \boldsymbol{P}_{k|k} \boldsymbol{J}^{T} \end{bmatrix}$$

式中J为新增加的相机位姿状态量所对应的雅克比矩阵,可由方程(3-26)导出:

$$\boldsymbol{J}_{6\times(21+6N)} = \begin{bmatrix} \boldsymbol{C} \begin{pmatrix} I & \hat{\boldsymbol{q}} \\ G & \boldsymbol{q} \end{pmatrix}^T & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ - \begin{pmatrix} \boldsymbol{C} \begin{pmatrix} I & \hat{\boldsymbol{q}} \\ G & \boldsymbol{q} \end{pmatrix} \end{pmatrix}^T & \boldsymbol{I} & \boldsymbol{p}_C \times \boldsymbol{J} & \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} & \begin{pmatrix} \boldsymbol{C} \begin{pmatrix} I & \hat{\boldsymbol{q}} \\ G & \boldsymbol{q} \end{pmatrix} \end{pmatrix}^T \end{bmatrix}$$

系统状态协方差矩阵更新流程如图 3-6 所示,每当新一帧 IMU 数据或相机信息到达时,都会对雅克比矩阵 J 进行相应的更新。







Fig.3-6 System State Covariance Matrix Update Process

### 3.4 相机状态更新

#### 3.4.1 状态观测模型

假设在 $M_j$ 个相机位姿 $\begin{bmatrix} c_i q^T & c_j p_{c_i}^T \end{bmatrix}$ 中,同一个特征点 $f_j$ 被连续观测到, $i \in S_j$ ,在每个相机位姿下,该特征点都由模型描述:

$$\boldsymbol{z}_{j,i} = \frac{1}{z_j^i} \begin{bmatrix} \boldsymbol{x}_j^i \\ \boldsymbol{y}_j^i \end{bmatrix} + \boldsymbol{n}_{j,i} \quad , \quad i \in S_j$$
(3-27)

式中 $n_{j,i}$ 为相机i时刻观察第j个特征点时的 $2 \times 1$ 维图像噪声向量。在相机帧中表示的特征 位置 $c_i p_i$ 为:

$${}^{C_i}\boldsymbol{p}_j = \begin{bmatrix} x_j^i \\ y_j^i \\ z_j^i \end{bmatrix} = \boldsymbol{C} \begin{pmatrix} C_i \overline{\boldsymbol{q}} \end{pmatrix} \begin{pmatrix} {}^{G}\boldsymbol{p}_{f_j} - {}^{G}\boldsymbol{p}_{C_i} \end{pmatrix}$$
(3-28)

在算法的第一步中,地面坐标系中特征位置<sup>*c*</sup> $p_{f_j}$ 是未知的。为了解决这个问题,采用 了最小二乘法来最小化误差,从而得到这些特征位置的估计值<sup>*c*</sup> $\hat{p}_{f_j}$ 。观测项为一系列归一 化平面上的点,误差为归一化平面上x、y方向上的误差。得到特征位置的估计后,计算 测量残差:

$$\mathbf{r}_{j,i} = \mathbf{z}_{j,i} - \hat{\mathbf{z}}_{j,i} \tag{3-29}$$

式中 $z_{j,i}$ 和 $\hat{z}_{j,i}$ 分别是第j个特征点在第i时刻相机位姿观察时的观测量和估计量。

对特征位置和摄像机姿态的估计进行线性化,可以将式(3-29)表示为一种线性形式:  $r_{j,i} \simeq H_{x_{j,i}} \tilde{X} + H_{j,i}{}^{G} \tilde{p}_{j} + n_{j,i}$  (3-30)

 $H_{j,i}$ 和 $H_{x_{j,i}}$ 分别是观测量关于特征位置和状态的雅克比矩阵,将该特征的所有 $M_{j}$ 个测量值的残差叠加,得到:

$$\boldsymbol{r}_{j} \simeq \boldsymbol{H}_{X_{j}} \tilde{\boldsymbol{X}} + \boldsymbol{H}_{j}^{~G} \tilde{\boldsymbol{p}}_{j} + \boldsymbol{n}_{j}$$
(3-31)

由于 EKF 用于状态估计,对于构建测量模型,根据一般形式 $r = H\tilde{X} + n$ 定义一个线性依赖于状态误差  $\tilde{X}$ 的残差r就足够了,噪声项必须为零均值白噪声且与状态误差不相关,才能应用 EKF 框架,其中 H 为测量雅可比矩阵。在式(3-31)中,误差<sup>*G*</sup> $\tilde{p}_{j}$ 与误差  $\tilde{X}$  是相关的,所以残差 $r_{j}$ 与一般形式不一致,这使得它不适合直接应用于 EKF 的测量更新过程中。为了解决这个问题,可以通过将 $r_{j}$ 投影到矩阵  $H_{j}$ 的左零空间 $V^{T}$ 得到 $\bar{r}_{j}$ ,重新定义的新残差与一般形式保持统一,即有:

### $\boldsymbol{V}^T\boldsymbol{H}_i = \boldsymbol{0}$

$$\overline{\boldsymbol{r}}_{j} = \boldsymbol{V}^{T} \boldsymbol{r}_{j} \simeq \boldsymbol{V}^{T} \boldsymbol{H}_{X_{j}} \widetilde{\boldsymbol{X}} + \boldsymbol{V}^{T} \boldsymbol{n}_{j} = \overline{\boldsymbol{H}}_{X_{j}} \widetilde{\boldsymbol{X}}_{j} + \overline{\boldsymbol{n}}_{j}$$

最后叠加处理所有特征点的观测方程,得到观测方程的最终形式:

$$\boldsymbol{r}_o = \boldsymbol{H}_X \boldsymbol{\tilde{X}} + \boldsymbol{n}_o \tag{3-32}$$

### 3.4.2 滤波器更新

当相机在机体上随 ROV 移动时,会不断观测到新的水下环境信息,同时为滤波器提 供新的位置和图像特征,从而使相机的协方差矩阵和状态向量的维度持续上升,导致计算 任务不断加重,最终可能会大大降低算法的执行效率。利用滑动窗口对相机的状态数量进 行管理是一种在后端数据融合中确保计算效率的策略,这种策略主要是通过保持一个用 于按时间序列存储相机状态的滑动窗口。限定了滑动窗口的长度对有限数量的相机状态 进行维护,还对每个相机状态下的特征点信息进行了相应的记录。接着对不同相机状态下 观测到的同一特征点建立几何约束,在此基础上建立了相应的观测模型。随着相机的移动, 滑动窗口中的相机状态也在持续更新,这使得能够实时估计系统的运动情况。滑动窗口的 引入确保了相机状态的数量始终被限制在一个预定的阈值范围内,避免了随着系统移动 而相机状态的持续增长,从而减少了算法所需的计算资源。

在依据前文所描述的特征点几何约束建立了观测模型之后,系统仅在满足特定预设 条件时,才会对观测数据进行更新处理。图 3-7 所示为触发观测更新示意图。 f<sup>\*</sup>标记的 是被跟踪到的特征点,而在滑动窗口中则维护了最近7个时刻的相机状态信息。



Fig.3-7 Sliding Window Triggers Observation Update Diagram

触发滤波器更新的第一种情况是特征点跟踪丢失。图中可以看出,在k时刻记录的相 机状态中,当前帧跟丢了之前帧中连续观测到的特征点1、2、4、5、6。此时滑动窗口内 观测到的丢失特征点的相机状态数量达到了最大值,对跟丢的特征点做三角化,当前建立 的约束更多,并用三角化成功解算得到的的特征点进行滤波器观测更新,这样用于更新的 三维特征点坐标和更新得到的位姿会更加准确。

第二种情形是同一特征点的连续跟踪长度达到设置的最大滑动窗口长度,同样会触 发观测更新。下一时刻滑动窗口内最初的相机状态会被挤掉,如果继续跟踪到此特征点, 接下来继续进行滤波器的观测更新。另外系统会根据当前时刻的移动状态决定是否进行 更新,如果判断当前相机的移动速度过快时,系统会保留当前时刻的相机状态但不再做观 测更新。

有了由特征点建立的观测模型,接下来对矩阵 $H_x$ 通过 QR 分解进行降维,来降低 EKF 更新的计算复杂度:

$$\boldsymbol{H}_{X} = \begin{bmatrix} \boldsymbol{Q}_{1} & \boldsymbol{Q}_{2} \end{bmatrix} \begin{bmatrix} \boldsymbol{T}_{H} \\ \boldsymbol{0} \end{bmatrix}$$
(3-33)

根据这个定义,将式(3-33)代入式(3-32)的观测方程中,得到:

$$\boldsymbol{r}_{o} = \begin{bmatrix} \boldsymbol{Q}_{1} & \boldsymbol{Q}_{2} \end{bmatrix} \begin{bmatrix} \boldsymbol{T}_{H} \\ \boldsymbol{0} \end{bmatrix} \tilde{\boldsymbol{X}} + \boldsymbol{n}_{o}$$
(3-34)

$$\begin{bmatrix} \boldsymbol{Q}_{1}^{T}\boldsymbol{r}_{o} \\ \boldsymbol{Q}_{2}^{T}\boldsymbol{r}_{o} \end{bmatrix} = \begin{bmatrix} \boldsymbol{T}_{H} \\ \boldsymbol{0} \end{bmatrix} \tilde{\boldsymbol{X}} + \begin{bmatrix} \boldsymbol{Q}_{1}^{T}\boldsymbol{n}_{o} \\ \boldsymbol{Q}_{2}^{T}\boldsymbol{n}_{o} \end{bmatrix}$$
(3-35)

残余的
$$Q_2^T r_o$$
只是噪声,完全丢弃。为了更新 EKF,使用以下残差进行计算::  
 $r_n = Q_1^T r_o = T_H \tilde{X} + n_n$  (3-36)

式中 $\boldsymbol{n}_n = \boldsymbol{Q}_1^T \boldsymbol{n}_o$ 为噪声向量,其协方差矩阵为 $\boldsymbol{R}_n = \boldsymbol{Q}_1^T \boldsymbol{R}_o \boldsymbol{Q}_1 = \sigma_{im}^2 \boldsymbol{I}_r$ ,r为 $\boldsymbol{Q}_1$ 中的列数。 EKF 更新通过计算卡尔曼增益进行:

$$\boldsymbol{K} = \boldsymbol{P}\boldsymbol{T}_{\boldsymbol{H}}^{T} \left( \boldsymbol{T}_{\boldsymbol{H}} \boldsymbol{P} \boldsymbol{T}_{\boldsymbol{H}}^{T} + \boldsymbol{R}_{n} \right)^{-1}$$
(3-37)

而状态的修正是由以下向量给出的:

$$\Delta X = Kr_n \tag{3-38}$$

最后,状态协方差矩阵更新为:

$$\boldsymbol{P}_{k+1|k+1} = \left(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{T}_{\boldsymbol{H}}\right)\boldsymbol{P}_{k+1|k}\left(\boldsymbol{I} - \boldsymbol{K}\boldsymbol{T}_{\boldsymbol{H}}\right)^{T} + \boldsymbol{K}\boldsymbol{R}_{n}\boldsymbol{K}^{T}$$
(3-39)

# 3.5 组合导航定位算法仿真分析

在实际采样过程中,惯性测量单元 IMU 采样频率可以达到几百赫兹,而相机的采样 频率一般比较低,这会导致两者采样数据不一致的问题。在实际应用中,需要在每个图像 时刻对相机位姿进行求解。为了解决此问题,需要对邻帧间 IMU 的测量数据进行累积, 以得到邻帧间相机位姿和速度之间的关系。然而,由于 IMU 采样频率很高,每次更新所 有状态会导致数据量非常大,从而使计算复杂度非常高。因此,预积分是解决这个问题的 一种方法。预积分是指通过将 IMU 测量数据与时间相乘并进行离散化处理,可以将相邻 帧之间的 IMU 状态积分为一个单一的值,从而减少计算量和存储开销,同时保持准确性。 预积分的关键在于准确计算旋转矩阵、速度和位移增量。计算旋转矩阵时,需要使用 四元数或旋转向量来表示姿态状态,并通过 IMU 角速度计数据的积分来计算姿态状态的 变化。计算速度和位移增量时,则需要使用 IMU 加速度计数据来估计运动状态,并将其 积分得到速度和位移增量。IMU 预积分的基本思想是将 IMU 的测量值离散化后进行积分, 得到相邻时间步长之间的运动量的近似值。

选择基蒂(KITTI)公开数据集<sup>[66]</sup>进行仿真分析验证算法性能,数据集采集使用的驾驶 车上装有惯性导航系统、激光扫描仪、2个灰度相机、2个彩色相机和4个变焦镜头等多 个传感器,数据采样频率为10Hz。本文算法对惯导和视觉信息进行融合,惯导数据使用 KITTI数据集的惯性导航系统采集的IMU数据,视觉信息使用1个灰度相机提供的数据。 首先进行特征点提取并利用光流法跟踪特征,将图像数据转化为特征信息。

在评估算法的定位精度时,一般会重点关注两个核心指标:相对位置误差(Relative Pose Error, RPE)和绝对轨迹误差(Absolute Trajectory Error, ATE)。后者是通过比较实际轨 迹真值与估计结果来确定定位算法绝对误差的。均方根误差(Root Mean Square Error, RMSE)一般会来用作输出结果评估的准则,通过以下公式计算:

$$RMSE = \sqrt{\frac{1}{m} \sum_{i=1}^{m} (y_r(i) - y(i))^2}$$
(3-40)

式中 y<sub>r</sub>(i)为路径真值, y(i)为定位算法仿真输出值。本文将 RMSE 作为衡量定位算法 效果的第一种指标,将平均归一化估计误差作为第二种指标。

在 KITTI 上选取编号为 0001 和 0036 的数据集进行仿真,对比结果如图 3-8 所示。绿色曲线为真实轨迹,红色曲线是 SINS 解算轨迹,蓝色曲线表示 MSCKF 算法解算轨迹, 黑色圆圈为相机观测到的轨迹附近的特征点位置。







图 3-9 和图 3-10 分别为 MSCKF 从数据集 0001 和 0036 解算得到的位置与姿态角在 三轴方向上的误差,红色虚线为 3σ 曲线。表 3-1 记录了 MSCKF 与 SINS 的解算误差。



(a) xyz 三方向位置估计误差

(b) φθψ 三方向姿态估计误差



表 3-1 SINS 与 MSCKF 的解算误	差对比
Tab.3-1 Comparison of Solving Errors betwee	en SINS and MSCKF

<b>沒</b> 关	0001 数据集		0036 孝	0036 数据集	
庆左	SINS	MSCKF	SINS	MSCKF	
位置 ARMSE/m	0.784026	0.371095	0.300790	0.295535	
姿态 ARMSE/(°)	0.002625	0.009572	0.009192	0.007307	
位置误差/m	2.532430	1.104793	1.431185	0.596669	

结合仿真结果轨迹图 3-8 和误差对比表 3-1, SINS 和 MSCKF 都能够完整跟踪真实轨迹,并对特征点的三维坐标进行解算。与 SINS 相比,MSCKF 对系统位姿的估计更接近真实值,估计结果的位置 ARMSE 和归一化位置误差均明显小于 SINS 的解算结果。对数据集中这两段路径进行跟踪,结果显示 SINS 在终点的归一化位置误差高达 MSCKF 的两倍以上。随着系统长时间运动,SINS 的累计误差越来越大,估计结果的可靠性越来越低,

不能单独用于性能要求较高的实际平台。从图 3-9 和 3-10 可以看出, MSCKF 解算的位姿 误差符合 3σ 准则,因此 MSCKF 实现定位解算具有可行性,这也说明了视觉与惯导数据 融合后的结果明显优于单独的惯性导航。

(1) MSCKF 与 SWF 算法对比

滑动窗口滤波(Sliding Window Filter, SWF)<sup>[67]</sup>通过优化的方法估计全局的运动,其中 的局部优化问题一般会当作一个非线性最小二乘问题进行处理,选取高斯牛顿法(Gauss-Newton, GN)<sup>[68]</sup>优化在*K*个位姿的滑动窗口上观察*M*个特征,或者采取列文伯格一马夸尔 特法(Levenberg-Marquardt, LM)<sup>[69]</sup>计算。SWF 采用固定长度滑动窗口的策略保留了历史信 息,在滑动窗口中引入新的状态后,首先会移除窗口里最旧的状态,并对历史信息边缘处 理,随后处理的结果会作为一个先验条件被纳入下一轮的优化流程中。SWF 将特征位置 估计为滤波器状态本身的一部分,而 MSCKF 在单独的过程中优化特征位置,而不将其包 括在滤波器状态中。

采用编号 0001 的数据集进行仿真对比, SINS、MSCKF 与 SWF 解算的位置均方根误 差曲线如图 3-11 所示。SINS、MSCKF 与 SWF 解算结果对比见表 3-2。





	表 3-2	SINS	MSCKI	F与SW	/F 解算纟	结果对比		
Tab	3-2 Cot	nnarisoi	n of the I	Results	of SINS	MSCKF	and S	WF

	SINS	MSCKF	SWF		
位置 ARMSE/m	0.719718	0.335283	0.337230		
姿态 ARMSE/(°)	0.002509	0.009920	0.011263		
解算时间/s	/	10.892667	238.652496		

从图 3-11 可以观察到, MSCKF 和 SWF 两种视觉/惯导融合算法计算出的轨迹更接近 实际的运动轨迹, SINS 解算的误差最大。但由于 SWF 采用了优化的方法, 其解算时间远 高于 MSCKF, 不适用于对计算量和实时性有要求的水下移动平台。另外, 在文献[70]的 研究中, 调整滑动窗口的尺寸可能会对 SWF 算法的解算精度造成不良影响, 特别是当窗 口大小被设定为大约 20 帧时, 系统的解算精度达到最优。随后, 随着滑动窗口尺寸的持 续扩大, SWF 解算所需的时间也显著延长, 因此如何在解算精度和时效性之间找到一个 平衡点, 需要根据特定定位环境的具体需求来做出选择。

(2) 特征跟踪长度对 MSCKF 算法精度的影响

在用 MSCKF 算法分析同一组实验数据时,通过调整不同的最大特征跟踪长度,从而获取了与之相应的计算结果,见表 3-3。图 3-12 展示了在不同特征跟踪长度情况下 MSCKF 计算出的位置均方根误差的曲线图。

Tab. 5-5 Influence of Feature Tracking Length on MISCKI					
合 业			特征跟踪长度/	帧	
	10~20	10~30	10~40	10~50	10~Inf
位置 ARMSE/m	0.348542	0.307329	0.304175	0.310976	0.335283
姿态 ARMSE/(°)	0.017214	0.011165	0.008845	0.008720	0.009920
位置误差/m	1.1311060	1.035847	0.954047	0.952549	1.015345
解算时间/s	10.741528	9.430771	9.630112	9.705730	6.361847

表 3-3 特征跟踪长度对 MSCKF 的影响 Tab.3-3 Influence of Feature Tracking Length on MSCKF



图 3-12 不同特征跟踪长度 MSCKF 解算的位置 RMSE Fig.3-12 Position RMSE Calculated by MSCKF with Different Feature Tracking Length

在表 3-3 中需要注意的是, Inf 代表了最大特征跟踪长度无限大,即当某特征点连续 出现的长度大于设置的最小跟踪长度时,开始对这个特征点进行追踪,直至跟踪丢失。可 以观察到,随着特征追踪长度的增长,MSCKF 的解算准确性得到了提升,同时解算所需 的时间也有所缩短。通过与表 3-2 的比较,可以明显观察到 MSCKF 的解算速度明显超过 了 SWF。文献[71]探讨了特征密度如何影响 MSCKF 的精确度,并选择了不同数量的特征 点进行算法的求解。当特征点数量增多时,MSCKF 的算法精度会随着特征点数量的增长 而上升。

# 3.6本章小结

本章的主要内容是研究基于滤波技术的视觉/惯导信息融合方法。根据紧耦合的融合 策略,首先在图像处理部分进行特征点检测、匹配跟踪与误匹配剔除,得到视觉图像的特 征点信息。随后详细介绍了本次研究中使用的扩展卡尔曼滤波器,选取了系统状态向量并 基于系统的运动模型和观测模型进行滤波器的设计。接下来深入探讨了包含系统状态预 测、状态增广以及滤波器观测更新在内的位姿解算方案。最后,根据上述位姿解算方案利 用数据集对定位算法进行仿真,并与 SWF 算法进行了对比分析。

# 4 组合导航定位系统的位姿修正方法研究

对于本文研究的视觉与惯导的组合导航定位系统,上一章已经进行传感器数据融合 得到了位姿估计结果。尽管对 IMU 和视觉传感器建立了相应的测量模型并进行了标定, 标定结果和测量模型的准确性都会直接影响最终的估计结果,但是由于各种因素影响导 致传感器测量结果依然存在误差。随着系统的长时间运行,算法解算的误差会不断累积, 导致输出的位姿结果可靠性逐渐降低。同样从仿真分析可以看出,虽然上文的融合方案已 经尽可能得到优于融合前的结果,但算法解算结果与真实值仍不可避免地存在误差。因此 本章主要研究组合导航定位系统的位姿修正方法,以减小位姿估计结果的误差。

针对系统运动估计存在的累积误差,本章将根据两种不同情况进行纠正。如果机体在 运动过程中经过了历史位置,即会出现视觉回环,因此可以利用回环检测对位姿进行校正。 对于运动机体不经过历史位置的情况,由于无法检测到视觉回环导致回环校正失效,需要 利用已知标志点进行外环位姿修正。

# 4.1 回环检测

### 4.1.1 回环检测简介

回环检测的核心问题是通过图像信息高效检测机体是否经过了相同的位置,如果机体到达了历史位置就可以检测到视觉回环,这样就能提供更为有效的信息进行融合算法的后端优化,对带有累积偏差的位姿估计结果进行修正使其更接近真实值,从而提升整个导航定位系统的精度和稳健性。另外回环系统在纯视觉导航中发挥了很大的作用,可以用来衡量估算出的轨迹和长期使用的地图是否正确,这对整个系统来说是非常重要的。有些时候将带有回环检测的视觉导航系统称为 SLAM,可以进行后端的全局优化。只有前端而不带有回环检测的视觉导航系统,只能叫做 VO。

图 4-1 所示为回环检测示意图,矩形块表示不同相机位姿下的关键帧,帧 j 与帧 i 构成了视觉回环。通常在提及视觉回环检测时,最直接的思路是将每两幅图像的像素直接相减,这种暴力的检测方法准确率和效率都比较低,计算量巨大,要想得到准确的位姿和地图是不可靠的。第二种思路是将每两张图片进行特征匹配,然后根据特征点匹配正确的数量来确定两个图像之间是否存在关联。该思路的问题在于检测数量太大,无法应用于实时系统。还有一种思路是通过任意选取历史数据来进行回环检测,这种方式能够保持恒定时间的计算量,但当帧数逐步增加时,这种不加思考试探的方式抽到回环的可能性大大降低,从而降低检测效率。



图 4-1 回环检测示意图 Fig.4-1 Loop Detection Diagram

### 4.1.2 水下视觉字典

为了完成视觉回环的高效检测,本文使用基于词袋(Bag of Words, BOW)的模型来执行回环的检测工作,本节首先介绍词袋法的相关原理。词袋模型将出现在图片上的"花、马、船"等特征称为单词,每张水下图像都可以由多个单词共同描述,这些单词共同构成了一个专门用于描述水下场景的视觉词典。

字典可以被视为单词的集合,聚类问题与字典生成过程有许多相似之处。通过对大量的图像提取的 *N* 个特征来构建一个大小为 *k* 的字典,常用的构建词典的方法为 K-means 算法<sup>[72]</sup>,该算法通过建立 *k* 叉树进行特征聚类建立字典。 *k* 叉树字典图如图 4-2 所示。

(1) 首先, 需要确定字典的分叉数(即每个节点的子节点数)以及树的深度d;

(2) 接下来,从数据集中提取出大量的特征点,这些特征点是后续分析的基础;

(3) 对于提取出的每一个特征点,计算其对应的描述子,并将这些描述子收集到一个数组中,以便后续处理;

(4) 开始第一层聚类过程。在这一步中,随机选择数组中的一部分描述子作为初始的 聚类中心。然后,计算数组中剩余的描述子与这些聚类中心的距离,根据距离的大小将描 述子分配给最近的聚类中心。对于每个聚类中心,统计其所属描述子在某个特定位置上为 1 的数量。如果这个数量超过了设定的阈值,则该位置在新的聚类中心上被标记为1,否 则为0。通过这个过程,得到了新的聚类中心,并重复上述步骤,直到聚类中心不再发生 变化;

(5) 当第一层聚类完成后,开始递归地创建剩余深度的字典树。这个过程将重复进行, 直到达到预先设定的树的深度*d*。最终,将得到一个完整的、深度为*d*的叉树字典。

图 4-2 所示的一个 k 分支的树,其深度为 d,根据以上步骤在叶子层上成功地构造了 k<sup>d</sup> 个单词,而树状结构中的中间节点仅在需要快速搜索时才会使用。在寻找与特定特征 相匹配的单词时,通过 d 次比较,将待查询项与每个中间节点的聚类中心进行匹配,从而迅速定位到最终的单词,这种查找方式确保了算法具有对数级别的查找效率。



# 4.1.3 相似度计算

在成功构建 k 叉树的视觉字典之后,对于任何一个给定的特征在字典树查找得到一个 对应的单词,进而比较两张图像之间的相似度。若直接进行单词检索不能充分利用特征间 的差异性,因此词袋模型使用 TF-IDF 算法<sup>[73]</sup>来进行文本的查找。TF 部分的核心观点在 于,当某特征词汇在图像中频繁出现时,它的区分能力会显著提高; IDF 意味着一个单词 在字典里出现的次数越少,它在进行图像分类时的区分能力就越强。

假设某叶子结点 w<sub>i</sub> 包含数量为n<sub>i</sub>的特征,而词典内包含总数为n 的特征,将两者的比值作为该单词的 IDF 部分:

$$IDF_i = \log \frac{n}{n_i} \tag{3-41}$$

某个特征在单个图像中出现的频率作为 TF 部分。对于单个图像,若观察到n<sub>i</sub>次叶子结点w<sub>i</sub>,而该叶子结点在所有图像中的出现次数为n,那么 TF 为:

$$TF_i = \frac{n_i}{n} \tag{3-42}$$

记该叶子结点的权重为 $\eta_i$ ,等于 TF 与 IDF 之积:

$$\eta_i = TF_i \times IDF_i \tag{3-43}$$

在考虑到权重因素后,可以利用单词和权重对任意图像P进行表示:

$$P = \{ (w_1, \eta_1), (w_2, \eta_2), \cdots, (w_n, \eta_n) \} = v_P$$
(3-44)

两幅图像比较时,计算图像 $P_A$ 与 $P_B$ 之间的相似度:

$$s(v_{P_{A}} - v_{P_{B}}) = 2\sum_{i=1}^{n} \left( \left| v_{P_{A_{i}}} \right| + \left| v_{P_{B_{i}}} \right| - \left| v_{P_{A_{i}}} - v_{P_{B_{i}}} \right| \right)$$
(3-45)

在根据公式(3-45)计算了两张图像之间的相似性后,采用一个预设的阈值来评估这一 计算出的相似度,进而判断这两张图像是否满足条件以形成一个闭环。

# 4.2 水下回环识别与改进研究

### 4.2.1 回环候选帧筛选

在本文定位算法研究的后端传感器数据融合部分,使用了滑动窗口的机制记录相机 采集到的图像信息。由于相机的采集频率高产生的数据量庞大,若直接将采集的图像信息 做为回环候选帧提供给回环检测部分,会严重降低回环检测的效率。为了发挥回环帧的有 效信息提高检测效率,本小节通过回环帧筛选策略选出关键帧用于回环检测。

(1)在实际的检测流程中,为了进行回环检测,所使用的图像帧应当更为稀疏,它们 之间应该存在差异,同时还需要能够覆盖整个检测环境。假设第1帧和第n帧检测到了回 环,对于优化是有帮助的,若回环候选帧过于密集的话接下来的n+1和n+2帧都会产生回 环,所带来的帮助就不大了,甚至有些多余。这种情况下,把相近的回环聚成一类,每5 帧开始筛选一次关键帧,如果不符合关键帧的标准,那么会进一步确定下一帧的图像帧。

(2) 为了确保图像帧能够提供充足的信息来判断是否构成回环,需要确定特征点数量 的最小值,并确保在图像帧中追踪到的特征点数目不会少于这一设定的阈值。也就是说, 回环候选帧与当前帧之间需要存在大量的匹配点。若匹配点数量显著不足,则应直接排除 当前的回环候选帧,以避免不准确的回环检测结果。

(3)为了避免系统过快的运动导致关键帧信息的重复,需要确定位移和角度的最小值,即当前图像帧与前一个关键帧之间的平移和旋转动作必须不低于预设的阈值,从而防止 出现关键帧信息冗余的情况。

### 4.2.2 几何约束及改进

词袋中的回环检测方法完全基于图像的外观,而未考虑到任何几何细节,这使得外观 相近的图片容易被误判为回环。此外,由于词袋并不关心单词的排列顺序,而是专注于单 词是否存在的表达形式,这更容易导致感知上的偏差。

在回环检测的过程中,增加了一个重要的步骤,即考虑特征点间的几何约束。这种几 何约束指的是特征点之间在图像中的空间布局关系。为了确保回环候选帧的有效性,其特 征点间的几何约束必须与当前帧中的保持一致。通过这一约束,能够显著减少误匹配的数 量,提高回环检测的准确性。特征点的几何约束可以表现为多种形式,例如两点连线后线 与线之间的夹角、点与点之间的距离或者由多个点组成的特定形状等。图 4-3 为几何约束 示意图。在本文中,引入了文献[74]所提出的仿射不变量作为特征点间几何约束的一种度 量方式,以进一步优化回环检测的性能。

50



图 4-3 几何约束示意图 Fig.4-3 Geometric Constraint Diagram

记 $C = \{(p_1, q_1), (p_2, q_2), \dots, (p_k, q_k)\}$ 为当前帧和候选帧中匹配点的集合。考虑当前帧 中任意三个特征点的齐次坐标 $p_1 = (x_1, y_1, 1), p_2 = (x_2, y_2, 1), p_3 = (x_3, y_3, 1),$ 则由这三点组成 的三角形表示为:

$$TAB(p_1, p_2, p_3) = \frac{1}{2} \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix}$$
(3-46)

当前帧和候选帧的匹配点满足如下变换关系:

$$\begin{bmatrix} \overline{x}_n \\ \overline{y}_n \end{bmatrix} = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} x_n \\ y_n \end{bmatrix} + \begin{bmatrix} e \\ f \end{bmatrix}$$
(3-47)

式中:

 $(x_n, y_n)$ ——当前帧中某个特征点的图像坐标;  $(\bar{x}_n, \bar{y}_n)$ ——候选帧中匹配点的图像坐标。 则有:

$$TAB(q_1, q_2, q_3) = (ad - bc)TAB(p_1, p_2, p_3)$$
(3-48)

当两帧之间的匹配关系准确无误时,这两帧中由相应特征点组成的三角形,其面积的 比例应当保持相等。在图 4-3 中,对于当前帧中的任意一个特征点  $p_i$ ,可以随机选择两个 点  $p_u$ 和  $p_v$ ,并定义三角形的面积的绝对值表示为  $s_{uv} = |TAB(p_i, p_u, p_v)|$ 。类似地,在对应 的 另一 帧 中,与  $q_i$ 、  $q_u$ 、  $q_v$ 相匹 配 的 点 所 形 成 的 三 角 形 面 积 绝 对 值 表 示 为  $s'_{uv} = |TAB(q_i, q_u, q_v)|$ 。于是,有如下关系,面积比应当近似相等。

$$\frac{s_{12}}{s_{12}'} \approx \frac{s_{23}}{s_{23}'} \approx \dots \approx \frac{s_{r1}}{s_{r1}'}$$
(3-49)

定义集合  $S = \{s_{12}/s'_{12}, s_{23}/s'_{23}, \dots, s_{r_1}/s'_{r_1}\}$ ,在对集合进行排序后,选取位于集合中间 3/5 k 的元素来计算其方差,其中 k 是集合中元素的个数。若计算得出的方差小于预设的 某个阈值,那么可以认为当前点对是正确的匹配点。

### 4.3 基于视觉回环的位姿修正

图 4-1 中当前帧 j 与历史帧 i 之间建立了视觉回环,从而确认了两个候选的关键帧作为回环帧,记当前回环帧 j 的相机位姿为 $T_{\omega_2 \leftarrow j}$ ,而历史回环帧 i 的位姿为 $T_{\omega_1 \leftarrow i}$ 。由于系统在移动的过程中出现了累积误差,导致 i 帧和 j 帧所处的世界坐标系并不一致,坐标系已经发生了偏移。所以,确定坐标系 $\omega_1$ 与 $\omega_2$ 的相对位置成为了回环校正的核心目标,之后再对 j 帧位姿进行修正。坐标系 $\omega_1$ 与 $\omega_2$ 之间的相对位置关系可以通过下式进行计算:

$$\boldsymbol{T}_{\omega_1 \leftarrow \omega_2} = \boldsymbol{T}_{\omega_1 \leftarrow i} \cdot \boldsymbol{T}_{\omega_2 \leftarrow i}^{-1}$$
(3-50)

式中 $T_{\omega, \leftarrow i}$ 为已知量由定位部分提供, $T_{\omega, \leftarrow i}$ 可以由 $T_{\omega, \leftarrow i}$ 和回环帧与当前帧的相对位姿求得:

$$\boldsymbol{T}_{\omega_{2}\leftarrow i} = \boldsymbol{T}_{\omega_{2}\leftarrow j} \cdot \boldsymbol{T}_{j\leftarrow i}$$
(3-51)

式中,  $T_{i \leftarrow i}$  是基于第i 帧和第j 帧关键帧的图像特征点数据来进行计算的。

所有系统接收到的关键帧都会被存储整合,一旦发现视觉回环便开始修正处理。通过 IMU 的静态初始化,重力方向可以被精确确定,从而机体在运动中的 $\phi 和 \theta$ 能够被精准地 解算出来,因此仅有 $\psi$ 和三维位置会出现偏差。在构建的完整位姿图里,每两个关键帧一 起形成了边。在这里,不含回环帧的边则被称作序列边,而含有回环帧的边被称作回环边。 对于任意序列边,假设连接了 j 和 i两个关键帧,那么它们之间的偏航角 $\hat{\psi}_{ij}$ 和相对位置 $\hat{P}_{ij}^{i}$ 分别为:

$$\begin{cases} \hat{\boldsymbol{P}}_{ij}^{i} = \left(\hat{\boldsymbol{R}}_{i}^{w}\right)^{-1} \left(\hat{\boldsymbol{P}}_{j}^{w} - \hat{\boldsymbol{P}}_{i}^{w}\right) \\ \hat{\boldsymbol{\psi}}_{ij} = \hat{\boldsymbol{\psi}}_{j} - \hat{\boldsymbol{\psi}}_{i} \end{cases}$$
(3-52)

式中:

 $\hat{P}_{i}^{w}$  ——第i 帧相对于世界系发生的平移关系;

 $\hat{R}_{i}^{w}$ ——第i帧相对于世界系发生的旋转关系;

 $\hat{P}_{ii}^{i}$ ——第 j 帧相对第 i 帧的位置。

在发现 j 和 i 帧出现回环现象后,可以构建相应的误差方程:

$$\boldsymbol{r}_{i,j}\left(\boldsymbol{P}_{i}^{w},\boldsymbol{\psi}_{i},\boldsymbol{P}_{j}^{w},\boldsymbol{\psi}_{j}\right) = \begin{bmatrix} \boldsymbol{R}^{-1}\left(\boldsymbol{P}_{j}^{w}-\boldsymbol{P}_{i}^{w}\right)-\hat{\boldsymbol{P}}_{ij}\\ \boldsymbol{\psi}_{j}-\boldsymbol{\psi}_{i}-\hat{\boldsymbol{\psi}}_{ij} \end{bmatrix}$$
(3-53)

为了对位姿图中的边进行优化,通过 Ceres 非线性优化库最小化下式所示的损失函数:

$$\min_{\boldsymbol{P},\psi} \left\{ \sum_{(i,j)\in\mathcal{S}} \left\| \boldsymbol{r}_{i,j} \right\|^2 + \sum_{(i,j)\in L} \rho\left( \left\| \boldsymbol{r}_{i,j} \right\|^2 \right) \right\}$$
(3-54)

式中S是由所有的序列边构成的,而L则是由所有的回环边构成的,鲁棒核函数 $\rho(\cdot)$ 的功能是减少不正确匹配带来的干扰。

由于 KITTI 在数据采集过程中,车辆大多情况下往前开不会经过历史位置,也就不 会出现视觉回环,因此无法在上一章的基础上进行位姿修正减小累计误差。本小节使用无 人机采集的 EUROC 数据集对回环检测进行仿真分析,由于水下图像的清晰度不高,在回 环检测时容易出现很多错误的匹配,无法用于系统后端的位姿修正部分,改进后的回环检 测的结果如图 4-4 所示,MSCKF 算法估计的位姿结果在 ROS 环境显示,如图 4-5 所示。



图 4-4 回环检测结果 Fig.4-4 Loopback Test Result



图 4-5 ROS 仿真环境位姿显示 Fig.4-5 ROS Simulation Environment Pose Display

图 4-4 左右两幅图片都是满足约束条件筛选得到的关键帧,并通过词袋模型匹配成为 了回环候选帧,为了防止回环误判,根据上文的几何约束进一步检查这对回环候选帧,最 终匹配成功确认为回环帧,可以用于机体位姿修正。

# 4.4 基于已知标志点的位姿修正

#### 4.4.1 标志点检测

采用基于 MSCKF 的紧密耦合定位技术,可以将 IMU 的测量数据与相机的感知数据

结合起来,从而对 ROV 的实时位姿进行估计。这种定位方法在本质上属于开环技术,由于缺少闭环修正环节,如果 ROV 在定位时受到了显著的干扰,可能会导致滤波器存在累积误差,从而降低定位的准确性,甚至可能导致其发散。针对上述问题,本节基于上一章的位姿解算方案,着重于利用视觉图像获取水下已知位置的标志点信息,对 ROV 自身的位姿进行外环修正,这种方法有效防止了误差的累积,进而提升了整个定位系统的精确性。

ROV 搭载的相机由于像素有限,使得拍摄图像的清晰度受到一定影响,而机载处理器的计算能力同样受到限制,导致图像处理能力受限。本文对一种与 ROV 特性相匹配,具有高鲁棒性和低延迟的标志点检测技术进行研究。如图 4-6 所示的是标志点的检测示意图,系统接收增强后的水下图像作为输入,并期望得到标志区域的角点像素坐标。以下是具体检测步骤:

(1) 为了明确区分标志区域与背景,首先进行了二值化处理;

(2) 在二值化图像中,精确地提取出了标志区域的轮廓;

(3) 随后进一步从二值化图像中提取了角点信息;

(4) 为了筛选出真正的角点坐标,采用了非极大值抑制技术,成功地从标志区域内提 取了实际角点坐标。

为了优化计算效率并增强算法的稳定性,选择了一个矩形区域作为目标标志区域,并 设定该区域的四个角点作为待提取的关键标志点。当系统捕获到水下视觉信息后,首先通 过二值化操作将灰度图像中的背景与标志区域区分开来。接下来,从处理后的图像中精准 地提取出矩形轮廓,并计算出该轮廓的最小外接矩形,从而获取其角点数据。然而,由于 水下环境中标记物可能经历仿射变换,直接以最小外接矩形的角点作为标记物的角点并 不准确。因此,利用 Harris 角点检测法<sup>[75]</sup>对二值化后的图像进行角点识别,从而捕获到角 点的详细信息。由于角点可能会密集出现,进一步采用了非极大值抑制技术来筛选角点。 最终,选择了与矩形轮廓距离最短的四个 Harris 角点,并将它们作为标志物的四个关键特 征点。这一方法不仅降低了处理器的计算负担,还显著提高了算法的鲁棒性和准确性。



图 4-6 标志点检测 Fig.4-6 Mark Point Detection

#### 4.4.2 机体位姿修正

通过识别标志点,能够精准地确定它们在像素平面上的坐标位置。鉴于这4个标志点在 $\{G\}$ 系中的确切坐标是预先已知的,可以利用这一信息将问题转化为一个经典的 PnP 问题。随后结合标志点在像素坐标系与 $\{G\}$ 系之间的转换关系,可以精确地确定相机坐标系在 $\{G\}$ 系中的具体位置和姿态,可以确定相机坐标系在系中的位置和姿态,从而确定 IMU 在 $\{G\}$ 中的位置和姿态,并对内环估计的状态进行适当的修正。假设从标志点获得的相机坐标系当前时刻在 $\{G\}$ 系中的位姿为 $\begin{pmatrix} c \ q \ m \end{pmatrix}, \begin{pmatrix} c \ p \ m \end{pmatrix}$ ,当相机坐标系相对于 $\{I\}$ 系的位姿已知时,可以确定当前时刻 IMU 相对于 $\{G\}$ 系的位姿:

$${}^{I}_{G}\hat{\overline{q}}^{(m)} = {}^{I}_{C}\overline{\overline{q}} \otimes {}^{C}_{G}\hat{\overline{q}}^{(m)}$$

$$(4-1)$$

$${}^{G}\hat{\boldsymbol{p}}_{I}^{(m)} = {}^{G}\hat{\boldsymbol{p}}_{C}^{(m)} + \left(\boldsymbol{C}\left({}^{C}_{G}\hat{\boldsymbol{q}}^{(m)}\right)\right)^{T} {}^{C}\hat{\boldsymbol{p}}_{I}$$

$$(4-2)$$

$$\boldsymbol{z}_{I}^{(m)} = \begin{bmatrix} {}_{G}\boldsymbol{\hat{q}}^{(m)T} & {}_{G}\boldsymbol{\hat{p}}_{I}^{(m)T} \end{bmatrix}^{T}$$
(4-3)

基于特征点观测量的 IMU 状态量位姿残差观测量为:

$$\boldsymbol{r}^{(m)} = \boldsymbol{z}_{I}^{(m)} - \boldsymbol{z}_{I} = \begin{bmatrix} {}_{G} \tilde{\boldsymbol{\theta}}^{(m)T} & {}^{G} \tilde{\boldsymbol{p}}_{I}^{(m)T} \end{bmatrix}^{T}$$
(4-4)

建立与外环残差观测相关的观测方程:

$$\boldsymbol{r}^{(m)} = \boldsymbol{H}^{(m)} \tilde{\boldsymbol{X}} + \boldsymbol{n}^{(m)}$$
(4-5)

式中观测矩阵 $\boldsymbol{H}^{(m)} = \begin{bmatrix} \boldsymbol{I}_6 & \boldsymbol{0}_{6\times(6N+9)} \end{bmatrix}$ 。

观测矩阵因其稀疏性和较快的计算速度,可以直接应用于下面的运算中。修正后的增益矩阵、状态量以及协方差矩阵通过以下公式计算:

$$\boldsymbol{K}^{(m)} = \boldsymbol{P}\boldsymbol{H}^{(m)T} \left( \boldsymbol{H}^{(m)}\boldsymbol{P}\boldsymbol{H}^{(m)T} + \boldsymbol{R}_n \right)^{-1}$$
(4-6)

$$\Delta \boldsymbol{X} = \boldsymbol{K}^{(m)} \boldsymbol{r}^{(m)} \tag{4-7}$$

$$\boldsymbol{P}_{k+1|k+1}^{(m)} = \left(\boldsymbol{I} - \boldsymbol{K}^{(m)} \boldsymbol{H}^{(m)}\right) \boldsymbol{P}_{k+1|k+1} \left(\boldsymbol{I} - \boldsymbol{K}^{(m)} \boldsymbol{H}^{(m)}\right)^{T} + \boldsymbol{K}^{(m)} \boldsymbol{R}_{n} \boldsymbol{K}^{(m)T}$$
(4-8)

由于 MSCKF 在选择相关参数的初始值时非常敏感,这可能引发 MSCKF 无法收敛的问题。为了降低标志点修正对滤波器收敛的不良影响,在初始运行阶段,仅使用内环的 MSCKF 来进行定位。当其达到稳定的收敛状态后,再加入外环的修正流程来减少内环滤 波器的偏移。在 3.5 节仿真分析的基础之上进行外环修正,位置平均均方根误差对比结果 见表 4-1,修正轨迹如图 4-7 所示,位置 RMSE 对比曲线如图 4-8 所示。

表 4-1 修正前后位置 ARMSE 对比

	SINS	MSCKF	本文
位置 ARMSE/m	0.784026	0.371095	0.285488



图 4-7 轨迹修正

Fig.4-7 Trajectory Correction



图 4-8 修正位置均方根误差对比 Fig.4-8 Corrected Position RMSE Comparison

从图 4-7 可以看出,系统开始阶段没有设置标志点,位姿结果完全由内环 MSCKF 提供。在坐标(50m,-2.5m,0.55m)处放置了矩形块,通过图像处理检测到该标志点,轨迹开始向真实值靠近,图 4-8 所示的误差也明显有所降低。由表 4-1 可知,本文算法和 MSCKF 位置估计的误差均小于 SINS 解算的结果,本文算法对 MSCKF 进行外环位姿修正,位置 ARMSE 在 MSCKF 基础上降低了 23.07%,有效降低了位姿估计的累积误差。

## 4.5 本章小结

针对系统估计存在累积误差的问题,本章在运动机体经过同一位置和不经过同一位 置两种情况下,分别利用从摄像机图像检测到的视觉回环和位置已知的标志点,得到更接 近真实值的位姿信息并对融合算法进行外环位姿修正。本章首先探讨了回环检测的基本 原理,通过词袋模型建立了水下视觉字典进行相似度计算来评估两幅图像是否构成回环。 接下来详细描述了本次研究中回环检测和校正的工作原理,给出了回环候选帧筛选策略 并采用几何约束对改进的水下回环识别进行研究,然后归纳了算法处理流程并基于 EUROC 数据集进行仿真验证。最后阐述了标志点检测的过程对机体位姿进行修正,并在 上一章基础上进行了仿真分析,仿真结果表明该方法有效降低了位姿估计的累积误差。

# 5 水下试验与结果分析

# 5.1 引言

本章节主要是针对前两个章节关于视觉/惯导组合导航定位方法进行水下环境的试验 验证。前四章对水下视/惯导组合导航定位系统的整体架构进行介绍,并用数据集对各部 分进行了仿真工作与分析,然而仿真环境是理想情况下的,真实情况可能更为复杂,因此 有必要在实际环境中对所提出的方法进行测试。为了验证本文所提方案在真实水下环境 中的实现效果,本章将介绍水下试验的准备工作,包括 ROV 软硬件系统调试、实验平台 搭建、传感器数据获取与标定,并进行了试验方案设计,详细阐述本文方法在实际应用中 的验证效果。

## 5.2 实验平台搭建

#### 5.2.1 实验平台

为了进行水下实验,本研究的实验平台需满足搭载 IMU 和相机两个传感器的基本要求,同时要求操作简单易于调试,未来计划引入磁力计、DVL 等新的传感器,以支持课题的持续深入研究和拓展。在商用领域,常见的 UUV 平台主要聚焦于功能性表现,然而因其内部设计的封闭性特点在一定程度上限制了研究工作的深入进行。经过深入的市场调研和技术评估,最终决定将实验平台选定为由美国知名公司 BlueRobotics 开发的BlueROV2 Heavy 型号的开源水下机器人。ROV 正视图和俯视图如图 5-1 所示。该机器人是一种轻型六自由度观测级 ROV,长 457mm,宽 575mm,高 254mm,净重约 11kg,最大测试深度可达 300m。

![](_page_65_Picture_7.jpeg)

图 5-1 BlueROV2 Heavy 正视图和俯视图 Fig.5-1 BlueROV2 Heavy Front View and Top View

图 5-2 所示为 ROV 系统示意图。ROV 系统主要由地面控制站(QGroundControl, QGC)、

信号中继站和 ROV 本体三部分组成。ROV 本体主要包含电子水密舱、电池水密舱、4 个 水平推进器、4 个垂直推进器等。容量 18Ah 的可充电电池安装于电池舱提供电源,控制 器和传感器模块都安装于电子舱,下水前需要进行气密性检查。机器人尾部带有 100 米的 脐带缆与岸上电脑进行通讯以确保信号的稳定和可靠传输,在实际使用过程中脐带缆的 长度决定了 ROV 在水下 100 米以内进行运动。安装于岸上电脑端的 QGC 地面控制站提 供实时的视频流和传感器信息,实现对 ROV 的安全保障、任务计划和指令下达、数据处 理和分析,以及远程操控和监控等。

![](_page_66_Figure_2.jpeg)

图 5-2 ROV 系统示意图 Fig.5-2 ROV System Diagram

BlueROV2 Heavy 使用 ArduSub 控制固件,控制器包括 Pixhawk 和树莓派两个部分。 操作人员借助遥控手柄发送控制指令,这些指令通过无线传输方式被发送到 QGC。随后 QGC 利用 UDP 协议将接收到的指令信息转发给运行在树莓派上的 Mavproxy 进程。 Mavproxy 进程在接收到信息后,会进一步将指令发送至 Pixhawk 飞行控制器。最终, Pixhawk 根据接收到的指令对八个 T-200 执行单元进行精确控制,从而实现 ROV 六自由 度的运动。BlueROV2 Heavy 上搭载了摄像机,陀螺仪,加速度计等传感器。摄像头捕捉 的图像通过 Gstreamer 框架进行实时解析和编码,随后这些图像数据利用 UDP 协议发送 至 QGC 完成接收并显示。其他传感器数据以日志的方式进行存储。树莓派装有 Linux 系 统,用户可使用网页端的命令行界面进行操作,还可以进行参数设置、固件烧录、文件下 载等,方便与机器人进行交互。

### 5.2.2 传感器数据获取

安装于 ROV 本体电子水密舱的 IMU 和摄像头如图 5-3 所示。传感器数据流如图 5-4 所示。BlueROV2 Heavy 自带的 IMU 可获得较高精度的三轴加速度和三轴角速度,传感器数据存储在日志文件需要进行解析,更新频率最高为 50Hz, IMU 原始数据如图 5-5 所示。

#### 5 水下试验与结果分析

![](_page_67_Picture_1.jpeg)

图 5-3 摄像头和 IMU 实物安装图 Fig.5-3 Camera and IMU Physical Installation Diagram

![](_page_67_Figure_3.jpeg)

![](_page_67_Figure_4.jpeg)

Fig.5-4 Sensor Data Stream

![](_page_67_Figure_6.jpeg)

Fig.5-5 Raw Data Obtained by IMU

在视觉导航失效的情境下,高精度的加速度和角速度信息在短时间内对于限制机器 人定位累积误差尤为关键。在本文选择不使用 BlueROV 自带的 IMU,主要基于以下两点 考量: 首先,该 IMU 的更新频率相对较低,最高仅为 50Hz,可能无法满足对定位精度的 要求;其次,为了获取该 IMU 数据的时间戳,需要对源码进行改动,这一过程相对繁琐 且耗时。因此倾向于采用其他更高性能或更易于集成的 IMU 解决方案。在原有 BlueROV 基础上,本文增加了 JY901 模块,该模块是由维特智能公司生产的,集成了较高精度的陀 螺仪和加速度计,可以实时提供高达 200Hz 更新速率的三轴加速度和三轴角速度。

玻璃罩内的 USB 单目摄像头捕获了清晰的水下图像,这些图像数据通过脐带缆高效 地传输到岸边的电脑中。在 QGC 界面上,用户可以实时查看这些图像信息,如图 5-6 所 示。此外该摄像头配备了云台伺服系统,允许用户远程调整摄像头的拍摄角度,以便更全 面地观察和记录水下的情况。

![](_page_68_Picture_3.jpeg)

图 5-6 QGC 界面显示的图像信息 Fig.5-6 Image Information Displayed on the QGC Page

### 5.2.3 相机-IMU 标定

对系统运动估计时,需要考虑 IMU 测量模型中的噪声项和相机的内外参数,这些参数必须通过标定获取,精确标定 IMU 和相机对于提高定位算法的准确性至关重要。本文采用了 Kalibr 工具包进行联合标定,以生成 MSCKF 算法所需的关键参数文件。

在水下环境中,首先放置一个标定板,并随后引导 BlueROV 以多种不同的姿态来捕获该标定板的影像。IMU 的内部参数标定对于 ROV 的定位精度至关重要,因为如预积分和视觉惯性联合初始化等关键过程都依赖于 IMU 的测量模型。MSCKF 算法需要输入 IMU 的四个关键参数:加速度随机游走噪声的标准差、加速度测量噪声的标准差、角速度随机游走噪声的标准差。为了标定这些参数,本文采用了 IMU\_util 工具,它要求将 IMU 长时间静止放置,并通过基于时域的 Allen 方差法来进行精确估计。

摄像头与 IMU 之间的外参代表了它们各自坐标系之间的转换关系,这一参数对 MSCKF 的运行具有显著影响。特别是,如果旋转部分的估计存在误差,将会导致轨迹发 生漂移现象。为了准确标定这两者之间的外参,本文采用了 Kalibr 工具进行标定,并将得 到的外参作为初始估计值。在 MSCKF 的运行过程中,这些外参会被不断地优化以提高系统的稳定性和准确性。

# 5.3 水下定位试验

### 5.3.1 小范围静水试验

如图 5-7 所示,本实验平台主要由 QGC、信号中继站、通讯缆、水箱和 ROV 组成。 水箱长 3m,宽 2.5m,深 0.6m,能够在一定程度上满足实验所需的空间和容量要求。

![](_page_69_Picture_5.jpeg)

图 5-7 ROV 实验平台 Fig.5-7 ROV Test Platform

水池底部提前贴好了红色网格线,用于引导 ROV 按照期望路径运动,方便与定位轨 迹进行对比分析。网格的大小为 2m×2m,如图 5-8 所示。

![](_page_69_Figure_8.jpeg)

图 5-8 水池引导网格及 12 条引导路径 Fig.5-8 Pool Guide Grid and 12 Guide Path

将O点定为ROV运动的起点和终点,以不走回头路且中途不经过O点的规则回到起

点,在终点出现视觉回环,可用于本文算法的回环检测与位姿修正部分。在水箱试验了 L1~L12 共 12 条不同的路径,路径长度分别为 4m、6m、8m,每条路径采集了 3 组数据, 将引导路径作为 ROV 的真实轨迹,部分引导路径与定位轨迹如图 5-9 所示。红色和蓝色 曲线分别为 SINS 和本文解算的轨迹,并与绿色的引导路径做差,可以得到三维位置误差 如图 5-10 所示。

![](_page_70_Figure_2.jpeg)

图 5-10 三方向位置误差 Fig.5-10 Position Error in Three Directions

从图中可以看出,在对引导路径 L1 跟踪的过程中,两种算法在x、y方向上的估计 误差都在 0.1m 以内,本文算法估计误差小于 SINS 解算的结果误差,实验结果表明数据 融合后的位姿估计结果优于单一的惯性导航,与理论分析和仿真结果一致。

为了验证定位算法的性能,将上述 12条运动轨迹的终点位置进行分析。ROV 通过引导线从起点 O 出发回到 O 点,但由于系统误差和外部干扰等多种因素的影响,终点位置估计的结果与起点存在微小偏移,导致定位轨迹不是一个闭合的路径。终点的定位结果将在 O 点周围形成散布,如图 5-11 所示为 12条路径终点估计的分布图,散点均分布在 O 点附近。

![](_page_71_Figure_3.jpeg)

Fig.5-11 Distribution of Endpoint Estimates

根据相同条件下 ROV 多次达到终点时系统状态估计的位置分布情况,采用圆概率误差(Circular Error Probable, CEP)来评估定位结果的准确性和稳定性。CEP 是一个衡量二维点位离散分布的指标,它定义了一个以目标位置为圆心的圆,在这个圆内的点位有 50%的概率出现,反映了目标位置估计的不确定性。上图中红色圆圈为目标位置,蓝色散点表示不同路径的终点位置分布。以O点为中心绘制一个圆形,如果定位结果中的散点落入这个圆的几率至少为一半,那么这个圆的半径便被定义为 CEP。半径越小,说明定位精度越高。首先计算二维位置坐标分量的标准差 $\sigma_\lambda$ 和 $\sigma_o$ ,则 CEP 可用下面的公式计算得到:

$$CEP = 0.589(\sigma_1 + \sigma_m) \tag{5-1}$$

式中 0.589 为覆盖 50%点的系数。图中绿色圆圈以散布中心为圆心, CEP=0.09m 为半径, 50%的散点落于圆圈之内。散布中心到目标中心的距离为定位系统误差。

### 5.3.2 室外水池试验

为了进一步验证定位系统的有效性,在室外大空间水池进行了试验。由于在大空间水
池没有真实位置作为参考,为此在 ROV 上搭载 GPS 模块获取经纬度数据进行实验对比。 在 ROV 顶部安装支架将天线架在水面之上用于获取 GPS 数据,安装图如图 5-12 所示。



图 5-12 ROV+WTGPS300 模块 Fig.5-12 ROV&WTGPS300 Module

本次试验选用的 GPS 模块为维特智能公司生产的 WTGPS300 模块,该模块集成 GNSS 接收机、6 轴惯性传感器能够为移动机体提供连续的高精度定位,具有低成本、低功耗、高灵敏度等优势。惯性导航系统一直处于工作状态,输出位置、速度、姿态、加速度、速度的 15 维信息,卫星导航系统输出位置和速度的 6 维信息。WTGPS300 导航系统采用了一种基于卫星导航精度的智能识别算法,设计方案原理如图 5-13 所示。





该导航系统采用组合导航所提供的高度精确的导航数据,以评估卫星导航系统的定位准确性。当卫星导航具有较高的准确性时,该系统将采用组合导航方式。但是,当卫星的导航信号表现得极其不佳或甚至是丢失时,该系统会选择使用纯惯性导航技术。 WTGPS300 导航系统成功地实现了组合导航与纯惯性导航之间的自动切换功能,定位精度可达 1.5m,可用于与本文定位效果进行对比。

WTGPS300 传感器模块与 ROV 进行刚性连接,天线固定在支架确保其暴露在水面之上可以接收到卫星信号,传感器与外接树莓派通过串口连接,启动之后将传感器数据(时间戳、经度、纬度、高度)保存在树莓派。在室外 50m×21m×1m 的水池进行水下试验,测试环境如图 5-14 所示。



图 5-14 ROV 测试环境 Fig.5-14 ROV Test Environment

通过手柄遥控 ROV 绕水池运动一周,记录 WTGPS300 传感器数据、ROV 内置 IMU 和单目相机的视频流,并进行数据处理,得到 WTGPS300 与本文算法估计的 ROV 运动轨 迹如图 5-15 所示。实验结果误差对比见表 5-1。图 5-16 所示为位置解算误差曲线。鉴于 实验条件的限制,无法提供 ROV 的位姿真值,将水池矩形外围作为真值引导 ROV 运动。 ROV 从起点出发,尽量保持高度不变,紧贴水池四周运动一圈,理想的运动轨迹应为一 个二维平面的闭合矩形。

算法	位置 ARMSE/m	闭环误差/m
WTGPS300	1.04	4.22
本文	1.59	5.09
SINS	1.87	6.92

表 5-1 试验结果误差对比 ab.5-1 Error Comparison of Test Resu



图 5-15 室外定位轨迹 Fig.5-15 Outdoor Positioning Track



图 5-16 三方向位置误差 Fig.5-16 Position Error in Three Directions

结合图表可以看出,三种解算方法都能够完整跟踪 ROV 长达 142m 的直线运动,同时都存在定位误差,其中 SINS 的闭环误差高达 6.92m。WTGPS300 模块定位精度更高,更接近真实轨迹,但该模块在 z 轴的位置估计存在较大偏差。与水池外围真实轨迹进行对比,从表 5-1 的位置 ARMSE 可以得出,本文算法解算轨迹更接近精度更高的 WTGPS300 模块,且优于 SINS 的估计结算,与仿真结果一致。虽然 WTGPS300 的估计结果略胜一筹,但是这种方法只能用于水面之上,当 ROV 潜入水下,WTGPS300 的组合导航效果将

大打折扣。而本文导航定位方法不用受此限制,定位结果接近高精度的商用 WTGPS300 模块,提供了一种 ROV 水下导航定位的有效解决方案。

### 5.4 本章小结

在这一章节中,主要针对本文研究的水下组合导航定位系统进行了实际的测试。首先 搭建了实验平台并进行了实验方案设计,在小型水箱和大型室外水池中收集了传感器相 关数据,并进行了数据预处理工作,根据位姿解算方案和修正方法来验证算法的有效性。 由于缺少获取 ROV 真实轨迹的条件,参考相关论文,本文通过引导线指引 ROV 运动的 方式获得了较为准确的轨迹以衡量算法的精度,并与其他导航方式进行对比。实验结果表 明,本文研究的算法有效提高了系统位姿估计的精度,从而验证了其在水下应用中的优越 性,为水下 ROV 导航定位提供了有效的解决方案。

## 6总结与展望

#### 6.1 论文总结

本文对基于视觉与惯导信息融合的 ROV 定位方法进行了研究,针对 MSCKF 算法长时间运行产生累积误差的现象分别从惯性测量值与图像预处理、数据融合技术以及位姿修正等三个方面进行理论研究和实验分析,然后通过数据集仿真以及实物测试来验证本文研究的定位算法的精度。本论文主要工作归纳如下:

(1) 详细阐述了视觉/惯导融合定位的研究现状和发展历程,强调了本文的研究意义。 继而介绍了系统传感器模型,对惯导定位的解算过程进行介绍,对误差来源进行分析与建模,并采用水下图像增强算法对视觉信息进行优化进而提取图像特征。

(2) 对基于滤波的视觉与惯导融合算法 MSCKF 进行研究,首先在图像处理部分进行特征点检测、匹配跟踪与误匹配剔除,采用紧耦合的方案将图像特征与 IMU 数据进行融合,详尽推导了基于滤波的视觉与惯导融合算法的理论和公式,选取了系统状态向量并基于系统的运动模型和观测模型进行滤波器的设计,深入探讨了包含系统状态预测、状态增广以及滤波器观测更新在内的位姿解算方案,最终得到机体位置与姿态,并通过 KITTI 数据集仿真与 SWF 算法进行了对比分析,实验结果表明了本文研究的定位算法具有较高的准确性和实时性。

(3) 针对系统运动估计存在的累积误差,根据两种不同情况进行纠正:在 ROV 经过 历史位置的情况下,进行水下回环检测与校正;在 ROV 不经过历史位置的情况下,则利 用已知标志点进行外环位姿修正。探讨了回环检测的基本原理,通过词袋模型建立了水下 视觉字典进行相似度计算来评估两幅图像是否构成回环。详细描述了本次研究中回环检 测和校正的工作原理,给出了回环候选帧筛选策略并采用几何约束对改进的水下回环识 别进行研究,然后归纳了算法处理流程并通过数据集验证了修正方法的可行性。最后阐述 了标志点检测的过程对机体位姿进行修正,并在 KITTI 数据集仿真的基础上进行了验证, 定位算法的位置 ARMSE 减小了 23.07%,有效降低了位姿估计的累积误差。

(4) 基于 BlueROV 开源软硬件搭建了实验平台,通过实验方案设计分别在室内水箱 和室外水池两种不同环境中进行了水下实验,以验证所研究方法的有效性。实验结果表明, 本文研究的算法有效提高了系统位姿估计的精度,从而验证了其在水下应用中的优越性。

本文研究的 ROV 组合导航系统与 SLAM 的区别是不创建地图,计算资源消耗较少, 能更好地服务于水下作业,但与传统的视觉惯性里程计相比,增加了特征点三维位置解算 的过程,充分发挥了视觉导航系统图像信息丰富的优势,这也有助于实现高精度的导航定 位,以确保完成高复杂度的水下任务。

#### 6.2 未来展望

尽管本文通过一系列实验验证了所提出算法的可行性,但仍存在诸多待解决的问题 和挑战。

(1) 在进行水下摄像头的标定时,当前通过 Kalibr 获得的参数被视为初步或近似的估计值。为了更为精确地减少重投影误差,需要构建一个精细的光路模型,该模型将包含玻璃罩的折射率、厚度、半径等物理特性,以及玻璃罩中心与摄像头之间的相对位置和方向等参数。

(2)为了验证本文研究的效果,在清澈的静水下完成了试验,如果系统的运动速度太快导致图像信息变得模糊,那么特征提取可能会失败,但在后续阶段,可以先对图像进行 去模糊处理,然后再对其特征进行进一步的处理。

(3) 鉴于实验条件的限制,为了尽可能获取 ROV 在水下运动的实际轨迹,在小范围 水箱底部布置引导线作为轨迹参考进行结果对比分析,在大空间室外水池采用高精度卫 星/惯导组合导航模块的结果与本文研究结果进行对比,但这两种方案仅能够对 ROV 位置 进行对比,无法得到姿态信息,得到的结果精度欠佳不能替代水下真实轨迹。相关研究文 献在声呐传感器精确测量的情况下,将其作为参考轨迹提供对真实度的估计。

# 参考文献

- [1] 张景全. 我国海洋强国建设面临的机遇与挑战[J]. 人民论坛, 2023, 14(20): 26-29.
- [2] 金永明. 新时代海洋强国战略的演进历程与重要成就[J]. 国家治理, 2024, (2): 40-45.
- [3] 林善炜. 习近平海洋强国战略思想的核心要义与时代价值[J]. 理论视野, 2023, 21(7): 23-28.
- [4] Eva P. Shedding light on deep-sea biodiversity—a highly vulnerable habitat in the face of anthropogenic change[J]. Frontiers in Marine Science, 2021, (3): 667048.
- [5] Danovaro R. Climate change impacts on the biota and on vulnerable habitats of the deep Mediterranean Sea[J]. Rendiconti Lincei. Scienze Fisiche e Naturali, 2018, 29(3): 525-541.
- [6] Danovaro R, Corinaldesi C, Dell'Anno A, et al. The deep-sea under global change[J]. Current Biology, 2017, 27(11): R461-R465.
- [7] 吴有生,赵羿羽,郎舒妍,等.智能无人潜水器技术发展研究[J].中国工程科学,2020, 22(06):26-31.
- [8] 张强, 张雯. 水下机器人导航技术[M]. 北京: 科学出版社, 2019: 1-7.
- [9] 冀大雄. 水下机器人先进导航技术[M]. 北京: 科学出版社, 2019: 2-8.
- [10] Shkurti, Florian, et al. State estimation of an underwater robot using visual and inertial information[C]. International Conference on Intelligent Robots and Systems, San Francisco, USA: IEEE, 2011.
- [11] 许竞克, 王佑君, 侯宝科, 等. ROV 的研发现状及发展趋势[J]. 兵器装备工程学报, 2011, 32(004): 71-74.
- [12] Troni G, Whitcomb L L. Preliminary experimental evaluation of in-situ calibration methods for MEMS-based attitude sensors and Doppler sonars in underwater vehicle navigation[C]. Autonomous Underwater Vehicles (AUV), 2012 IEEE/OES. Southampton, UK: IEEE, 2012: 1-8.
- [13] Le D K, Nguyen D H, Ranmuthugala D, et al. A heading observer for ROVs under roll and pitch oscillations and acceleration disturbances using low-cost sensors[J]. Ocean Engineering, 2015, 110(08): 152-162.
- [14] 宋保维, 潘光, 张立川, 等.自主水下航行器发展趋势及关键技术[J]. 中国舰船研究, 2022, 17(05): 27-44.
- [15] 李经. 水下无人作战系统装备现状及发展趋势[J]. 舰船科学技术, 2017, 39(1): 1-5.
- [16] 连琏, 马厦飞, 陶军. "海马"号 4500 米级 ROV 系统研发历程[J]. 船舶与海洋工程, 2015, 31(1): 9-12.

- [17] Zeng Q, Zhang M, Liu H, et al. Research of underwater navigation on a ROV with structure detection and decontamination[C]. Control and Decision Conference(CCDC), 2015 27th Chinese. Qingdao, China: IEEE, 2015: 2585-2589.
- [18] Djuknic G, Richton R. Geolocation and assisted GPS[J]. Computer, 2001, 34(2): 123-125.
- [19] Paull L, Saeedi S, Seto M, et al. AUV Navigation and Localization: A Review[J]. IEEE Journal of Oceanic Engineering, 2014, 39(1): 131-149.
- [20] 黄玉龙, 张勇刚, 赵玉新. 自主水下航行器导航方法综述[J]. 水下无人系统学报, 2019, 27(3): 232-253.
- [21]梁益丰, 许江宁, 吴苗, 等. AUV 导航技术概述[J]. 舰船科学技术, 2020, 42(15): 152-156.
- [22] 朱庄生, 万德钧, 王庆. 航位推算累积误差实时修正算法研究[J]. 中国惯性技术学报, 2003, 12(3): 8-12.
- [23] 刘沛佳, 秦丽萍, 李广华, 等. 一种面向无人水下航行器的多普勒自主导航方法[J]. 水下无人系统学报, 2023, 31(03): 373-380.
- [24] 赖昱, 朱俊. 水下机器人惯性导航技术综述[J]. 控制与信息技术, 2023, 23(6): 9-15.
- [25] 王博,周明龙.水下重力辅助导航适配区选取的研究进展[J].导航定位学报,2020, 8(03): 32-39.
- [26] 王依能. 基于地形辅助的水下潜器导航技术研究[D]. 南京: 东南大学, 2022: 3.
- [27] 孙大军, 郑翠娥, 钱洪宝, 等. 水声定位系统在海洋工程中的应用[J]. 声学技术, 2012, 31(2): 125-132.
- [28] 商承超, 王伟, 谢广明, 等. 水下机器人定位方法综述[J]. 兵工自动化, 2013, 32(12): 46-50.
- [29] 魏博文, 吕文红, 范晓静, 等. AUV 导航技术发展现状与展望[J]. 水下无人系统学报, 2019, 27(01): 1-9.
- [30] DURRANT-WHYTE H, BAILEY T. Simultaneous localization and mapping: part I[J]. IEEE robotics and automation magazine: A publication of the IEEE Robotics and Automation Society, 2006, 13(2): 99-108.
- [31] BAILEY T, DURRANT-WHYTE H. Simultaneous localization and mapping(SLAM): Part II[J]. IEEE robotics and automation magazine: A publication of the IEEE Robotics and Automation Society, 2006, 13(3): 108-117.
- [32] Jie L, Michael K, M. R E, et al. Pose-graph SLAM using forward-looking sonar[J]. IEEE Robotics and Automation Letters, 2018, 3(3): 2330-2337.
- [33] 刘明雍. 水下航行器协同导航技术[M]. 北京: 国防工业出版社. 2014: 1-2.
- [34] Yulong H, Yonggang Z, Bo X, et al. A New Outlier-robust Student's t Based Gaussian Approximate Filter for Cooperative Localization[J]. IEEE/ASME Transactions on

Mechatronics, 2017, 22(5): 2380-2386.

- [35] Mourikis A I, Roumeliotis S I. A multi-state constraint Kalman filter for vision-aided inertial navigation[C]. Proceedings 2007 IEEE international conference on robotics and automation. Rome, Italy: IEEE, 2007: 3565-3572.
- [36] Li M, Mourikis A I. High-precision, consistent EKF-based visual-inertial odometry[J]. The International Journal of Robotics Research, 2013, 32(6): 690-711.
- [37] Ke S, Kartik M, Bernd P, et al. Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight[J]. IEEE Robotics and Automation Letters, 2018, 3(2): 965-972.
- [38] Geneva P, Eckenhoff K, Lee W, et al. OpenVINS: A Research Platform for Visual-Inertial Estimation[C]. Proc. of the IEEE International Conference on Robotics and Automation. Paris, France: IEEE, 2020.
- [39] Weiss S M. Vision based navigation for micro helicopters[D]. Zurich: ETH Zurich, 2012: 10.
- [40] Lynen S, Achtelik M W, Weiss S, et al. A robust and modular multi-sensor fusion approach applied to mav navigation[C]. 2013 IEEE/RSJ international conference on intelligent robots and systems. Tokyo, Japan: IEEE, 2013: 3923-3929.
- [41] Bloesch M, Omari S, Hutter M, et al. Robust visual inertial odometry using a direct EKFbased approach[C]. 2015 IEEE/RSJ international conference on intelligent robots and systems (IROS). Hamburg, Germany: IEEE, 2015: 298-304.
- [42] Bloesch M, Burri M, Omari S, et al. Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback[J]. The International Journal of Robotics Research, 2017, 36(10): 1053-1072.
- [43] Tong Q, Peiliang L, Shaojie S. Vins-mono: A robust and versatile monocular visual-inertial state estimator[J]. IEEE Transactions on Robotics, 2018, 34(4): 1004-1020.
- [44] Qin T, Shen S. Online temporal calibration for monocular visual-inertial systems[C]. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS). Madrid, Spain: IEEE, 2018: 3662-3669.
- [45] Liu H, Chen M, Zhang G, et al. Ice-ba:Incremental, consistent and efficient bundle adjustment for visual-inertial slam[C]. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018: 1974-1982.
- [46] Yu X, Li T, Zeng Z. A Visual-Inertial Information Fusion Method for SLAM Front-End Odometry[C]. Image and Graphics Technologies and Applications: 14th Conference on Image and Graphics Technologies and Applications, 2019: 184-192.
- [47] Triggs B, McLauchlan P F, Hartley R I, et al. Bundle adjustment—a modern synthesis[C]. Vision Algorithms: Theory and Practice:International Workshop on Vision Algorithms Corfu, Greece, 2000: 298-372.

- [48] Thomas S, Marcin D, Marius F, et al. Maplab: An open framework for research in visualinertial mapping and localization[J]. IEEE Robotics and Automation Letters, 2018, 3(3): 1418-1425.
- [49] Cramariuc A, Bernreiter L, Tschopp F, et al. maplab 2.0–A Modular and Multi-Modal Mapping Framework[J]. IEEE Robotics and Automation Letters, 2022, 8(2): 520-527.
- [50] Weiss S, Achtelik M W, Lynen S, et al. Real-time onboard visual-inertial state estimation and self-calibration of MAVs in unknown environments[C]. 2012 IEEE international conference on robotics and automation. Saint Paul: IEEE, USA, 2012: 957-964.
- [51] L. Ma, J. Falquez, S. McGuire, and G. Sibley. Large scale dense visual inertial slam[C]. 2015 International Conference on Field and Service Robotics, 2015.
- [52] Falquez J M, Kasper M, Sibley G. Inertial aided dense & semi-dense methods for robust direct visual odometry[C]. 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). Daejeon, Korea: IEEE, 2016.
- [53] Kneip L, Chli M, Siegwart R. Robust Real-Time Visual Odometry with a Single Camera and an IMU[C]. British Machine Vision Conference. 2011.
- [54] BrezovS D, MladenovaD C, MladenovM I. New perspective on the gimbal lock problem[J]. AIP Conference Proceedings, 2013, 1570(1): 367.
- [55] 张荣辉, 贾宏光, 陈涛, 等. 基于四元数法的捷联式惯性导航系统的姿态解算[J]. 光 学精密工程, 2008, 11(10): 1963-1970.
- [56] 高翔. 视觉 SLAM 十四讲:从理论到实践[M].北京:电子工业出版社,2017:23.
- [57] 汪首坤, 赵金枝, 姜明, 等. 基于圆形阵列标定板的张氏相机标定法[J]. 北京理工大学学报, 2019, 39(8): 859-863.
- [58] Diksha G, Kumar N G, Munish K. Underwater image enhancement using blending of CLAHE and percentile methodologies[J]. Multimedia Tools and Applications, 2018, 77(20): 26545-26561.
- [59] Rosten E, Drummond T. Machine learning for high-speed corner detection[C]. European Conference on Computer Vision. 2006: 430-443.
- [60] 赵小强, 徐铸业. 基于改进的 FAST 和 FREAK 的图像匹配算法[J]. 兰州理工大学学报, 2018, 44(3): 83-87.
- [61] Meinhardt-Llopis E, Pérez S J, Kondermann D. Horn-Schunck Optical Flow with a Multi-Scale Strategy[J]. Image Processing On Line, 2013, 3(8): 151-172.
- [62] 孙晓峰, 宫金良, 张彦斐. 改进光流法的避障与辅助导航算法[J]. 激光杂志, 2019, 40 (10): 16-20.
- [63] María J M, Itsaso R, Iñigo M, et al. RANSAC for Robotic Applications: A Survey[J]. Sensors, 2022, 23(1): 327.

- [64] 秦永元, 张洪钺, 汪叔华. 卡尔曼滤波与组合导航原理[M]. 西安: 西北工业大学出版 社, 2015: 1-5.
- [65] Pedro J C L, Jesús H G, Manuel J L M. Forecasting Nonlinear Systems with LSTM: Analysis and Comparison with EKF[J]. Sensors, 2021, 21(5): 1805.
- [66] Geiger A, Lenz P, Stiller C, et al. Vision meets robotics: The KITTI dataset[J]. International Journal of Robotics Research, 2013, 32(11): 1231-1237.
- [67] Huang G P, Mourikis A I, Roumeliotis S I. An observability-constrained sliding window filter for SLAM[C]. IEEE/RSJ International Conference on Intelligent Robots & Systems. San Francisco, USA: IEEE, 2011.
- [68] Hartley O H. The Modified Gauss-Newton Method for the Fitting of Non-Linear Regression Functions by Least Squares[J]. Technometrics, 2012, 3(2): 269-280.
- [69] Brown K M, Dennis J E. Derivative free analogues of the Levenberg-Marquardt and Gauss algorithms for nonlinear least squares approximation[J]. Numerische Mathematik, 1971, 18(4): 289-297.
- [70] Clement L E, Peretroukhin V, Lambert J, et al. The Battle for Filter Supremacy: A Comparative Study of the Multi-State Constraint Kalman Filter and the Sliding Window Filter[C]. Computer & Robot Vision. Halifax, Canada: IEEE, 2015.
- [71] 孙伟, 宋如意, 王宇航. 视觉/惯性组合导航中的 SWF 与 MSCKF 对比研究[J]. 中国矿 业大学学报, 2020, 49(1): 198-204.
- [72] 吴晓燕, 王海燕. 基于显著图的特征字典构造方法[J]. 计算机工程与设计, 2016, 37(08): 2243-2247.
- [73] Robertson S. Understanding inverse document frequency: on theoretical arguments for IDF[J]. Journal of Documentation, 2004, 60(5): 503-520.
- [74] Xu X, Yu C, Zhou J. Robust feature point matching based on geometric consistency and affine invariant spatial constraint[C]. 2013 IEEE International Conference on Image Processing. Melbourne, Australia: IEEE, 2013: 2077-2081.
- [75] 黄晓浪. 基于灰度变化的角点检测算法研究[D]. 南昌: 东华理工大学, 2019: 21-22.