

学校代码: 10286
分类号: V279
密 级: 公开
U D C: 681.5
学 号: 183206



室内动态环境下 无人机视觉惯性组合定位技术研究

研究生姓名: 施佳豪
导师姓名: 王 庆

申请学位类别 工学硕士 学位授予单位 东南大学
一级学科名称 仪器科学与技术 论文答辩日期 20 年 月 日
二级学科名称 仪器科学与技术 学位授予日期 20 年 月 日
答辩委员会主席 评 阅 人

20 年 月 日

東南大學

硕士学位论文

室内动态环境下 无人机视觉惯性组合定位技术研究

专业名称: 仪器科学与技术

研究生姓名: 施佳豪

导师姓名: 王 庆

本论文获国家自然科学基金课题（42074039）和国家重点研发计划（2020YFD110011）资助。

RESEARCH ON POSITIONING TECHNOLOGY OF UAV IN INDOOR DYNAMIC ENVIRONMENT BASED ON VISUAL-INERTIAL FUSION

A Thesis Submitted to

Southeast University

For the Academic Degree of Master of Engineering

BY

SHI Jia-hao

Supervised by

Prof. Wang Qing

School of Instrument Science & Engineering

Southeast University

May, 2021

摘要

随着无人机应用领域不断拓宽，无人机的自主导航定位也变得越发重要。尤其是在未知的室内环境下，面对卫星信号丢失、动态环境干扰等挑战，实现无人机准确稳定的自主定位具有重要意义。因此本文针对无人机在室内动态环境下定位的应用场景，研究视觉与惯性测量单元（IMU）紧耦合的里程计算法，在保证实时性的前提下，准确识别环境中的动态特征并减少其对位姿估计的干扰，同时融合视觉信息降低 IMU 累积误差对定位结果的影响，从而提高整体的定位精度。本文的主要研究内容包括：

第一，研究传感器的标定算法。针对传统相机标定方法维度信息丢失、标定结果不稳定等缺陷，提出改进的三维标定算法，以提升相机内参标定的准确性与稳定度。同时研究 IMU 内参、相机与 IMU 外参联合标定方法，为后端紧耦合算法提供准确的标定参数。

第二，研究视觉惯导里程计算法。在视觉前端，提出一种基于运动分割的动态特征识别方法，利用 IMU 测量信息对视觉前端进行运动补偿，并计算补偿后的特征点像素差，根据动、静特征像素差不同的特性，使用改进的 K-means 算法对特征点进行聚类，从而识别出环境中的动态特征点并予以剔除；在扩展卡尔曼滤波的框架上对视觉与 IMU 测量信息进行紧耦合，为降低运算的复杂性，使用相机位姿代替特征点作为系统的状态量，并建立多状态几何约束对系统状态进行更新，同时为降低 IMU 累积误差对定位精度的影响，改进滤波过程中相机状态扩增方式，使用视觉与 IMU 的融合计算结果作为状态向量中新的相机位姿。

第三，以英伟达 TX2 核心开发板为主搭建无人机平台，进行实验验证。在实际的动态室内环境下采集数据，验证动态特征识别算法的有效性与准确性。同时估计无人机的飞行轨迹，并将结果与经典的 MSCKF 算法进行对比，验证本文算法在定位精度上的优越性。

数据集仿真与无人机真实飞行实验结果表明，本文提出的 VIO 算法能准确区分环境中的动态特征，并能在保证实时性的前提下获取接近 VINS_Fusion 算法的定位精度，相较于经典的 MSCKF 算法而言，本文算法也能更好的抑制累积误差，提升无人机飞行轨迹的估计精度。

关键词：无人机，视觉惯导里程计，动态场景，传感器标定，扩展卡尔曼滤波

ABSTRACT

With the application fields of unmanned aerial vehicles (UAV) being further extended, the autonomous navigation and positioning of UAV has become increasingly important especially in the indoor environment where the UAV could face challenges of satellite signal loss and disturbance of dynamic features. This paper focuses on a tightly coupled Visual Inertial Odometry (VIO) which operates for UAV's positioning in the indoor dynamic environment. On the premise of satisfying the real-time quality of the system, our work aims at identifying dynamic features to reduce their interference and using the visual information to reduce the cumulative error from the IMU. The main contents are as follows:

Firstly, the calibration algorithm is studied to accurately integrate the measurement information from the camera and IMU. Considering the defects of traditional camera calibration methods such as the loss of dimensional information and unstable calibration results, an improved 3D calibration algorithm is proposed to achieve more accurate and stable camera internal parameters. Meanwhile, the methods of calibrating IMU internal parameters and camera-IMU external parameters are studied to provide accurate calibration parameters.

Secondly, the tightly coupled VIO is mainly focused. In the visual front end, a dynamic feature recognition method based on motion segmentation is proposed, which uses IMU measurements to perform motion compensation and calculates pixel differences. The dynamic features in the environment can be identified and removed by using the improved K-means algorithm to cluster features according to the different pixel differences between the static and dynamic features. The measurements from the camera and IMU are tightly coupled in the framework of Extended Kalman Filter (EKF). In order to reduce the complexity of calculation, camera poses are used as the states of the system instead of features and the states are updated by establishing multi-state geometric constraints. The method of camera state amplification during the filtering is improved to reduce the impact of the cumulative error from IMU, by using the fusion result of the vision and IMU as the new camera pose in the state vector.

Thirdly, an experimental UAV platform is built mainly based on the NVIDIA TX2 core development board on which the experiment is carried out. By collecting data in the actual dynamic indoor environment, the performance of the dynamic feature recognition algorithm can be evaluated and the trajectory of the flight can be estimated. Compared with the original MSCKF algorithm, the superior accuracy of the VIO proposed here is verified.

The simulation on datasets and real flight experiment results show that the VIO algorithm proposed in this paper can accurately distinguish dynamic features in the environment and can obtain accurate pose estimations which are comparable to the VINS_Fusion in real time. Compared with the classic MSCKF method, the VIO proposed here can better reduce cumulative errors and improve the accuracy of the whole system.

Keywords: UAV, Visual Inertial Odometer, Dynamic environment, Sensor calibration, Extended Kalman Filter

目录

摘要.....	I
ABSTRACT.....	III
目录.....	V
第一章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.2.1 视觉惯导融合定位技术研究现状.....	2
1.2.2 动态特征识别与滤除研究现状.....	4
1.3 本文研究内容与组织结构.....	5
第二章 视觉惯性里程计算法相关原理.....	7
2.1 刚体运动模型.....	7
2.2 坐标系定义.....	9
2.2.1 惯导坐标系.....	9
2.2.2 针孔相机模型.....	9
2.3 最小二乘问题求解方法.....	11
2.3.1 奇异值分解.....	11
2.3.2 非线性优化.....	12
2.4 EKF 滤波优化算法.....	14
2.5 本章小结.....	15
第三章 传感器标定方法.....	17
3.1 改进的相机标定算法.....	17
3.2 双目相机外参标定算法.....	19
3.3 IMU 误差分析和标定算法.....	20
3.3.1 IMU 测量模型.....	20
3.3.2 IMU 标定算法.....	21
3.4 IMU 和相机联合标定算法.....	22
3.5 传感器标定实验.....	23
3.5.1 相机内参标定实验与分析.....	23
3.5.2 IMU 标定实验.....	26
3.5.3 IMU 与相机联合标定实验.....	27
3.6 本章小结.....	28
第四章 视觉与 IMU 组合导航定位.....	29
4.1 视觉惯导里程计算法总体框架.....	29
4.2 视觉里程计前端.....	29
4.2.1 光流法.....	30
4.2.2 特征跟踪.....	30
4.3 动态特征识别与滤除.....	32
4.3.1 IMU 运动补偿.....	32
4.3.2 基于改进 K-means 算法的动态特征识别.....	33
4.4 基于滤波的多状态约束紧耦合 VIO.....	34
4.4.1 EKF 状态向量.....	35
4.4.2 运动方程.....	36
4.4.3 改进的相机状态扩增方式.....	37

4.4.4 基于多状态约束的观测模型	39
4.4.5 EKF 更新策略	40
4.5 数据集仿真实验	41
4.6 本章小结	46
第五章 无人机飞行实验设计与分析	47
5.1 实验平台与场景	47
5.2 动态特征识别实验	48
5.3 定位精度评估	50
5.4 本章小结	53
第六章 总结与展望	55
致谢	57
参考文献	59
作者简介	63

第一章 绪论

1.1 研究背景及意义

随着智能技术的发展以及制造工艺的提升,无人机的适用领域变得越来越广泛^[1]。无人机的诞生最早可以追溯到第一次世界大战时期,1910 年左右,美国俄亥俄州的年轻军事工程师科特林就提出了使用无人驾驶飞行器的建议,并使用钟表机械装置设计出了可以自动飞行的飞行器,受到了美国军方的关注与支持。在后来的越南战争和第五次中东战争中,无人机也得到了广泛的使用,成为了军事界的宠儿。在计算机技术广泛发展、制造成本降低之后,无人机也开始逐步进军消费级民用行业。现如今已经可以在很多领域看到无人机的身影,包括电影制作、航空拍摄、土地勘测^[2-4]等等。

市面上现存的无人机的种类有很多,按其飞行特点可以将其大致分为固定翼无人机^[5]和旋翼无人机^[6]两种。固定翼无人机因其机翼固定而得名,其飞行主要升力来自机翼与空气的相对运动,因此其具有飞行速度快、经济、运载能力大等优点,被广泛应用于军事领域。相比于固定翼无人机,旋翼无人机在生活中更为常见,其依靠多个旋翼产生的升力来驱动飞行器进行飞行。旋翼无人机虽然对动力要求高、航程短,但是其可操控性好、机动灵活性高,因此在近几年受到了市场的广泛关注,也成为了企业和学界的主要研究对象。

在无人机的各项研究方向中,自主定位一直是研究的热点内容,特别是在未知环境中,无人机的精准定位是其进行导航规划、安全飞行的重要前提保障。现阶段无人机在未知环境下的自主定位方法主要分为两大类:一类是借助于全球卫星定位技术^[7],如 GPS、北斗等,通过多颗卫星的空间距离测算来获取无人机的实时位置;另一类是借助相应的机载传感器来实现无人机的主动定位,例如同时定位与建图(Simultaneous Localization and Mapping, SLAM)技术^[8],该技术使用轻便、低成本的传感器来获取无人机飞行过程中的环境信息,从而实现自主定位。对于卫星定位技术而言,其位置信息的获取与无人机的运动过程无关,因此不存在累积误差,精度相对较高,但是其只能在卫星信号较好的室外环境下得到较好的应用。现如今,随着无人机用途的不断拓宽,其在卫星信号受到遮挡的室内环境下也有了很大的应用需求,因此以 SLAM 为主的主动定位技术也将大显身手。

根据使用传感器的不同,可以将常用的 SLAM 算法分成激光 SLAM 和视觉 SLAM 两种。激光 SLAM 使用激光雷达感知环境信息,从而实现导航定位。激光雷达精度高,对环境不敏感,但是其价格昂贵,且重量较大^[9],对于无人机的载重和续航是一大考验。相比之下,视觉 SLAM 算法更适合运用于无人机平台上,可以使用双目相机采集无人机飞行时的图像数据,在控制载重量和成本的同时也能为系统提高较为可靠的尺度信息。

纯视觉 SLAM 定位方法也存在一定的缺陷,其容易受到光照变化等外界环境带来的影响,鲁棒性不高,因此常使用视觉与其他传感器实现组合导航定位,其中最常用的是惯性测量单元(IMU)。IMU 测量运动信息时对环境不敏感,鲁棒性较高,同时视觉信息又可以弥补 IMU 易产生累积误差的缺陷,两者融合能够起到优势互补的作用。无人机在室内飞行时,也要考虑环境本身的特点,室内环境视野较窄,当环境中动态特征较多时必然会对无人机的自主定位产生影响。普通的无人机平台计算资源也相对比较紧俏,如何利用有限的资源,降

低动态特征的影响，保证无人机定位算法的实时性与准确性具有非常大的研究意义。

本论文对于无人机在室内实现自主定位这一应用场景，研究视觉与 IMU 组合的视觉惯导里程计算法（Visual Inertial Odometry, VIO），力求在嵌入式级别的无人机运算平台上，实现高精度的实时定位，并提升算法在动态环境中的抗干扰能力，增强系统的鲁棒性。

1.2 国内外研究现状

1.2.1 视觉惯导融合定位技术研究现状

无人机飞行时非常灵活，速度也相对较快，对于单一的纯视觉定位方法而言，可能在获取运动信息时出现图像模糊、特征点丢失、尺度测量不准等问题，从而估计出错误的飞行轨迹。因此，近来国内外学者都开始将研究重点转向组合导航定位方向，其中研究最为广泛的就是视觉与IMU的组合导航定位。

视觉与IMU的组合方法主要有松耦合与紧耦合两种，两者的区别在于组合信息不同。松耦合框架中视觉与IMU各自计算自己的定位结果，最终将两种结果进行组合。松耦合方法学界研究的较少，以苏黎世联邦工业大学Stephen Weiss提出的MSF框架为典型^[10]，该方法在扩展卡尔曼滤波（Extended Kalman Filter, EKF）的框架下为多传感器融合定位提供了通用的融合方案，在个别传感器信号丢失时也能获取绝对状态和相对状态更新的最佳线性化点。松耦合方法的优势在于计算量小，复杂度低，但是信息融合度不高，从而并不能完全发挥出两者的互补优势。

紧耦合方法是将IMU与视觉测量的原始数据进行组合从而对系统的定位状态进行更全面的估计，此类算法的复杂度高，但是其定位效果相比松耦合方法要更好。国外有很多学者提出了自己的紧耦合方案，根据其数据融合方式的不同，主要可以分为基于非线性优化的方案与基于滤波的方案。基于优化的方案是近年来的研究热点，其计算精度相比滤波算法更好^[11]，适用性更广，在状态变化较大时亦能适用。

非线性优化框架中比较经典的算法是Stefan Leutenegger等人提出OKVIS算法^[12]，该方法是一种基于双目视觉与IMU的视觉惯导里程计。OKVIS 的算法流程是通过 IMU 测量值对当前状态做预测，根据预测在前端进行特征提取和特征匹配，使用3D点特征和二维图像特征构成优化中的重投影误差，同时在预测的 IMU 状态量和优化的参数之间构成 IMU 测量误差，并使用滑动窗口对整体位姿进行优化。

2017年，Raul Mur-Artal等人在自己提出的纯视觉SLAM算法ORB-SLAM的基础之上，引入IMU测量信息，提出了紧耦合的VIO SLAM^[13]。在该方案中借助IMU初始化系统的尺度比例以及重力方向，并使用IMU预积分解决了视觉与IMU时间戳对齐问题，在后端采用位姿图优化的方法优化了紧耦合的定位结果。根据作者实验表明，该方案在地图重用以及降低漂移累计方面具有明显的作用。

在这些经典的开源框架基础上，国外很多学者提出了自己的改进思路。Christian Forster等人^[14]针对VIO算法框架中，惯性测量速率过高，实时优化无法保证的问题提出了IMU预积分理论，并设计了因子图优化框架，将IMU预积分模型无缝集成到视觉惯导

里程计中,提升了后端优化的速度,实现了实时准确的位姿估计;Hartley等人^[15]提出了一种能够集成任意数量传感器信息的预积分理论,系统运行过程中只对图像关键帧的位姿状态进行估计,从而减少了非线性优化中的变量,提升了算法的实时性;Vladyslav Usenko等人^[16]针对VIO系统在关键帧之间运行时,出现惯性有用信息丢失的问题,使用非线性因子对VIO估计的轨迹信息进行了最佳逼近,从而实现从视觉惯性里程计中提取视觉惯性映射的相关信息。

基于非线性优化的方法虽然精度较高,但是其对计算资源的占用也更多,在一些算力局限的设备上无法保证较好的实时性与鲁棒性。相比之下,基于滤波的算法框架计算复杂度更低,定位精度也能达到与优化方法相接近的水准。基于滤波的算法框架中,最为经典的是Mourikis等人^[17]提出MSCKF算法,该方法使用扩展卡尔曼滤波(EKF)对IMU和视觉数据进行紧耦合,并使用相机状态代替环境中的三维特征点作为系统的待估计状态,从而降低了运算复杂度,提升了效率。在MSCKF的基础上MingyangLi等人^[18]做出改进,确保了线性系统模型正确的观测性并提供了在线估计校正相机和IMU外参的功能。

2015年,Michael Bloesch等人^[19]提出 ROVIO算法。该算法的前端直接使用图像中的像素强度误差对特征点进行准确跟踪,并将特征点之间的误差项作为观测量进行更新。后端也使用扩展卡尔曼滤波器实现数据紧耦合,滤波过程中将地标位置按方位进行分解,并将距离参数化,从而提高了计算性能。

在滤波框架改进方面,Jeff Delaune等人^[20]针对在没有加速度计激励时,单目视觉惯性里程计无法观察到尺度信息并产生严重误差漂移的情况,提出了基于面约束的范围测量观测模型,使得VIO算法在匀速运动场景下也能实现较好的定位效果;Trung Nguyen等人^[21]利用三焦点张量几何的视觉测量方法来代替传统VIO系统中计算量大的三维特征点重建,并使用基于非线性采样的卡尔曼滤波器来处理非线性系统。这种替换简化了系统架构,并显著减少了处理时间。与传统的扩展卡尔曼滤波器相比,该方法有助于获取更好的估计精度和鲁棒性能;Lee Kyuman等人^[22]针对扩展卡尔曼滤波过程中存在的IMU数据和视觉数据时间偏移问题,提出了一种未知延迟参数的补偿方法。将参数估计合并到特征初始化和状态估计中,从而实现时间的在线校准并估计延迟可校正残差、雅可比和协方差矩阵。

国内对VIO算法的研究起步较晚,提出的VIO算法框架较少,但也不乏优秀的解决方案,其中最具代表性的是香港科技大学沈劭劼老师团队提出的VINS算法^[23]。该算法是基于优化和滑动窗口的 VIO,使用 IMU预积分构建紧耦合框架,同时还有自动初始化、在线外参标定、重定位、闭环检测以及全局位姿图优化功能。VINS的定位精度可以与OKVIS相媲美,但其CPU占用率更低,系统更加完善,鲁棒性更高。

虽然国内提出的算法框架较少,但是也有许多学者在这方面做了很多改进。W HUANG等人^[24]针对VIO算法中的初始化问题,提出了一种基于优化的在线初始化方法以在线校准相机和IMU之间的相对变换和时间偏移。通过最小化相机和IMU之间的旋转差异来估计外部旋转、陀螺仪偏置和时间偏移,并通过使用补偿的摄像机姿态近似估算尺度比例、重力方向与外部平移;Xiaochen Qiu等人^[25]在研究VIO时使用更具辨识度的

汉密尔顿符号来表示四元数，并从零开始推导了误差状态转换方程，提出了易于计算的完全线性闭式解。同时针对滤波器后端容易受到匹配外点的影响，开发了以描述子辅助的光流跟踪前端；Yijia He等人^[26]提出了一种利用点和线特征的视觉惯性里程计系统，为了紧凑表示3D空间线并实现简便计算，采用了Plücker坐标和线的正交表示方法。算法后端将预积分的IMU误差项与点、线的重投影误差项组合在一起进行窗口优化，从而实现视觉与IMU的紧耦合。

1.2.2 动态特征识别与滤除研究现状

对于视觉惯导组合的里程算法而言，无论是松耦合还是紧耦合方案，都需要使用环境中的特征点信息来估计自身的位姿参数。在传统的SLAM算法中，一般都会假设环境中提取的特征静止不动，一旦环境中出现大量动态物体，致使该假设不成立时，则会影响算法整体的定位精度。为此，国内外很多学者在动态特征识别与滤除方向做了大量的研究工作。

Nikolas Brasch等人^[27]针对高动态环境提出了一种单目视觉SLAM方法，该方法基于卷积神经网络（Convolutional Neural Networks, CNN）预测的语义先验信息，使用联合概率模型对动态异常值进行建模。为了在快速的相机运动和无纹理的环境中提取足够的特征，使用了具有描述性功能的直接特征点法来提取特征。同时扩展CNN进行语义分割，基于图像上下文直接预测属于动态对象的像素概率，从而减少获得可靠比率值所需的观测次数。

Chee Sing Lee等人^[28]提出了一种可以估计环境中动态和静态特征位置的算法，该算法将基于特征的SLAM问题建模为单集群过程，并定义了一种易处理的贝叶斯滤波器递归公式。该滤波器在处理过程中，能为多个目标提供鲁棒的极大似然估计。同时，该算法针对定位过程中视野改变、静态和动态地图特征混合等情况，使用粒子滤波和高斯滤波混合的滤波方式对环境特征进行过滤。

Bescos Berta等人^[29]提出了名为DynaSLAM的算法框架，该框架在识别动态特征时融合了深度学习与多视图几何方法，使用Mask R-CNN网络对动态物体进行像素分割，随后使用几何方法提升动态物体的识别效果。同时该框架在静态地图的基础上，对动态物体遮挡的图像帧进行背景修复。

Strecke Michael等人^[30]在RGB-D相机上应用了一种新的动态SLAM算法，该算法使用密集的物体层级表示，在局部体积内用有符号距离函数（Signed Distance Function, SDF）表示刚性对象，从而实现跟踪对象与RGB-D图像的直接绑定。该算法同时改进了概率公式，实现数据的直接关联并对运动物体遮挡进行了处理。

国内对动态环境下的SLAM算法也做了大量的研究工作，并提出了很多解决方案。Shuhuan Wen等人^[31]针对传统SLAM建图过程中语义信息过少的情况，提出了一种基于语义拓扑图且具有闭环检测的立体视觉惯性定位系统。该系统使用混合3D点云语义拓扑映射框架，通过提供路径规划地图并同时在动态环境下存储语义信息的方式实现自主导航。具体实施过程中采用最新的Mask R-CNN方法来获取2D语义信息，并将其投影到由立体视觉惯性SLAM构建的半密集3D语义拓扑图上，从而实现物体的语义判别。

Yongbao Ai等人^[32]提出了一种名为动态深度学习的RGB-D SLAM框架,该框架在ORB-SLAM2算法的基础上,增加了动态对象分割和背景修复的功能。利用语义分割和多视图几何来检测运动对象,并在静态的场景图中对已被移动物体遮挡的背景进行修复,从而实现在动态环境中提高定位精度的效果。

Linhui Xiao等人^[33]利用深度学习在对象检测中的优势,提出了一种语义SLAM框架,以解决动态环境中的SLAM问题。该方法基于卷积神经网络,在语义级别上检测环境中的动态对象。针对现有神经检测网络查全率低的问题,提出一种基于相邻帧速度不变性的漏检补偿算法,大大提高了检测的查全率。该系统通过跟踪线程中的选择性跟踪算法处理动态物体的特征点,从而大大减少了由于误匹配而引起的姿态估计误差。

使用深度学习算法和语义分割能够对动态特征起到较好的识别效果,但是这些算法复杂度相对较高,无法保证定位算法运行的实时性。对此,也有一些学者提出了计算相对简单的动态滤除方法,例如北京航空航天大学的魏彤等人^[34]提出了一种基于动态区域剔除的双目视觉SLAM算法。该算法在场景分割的基础上,利用立体视觉识别动态特征,并以此标记出动态区域,从而以较低的计算代价实现了动态场景的识别。

1.3 本文研究内容与组织结构

无人机的实时位姿估计是其准确规划路径、完成飞行任务的重要前提,特别是在室内环境下,面对卫星信号丢失和动态物体干扰等情况,保证无人机自身的准确定位具有重要意义。然而无人机自身续航与载重有限,所携带的传感器必须轻便小巧,同时无人机大多使用嵌入式级别的运算平台,计算资源较为紧俏,设计定位算法也必须满足无人机对于实时性的高要求。本文针对上述应用场景以及种种因素限制,研究能在无人机平台上实时运行且精度较高的定位算法,并予以实现。本文的主要研究内容和组成结构如下:

第一章:介绍研究背景,阐述了无人机在室内动态环境下实现准确定位的意义和面临的挑战,并介绍了当前国内外对于视觉惯导里程计以及动态特征识别滤除算法的研究现状。

第二章:主要介绍了视觉惯导里程计算法中涉及到的基本理论。首先介绍了刚体运动变换中常用的数学表示方法,并对它们的特点进行分析;其次给出了惯性导航坐标系和以针孔相机模型为基础的视觉导航坐标系的定义,并推算不同坐标之间的转换关系;然后针对视觉惯导里程计中涉及到的线性与非线性最小二乘问题,介绍几种常用的求解方法;最后介绍了卡尔曼滤波在定位系统中的具体定义,并结合系统非线性特点引入扩展卡尔曼滤波方法。

第三章:主要研究了组合导航定位中的传感器标定方法。针对相机平面标定方法的弱几何约束以及标定不稳定问题,提出了一种基于三维标定板的相机内参标定理论,并介绍了双目相机之间外参数的标定方法;接着研究了IMU常用的运动测量模型,给出相关误差参数的标定方法,并介绍了IMU与视觉联合标定的相关算法;最后通过标定实验给出传感器的内、外参的标定结果,并评估本文改进标定算法的稳定性与准确性。

第四章：主要研究视觉与 IMU 的紧耦合算法。为保证算法运行的实时性，在前端使用直接法，对图像提取 FAST 角点后使用光流进行跟踪。为降低动态环境的影响，提出一种基于运动分割^[35]的动态特征点识别方法，借助 IMU 的测量信息对图像中的像素点进行运动补偿，并使用改进的 K-means 算法区分出环境中的动态特征并予以剔除。选用计算量更小的扩展卡尔曼滤波作为视觉惯导紧耦合后端框架，为降低运算复杂度，选用相机位姿代替特征点作为系统状态量，并建立多状态约束的观测模型进行状态更新，同时改进相机状态扩增方式，使用 IMU 与视觉两者的融合计算结果来作为新图像帧相机的位姿参数，以降低 IMU 累积误差所带来的影响。在 EuRoC 数据集上评价本文紧耦合算法的定位效果，通过与其它 VIO 算法进行比较，分析本文算法在该数据集上的位姿估计表现。

第五章：主要在无人机平台上对本文算法进行实验验证。介绍了无人机平台的组成以及实验场景，使用无人机平台在实际的动态场景中采集运动数据，评估本文提出的动态特征识别方法的有效性；设计不同的无人机飞行轨迹，根据实验结果对比改进前后 MSCKF 算法的估计轨迹与闭环误差，验证本文算法在定位精度方面的优越性。

第六章：对本文工作进行总结与展望。

第二章 视觉惯性里程计算法相关原理

本论文研究基于滤波的视觉惯性里程计算法，因此本章节介绍视觉里程计（Visual Odometry, VO）和惯性导航相关知识以及其中涉及的数学理论。首先，对三维空间中的刚体运动进行介绍，并引入表示位姿的几种数学概念；其次，介绍了惯性导航坐标系和以针孔相机模型为基础的视觉坐标系，并阐述各个坐标系之间的相互转换关系；随后，对于视觉里程计中涉及到的线性与非线性最小二乘问题，介绍几种常用的求解方法；最后，描述经典的扩展卡尔曼滤波（EKF）算法，对其中的公式进行推导，从而为后续紧耦合的视觉惯性里程计算法奠定基础。

2.1 刚体运动模型

刚体在三维空间中运动时，需要描述它的位置和姿态。位置描述相对容易，可以直接使用平移量 t 进行表示。对于刚体的姿态，需要使用其在运动过程中发生的旋转变换进行表示，常用的旋转表达方式有如下几种：

（1）欧拉角

对于刚体旋转而言，最直观的描述方式是使用欧拉角^[36]进行表述。对于刚体在三维空间中的任一旋转而言，都可以将其分解为三次绕不同轴的旋转，从而可以用三个分离的转角对旋转进行描述。对于不同的分解方式，欧拉角有不同的定义方法，最常用的欧拉角是 $rp\gamma$ 角。在以右手系规则建立的坐标系中，先将运动绕 Z 轴分解出旋转分量，得到偏航角 Yaw ，再分解出绕 Y 轴旋转的俯仰角 $Pitch$ ，最后分解出绕 X 轴旋转的横滚角 $Roll$ ，各个角与坐标轴的对应关系如图 2-1 所示。欧拉角的表示简单直观，但是却不适合用于数值计算。因为欧拉角存在万向锁问题，即在俯仰角为 $\pm 90^\circ$ 时，第一次旋转将和第三次旋转共用一条坐标轴，从而使系统丢失了一个自由度，产生奇异性问题，这会导致其不适用于插值和迭代，从而也无法应用于后端优化和滤波算法中。但是欧拉角在理解物体的运动姿态、验证算法的准确性方面依然有着难以替代的作用。

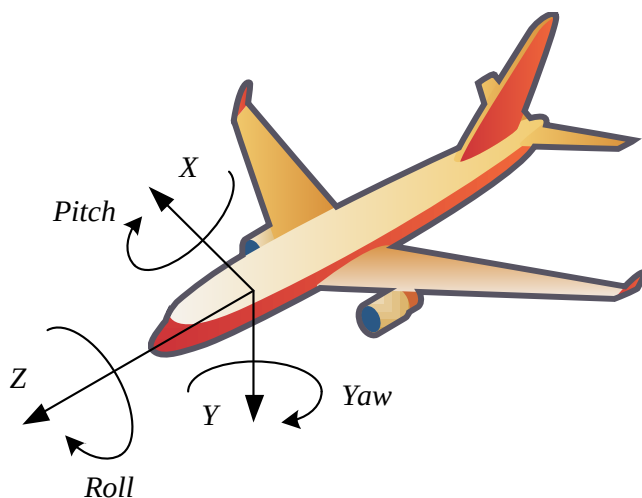


图 2-1 欧拉角示意图

(2) 旋转矩阵

刚体运动中, 物体不会产生形态上的变化, 所以运动前后其所在的坐标系在长度和夹角上始终保持一致。设某一坐标系的正交基 (e_1, e_2, e_3) 经过旋转变换后变为 (e'_1, e'_2, e'_3) , 空间中某一静止点在前后两坐标系中的坐标分别为 $[a_1, a_2, a_3]^T$ 和 $[a'_1, a'_2, a'_3]^T$, 则有如下定义:

$$\begin{bmatrix} e_1 & e_2 & e_3 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} e'_1 & e'_2 & e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \quad (2-1)$$

对上述等式的两边同乘以 $\begin{bmatrix} e_1^T & e_2^T & e_3^T \end{bmatrix}$, 使等式左边系数变成单位矩阵, 则有:

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} e_1^T e'_1 & e_1^T e'_2 & e_1^T e'_3 \\ e_2^T e'_1 & e_2^T e'_2 & e_2^T e'_3 \\ e_3^T e'_1 & e_3^T e'_2 & e_3^T e'_3 \end{bmatrix} \begin{bmatrix} a'_1 \\ a'_2 \\ a'_3 \end{bmatrix} \quad (2-2)$$

等式右边的系数矩阵即为旋转矩阵^[37], 一般用 R 进行表示。旋转矩阵描述的是旋转运动本身, 与具体的向量没有必然关系, 换言之, 对于某一旋转变换, 同一刚体上所有点的坐标变换都可以用相同的旋转矩阵进行描述。旋转矩阵有非常好的属性特征, 例如其是正交矩阵、对旋转矩阵求转置即表示相反的旋转变换等等。同时旋转矩阵在表达刚体在空间中的运动时也非常简洁, 结合运动平移量一起考虑, 假设某点 a 在经过一次旋转 R 和一次平移 t 后得到 a' , 则此运动变换可以用式(2-3)进行表示:

$$a' = Ra + t \quad (2-3)$$

但是旋转矩阵也存在缺陷, 其使用了九个参数来描述三个自由度的旋转, 参数使用上略显冗余。

(3) 四元数

四元数^[38]是由数学家汉密尔顿所扩展的复数, 一个四元数由一个实部和三个虚部构成:

$$q = q_0 + q_1 i + q_2 j + q_3 k \quad (2-4)$$

其中 q_0 表示四元数实部的数值, q_1 、 q_2 、 q_3 分别为虚部分量。对于虚部 i 、 j 、 k 而言, 其满足如下关系:

$$\begin{cases} i^2 = j^2 = k^2 = -1 \\ ij = k, ji = -k \\ jk = i, kj = -i \\ ki = j, ik = -j \end{cases} \quad (2-5)$$

当四元数的实部为0时, 称为虚四元数, 其表示三维空间中的某一点。而三维空间中的旋转运动是由单位四元数定义的, 设某个运动绕单位向量 $n = [n_x \ n_y \ n_z]^T$ 旋转了角度 θ , 则这个旋转对应的四元数为:

$$q = \begin{bmatrix} \cos \frac{\theta}{2} & n_x \sin \frac{\theta}{2} & n_y \sin \frac{\theta}{2} & n_z \sin \frac{\theta}{2} \end{bmatrix}^T \quad (2-6)$$

对于空间中的某一点 p (以虚四元数进行表示), 在经历了以四元数 q 对应的旋转后得到点 p' , 则有如下等式:

$$\dot{p} = qpq^{-1} \quad (2-7)$$

这里的乘法为四元数的乘法。由上述描述可知，虽然四元数表达刚体的旋转运动时不够直观，且计算复杂，但其仅用四个参数进行描述，相较于旋转矩阵更加简洁，同时其也不存在欧拉角这样的奇异性问题，便于迭代和插值，很适合应用于后端优化和滤波算法。

2.2 坐标系定义

2.2.1 惯导坐标系

在惯性导航中，通常将坐标系分为惯性坐标系、非惯性坐标系两类。所谓惯性坐标系，是原点取在不动点，而又无转动的参考系。如日心惯性系、地心惯性系等。非惯性坐标系与之相对，主要包括：地球坐标系、地理坐标系、载体坐标系、导航坐标系等。相关坐标系定义如下^[39]：

(1) 地心坐标系：以地球中心为坐标原点 O ， Z 轴沿地球自转轴方向， X 轴是地球绕太阳公转的黄道平面与地球赤道平面的交线， Y 轴以右手规则确定。地心坐标系不与地球固联，它不随地球自转而转动。

(2) 地球坐标系：取地球中心为坐标原点 O ， Z 轴沿地球自转轴方向， X 轴是赤道平面和本初子午面的交线， Y 轴根据右手规则确定。地球坐标系与地心坐标系的区别在于，它是和地球固连的，会和地球一起相对惯性坐标系以地球的自转角速度进行转动。

(3) 地理坐标系：该坐标系的定义有很多种，常用的地理坐标系为“东—北—天”坐标系，即以地球表面某一点为原点 O ， X 轴在当地水平面内指东， Y 轴与当地子午线一致指向正北， Z 轴沿当地垂线垂直向上，总而构成右手坐标系。

(3) 载体坐标系：一般取载体重心为原点 O ， X 轴沿载体横轴向右， Y 轴沿载体纵轴向前， Z 轴根据右手规则确定，垂直向上。载体坐标系与运动物体固联，随其一起相对地球运动。本文将双目相机作为运动载体，建立载体坐标系。

(4) 导航坐标系：导航坐标系用于表示 IMU 在运动过程中的旋转与平移，从而便于得到它在其他坐标系下的位姿参数。本文将导航坐标系的原点 O 定于 IMU 初始时刻的中心， Z 轴指向重力相反的方向， X 轴垂直于 Z 轴向右， Y 轴以右手规则确定。

2.2.2 针孔相机模型

视觉导航算法中，在光学成像上最常采用的成像模型是针孔相机模型^[40]，该模型示意图如图 2-2 所示。现实空间中的三维点 P ，经过相机光心 O 投影后，成像在物理平面上，像点为 P' 。针孔相机模型中涉及四个坐标系：世界坐标系、相机坐标系、图像坐标系、像素坐标系。

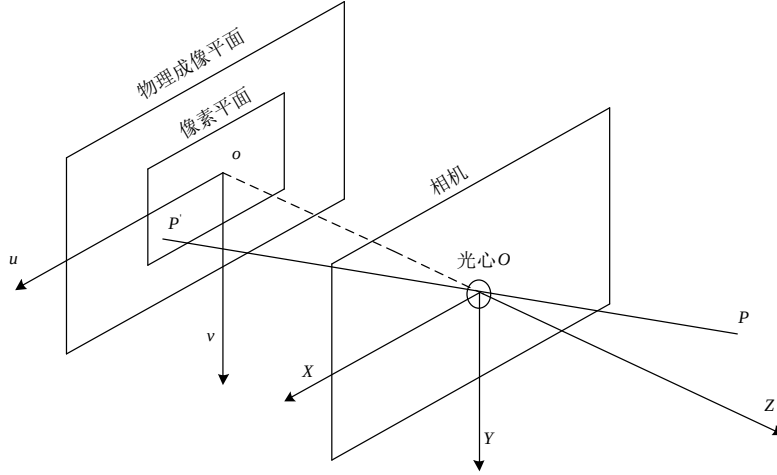


图 2-2 针孔相机模型

相机坐标系：相机坐标系即为图 2-2 中的 $O-XYZ$ 坐标系所示，坐标原点建立在相机的光心处， OZ 轴垂直于像平面，与相机的光轴重合， OX 轴垂直于光轴水平向右， OY 轴垂直向下。相机坐标系描述了空间中的三维坐标点和相机之间的位置关系。

世界坐标系：世界坐标系是视觉导航算法中的基准坐标系，它既可以用来表示真实世界中三维空间的坐标，同时也可以用于描述相机在运动过程中的速度、位姿等运动参数。世界坐标系的定义比较灵活，在相机标定中通常将世界坐标系建立在标定板上，而在视觉导航算法中通常取相机第一帧图像的相机坐标系作为世界坐标系。假设点 P 在世界坐标系下的齐次坐标 P_w 为 $(X_w, Y_w, Z_w, 1)$ ，经过旋转平移变换后可得点 P 在相机坐标系下的坐标 P_c 为 (X_c, Y_c, Z_c) ，则两者有如下对应关系

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_x \\ R_{21} & R_{22} & R_{23} & t_y \\ R_{31} & R_{32} & R_{33} & t_z \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \quad (2-8)$$

其中 R ， t 分别表示世界坐标系和相机坐标系之间的旋转矩阵和平移向量。

图像坐标系：图像坐标系的原点建立在相机成像平面的中心， x 轴和 y 轴分别平行于图像平面的两条垂直边，图像坐标系以物理单位表示像素在图像中的位置。设点 P 在相机坐标系下的坐标为 (X_c, Y_c, Z_c) ，在图像坐标系下的坐标为 (x, y) ，则根据三角形相似关系，可以得到如下等式：

$$\begin{cases} x = f \frac{X_c}{Z_c} \\ y = f \frac{Y_c}{Z_c} \end{cases} \quad (2-9)$$

其中 f 为相机镜头的焦距。将其改写为齐次方程形式，可得：

$$Z_c \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \\ 1 \end{bmatrix} \quad (2-10)$$

该式描述了同一个点在相机坐标系和图像坐标系之间的对应关系。

像素坐标系：图像坐标系描述了三维点和像点之间的对应关系，但实际得到的图像是以像素为单位进行描述的。像素坐标系的建立就是为了描述传感器量化采样得到像素坐标的过程。像素坐标系使用 $o-uv$ 表示，以图像的左上角作为像素平面的原点 o ， u 轴和 v 轴分别与图像坐标系的 x 轴和 y 轴平行。设像素坐标系在量化采样时分别在 u 轴和 v 轴上，缩放了 α 倍和 β 倍，原点相对于图像坐标系平移了 (c_x, c_y) ，则像素坐标和图像坐标之间的关系可以用如下等式进行表示：

$$\begin{cases} u = \alpha x + c_x \\ v = \beta y + c_y \end{cases} \quad (2-11)$$

将等式(2-9)代入式(2-11)，并把 αf 合并为 f_x ， βf 合并为 f_y ，可得像素坐标系和相机坐标系之间点的对应模型：

$$\begin{cases} u = f_x \frac{X_c}{Z_c} + c_x \\ v = f_y \frac{Y_c}{Z_c} + c_y \end{cases} \quad (2-12)$$

综上所述，世界坐标系下的点，经过旋转平移变换、光学投影和采样量化最终得到对应的像素坐标点。此完整变换过程在齐次坐标下可以用式(2-13)进行表示：

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_x \\ R_{21} & R_{22} & R_{23} & t_y \\ R_{31} & R_{32} & R_{33} & t_z \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = KTP_w \quad (2-13)$$

其中 s 代表尺度因子； K 为相机的内参矩阵，包含相机的固有属性参数； T 被称为相机的外参矩阵，包含相机位姿变换的旋转矩阵 R 和平移向量 t 。

2.3 最小二乘问题求解方法

2.3.1 奇异值分解

奇异值分解^[41]，也被称为 SVD 分解，是一种常用的矩阵分解方法，其应用领域也十分广泛，特别是在机器学习领域，多用于推荐系统以及自然语言处理等方面，是很多机器学习算法的基础。SVD 分解最基础的用法是对超定的齐次线性方程组进行求解，对于齐次线性方程 $A * X = 0$ 而言，当系数矩阵 A 的秩大于其本身的列数时，需要求解该方程的最小二乘解，此时需使用 SVD 分解方法。

对于齐次线性方程而言，要获取它的解向量，一般是通过计算系数矩阵的特征值和特征向量来实现。如果系数矩阵是方阵，则可以直接使用线性代数里面的特征值分解方法进行计算，而超定线性方程组的系数矩阵并不是方阵，所以使用 SVD 分解的方法对其进行计算。假设系数矩阵 A 是一个 $m \times n$ 的矩阵，则矩阵 A 的 SVD 分解定义为：

$$A = U \Sigma V^T \quad (2-14)$$

其中 U 是一个 $m \times m$ 的矩阵， Σ 是一个 $m \times n$ 的矩阵，该矩阵只在主对角线上有元素，

其余全为 0，主对角线上的元素也被称为是系数矩阵 A 的奇异值， V 是一个 $n \times n$ 的矩阵。 U 和 V 都是酉矩阵，即满足 $U^T U = V^T V = I$ 。图 2-3 形象地表示了 SVD 的定义：

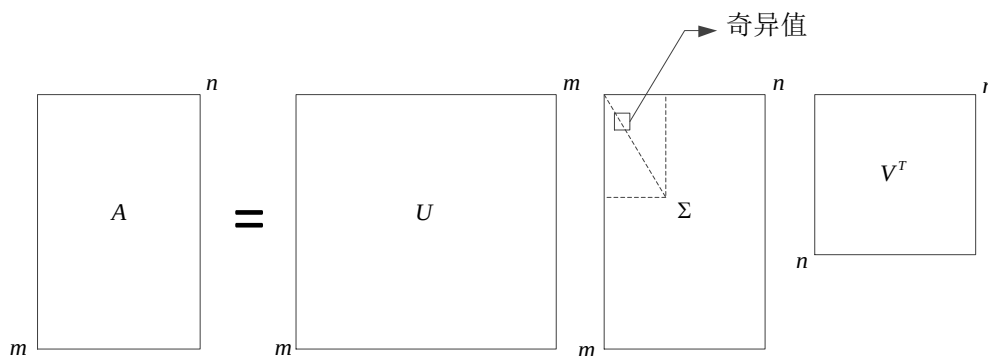


图 2-3 奇异值分解示意图

从 SVD 分解的定义中可以看出，其关键是求取 U ， Σ ， V 。首先为了获取矩阵 V ，将 A 的转置矩阵和 A 做矩阵乘法，得到一个 $n \times n$ 的方阵，由于其是方阵，进而可以对其进行特征分解，得到其特征值 λ 和特征向量 v ，如下所示：

$$(A^T A)v_i = \lambda_i v_i \quad (2-15)$$

得到矩阵 $A^T A$ 的 n 个特征向量后，将其合并为一个 $n \times n$ 的矩阵，该矩阵即为 SVD 分解中所待求的 V 矩阵。矩阵 V 中的每个特征向量被称之为系数矩阵 A 的右奇异向量。

类似地，将 A 和 A 的转置矩阵做矩阵乘法，会得到一个 $m \times m$ 的方阵。同样对其进行特征分解，得到其特征值 λ 和特征向量 u ：

$$(AA^T)u_i = \lambda_i u_i \quad (2-16)$$

将 m 个特征向量合并起来，即构成 SVD 公式分解中的 U 矩阵。 U 中的每个特征向量为系数 A 的左奇异向量。在矩阵 U 和矩阵 V 的特征分解过程中，所求得特征值是一样的，但对应的特征向量不同。

对于 Σ 矩阵而言，其只在对角线上有元素，这些元素由奇异值构成，因此只需要求出每个奇异值 σ 就可以求得 Σ 矩阵。在计算完 U 和 V 矩阵后，奇异值的求法就变得很简便，根据相关证明，特征值和奇异值满足如下关系：

$$\sigma_i = \sqrt{\lambda_i} \quad (2-17)$$

因此，只需对上文所计算得出的 $A^T A$ 的特征值取平方根即可得到奇异值，进而也得到了奇异值矩阵 Σ 。

根据相关理论证明，超定的齐次线性方程组的最小二乘解，即为奇异值矩阵 Σ 中最小的奇异值所对应的 V 中的奇异向量。因此，计算完系数矩阵的 SVD 分解，也就间接得到了齐次线性方程组的最小二乘解。

2.3.2 非线性优化

非线性优化方法建立在求解线性最小二乘的基础上。在求解矩阵方程时，如果是超定的齐次线性方程组，可以采用上文中的 SVD 进行求解。而对于普通的线性方程组，可以采用线性最小二乘的理论进行求解。对于函数模型 $f(x) = Ax$ ，采集它的一组输出

样本 b ，线性最小二乘的目的就是希望通过这组样本数据拟合出函数模型中的未知量，进而将问题转化为求解 $\min \|Ax - b\|^2$ ，显然最理想的情况是求 x 使得 $Ax = b$ 。一般情况下，对系数矩阵 A 求逆即可解出变量 x 来，但往往样本个数要多于函数模型中的未知量，因此 A 多半情况下是不可逆的，此时的处理方式是在方程的左侧同乘 A^T ，进而求矩阵 A 的伪逆形式：

$$A^T Ax = A^T b, \quad x = (A^T A)^{-1} A^T b \quad (2-18)$$

可以看出，对于线性函数模型，可以获取待求变量的解析解，但是在视觉惯导融合的定位算法中，由于涉及到投影模型、旋转矩阵等参数，因此模型常常是非线性的，此时需要使用非线性优化理论计算最优解。假设函数模型 $f(x)$ 此时为任意关于 x 的非线性方程，此时最小化的代价函数为 $\min \|f(x)\|^2$ 。由于问题为非线性的，函数形式过于复杂，从而不便于直接对函数进行求导来获得全局最小值。所以非线性优化的思路是从一个较为精准的初始值出发，通过迭代计算寻找到函数的局部最小值，以该局部最小值作为函数模型的最优解。在每一次迭代中，对变量 x 寻求一个增量 Δx ，使得 $\|g(x + \Delta x)\|_2^2$ 小于 $\|g(x)\|_2^2$ ，当增量 Δx 小于一定程度时，则认为 x 已经到达了局部最小值的位置。由此可见，非线性优化求解的关键是计算合适的 Δx 进行迭代优化，该过程也被称为梯度下降方向搜索。

为了求解下降梯度，最直观的想法是直接对代价函数进行求导，此类方法以急速下降法为代表，该方法流程简单、收敛快，但是对非线性方程求导并不容易，特别是求二阶导时，计算极其复杂，因此该方法并不常用。常用的梯度下降搜索方法有两种，基于线搜索的高斯牛顿方法和基于信任域理论的列文伯格马尔夸克算法^[42]（L-M 方法）。高斯牛顿的基本思想是将函数 $F(x + \Delta x)$ 在 Δx 处进行一阶泰勒展开：

$$F(x + \Delta x) = F(x) + J(x)\Delta x \quad (2-19)$$

其中 $J(x)$ 为 $F(x)$ 对 Δx 的一阶导数，也被称为雅可比（Jacobian）矩阵。将式(2-19)代入到代价函数中展开可得

$$\frac{1}{2} \|F(x) + J(x)\Delta x\|^2 = \frac{1}{2} (\|F(x)\|^2 + 2F(x)^T J(x)\Delta x + \Delta x^T J(x)^T J(x)\Delta x) \quad (2-20)$$

为了获取使得式(2-20)达到最小值的增量 Δx ，将该方程直接对 Δx 进行求导并令导数为 0：

$$J(x)^T J(x)\Delta x = -J(x)^T F(x) \quad (2-21)$$

式(2-21)也被称为高斯牛顿方程，其形式与线性最小二乘中的正规方程形式十分类似，因此也可以被称为正规方程。理论上，要求解非线性优化中的增量 Δx ，只需要求解高斯牛顿方程中 $J(x)^T J(x)$ 矩阵的逆矩阵，并将其乘到等式右侧即可。但是根据梯度下降理论可知，只有当 $J(x)^T J(x)$ 矩阵是正定矩阵的时候，才能确保此方法最终所求解的 x 为函数模型的局部最小值，但是在实际中该矩阵很有可能不是正定矩阵，因此高斯牛顿法在计算过程中可能会出现算法不收敛或结果不准确的情况。

为了解决高斯牛顿计算结果不收敛的问题，在该方法的基础之上提出了 L-M 方法。

L-M 方法在正规方程的左侧人为添加了一个阻尼系数 λ ：

$$(H + \lambda)\Delta x = g \quad (2-22)$$

阻尼系数的引入一方面可以确保正规方程左侧的系数矩阵为正定矩阵，从而保证迭代算法的收敛性。另一方面其也能控制 Δx 的增加幅度，将其限制在一定的信任区域内，并能根据迭代情况对信任域进行大小控制。如果当前解得的 x 离最终的局部最小值距离较远时，则可以适当增加阻尼系数，此时 L-M 方法更接近于急速下降法，使得下一步迭代步长更为激进；反之，则减小阻尼系数，使 L-M 方法更接近于高斯牛顿法。L-M 方法虽然弥补了高斯牛顿方法的缺点，但是计算流程更为复杂，同时阻尼系数需要人为进行设定，一般会将其设定为默认值，例如拉格朗日乘子^[43]等等。

2.4 EKF 滤波优化算法

在视觉与惯导融合的 SLAM 定位系统中，惯导可以直接获取系统的加速度、角速度等运动参数，视觉传感器则可以通过获取对外界环境的观测信息，来计算系统的运动状态。因此，对于系统在某一时刻的位姿状态，其由运动信息和观测信息两方面组成，所以通常使用运动方程和观测方程来联合描述定位状态。假设在一段连续的时间内，传感器观测到了外部环境中的 M 个坐标点，分别记为 $y_1, \dots, y_M$ ，系统在其中 N 个离散时刻位姿记为 $x_1, \dots, x_N$ ，则可以构建如下的运动方程和观测方程：

$$\begin{cases} x_k = f(x_{k-1}, u_k) + w_k, & k=1, \dots, N, \quad j=1, \dots, M \\ z_{k,j} = h(y_j, x_k) + v_{k,j} \end{cases} \quad (2-23)$$

其中 $f(\bullet)$ 表示运动方程的传感器模型， $h(\bullet)$ 为观测方程的传感器模型， u 为系统传感器记录的相关运动状态的输入，例如加速度、角速度等等， w 为运动传感器模型噪声， v 为观测传感器模型噪声，通常假设这两部分噪声服从 0 均值的高斯分布， $z_{k,j}$ 表示在 k 时刻，传感器观测到了第 j 个坐标点。

根据上述运动模型，要获取时刻 k 处系统的状态估计，就是在初始时刻的系统状态基础上，结合已知传感器的运动输入数据 u 和观测数据的 z ，计算状态量 x 的条件概率分布：

$$P(x_k | x_0, u_{1:k}, z_{1:k}) \quad (2-24)$$

在概率学上，求解该方程需要使用贝叶斯公式进行展开：

$$P(x_k | x_0, u_{1:k}, z_{1:k}) \propto P(z_k | x_k) P(x_k | x_0, u_{1:k}, z_{1:k-1}) \quad (2-25)$$

该式右侧第一项称为似然，可以通过观测方程得出。右侧第二项则为先验部分，从中可以看出当前的状态估计是与过去的所有状态估计息息相关的。处理先验这一部分主要有两种思路，一种是使用以非线性优化理论为主体的优化框架，如 BA 图优化^[44]、因子图优化^[45]等，此类方法会严格根据之前的所有状态来估计当前时刻的运动状态，这种方法准确度较好，但是对计算量的要求比较高。另一种是基于扩展卡尔曼滤波（EKF）的滤波优化方法，根据马尔可夫性假设，可以认为当前时刻的运动状态只与前一帧的运动状态相关，进而可以去除之前状态量的影响，从而降低系统的运算负荷，保证定位的

实时性。

扩展卡尔曼滤波算法是在卡尔曼滤波算法的基础上演变而来的，普通的卡尔曼滤波算法能够处理简单的线性高斯系统，假设视觉惯导里程计是一个线性系统，则式(2-23)可以展开为如下形式：

$$\begin{cases} x_k = A_k x_{k-1} + u_k + w_k \\ z_k = C_k x_k + v_k \end{cases} \quad k=1, \dots, N \quad (2-26)$$

其中 A_k 和 C_k 为系数矩阵。卡尔曼滤波算法在求解状态时主要分为预测和更新两步，预测主要是求解 k 时刻的先验状态分布及其协方差，若已知 $k-1$ 时刻的最大后验状态估计及其协方差，则预测方程可以如式(2-27)进行计算：

$$P(x_k | x_0, u_{1:k}, z_{1:k-1}) = N(A_k \hat{x}_{k-1} + u_k, A_k \hat{P}_{k-1} A_k^T + R) \quad (2-27)$$

其中 R 为运动传感器输入噪声的协方差。更新部分主要是获取系统状态的后验概率，具体的计算方式如下：

$$\begin{cases} \hat{x}_k = \bar{x}_k + K(z_k - C_k \bar{x}_k) \\ \hat{P}_k = (I - KC_k) \bar{P}_k \end{cases} \quad (2-28)$$

其中 K 被称为卡尔曼增益，由以下公式计算得出：

$$K = \bar{P}_k C_k^T (C_k \bar{P}_k C_k^T + Q_k)^{-1} \quad (2-29)$$

在得出卡尔曼滤波的预测方程和更新方程后，可以由 $k-1$ 时刻的状态分布推导出 k 时刻的状态分布。

然而事实上，视觉惯导里程计中的运动方程和观测方程通常都不可能是线性函数，因此不能直接使用卡尔曼滤波的方式进行求解。针对非线性系统，需要使用扩展卡尔曼滤波进行优化，其基本思路是对系统的运动方程和观测方程进行一阶泰勒展开，只保留其中的一阶项，从而将一个非高斯分布近似为高斯分布。在 k 时刻，对运动方程和观测方程进行一阶泰勒展开可得：

$$\begin{cases} x_k \approx f(\hat{x}_{k-1}, u_k) + F(x_{k-1} - \hat{x}_{k-1}) + w_k \\ z_k \approx h(\bar{x}_k) + H(x_k - \bar{x}_k) + n_k \end{cases} \quad (2-30)$$

其中 $F(\bullet)$ 和 $H(\bullet)$ 为偏导数。从而可以得出新的预测更新方程，以及卡尔曼增益：

$$\begin{cases} \bar{x}_k = f(\hat{x}_{k-1}, u_k), \bar{P}_k = F \hat{P}_{k-1} F^T + R_k \\ K_k = \bar{P}_k H^T (H \bar{P}_k H^T + Q_k)^{-1} \\ \hat{x}_k = \bar{x}_k + K_k (z_k - h(\bar{x}_k)), \hat{P}_k = (I - K_k H) \bar{P}_k \end{cases} \quad (2-31)$$

2.5 本章小结

本章节主要介绍了视觉惯导融合导航定位系统的基础理论，首先定义了惯性导航和视觉导航下的坐标系，并具体阐述了视觉投影模型中各坐标系之间的对应关系；随后介绍了几种在算法处理流程中计算最小二乘的优化方法，包括处理超定齐次线性方程组的奇异值分解方法以及针对非线性最小二乘的非线性优化理论，其中非线性优化

理论中的 L-M 方法使用最为广泛，本文在使用非线性优化方法时也以 L-M 方法为主；最后介绍了滤波优化算法，建立了以运动方程和观测方程为主体的系统状态模型，并在此基础上推导了扩展卡尔曼滤波的预测更新方程。

第三章 传感器标定方法

传感器的准确标定是融合定位算法准确可靠的前提,准确地标定传感器除了能提升原始定位信息的准确度以外,也有助于减少算法后端的优化时间,提升系统的运行效率。对于融合了双目视觉与惯导的定位系统而言,首先需要单独标定所有相机的内参,然后需要获取双目相机左、右目之间的相对外参。对于 IMU 而言,由于其测量值中包含噪声,所以需要对这些噪声项进行标定。在单独标定完相机和 IMU 的参数之后,需要获取相机和 IMU 的相对外参,以便后续更好地对两种传感器获取的数据进行融合。

3.1 改进的相机标定算法

在讨论相机内参标定算法时,沿用上文 2.2.2 节的针孔相机模型。对于标定单目相机内参而言,常用的相机标定算法有很多,使用最为广泛的是张氏相机标定方法^[46]。该方法鲁棒性较好,操作简便,仅利用一块平面棋盘标定板即可完成相机的完整标定。但是该方法中由于标定角点均位于同一平面,几何约束性较弱,同时该方法在计算过程中将三维点的坐标信息弱化到两个维度,从而导致坐标维度信息缺失,限制了标定的精确性与稳定性。为了更好地改善这一问题,本文在张氏标定方法的基础上,提出一种基于三维棋盘格的相机标定方法。

张氏相机标定方法选用平面棋盘格作为相机标定板,在标定过程中,假定棋盘格所在的平面为世界坐标系中 $z_w=0$ 的平面,将棋盘格上的某一个角点设置为世界坐标系的原点,则理论上只要知道棋盘格每一个格子的真实物理尺寸就能得出所有棋盘格内角点的世界坐标。同时棋盘格内角点的像素坐标可以通过图像处理的方法得出,从而可以借助于平面单应性^[47]、旋转矩阵尺度特性、非线性优化等理论求解出相机的内外参数,并对其进行优化。

可以发现,在求解过程中,由于已经假定棋盘格上所有点的 z_w 为 0,所以在使用平面棋盘格进行相机标定时,未使用 z_w 这一维度,导致在求解标定参数时忽略了部分有用信息。因此,使用该方法进行相机标定时,虽然简单易操作,但是其标定精度还有进一步提升的空间。对此,本文在张氏相机标定方法的基础上提出了一种基于三维标定板的相机标定方法。该方法需要使用图 3-1 所示的立体棋盘格,利用其上角点的三维坐标来对相机进行标定,从而在计算过程中保留坐标点 z_w 维度的信息。

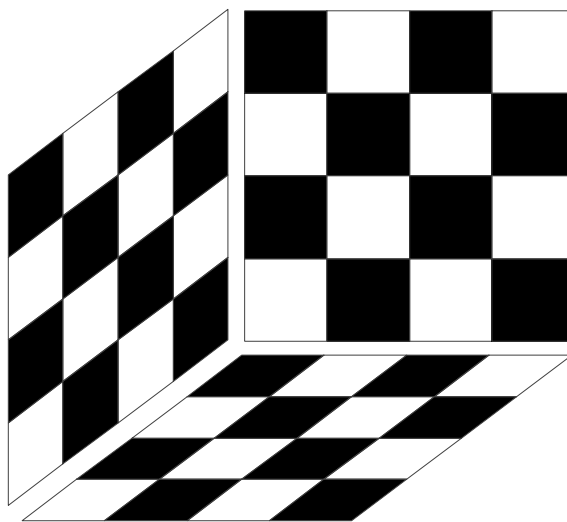


图 3-1 三维棋盘格示意图

为了求解相机的内参矩阵，首先将式(2-13)中的 KT 矩阵乘积用如下矩阵表示：

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{14} \\ h_{21} & h_{22} & h_{23} & h_{24} \\ h_{31} & h_{32} & h_{33} & h_{34} \end{bmatrix} \quad (3-1)$$

再将式(2-13)按行展开：

$$\begin{cases} u = \frac{h_{11}x_w + h_{12}y_w + h_{13}z_w + h_{14}}{h_{31}x_w + h_{32}y_w + h_{33}z_w + h_{34}} \\ v = \frac{h_{21}x_w + h_{22}y_w + h_{23}z_w + h_{24}}{h_{31}x_w + h_{32}y_w + h_{33}z_w + h_{34}} \end{cases} \quad (3-2)$$

根据分析， H 矩阵中有 12 个未知数，因此至少需要 12 条方程求解。一对世界坐标和图像像素坐标的匹配点可以构建式(3-2)中的两条方程，原则上至少需要六对匹配点来求解出 H 矩阵。在实际求解过程中，为了排除测量噪声对求解结果的影响，往往使用 n 对匹配点 ($n > 6$) 来解算 H 。所以需将式(3-2)以矩阵相乘的形式改写为齐次方程组，再使用奇异值分解 (SVD) 的方法求解矩阵 H 。

同时由式(2-13)可知， H 矩阵还可以展开为如下形式：

$$H = KT = \begin{bmatrix} f_x R_{11} + c_x R_{31} & f_x R_{12} + c_x R_{32} & f_x R_{13} + c_x R_{33} & f_x t_x + c_x t_z \\ f_y R_{21} + c_y R_{31} & f_y R_{22} + c_y R_{32} & f_y R_{23} + c_y R_{33} & f_y t_y + c_y t_z \\ R_{31} & R_{32} & R_{33} & t_z \end{bmatrix} \quad (3-3)$$

为了从中解出相机的内参系数，先假设像素坐标系的原点位于图像的正中心，即 $c_x = a/2, c_y = b/2$ ，其中 a, b 分别为相机图像的长和高。借助 H 矩阵的第三行，消去第一、二行中的 c_x 和 c_y ，得到如下矩阵：

$$H' = \begin{bmatrix} f_x R_{11} & f_x R_{12} & f_x R_{13} & f_x t_x \\ f_y R_{21} & f_y R_{22} & f_y R_{23} & f_y t_y \\ R_{31} & R_{32} & R_{33} & t_z \end{bmatrix} = \begin{bmatrix} h'_{11} & h'_{12} & h'_{13} & h'_{14} \\ h'_{21} & h'_{22} & h'_{23} & h'_{24} \\ h'_{31} & h'_{32} & h'_{33} & h'_{34} \end{bmatrix} \quad (3-4)$$

将式(2-13)中的 T 矩阵表示为列向量的形式： $[r_1, r_2, r_3, t]$ ，其中前三列为旋转向量，

最后一列为平移向量。旋转向量有两个特性：(1) 两个旋转向量点积为 0；(2) 旋转向量模相等，均为 1。由此根据一张图像可以列出五条方程：

$$\begin{cases} r_1^T r_2 = 0, & r_1^T r_3 = 0, & r_2^T r_3 = 0 \\ r_1^T r_1 = r_2^T r_2, & r_1^T r_1 = r_3^T r_3 \end{cases} \quad (3-5)$$

以等式 $r_1^T r_2 = 0$ 为例，将其用 T 矩阵中的元素进行展开：

$$R_{11}R_{12} + R_{21}R_{22} + R_{31}R_{32} = 0 \quad (3-6)$$

根据式(3-4)可将式(3-6)改写为

$$h_{11}'h_{12}' \frac{1}{f_x^2} + h_{21}'h_{22}' \frac{1}{f_y^2} + h_{31}'h_{32}' = 0 \quad (3-7)$$

以此类推，则式(3-5)可以写成如下矩阵相乘的形式：

$$\begin{bmatrix} h_{11}'h_{12}' & h_{21}'h_{22}' & h_{31}'h_{32}' \\ h_{11}'h_{13}' & h_{21}'h_{23}' & h_{31}'h_{33}' \\ h_{12}'h_{13}' & h_{22}'h_{23}' & h_{32}'h_{33}' \\ h_{11}'^2 - h_{13}'^2 & h_{21}'^2 - h_{23}'^2 & h_{31}'^2 - h_{33}'^2 \\ h_{11}'^2 - h_{13}'^2 & h_{21}'^2 - h_{23}'^2 & h_{31}'^2 - h_{33}'^2 \end{bmatrix} \begin{bmatrix} \frac{1}{f_x^2} \\ \frac{1}{f_y^2} \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad (3-8)$$

同样采用 SVD 分解的方法对式(3-8)进行求解，从而算出 f_x 和 f_y 的初值。

获取相机内参初值后可以将内参系数代回式(3-3)求取相机的外参系数，并使用张氏标定论文中的方法求解相机畸变模型。由于在上文计算过程中对某些参数进行了近似估计，所以上述结果并不能作为标定的最终结果。为了获取更为精确的结果，使用非线性优化方法对相机的内、外参进行优化。构建优化方程：

$$\min \left(\sum_{i=1}^n \|U_i - U_i'(K, R_i, t_i, k_1, k_2)\|^2 \right) \quad (3-9)$$

其中 k_1, k_2 为畸变系数， U 表示运用图像处理方法提取出的棋盘点的像素坐标， $U'(\bullet)$ 表示通过相机内外参和畸变系数计算像素坐标的函数。使用非线性优化中的 L-M 方法对式(3-9)进行不断迭代优化，直至收敛，以此作为标定参数的最终结果。

3.2 双目相机外参标定算法

双目相机需要标定的空间位置主要是从左相机坐标系变换到右相机坐标系的旋转矩阵 R_r 和平移向量 t_r 。假设空间中一点 P 的世界坐标为 P_w ，其对应的左相机坐标系下的坐标为 P_l ，右相机坐标系下的坐标为 P_r ，则可以建立等式：

$$\begin{cases} P_l = R_{wl}P_w + t_{wl} \\ P_r = R_{wr}P_w + t_{wr} \end{cases} \quad (3-10)$$

其中 R_{wl} 和 R_{wr} 分别表示世界坐标系到左、右相机坐标系的旋转矩阵， t_{wl} 和 t_{wr} 表示世界坐标系到左、右相机坐标系的平移向量。同时 P_l 和 P_r 之间具有如下关系：

$$P_r = R_{lr}P_l + t_{lr} \quad (3-11)$$

其中 R_{lr} 表示左相机坐标系到右相机坐标系的旋转矩阵, t_{lr} 表示左相机坐标系到右相机坐标系的平移向量。上述涉及到的坐标系以及旋转平移量之间的关系如图 3-2 所示:

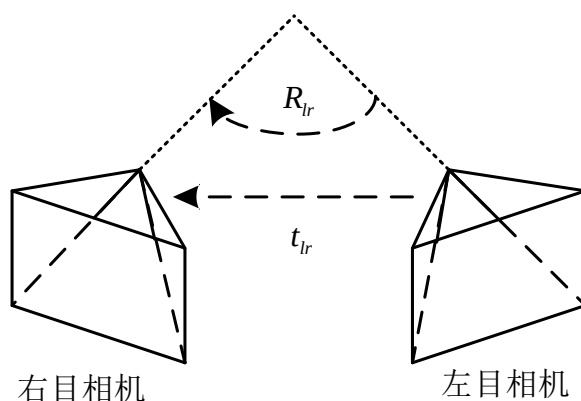


图 3-2 双目相机外参示意图

联立式(3-10)和(3-11)可得:

$$R_{wr}P_w + t_{wr} = R_{lr}(R_{wl}P_w + t_{wl}) + t_{lr} \quad (3-12)$$

通过观察:

$$\begin{cases} R_{lr} = R_{wr}R_{wl}^T \\ t_{lr} = t_{wr} - R_{lr}t_{wl} \end{cases} \quad (3-13)$$

其中 R_{wl} 、 R_{wr} 、 t_{wl} 、 t_{wr} 四个变量可以用 3.1 节中对单目的标定方法获得, 一般双目相机的左、右相机的相对位置是固定的, 但在具体的标定过程中, 每拍摄一组图像, 就可以求出一对 R_{lr} 和 t_{lr} , 所以为了使标定更为准确, 可以采用迭代算法对这组数值进行优化。

3.3 IMU 误差分析和标定算法

惯性测量单元 IMU 内部一般含有两种元器件: 三轴正交的加速度计和三轴正交的陀螺仪。加速度计可以通过对一段时间内的加速度进行积分获取自身所处的位置, 陀螺仪则不断地记录自身旋转角度从而能获取自身的方向与姿态。不同的使用场景对 IMU 的精度有着不同的要求, 在视觉惯导定位系统中, 对 IMU 的精度要求不高, 同时高精度的 IMU 重量、体积、功耗等也往往更大。因此在此种运用场景中, 优先考虑使用消费级 MEMS 类型的 IMU。市面上常见的消费级 IMU 在工作过程中, 往往会受到温度、噪声等因素的影响, 所以在使用前需要对 IMU 的测量误差进行相应的标定。本节先分析影响陀螺仪和加速度计精度的误差组成, 建立 IMU 的测量模型, 随后使用相应的标定算法对模型中的相关参数进行标定。

3.3.1 IMU 测量模型

对于消费级的 IMU 而言, 其应用场景对精度的要求相对较低, 因此在建立 IMU 的测量模型^[48]时, 可以适当简化, 仅引入随机过程误差和噪声项。IMU 的测量模型由式(3-14)进行表示:

$$\begin{cases} \hat{a} = a + b^a + n^a \\ \hat{\omega} = \omega + b^g + n^g \end{cases} \quad (3-14)$$

式中, a 、 ω 分别表示加速度、角速度的期望值; b 表示随机游走的零偏, n 为高斯白噪声, 上标 g 、上标 a 表示该值是陀螺仪、加速度计的相关值; 等号左边为加速度和角速度的实际测量值。该式中加速度的测量值中包含了重力加速度。

高斯白噪声是一种波动剧烈的测量白噪声, 一般是由 IMU 内部 AD 转换等电路元件在工作时所产生。白噪声的波动频率要远远比 IMU 自身的采样频率来得高, 这会导致 IMU 的测量值中包含一个零均值方差恒定的不相关随机变量序列。对加速度计的影响具体表现为白噪声在一个零均值的位置上随机游走, 陀螺仪则在一个零均值的角度上漂移。前者的标准差大小正比于时间 $t^{(3/2)}$, 后者则与时间的平方根成正比。

随机游走零偏是 IMU 的一种内部误差, 受到机械结构、温度等物理因素的影响, 其描述的是在固定条件下 MEMS 类型 IMU 偏置的变化情况, 也即零偏稳定性, 其标准差正比于时间的平方根。理论上, 随机游走的零偏偏差会随时间产生波动, 每一次新的随机游走误差都会在上一个误差值的基础上随机变化, 这会导致设备的零偏不确定性会无限增大。但是实际情况中, 随机游走的误差限定在某一范围之内, 且零偏的随机游走模型在短时间内与实际情况较为吻合。所以在标定过程中, 取在不同时间下的最小平均变化量作为随机游走零偏值。

3.3.2 IMU 标定算法

IMU 在运动过程中, 其测量值和期望值会存在一定的差异, 当对其进行多次测量后, 这种差异会以一种固定的形式表现出来, 这种差异被定义为 IMU 的结构误差。标定 IMU 的目的就是为了能补偿 IMU 测量输出中的结构误差, 以使输出量更接近于期望值^[49]。IMU 标定算法的总体思路就是在 IMU 保持静置的状态下, 标定出陀螺仪和加速度计的各噪声项, 从而得到 IMU 测量输出中的结构误差, 以在后续系统运行过程中, 纠正 IMU 的相关测量值。

当 IMU 静置时, 加速度计的测量值二范数就是其所处位置的重力加速度值, 这一特性可以作为加速度计标定的依据原理。基于此, 可以使用优化误差的方法对加速度计进行标定, 优化项即为二范数与重力加速度二范数差的平方和。根据式(3-14)的测量模型方程, 将加速度计高斯噪声和零偏投影到三个坐标轴上, 构建待求解向量:

$$\zeta^{acc} = [b_x^a \quad b_y^a \quad b_z^a \quad n_x^a \quad n_y^a \quad n_z^a] \quad (3-15)$$

根据式(3-15)改写式(3-14)中的状态方程:

$$\hat{a} = h(a, \zeta^{acc}) = a + b^a + n^a \quad (3-16)$$

待优化的代价函数如式(3-17)所示:

$$L(\zeta^{acc}) = \arg \min_{\theta^{acc}} \sum_{k=1}^M \left(\|g\|^2 - \|h(a_k^s, \zeta^{acc})\|^2 \right)^2 \quad (3-17)$$

其中 g 表示当地的重力向量; M 表示静置状态下加速度计采集的样本数。使用 L-M 等非线性优化方法求解式(3-17)即可获得加速度计的噪声和零偏量。

陀螺仪噪声的确定需要借助于 Allan 方差^[50]。Allan 方差是一种时域分析技术，其通过相关时间来区分不同噪声的特征，从而能相对简便地表示不同误差源的统计特性。此技术可以用于包括 IMU 在内的多种仪器的噪声研究。Allan 方差既可以定义为角速度，也可以定义为角度，这里以角度来计算 Allan 方差。

利用陀螺仪数据计算 Allan 方差时，同样使 IMU 以静止状态保持一段时间，在这段时间内假设共有 N 个样本信号，每个样本的采样周期为 τ_0 ，陀螺仪的输出角速度为 $\psi(t)$ 。则陀螺仪的输出角度 θ 表示为：

$$\theta(T) = \int_0^T \psi(t) dt \quad (3-18)$$

其中 $T = k\tau_0$ ， k 取值从 1 到 N ，即有 N 个 θ 值。Allan 方差函数由式(3-19)进行计算：

$$\sigma^2(\tau) = \frac{1}{2\tau^2(N-2m)} \sum_{k=1}^{N-2m} (\theta_{K+2m} - 2\theta_{K+m} - \theta_K)^2 \quad (3-19)$$

其中 m 是相邻的样本集群个数， $\tau = m\tau_0$ ，相关理论表明，不同的 τ 域对应着不同的随机误差，且各误差项之间互不相关，因此要分离出各项误差只需在 Allan 方差与相关时间双对数图上寻找对应的斜率即可。

3.4 IMU 和相机联合标定算法

相机和 IMU 之间的外参标定主要是获取两者之间较为精确的相对旋转。在运动过程中，相机可以通过特征匹配解算出相机坐标系下的旋转序列，同时 IMU 通过积分可以求出 IMU 坐标系下旋转序列，对于刚体运动而言，只要将两者求出的旋转序列一一对应，即可求解相机和 IMU 之间的相对旋转。假设在第 k 帧和第 $k+1$ 帧之间，相机求解出的旋转为 $R_{c_{k+1}}^{c_k}$ ，IMU 求解出的旋转为 $R_{b_{k+1}}^{b_k}$ ，相机与 IMU 之间的旋转为 R_c^b ，则可以构建如下等式：

$$R_{b_{k+1}}^{b_k} \cdot R_c^b = R_c^b \cdot R_{c_{k+1}}^{c_k} \quad (3-20)$$

该方程的推导来源于经典的手眼标定原理^[51]，其中等式左边左乘了 IMU 坐标系下的旋转分量，右边右乘了相机坐标系下的旋转分量。进一步将其展开为四元数表示形式：

$$[Q_1(q_{b_{k+1}}^{b_k}) - Q_2(q_{c_{k+1}}^{c_k})] \cdot q_c^b = Q_{k+1}^k \cdot q_c^b = 0 \quad (3-21)$$

其中 Q_1 和 Q_2 是四元数对应的矩阵形式。标定过程中相机记录连续的图像帧，通过特征点可以对这些图像帧进行匹配，多对匹配图像可以构造如下的线性方程组：

$$\begin{bmatrix} w_1^0 \cdot Q_1^0 \\ w_2^1 \cdot Q_2^1 \\ \dots \\ w_N^{N-1} \cdot Q_N^{N-1} \end{bmatrix} \cdot q_c^b = Q_N \cdot q_c^b \quad (3-22)$$

这里的 N 表示图像帧数， w_{k+1}^k 表示每一帧的权重，作用是降低退化运动所导致的错误对应关系的影响。通过 SVD 分解，求解出 Q_N 的最小奇异值对应的右奇异向量即为所求的旋转。因为 N 在标定过程中是动态增加的，所以需要人为对旋转标定设置终止条件。

通常情况下,如果 SVD 分解结果的第二小奇异值大于设置的某个阈值时,则认为外参旋转量的标定求解达到终止条件。在求解出 IMU 与相机之间相对旋转后,进而可以通过最小化观测误差的方法优化出两者之间的相对平移量,具体优化过程可以参看相关论文[52]。

3.5 传感器标定实验

3.5.1 相机内参标定实验与分析

(1) 实验流程

为了评估三维标定方法的准确性与稳定性,在对相机标定内参时,使用基于三维标定板的相机标定方法,并将结果与传统平面相机标定方法进行对比分析。进行三维标定时,为了保证相机能在视野中同时获取三张标定板的图像,制作如图 3-3(a)所示的三维标定板,任意两块标定板之间相互垂直,成“类墙角”型。每块标定板上所用大小相同的黑白棋盘格,格子数均为 7×5 ,各有 24 个内角点,每一块棋盘格大小为 $3 \times 3 \text{cm}$ 。实验所标定的相机如图 3-3(b)所示,是小觅公司生产的 MYNTEYE-S1030,该相机为双目相机,分辨率为 752×480 ,像素尺寸为 $6.0 \times 6.0 \mu\text{m}$,官方给定焦距为 2.1mm ,双目基准线为 120mm 。在分析三维标定方法优劣时,仅以左目相机为例进行标定与对比分析。在实验过程中将相机设置为自动曝光模式,采集频率设置为 25Hz 。



(a) 三维标定板



(b) 标定相机

图 3-3 标定板和相机示意图

为了更好地获取标定板上棋盘格点在三维世界中的真实坐标,使用全站仪获取三维标定板上所有 72 个内角点的世界坐标。所用全站仪为南方测绘公司的 NTS-340R6A 全站仪,其距离测量精度为 $\pm 2 \text{mm}$,角度测量精度为 2° 。保持三维标定板固定不动,任意移动相机获取不同视角下的标定板图像,每张图像均需包含三张棋盘格的所有内角点信息。标定过程中,为了使每一个角点的世界坐标和像素坐标一一对应,准确获取棋盘角点的顺序很重要,因此需要采取恰当的方式对所有的棋盘格内点进行标号。本实验在提取第一张图片中的角点像素坐标后,以该识别顺序对棋盘格点进行标号。在处理剩余图片时,借助于光流跟踪的方法,自动将其它图片中的棋盘格点与第一幅图像中的点对应,从而避免了重复的人工选点,确保角点识别顺序的准确。

同时为了进行比较分析,将三维标定的实验结果与 MATLAB 提供的平面标定结果

进行对比。两种方法所用标定板大小尺寸相同,因此平面标定板中总共仅有 24 个棋盘格点。为了排除棋盘格点数对标定结果的影响,同时为了保留部分角点用于投影误差分析,在三维标定和平面标定的过程中均只使用 12 个棋盘内角点进行计算,并保证三维标定的 12 个点平均分布在三块棋盘平面上,角点匹配如图 3-4 所示。

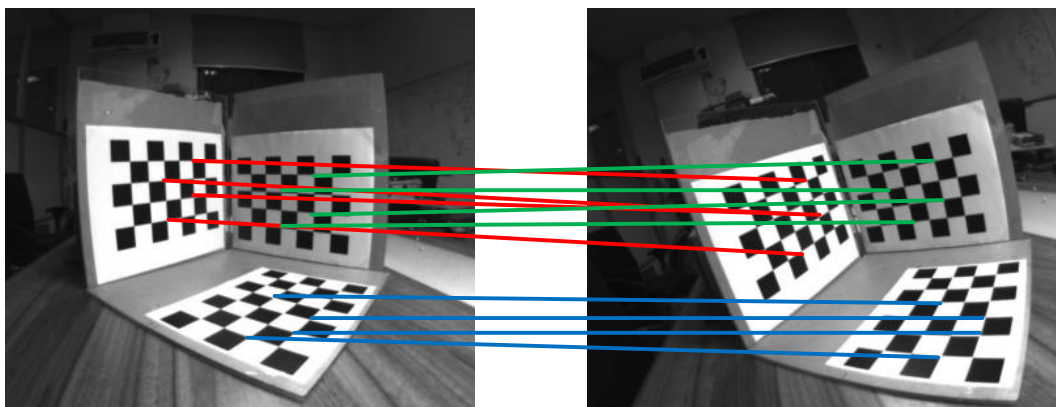


图 3-4 角点匹配

(2) 标定结果稳定性分析

为了验证本文提出的三维标定算法相比于平面标定算法具有更高的稳定性,分别用该两种方法对相机进行 20 组标定实验,得到标定参数的数值,并计算相关参数标定结果的标准差,绘制成图 3-5 所示的柱状图。图 3-5(a)对比了相机内参系数 f_x , f_y , c_x , c_y 实验数据的标准差大小,从中可以看出,使用三维标定方法重复标定相机内参系数时,其结果标准差明显小于平面标定方法,因此在标定内参系数时三维标定方法更加稳定。图 3-5(b)对比了畸变系数 k_1 , k_2 的标准差大小,从图中数据可以看出,由于在求取相机畸变时都是用相同的非线性优化进行迭代计算,因此两种方法标定结果的标准差在数值上很接近,都表现出较高的稳定性,相比而言,三维标定所得出的标准差结果更小。综上所述,无论是相机内参还是畸变系数,三维标定得出的结果稳定性更强。

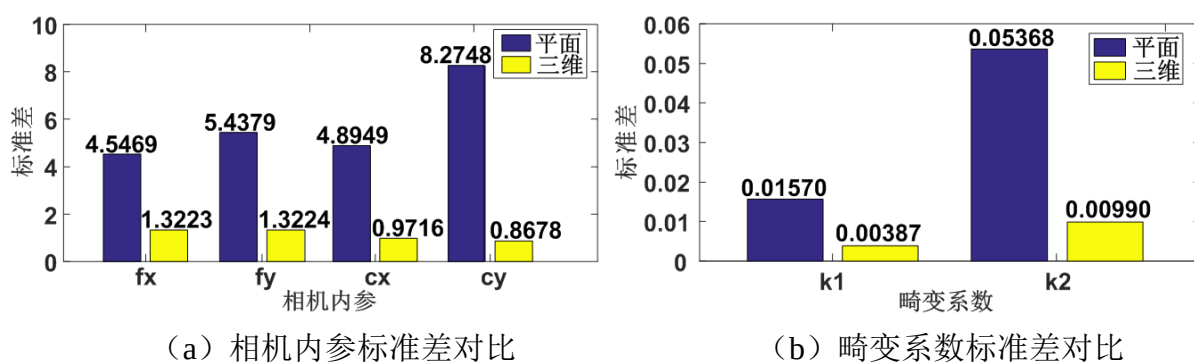


图 3-5 标准差对比

(3) 标定结果准确性分析

由于相机内、外参数的真值不容易获取,因此无法简单地根据标定参数的数值大小判定标定结果是否准确,需要额外设计合理的评价指标。本文使用“内符合重投影误差”和“外符合重投影误差”对标定参数进行评价,两者定义如下:

内符合重投影误差: 利用标定得到的相机模型参数,将标定过程中所使用到的 12

个点的世界坐标投影到像素坐标系中得到新的像素坐标 (u',v') ，计算新的像素坐标与图像处理提取的原始像素坐标 (u,v) 之间的平均误差。

外符合重投影误差：利用标定得到的相机模型参数，将标定过程中所有未使用到的点的世界坐标投影到像素坐标系中得到新的像素坐标 (u',v') ，计算新的像素坐标与图像处理提取的原始像素坐标 (u,v) 之间的平均误差。

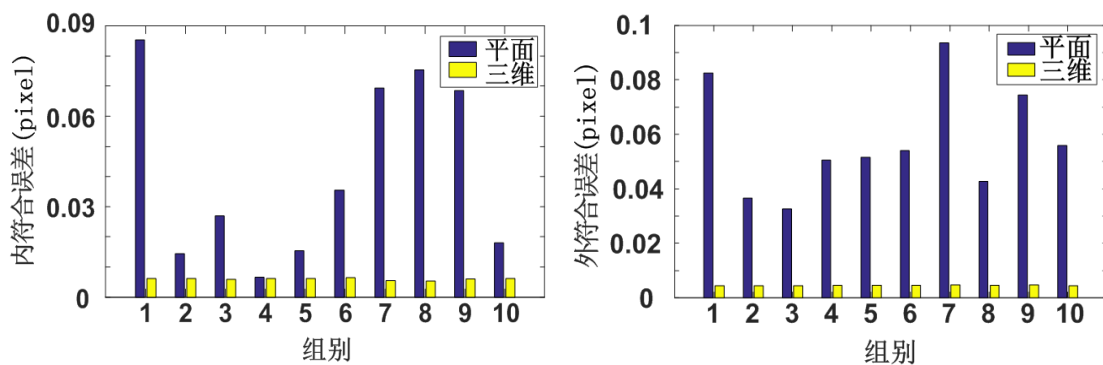
虽然在计算两种误差时，使用的像素点不同，但是均可以用式(3-23)进行表示，计算结果单位为像素 pixel。

$$e = \frac{\sum_{i=1}^n \sqrt{(u'_i - u_i)^2 + (v'_i - v_i)^2}}{n} \quad (3-23)$$

根据上述定义，计算 10 组标定结果的内、外符合重投影误差，将结果绘制成图 3-6 所示柱状图，并计算平均值和最大值，记录于表 3-1。从图中可以明显看出，对于任一组实验数据，无论是内符合重投影误差还是外符合重投影误差，三维标定的结果都要优于平面标定，且对于三维标定而言，平均外符合重投影误差比平均内符合重投影误差更小，这表明三维标定方法不仅能根据实验所用角点准确计算模型参数，同时能更好地拟合其余角点的投影规律。因此，相对于平面标定，三维标定获取的标定参数更符合针孔相机模型。

表 3-1 平面、三维标定的重投影误差对比

指标	平均值(pixel)	最大值(pixel)
平面内符合	0.04153	0.08531
三维内符合	0.00597	0.00647
平面外符合	0.05742	0.09351
三维外符合	0.00455	0.00475



(a) 内符合重投影误差对比

(b) 外符合重投影误差对比

图 3-6 内、外符合重投影误差对比

(4) 相机内参标定结果

根据上述实验流程，分别对双目相机的左、右目进行标定，最终的标定结果如下表所示：

表 3-2 双目相机内参标定结果

参数	左目相机	右目相机
f_x	367.610995	368.754216
f_y	365.595968	366.558067
c_x	369.583745	374.515657
c_y	234.734398	241.908339
k_1	-0.053863	-0.024091
k_2	0.023704	0.030881

3.5.2 IMU 标定实验

本课题中所使用的小觅相机 MYNTEYE-S1030 中, 自带 IMU 模块, 其型号为 BOSCH 的 BMI088。根据本文 3.3.1 节中介绍的 IMU 模型, 标定时主要分析该模块的高斯白噪声和随机游走零偏。首先将含有 IMU 模块的双目相机静置两小时, 利用 ROS 平台录制 IMU 在静止状态下的输出值, 包括加速度计和陀螺仪数据, 录制过程中 IMU 的采样频率为 200Hz。标定 IMU 的算法如 3.3.2 节所示, 对于加速度计而言, 依据其在静置状态下的测量值的二范数等于当地的重力加速度来构建优化方程, 并计算噪声参数。陀螺仪参数的确定借助于 Allan 方差曲线, 根据该曲线的性质, 陀螺仪的高斯白噪声出现在曲线斜率为 $-1/2$, $\tau = 1$ 处, 随机游走零偏则对应于曲线斜率为 0 的最小值处。图 3-7 所示为根据标定数据所绘制的 Allan 方差曲线, 表 3-3 是 IMU 标定的最终结果:

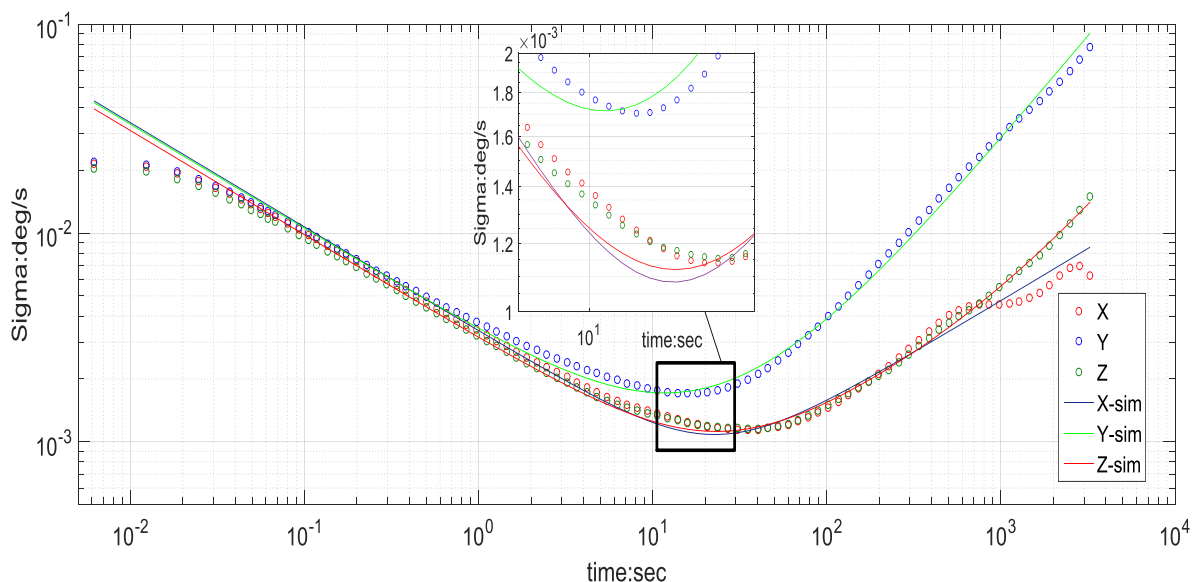


图 3-7 Allan 方差曲线

表 3-3 IMU 参数标定结果

传感器		白噪声	零偏
加速度计 (m/s^2)	X 轴	1.008461e-02	1.966391e-04
	Y 轴	1.184606e-02	2.383037e-04
	Z 轴	8.744638e-03	3.456378e-04
	轴平均	1.022511e-02	2.601935e-04
陀螺仪 ($^\circ/\text{s}$)	X 轴	4.336114e-02	1.083346e-03
	Y 轴	4.451973e-02	1.713925e-03
	Z 轴	4.031936e-02	1.120195e-03
	轴平均	4.273341e-02	1.305822e-03

3.5.3 IMU 与相机联合标定实验

本课题中,为了更好的融合 IMU 和双目相机进行联合导航定位,必须准确获取双目相机各镜头之间以及 IMU 与相机之间的相对位姿关系。本实验中使用 Kalibr 工具箱^[53]对以上参数进行标定,该工具箱由苏黎世工业大学开发,支持多相机标定、相机与 IMU 标定以及多 IMU 标定等功能。

(1) 双目相机标定实验与结果

在标定双目相机左右镜头的外参数时,使用如图 3-8 所示的 Aprilgrid 标定板进行标定。固定标定板,在保证左、右目视野中均包含标定板图像的前提下,任意旋转相机,通过 ROS 系统录制相机的左目图像和右目图像数据,Kalibr 工具箱会利用记录的图片数据自动计算双目之间的位置姿态变换矩阵,并分别计算左、右目相机的重投影误差结果。表 3-4 是双目相机外参标定结果及左、右目相机重投影误差值。

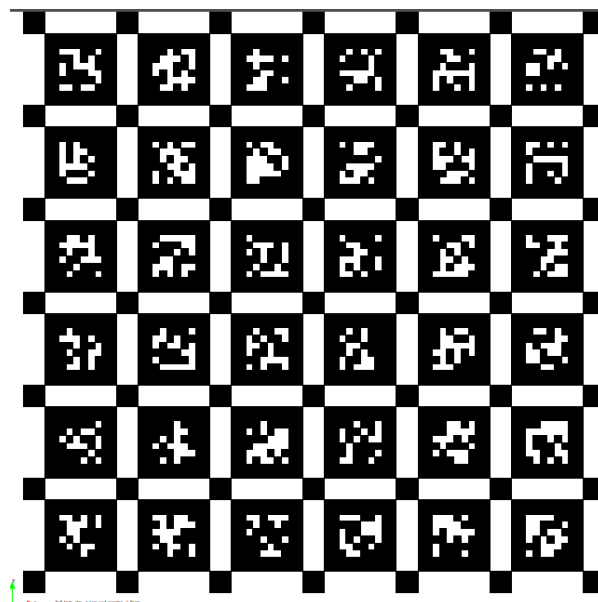


图 3-8 Aprilgrid 标定板

表 3-4 双目相机外参标定结果

参数	标定结果
旋转矩阵	$\begin{bmatrix} 0.999919 & -0.002222 & -0.012484 \\ 0.002180 & 0.999992 & -0.003430 \\ 0.012492 & 0.003403 & 0.999916 \end{bmatrix}$
平移向量	$[-0.119956 \quad -0.000081 \quad -0.001084]$
左目重投影误差	$[0.000011 \quad -0.000004] \pm [0.076718 \quad 0.092650]$
右目重投影误差	$[-0.000014 \quad 0.000007] \pm [0.056718 \quad 0.088612]$

标定结果中, 旋转矩阵表示的是右目相机旋转到左目相机的变换结果, 因为本实验标定的小觅相机的左、右目为平行向前放置, 因此标定出的旋转矩阵也近似于单位矩阵。根据标定的平移向量可知, 左右目相机之间的平移距离绝对值约为 119.96mm, 与官方给定的双目基线长度 120mm 基本相符。

(2) 相机与 IMU 联合标定结果

标定双目相机与 IMU 的参数时, 同样借助于 Aprilgrid 标定板和 Kalibr 工具箱, 在标定过程中, 除了要确保相机能看到图案外, 同时也要对 IMU 的三个坐标轴进行充分的运动激励, 包括旋转和平移, 获取 IMU 在六个自由度上的运动数据。标定数据同样借助于 ROS 系统获取, 相机和 IMU 的采样频率分别为 25Hz 和 200Hz, 最终的标定结果如表 3-5 所示:

表 3-5 IMU 与相机外参标定结果

参数	标定结果
IMU 与左目	旋转矩阵 $\begin{bmatrix} -0.00316251 & 0.99998686 & -0.00403469 \\ -0.99994693 & -0.00312277 & 0.00981798 \\ 0.00980525 & 0.00406552 & 0.99994366 \end{bmatrix}$
	平移向量 $[0.00615005 \quad -0.00197907 \quad -0.0021587]$
	时间漂移 $-0.0721110122547s$
IMU 与右目	旋转矩阵 $\begin{bmatrix} -0.00106189 & 0.99986264 & -0.01653978 \\ -0.9999792 & -0.00095649 & 0.00637919 \\ 0.0063625 & 0.01654621 & 0.99984286 \end{bmatrix}$
	平移向量 $[-0.11377586 \quad -0.00203989 \quad -0.00317262]$
	时间漂移 $-0.0835961320016s$

3.6 本章小结

本章首先研究了单目相机的内参数标定方法, 针对平面标定存在的不足提出了改进的三维标定方法, 并对三维标定算法进行了推导。明确了双目相机左、右目之间需要标定的外参数, 并确定了标定理论。分析了 IMU 的测量原理, 确定其噪声组成项, 并以此建立了 IMU 测量模型, 提出了标定方法, 同时基于手眼标定原理, 提出了 IMU 与双目相机之间的联合姿态标定算法。最后, 通过标定实验, 给出了视觉传感器与 IMU 的内、外参标定结果, 并通过数据对比, 体现本文改进的三维标定方法的稳定性与准确性。

第四章 视觉与 IMU 组合导航定位

现阶段无人机常用的主动导航定位算法主要分为纯视觉的 VO 算法和融合了视觉与 IMU 的 VIO 算法。纯视觉算法相对简单但鲁棒性不高，在一些单一场景或者是运动速度过快的情况下容易失准。此外纯视觉算法在进行旋转运动时尺度估计不准确，容易影响无人机位姿状态的估计。而 IMU 在导航定位过程中对外界环境因素不敏感，应对旋转运动也有较强的鲁棒性，因此本章研究融合了视觉导航与惯性导航的 VIO 算法，以提升无人机定位的鲁棒性与准确性。

4.1 视觉惯导里程计算法总体框架

本文提出的 VIO 算法主要分为前端和后端两部分，总体的算法框图如图 4-1 所示。前端利用视觉传感器采集图像，从图像中提取特征角点并采用光流法进行跟踪，从而利用视觉特征计算无人机的运动位姿。同时，前端使用 IMU 的数据信息初始化 VIO 系统，并辅助视觉算法实现动态特征的识别与滤除功能。后端以 EKF 框架为主体，使用视觉与 IMU 共同估计的位姿结果作为相机的扩增状态，并代替特征点组成滤波的状态向量，以降低状态向量的整体维度。使用滑动窗口维系一定数量的位姿状态，并在观测到相同特征点的状态之间建立多状态约束，当相机帧数过多或特征点跟踪丢失时，根据建立的几何多状态约束对系统的状态进行更新。

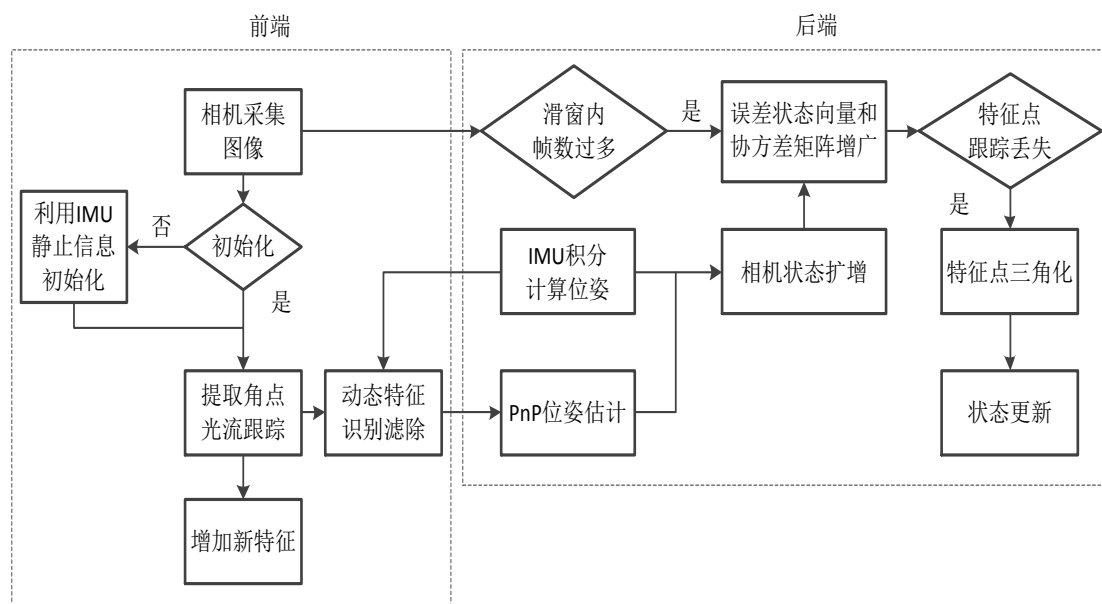


图 4-1 VIO 算法总体框架

4.2 视觉里程计前端

VIO 算法的一部分即由视觉里程计（VO）算法构成，常用的 VO 算法根据是否提取描述性特征分为特征点法和直接法。特征点法在使用过程中需要对图像计算描述子，例如 ORB^[54]描述子、SURF^[55]描述子等等。描述子在描述图像特征时，会考虑特征的朝

向、旋转等属性,因此在描述特征时更具有区分性。但是由于描述子包含的信息更丰富,因此也会较大地占用系统的计算资源,影响系统的实时性。因此,本文在视觉里程计前端采用直接法,对图像提取 FAST 角点后,利用 LK 光流法^[56]进行跟踪。

4.2.1 光流法

LK 光流是稀疏光流的一种,在计算过程中只对图像中的部分像素进行跟踪,从而能更好地保证系统的实时性。在 LK 光流中,将图像视作是关于时间和位置的函数 $I(\bullet)$,则在 t 时刻,图像上像素坐标 (x, y) 处点的灰度值可以表示为 $I(x, y, t)$ 。对于某一像素点,经过 dt 时刻后其运动到了 $(x+dx, y+dy)$ 处,假设其灰度值未发生变化,则有:

$$I(x+dx, y+dy, t+dt) = I(x, y, t) \quad (4-1)$$

对等式左边进行一阶泰勒展开,消去相等的灰度值后可以得到:

$$\frac{\partial I}{\partial x} \frac{dx}{dt} + \frac{\partial I}{\partial y} \frac{dy}{dt} = -\frac{\partial I}{\partial t} \quad (4-2)$$

将该式进一步改写为矩阵形式:

$$\begin{bmatrix} I_x & I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -I_t \quad (4-3)$$

其中 u, v 分别为该像素值在 x 轴和 y 轴上的运动速度, I_x, I_y 为灰度值在 x 轴和 y 轴上的变化梯度, I_t 为灰度值对时间的变化量。一个像素点可以得到式(4-3)的一条方程,但是对于求解出 u, v 两个变量还不够。假设在某一个有 w ($w \geq 2$) 个像素点的窗口中,所有的像素点都具有相同的运动,即可构建如下方程组进行求解:

$$\begin{bmatrix} I_x & I_y \end{bmatrix}_k \begin{bmatrix} u \\ v \end{bmatrix} = -I_{tk}, k=1, \dots, w \quad (4-4)$$

记

$$A = \begin{bmatrix} \begin{bmatrix} I_x & I_y \end{bmatrix}_1 \\ \vdots \\ \begin{bmatrix} I_x & I_y \end{bmatrix}_k \end{bmatrix}, b = \begin{bmatrix} I_{t1} \\ \vdots \\ I_{tk} \end{bmatrix} \quad (4-5)$$

则有:

$$\begin{bmatrix} u \\ v \end{bmatrix} = -(A^T A)^{-1} A^T b \quad (4-6)$$

对于此方程,可以采用最小二乘法求解 u, v ,经过多次迭代,可以获取较为准确的解。在获取了像素点的运动速度之后,可以计算出 dt 时间后该点所处的坐标位置。

4.2.2 特征跟踪

为了提升系统的定位鲁棒性,本文采用双目相机融合 IMU 进行定位。因此在特征跟踪过程中,除了需要跟踪左目前后两帧图像中的特征,同时也要匹配同一时刻,左目与右目之间的特征。

在运用 LK 光流对左目前后两帧图像进行跟踪的过程中,为了避免特征点集中在图像上的某一小块,需对图像特征点均匀化提取。如图 4-2 所示,将图像分成了 4×5 个网

格，每个网格中的特征点数限制在 10 个以下，当跟踪的特征点数量超过限制时，对网格中的部分特征点进行剔除，保留被跟踪的帧数较多的特征点。

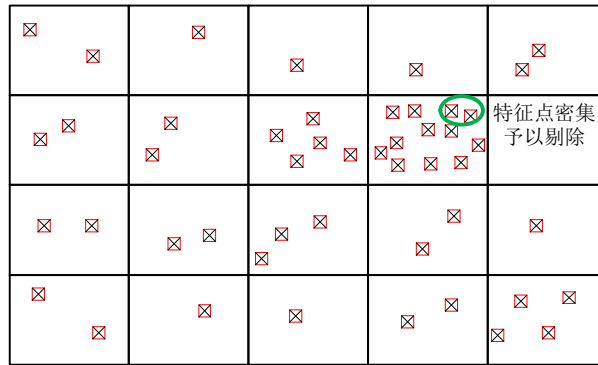


图 4-2 特征点均匀化示意图

在左、右目的特征跟踪过程中，对当前帧左目图像中提取出的特征点，在当前帧右目图像中进行光流跟踪，从而获得匹配特征。同时，为了获取双目之间较为准确的匹配，使用对极几何原理对跟踪后的特征点进行极线约束判断。对极几何^[57]原理如图 4-3 所示， O_1 和 O_2 称为左、右两相机的主点， P 为空间中一个三维点。 p_1 和 p_2 是 P 点在左右相机像平面上的投影点， e_1 、 e_2 为像平面和基线 O_1O_2 的交点，称作极点。由 O_1PO_2 组成的平面称为极平面， p_1e_1 和 p_2e_2 分别称为左极线和右极线。

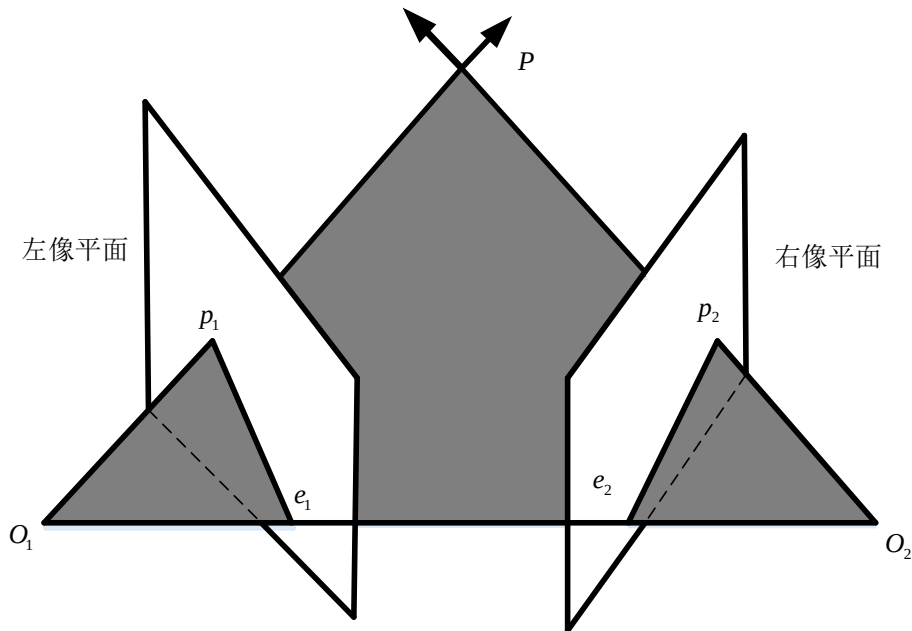


图 4-3 对极几何示意图

以左相机为例，射线 O_1p_1 上的任何一点在右像平面上的成像都会落在右极线 p_2e_2 上，从而构成极线约束。对于左、右目中投影到归一化平面上的点 p_1 、 p_2 而言，若其符合极线约束，则有

$$p_2^T E p_1 = 0 \quad (4-7)$$

其中 E 被称为本质矩阵，其定义如下：

$$E = t^{\wedge} R \quad (4-8)$$

其中左、右目之间的旋转矩阵 R 和平移向量 t 在标定双目相机外参时获得。在使用

光流对左、右目进行特征匹配时,对于不满足式(4-7)的匹配点对,可以认为是误跟踪的点,予以剔除。同时为了保证特征点的数量以及均匀性,在当前帧的左目图像上进行新特征点的提取,并将新的 FAST 角点投射到网格中,若网格中特征点数超出了上限则删除该新角点。

4.3 动态特征识别与滤除

4.3.1 IMU 运动补偿

在运用视觉算法进行定位时,通常都假设环境中的物体处于静止状态,在这种假设前提下,如果环境中出现较多动态物体,并且这些动态物体的特征点也被前端视觉跟踪提取,则将会降低前端视觉算法的精度,进一步导致后端优化过程中残差过大,影响算法整体的组合定位效果。因此有必要在前端提取特征时,准确地识别动态特征点,并将其剔除。目前运用比较高效的动态特征剔除方法是结合语义信息,对动态物体进行整体识别。但是语义建模过程计算复杂度较高,在一些算力局限的嵌入式定位平台上无法保证较好的实时性。

相对于视觉而言,IMU 在获取测量数据的过程中对外界环境不敏感,即使处于复杂的动态环境之中,IMU 的运动估计也不会受到影响。根据 IMU 的此种特性,本文提出一种基于运动分割且计算相对简单的动态特征识别方法,借助于 IMU 测得的旋转信息,对图像场景中的动态特征进行识别和滤除,从而实现在保证系统实时运行的前提下,降低动态物体对视觉定位的影响。

假设在运动过程中,无人机未发生横滚运动。相机在拍摄第 $k-1$ 帧和第 k 帧图像的过程中,IMU 测得了系统偏航角和俯仰角的变化,前者用于补偿系统的有意旋转,后者用于补偿相机的振动。由于相机的拍摄频率较高,图像帧之间的平移运动被认为足够小,可以先忽略相机在拍摄两帧图像之间的平移运动,则可以根据如下公式计算第 $k-1$ 帧中的特征点在第 k 帧中的坐标:

$$\begin{cases} p(x) = \frac{f}{dp} \tan \left(\Delta Yaw + \arctan \left(\left(x - \frac{I_w}{2} \right) * \frac{dp}{f} \right) \right) \\ p(y) = \frac{f}{dp} \tan \left(\Delta Pitch + \arctan \left(\left(y - \frac{I_h}{2} \right) * \frac{dp}{f} \right) \right) \end{cases} \quad (4-9)$$

其中 ΔYaw 和 $\Delta Pitch$ 是由 IMU 测量的连续图像帧之间的偏航角和俯仰角, (x, y) 是某个特征点在图像 $k-1$ 帧的像素坐标, $p(x), p(y)$ 是投影到第 k 帧图像中的像素坐标, f 是相机的焦距, dp 是像素宽度, I_w 图像宽度, I_h 是图像高度。以偏航角为例,投影的示意图如图 4-4 所示。

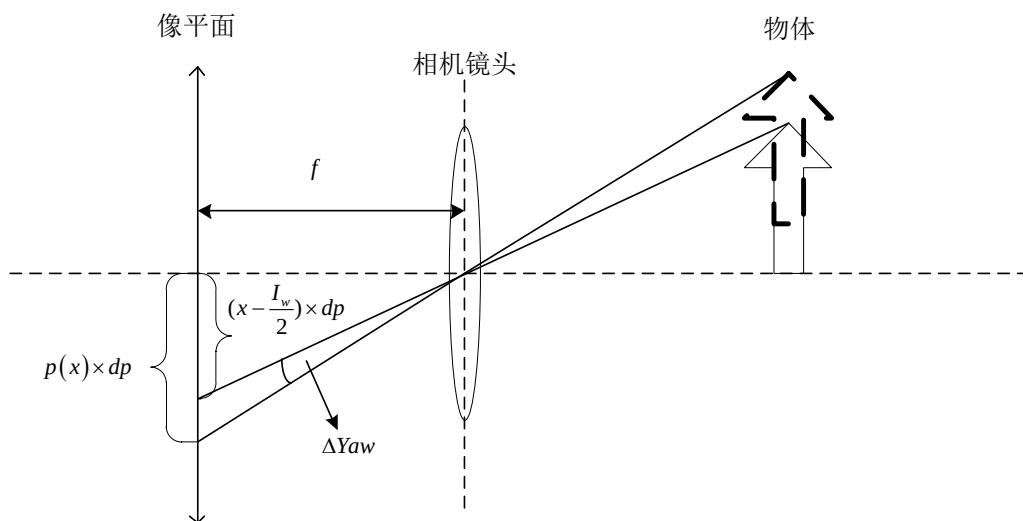


图 4-4 偏航补偿示意图

图 4-4 为了便于表示，将相机视为静止，物体相对相机发生旋转。由于运动足够小，忽略物体相对于相机的平移分量。

4.3.2 基于改进 K-means 算法的动态特征识别

对于环境中的动态物体，其本身与相机之间会产生相对运动，因此对于动态物体上检测到的特征点而言，其在第 k 帧图像中的坐标与式(4-9)计算出的结果具有一定的偏差。设图像 $k-1$ 帧中的某个点在 k 帧中实际的像素坐标为 (x, y) ，用欧式距离定义像素差：

$$\Delta d = \sqrt{(p(x) - x)^2 + (p(y) - y)^2} \quad (4-10)$$

由于传感器测量过程中受噪声影响，图像中静态特征的像素差也不会完全等于 0，因此要设定合适标准对图像中的特征点进行区分。在区分之前，借助正态分布判定当前图像中是否存在动态特征，对于两帧图像中检测到的所有特征点计算像素差，并按照如下公式计算求取的所有像素差的标准差 σ ：

$$\sigma = \frac{1}{N} \sqrt{\sum_{i=1}^N (\Delta d_i - \mu)^2} \quad (4-11)$$

其中 μ 为所有像素差的均值。根据正态分布，当某个特征点的像素差与 μ 相距超过阈值 3σ ，该点即可被认定为异常点。根据实验经验可知，当异常点对应的像素差均值大于 3μ 时，即可认为该图像环境中存在动态特征，需要对其进行准确识别与滤除。

在确定环境中存在动态特征后，本文使用改进的 K-means 聚类算法对图像中提取的特征点进行归类，将其划分为静态与动态两大聚簇。K-means 算法是一种比较常用的划分聚类方法，算法流程如图 4-5 所示。该方法在给定 K 个初始类簇中心点的情况下，把数据集里的每个数据记录分到离其最近的类簇中心点所代表的类簇中。当所有点被分配完毕之后，在每一个类簇内重新计算类簇中心点，并对数据记录进行重新分配。如此循环迭代，直至类簇中心点变化很小为止。K-means 算法运算速度快，易于实现，但是该方法在选取初始簇类中心时，采用随机的方式进行选取，容易导致算法迭代次数过多甚至得到不符合预期的聚类结果。

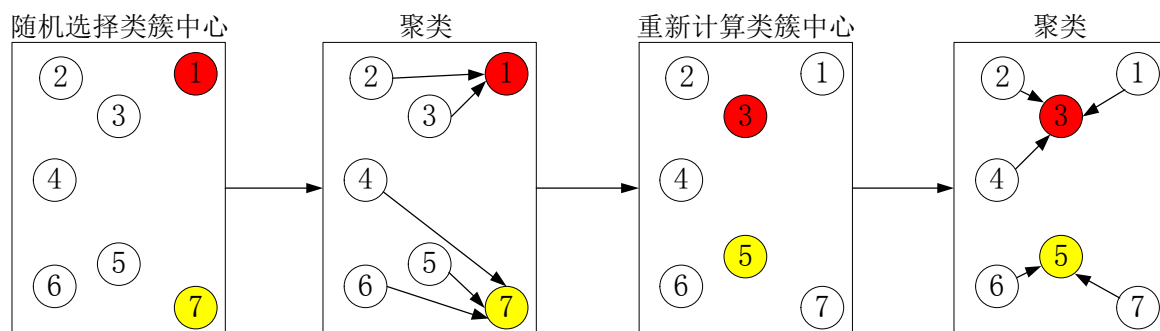


图 4-5 K-means 算法流程

结合本文的运用场景，使用改进的 K-means 算法对计算的像素差进行聚类。本文的应用场景限定于室内，该环境下动态物体主要以行人为主，运动速度基本一致，因此可以简单地将图像中的特征点分类为两大聚簇，即将 K 值选为 2。同时为了减少迭代次数，选取初始簇类中心时，不采用随机选取方式，而是将最小和最大的两个像素差分别作为静态簇和动态簇的初始中心，以此进行迭代计算。为了提升识别的准确性，分别对同一时刻左、右目图像中的点进行聚类，只有在双目图像中同时被划分为动态簇的特征点才被认定为环境中的动态特征，并予以剔除。

整体的动态滤除算法框架如图 4-6 所示，通过该识别方式可以在一定程度上辨别环境中的动态特征，减少动态物体对 VIO 定位精度的影响。

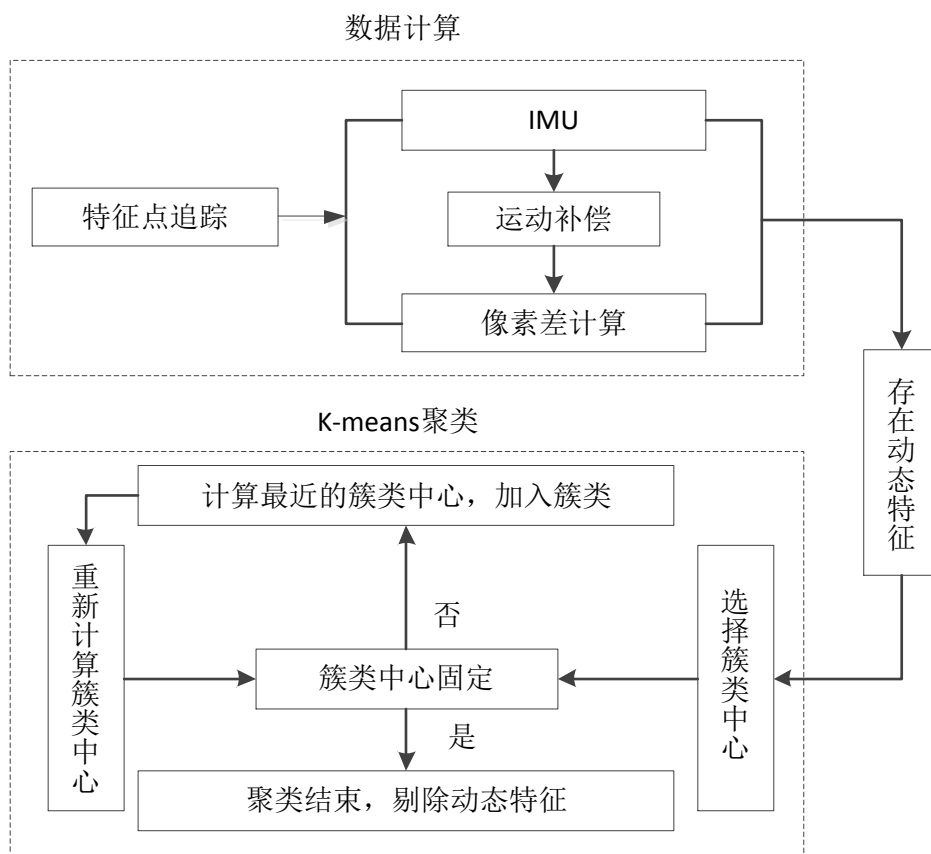


图 4-6 动态识别流程

4.4 基于滤波的多状态约束紧耦合 VIO

现有的紧耦合 VIO 解决方案大致可以分为两种，基于优化的方法和基于滤波器的方

法。基于优化的方法通过使用来自 IMU 和图像的测量值来联合最小化残差，从而获得最佳估计值。在同等条件下，基于优化的定位算法精度一般要高于滤波算法。然而，由于后端在优化位姿时需要经历迭代优化过程，所以优化方法需要消耗大量的计算资源，这对于定位系统的实时性是一大考验。相较而言，基于滤波器的方法能在确保较高精度的情况下，更好地节约平台算力，提升系统的实时性。因此本文使用一种基于扩展卡尔曼滤波的方法，对视觉和 IMU 的定位信息进行紧耦合优化。

4.4.1 EKF 状态向量

视觉与 IMU 紧耦合的系统中，需要组合 IMU 和视觉的测量信息作为系统的状态向量。对于 IMU 而言，其不断变化的状态由式(4-12)所示的向量描述：

$$X_{IMU} = \begin{bmatrix} {}^I\bar{q}^T & b_g^T & {}^Gv_l^T & b_a^T & {}^Gp_l^T \end{bmatrix}^T \quad (4-12)$$

式中 ${}^I\bar{q}$ 是描述旋转的四元数，表示从世界坐标系（即符号 G，下称 G 系）到 IMU 本体坐标系（即符号 I，下称 I 系）的旋转， Gp_l 和 Gv_l 表示 IMU 在 G 系中的位置和速度。这里使用 3.3.1 节中的模型对 IMU 的测量偏差进行建模， b_g 和 b_a 为两个 3×1 向量，描述了陀螺仪和加速度计测量的偏差，两者均服从高斯分布。根据式(4-12)，可以进一步计算 IMU 的状态误差向量。对于位置、速度和偏差而言，可以直接将真值与估计值进行做差。但对于四元数，误差定义有所不同。假设四元数 q 的估计值为 $\hat{q}$ ，两者之间的误差为 δq ，则根据四元数的乘法定义，三者存在如下数学关系：

$$q = \delta q \otimes \hat{q} \quad (4-13)$$

其中符号 $\otimes$ 表示四元数乘法。进一步地，将四元数表示的误差用旋转角 $\delta\theta$ 来描述：

$$\delta q \cong \begin{bmatrix} \frac{1}{2}\delta\theta^T & 1 \end{bmatrix}^T \quad (4-14)$$

综合上述推导，系统的状态误差向量可以由式(4-15)进行描述：

$$\tilde{X}_{IMU} = \begin{bmatrix} \delta\theta_l^T & \tilde{b}_g^T & {}^G\tilde{v}_l^T & \tilde{b}_a^T & {}^G\tilde{p}_l^T \end{bmatrix}^T \quad (4-15)$$

对于后端紧耦合算法而言，需要将相机的测量数据和 IMU 的状态融合在一起估计。传统的 EKF 紧耦合方法是将视觉提取到的所有特征点加入到状态向量中与 IMU 状态一起估计，这种做法简单直观，但是当环境视野很大时，视觉前端提取的特征点可能会非常多，从而导致状态向量维数变得非常大，不便于维护。为了避免在后端估计时维护维数过大的状态向量，本文在后端紧耦合时，不把特征点加入到状态向量中，而是将不同时刻的相机位姿加入到状态向量中进行估计。在相机的运动过程中，环境中的同一特征点会在多个时刻被前端视觉算法所提取，进而能够在多个相机状态之间构建几何约束，可以利用这种多状态约束构建观测模型对滤波器状态进行更新。相比于特征点的数量而言，相机位姿的个数明显要少，因此此种方法可以有效降低状态向量的维度。添加 N 个相机位姿之后，系统整体的状态向量如式(4-16)所示：

$$\hat{X}_k = [X_{IMU_k}^T \quad {}^G\hat{q}^T \quad {}^G\hat{p}_{C_1}^T \quad \cdots \quad {}^G\hat{q}^T \quad {}^G\hat{p}_{C_N}^T]^T \quad (4-16)$$

其中 ${}^G\hat{q}$ 和 ${}^G\hat{p}_{C_i}$ ， $i = 1 \dots N$ 表示在 G 系中第 i 个相机姿态和位置的估计值。仿照 IMU 状态误差向量的推导方式，EKF 系统整体的误差状态向量如下所示：

$$\tilde{X}_k = [\tilde{X}_{IMU_k}^T \quad \delta\theta_{C_1}^T \quad {}^G\hat{p}_{C_1}^T \quad \cdots \quad \delta\theta_{C_N}^T \quad {}^G\hat{p}_{C_N}^T]^T \quad (4-17)$$

4.4.2 运动方程

EKF 后端紧耦合的运动方程以 IMU 的传播方程为基础, IMU 状态的传播方程是通过将连续时间下的 IMU 系统模型离散化所得出的, 对于连续时间模型, IMU 的各状态量之间真实的物理关系如式(4-18)所示:

$$\begin{cases} {}^I_G\dot{\hat{q}}(t) = \frac{1}{2}\Omega(\omega(t)) {}^I_G\hat{q}(t), & \dot{b}_g(t) = n_{wg}(t) \\ {}^G\dot{v}_I(t) = {}^Ga(t), & \dot{b}_a(t) = n_{wa}(t), \quad {}^G\dot{p}_I(t) = {}^Gv_I(t) \end{cases} \quad (4-18)$$

其中, Ga 是在 G 系中的加速度, $\omega = [\omega_x \quad \omega_y \quad \omega_z]^T$ 是在 I 系中表示的旋转角速度, 且有

$$\Omega(\omega) = \begin{bmatrix} -[\omega \times] & \omega \\ -\omega^T & 0 \end{bmatrix}, \quad [\omega \times] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (4-19)$$

同时, IMU 的测量值可以由式(4-20)定义:

$$\begin{cases} \omega_m = \omega + b_g + n_g \\ a_m = C({}^I_G\hat{q})({}^Ga_I - {}^Gg) + b_a + n_a \end{cases} \quad (4-20)$$

其中, 函数 $C(\bullet)$ 根据四元数计算其所对应的旋转矩阵, n_g 和 n_a 为零均值高斯白噪声, 表示 IMU 的测量误差, Gg 是所处环境的重力加速度值。

根据式(4-18)和(4-20), 可以直接推导出状态估计量的模型:

$$\begin{cases} {}^I_G\dot{\hat{q}} = \frac{1}{2}\Omega(\hat{\omega}) {}^I_G\hat{q}, & \dot{b}_g = 0_{3 \times 1}, \\ {}^G\dot{v}_I = C({}^I_G\hat{q})^T \hat{a} + {}^Gg, & \dot{b}_a = 0_{3 \times 1}, \quad {}^G\dot{p}_I = {}^G\hat{v}_I \end{cases} \quad (4-21)$$

根据状态估计量的系统模型可以推导 IMU 误差状态的运动模型, 为了简化表示, 将其写成矩阵相乘的线性方程:

$$\dot{\tilde{X}}_{IMU} = F\tilde{X}_{IMU} + G n_{IMU} \quad (4-22)$$

其中:

$$F = \begin{bmatrix} -[\hat{\omega} \times] & -I_3 & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ -C({}^I_G\hat{q})^T [\hat{a}_x] & 0_{3 \times 3} & 0_{3 \times 3} & -C({}^I_G\hat{q})^T & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & I_3 & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \quad (4-23)$$

$$G = \begin{bmatrix} -I_3 & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & I_3 & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & -C({}^I_G\hat{q})^T & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & I_3 \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix} \quad (4-24)$$

I_3 表示 3×3 的单位矩阵。

在实际情况下, IMU 以一定的采样频率获取角速度和加速度的测量值, 时间是离散化的, 在这种情况下需使用离散测量值对 EKF 中的状态进行更新。每当后端滤波接收到新的 IMU 测量值时, 会对等式(4-21)计算 4 阶 Runge-Kutta^[58]数值积分, 从而对 IMU 的状态进行估计。与此同时, 也要预测 EKF 的协方差矩阵。为了简化计算, 首先对协方差矩阵进行划分:

$$P_{k|k} = \begin{bmatrix} P_{II_{k|k}} & P_{IC_{k|k}} \\ P_{IC_{k|k}}^T & P_{CC_{k|k}} \end{bmatrix} \quad (4-25)$$

其中, $P_{II_{k|k}}$ 是计算 IMU 状态的 15×15 协方差矩阵, $P_{CC_{k|k}}$ 是相机姿态估计的协方差矩阵, 维度为 $6N \times 6N$, N 为维持的相机姿态个数。 $P_{IC_{k|k}}$ 表示的是 IMU 状态误差和相机位姿误差之间的相关性估计。借助于上述划分, 协方差矩阵的更新过程由式(4-26)给出:

$$P_{k+1|k} = \begin{bmatrix} P_{II_{k+1|k}} & \Phi(t_k + T, t_k) P_{IC_{k|k}} \\ P_{IC_{k+1|k}}^T & P_{CC_{k|k}} \end{bmatrix} \quad (4-26)$$

其中, T 为 IMU 采集数据的时间间隔, $\Phi(\bullet)$ 表示某段时间间隔的状态转移矩阵。 $P_{II_{k+1|k}}$ 通过如下数值积分方程进行计算:

$$\dot{P}_{II} = FP_{II} + P_{II}F^T + GQ_{IMU}G^T \quad (4-27)$$

该方程在时间间隔 $(t_k, t_k + T)$ 上, 以 $P_{II_{k|k}}$ 作为初始值进行数值积分。类似地, 状态转移矩阵 $\Phi(t_k + T, t_k)$ 通过微分方程的数值积分来进行计算:

$$\dot{\Phi}(t_k + \tau, t_k) = F\Phi(t_k + \tau, t_k), \quad \tau \in [0, T] \quad (4-28)$$

4.4.3 改进的相机状态扩增方式

在 4.4.1 节中可以看到, 状态向量中包含了 IMU 状态和 N 个相机状态。对于每一帧新的相机状态, 都需要加入到状态向量中。在扩增相机位姿时, 经典的做法是在获取一帧新的图像后, 先利用 IMU 运动更新方程预测 IMU 的运动状态, 再根据 IMU 和相机之间的相对位置外参推算出当前帧的相机位姿, 从而将最新的相机状态加入到状态向量中并扩展协方差矩阵。

这种方法简单高效, 但是在扩增相机状态时只依赖于 IMU 的运动信息, 并没有很好地融合视觉定位方法, 这也容易导致系统在长时间估计运动位姿时受到 IMU 累积误差的影响, 从而降低算法的定位精度。因此本文对后端滤波中扩增相机状态的方法进行改进, 当视觉前端跟踪到较多的特征点时, 利用纯视觉信息计算无人机的位姿变化, 并将此结果与 IMU 的测量信息进行融合, 以此对相机状态进行扩增, 从而降低 IMU 误差对系统定位精度的影响。

视觉前端在连续两帧图像中, 可以匹配到相同的特征点, 利用这些匹配的特征信息可以使用 PnP^[59, 60]算法估计相机在两帧之间的相对运动。对于空间中的三维点 $P(X_i, Y_i, Z_i)$, 其与对应的像素点 $p(u_i, v_i)$ 之间符合 2.2.2 节中的针孔相机投影模型, 这里用李代数的方式重新进行表示:

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \exp(\xi) \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (4-29)$$

其中 s 为尺度因子， K 为相机的内参矩阵， ξ 为包含旋转与平移信息的李代数，函数 $\exp(\bullet)$ 根据李代数计算其对应的变换矩阵。当图像中匹配的点数较多时，可以借助图 4-7 所示的重投影误差来构建最小二乘方程，从而优化相机位姿，优化函数如下：

$$\xi^* = \arg \min_{\xi} \frac{1}{2} \sum_{i=1}^n \left\| p_i - \frac{1}{s_i} K \exp(\xi) P_i \right\|_2^2 \quad (4-30)$$

其中 n 为匹配点的数量。使用 L-M 方法对该优化函数进行求解，即可获取视觉前端计算出的相对位姿，用 ${}^C_1 \hat{q}$ 和 ${}^C_1 \hat{p}_C$ 表示。

将式(4-30)计算估计的位姿与 IMU 的测量信息求平均值，从而实现融合补偿，对于平移分量而言，直接对二者取算数平均，计算公式如下：

$$\begin{cases} {}^G \hat{p}_C^1 = {}^G \hat{p}_I + C({}^I_1 \hat{q})^T {}^I p_C \\ {}^G \hat{p}_C^2 = {}^G \hat{p}_{C_1} + C({}^{C_1}_G \hat{q})^T {}^{C_1} \hat{p}_{C_2} \\ {}^G \hat{p}_C = ({}^G \hat{p}_C^1 + {}^G \hat{p}_C^2) / 2 \end{cases} \quad (4-31)$$

其中， ${}^G \hat{p}_C^1$ 、 ${}^G \hat{p}_C^2$ 为 IMU 和视觉计算的平移分量， ${}^I p_C$ 是相机在 I 系中的初始位姿。

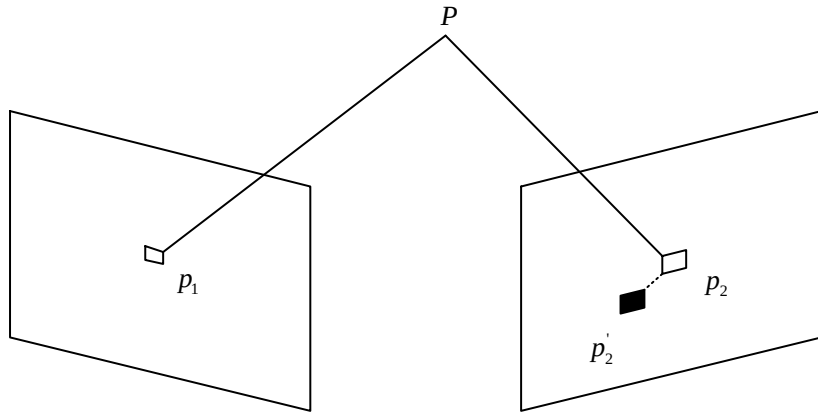


图 4-7 重投影误差示意图

对于旋转分量而言，需要对四元数求平均计算，旋转四元数的平均值计算公式如下所示：

$$Avg(q_1, q_2) = \frac{[(w_1 - w_2 + z)q_1 + 2w_2(q_1^T q_2)q_2]}{\| (w_1 - w_2 + z)q_1 + 2w_2(q_1^T q_2)q_2 \|_F} \quad (4-32)$$

其中 $\|\bullet\|_F$ 表示 F-范数， w_1 ， w_2 代表四元数的权重，本文在求四元数平均时将两者的权重均设置为 0.5， z 的计算方式如下所示

$$z = \sqrt{(w_1 - w_2)^2 + 4w_1w_2(q_1^T q_2)^2} \quad (4-33)$$

利用 IMU 和视觉计算完两帧之间的旋转后，使用如下公式扩展相机的旋转状态分

量

$$\begin{cases} {}^c\hat{q}_I = {}^c_Iq \otimes {}^I\hat{q} \\ {}^c\hat{q}_C = {}^c_{C_1}\hat{q} \otimes {}^{C_1}_Cq \\ {}^c\hat{q} = \text{Avg}({}^c\hat{q}_I, {}^c\hat{q}_C) \end{cases} \quad (4-34)$$

其中 ${}^c\hat{q}_I$ 和 ${}^c\hat{q}_C$ 分别表示 IMU 和视觉计算出的旋转四元数。

视觉前端只有提取到足够的特征点时, 计算出的位姿估计才相对准确。因此当视觉前端滤除过多的动态特征点, 或者跟踪特征点数量不足的情况下, 不进行视觉补偿, 只使用 IMU 的计算结果对相机状态进行扩增, 此时的扩增方程如下所示:

$$\begin{cases} {}^c\hat{q}_I = {}^c_Iq \otimes {}^I\hat{q} \\ {}^c\hat{p}_C = {}^C\hat{p}_I + C({}^I\hat{q})^T {}^I p_C \end{cases} \quad (4-35)$$

相机状态扩增过程中, 除了使用了视觉本身的观测信息外, 也融合了 IMU 的运动状态, 从而将相机状态与 IMU 状态紧密联系了起来。通过关联协方差 p_{IC} 描述相机和 IMU 状态之间的关系, 这样在更新相机状态的同时, 也会间接更新 IMU 状态。

4.4.4 基于多状态约束的观测模型

为了更新状态估计, 需要建立一个观测模型对状态估计值进行更新。根据前面的内容, 本文使用相机位姿代替环境中的特征点作为状态估计量。在运动过程中, 相邻的相机帧之间能够观测到环境中相同的特征点, 利用这些观测信息可以在观测到相同特征点的相机位姿之间构建约束关系。因此本文在定义残差观测模型时, 先根据相机观测值, 三角化特征点的空间坐标, 然后定义特征点到各相机的重投影残差作为观测量, 并推导残差与状态向量之间的线性化关系。最后, 直接套用 EKF 公式对状态向量和协方差进行更新。

三角化特征点的过程中, 对于某一个具体的特征点 $f_j(X_j, Y_j, Z_j)$ 而言, 其总共被 S_j 个相机位姿观测到, 其中每一次观测都可以用如下公式进行表示:

$$z_i^{(j)} = \frac{1}{c_i Z_j} \begin{bmatrix} c_i X_j \\ c_i Y_j \end{bmatrix} + n_i^{(j)}, \quad i \in S_j \quad (4-36)$$

其中 $n_i^{(j)}$ 为图像噪声。为了提高数值计算的稳定性, 特征点坐标采用逆深度来表示。设特征点在相机坐标系下的坐标为 (x_C, y_C, z_C) , 其逆深度表示为:

$$[\alpha, \beta, \rho]^T = \left[\frac{x_C}{z_C}, \frac{y_C}{z_C}, \frac{1}{z_C} \right]^T \quad (4-37)$$

对特征点 f_j , 选取 C_n 帧为参照帧, 根据相机之间的相对位姿关系可以将其转到任意一帧相机坐标系中:

$${}^{C_i}p_{f_j} = C\left({}^{C_n}_Cq\right) {}^{C_n}p_{f_j} + {}^{C_i}p_{C_n} \quad (4-38)$$

将逆深度坐标代入可得:

$${}^{C_i}p_{f_j} = {}^{C_n}z_j \left(C \left({}^{C_i}q \right) \frac{{}^{C_n}y_j}{{}^{C_n}z_j} + \frac{1}{{}^{C_n}z_j} {}^{C_i}p_{C_n} \right) = {}^{C_n}z_j \left(C \left({}^{C_i}q \right) \begin{bmatrix} \alpha_j \\ \beta_j \\ 1 \end{bmatrix} + \rho_j {}^{C_i}p_{C_n} \right) \quad (4-39)$$

设特征点 f_j 在 $C_1, \dots, C_n$ 相机下的观测分别为 $z_1^{(j)}, \dots, z_n^{(j)}$ ，则最小二乘优化问题为：

$$\min \frac{1}{2} \sum_{i=1}^n \|e_i(\alpha_j, \beta_j, \rho_j)\|^2 = \min \frac{1}{2} \sum_{i=1}^n \|z_i^{(j)} - \hat{z}_i^{(j)}\|^2 \quad (4-40)$$

该方程通过 L-M 方法进行最小二乘求解，即可得到三角化后的特征点坐标。对于单个三角化的特征点，会保存所有观测到该点的相机位姿及观测信息。特征点和相机位姿之间的残差定义如下：

$$\begin{cases} r_i^{(j)} = z_i^{(j)} - \hat{z}_i^{(j)} \\ \hat{z}_i^{(j)} = \frac{1}{{}^{C_i}\hat{Z}_j} \begin{bmatrix} {}^{C_i}\hat{X}_j \\ {}^{C_i}\hat{Y}_j \end{bmatrix}, \begin{bmatrix} {}^{C_i}\hat{X}_j \\ {}^{C_i}\hat{Y}_j \\ {}^{C_i}\hat{Z}_j \end{bmatrix} = C \left({}^{C_i}\hat{q} \right) \left({}^G\hat{p}_{f_j} - {}^G\hat{p}_{C_i} \right) \end{cases} \quad (4-41)$$

其中 $z_i^{(j)}$ 为特征 f_j 在相机帧 C_i 的观测值， $\hat{z}_i^{(j)}$ 为该特征在此相机帧下的重投影。对所有特征点的残差模型进行线性化并合并，可以得到整体残差模型：

$$r_o = H_x \tilde{X} + n_o \quad (4-42)$$

其中 n_o 表示噪声，系数矩阵 H_x 由式(4-43)进行定义：

$$H_x = \begin{bmatrix} H_x^{(1)} \\ H_x^{(2)} \\ \vdots \\ H_x^{(N)} \end{bmatrix} \quad (4-43)$$

其中每一个小分量代表的是残差对状态向量的雅可比。具体的推导及一些数学处理，参考论文[61]。

4.4.5 EKF 更新策略

在 EKF 更新之前，先对模型进行简化处理。在式(4-43)中， H_x 的大小与状态向量的总长度有关，因此如果当状态向量中维持的相机位姿数较多时， H_x 的维度是相当可观的。为了减少观测模型的规模，对 H_x 进行 QR 分解^[62]：

$$H_x = [Q_1 \quad Q_2] \begin{bmatrix} T_H \\ 0 \end{bmatrix} \quad (4-44)$$

其中 Q 是分解后的正交矩阵，而 T_H 是上三角矩阵。将该分解结果代入式(4-42)可得：

$$r_n = Q_1^T r_o = T_H \tilde{X} + n_n \quad (4-45)$$

更新过程只需要套用 EKF 更新公式即可，其中卡尔曼增益 K 由如下公式进行计算：

$$K = P T_H^T (T_H P T_H^T + R_n)^{-1} \quad (4-46)$$

式中 R_n 是噪声 n_n 的协方差矩阵。状态向量的更新量 Δx 由式(4-47)进行计算:

$$\Delta X = K r_n \quad (4-47)$$

协方差矩阵由如下等式进行更新:

$$P_{k+1|k+1} = (I - K T_H) P_{k+1|k} (I - K T_H)^T + K R_n K^T \quad (4-48)$$

其中 I 表示单位矩阵。

EKF 的更新阶段中,借助了相机位姿对特征的观测。如图 4-8 所示,当视觉系统接收到新的图像或某一跟踪的特征点从图像序列中消失后, EKF 都需要触发更新。对于后者,直接使用 4.4.4 节中的方法直接对消失的特征点进行三角化,并扩增到状态向量中。当记录新图像时,当前相机姿态估计值会直接加入到状态向量中。如果此时相机位姿数已经达到了上限,则将最早的那一帧相机位姿进行移除。

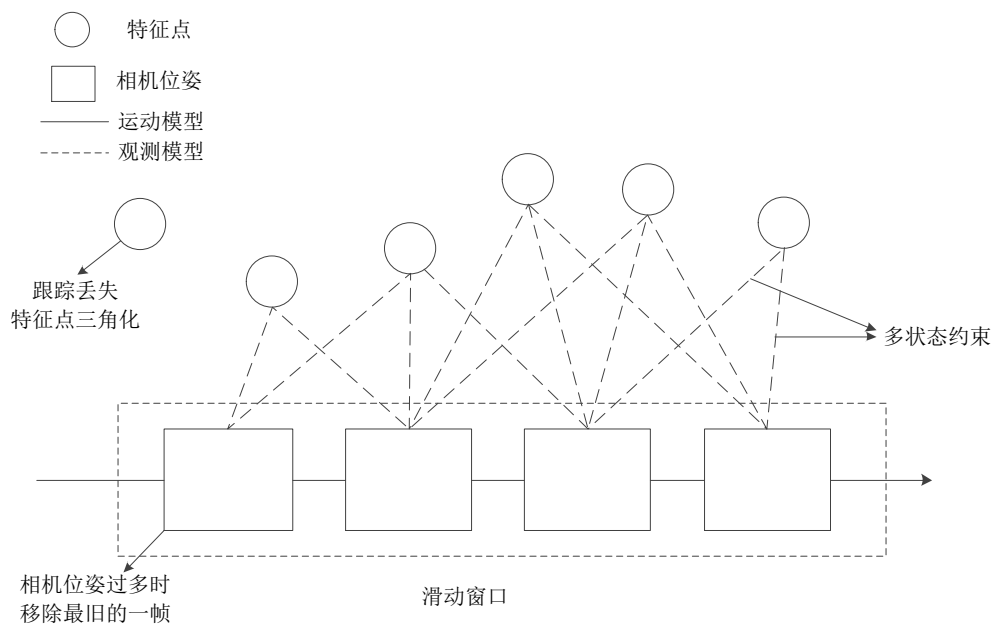


图 4-8 更新策略

4.5 数据集仿真实验

本节利用数据集对本文改进的视觉惯导紧耦合里程计算法进行实验仿真。结合本课题的应用场景和所用传感器类型,使用 EuRoC 数据集进行仿真实验。该数据集由苏黎世联邦理工学院提供,记录了无人机在室内飞行时双目相机和 IMU 的运动数据。该数据集总共包含机房(MH)和普通房间(V)两种场景,根据无人机飞行状态以及环境条件的不同,每个场景将数据集分为简单(easy)、中等(medium)、困难(difficult)三个等级,随着困难等级的提升,无人机的飞行速度会变快,且环境光照变化越剧烈。

为了更好地进行对比分析,本节分别对不同的视觉惯导里程计算法进行仿真实验,包括 VINS_Fusion、MSCKF 以及本文改进的 MSCKF 算法。为了更准确地比较里程计精度,在运行 VINS_Fusion 时关闭其回环检测功能。仿真过程中使用了以下几个数据集: V1_01_easy、V1_02_medium、V1_03_difficult、MH_02_easy、MH_03_medium、MH_04_difficult。实验仿真界面如图 4-9 所示:

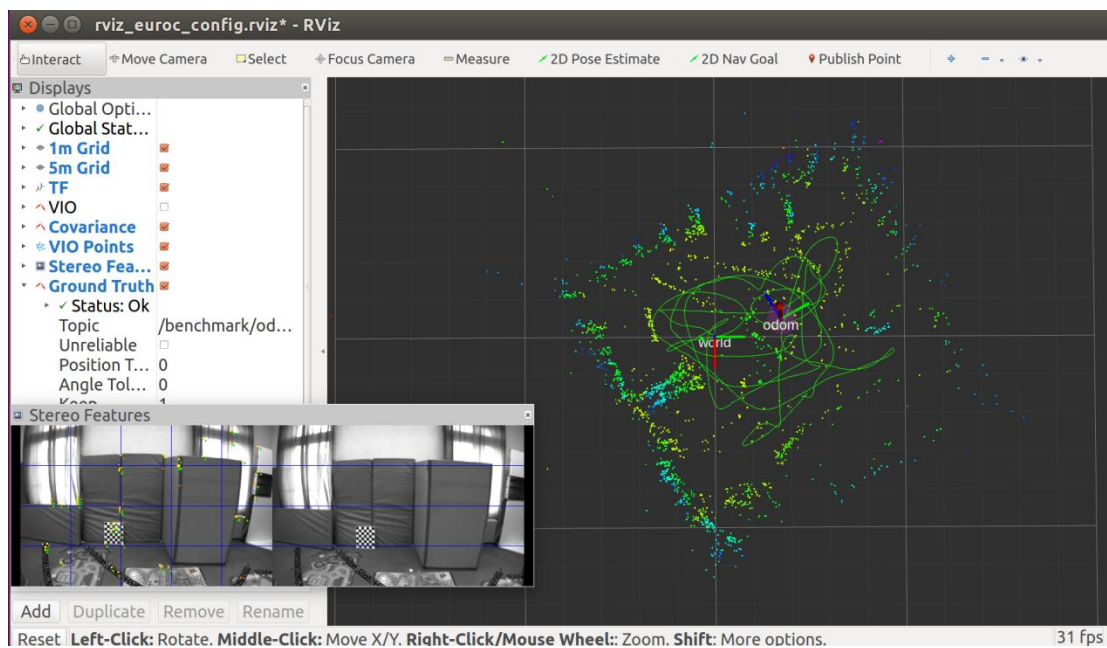
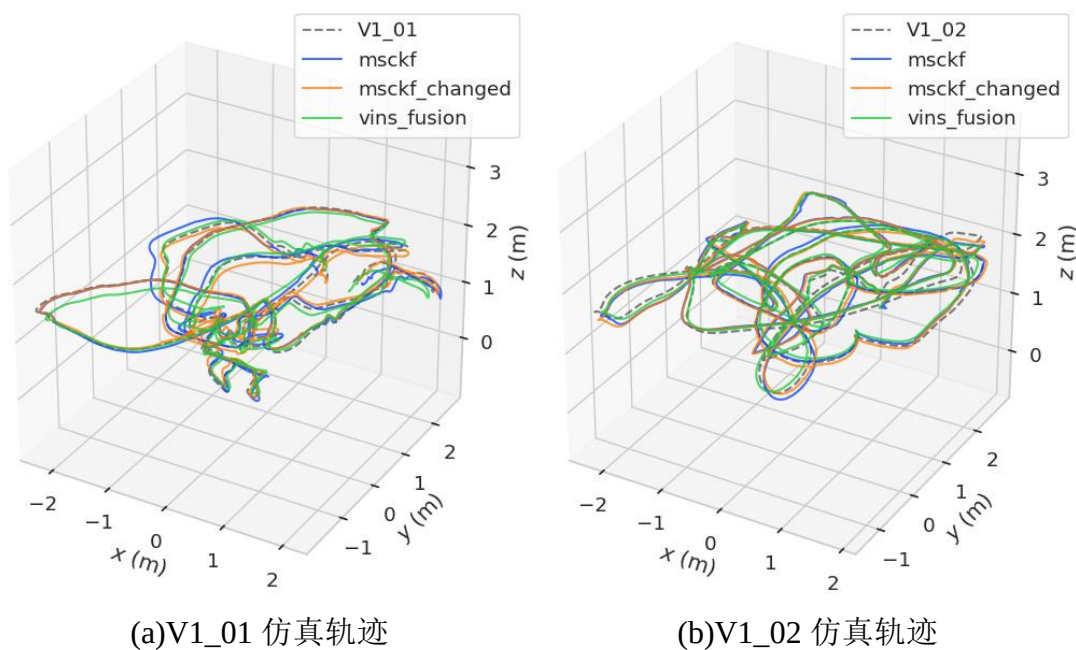


图 4-9 数据集仿真实验界面

仿真过程中记录各个算法跑不同数据集所估计的位姿数据，并绘制如下所示的轨迹图：



(a)V1_01 仿真轨迹

(b)V1_02 仿真轨迹

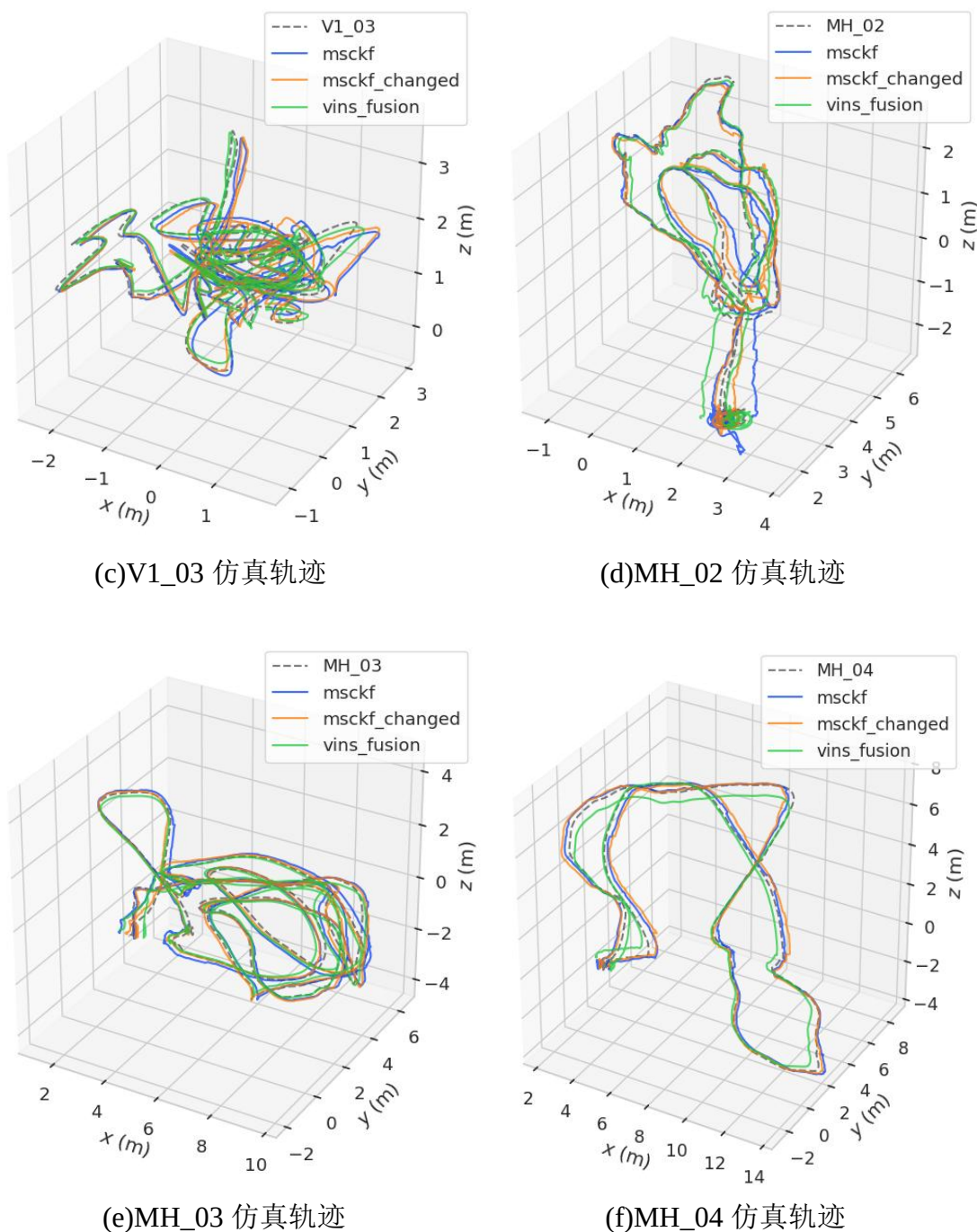


图 4-10 数据集仿真轨迹图

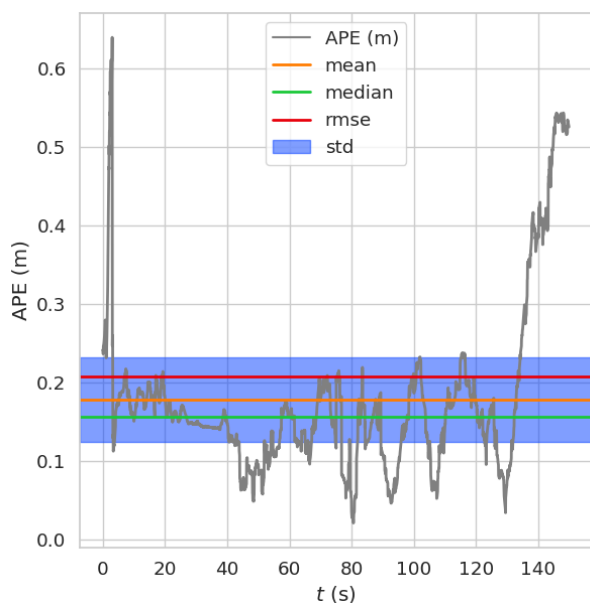
图中黑色虚线部分代表各个数据包所提供的无人机飞行真实轨迹。从图中可以看出，三种算法所计算得出的轨迹形状与真实轨迹基本一致，但是在路径吻合度上存在较大差别，为了定量地比较不同算法路径的吻合度，引入绝对姿态误差（Absolute Pose Error, APE）指标进行评价。APE 直接计算每一帧位姿的真实值与算法估计值之间的差，从而评定算法的整体定位精度。APE 的计算公式如下：

$$E_{ape}^i = Q_i^{-1} P_i \quad Q, P \in SE(3) \quad (4-49)$$

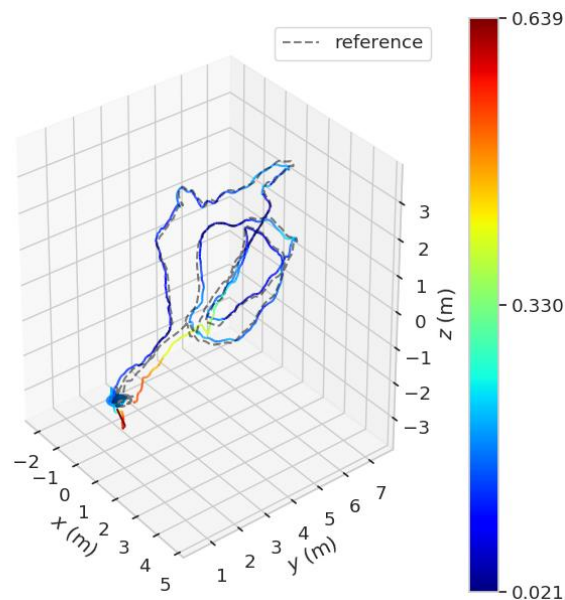
其中， Q_i 表示第 i 帧的真实轨迹， P_i 表示估计轨迹。APE 的评估指标有很多，包括平均值、中位数、均方根误差（Root Mean Square Error, RMSE）等等，本文使用 RMSE 来描述相对误差情况，其定义如下：

$$RMSE(E_{ape}) = \left(\frac{1}{m} \sum_{i=1}^m \left\| trans(E_{ape}^i) \right\|^2 \right)^{\frac{1}{2}} \quad (4-50)$$

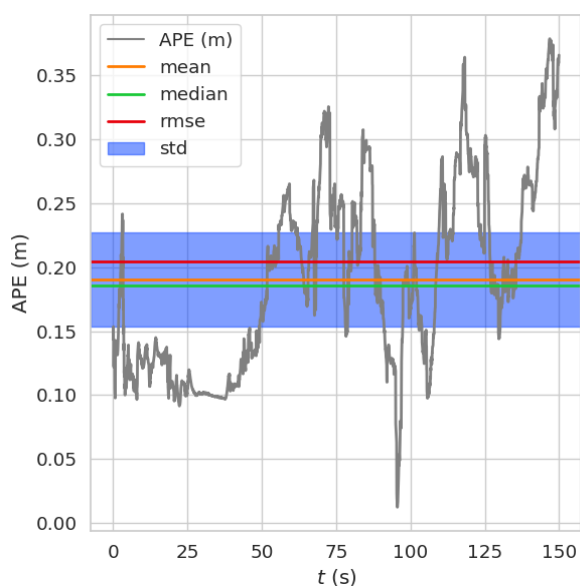
其中 $trans(E_{ape}^i)$ 为绝对位姿误差中的平移部分, m 为总的位姿个数。RMSE 值越小, 则该里程计算法估计的位姿更接近于真实值, 定位精度越高。以 MH_02 为例, 计算 APE 的效果如图 4-11 所示, 不同算法跑所有数据包的结果记录在表 4-1 中:



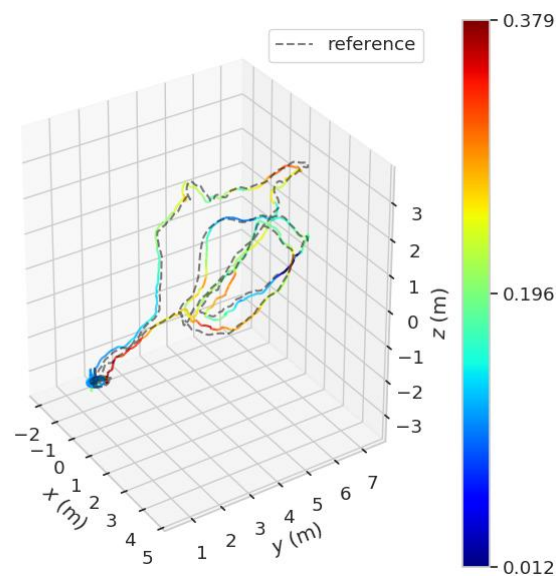
(a) MSCKF 算法 APE 结果



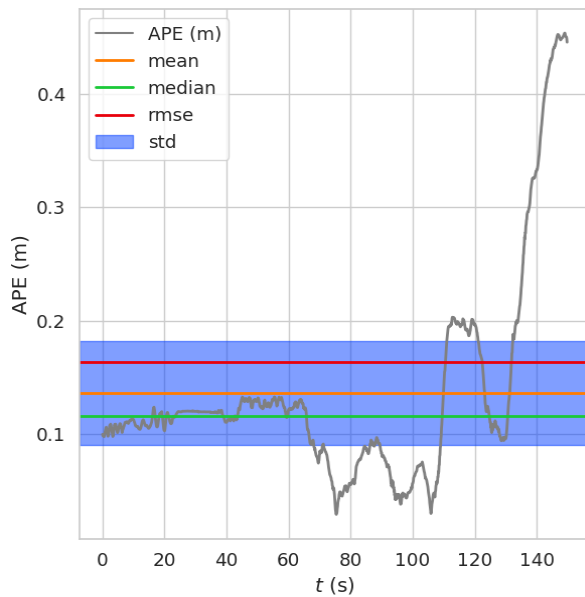
(b) MSCKF 算法轨迹误差



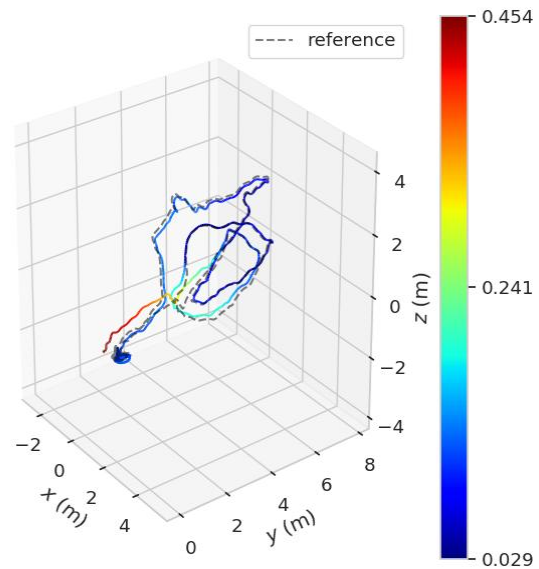
(c) 本文算法 APE 结果



(d) 本文算法轨迹误差



(e) VINS_Fusion 算法 APE 结果



(f) VINS_Fusion 算法轨迹误差

图 4-11 MH_02 数据集 APE 值计算结果

表 4-1 绝对位姿均方根误差

数据集	MSCKF(m)	MSCKF_changed(m)	VINS_Fusion(m)
MH_02_easy	0.213691	0.202241	0.163655
MH_03_medium	0.268799	0.205399	0.286072
MH_04_difficult	0.310635	0.324117	0.422505
V1_01_easy	0.095045	0.073065	0.116821
V1_02_medium	0.111361	0.098461	0.094772
V1_03_difficult	0.251352	0.240963	0.148547

根据表中数据,对于同一种算法而言,数据包的难度等级上升,其对应的 RMSE 值基本呈上升的趋势,定位精度随之下降。基于优化的 VINS_Fusion 算法与基于滤波的 MSCKF 框架在不同数据集上的表现各有千秋,VINS_Fusion 在普通房间内进行短距离、大幅度运动时定位精度要优于滤波框架,而基于滤波的 MSCKF 框架在机房大尺度飞行下的总体表现更好。

进一步分析改进前后 MSCKF 算法的定位差距,使用相对姿态误差(Relative Pose Error, RPE)进行评估。RPE 主要描述相隔固定时间差的两帧之间,真实位姿和估计位姿之间的误差,相当于直接测量里程计的误差,能更好地描述轨迹漂移。设两帧时间间隔为 Δ ,则第 i 帧位姿对应的 RPE 定义如下:

$$E_{rpe}^i = (Q_i^{-1} Q_{i+\Delta})^{-1} (P_i^{-1} P_{i+\Delta}) \quad Q, P \in SE(3) \quad (4-51)$$

其中, Q_i 表示第 i 帧位姿的真值轨迹, P_i 表示位姿的估计轨迹。仍使用 RMSE 来描述相对误差情况,其定义如下:

$$RMSE(E_{rpe}) = \left(\frac{1}{m} \sum_{i=1}^m \left\| \text{trans}(E_{rpe}^i) \right\|^2 \right)^{\frac{1}{2}} \quad (4-52)$$

其中 $\text{trans}(E_{rpe}^i)$ 为相对位姿误差中的平移部分, m 为总的位姿个数。根据定位结果计算出的 RMSE 值越小, 则该里程计的漂移越小。计算每组实验的 RMSE 值, 将结果统计在表 4-2 中:

表 4-2 相对位姿均方根误差

	V1_01	V1_02	V1_03	MH_02	MH_03	MH_04
MSCKF(m)	0.053199	0.101143	0.061926	0.044573	0.114828	0.098539
本文算法(m)	0.037482	0.069791	0.062033	0.039660	0.102438	0.098072

根据表中数据显示, 本文对 MSCKF 后端框架的改进有一定效果, 改进后的算法在处理 easy 和 medium 级别的场景时, 相较于原本的 MSCKF 算法具有更好的累积误差抑制作用。但对于 difficult 场景, 两者结果非常接近, 甚至原本的 MSCKF 算法结果要略好。这是因为在 difficult 场景中, 光照环境变化明显, 视觉前端提取到的特征点有限, 此时在后端滤波扩展相机状态时, 仍然以 IMU 的运动数据进行计算, 无法起到累积误差补偿作用。另外当无人机做大幅度的运动时, 根据 PnP 算法计算出的旋转变换可能存在较大误差, 用此结果进行补偿的效果也不理想。综上所述, 本文改进的视觉惯导里程计在室内光照环境一致性较强, 无人机运动较为平稳时, 在轨迹估计上有更小的漂移, 定位表现较为优异, 而对于苛刻的定位场景而言, 本文提出的累计误差抑制方法没有较好的效果。

4.6 本章小结

本章节主要研究视觉与惯导的融合导航定位算法, 阐述了基于 EKF 滤波的视觉与 IMU 组合定位框架模型。为了满足系统定位的实时性要求, 在视觉前端采用直接法, 对图像提取 FAST 角点后, 利用 LK 光流进行特征点跟踪; 针对室内动态环境的影响, 借助 IMU 旋转测量信息, 对环境中的特征点运动进行补偿, 并使用改进的 K-means 算法对动态特征点进行识别滤除; 最后介绍了基于滤波的视觉与 IMU 紧耦合算法, 将不同时刻的相机位姿和 IMU 状态作为状态向量进行估计, 并融合视觉与 IMU 信息来扩增相机位姿状态, 以降低 IMU 累积误差对系统定位精度的影响。利用环境中特征点在多个相机状态之间形成的几何约束, 构建观测模型并对 EKF 进行更新, 从而实现无人机的位姿估计。仿真实验表明, 本文算法在 EuRoC 数据集上的定位精度与 VINS_Fusion 相接近, 轨迹漂移整体也小于经典的 MSCKF 算法。

第五章 无人机飞行实验设计与分析

在前面的章节中，对动态场景下基于滤波的视觉惯导里程计算法进行了研究，并通过无人机数据集仿真对后端改进的紧耦合算法进行了评估。但由于 EuRoC 数据集中不包含明显的动态特征，因此无法对动态识别算法的有效性进行验证。本章将搭建无人机实验平台，并在实际的室内动态场景中开展无人机飞行实验，评估动态滤除算法的实际效果，并分析本文 VIO 算法在无人机实验平台上的定位精度，验证其在实际室内动态场景中的适用性。

5.1 实验平台与场景

本节通过将视觉惯导里程计算法移植到无人机套件上，评估该算法在实际室内动态环境中的定位表现。图 5-1 所示为本实验所使用的无人机开发套件，除了四旋翼无人机机体外，还包括 Pixhawk 无人机飞控、英伟达 Jetson TX2 核心开发板、小觅双目相机以及 3S 电池。其中，小觅双目相机内部集成了 MEMS 级别的 IMU 模块，其相关的技术规格在 3.5 节中已详细介绍。为保证 VIO 算法的准确运行，使用 3.5 节的标定实验结果作为无人机平台的传感器标定参数。TX2 开发板为本文定位算法移植的嵌入式平台，其板载双核 NVIDIA Denver 2 64 位 CPU 与四核 ArmCortex-A57 复合处理器，拥有 8G 的运行内存，32G 的存储空间，运行功耗最低为 15w，适合用于无人机机载算法验证。

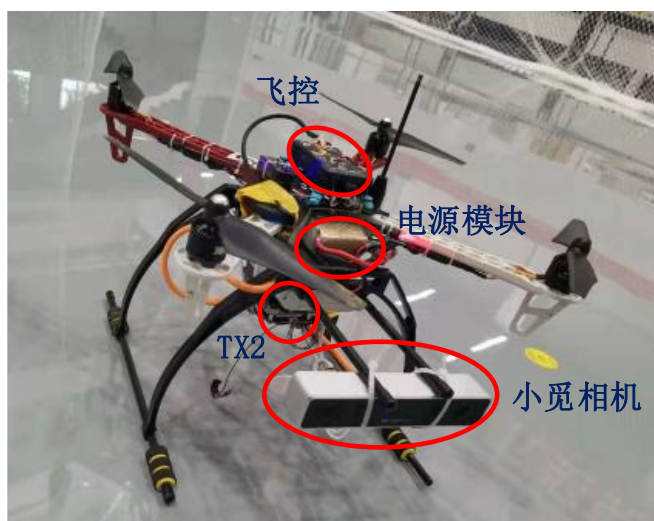


图 5-1 无人机实验平台

实验过程中，使用遥控器控制无人机飞行，借助 SSH 服务使个人电脑远程连接 TX2 开发板，实现在个人电脑上远程控制 TX2 运行定位算法并记录传感器数据。TX2 开发板通过 USB 接口实时获取小觅相机传输的图像信息与 IMU 数据。为了节省无人机平台的整体功耗，实验过程中 TX2 处于默认的 Max-Q 模式，此模式下，只有 4 块 ARM CortexA57 进行工作。

本次实验地点为东南大学四牌楼校区李文正楼一楼大厅，实验实景如图 5-2 所示，该实验场所存在较大尺度的运动空间，适合用于本课题实验，实验场景中出现的行人可以为实验提供的动态特征。

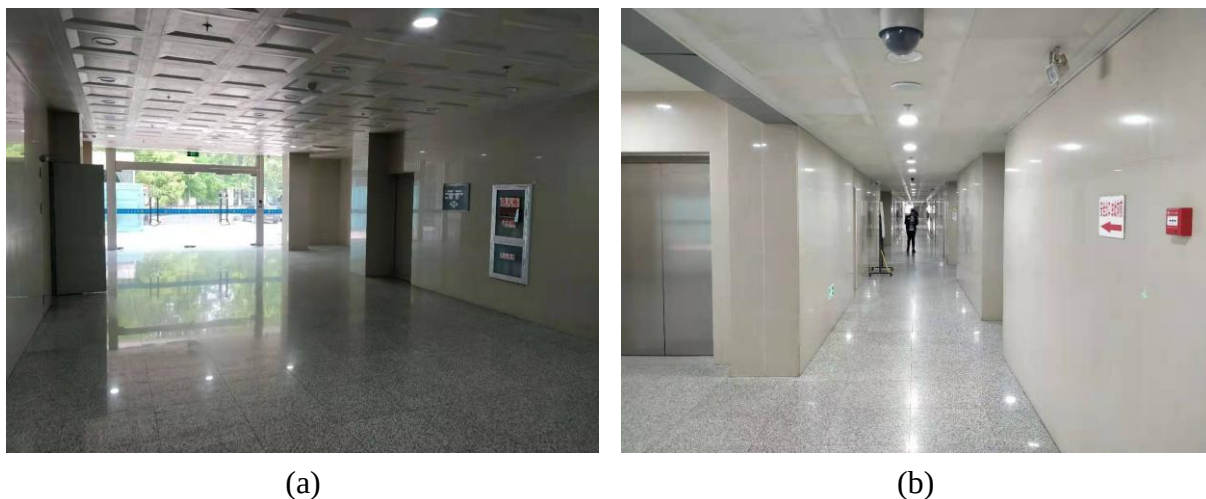
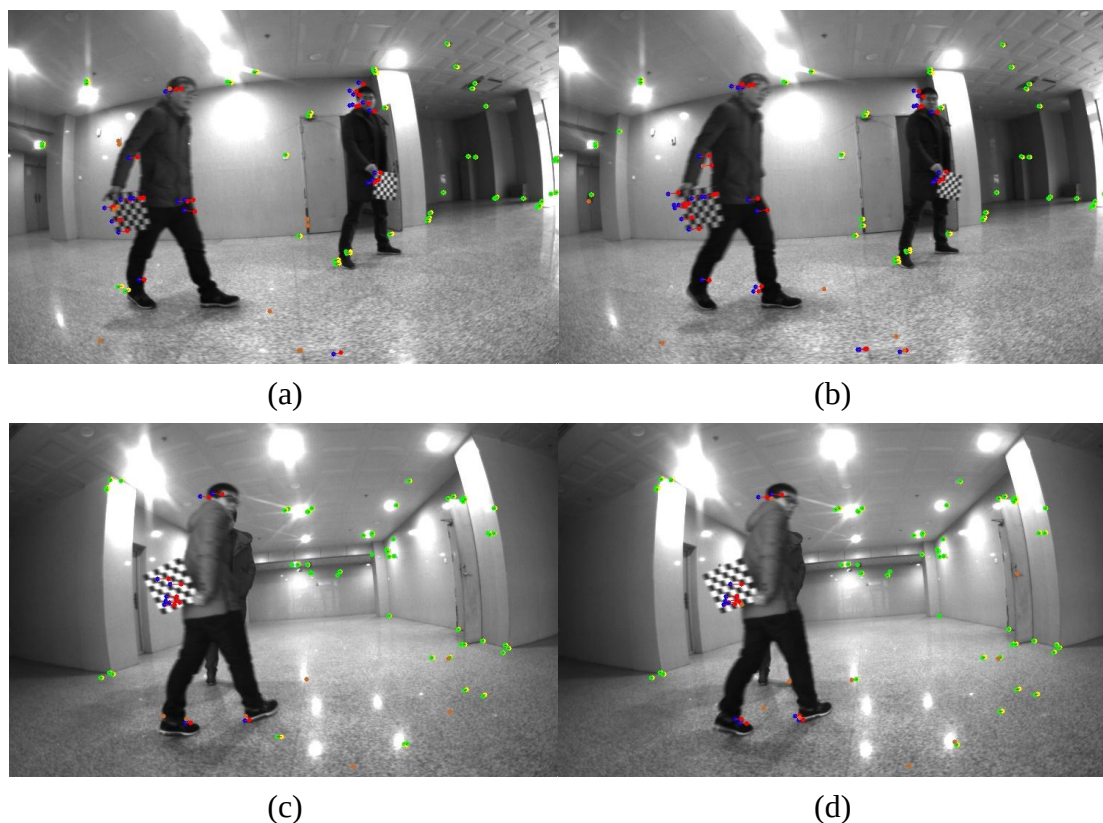


图 5-2 实验场景

5.2 动态特征识别实验

本文使用的视觉惯导里程计算法在识别动态特征点时，利用了两帧图像之间 IMU 的测量信息，对视觉传感器的运动进行补偿，根据补偿后的特征运动变化区分环境中动、静特征。由于现阶段还没有包含很多动态特征的室内 VIO 数据集，因此本实验使用无人机自行采集数据对动态特征识别算法的效果进行评估。

为了模拟室内动态环境，在录制数据时，会有行人在环境中任意走动。行人在行走时会手持黑白棋盘格标定板，以提高其上特征点被提取的概率。为了保证无人机能观测到运动中的动态特征，将其飞行高度保持在 1m-1.5m，飞行速度约为 1m/s，行人为正常的行走速度。整个实验过程中，视觉前端总共处理了 1034 张图像，选取其中三组连续的动态图像进行分析：



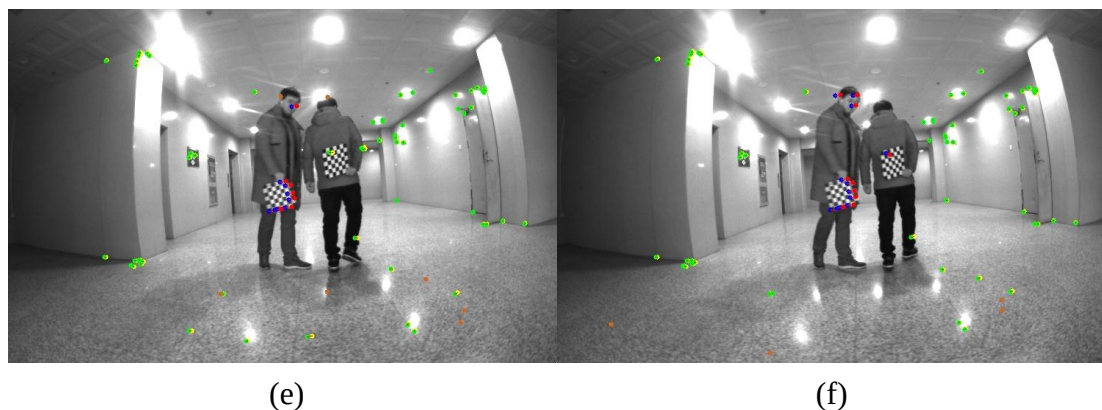


图 5-3 动态特征识别实验图

图 5-3 为前端特征点提取的实验效果图，图中黄色点表示在当前帧中光流所追踪到的静态特征点，绿色点是该点在上一帧图像中的坐标，红色点表示在当前帧中追踪到的动态特征点，蓝色点是该点在上一帧中的匹配点。每一对匹配点中间都有红色小短线相连接，红线越长代表该特征在前后两帧中运动的距离差越大。另有小部分橙色点表示当前帧中所提取的新特征。从图像中可以看出，当环境中出现动态行人时，视觉前端会在其上提取相关特征，如果把这些特征点当作静态特征放入后端优化估计，会影响算法整体的定位精度。从图像中也可以发现，本文提出的基于运动分割的动态滤除方法，能够在此场景下较为准确地识别出大量动态特征点，但也有部分属于动态行人的特征点并没有被当作动态特征识别出来，例如行人脚部的一些特征，这是因为此类特征虽然属于动态物体，但是其在前后两帧中并没有发生较为明显的运动，因此无法更精准地利用运动分割方法将其和场景中的静态特征进行区分。

本文使用查准率 P 和查全率 R 作为动态特征检测的评价指标，两者的定义如下：

$$\begin{cases} P = TM / (TM + FS) \\ R = TM / (TM + FM) \end{cases} \quad (5-1)$$

其中， TM 为动态物体上被正确识别出的特征点数， FM 为动态物体上未识别的特征点数， FS 为静态区域内被误识别为动态特征的数量。

对图 5-3 中的相关特征点数进行统计，结果如表 5-1 所示。从表中可以看出，上述六幅图像中的查准率比较高，基本不存在静态错检的情况，不同图像的查全率存在一定的差别，但整体表现较好，平均查全率能达到 87.19% 左右，说明本文提出的方法能较为有效地识别出该场景下的动态特征点。

表 5-1 动态特征提取结果

序号	动态正确识别	动态漏检	静态错检	查准率	查全率
a	22	6	1	95.65%	78.57%
b	25	4	2	92.59%	86.21%
c	10	0	0	100%	100%
d	11	0	0	100%	100%
e	8	4	0	100%	66.67%
f	11	1	0	100%	91.67%

另外观察图 5-3 的(a)、(b)可以发现,静态环境区域中的错检点主要集中于图像的边缘部分。根据分析,一方面由于图像边缘的地面环境有较为相似的图像点特征,灰度值比较接近,光流跟踪时可能会产生误匹配的情况。另一方面,由于需要保证算法的实时性,在视觉前端使用的是传感器采集的原始图像,并没有对其进行畸变矫正,同时这些误识别点在图像中所处位置均接近于图像边缘,此处图像像素容易发生桶形畸变。以上两种原因都可能导致像素点的运动距离较大,超出了静态特征点的运动阈值,从而被聚类算法误认为是环境中的动态特征。针对图像畸变导致的误识别情况,可以采取畸变矫正的方式进行简单优化。为了不影响算法的实时性,在进行畸变矫正时并不对整张图像进行去畸变处理,先采用 4.3 节中的方法识别出场景中的动态特征点,再将所有的动态特征点进行畸变矫正。计算矫正后所有动态特征点的像素差,并重新进行聚类判断。特征点去畸变处理后,静态特征的误识别情况有良好的改观。图 5-4 是图 5-3(a)、(b)加入了去畸变优化后的处理结果,从中可以看出,图像底部所追踪到的静态环境特征已被准确识别,并且对行人身上的动态特征提取没有明显的影响。

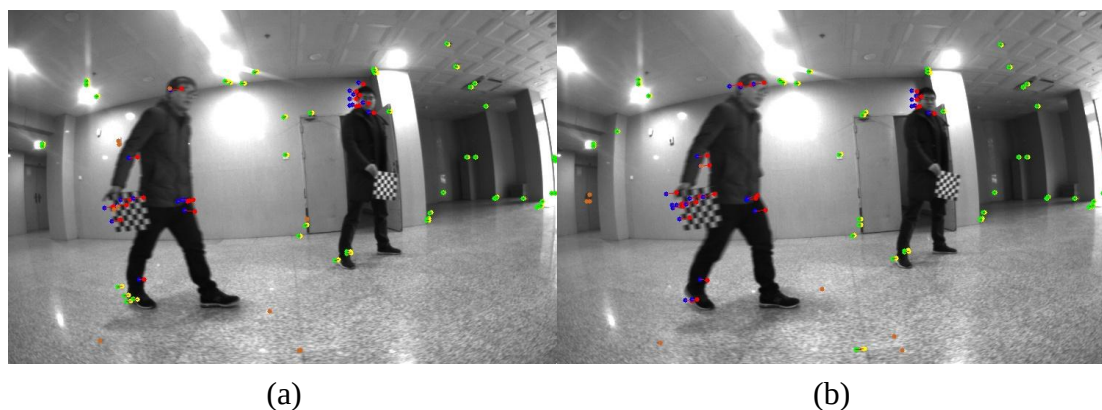


图 5-4 去畸变后的动态识别效果

本组实验采集的 1034 张图片中,存在有动态环境的图片 255 张,在这些图片中对提取出的各类特征点进行统计计数,计算每一副图像的查全率与查准率指标。经分析,整体的平均查全率为 79.36%,平均查准率为 91.22%,查全率最低为 50%,查准率最低为 52.94%,在动态行人身上特征点提取较少的图像中,两项指标最高能达到 100%。综合上述分析可以认为,在该实验场景下,本文提出的动态特征滤除方法具有较好的识别效果。

5.3 定位精度评估

在无人机平台飞行实验过程中,由于条件所限,无法准确获取无人机的实时位置信息,因此在本实验过程中,限定无人机的飞行路迹,使其形成闭环,通过评判无人机飞行始末的位置差来评判本算法的定位精度。本实验共设计三种飞行路径:

路径一:在狭长空间中进行直线折返飞行

路径二:沿室内大厅边缘矩形框飞行

路径三:在室内大厅中自由飞行

实际飞行实验的算法运行界面如图 5-5 所示。实验中采集无人机在三种路径下的飞

行数据，并使用改进前后的 MSCKF 算法计算无人机的飞行轨迹，两种算法在不同路径下的飞行结果如图 5-6 所示：

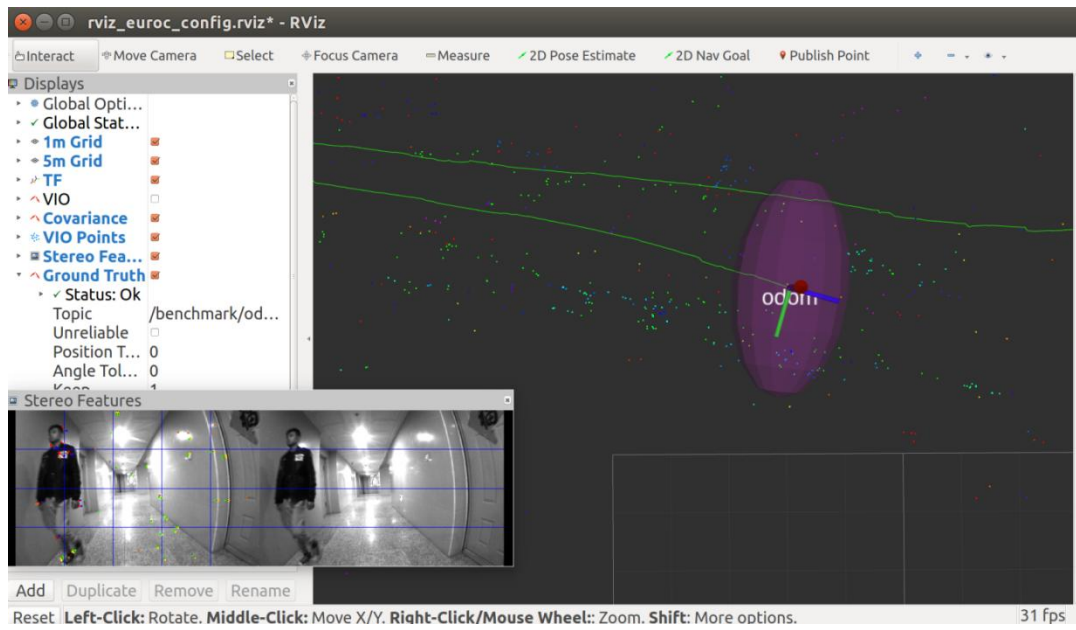
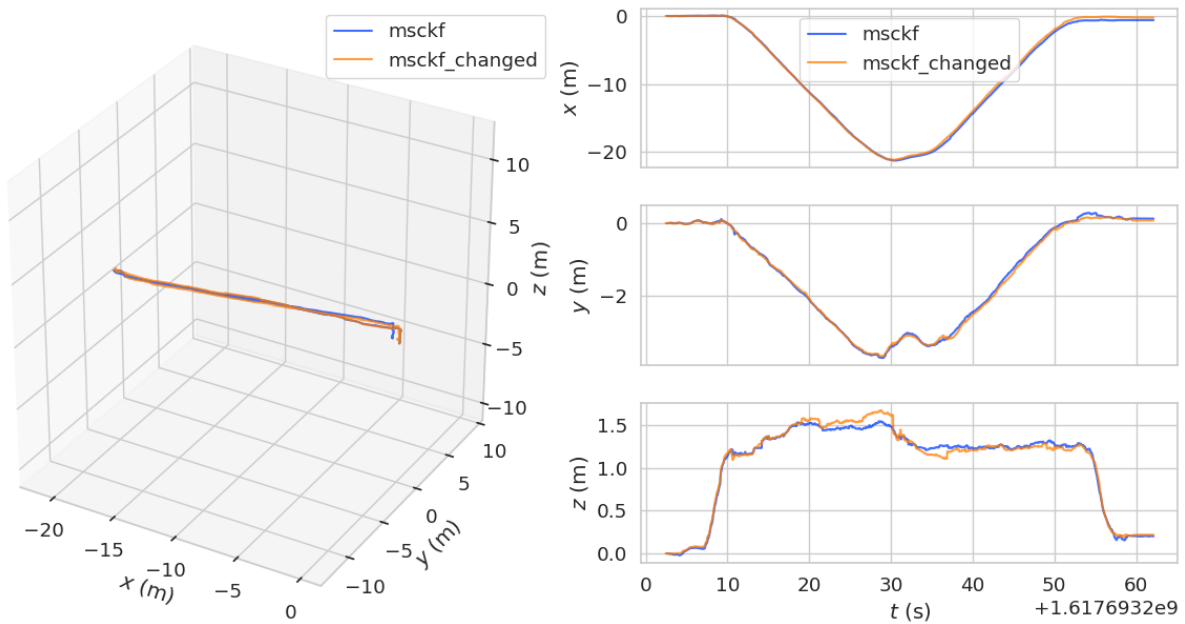
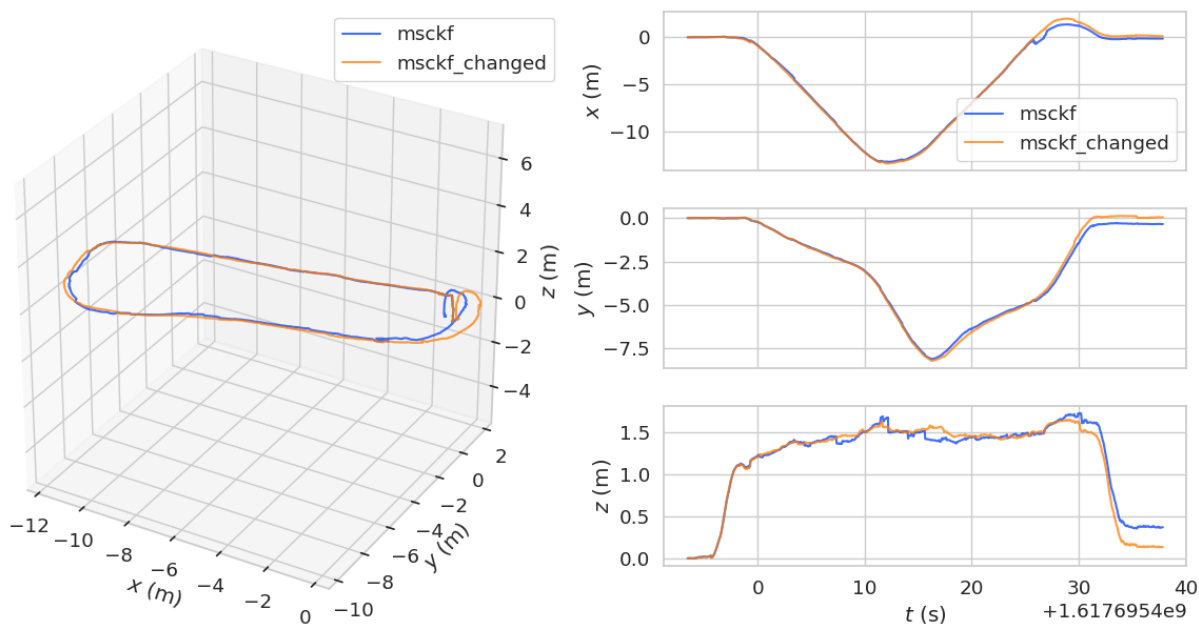


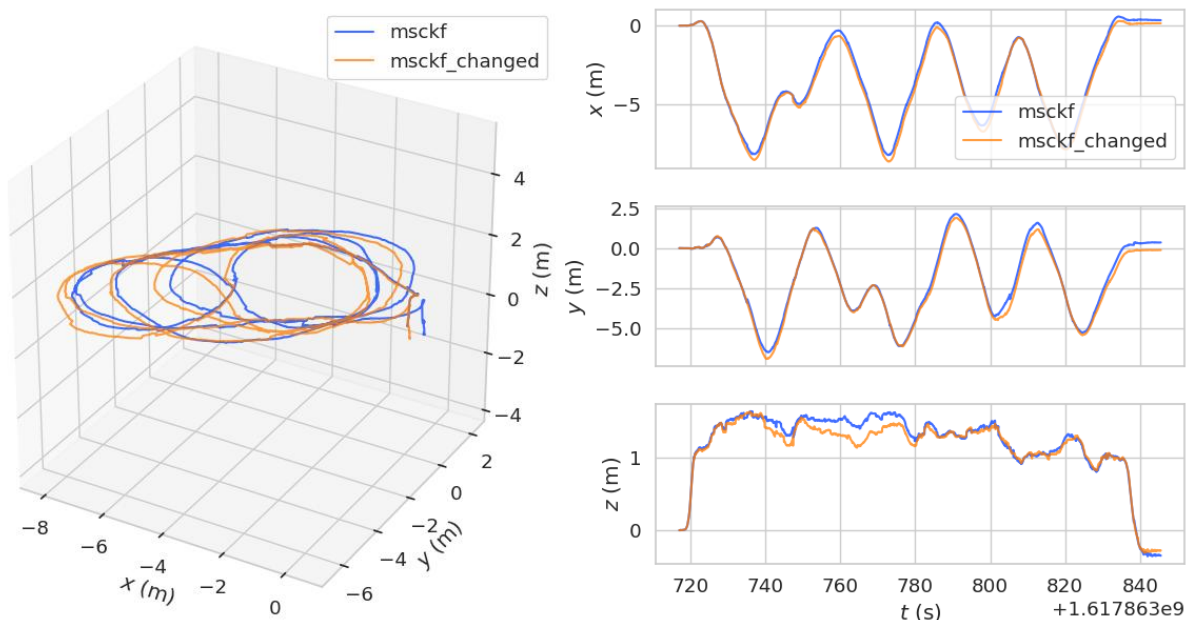
图 5-5 无人机飞行实验算法运行图



(a) 路径一飞行轨迹



(b)路径二飞行轨迹



(c)路径三飞行轨迹

图 5-6 无人机飞行实验轨迹图

从以上三种轨迹图中可以看出,改进后的算法回环结果普遍更接近于飞行起始位置。在轨迹 1 和轨迹 2 中的直线飞行路段,两种算法的估计轨迹基本吻合,而在轨迹 2、3 中出现旋转场景的时候,两种轨迹出现明显区分,原始的 MSCKF 算法在旋转时轨迹出现漂移,从而在最终闭环时出现一定的尺度偏差,而改进后的算法闭环结果更加准确。在不同的方向轴上,改进后的算法在轨迹二的 z 轴方向和轨迹三的三个轴方向对误差都有较好抑制作用,但总体而言其在 z 轴方向的估计偏差依旧较为明显。

定义闭环误差对实验的闭环精度进行估计,具体定义如下:

$$e = \sqrt{(x' - x)^2 + (y' - y)^2 + (z' - z)^2} \quad (5-2)$$

其中 (x',y',z') 为轨迹终点的坐标, (x,y,z) 为起始坐标。使用两种算法对三种轨迹进行多次估计,对结果计算平均值并记录到表 5-2 中。分析表中数据可知,与原本的 MSCKF 算法相比,改进后的 MSCKF 闭环偏差更小,相较改进前降低了约 43%。

表 5-2 轨迹偏差与闭环精度

	轨迹 1	轨迹 2	轨迹 3
飞行距离(m)	32.314	43.085	103.093
MSCKF 闭环误差(m)	0.663	0.554	0.772
本文算法闭环误差(m)	0.317	0.348	0.466

5.4 本章小结

本章节主要在无人机实验平台上对本文提出的算法进行实验验证。在实际的室内动态环境中,本文提出的动态识别算法能较好地地区分场景中的动、静特征,实验数据表明,在本文的实验场景下,动态识别算法的动态特征查全率与查准率均能保持在 50%以上,其中平均查全率为 79.36%,平均查准率高达 91.22%。改进前后算法估计的轨迹形状基本一致,改进后算法的闭环定位精度要优于改进前的 MSCKF 算法,闭环误差减少了将近一半,可以认为本文提出的无人机 VIO 算法能在室内动态环境下取得良好的定位效果。

第六章 总结与展望

本文主要研究室内动态场景下的无人机视觉惯导里程计算法,在保证无人机飞行实时可靠的前提下提升定位算法的鲁棒性与准确性。本论文在经典的 MSCKF 算法基础上,添加动态特征识别与滤除功能,并在后端滤波过程中改进了相机状态扩增方法,降低 IMU 累积误差对定位精度的影响,通过数据集仿真对改进后的算法进行评估。同时搭建实际的无人机实验平台,改进传感器标定方法,获取准确的传感器内、外参数,并在实际的室内动态场景中进行定位效果评估。本文的主要内容总结如下:

(1) 分析了无人机在室内动态环境下实现自主定位所面临的问题,提出基于滤波框架的视觉惯导里程计解决方案,并介绍了视觉惯导里程计的基本原理。

(2) 为保证视觉传感器与 IMU 融合的准确性,对传感器的标定方法进行研究。针对传统相机标定方法维度信息丢失、标定结果不稳定等问题,提出改进的三维标定方法,保存标定角点完整的三维坐标信息,从而在标定过程中建立更严格的几何约束关系。与平面标定算法的结果相比,改进的三维标定算法稳定性更强,标定结果更符合针孔相机模型投影规律。同时提出 IMU 与视觉的内、外参标定方法,并对实际所用传感器模块的内外参数进行标定,保证实际无人机平台的准确运行。

(3) 研究视觉与惯导紧耦合的定位算法。使用运动分割方法识别室内环境中的动态特征点,利用 IMU 获取的运动信息对两帧图像中的特征点进行运动补偿,计算补偿后特征点在前后帧图像中的像素差,并使用改进的 K-means 算法进行特征聚类,从而识别出动态特征并予以滤除。通过实景实验,本文提出的算法能较为准确地识别出场景中大部分的动态特征。在后端滤波框架中,改进相机状态扩增方式,利用前端光流跟踪的匹配点计算相机的运动状态,将该结果与 IMU 的计算结果进行组合,以降低 IMU 累积误差对定位精度的影响。通过数据集验证,本文算法能够在保证较好实时性的前提下取得与非线性优化框架相接近的定位精度,在抑制轨迹漂移方面也优于经典的 MSCKF 算法。

(4) 以英伟达的 TX2 核心开发板为基础搭建无人机实验平台,并在实际的室内动态场景中进行飞行定位实验,通过闭环误差评价改进前后视觉惯导里程计的定位精度,体现出本文算法的优越性。

虽然本文针对课题应用场景提出了相应的改进方法,但由于本人水平有限,仍存在很多地方值得日后进一步深究与完善,具体如下:

(1) 本文提出的动态特征识别方法计算复杂度低,但仅能在环境中物体出现移动时才能实现动态检测,当运动物体突然停止运动或运动速度降低时系统仍将其作为静态特征进行处理。针对这部分特征的识别,可以考虑利用连续多帧图像对同属于同一物体的特征点进行约束,以提高检测的准确率。

(2) 本文在后端融合视觉与 IMU 测量信息时,只是对 IMU 与视觉的计算结果取平均,没有根据特殊的运动场景设置合理的置信比例,从而使得本文算法在无人机运动较为剧烈的数据集上没有体现出较好的优化效果。可以考虑根据无人机的运动状态设置

不同的数据融合置信率，从而更好地融合视觉与 IMU 的定位信息。

(3) 在无人机的实际飞行实验中，由于设备条件的限制，没能获取无人机在飞行过程中的位姿真值。后期在有条件的情况下可以根据无人机飞行的轨迹真值更好地评估算法的定位误差。

(4) 本文只研究了结合视觉与 IMU 的里程计算法，可以考虑在后期添加闭环检测等功能模块，进一步提升位姿估计的准确性。

致谢

不知不觉中，三年的硕士研究生生涯即将步入尾声。时光虽短，但是却留给了我无数珍贵的回忆，让我在历练中一步步走向成熟。

首先，我要感谢我的导师王庆老师在这三年里对我的指导与帮助。王老师一丝不苟的工作作风、严谨的治学态度、循循善诱的从教风格使我印象深刻。在学习上，王老师不仅为我们提供良好的科研环境和实践锻炼机会，同时也会时刻为我们的工作任务进行把关，给予了我们很多指导性的意见。不仅如此，王老师在生活上也非常关心我们，时常与我们亲切谈心，帮助我们在迷茫的困境中寻找人生方向，教授了我们很多为人处世的态度与方法。在此，向恩师表达由衷的感谢与敬意。

其次，特别感谢身边众多同学给予的帮助。感谢冯悠扬、杨高朝、许九靖、严超、刘鹏飞等多位博士在学术上的帮助与指导，让我及早地找到了自己的研究方向，在科研道路上少走了许多弯路。感谢鲁锦涛、刘宏明、乔云侠三位同门的照顾与陪伴，给了我很多克服困难、砥砺前行的勇气。感谢李明、徐建龙、施上、李静等师弟师妹在科研上对我的协助，让我节省了很多时间与精力。感谢王瀚、兰庭信同学在我做实验过程中给予的帮助，也十分感谢室友杨冀东、周志雄、朱留涛在生活上的关心与照顾。

最后，非常感谢我的父母。他们不仅无私地养育了我二十多年，也一直是成长路上坚实的后盾，让我无论遇到什么样的挑战与境遇，都能无所顾虑地去拼搏奋斗。正是他们的支持、理解与鼓励，让我体会到了人生的精彩。

由衷地珍惜在东南大学的求学时光，这段时光将是我人生道路上宝贵的精神财富，相信在日后岁月里都会以一个东大学子的身份而感到骄傲。

参考文献

- [1]Tuna G, Gulez K. Aided Navigation Techniques for Indoor and Outdoor Unmanned Vehicles[C]//2012 5th International Conference on New Technologies, Mobility and Security (NTMS). 2012:1-4.
- [2]Wirbel E, Steux B, Bonnabel S, et al. Humanoid robot navigation: from a visual SLAM to a visual compass[C]//IEEE International Conference on Networking Sensing and Control. 2013:678-683.
- [3]Yousif K, Hadiashar A B, Hoseinnezhad R. An Overview to Visual Odometry and Visual SLAM: Applications to Mobile Robotics[J]. Intelligent Industrial Systems, 2015,1(4):289-311.
- [4]李月华, 朱世强, 于亦奇. 工厂环境下改进的视觉SLAM算法[J]. 机器人, 2019,41(01):95-103.
- [5]景鑫, 李阳, 高嘉瑜. 基于视觉的固定翼无人机自主着陆技术研究[C]//第十届中国卫星导航年会. 中国北京: 2019:5:99-103.
- [6]陈钊, 年晓红, 熊红云, 等. 四旋翼无人机自主飞行系统优化设计与实现[J]. 控制与信息技术, 2019(01):32-38.
- [7]Oshman Y, Koifman M. Robust Navigation Using the Global Positioning System in the Presence of Spoofing[J]. Journal of Guidance Control and Dynamics, 2006,29(1):95-104.
- [8]Michael N, Fink J, Kumar V. Cooperative manipulation and transportation with aerial robots[J]. Autonomous Robots, 2011,30(1S1):73-86.
- [9]欧阳仕晗, 刘振宇, 赵怡巍, 等. 移动机器人三维激光SLAM算法研究[J]. 微处理机, 2020,41(05):58-64.
- [10]Lynen S, Achtelik M W, Weiss S, et al. A Robust and Modular Multi-Sensor Fusion Approach Applied to MAV Navigation[C]//IEEE International Conference on Intelligent Robots and Systems. 2013:3923-3929.
- [11]Strasdat H, Montiel J M M, Davison A J. Visual SLAM: Why filter?[J]. Image And Vision Computing, 2012,30(2):65-77.
- [12]Leutenegger S, Lynen S, Bosse M, et al. Keyframe-based visual-inertial odometry using nonlinear optimization[J]. International Journal of Robotics Research, 2015,34(3):314-334.
- [13]Mur-Artal R, Tardos J D. Visual-Inertial Monocular SLAM With Map Reuse[J]. IEEE Robotics and Automation Letters, 2017,2(2):796-803.
- [14]Forster C, Carlone L, Dellaert F, et al. On-Manifold Preintegration for Real-Time Visual-Inertial Odometry[J]. IEEE Transactions on Robotics, 2017,33(1):1-21.
- [15]Hartley R, Jadidi M G, Gan L, et al. Hybrid Contact Preintegration for Visual-Inertial-Contact State Estimation within Factor Graphs[C]//International Conference on Intelligent Robots and Systems. 2018:337-343.
- [16]Usenko V, Demmel N, Schubert D, et al. Visual-Inertial Mapping With Non-Linear Factor Recovery[J]. IEEE Robotics and Automation Letters, 2020,5(2):422-429.
- [17]Mourikis A I, Roumeliotis S I. A multi-state constraint Kalman filter for vision-aided inertial navigation[C]//IEEE International Conference on Robotics and Automation ICRA. 2007:3565.
- [18]Li M, Mourikis A I. Improving the Accuracy of EKF-Based Visual-Inertial Odometry[C]//IEEE International Conference on Robotics and Automation ICRA. 2012:828-835.
- [19]Bloesch M, Omari S, Hutter M, et al. Robust visual inertial odometry using a direct EKF-based approach[C]//IEEE International Conference on Intelligent Robots and Systems. IEEE, 2015:298-304.

- [20]Delaune J, Bayard D S, Brockers R. Range-Visual-Inertial Odometry: Scale Observability Without Excitation[J]. IEEE Robotics and Automation Letters, 2021,6(2):2421-2428.
- [21]Trung N, Mann G K I, Vardy A, et al. Developing Computationally Efficient Nonlinear Cubature Kalman Filtering for Visual Inertial Odometry[J]. Journal of Dynamic Systems Measurement and Control-Transactions of the Asme, 2019,141(8).
- [22]Lee K, Johnson E N. Latency Compensated Visual-Inertial Odometry for Agile Autonomous Flight[J]. Sensors, 2020,20(8):2209.
- [23]Qin T, Li P, Shen S. VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator[J]. IEEE Transactions on Robotics, 2018,34(4):1004-1020.
- [24]Huang W, Liu H, Wan W. An Online Initialization and Self-Calibration Method for Stereo Visual-Inertial Odometry[J]. IEEE Transactions On Robotics, 2020,36(4):1153-1170.
- [25]Qiu X, Zhang H, Fu W, et al. Monocular Visual-Inertial Odometry with an Unbiased Linear System Model and Robust Feature Tracking Front-End[J]. Sensors, 2019,19(8):1941.
- [26]He Y, Zhao J, Guo Y, et al. PL-VIO: Tightly-Coupled Monocular Visual-Inertial Odometry Using Point and Line Features[J]. Sensors, 2018,18(4):1159.
- [27]Brasch N, Bozic A, Lallemand J, et al. Semantic Monocular SLAM for Highly Dynamic Environments[C]//IEEE International Conference on Intelligent Robots and Systems. 2018:393-400.
- [28]Lee C S, Clark D E, Salvi J. SLAM With Dynamic Targets via Single-Cluster PHD Filtering[J]. IEEE Journal of Selected Topics in Signal Processing, 2013,7(3SI):543-552.
- [29]Bescos B, Facil J M, Civera J, et al. DynaSLAM: Tracking, Mapping, and Inpainting in Dynamic Scenes[J]. IEEE Robotics and Automation Letters, 2018,3(4):4076-4083.
- [30]Strecke M, Stueckle J. EM-Fusion: Dynamic Object-Level SLAM With Probabilistic Data Association[C]//IEEE International Conference on Computer Vision. 2019:5864-5873.
- [31]Wen S, Zhao Y, Liu X, et al. Hybrid Semi-Dense 3D Semantic-Topological Mapping From Stereo Visual-Inertial Odometry SLAM With Loop Closure Detection[J]. IEEE Transactions on Vehicular Technology, 2020,69(12):16057-16066.
- [32]Ai Y, Rui T, Lu M, et al. DDL-SLAM: A Robust RGB-D SLAM in Dynamic Environments Combined With Deep Learning[J]. IEEE ACCESS, 2020,8:162335-162342.
- [33]Xiao L, Wang J, Qiu X, et al. Dynamic-SLAM: Semantic monocular visual localization and mapping based on deep learning in dynamic environment[J]. Robotics and Autonomous Systems, 2019,117:1-16.
- [34]魏彤, 李绪. 动态环境下基于动态区域剔除的双目视觉SLAM算法[J]. 机器人, 2020,42(03):336-345.
- [35]Saputra M R U, Markham A, Trigoni N. Visual SLAM and Structure from Motion in Dynamic Environments: A Survey[J]. ACM Computing Surveys, 2018,51(2):31-37.
- [36]黄真, 李艳文, 高峰. 空间运动构件姿态的欧拉角表示[J]. 燕山大学学报, 2002(03):189-192.
- [37]武星, 邹婷. 旋转矩阵在移动机器人建模教学中的应用[J]. 中国教育技术装备, 2020(08):87-89.
- [38]张捍卫, 喻铮铮, 雷伟伟. 四元数的基本概念与向量旋转的欧拉公式[J]. 大地测量与地球动力学, 2020,40(05):502-506.
- [39]秦永元. 惯性导航[M]. 科学出版社, 2014.
- [40]高翔. 视觉SLAM十四讲[M]. 电子工业出版社, 2019.
- [41]Kalman, Dan. A singularly valuable decomposition: The SVD of a matrix[J]. College Mathematics Journal, 1996,27(1):2-23.

- [42] Nocedal J, Wright S J. Numerical Optimization[M]. Springer Science & Business Media, 2006.
- [43] Bertsekas, Dimitri P. Constrained optimization and Lagrange multiplier methods[M]. Academic press, 2014.
- [44] Sibley G, Mei C, Reid I, et al. Vast-scale Outdoor Navigation Using Adaptive Relative Bundle Adjustment[J]. International Journal of Robotics Research, 2010,29(8):958-980.
- [45] Rosen D M, Kaess M, Leonard J J. RISE: An Incremental Trust-Region Method for Robust Online Sparse Least-Squares Estimation[J]. IEEE Transactions on Robotics, 2014,30(5):1091-1108.
- [46] Zhang Z. A Flexible New Technique for Camera Calibration[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000,22(11):1330-1334.
- [47] 邹朋朋, 张滋黎, 王平, 等. 基于共线向量与平面单应性的双目相机标定方法[J]. 光学学报, 2017,37(11):244-252.
- [48] 赵桂玲, 姜雨含, 李松. IMU标定数学建模及误差分析[J]. 传感技术学报, 2016,29(06):886-891.
- [49] 郭刚强. MEMS-IMU标定补偿技术研究[D]. 西安微电子技术研究所, 2017.
- [50] Ma Q, Wang Q, Yang Y, et al. Random error identification and suppression of MEMS gyroscope based on Allan variance[J]. Transducer and Microsystem Technology, 2019,38(6):62-65.
- [51] TSAI R Y, LENZ R K. A New Technique for Fully Autonomous and Efficient 3D Robotics Hand Eye Calibration[J]. IEEE Transactions on Robotics and Automation, 1989,5(3):345-358.
- [52] Furgale P, Rehder J, Siegwart R. Unified Temporal and Spatial Calibration for Multi-Sensor Systems[C]//IEEE International Conference on Intelligent Robots and Systems. 2013:1280-1286.
- [53] Rehder J, Nikolic A, Schneider T, et al. Extending kalibr: Calibrating the Extrinsic of Multiple IMUs and of Individual Axes[C]//IEEE International Conference on Robotics and Automation ICRA. 2016:4304-4311.
- [54] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF[C]//IEEE International Conference on Computer Vision. 2011:2564-2571.
- [55] Bay H, Tuytelaars T, Van Gool L. SURF: Speeded up robust features[C]//European conference on computer vision. 2006:404-417.
- [56] Kim K, Jang D, Choi H. Real time face tracking with pyramidal Lucas-Kanade feature tracker[C]//International Conference on Computational Science and Its Applications. 2007:1074-1082.
- [57] Feng X F, Fang B. Algorithm for epipolar geometry and correcting monocular stereo vision based on a plane mirror[J]. Optik, 2021,226:165890.
- [58] Luan V T. Efficient exponential Runge-Kutta methods of high order: construction and implementation[J]. Bit Numerical Mathematics, 2021:1-26.
- [59] Lepetit V, Moreno-Noguer F, Fua P. EPnP: An Accurate O(n) Solution to the PnP Problem[J]. International Journal of Computer Vision, 2009,81(2):155-166.
- [60] Penate-Sanchez A, Andrade-Cetto J, Moreno-Noguer F. Exhaustive Linearization for Robust Camera Pose and Focal Length Estimation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013,35(10):2387-2400.
- [61] Sun K, Mohta K, Pfommer B, et al. Robust Stereo Visual Inertial Odometry for Fast Autonomous Flight[J]. IEEE Robotics and Automation Letters, 2018,3(2):965-972.
- [62] Ma G, Li H, Zhao J. Quantum QR decomposition in the computational basis[J]. Quantum Information Processing, 2020,19(8):1-16.

