分类号：TP242; TP391.41　　学校代码：10363
密　级：公开　　学　　号：2220910113

安徽工程大学
Anhui Polytechnic University

硕士学位论文

题目　四足机器人定位及视觉感知算法研究

论 文 作 者　郑方军
校 内 导 师　曹雏清
校 外 导 师　陈智君
专业学位类别　计算机技术
研究方向（领域）　智能系统与应用

论文提交日期：2025 年 5 月 9 日

分类号：TP242; TP391.41　　单位代码: 10363
密　级：公开　　学　号：2220910113

题　　目 四足机器人定位及视觉感知算法研究

题　　目 The Research on Localization and Visual Perception Algorithms for Quadruped Robot

学 生 姓 名：郑方军
校 内 导 师：曹雏清
校 外 导 师：陈智君
专业学位类别：计算机技术
研究方向（领域）：智能系统与应用

论文答辩日期：2025 年 5 月 10 日

安徽工程大学学位论文原创性声明

本人郑重声明：我恪守学术道德，崇尚严谨学风。所呈交的学位论文，是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已明确注明和引用的内容外，本论文不包含任何其他个人或集体已经发表或撰写过的作品及成果的内容。论文为本人亲自撰写，我对所写的内容负责，并完全意识到本声明的法律后果由本人承担。

学位论文作者签名：郑方军

日期：2025年 5月 9日

安徽工程大学学位论文版权使用授权书

学位论文作者完全了解学校有关保留、使用学位论文的规定，同意学校保留并向国家有关部门或机构送交论文的复印件和电子版，允许论文被查阅或借阅。本人授权安徽工程大学可以将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存和汇编本学位论文。

保密 □，在 ____ 年解密后适用本版权书。

本学位论文属于

不保密 ☑。

学位论文作者签名：郑方军　　指导教师签名：

日期：2025 年 5 月 9 日　　日期：2025 年 5 月 9 日

四足机器人定位及视觉感知算法研究

摘 要

随着人工智能的发展，机器人技术的应用日益广泛。其中，四足机器人凭借稳定性和灵活性，在物流运输，工业巡检等多个领域备受关注。同时，视觉传感器的普及显著提升了四足机器人对环境信息的获取能力。然而，在复杂多变的任务环境及机器人自身运动特性的影响下，其定位和视觉感知仍面临诸多挑战，如动态物体干扰、光照变化、视角切换等，都可能对机器人的运动鲁棒性和稳定性造成不利影响。因此，提升四足机器人的定位及视觉感知能力对其实际应用至关重要。针对上述挑战，本文深入研究了四足机器人的定位及视觉感知算法，旨在增强四足机器人在复杂场景中的稳定性和鲁棒性，具体内容如下：

第一，基于动态场景分割的视觉 SLAM 算法(DSS-SLAM)，该算法通过全景分割将动态物体与静态背景分离，生成动态掩码以剔除动态干扰，并结合光流估计和几何一致性验证优化动态特征点的分类，同时，引入深度辅助验证机制，利用深度信息提升特征点分类精度，从而增强算法在动态环境中的鲁棒性。在 TUM，EuRoC 等数据集上的实验表明，DSS-SLAM 能够有效应对多种复杂动态环境。

第二，基于深层特征匹配器(Ada-Matcher)的视觉重定位方法，该方法的核心是特征匹配技术。其中 Ada-Matcher 采用自适应权重共享，掩码注意力和特征变换技术，构建深层注意力图神经网络，减少参数量的同时，提升特征匹配的鲁棒性和精度。结合图像检索技术，实现高效的全家重定位。在 MegaDepth，InLoc 等数据集上的实验表明，Ada-Matcher 能够适应各种复杂环境下的位姿估计和视觉定位等任务。

第三，基于宇树科技 B1 四足机器人与 RealSense D455 相机构建的实验平台。通过多个真实环境测试表明所提出的 DSS-SLAM 和 Ada-Matcher 有效解决了四足机器人在面对低纹理，视角抖动，遮挡以及动态物体等复杂环境下的定位和视觉感知难题，显著提升了四足机器人的智能导航和自主探索能力。

关键词：四足机器人，全景分割，动态 Slam，特征匹配，视觉重定位

The Research on Localization and Visual Perception Algorithms for Quadruped Robot

ABSTRACT

With the development of artificial intelligence, the application of robotics technology is becoming more and more extensive. Among them, quadruped robots have attracted much attention in many fields such as logistics and transportation, industrial inspection, etc. due to their stability and flexibility. At the same time, the popularity of visual sensors has significantly improved the ability of quadruped robots to obtain environmental information. However, under the influence of complex and changeable task environments and the robot's own motion characteristics, its positioning and visual perception still face many challenges, such as dynamic object interference, lighting changes, perspective switching, etc., which may have an adverse effect on the robot's motion robustness and stability. Therefore, improving the positioning and visual perception capabilities of quadruped robots is crucial to their practical applications. In response to the above challenges, this paper conducts an in-depth study of the positioning and visual perception algorithms of quadruped robots, aiming to enhance the stability and robustness of quadruped robots in complex scenes. The specific contents are as follows:

First, a visual SLAM algorithm based on dynamic scene segmentation (DSS-SLAM) is used. This algorithm separates dynamic objects from static backgrounds through panoramic segmentation, generates dynamic masks to eliminate dynamic interference, and optimizes the classification of dynamic feature points by combining optical flow estimation and geometric consistency verification. At the same time, a depth-assisted verification mechanism is introduced to use depth information to improve the classification accuracy of feature points, thereby enhancing the robustness of the algorithm in dynamic environments. Experiments on datasets such as

TUM and EuRoC show that DSS-SLAM can effectively cope with a variety of complex dynamic environments. Second, a visual relocalization method based on a deep feature matcher (Ada-Matcher), the core of which is feature matching technology. Ada-Matcher uses adaptive weight sharing, mask attention and feature transformation technology to construct a deep attention graph neural network, which reduces the number of parameters while improving the robustness and accuracy of feature matching. Combined with image retrieval technology, efficient whole-home relocalization is achieved. Experiments on datasets such as MegaDepth and InLoc show that Ada-Matcher can adapt to tasks such as pose estimation and visual positioning in various complex environments. Third, an experimental platform built based on the Yushu B1 quadruped robot and the RealSense D455 camera. Multiple real environment tests show that the proposed DSS-SLAM and Ada-Matcher effectively solve the positioning and visual perception problems of quadruped robots in complex environments such as low texture, perspective jitter, occlusion and dynamic objects, and significantly improve the intelligent navigation and autonomous exploration capabilities of quadruped robots.

Zheng Fangjun(Computer Technology)
Supervised by Cao Chuqing

KEY WORDS: Quadruped Robot, Panoptic Segmentation, Dynamic Slam, Feature Matching, Visual Relocalization

目录

摘 要 ..... **I**
ABSTRACT ..... **II**
第 1 章 绪论 **1**
1.1 课题研究背景与意义 1
1.2 国内外研究现状 2
1.2.1 四足机器人研究现状 2
1.2.2 视觉感知相关算法研究现状 3
1.2.3 基于结构的视觉重定位方法研究现状 8
1.3 论文研究内容与结构安排 11
第 2 章 四足机器人定位及视觉感知系统搭建 **13**
2.1 引言 13
2.2 视觉定位相关理论基础 13
2.2.1 RANSAC 算法 13
2.2.2 PnP 算法 14
2.3 视觉感知相关理论基础 15
2.3.1 针孔相机模型 15
2.3.2 视觉 SLAM 系统框架 17
2.3.3 ORB-SLAM3 算法框架 20
2.4 定位及视觉感知平台搭建 22
2.4.1 四足机器人平台搭建 22
2.4.2 视觉感知方案设计 22
2.4.3 视觉重定位方案设计 23
2.5 本章小结 24
第 3 章 基于动态场景分割的视觉 SLAM 算法研究 **25**
3.1 引言 25
3.2 动态视觉 SLAM 算法框架描述 25
3.2.1 动态场景分割 26
3.2.2 运动分析 30
3.2.3 动态特征点过滤 33
3.3 实验结果与分析 34
3.3.1 TUM 数据集实验 34
3.3.2 KITTI 数据集实验 38
3.3.3 EuRoC 数据集实验 40

3.3.4 动态特征点过滤实验 41
3.3.5 消融实验 44
3.4 基于四足机器人的动态视觉 SLAM 实验分析 46
3.5 本章小结 48
第 4 章 基于深度特征匹配的视觉重定位算法研究 49
4.1 引言 49
4.2 基于特征匹配的重定位算法框架 49
4.2.1 基于余弦相似度的图像检索 50
4.2.2 基于自适应权重共享特征匹配网络 51
4.2.3 基于 PnP 的机器人位姿求解 57
4.3 实验结果与分析 57
4.3.1 单应性估计 57
4.3.2 图像匹配 59
4.3.3 位姿估计 60
4.3.4 室外视觉定位 61
4.3.5 室内视觉定位 62
4.3.6 效率分析 63
4.3.7 消融实验 64
4.4 基于四足机器人的视觉重定位实验分析 66
4.5 本章小结 68
第 5 章 总结与展望 69
5.1 研究总结 69
5.2 未来展望 69
参考文献 71
致 谢 78

表目录

表 3-1 绝对轨迹误差结果 36
表 3-2 平移相对位姿误差结果 36
表 3-3 旋转相对位姿误差结果 37
表 3-4 DSS-SLAM 与其他动态 SLAM 的 ATE 对比 38
表 3-5 KITTI 数据集实验 39
表 3-6 EUROC 数据集实验 41
表 3-7 全景分割模块消融实验 45
表 3-8 运动分析模块消融实验 45
表 3-9 深度辅助验证模块消融实验 45
表 3-10 实时性分析实验 46
表 3-11 真实环境对比实验 48
表 4-1 单应性估计实验 58
表 4-2 不同像素阈值下角度误差以及精度和召回率 59
表 4-3 室外位姿估计实验 60
表 4-4 室外视觉定位实验 61
表 4-5 室内视觉定位实验 62
表 4-6 效率分析实验 63
表 4-7 Ada-Matcher 消融实验 64
表 4-8 掩模覆盖率消融实验 65
表 4-9 不同权重共享策略的消融实验 65
表 4-10 权重共享消融实验 65
表 4- 11 实际场景重定位实验 67

图目录

图 1-1 国内四足机器人系列 3
图 1-2 ORB-SLAM 系统框架图 5
图 1-3 ORB-SLAM2 稠密建图 5
图 1-4 LSD-SLAM 系统框架图 6
图 1-5 Droid-slam 地图构建示意图 7
图 1-6 SuperPoint 示意图 9
图 1-7 SuperGlue 算法示意图 10
图 1-8 HLoc 定位框架图 10
图 1-9 文章内容与结构安排 11
图 2-1 PnP 算法示意图 14
图 2-2 针孔相机模型 15
图 2-3 相机类型 17
图 2-4 回环检测示意图 19
图 2-5 稀疏建图(左)与稠密建图(右) 20
图 2-6 ORB-SLAM3 系统框架 21
图 2-7 四足机器人平台与实验环境 22
图 3-1 动态视觉 SLAM 算法框架 25
图 3-2 YOSO 与传统全景分割对比图 27
图 3-3 YOSO 分割结果 28
图 3-4 TUM 数据集分割结果 28
图 3-5 FastFlowNet 算法框架[65] 31
图 3-6 运动分析流程图 33
图 3-7 不同动态序列的轨迹误差对比 37
图 3-8 KITTI 数据集 00 与 09 序列轨迹误差效果图 40
图 3-9 EuRoC 数据集中部分序列的轨迹 42
图 3-10 TUM 数据集中的动态特征点过滤示意图 43
图 3-11 TUM 数据集中的动态掩码示意图 44
图 3-12 室外场景的动态 SLAM 实验 47
图 3-13 室内场景的动态 SLAM 实验 47
图 4-1 基于深层特征匹配的视觉重定位框架 49
图 4-2 Ada-Matcher 与 SuperGlue 的对比 51
图 4-3 Ada-Matcher 网络结构示意图 52
图 4-4 自适应权重共享算法 55
图 4-5 基于 PnP 的机器人位姿求解流程 57
图 4-6 SuperGlue 和 Ada-Matcher 在 HPatches 数据集上的可视化 58

图 4-7 HPatches 数据集上的图像匹配评估.....59
图 4-8 Ada-Matcher 和 SuperGlue 在 MegaDepth 数据集上的匹配结果.....61
图 4-9 定位轨迹示意图.....66
图 4-10 累计误差分布曲线.....67

第 1 章 绪论

1.1 课题研究背景与意义

随着人工智能技术的发展，机器人技术逐渐成为新兴产业中最具前景的领域之一。在各类机器人中，四足机器人以其卓越的性能，在隧道巡检、灾难救援以及物资运输等复杂环境下得到了广泛应用，成为当前研究的热点方向之一。四足机器人在执行具体任务时主要由感知系统和决策系统组成。感知系统负责与外界环境进行信息交互，决策系统负责运动规划与执行。感知系统通过快速整合和处理内部传感器数据和外部环境信息，实现对周围环境的全面感知与理解，提供关键的位置信息和环境感知。这些关键数据为决策系统的规划和控制提供了坚实基础，保证了四足机器人能够稳定可靠的完成各种复杂任务。

四足机器人能适应更多复杂的工作场景，这也意味着其在运动过程中需要面临更多的挑战，因此，如何提高机器人在复杂环境中的运动鲁棒性，成为当前的热门研究方向。在传感器的选择上，尽管激光雷达能够直接提供环境的三维几何信息，但高昂的成本且语义信息有限，使其应用受限。相比之下，视觉传感器成本更低，并能捕捉更丰富的环境信息，因此在机器人领域得到了更广泛的关注。然而视觉传感器易受环境变化的影响，鲁棒性不足。这使得提高视觉传感器在长期运行中的鲁棒性成为移动机器人实际应用的关键问题。具体而言，四足机器人运动需要利用视觉传感器信息完成两方面的任务：一是实现对环境的感知，获取动态物体、障碍物的信息，构建环境地图，为机器人导航提供实时的环境感知；二是实现精准定位，确保机器人能够在复杂的环境中保持稳定的位姿估计。在四足机器人长期运行中，环境会不可避免地发生变化，如日夜切换、结构变化等，或因机器人自身运动造成的遮挡、视角抖动等情况，视觉传感器对这些变化尤为敏感。这些变化和复杂场景对四足机器人运动提出了以下挑战：

1）视觉特征匹配器的鲁棒性不足。对跟踪定位和全局定位而言，传统视觉特征匹配器对环境变化不鲁棒导致错误匹配过多时，会干扰定位后端的优化，进而导致四足机器人的初始位姿估计有巨大偏差。而匹配过少则会导致定位失败。

2）姿态变化与复杂运动轨迹。四足机器人在运动过程中，虽然比其他类型的机器人有更高的稳定性，但由于其多自由度的运动特性，特别是在不平整或复杂地形上，其抬腿、转向等动作会导致视角频繁变化。这种动态视角的切换会影响视觉传感器的图像捕捉，特别是在视角抖动或遮挡情况下，导致特征点提取困难

或匹配不准确，进而影响定位精度。

3）动态场景下的环境感知与定位。在动态环境中，四足机器人需要具备高效的环境感知与精准的定位能力。传统视觉 SLAM 基于静态环境假设，面对动态场景中的移动物体的干扰，会导致感知信息的不准确，进而影响机器人估计位姿的准确性以及构建地图的一致性。因此，四足机器人需要一种能够适应动态环境的视觉感知与定位方法，以确保其在复杂场景中能够稳定、准确地执行任务。

伴随科学技术的发展，深度学习在各种视觉任务中展现了其强大的潜力。当前流行的卷积神经网络(Convolutional Neural Network, CNN)与 Transformer 模型凭借出色的学习能力，在目标检测，语义分割，SLAM 等领域都有了显著的进展，超越了相关领域传统方法。通过深层神经网络强大的学习能力，能够获取更丰富的视觉信息同时可以赋予机器人对场景内容理解和识别的能力。针对上述挑战，本研究旨在结合深度学习，探索四足机器人鲁棒视觉定位与感知为目标，以提高其在复杂环境中的定位精度和环境感知能力，实现稳定与鲁棒的实际效果，满足上层决策的需求。

1.2 国内外研究现状

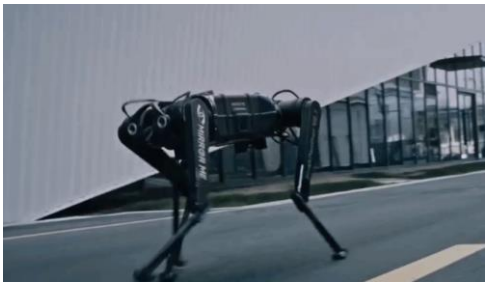
1.2.1 四足机器人研究现状

四足机器人的研究可以追溯到 19 世纪，俄罗斯工程师 Chebyshev 于 1870 年基于四杆机构的原理设计了首套四足行走装置。随着科技的发展，四足机器人逐步迈向实用化。20 世纪 40 年代，Hutchinson 研发出第一款能够独立控制四肢的四足机器人，开启了四足机器人控制技术的新篇章。1968 年，通用电气公司推出了 Walking Truck，这是世界上首个具备力反馈功能的四足机器人。1986 年，麻省理工学院(MIT)成功研制出了全球首台动态稳定的四足机器人，标志着四足机器人技术的一次重大飞跃。

进入 21 世纪，四足机器人得到了快速发展，多个知名科研机构和企业纷纷推出了一系列先进的四足机器人。波士顿动力公司(BD)在 2005 年推出了 BigDog，这款机器人在崎岖地形中展现出卓越的适应性，并得到了美国国防局的资助[1]。此后，BD 不断改进其四足机器人，推出了 LS3、LittleDog、Wildcat 等多款机器人，具有更高的稳定性、灵活性和速度。2016 年，BD 推出了 Spot 机器人，这款机器人在自主导航、感知和动作控制方面表现突出[2]。此外，苏黎世联邦理工学院(ETH)在 2015 年开发了 StarlETH 机器人，采用了高标准弹性执行器，模拟人类

肌肉的蓄能功能；同时，MIT 也在四足机器人领域持续创新，推出了 MIT Cheetah 系列机器人，并且在 2018 年实现了无感知爬楼梯的功能[3]。

国内对四足机器人领域的探索虽相对较晚，但近年来已取得了显著的进展，如图 1-1 所示。2002 年清华大学研制了 BIOBOT，2010 年山东大学在液压驱动四足机器人上取得进展。此外，哈尔滨工业大学的“豹”型机器人和上海交通大学的四足小象机器人等也展现优异的性能。在产品化方面，宇树科技的 Go 和云深处科技的“绝影”系列成为国内四足机器人的代表性产品。其中，绝影 X20 具备全天候作业和自主充电等先进功能，而 Go1 在速度上超越了同等规格的 Mini Cheetah。截至 2025 年，四足机器人研究持续进步。浙江大学推出全球最快的四足机器人“黑豹”，速度可达 10 米/秒，宇树科技的 B2W 展现了强悍的野外极限性能，市场上的表现也证明了四足机器人的应用潜力。



(a) 黑豹

(b) B2W

图 1-1 国内四足机器人系列

Fig.1-1 Domestic quadruped robot series

1.2.2 视觉感知相关算法研究现状

环境感知是四足机器人技术的关键，随着技术的不断进步，四足机器人在环境感知领域实现了重大突破，包含了深度学习[4]、场景分割[5]与目标检测与追踪[7]等核心技术的发展。这些技术突破显著的提升了四足机器人对环境的理解与适应能力，使其能够在复杂多变的环境中更加自主、高效地运行。视觉感知技术的不断发展，使其逐渐成为环境感知的重要分支，它通过相机获取环境的图像信息，为四足机器人提供了更为丰富、直观的环境认知，从而进一步增强了四足机器人的环境适应性和交互能力。

视觉 SLAM 作为四足机器人领域的关键技术之一，目的是在没有预先获取环境信息的情况下，利用传感器在移动过程中完成定位与环境感知。随着科学技术的发展，视觉 SLAM 也经历了革新。从最初的特征点法[9]与直接法[10]，到卡尔曼滤波(kalman filtering)方法[11]的广泛应用，这些传统方法在静态环境中有效地支持

了机器人的定位与建图。然而，随着环境复杂性的增加，视觉 SLAM 技术逐渐进入了一个新的发展阶段，借助深度学习的突破，尤其是基于神经网络的目标检测[8]、语义分割[4]等技术的引入，视觉 SLAM 能够更好地处理动态环境中的变化和不确定性。视觉 SLAM 的适用场景也从静态假设扩展到复杂且动态的环境。传统的 SLAM 方法往往依赖于固定的环境特征，难以应对如移动物体、光照变化或遮挡等动态因素。而现代的视觉 SLAM 系统[12-14]通过增强的鲁棒性和自适应能力，能够在更加复杂和动态的环境中保持稳定的定位与建图效果。这一转变不仅提升了四足机器人在现实世界中的实用性，也为其在各种复杂任务中的应用提供了强大的技术支持。

(1) 传统视觉 SLAM

SLAM 技术最早可追溯至 1986 年，Smith[15]在研究空间不确定性的描述与变换表示时，首次提出了概率 SLAM 概念，标志着 SLAM 问题进入公众视野。同年，在美国加州旧金山举行的 IEEE 机器人与自动化会议上，未知环境感知概念被首次提出，概率方法也被引入机器人和人工智能领域，这为 SLAM 技术的发展奠定了基础。经过多年的发展，以视觉为主的 SLAM 技术，成为机器人领域的重要研究方向。

Davison 于 2007 年提出了 MonoSLAM[16],这是首个基于扩展卡尔曼滤波(EKF)的实时单目视觉 SLAM 系统，它通过单目相机实现了实时的定位与建图功能，在此之前，受制于计算性能，大多数 SLAM 系统只能在采集完数据之后，再进行离线状态下的定位与建图工作，无法满足实时应用的迫切需求，但是其依赖稀疏特征点跟踪，路标数量有限，适应场景小，长时间运行时定位与建图性能会下降。同年，Klein 提出了 PTAM[9]这是首个实现跟踪和建图并行化的视觉 SLAM 算法，并引入了关键帧概念，仅对关键帧进行定位和建图，大幅减少了计算量。PTAM 还率先采用非线性优化替代滤波器作为后端，解决了数据增长过快的问题。然而，PTAM 主要适用于小场景，在大范围环境下全局估计能力不足，且易出现跟踪丢失。2015 年，Mur-Artual 基于 PTAM 提出了 ORB-SLAM[17]，包含跟踪、局部建图与回环检测三个并行线程，如图 1-2 所示，该系统使用 ORB 特征点进行匹配来完成位姿估计，并通过词袋模型(Bag of Words, BoW)进行闭环检测，同时在后端实现全局优化，从而显著提升了 SLAM 系统的精度和鲁棒性。然而，ORB-SLAM 在稀疏纹理区域或动态物体出现时容易发生跟踪丢失，导致较大位姿估计误差，且稀疏的点云信息限制了其对环境的深入理解。2017 年该团队推出了 ORB-SLAM2[18]，基于相同框架引入了双目和 RGB-D 相机模型，能够更有效地获取深

度信息，增强了算法的适应性与泛化能力，此外，ORB-SLAM2 还支持稠密地图的构建，进一步提升了环境感知能力，如图 1-3 所示。2021 年，ORB-SLAM3[19]被提出，进一步融合了 IMU 传感器和鱼眼相机模型实现了多传感器融合，减少了视觉传感器带来的误差，增强了算法在快速运动和低纹理环境中的鲁棒性。此外，ORB-SLAM3 引入了多地图管理系统，通过 ATLAS 实现多个子地图的存储，便于长时间在未知或大规模环境中进行探索。作为当代最为流行与完善的 SLAM 算法之一，ORB-SLAM 系列一直被大量研究者研究和改进。

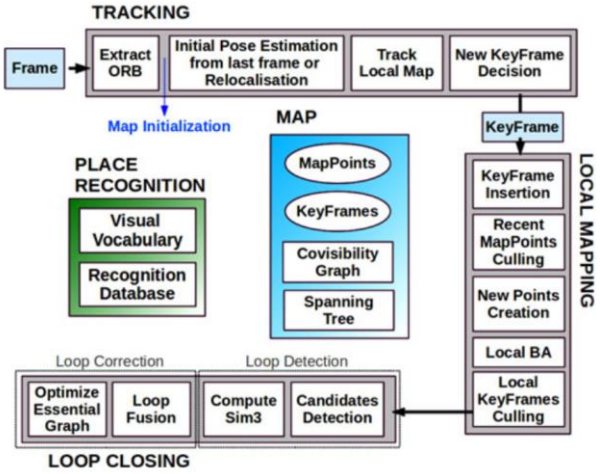


图 1-2 ORB-SLAM 系统框架图[18]

Fig.1-2 ORB-SLAM system framework diagram[18]

图 1-3 ORB-SLAM2 稠密建图

Fig.1-3 Dense mapping of ORB-SLAM2

与上述基于特征点的方法不同，直接法利用图像像素的灰度信息，通过计算两帧图像之间的光度误差来实现高精度和鲁棒性，尤其在纹理稀缺的环境中表现突出。2011 年 Newcombe 提出 DTAM[20]，这是第一个直接法视觉 SLAM，通过对每个像素进行深度测量并使用整幅图像进行对齐，从而生成稠密地图并估计相机

位姿。然而，该方法由于计算量巨大，限制了其在实时应用中的表现。2014 年，Engel 等人提出了 LSD-SLAM[10]，如图 1-4 所示，该算法同时利用关键帧位姿图和相应的半稠密深度图来重建三维环境，但是其对光照较为敏感。同年，Forster 等人提出了 SVO[21]半直接视觉里程计，避免了运动估计中特征提取和匹配的需求，直接作用于像素强度，提升了计算速度，但缺乏回环检测和后端优化，导致了较大的累计误差。2016 年，Engel 再次提出 DSO[22]算法，通过最小化光度误差来估计相机的位姿和场景的三维结构，可以有效利用任何图像像素，从而提高了模型在低纹理或模糊区域的鲁棒性，但是其不是一个完整的 SLAM 系统。

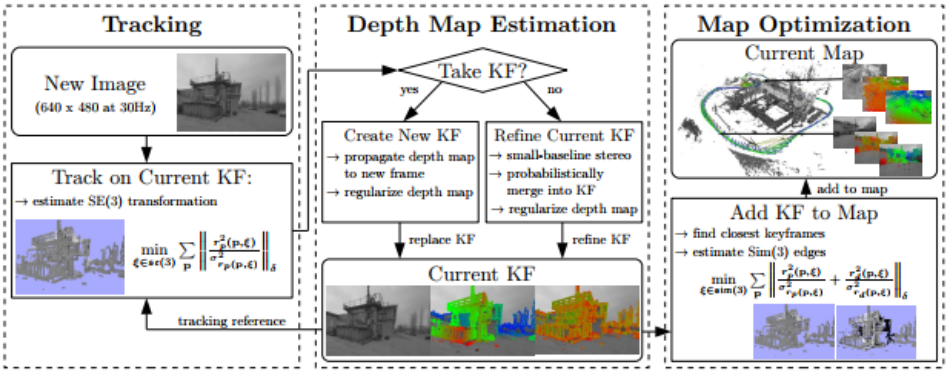


图 1-4 LSD-SLAM 系统框架图[10]

Fig.1-4 LSD-SLAM system framework diagram[10]

此外，2018 年香港科技大学发布了 VINS-Mono[23]，创新的整合了 IMU 和弹幕相机，采用紧耦合方案将 IMU 预积分与特征观测数据相融合，结合滑动窗口，利用两种传感器互补效应，提高了 SLAM 在复杂环境中的适应性。同年，宾尼法尼亚大学提出 MSCKF-VIO[24]，一种基于 EKF 的视觉惯性里程计，通过约束视觉和惯性数据之间的关系，简化了传统卡尔曼滤波器的复杂性，同时保留了其较强的状态估计能力。

(2) 深度学习视觉 SLAM

得益于深度学习技术的飞速发展，深度学习视觉 SLAM 正日益成为一个热门的研究领域。传统的视觉 SLAM 方法依赖于手工设计的特征提取与优化算法，而深度学习视觉 SLAM 则通过神经网络自动学习特征表示和环境建图，使得系统能够在复杂和动态环境中表现出更高的鲁棒性和精度。深度学习的引入不仅提升了对低纹理和动态场景等环境中的适应能力，还促进了端到端的优化方法的发展，使得视觉 SLAM 技术能够在更广泛的应用场景中实现高效、实时的定位与建图。

一方面，深度学习被广泛应用于 SLAM 中的各个子模块。2017 年，为解决单目 SLAM 中绝对尺度估计问题，Tateno 提出了 CNN-SLAM[25]单目视觉 SLAM，

该方法使用 CNN 特征来代替传统人工特征模块，采用了深度图预测与单目 SLAM 点云数据的概率融合策略。2020 年，Dusmanu 等[26]提出使用多视图优化的方式，改进局部特征的几何精度和一致性，从而提高特征匹配的准确性和鲁棒性。在 SLAM 前端中，一些基于深度学习的特征[27]用来替换手工设计的特征点[33][34]来增强特征点提取算法的鲁棒性和稳定性，以及 SLAM 后端中通过深度学习的方式重建和优化三维地图表示[35]。

另一方面，深度学习的引入促进了端到端 SLAM 的发展。例如，2017 年，DeepVO[38]提出了一个端到端单目视觉里程计(VO)算法。通过结合卷积神经网络（CNN）和循环神经网络（RNN），实现了单目视觉里程计的表示学习和时序建模。2018 年，DeepTAM[39]通过深度学习方法从单幅 RGB 图像中实时估计相机位姿并进行场景重建，能够自动学习特征并提高鲁棒性和精度。2020 年，DeepFactors[36]作为最完整的深度 SLAM 系统之一，基于 CodeSLAM[35]，采用姿态和深度变量的联合优化来处理闭环检测问题，具有短周期和长周期的闭环能力。2021 年，Droid-slam[40]通过一个可微分的密集束调整层进行迭代更新相机姿态和深度图，具有高精度，高鲁棒性以及强泛化性的优势，能够在复杂环境中进行高效的场景定位与地图构建，如图 1-5 所示。

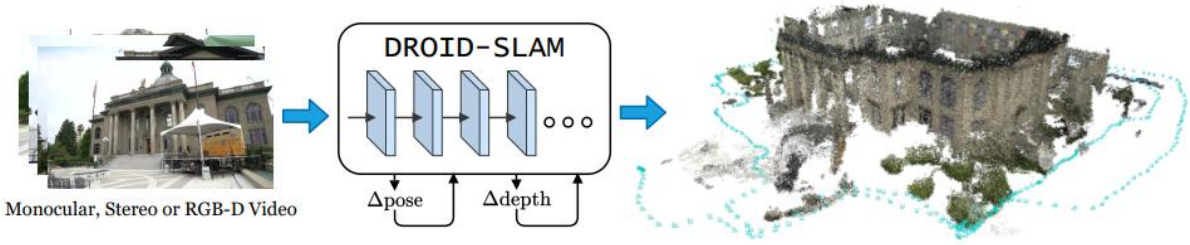


图 1-5 Droid-slam 地图构建示意图[40]

Fig.1-5 Schematic diagram of Droid-slam map construction[40]

此外，大多数视觉 SLAM 算法假设环境静态或物体运动缓慢，这限制了其在实际场景中的适用性。在动态环境中，移动物体导致传感器数据与地图不一致，遮挡环境造成特征点丢失和姿态估计困难，物体外观变化影响特征提取与匹配准确性，增加定位和地图更新的不确定性，运动误差累积进一步降低定位精度，这些因素共同限制了视觉 SLAM 系统在动态环境下的性能和鲁棒性。

2018 年，DynaSLAM[41]在 ORB-SLAM2 基础上利用语义分割和多视图几何的方法检测画面中存在的动态物体，并将他们从图像中剔除，以降低动态物体对相关算法的影响。同时结合背景修复，恢复被遮挡的静态背景。但该方法会剔除所有具有潜在运动特性的目标，会导致剩余的静止特征点过少而影响相机位姿估计

的准确性。DS-SLAM[42]基于 ORB-SLAM2，通过引入语义分割和密集语义地图构建线程，过滤场景中动态部分，采用语义分割网络与运动一致性检查方法结合，显著改进了动态环境中的鲁棒性。同时，利用单独线程构建密集语义八叉树地图，以支持复杂任务。在 Detect-SLAM[43]中，利用语义信息来消除 SLAM 流程中移动物体引起的负面影响。为了克服通信和检测引起的语义信息延迟，该方法提出了一种实时传播每个关键点移动概率的方法。此外它们还在 SLAM 流程中构建了一个对象地图，即由映射线程中所有检测到的静态对象组成的语义地图。该对象地图可以被视为包含有关对象类别和位置的知识的数据库。2019 年，Dynamic-SLAM[44]提出了一个基于特征的视觉 SLAM 里程计，通过相邻帧速度不变性的丢失检测补偿算法，提升了检测的召回率，并利用选择性跟踪算法处理动态特征点，减少了动态物体对位姿估计的影响。2022 年，DeFlowSLAM[45]采用了新的场景运动双流表示方法，它将光流分解为由相机运动引起的静态光流场和由场景中物体运动引起的另一个动态光流场。同时，提出了迭代动态更新模块和基于帧间共可视性的因子图优化，它可以鲁棒地应对具有挑战性的场景。在静态和不太动态的场景中表现出与最先进的 DROID-SLAM 相当的性能，而在高动态环境中明显优于 DROID-SLAM。2024 年，DVDS[12]开发了一种动态对象独立的机制，该机制可以过滤动态组件以进行更准确的光流估计，从而仅利用静态像素来检索相机位姿，显著提升了应对动态复杂环境的效果。

1.2.3 基于结构的视觉重定位方法研究现状

视觉重定位是通过给定的 RGB 图像估计相机坐标系到世界坐标系的六自由度(6-DoF)位姿，是机器人感知系统中的一个重要环节。它在自动驾驶、同步定位与地图构建等应用中发挥着关键作用。

基于结构的视觉重定位方法通过建立图像特征点的 2D 坐标与路标点的 3D 坐标之间的对应关系，并运用基于 RANSAC 的 PnP 算法来计算相机的位姿。2016 年，Sattler 等[46]提出了一种高效的索引方法和优先级搜索策略，用于大规模图像定位中的二值特征匹配，通过监督训练构建随机树，实现了快速的近似最近邻搜索，显著提高了定位速度，但因查询结果的不确定性造成一定精度损失。2017 年，Liu 等人[47]通过构建基于 3D 地图点共现关系的马尔可夫网络，并使用随机游走算法进行全局推理，提高相机定位的鲁棒性，然而针对大规模点云数据，此类方法的效率低下且会提取大量的 2D-3D 误匹配，从而极大地降低视觉定位精度。为解决这个问题，当前流行的基于结构的视觉定位方法首先通过图像检索算法[48]限

制了 2D 像素点与 3D 路标点之间的搜索范围，接着，这些方法运用先进的特征匹配算法(例如 SuperPoint[27]、SGMNet[49]、LoFTR[50]、MatchFormer[51])生成精确的 2D-2D 匹配关系。在此基础上，结合检索到的图像的位姿信息和深度图，进一步建立 2D-3D 匹配，从而实现高精度的视觉定位。

传统特征匹配方法主要依靠手工设计的算子(例如 SIFT[33]、ORB[34])与最近邻搜索算法相结合，以完成特征匹配。然而，在应对真实复杂环境时，例如低纹理区域和视角变化时，往往表现出较弱的鲁棒性，从而影响下游任务的表现。随着深度学习技术在各项任务中的优异表现，一些研究将该技术引入特征匹配领域。其中一部分研究着重关注于特征检测与描述，如 DISK[28]，SuperPoint[27]，R2D2[30]等，这些基于深度学习提取的视觉描述子在复杂环境下表现出显著优异的性能。作为最广泛使用的特征检测器，SuperPoint 提出了一种自监督的特征点检测和描述方法，利用两个解码器分别进行特征点定位和描述子生成，如图 1-6 所示。其被广泛应用与多个匹配框架，并在匹配任务中取得了显著提升。另一部分聚焦于更加强大的匹配算法，根据视觉描述子建立图像对之间正确的几何对应关系。尤其是近年来，Transformer[52]技术逐渐成为计算机视觉领域的热点，基于 Transformer 的特征匹配方法也展现出强大的鲁棒性。

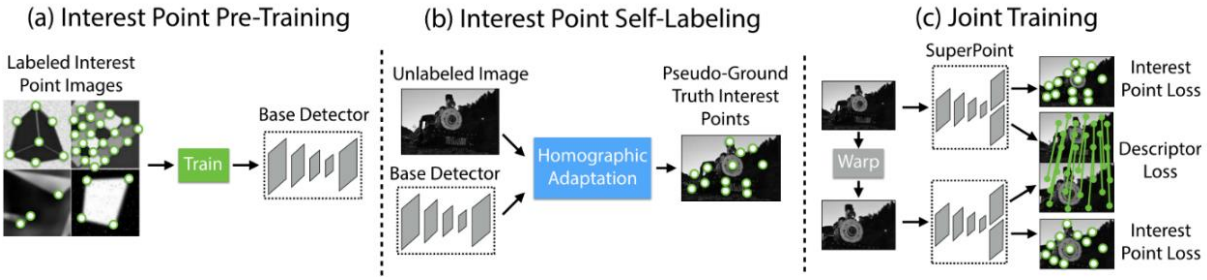


图 1-6 SuperPoint 示意图[27]

Fig.1-6 Schematic diagram of SuperPoint algorithm[27]

SuperGlue[53]作为首个将 Transformer 应用于特征匹配网络的工作，如图 1-7 所示。它将特征点视为图神经网络中的节点，利用自注意力和交叉注意力机制整合图像内外的全局信息，增强视觉描述符。该方法将特征匹配视为最优分配问题，通过最优传输算法获取软分配矩阵，最终提取满足最近邻准则且匹配置信度高的匹配结果。由于 Transformer 的计算复杂度较大，SuperGlue 在处理大量特征点时效率较低。为应对 SuperGlue 的局限性，SGMNet[54]利用图神经网络的稀疏性来减少冗余连接并学习紧凑的表示，从而降低计算复杂度。ClusterGNN[55]自适应地将关键点划分为不同的子图以最小化不必要的连接，并采用由粗到细的策略来减少图像内错误分类。LightGLue[56]重新审视了 SuperGlue 的设计决策，引入早退机制，

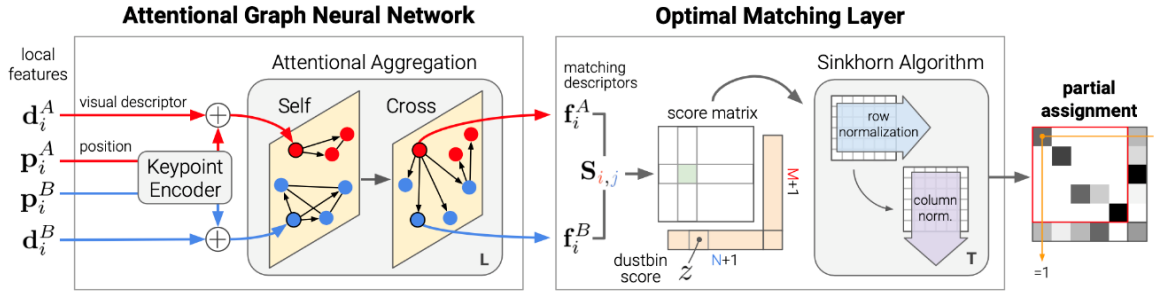


图 1-7 SuperGlue 算法示意图[53]

Fig.1-7 Schematic diagram of SuperGlue algorithm[53]

实现基于匹配图像对难度的自适应计算，从而更轻松地训练模型，同时确保更快的推理速度并保持高准确率。OETR[57]采用轻量化网络提取图像对之间的共视区域，有效限制了 Transformer 特征交互的范围，增强了描述符的效果，并减少了非共视区域的误匹配。

基于这些特征匹配方法的卓越性能，为实现高效且鲁棒的定位效果，一种由粗到细的视觉分层定位框架 HLoc[58]被提出，如图 1-8 所示。通过离线构建场景数据库并使用图像检索算法进行粗匹配，结合 PnP 算法计算查询图像的位姿，最终通过聚类获得精确的预测位姿。为提升效率，HLoc 引入了 HF-Net，集成 NetVLAD 和 SuperPoint，降低计算成本同时保证鲁棒性。在室内场景中，由于低纹理和可移动物体的存在，传统稀疏特征方法的定位精度受到限制，InLoc[59]通过稠密特征提取应对稀疏特征点的问题，利用 VGG16 提取高维特征并通过由粗到细的匹配建立准确的位姿预测。针对大规模场景存储问题，SceneSqueezer[60]提出了一种在保持定位精度的同时减少场景点数量和压缩特征维度的方法。通过这种方式，SceneSqueezer 能显著降低存储场景地图所需的内存占用，使得视觉定位算法能够更容易地部署在资源受限的设备上。

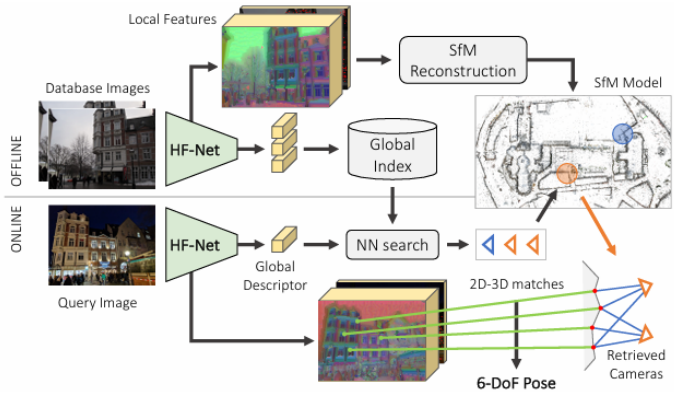


图 1-8 HLoc 定位框架图[58]

Fig.1-8 HLoc positioning framework diagram[58]

1.3 论文研究内容与结构安排

本课题是以四足机器人为平台，聚焦于鲁棒视觉感知与定位算法。文章以四足机器人在室内/室外场景中长期自主运动为背景，结合当前流行的深度学习方法，克服四足机器人在复杂场景下面对的各项挑战，提高四足机器人对环境的感知能力以及对恶劣场景的鲁棒性。如图 1-9 所示，文章具体结构内容如下：

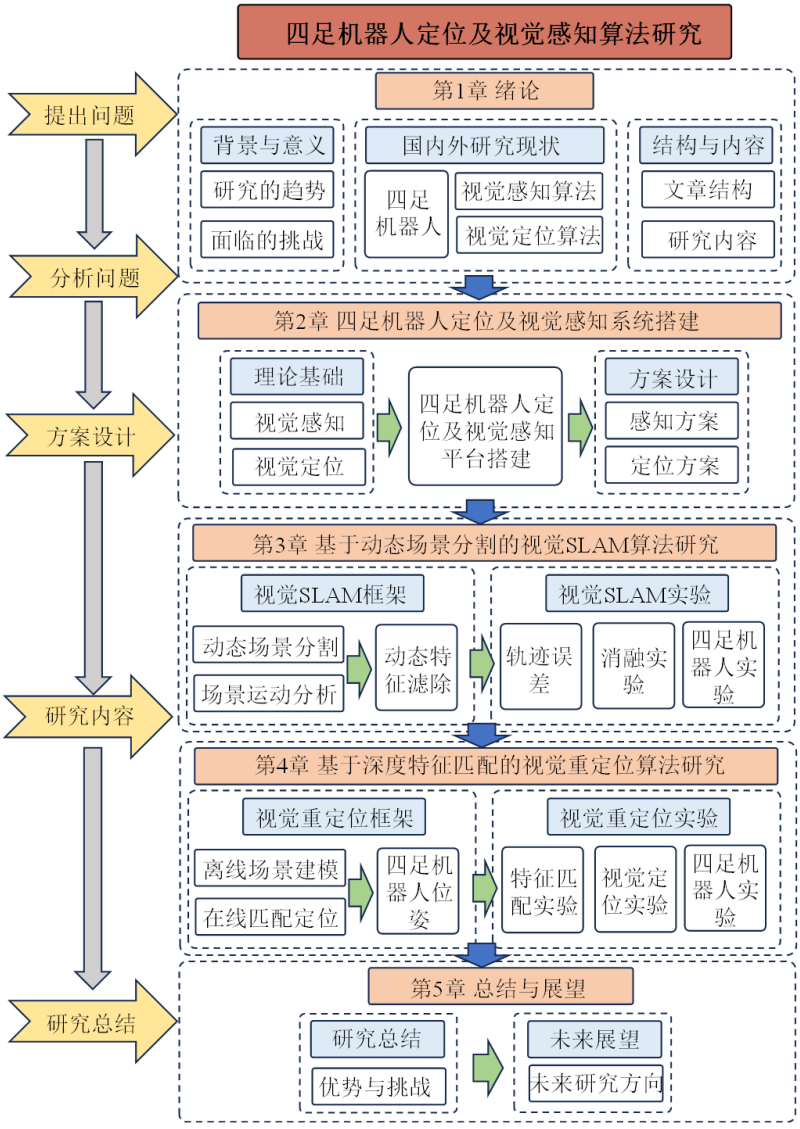


图 1-9 文章内容与结构安排

Fig.1-9 Article content and structure

第 1 章，首先介绍了四足机器人在复杂场景中面临的挑战，分析了研究背景与意义，总结了国内外在视觉定位、视觉 SLAM 以及四足机器人领域的研究现状，重点探讨了基于结构的视觉重定位方法与动态视觉 SLAM 技术，强调其在四足机器人运动过程中的重要性。最后，概述了本文的主要研究内容与结构安排。

第 2 章，简要介绍了视觉定位与感知的基础理论，接着详细阐述了基于四足机器人的定位与视觉感知方案，包括其总体流程和具体研究内容。

第 3 章，重点研究了基于场景分割的动态视觉 SLAM 算法。首先，分析了传统视觉 SLAM 面临的挑战，并提出了结合全景分割与运动分析的动态视觉 SLAM 方法，通过分割动态场景、光流、几何一致性验证及深度信息，增强了在动态场景中的鲁棒性。该方法在 TUM、EuRoc 等数据集上进行了验证，并进行了针对四足机器人真实环境的实验分析。

第 4 章，讨论了基于深度特征匹配的视觉定位算法。首先，分析了当前视觉重定位的挑战，提出了基于深层 Transformer 的特征匹配方法，并在 Hpatches，MegaDepth 等数据集中进行了多场景的视觉任务实验，如室内外视觉定位和单应性估计实验，最后在四足机器人真实环境中进行了定位实验分析。

第 5 章，总结了论文的主要研究内容与解决的问题，讨论了尚未解决的挑战，并展望了未来的研究方向。

第 2 章 四足机器人定位及视觉感知系统搭建

2.1 引言

要实现自主运动，四足机器人必须依赖高精度的定位与视觉感知系统，这对其在动态和非结构化环境中的导航与决策尤为关键。本章主要介绍四足机器人定位与视觉感知的相关理论，如视觉定位中的 RANSAC(Random Sample Consensus)与 PnP(Perspective-n-Point)算法，以及视觉感知中的 SLAM 系统框架、相机模型。基于这些理论，本章提出了一种适用于动态环境的视觉 SLAM 方案以及基于深度特征匹配的视觉重定位方法，为四足机器人在复杂环境下的高精度定位与感知提供理论支撑。

2.2 视觉定位相关理论基础

2.2.1 RANSAC 算法

随机抽样一致算法(RANSAC)[61]是一种鲁棒性估计方法，主要用于处理含有噪声和异常值的数据集，广泛应用于视觉定位和特征匹配等任务中。其核心思想通过随机抽样和迭代优化，并通过验证模型的一致性来筛选内点和剔除外点。具体而言，算法流程涵盖随机采样、内点筛选、迭代优化与模型重拟合等阶段：首先从数据集中抽取最小子集构建初始模型参数；随后基于重投影误差函数评估数据一致性，将满足阈值条件的点归为内点，最终选择内点数量最多的模型，并通过内点集合的全局优化提升参数精度。具体阈值条件公式如下：

$$\left\| \boldsymbol{u}_i - \pi(\mathbf{T}\boldsymbol{X}_i) \right\|^2 < \varepsilon \tag{2-1}$$

其中，$\boldsymbol{u}_i$为图像二维观测坐标，$\boldsymbol{X}_i$为三维空间点，$\pi(\cdot)$表示相机投影函数，ε为预设阈值，$\mathbf{T} \in SE(3)$。RANSAC 算法在视觉定位中常用于估计图像间的单应性矩阵或基础矩阵，具有较强的鲁棒性，能够有效应对数据中的噪声和异常值。在视觉定位中，单应性变换模型的估计可以通过以下公式表示：

$$s\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \mathrm{H}\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \tag{2-2}$$

其中(u, v)和(u', v')分别表示图像中两点对的坐标，H为3×3的单应性矩阵，s是尺度因子。通过最少四组对应点，便可对H进行估计。算法通过最小二乘或奇异值分解求解初始模型后，结合内点筛选机制抑制异常值干扰。迭代次数k的确定依

赖于概率模型：

$$k \geq \frac{\log(1-p)}{\log(1-(1-\eta)^n)} \tag{2-3}$$

其中，p 为至少一次成功拟合模型的概率，η 为外点比例，n 为单次采样所需点数。通过合理设定 k 值，RANSAC 可以在精度与效率之间取得平衡。

2.2.2 PnP 算法

PnP 算法[62]是一种广泛应用于计算机视觉中的位姿估计方法，主要用于根据已知的 3D 空间点 $P_i=(X_i,Y_i,Z_i)$ 与其对应的 2D 图像点 $p_i=(u_i,v_i)$ 来估计相机的外部参数，即相机的位姿(旋转和平移)，如图 2-1 所示。

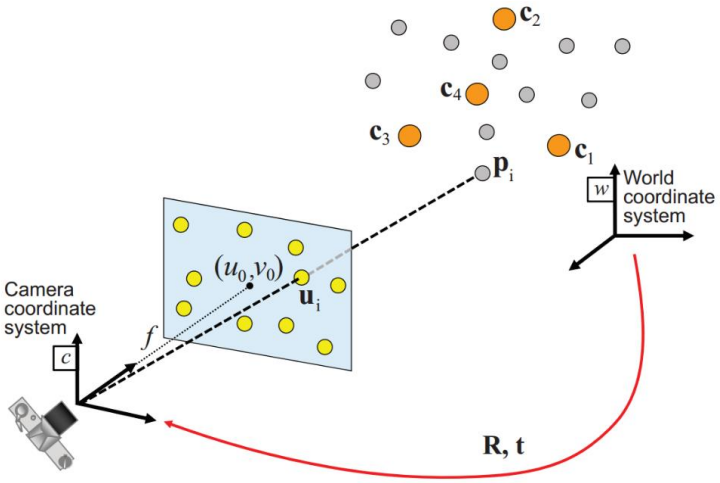


图 2-1 PnP 算法示意图

Fig.2-1 PnP algorithm diagram

该算法在视觉定位、物体识别和增强现实等应用中具有重要作用，尤其是在通过图像中的特征点进行空间定位时。其基本过程是利用最小化投影误差的原理，构建并优化投影模型，最终求得最优解。为提高算法的效率与稳定性，PnP 衍生出了多种变体，如高效 PnP(EpnP)通过简化计算过程提高实时性能，而无约束 PnP(UpnP)则用于复杂的非线性优化问题。在视觉定位任务中，PnP 算法因其高精度而被广泛应用，根据透视投影模型，点 P_i 在图像平面上的投影关系可表示为：

$$s\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = K[R\,|\,T]\begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \tag{2-4}$$

其中，K 为摄像机内参矩阵，R 是旋转矩阵，T 是平移向量。对多个点进行求解时，

PnP 问题可通过最小化投影误差的方式表示为：

$$\min_{R,T}\sum_{i=1}^{n}\left\|p_i-\hat{p}_i\right\|^2 \tag{2-5}$$

其中，$\hat{p}_i$为通过当前估计的位姿参数计算的投影点。由于 PnP 算法对外点敏感，通常需要结合 RANSAC 等鲁棒性方法，以提高在含有噪声或异常值的数据集中的估计精度。此外，在实际应用中，PnP 算法的精度与所选特征点的质量密切相关，尤其在低纹理和动态场景中，鲁棒的特征匹配方法变得尤为重要，能够有效地缓解这些问题，从而确保特征点的可靠性和准确性。

2.3 视觉感知相关理论基础

2.3.1 针孔相机模型

在视觉感知系统中，相机模型用于描述真实世界三维点到图像平面二维点投影关系，其准确性直接影响到视觉 SLAM 系统的定位和建图性能。通常，视觉 SLAM 系统采用针孔相机模型来简化相机的成像过程，并通过畸变矫正补偿真实相机镜头的几何畸变，从而提高位姿估计和地图构建的精度，如图 2-2 所示。

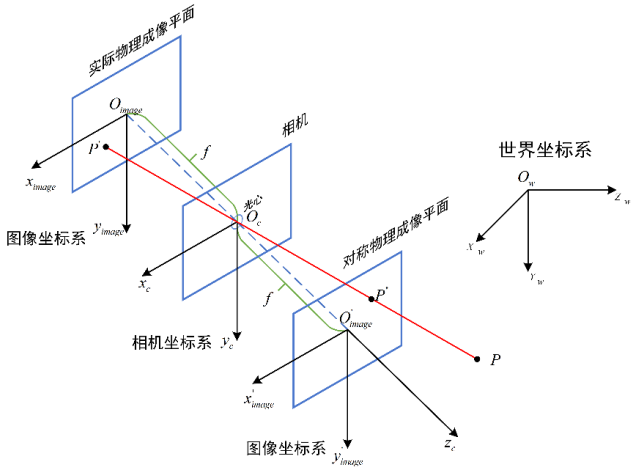


图 2-2 针孔相机模型

Fig.2-2 Pinhole camera model

针孔相机模型将相机抽象为一个仅包含一个极小孔径的理想装置，光线通过小孔将三维空间中的物体直射到一个假想的图像平面上，从而形成物体的倒立、缩小的真实投影。在该模型中，所有平行于光轴的光线在图像平面上的投影汇聚于一个点，该点被称为焦点，而针孔到图像平面的距离被定义为焦距。焦距的大小决定了成像的放大倍数以及相机的视场角范围。针孔相机模型为计算机视觉领域提供了重要的理论基础，使得三维点可以通过几何投影映射至二维图像。利用

这种映射关系，不仅能够推算三维场景的几何信息，还能完成相机位姿估计、三维重建等核心视觉任务。

在针孔相机模型中，三维点首先通过相机的光心投影到对称物理成像平面，然后通过几何反转映射到图像平面中。设三维点 P 的相机坐标系下坐标为 $P(X_c,Y_c,Z_c)$，在图像平面上的投影坐标为 (x_c,y_c)，其映射关系可以表示为：

$$x_c=\frac{f\cdot X_c}{Z_c},y_c=\frac{f\cdot Y_c}{Z_c} \tag{2-6}$$

其中，Z_c 是点 P 到相机光心的深度，f 为焦距。为了进一步映射到像素坐标系，需要考虑相机的内参。内参矩阵 $\boldsymbol{K}$ 用于描述相机的内部参数，包括像素焦距 f_x,f_y 和主点偏移量 c_x,c_y。其矩阵形式为：

$$\boldsymbol{K}=\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2-7}$$

结合相机坐标系中的点 x_c,y_c,z_c，图像像素坐标 (u,v) 可通过以下关系计算：

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}=\boldsymbol{K}\begin{bmatrix} x_c \\ y_c \\ Z_c \end{bmatrix} \tag{2-8}$$

其中 s 是尺度因子，用以归一化坐标。此外，针孔相机模型还支持从世界坐标系到图像平面的完整投影过程。三维点 $P(X_W,Y_W,Z_W)$ 在世界坐标系中的位置首先通过外参矩阵 $\boldsymbol{R}$ 和 $\boldsymbol{T}$ 转换到相机坐标系中：

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}=\boldsymbol{R}\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix}+\boldsymbol{T} \tag{2-9}$$

其中，$\boldsymbol{R}$ 是旋转矩阵，$\boldsymbol{T}$ 是平移向量。将外参矩阵与内参矩阵结合，可得到从世界坐标系到图像平面的完整投影关系：

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix}=\boldsymbol{K}[\boldsymbol{R}\ \ \boldsymbol{T}]\begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \tag{2-10}$$

上述公式完整描述了三维点从世界坐标系经过相机坐标系，最终映射到图像平面的过程。

2.3.2 视觉 SLAM 系统框架

2.3.2.1 传感器数据读取

在视觉 SLAM 中主要用于相机数据的读取和预处理。视觉 SLAM 系统的输入通常来自于单目、双目、RGB-D 或事件相机等视觉传感器，如图 2-3 所示。单目相机通过单个摄像头捕获图像，并利用连续帧的处理来估计相机的运动，但由于仅有一个摄像头，它无法直接获取深度信息，必须依赖结构光或运动来进行估计。双目相机利用一对相机从不同角度捕获图像，运用立体视觉技术直接测算出场景的深度信息。这种立体视觉方法通过分析图像间的视差，能够精确地测量深度，优化特征匹配，从而提升地图构建的精确度和可靠性。RGB-D 传感器通过融合彩色图像与深度数据，能够同时提供 RGB 图像和精确的深度图，从而获取更详尽的环境信息。事件相机采用与传统相机不同的工作原理，专注于检测场景中每个像素的亮度变化，一旦某个像素变化超过设定的阈值，便会生成一个包含事件详情、位置和变化方向(亮度增加或减少)的数据点，因此，事件相机产生的输出不是连续的图像序列，而是一个由离散事件组成的数据流。

a.单目相机　　b.双目相机　　c.RGB-D 相机　　d.事件相机

图 2-3 相机类型

Fig.2-3 Camera type

2.3.2.2 前端视觉里程计

视觉 SLAM 的前端通常被称为视觉里程计(Visual Odometry, VO)，其主要任务是基于相邻帧图像信息，初步估计摄像机的位姿变换与运动轨迹。目前主流方法可分为基于特征的方法与直接方法两大类。

基于特征的方法首先提取图像中的显著特征点，并通过特征描述与匹配估计相机的运动变换。这类方法的典型流程包括特征提取、特征匹配、外点剔除和位姿估计四个步骤。常用的特征提取算法包括 ORB、SIFT 等，其中 ORB 具有较高的计算效率，适合实时场景。特征匹配后，通常使用 RANSAC 算法剔除误匹配点，以提高匹配结果的鲁棒性。在位姿估计环节，利用 PnP 问题或最小二乘法，结合匹配点间的几何关系计算相机的旋转和平移矩阵。基于特征的方法对光照变化和

视角变化具有较好的适应性，但在弱纹理或特征点稀疏的场景中性能可能下降，且特征提取与匹配过程计算量较大。

直接方法则不依赖于特征点提取，而是直接利用图像的像素强度信息，通过最小化光度误差来估算相机的运动轨迹。其核心思想是：假设相邻两帧图像中的像素亮度在理想情况下保持一致，通过相机运动模型进行光度误差优化，估计相机位姿。光度误差的表达形式如下：

$$\mathrm{E}=\sum_{i}[I_t(u_i)-I_{t-1}(w(R,T,P_i))]^2 \tag{2-11}$$

其中，I_t和I_{t-1}分别为当前帧与上一帧的像素强度值，$w(R,T,P_i)$表示通过位姿变换(R,T)预测的像素点位置，E为光度误差。

2.3.2.3 后端非线性优化

后端模块是视觉 SLAM 系统的重要组成部分，其主要任务是对前端视觉里程计估算的相机位姿进行全局优化，从而获得一致的运动轨迹和更高精度的环境地图。后端算法主要分为基于滤波和基于优化的两类。

基于滤波的方法通常以扩展卡尔曼滤波(EKF)为代表，假设系统满足马尔可夫性，即当前状态仅与上一时刻状态相关，与之前的历史状态无关。在滤波方法中，通过递归地估计当前时刻的状态，再利用该状态预测下一时刻的位置和姿态。其核心思想是将状态估计与观测数据结合，通过贝叶斯推断实现状态更新。具体而言，EKF 通过线性化系统的状态转移方程和观测方程，以最小化估计误差的方式实现轨迹优化。然而，滤波方法在状态维度较高或观测数据噪声复杂时，容易引入误差累积，影响全局一致性。

相比之下，基于优化的方法不再局限于相邻时刻之间的状态关系，而是同时考虑所有历史状态与观测之间的约束，构建全局优化问题，以求解最优的相机轨迹和环境地图。优化框架通常基于图模型表示，将相机的位姿和地图点表示为图的顶点，将观测约束表示为图的边。其目标是通过非线性最小化重投影误差来实现全局一致性，优化问题的目标函数通常形式为：

$$E=\sum_{i,j}\left\|z_{i,j}-h(x_i,l_j)\right\|^2 \tag{2-12}$$

其中，$z_{i,j}$表示第j个特征点在第i帧中的观测值，$h(x_i,l_j)$表示相机模型预测的特征点投影值，x_i为相机位姿，l_j为三维地图点坐标。通过迭代优化，后端模块能够有效修正前端估算中的累积误差，保证轨迹和地图的全局一致性。

2.3.2.4 回环检测

在视觉 SLAM 中，随着时间的推移，机器人对自身位置和姿态的估计可能因多种因素逐渐累积误差。这些误差通常来源于特征匹配不准确、传感器噪声或非线性优化不足等问题。这些误差不断累积，可能导致显著的位置偏移，从而影响机器人的运行。因此，回环检测起到了至关重要的作用，其核心任务是判断机器人是否返回到了先前访问过的地点。回环检测通过比较当前帧与历史图像帧的相似性来实现，一旦成功检测到回环现象，系统将利用该信息对累计的位姿偏差进行全局校正，从而确保轨迹的一致性和地图的连贯性，如图 2-4 所示。

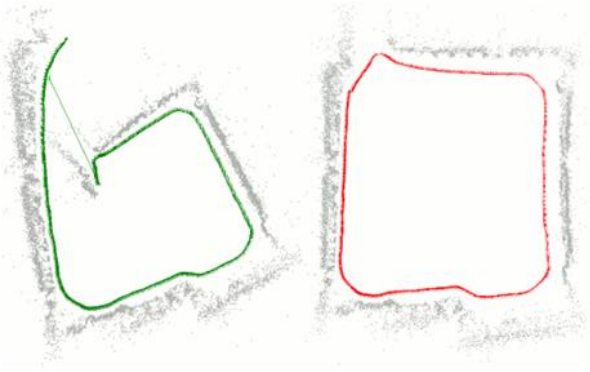


图 2-4 回环检测示意图

Fig.2-4 Loop detection diagram

回环检测的实现方法包括基于特征点匹配的传统方法、词袋模型以及基于深度学习的现代技术。其中，基于特征点匹配的方法通过提取图像中的局部特征点(如 ORB、SIFT)，并利用几何约束计算匹配关系，在纹理丰富的场景中具有较高的鲁棒性和效率。然而，这类方法在纹理稀疏或光照变化剧烈的环境中表现可能受限。词袋模型方法通过将图像描述为由视觉词汇构成的特征向量，能够快速评估图像之间的相似性，适用于较大规模的图像库匹配，但其准确性取决于词汇表的构建质量。基于深度学习的方法利用卷积神经网络提取图像的高层次特征，可以更好地处理纹理稀疏或复杂场景中的图像匹配问题。例如，利用预训练的视觉模型(如 ResNet)生成全局特征嵌入，再通过相似性度量判断是否存在回环现象。这些方法虽然能够显著提高检测准确率，但通常需要大量的计算资源支持，并依赖于高质量的训练数据集。

2.3.2.5 地图构建

地图构建是视觉 SLAM 系统的核心任务之一，其目的是生成准确且具有高可用性的环境地图，为机器人导航与任务执行提供可靠依据。根据地图的精细程度

和表示方式，通常分为稀疏建图和稠密建图两种，如图 2-5 所示。

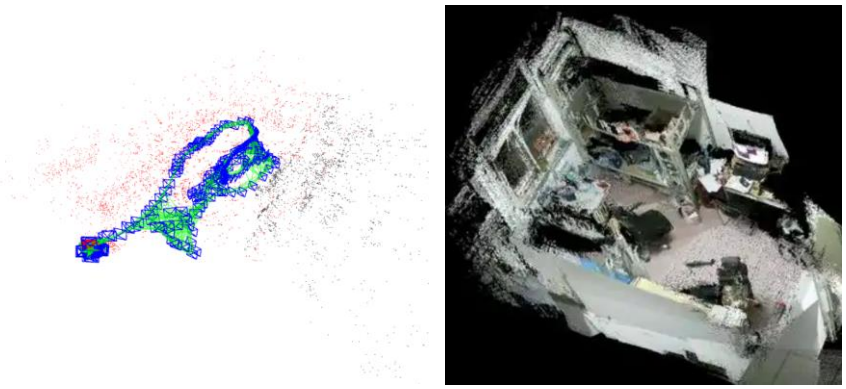


图 2-5 稀疏建图(左)与稠密建图(右)

Fig.2-5 Sparse mapping (left) and dense mapping (right)

稀疏建图主要利用前端视觉里程计提取的关键特征点，通过这些点的三维坐标构建一个稀疏的三维点云地图。这种方式计算效率高，占用存储资源较少，适用于实时性要求较高的应用场景。稀疏地图中的点主要分布在图像中纹理明显的区域，尽管点的密度较低，但其包含的信息足以满足机器人路径规划和定位的基本需求。例如，在 ORB-SLAM 中，稀疏建图通过特征点匹配、三角测量等步骤获取三维点云，结合后端优化确保地图的全局一致性和精度。

稠密建图通过计算每个像素点的深度信息，生成更为细致的三维环境表示。这种方法能够在复杂场景中提供更为直观和精确的环境描述，但其计算和存储成本较高。稠密建图通常依赖于多帧图像的深度估计，通过视差法或光度一致性优化生成高分辨率的深度图。在实际应用中，系统通过计算连续帧之间的相对运动来推测场景的几何结构，并通过非线性优化来提高深度估计的精度。常见的稠密建图方法包括利用 RGB-D 相机进行的实时三维重建技术(如 ElasticFusion[64])，以及基于单目或双目相机的稠密 SLAM 方法(如 DTAM)。

2.3.3 ORB-SLAM3 算法框架

ORB-SLAM3 是一种视觉与视觉-惯性同步定位与建图算法，广泛适用于多种传感器配置及复杂环境下的导航与地图构建任务。该系统不仅支持单目、双目、RGB-D 相机，还能融合惯性测量单元数据，显著提升了定位与建图的精度和稳健性，尤其在快速运动、弱纹理及大规模场景下表现出色。ORB-SLAM3 主要由跟踪、局部建图、回环检测与地图合并等核心模块构成，这些模块协同工作，实现高精度的实时定位与地图构建。系统框架如图 2-6 所示。

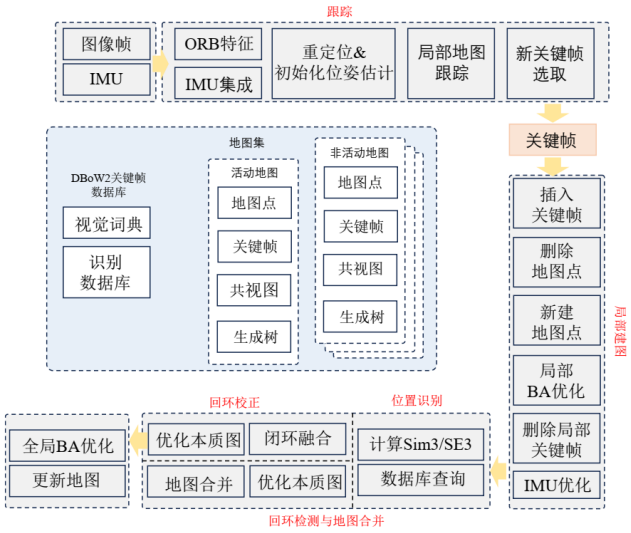


图 2-6 ORB-SLAM3 系统框架

Fig.2-6 ORB-SLAM3 system framework

跟踪线程作为系统的前端，主要负责实时处理传感器输入的图像数据，估计相机位姿，并提取关键帧以支持地图更新。具体来说，跟踪模块通过 ORB 特征的提取与匹配，对相邻帧之间的相机运动进行初步估计。同时，通过 PnP 算法和捆绑调整优化(Bundle Adjustment, BA)当前帧的位姿估计，确保系统在复杂场景中的实时性和稳定性。在当前帧与已构建地图的匹配程度较低时，跟踪模块会将当前帧选定为关键帧并存储，为后续的建图和全局优化提供数据支持。

局部建图线程负责对局部地图的关键帧和三维点云进行优化，逐步提高地图的几何精度。它通过滑动窗口策略，将最近的若干关键帧及其对应的地图点纳入局部优化窗口，利用捆绑调整对相机位姿和地图点的三维坐标进行联合优化。此外，该模块还会剔除冗余的低质量地图点，并对新的高质量特征点进行三维重建，保证局部地图的稀疏性和准确性。

闭环检测与地图合并线程是确保系统全局一致性的关键环节。闭环检测利用词袋模型与几何验证策略识别场景回环，解决长期运行的累计漂移问题。当检测到当前关键帧与历史地图存在关联时，模块计算相对位姿变换并执行局部窗口优化，通过位姿图或全局 BA 修正轨迹与地图的一致性。地图合并模块进一步扩展系统的多场景适用性，支持多会话 SLAM 任务。该模块通过跨地图的公共区域对齐，将子地图融合为全局一致模型，并利用焊接窗口优化与位姿图传播修正，显著提升大规模环境下的建图效率与精度。

多地图管理系统(Atlas)将不同会话或场景的子地图以非连续形式存储，当跟踪丢失时，系统自动创建新地图并持续探索，待重新访问已知区域时，通过地图

合并实现无缝切换。该机制不仅增强了对动态遮挡的适应性，还支持增量式多设备协作建图，为长期定位任务提供可靠保障。

2.4 定位及视觉感知平台搭建

2.4.1 四足机器人平台搭建

在本研究中使用了宇树科技的 B1 型号四足机器人平台，具体如图 2-7 所示。该机器人本体具有 12 个自由度(单腿 3 个自由度)，使其能够灵活应对各种地形，展现出卓越的环境适应能力。此外，B1 平台支持二次开发，用户可以根据具体需求对其进行功能拓展和算法定制，适合开展多领域的科研探索和应用研究。在视觉传感器方面，平台采用了 Intel RealSense D455 深度相机，该相机以其高精度和宽视角著称。D455 配备了 RGB 和深度传感器，能够同时捕获高分辨率彩色图像和精准的深度信息，极大地提升了机器人的环境感知能力。为满足算法调试的需求，本研究在将相应算法部署到讯圣 H110 工控机，该工控机搭载了 Intel i7-7700 处理器，32G 内存，并配备 Ubuntu20.04 操作系统，便于开发和部署机器人定位与视觉感知算法。

图 2-7 四足机器人平台与实验环境

Fig.2-7 Quadruped robot platform and experimental environment

2.4.2 视觉感知方案设计

在复杂动态环境中执行任务时，四足机器人面临诸多视觉挑战，这主要源于其多自由度运动和环境本身的动态变化。四足机器人在不平整地形或复杂场景中

运动时，其抬腿、转向、避障等动作会导致视角频繁变化，使得视觉传感器捕捉到的图像信息发生显著变化，进而影响视觉 SLAM 系统的位姿估计精度。此外，移动物体不仅会干扰视觉传感器的图像捕捉，还可能遮挡关键特征区域，导致特征点失效或误匹配，而由于机器人运动引起的场景变化(如物体的出现与消失)，使得传统视觉 SLAM 难以维持系统的鲁棒性。

为应对这些挑战，本研究提出了一种基于动态场景分割的视觉 SLAM 方案。方案首先利用全景分割技术对输入图像进行动态区域和静态背景的精确区分，并生成动态掩码标记动态物体。通过短期数据关联和目标分类，优化动态掩码的准确性和鲁棒性，确保动态物体特征点被有效剔除，而静态背景特征点得以保留。在此基础上，结合光流估计和几何一致性验证，对特征点运动矢量进行分析，并结合基础矩阵的几何约束，区分特征点的运动来源，优化动态掩码的分类结果。此外，引入深度辅助验证机制，利用深度信息进一步区分动态与静态特征点，提升特征点分类的准确性。最后，采用动态特征点剔除和基于置信度的加权策略，降低误匹配对定位和建图的影响。优化后的静态特征点被传递到后端优化模块以实现高精度位姿估计和地图构建，而动态特征点则反馈至局部建图和回环检测线程，以改善全局地图质量。该方案可以有效应对四足机器人自身复杂运动特性及动态环境中的各种干扰，展现出显著的鲁棒性和适应性。

2.4.3 视觉重定位方案设计

在复杂环境中，四足机器人在实现精准视觉重定位时面临诸多挑战。例如，在复杂地形(如台阶，沙地)行走时，导致的视角抖动，直接影响特征点的提取与匹配精度；移动物体(如行人，车辆)的干扰可能遮挡特征区域，导致部分特征点失效或误匹配；低纹理环境(如光滑墙面或地板)中，图像特征点稀疏，进一步增加了特征匹配和位姿估计的难度。为克服上述问题，本研究设计了一套基于深度特征匹配的视觉定位技术方案，采用离线建图与在线定位相结合的策略，充分发挥场景模型与实时感知的优势，解决四足机器人在动态复杂环境中的定位难题。

在离线阶段，利用 Intel RealSense D455 相机采集高分辨率的 RGB 图像，构建图像检索库；通过深度学习提取检索图像的全局描述子，构建全局描述子库；借助视觉 SLAM 系统获取每张图像的 6-DoF 位姿，构建稀疏场景地图，并利用 MVSNet 预测稠密深度图，生成场景点云，构建场景位姿数据库。在线定位阶段，实时采集的查询图像通过图像检索技术，选取最相似的检索图，然后利用基于深层 Transformer 网络的特征匹配方法与检索图像进行匹配，建立查询图像与数据

库图像间的二维像素点匹配关系，进一步结合三维路标点信息构建 2D-3D 对应关系。基于这些关系，最后采用 RANSAC 优化的 PnP 算法计算相机的 6-DoF 位姿，确保定位精度。该方案在动态遮挡和视角变化等复杂条件下，依然能够准确完成匹配并估算机器人位置。

2.5 本章小结

本章介绍了四足机器人定位与视觉感知的理论基础，包括视觉定位中的 RANSAC 和 PnP 算法，以及视觉感知中的针孔相机模型、视觉 SLAM 框架和 ORB-SLAM3 算法。其次，基于宇树 B1 机器人平台与 RealSense D455 相机搭建了实验平台。最后，提出了动态场景分割的视觉 SLAM 方案和基于特征匹配的重定位方法，为动态环境下的鲁棒定位提供了理论支撑与硬件实现基础。

第 3 章 基于动态场景分割的视觉 SLAM 算法研究

3.1 引言

视觉 SLAM 是四足机器人实现自主导航和环境感知的核心技术之一。然而，传统的视觉 SLAM 方法是在基于静态假设前提下运行的，其鲁棒性和适应性表现出明显不足，动态物体的干扰影响了地图的一致性和位姿估计的精度。随着深度学习技术的发展，基于检测或分割的视觉 SLAM 方法取得了显著进展，能有效减少动态物体的干扰，提升视觉 SLAM 系统在动态环境下的鲁棒性。然而，现有方法仍然存在一些缺陷。例如，部分方法采用“一刀切”策略，将检测到的动态区域全部剔除。这可能会误剔除动态物体遮挡下的静态特征，或忽略准静态特征，导致性能下降。此外，在动态遮挡或纹理稀疏的场景中，现有方法难以准确区分由相机运动引起的表观变化与物体自身的真实运动，导致特征点被错误剔除或保留，进而影响视觉 SLAM 效果。

因此，本研究从物体分割与运动来源分析的角度出发，提出一种结合全景分割和几何信息的动态视觉 SLAM，旨在提升四足机器人在复杂动态场景下的性能。

3.2 动态视觉 SLAM 算法框架描述

基于动态场景分割的视觉 SLAM 算法 DSS-SLAM，整体流程如图 3-1 所示。

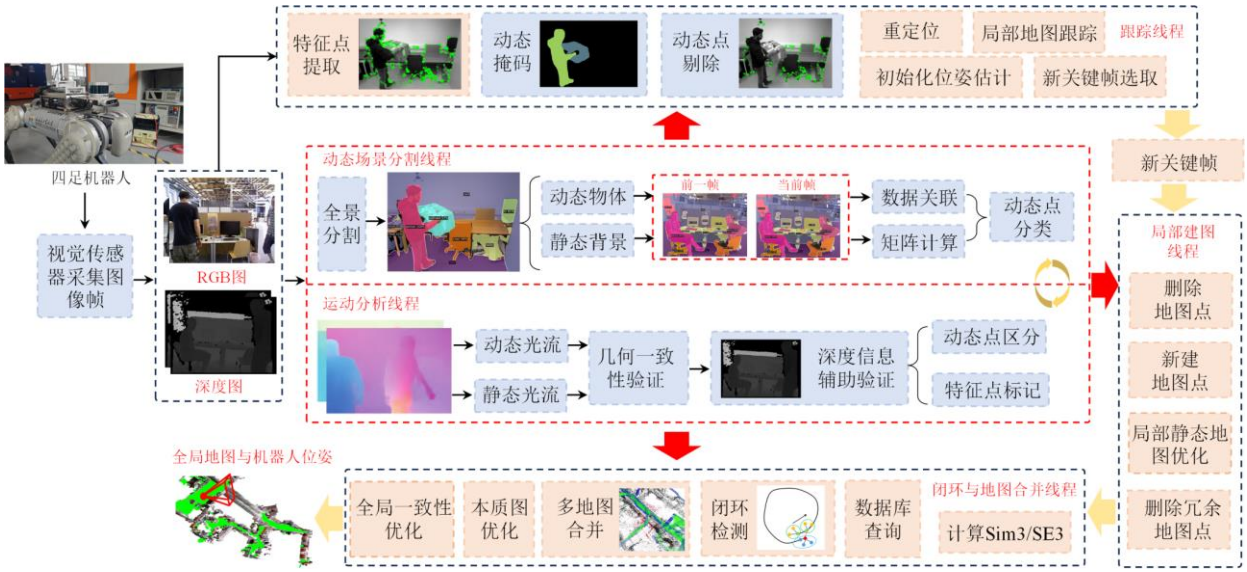


图 3-1 动态视觉 SLAM 算法框架

Fig.3-1 Dynamic visual SLAM algorithm framework

依托于 ORB-SLAM3 框架，结合全景分割、动态特征滤除以及几何运动分析技术，并作为单独的线程与 ORB-SLAM3 中的跟踪、局部建图以及回环与地图合

并线程形成了紧密的交互关系，可以有效解耦模块，提高并行能力，保证整体算法的实时性和鲁棒性。

动态场景分割线程的核心任务是利用全景分割技术，对输入的图像进行动态区域和静态区域的精确区分。通过生成动态掩码，该线程能够有效标记动态物体区域、静态背景区域。动态掩码直接传递给跟踪线程和运动分析线程，在前端跟踪阶段用于剔除动态区域内的特征点,减少动态物体对初始相机位姿估计的干扰，从而提升特征提取和匹配的精度。同时，掩码信息还为运动分析提供了动态区域的基础标记，便于后续的动态特征点分类与验证。此外，结合多帧序列信息优化分割结果，有效提升动态场景的处理能力和鲁棒性。

运动分析线程负责深入分析动态场景分割结果的运动来源。基于动态掩码的初始标记，结合前后帧的光流计算与几何一致性验证，该线程能够区分动态区域的运动来源是由相机运动还是物体自身的真实运动引起。进一步地，借助深度信息辅助验证，对动态特征点进行精确分类和标记，为后续的特征过滤和地图优化提供可靠的支持。标记结果传递给前端跟踪线程进一步过滤动态特征点，同时也反馈给局部建图线程，确保优化后的局部地图中仅包含高质量的静态点。此外，该线程的结果还为回环检测提供了动态区域的过滤依据，避免动态物体对回环匹配过程的干扰。

在此基础上，结合 ORB-SLAM3 多地图管理系统，动态场景分割线程和运动分析线程的结果能够在全局地图中进一步消除动态干扰，适应场景长期动态变化和大规模环境中的全局一致性需求。

3.2.1 动态场景分割

动态场景分割线程是 DSS-SLAM 系统中的关键模块，主要是通过全景分割技术对输入图像中的动态区域和静态背景进行精准分类，为运动分析、特征提取和地图构建提供高质量的基础数据。本研究在此线程中结合了全景分割技术与短期数据关联方法，通过对动态物体、静态背景以及未知区域的精确标记，生成高质量的动态掩码，并进一步优化掩码的准确性与鲁棒性。

3.2.1.1 全景分割与掩码生成

全景分割的核心目标是将图像中的像素划分为“对象”和“背景”两大类别。其中，“对象”是指具有明确边界且可计数的目标，例如人、动物、车辆等动态物体，这些目标通常具有潜在的移动性；而“背景”则是指图像中不可计数的、形状模糊的区域，例如天空、地板、墙壁等，这些区域通常是静止的。全景分割融

合了实例分割和语义分割的优势，不仅在精度上表现优异，还在效率方面显著优于传统的基于框的分割模型，能够更全面地捕捉图像的全局信息。本研究在动态场景分割线程中采用了 YOSO 模型[63]来实现全景分割，如图 3-2 图所示。

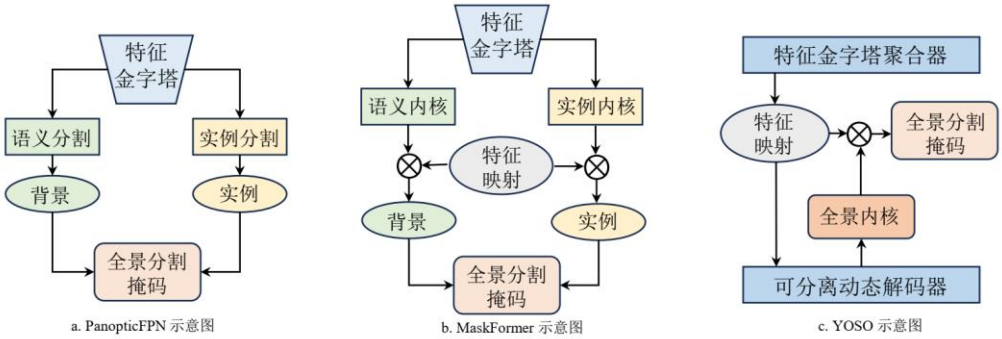


图 3-2 YOSO 与传统全景分割对比图

Fig.3-2 Comparison between YOSO and traditional panoptic segmentation

与传统的基于特征金字塔网络和多分支结构的全景分割方法不同，YOSO 通过在全景内核和图像特征图之间执行动态卷积，直接预测掩码，实现了实例分割和语义分割的统一处理。具体来说，YOSO 利用特征金字塔聚合器提取多尺度特征，这一模块能够有效建模像素之间的全局关系，为分割任务提供了更加精准的上下文信息。此外，YOSO 还通过卷积优先的方式重新设计插值模块，并对其进行参数化优化，从而显著提升了模型的处理速度。与此同时，模型引入了一个用于生成全景核的可分离动态解码器，该解码器利用可分离动态卷积执行多头交叉注意力操作，不仅提升了效率，还显著增强了分割结果的准确性。基于这些设计，YOSO 在单一网络中同时完成了实例分割和语义分割任务，无需传统方法中独立分支的复杂后处理步骤，能够直接生成动态区域和静态区域的动态掩码。这种高效的全景分割流程为动态视觉 SLAM 系统提供了可靠的分割支持，极大地提高了系统的实时性和精度。

在本研究中，YOSO 模型在 COCO 数据集上进行训练，该数据集涵盖了多达 80 个“对象”类别标签和 91 个“背景”类别标签。这种全面的标签范围使得 YOSO 模型能够适应多样化的场景，并生成高度精确的分割结果。在动态场景分割线程中，通过全景分割技术生成的动态掩码被进一步细化为两种类型，即动态物体掩码和静态背景掩码，为动态视觉 SLAM 的后续处理提供了多层次的支持。(1) 动态物体掩码用于标记具有明确边界且具备潜在移动性的目标，例如行人、车辆和动物等。这些目标通过全景分割的“对象”类别识别并分割出来。在动态视觉 SLAM 系统中，这些动态物体可能对特征匹配和相机位姿估计产生干扰，因此需要在后

续处理阶段剔除相关特征点。(2) 静态背景掩码用于标记图像中不可计数的、无潜在移动性的区域，例如地面、墙壁和天空等。这些区域为构建稳定地图和实现高精度定位提供了核心支持。此外，那些由于遮挡或未标注类别导致分割结果不确定的区域，可能干扰动态物体和静态背景的准确分类，因此需要通过后续的运动分析线程进一步细化处理，明确其运动属性。分割结果如图 3-3 和图 3-4 所示。

然而，单纯依赖全景分割的结果可能不足以完全区分动态物体与静态背景，特别是在一些复杂场景中，例如遮挡或未标注区域的干扰。为了进一步提高动态掩码的准确性和鲁棒性，本研究引入了基础矩阵用于对动态物体掩码进行优化。

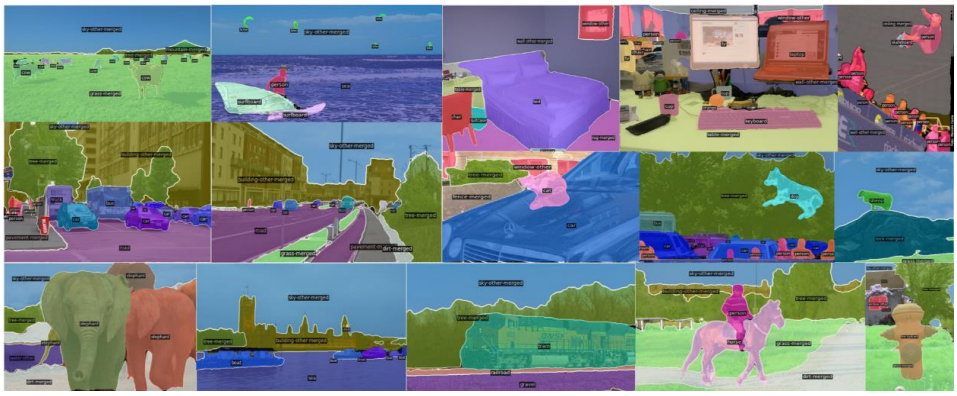


图 3-3 YOSO 分割结果

Fig.3-3 YOSO segmentation results

图 3-4 TUM 数据集分割结果

Fig.3-4 TUM dataset segmentation results

基础矩阵是视觉 SLAM 中相邻两帧图像间几何关系的重要表示。在动态场景分割线程中，它通过对特征点的几何一致性检测，将动态特征点与静态特征点进行初步分类，进一步优化动态掩码的质量。基础矩阵满足以下约束关系：

$$x_2^{\mathrm{T}} F x_1 = 0 \tag{3-1}$$

其中 $x_1 = [u_1, v_1, 1]$ 和 $x_2 = [u_2, v_2, 1]$ 分别表示第一帧和第二帧图像中的对应点齐次坐标，F 表示基础矩阵。基础矩阵的计算过程主要分为特征点提取与匹配、八点算法求

解、RANSAC 鲁棒估计和几何一致性验证四个步骤。首先，从相邻两帧图像中提取 ORB 关键点，并通过特征匹配找到特征点对。然后，通过八点算法对基础矩阵进行初步估计：

$$A \cdot f = 0 \tag{3-2}$$

其中，A是由特征点对构成的$N\times9$的系数矩阵$N\geq8$，f为基础矩阵F的向量化形式。通过奇异值分解对A进行分解，可以求解基础矩阵的初步估计值。然而，实际应用中，由于错误匹配点的存在，直接使用八点算法求得的基础矩阵缺乏鲁棒性，因此引入 RANSAC 算法进行优化。在 RANSAC 算法中，随机采样若干对匹配点计算基础矩阵，并根据以下几何一致性条件筛选内点：

$$\left| x_2^{\mathrm{T}} F x_1 \right| < \varepsilon \tag{3-3}$$

其中，ε为误差阈值。通过多次采样迭代，选取内点数最多的基础矩阵作为最终结果。几何一致性检测过程中，满足约束条件的点对被标记为静态特征点，未满足条件的点对则被标记为动态特征点。将静态特征点剔除出动态掩码，得到更高精度的动态区域标记。动态特征点集合和静态特征点集合的划分公式如下：

$$\begin{aligned} P_d &= \{(x_1, x_2) \,||\, x_2^{\mathrm{T}} F x_1 | \geq \varepsilon\} \\ P_s &= \{(x_1, x_2) \,||\, x_2^{\mathrm{T}} F x_1 | < \varepsilon\} \end{aligned} \tag{3-4}$$

3.2.1.2 短期数据关联与动态目标分类

在全景分割线程生成动态掩码后，需要进一步对动态物体的特征进行分类与关联，以实现动态目标的有效跟踪与特征点的精准剔除。短期数据关联通过相邻帧间的目标匹配和轨迹跟踪，确保动态目标在时间维度上的连续性和一致性，而动态目标分类则进一步利用分割结果对动态目标进行精确标记。

短期数据关联是对连续帧中的动态目标进行匹配与跟踪的关键环节。首先，通过全景分割生成的动态掩码，提取当前帧与上一帧中的动态目标边界框及轮廓信息。利用 IoU 计算两帧中动态目标之间的匹配度，IoU 的计算公式为：

$$\mathrm{IoU}(i,j) = \frac{\left| C_k(i) \cap C_{k-1}(j) \right|}{\left| C_k(i) \cup C_{k-1}(j) \right|} \tag{3-5}$$

其中，$C_k(i)$和$C_{k-1}(j)$分别表示当前帧和上一帧中目标的掩码轮廓。当$\mathrm{IoU}(i,j)>\tau$时(τ为阈值)，判定目标i和j属于同一物体。通过 IoU 的计算，动态目标在短时间尺度上的匹配得以初步实现。对于多目标场景中的同类别目标，进一步引入特征向量匹配，通过计算当前帧与上一帧目标特征向量间的欧几里得距离$D(i,j)$完成区分：

$$D(i,j)=\left\|f_k(i)-f_{k-1}(j)\right\| \tag{3-6}$$

其中，$f_k(i)$和$f_{k-1}(j)$分别表示目标i在当前帧和上一帧中的特征向量。当距离小于阈值时，进一步确认目标匹配。为提升匹配的鲁棒性，短期数据关联引入了运动轨迹预测机制。对于已匹配的动态目标，利用卡尔曼滤波对其运动状态进行预测，以优化目标匹配过程。卡尔曼滤波的预测公式为：

$$\hat{x}_{k|k-1}=F_t\cdot x_{k-1|k-1}+w_k \tag{3-7}$$

其中，$\hat{x}_{k|k-1}$为预测状态，F_t为状态转移矩阵，w_k过程噪声。预测轨迹与当前帧的目标位置相结合，确保动态目标的匹配更加精准。对于当前帧中未能与上一帧匹配的目标，将其标记为新目标并初始化轨迹信息，以便后续追踪。

动态目标分类通过分割结果、几何一致性验证以及短期数据关联的轨迹信息，对动态目标进行精确分类。分类任务以动态目标的特征向量为基础，通过结合全景分割的类别标签$L_{pan}(i)$和目标的运动属性，输出最终分类结果。具体来说，通过分类器对每个动态目标i的特征进行分析，其分类公式为：

$$C_d(i)=\Gamma(f_i,L_{pan}(i)) \tag{3-8}$$

分类结果包括目标的类别以及运动特性。这一分类进一步细化了动态目标的属性，为后续运动分析线程的动态来源验证提供了更加准确的数据支持。

3.2.2 运动分析

运动分析线程的主要任务是基于动态场景分割线程生成的动态掩码，区分物体的真实运动和相机运动造成的表观运动，进而优化动态掩码和动态特征点的处理。在这一过程中，采用光流估计作为运动信息捕捉的基础，与几何一致性验证相结合，为动态视觉 SLAM 系统提供了更为精确的动态与静态特征点分类。

3.2.2.1 动态场景分析

FastFlowNet[65]是一种基于深度学习的轻量级光流估计网络，通过粗到精(coarse-to-fine)策略和轻量级特征提取器，在保持较高精度的同时显著提升计算效率，如图 3-5 所示。

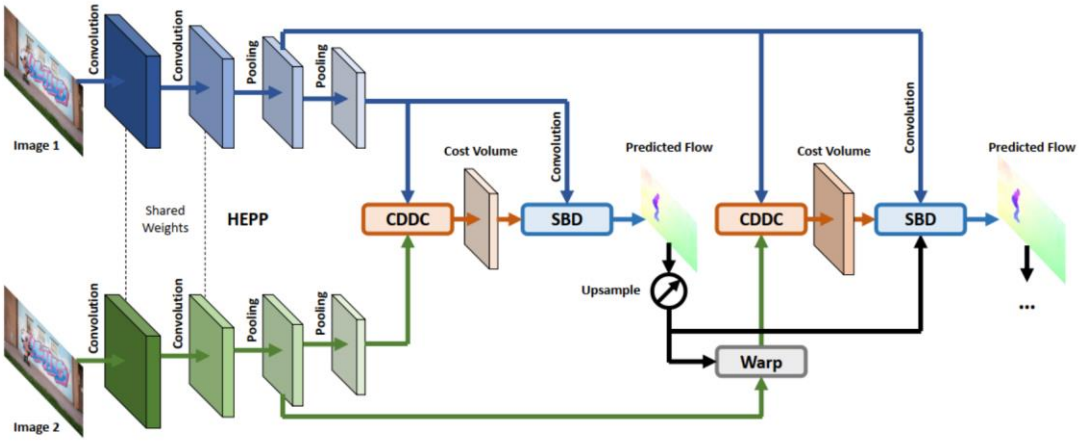


图 3-5 FastFlowNet 算法框架[65]

Fig.3-5 FastFlowNet algorithm framework[65]

FastFlowNet 通过学习图像间的运动模式，估计像素点从 t 时刻到 $t+1$ 时刻的位移 (u,v)。对于输入的关键帧 I_t 和 I_{t+1}，其利用卷积层和池化层构建轻量的特征提取器来提取多尺度特征，表达如下

$$F_t^k, F_{t+1}^k = \text{Extract}_k(I_t, I_{t+1}) \tag{3-9}$$

其中 F_t^k, F_{t+1}^k 分别表示 I_t, I_{t+1} 在第 k 个尺度上的特征图，$k=1,2,3,...,K$。FastFlowNet 采用粗到精的策略，在每个尺度上计算光流，并通过上采样和残差估计逐步优化，通过构建紧凑的成本量和用于回归光流的解码器，在保持高精度的光流估计的同时减少计算量。在第 k 个尺度上，光流估计依赖于 F_t^k, F_{t+1}^k 之间的特征匹配。匹配成本通过相关层计算，表示为：

$$C^k(x,y,u,v) = \sum_{(i,j)\in\mathcal{N}} F_t^k(x+i, y+j) \cdot F_{t+1}^k(x+u+i, y+v+j) \tag{3-10}$$

其中 $C^k(x,y,u,v)$ 表示位置 (x,y) 处候选位移 (u,v) 的匹配成本。基于成本 C^k，使用回归网络预测光流残差：

$$\Delta v^k = \text{Regressor}_k(C^k) \tag{3-11}$$

在第 k 个尺度上的光流估计结合了上一尺度的上采样光流和当前尺度的残差：

$$v^k = v_{\text{up}}^{k+1} + \Delta v^k \tag{3-12}$$

通过从粗到精的迭代，在最细尺度 $(k=1)$ 上，FastFlowNet 输出最终的光流场：

$$v = v^1 = (u,v) \tag{3-13}$$

在运动分析线程中，通过光流幅值和方向对特征点进行初步筛选。当光流幅值超过设定的阈值时，特征点被初步标记为动态点。光流估计为后续几何一致性验证奠定了基础，同时为动态运动来源的进一步分析提供了关键数据支持。

为了进一步优化动态特征点的分类，结合动态场景分割线程中已计算的基础矩阵 F，通过几何一致性验证区分动态点的运动来源。基础矩阵是描述相邻帧间几何关系的核心工具，利用几何误差 Err 作为判断特征点几何一致性的标准：

$$Err = \left| x_2^{\mathrm{T}} F x_1 \right| \tag{3-14}$$

当 $Err < \varphi$（φ 为设定阈值)时，特征点被判定为静态点；当 $Err \geq \varphi$ 时，特征点被标记为动态点。此外，为了区分动态点的真实运动来源，进一步结合基础矩阵计算的相机运动分量 v_{cam}，从光流矢量中扣除相机运动分量，得到动态物体的真实运动矢量 v_{obj}：

$$v_{obj} = v - v_{cam} \tag{3-15}$$

如果 v_{obj} 的幅值显著大于零，则该特征点被判定为物体真实运动；反之，则视为相机运动引起的伪动态点，并重新标记为静态点。结合光流估计和几何一致性验证的结果，运动分析线程进一步优化动态掩码分类，并细化特征点的动态属性。通过对动态特征点的光流矢量和几何误差进行联合筛选，可以有效区分由相机运动引起的表观动态点和物体自身的真实运动点。

3.2.2.2 深度辅助验证

在运动分析线程中，光流估计与几何一致性验证为动态与静态特征点的分类提供了基础。然而，在复杂动态场景中，仅依赖二维几何信息可能会因遮挡、或稀疏纹理而产生分类误差。为进一步提高动态特征点的分类准确性，本研究引入了深度辅助验证，通过结合深度信息对动态与静态特征点的运动属性进行更精确的分析。

深度辅助验证的核心是利用三维空间中特征点的深度差异，结合光流与几何信息，从三维几何视角剖析特征点的运动来源。具体而言，每个特征点 (x, y) 在图像平面上的运动可以扩展为三维空间的运动矢量 (X, Y, Z)，其中 Z 表示特征点的深度值。在相邻两帧图像中，特征点的运动满足以下公式：

$$X' = R \cdot X + T \tag{3-16}$$

其中，$X = [x, y, z]^{\mathrm{T}}$ 和 $X' = [x', y', z']^{\mathrm{T}}$ 分别表示前后帧中特征点的三维坐标；R 和 T 分别为相机的旋转矩阵和平移向量。通过摄像机内参矩阵 K 和深度图的辅助信息，可以将像素坐标映射到三维空间，从而计算每个特征点的真实位移。在深度辅助验证中，首先对动态掩码中标记的特征点进行深度信息的提取。对于每个特征点，根据其深度值计算其三维坐标，并结合相机的运动模型，估算由相机运动引起的

理论位移矢量 v_{cam}。特征点的实际运动矢量 v_{obj} 则通过以下公式得到：

$$v_{obj} = v_{obs} - v_{cam} \tag{3-17}$$

其中，v_{obs} 是特征点的观测运动矢量，由光流估计得到。如果 v_{obj} 的幅值接近零，则可以判定该点的运动主要由相机运动引起，将其标记为静态点；反之，则判定为物体自身的真实运动。

此外，为提高深度辅助验证的鲁棒性，针对动态场景中常见的遮挡现象，本研究引入了深度一致性检测。对于被遮挡的动态目标，其深度值在时间序列上通常表现出明显的突变。基于此，对于相邻帧中同一特征点的深度值差异，定义深度一致性误差：

$$\Delta Z = \left| Z_t - Z_{t+1} \right| \tag{3-18}$$

其中 Z_t 和 Z_{t+1} 分别为前后帧中特征点的深度值。当 ΔZ 超过设定的阈值时，特征点可能由于遮挡或物体真实运动而发生位置改变。对于此类点，进一步结合其光流和几何一致性信息进行验证，剔除由误差导致的动态点误分类。深度辅助验证的引入显著提升了动态与静态特征点分类的精度，为动态视觉 SLAM 系统在复杂动态场景中的鲁棒性提供了重要保障。运动分析流程如图 3-6 所示。

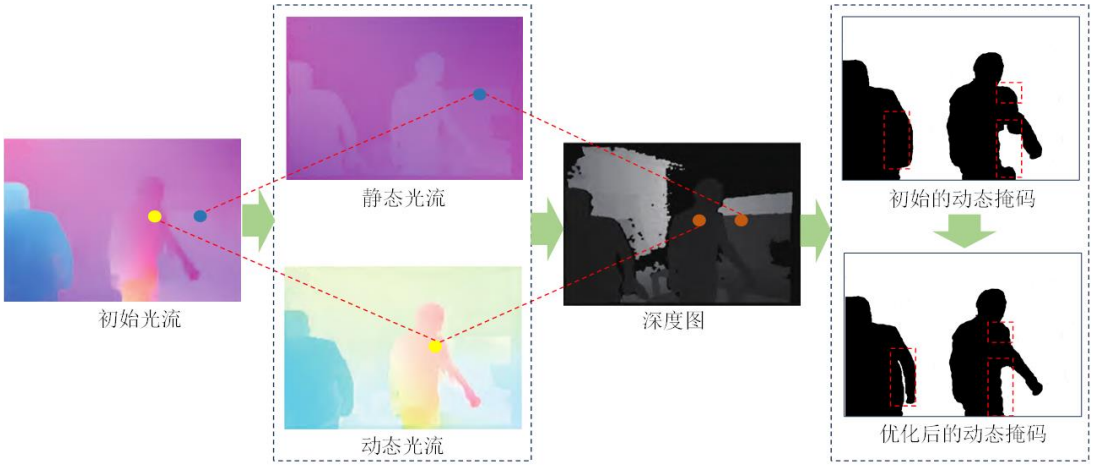


图 3-6 运动分析流程图

Fig.3-6 Motion analysis flowchart

3.2.3 动态特征点过滤

动态特征点过滤是动态视觉 SLAM 算法中关键的步骤，其主要目标是利用动态场景分割线程生成的动态掩码与运动分析线程标记的动态特征点，通过进一步的筛选和剔除，消除因动态物体干扰导致的误匹配特征点，以确保系统在动态场

景中的稳定性和鲁棒性。与运动分析线程中的几何一致性验证相衔接，动态特征点剔除直接基于已标记的动态点集合进行处理，进一步优化动态掩码和静态特征点的可靠性。

基于运动分析线程的验证结果，系统直接对动态点进行处理。这些特征点由运动分析线程标记，系统将这些点作为输入，通过检查其与静态点的空间分布关系和动态掩码标记，剔除与相机运动模式不符的动态点。对于动态点中光流矢量幅值异常大或方向与背景整体运动不一致的点，通过进一步结合深度辅助验证结果，重新评估其运动来源。若验证结果显示动态点的行为与静态背景相似，则将其重新标记为静态点；否则，将其保留为动态点并剔除。此外，对于短时间内变化剧烈的特征点，通过与相邻帧的动态掩码对比，进一步验证其运动来源是否异常。这一处理机制有效减少了因动态物体遮挡或噪声引起的误分类。

引入置信度机制对动态点进行加权剔除。为每个动态点分配置信度评分C，综合考虑光流一致性与动态掩码分割精度，计算公式为：

$$C = w_1 C_{flow} + w_2 C_{mask} \tag{3-19}$$

其中，C_{flow}, C_{mask}分别光流一致性评分与动态掩码分割评分，w_1, w_2为权重参数。当置信度评分小于设定阈值时，系统判定该点为低置信度动态点并剔除；否则，将其保留为高置信度动态点。置信度机制的引入显著提升了动态点剔除的灵活性，减少了误剔除的风险，同时使系统能够根据不同场景的需求动态调整剔除策略。

3.3 实验结果与分析

3.3.1 TUM 数据集实验

TUM 数据集是评估视觉 SLAM 算法性能的基准之一，广泛应用于验证算法在动态场景中的鲁棒性和定位精度。该数据集提供了高分辨率的 RGB-D 图像序列，并配备高精度的相机位姿真值，涵盖多种动态场景，如行走、旋转、部分动态以及完全静态场景等复杂环境。本实验为验证动态场景下运动目标对 SLAM 算法定位精度的影响，选用了 TUM 数据集中的 fr3 序列数据集。fr3 序列包括高动态 walking 场景，具有大范围运动和复杂动态，对 SLAM 算法的运动估计提出了极大挑战；低动态 sitting 场景，具有小范围的局部动态，有助于评估算法在相对静态场景中的表现。

在本实验中，为全面评价 SLAM 算法在动态场景中的性能，主要采用了相对

位姿误差(RPE)和绝对轨迹误差(ATE)作为 SLAM 算法的评价指标。ATE 用于衡量估计轨迹与真实轨迹的一致性，通过估计轨迹与真实轨迹配准后的位置偏差，反映全局误差。RPE 强调短时间间隔内的轨迹局部精度，通过分析相邻帧之间的位姿变化，评估局部一致性与漂移情况。为了直观地分析和比较 SLAM 算法的性能，本实验从均方根误差(RMSE)、平均值(Mean)和标准差(SD)三个方面对实验结果进行量化分析。实验结果如表 3-1，表 3-2 和表 3-3 所示。

实验显示，DSS-SLAM 在各项指标上均优于 ORB-SLAM3 算法。在高动态场景，动态物体的快速运动和相机剧烈移动对 SLAM 系统的轨迹估计带来了双重挑战。实验结果显示，DSS-SLAM 在这些场景中表现出显著的性能优势。从表 3-1 的 ATE 结果可以看出，DSS-SLAM 在 fr3_walking_xyz 场景中的 RMSE 降低了 98.23%，而在 fr3_walking_rpy 场景中则降低了 97.11%，同时 SD 也显著减小。这表明，DSS-SLAM 通过其动态特征点剔除和运动分析机制，有效减少了误匹配点的干扰，提高了轨迹估计的精度和鲁棒性。根据表 3-2 的结果，DSS-SLAM 在 fr3_walking_xyz 和 fr3_walking_rpy 场景中的 RMSE 分别降低了 92.12%和 86.93%。这一结果表明，DSS-SLAM 在局部位姿估计中表现出了更高的精度，能够更好地处理动态环境中的复杂运动模式。此外，从表 3-3 的结果来看，DSS-SLAM 在高动态场景中的旋转位姿估计同样表现出明显的优势，例如在 fr3_walking_xyz 场景中，其 RMSE 优化幅度高达 92.80%，而在 fr3_walking_rpy 场景中优化幅度也达到 87.73%。这些结果表明，DSS-SLAM 能够在复杂动态场景中准确处理相机的旋转运动，从而提升系统整体的鲁棒性。

在低动态场景中，动态物体的运动较小甚至静止，相机的运动也较为平稳，对 SLAM 系统的干扰显著减少。且 ORB-SLAM3 算法使用了 RANSAC 对匹配的动态特征进行剔除，在面对低速运动的物体时有着较好的处理结果。实验表明，在这种场景下 DSS-SLAM 和 ORB-SLAM3 的性能差距相对较小，但 DSS-SLAM 仍然表现出更高的稳定性和精度。例如，在 fr3_sitting_static 场景中，DSS-SLAM 的 ATE 的 RMSE 值为 0.0053m，相比 ORB-SLAM3 提升幅度达 39.08%；而在 RPE 的 RMSE 指标上，DSS-SLAM 的值为 0.0080m，相较于 ORB-SLAM3 的 0.0089m，提升幅度为 10.08%。这一系列结果表明，即使在动态干扰较少的场景中，DSS-SLAM 的动态特征点剔除和优化机制仍然能够有效发挥作用，通过更精确的动态点过滤，实现了对低动态场景性能的进一步提升。

通过对高动态和低动态场景的综合实验分析可以发现，DSS-SLAM 通过动态特征点剔除、运动分析和数据关联等机制，显著降低了 ATE、RPE 值，表现出极

强的动态干扰鲁棒性。尽管 ORB-SLAM3 在一些场景中的表现已经接近理论下限，但 DSS-SLAM 通过更稳定的特征点处理和优化机制，进一步减少了误差，提高了系统的稳定性。

表 3-1 绝对轨迹误差结果

Table 3-1 Absolute trajectory error results

fr3 序列	ORB-SLAM3/m			DSS-SLAM/m			提升幅度/%		
	RMSE	MEAN	SD	RMSE	MEAN	SD	RMSE	MEAN	SD
fr3_walking_xyz	0.6217	0.5288	0.3312	0.0110	0.0124	0.0060	98.23	97.66	98.19
fr3_walking_rpy	0.8127	0.6913	0.4342	0.0235	0.0207	0.0173	97.11	97.01	96.02
fr3_walking_static	0.4011	0.3723	0.1549	0.0066	0.0057	0.0030	98.35	98.47	98.06
fr_walking_half	0.3863	0.3394	0.1845	0.0183	0.0212	0.0107	95.26	93.75	94.20
fr3_sitting_xyz	0.0092	0.0076	0.0045	0.0082	0.0073	0.0040	10.87	3.95	11.11
fr3_sitting_static	0.0087	0.0075	0.0039	0.0053	0.0050	0.0029	39.08	33.33	25.64
fr3_sitting_half	0.0227	0.0185	0.0123	0.0131	0.0110	0.0101	19.38	12.97	14.63
fr3_sitting_rpy	0.0223	0.0163	0.0139	0.0200	0.0158	0.0116	10.31	3.07	16.55

表 3-2 平移相对位姿误差结果

Table 3-2 Translation relative pose error results

fr3 序列	ORB-SLAM3/m			DSS-SLAM/m			提升幅度/%		
	RMSE	MEAN	SD	RMSE	MEAN	SD	RMSE	MEAN	SD
fr3_walking_xyz	0.3772	0.2715	0.2431	0.0297	0.0175	0.0185	92.12	93.56	92.37
fr3_walking_rpy	0.3346	0.2261	0.2295	0.0437	0.0375	0.0341	86.93	83.41	85.13
fr3_walking_static	0.1872	0.0764	0.1746	0.0093	0.0060	0.0108	95.04	92.21	93.79
fr_walking_half	0.3472	0.1857	0.2887	0.0373	0.0290	0.0208	89.25	84.36	92.78
fr3_sitting_xyz	0.0144	0.0098	0.0057	0.0115	0.0077	0.0044	20.13	21.45	22.31
fr3_sitting_static	0.0089	0.0079	0.0045	0.0080	0.0068	0.0040	10.08	13.92	10.78
fr3_sitting_half	0.0237	0.0170	0.0165	0.0210	0.0151	0.0145	11.51	10.98	12.31
fr3_sitting_rpy	0.0260	0.0218	0.0167	0.0252	0.0208	0.0161	3.26	4.42	3.46

表 3-3 旋转相对位姿误差结果

Table 3-3 Rotational relative pose error results

fr3 序列	ORB-SLAM3/°			DSS-SLAM/°			提升幅度/%		
	RMSE	MEAN	SD	RMSE	MEAN	SD	RMSE	MEAN	SD
fr3_walking_xyz	6.5643	4.6112	3.7822	0.4726	0.3975	0.2886	92.80	91.38	92.37
fr3_walking_rpy	5.2982	4.0267	4.0113	0.6501	0.6664	0.5062	87.73	83.45	87.38
fr3_walking_static	2.7470	1.1768	2.8194	0.2027	0.1105	0.2334	92.62	90.61	91.72
fr_walking_half	6.1134	4.1202	5.0691	0.7104	0.4355	0.4699	88.38	89.43	90.73
fr3_sitting_xyz	0.3776	0.4037	0.2583	0.3489	0.3817	0.2423	7.61	5.46	6.21
fr3_sitting_static	0.2664	0.2502	0.1374	0.2500	0.2347	0.1257	6.16	6.18	8.54
fr3_sitting_half	0.5078	0.5346	0.2794	0.4818	0.5055	0.2583	5.12	5.45	7.56
fr3_sitting_rpy	0.8564	0.7077	0.4712	0.7538	0.6389	0.4209	11.98	9.72	10.67

为能直观体现算法在定位性能方面的误差分布情况，本文绘制了估计轨迹与真实轨迹之间的误差图，如图 3-7 所示。在 fr3_walking_xyz 等高动态场景下，ORB-SLAM3 估计的轨迹与真实轨迹之间存在较大偏差，尤其在动态物体频繁出现遮挡或相机快速移动时，误差幅度显著增大。轨迹在部分区域的偏离较为严重，并出现明显的异常跳跃现象，表明 ORB-SLAM3 在高动态场景中易受到动态特征点干扰的影响。相比之下，本方法 DSS-SLAM 的估计轨迹与真实轨迹的重合度显著提升，误差分布更加均匀且稳定，在动态区域内的偏差明显降低。这一结果表明，DSS-SLAM 在动态特征点剔除和运动分析机制上的优化，显著减轻了动态物体对轨迹估计的干扰，有效提升了系统在高动态场景中的鲁棒性和精度。

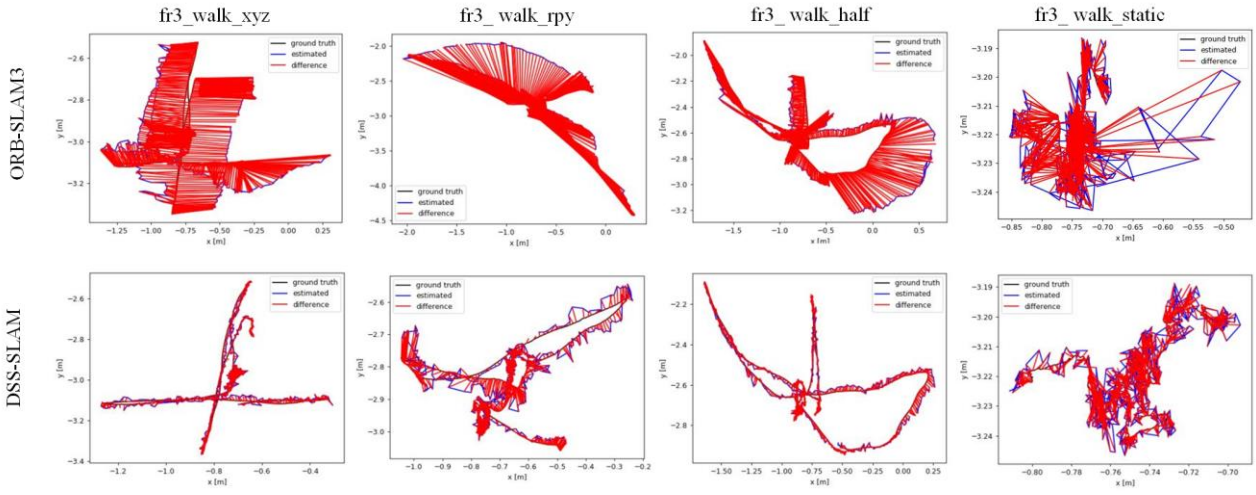


图 3-7 不同动态序列的轨迹误差对比

Fig.3.7 Comparison of trajectory errors of different dynamic sequences

为了进一步验证 DSS-SLAM 在动态场景中的性能优势，与 DS-SLAM、DynaSLAM 和 CFP-SLAM 三种经典动态视觉 SLAM 算法进行绝对轨迹误差的对比分析，实验结果如表 3-4 所示。

在低动态场景中，DSS-SLAM 仍表现出更高的稳定性和精度。例如，在 fr3_sitting_static 场景中，DSS-SLAM 的 RMSE 为 0.0053 m，低于 DS-SLAM 的 0.0078m，且标准差仅为 0.0029m，显示出其更高的轨迹一致性。这种优势归因于 DSS-SLAM 优化的动态特征点剔除机制，即使在动态干扰较少的情况下，仍能够有效剔除误匹配点，进一步提升轨迹估计的鲁棒性。在高动态场景中，DSS-SLAM 的性能优势更加显著。在 fr3_walking_xyz 场景中，DSS-SLAM 的 RMSE 为 0.0110 m，远低于 DS-SLAM 的 0.0247 m 和 DynaSLAM 的 0.0154 m，同时其标准差仅为 0.0060 m，比其他算法的标准差更低。在 fr3_walking_rpy 场景中，DSS-SLAM 的 RMSE 和标准差分别为 0.0235 m 和 0.0173 m，均优于 DS-SLAM、DynaSLAM 和 CFP-SLAM。这表明 DSS-SLAM 在处理快速运动和频繁遮挡的高动态场景时，能够更高效地过滤动态物体的干扰并优化相机位姿估计。

表 3-4 DSS-SLAM 与其他动态 SLAM 的 ATE 对比

Table 3-4 Comparison of ATE between DSS-SLAM and other dynamic SLAM

fr3 序列	DS-SLAM		DynaSLAM		CFP-SLAM		DSS-SLAM	
	RMSE	SD.	RMSE	SD.	RMSE	SD.	RMSE	SD.
fr3_sitting_static	0.0078	0.0038	0.0062	0.0029	0.0053	0.0027	0.0053	0.0029
fr3_sitting_half	0.0166	0.0078	0.0196	0.0091	0.0147	0.0070	0.0131	0.0101
fr3_sitting_rpy	--	--	0.0722	0.0578	0.0253	0.0154	0.0158	0.0116
fr3_sitting_xyz	0.0115	0.0056	0.0162	0.0071	0.0090	0.0042	0.0082	0.0040
fr3_walking_xyz	0.0247	0.0161	0.0154	0.0081	0.0141	0.0072	0.0110	0.0060
fr3_walking_rpy	0.4452	0.2356	0.0410	0.0260	0.0368	0.0231	0.0235	0.0173
fr3_walking_half	0.0305	0.0160	0.0277	0.0145	0.0237	0.0114	0.0183	0.0107
fr_walking_static	0.0081	0.0036	0.0065	0.0032	0.0067	0.0031	0.0066	0.0030

3.3.2 KITTI 数据集实验

KITTI 数据集是自动驾驶和视觉 SLAM 研究领域的标准数据集之一。该数据集包含丰富的真实驾驶场景，提供了高分辨率的 RGB 图像、深度数据以及高精度的 GPS/IMU 位姿真值。KITTI 数据集中的序列涵盖多种复杂场景，包括城区驾驶、高速行驶和乡村道路等场景，具有动态物体多、视角变化大等特点，为动态视觉 SLAM 算法的验证提供了极具挑战性的测试环境。本实验选取 KITTI 数据集

中的部分动态序列，采用绝对轨迹误差(ATE)和相对位姿误差(RPE)作为评价指标，实验结果如表 3-5 所示。

在 ATE 指标方面，DSS-SLAM 在多个动态序列上表现出优异的性能。例如，在 0000 序列中，DSS-SLAM 的 ATE 为 1.14m，相较于 ORB-SLAM3 的 1.31m 降低了约 12.98%。在 0008 序列中，DSS-SLAM 的 ATE 为 1.21 m，而 ORB-SLAM3 的 ATE 为 1.43m，误差减少了 15.38%。这一结果表明，DSS-SLAM 在处理具有大量动态物体和快速运动的复杂场景时，其动态特征点剔除与运动分析机制能够显著降低误匹配点对轨迹估计的干扰，优化了轨迹的整体精度。

在 RPE 指标方面，DSS-SLAM 同样表现出明显优势。以 RPE 的平移误差(m/f)为例，在 0005 序列中，DSS-SLAM 的 RPE 为 0.04m/f，明显优于 ORB-SLAM3 的 0.07 m/f，降低了 42.86%。在 0003 序列中，DSS-SLAM 的 RPE 为 0.05m/f，较 ORB-SLAM3 降低了 28.57%。此外，在相对旋转误差(°/f)上，DSS-SLAM 的结果也普遍优于 ORB-SLAM3 算法。例如，在 0002 序列中，DSS-SLAM 的旋转误差为 0.02°/f，而 ORB-SLAM3 的旋转误差为 0.04°/f。这进一步证明了 DSS-SLAM 在局部位姿估计中对动态场景的鲁棒性。整体来看，尽管 ORB-SLAM3 利用 RANSAC 过滤动态特征点，但在动态物体频繁遮挡或大范围运动场景中依然存在较大误差。DSS-SLAM 结合全景分割与运动分析有效提高了轨迹估计的精度与鲁棒性。实验结果证实了 DSS-SLAM 无论是在高动态环境还是相对稳定的场景，都具备良好的适用性与可靠性。

表 3-5 KITTI 数据集实验

Table 3-5 KITTI dataset experiments

序列	ORB-SLAM3			DSS-SLAM		
	ATE(m)	RPE^t(m/f)	RPE^r(°/f)	ATE(m)	RPE^t(m/f)	RPE^r(°/f)
0000	1.31	0.05	0.06	1.14	0.04	0.05
0001	2.01	0.05	0.04	1.78	0.04	0.04
0002	0.95	0.04	0.04	0.90	0.04	0.02
0003	0.82	0.07	0.05	0.70	0.05	0.04
0004	1.45	0.07	0.07	1.32	0.07	0.06
0005	1.22	0.07	0.03	1.21	0.04	0.03
0006	0.20	0.02	0.04	0.18	0.02	0.03
0007	2.47	0.07	0.07	2.28	0.05	0.04
0008	1.43	0.09	0.04	1.21	0.08	0.04
0009	3.98	0.10	0.07	3.43	0.05	0.06
0010	1.72	0.08	0.06	1.33	0.06	0.03

此外，如图 3-8 所示，在 KITTI 数据集中，本方法的轨迹估计结果与真实轨迹的重合度明显优于 ORB-SLAM3 和 DynaSLAM。具体而言，本方法估计的轨迹误差分布更加均匀，绝对轨迹误差的最大值和最小值范围均显著小于其他两种方法。这表明 DSS-SLAM 能够在复杂动态场景中有效减少轨迹估计误差，展现出更高的轨迹精度和鲁棒性。

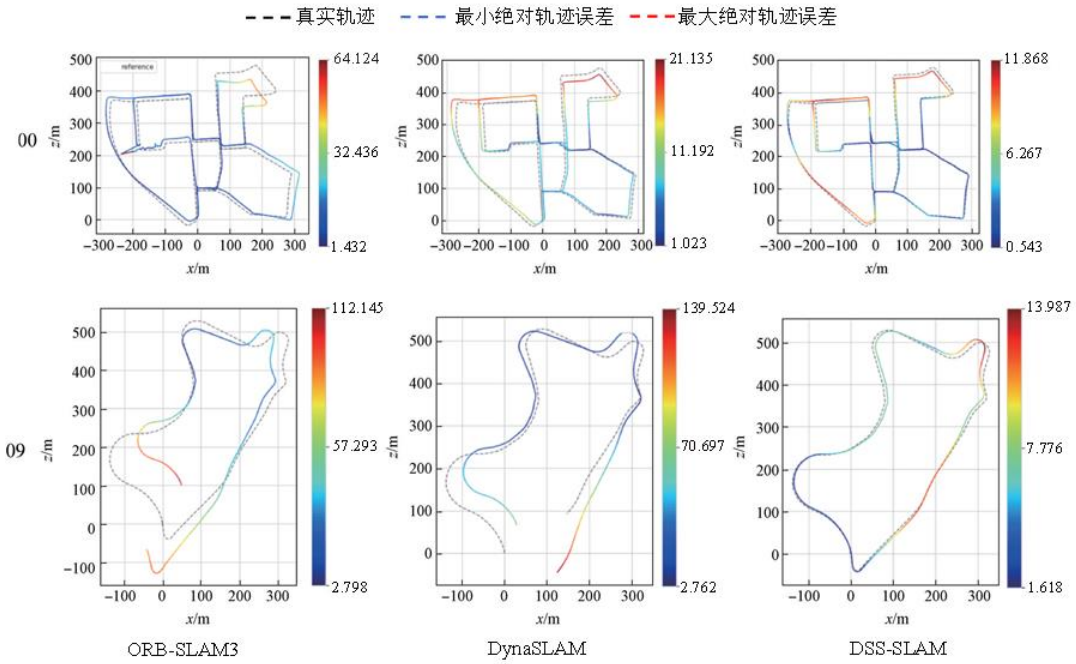


图 3-8 KITTI 数据集 00 与 09 序列轨迹误差效果图

Fig.3-8 Trajectory error effect diagram of KITTI dataset 00 and 09 sequences

3.3.3 EuRoC 数据集实验

EuRoC 数据集是一个广泛用于评估视觉 SLAM 算法性能的标准数据集，包含高精度的 IMU 传感器和立体摄像头采集的数据，覆盖了丰富的室内动态场景和多种难度等级。该数据集包含 11 个飞行序列，分为两个主要场景：Machine Hall 和 Vicon Room。每个场景中又根据飞行轨迹和动态环境的复杂性分为简单(easy)、中等(medium)和困难(difficult)三个难度等级，其中困难场景中动态物体较多且更具挑战性。实验以绝对轨迹误差的 RMSE 为指标，评估 DSS-SLAM 与 ORB-SLAM3 在 EuRoC 数据集上的定位精度，实验结果如表 3-6 所示。

DSS-SLAM 在所有序列上均优于 ORB-SLAM3，尤其是在 medium 和 difficult 难度场景中性能提升更加显著。在 MH_03_medium 场景下，DSS-SLAM 的 RMSE 为 0.093m，相较于 ORB-SLAM3 的 0.146m，降低了 36.3%；而在 MH_05_difficult 场景下，DSS-SLAM 的 RMSE 为 0.062m，较 ORB-SLAM3 的 0.122m，误差减少

了 49.2%。这表明 DSS-SLAM 在处理复杂动态场景时，能够更有效地减少动态物体对轨迹估计的干扰。对于 easy 场景，DSS-SLAM 与 ORB-SLAM3 的性能差距相对较小。在 MH_01_easy 场景下，DSS-SLAM 的 RMSE 为 0.056m，而 ORB-SLAM3 为 0.083m，降低了 32.5%；在 V1_01_easy 场景下，DSS-SLAM 的 RMSE 为 0.060m，较 ORB-SLAM3 的 0.068m，降低了 11.8%。这表明在动态干扰较少的简单场景中，两种算法的性能较为接近，但 DSS-SLAM 依然保持了更高的轨迹估计精度。

在更具挑战性的场景中，例如 V2_03_difficult，DSS-SLAM 的 RMSE 为 0.112m，而 ORB-SLAM3 的 RMSE 高达 0.182m，误差降低了 38.5%。这一结果进一步验证了 DSS-SLAM 在动态场景中更强的鲁棒性与适用性。其动态特征点剔除机制和运动分析线程的优化显著减少了由动态物体和遮挡造成的误差，提升了系统的定位精度。如图 3-9 所示，展示了在 EuRoC 数据集的 MH 序列，以及 V1 和 V2 序列中 DSS-SLAM 估计的轨迹与真实轨迹的示例。

表 3-6 EUROC 数据集实验

Table 3-6 EUROC dataset experiment

序列	ORB-SLAM3	DSS-SLAM
MH_01_easy	0.083	0.056
MH_02_easy	0.101	0.091
MH_03_medium	0.146	0.093
MH_04_diffcult	0.119	0.089
MH_05_diffcult	0.122	0.062
V1_01_easy	0.068	0.060
V1_02_medium	0.139	0.044
V1_03_diffcult	0.158	0.096
V2_01_easy	0.053	0.035
V2_02_medium	0.110	0.068
V2_03_diffcult	0.182	0.112
平均值	0.116	0.015

3.3.4 动态特征点过滤实验

在动态视觉 SLAM 系统中，动态特征点的剔除是确保复杂场景下系统稳定运行的关键环节。尤其在四足机器人的实际应用中，环境中动态物体会对特征点的提取与匹配产生干扰，进而影响位姿估计的准确性以及地图构建的全局一致性。动态特征点剔除不仅需要考虑物体的先验动态属性，还需结合实际的运动信息，以便在不同场景中实现更精确的动态点检测与过滤。

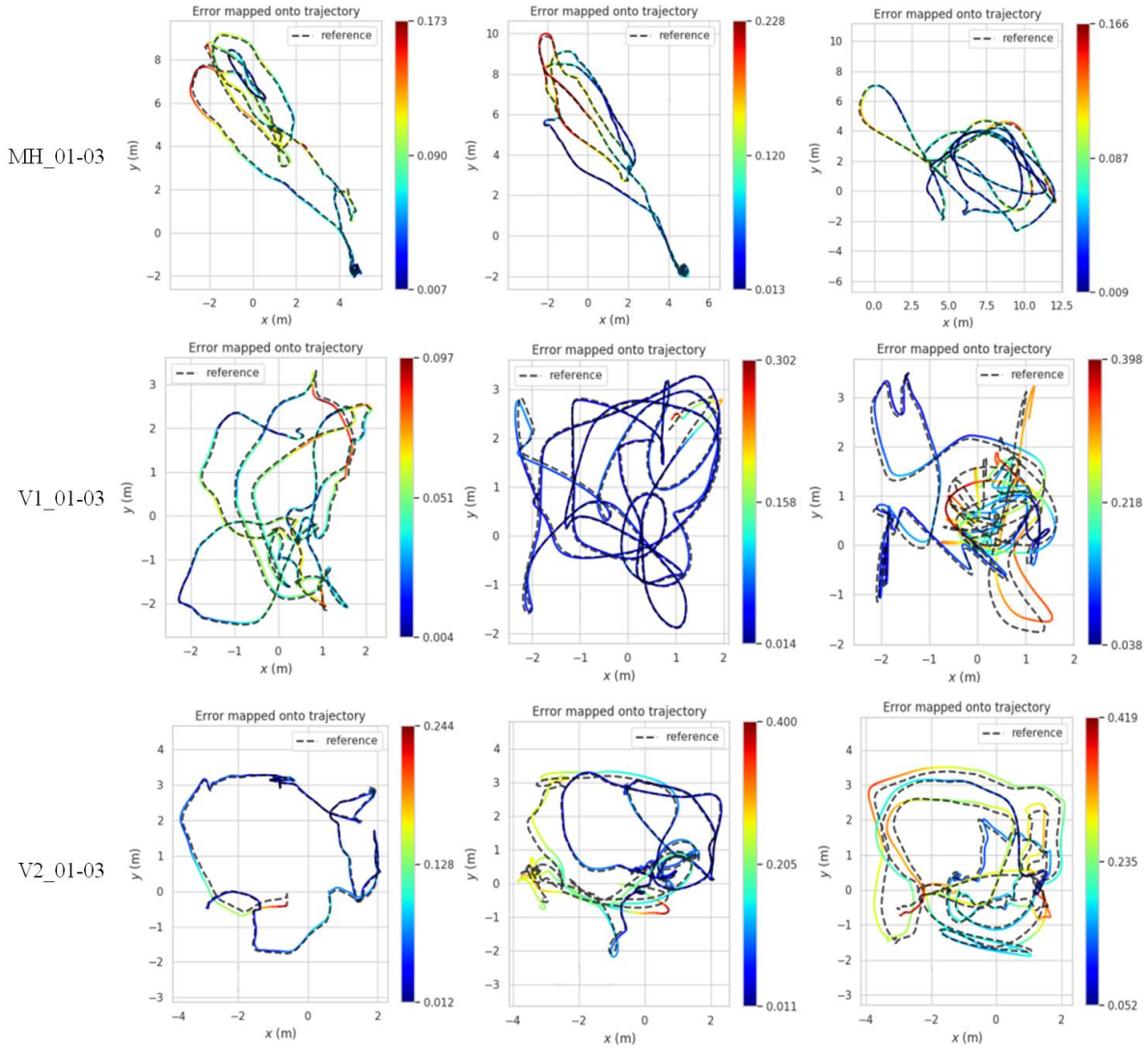


图 3-9 EuRoC 数据集中部分序列的轨迹

Fig.3-9 Trajectory comparison of some sequences in the EuRoC dataset

如图 3-10 所示，选取了 TUM 数据集中具有不同动态特征的 walking 与 sitting 序列，展示了在高动态和低动态场景下动态特征点剔除的效果。图中绿色点代表静态特征点，红色点表示被检测为动态特征点。第一行和第二行展示了 walking 序列中的高动态场景。在图 3-10 (a)中，左侧人物正在移动，因此其特征点在前后帧中存在显著的位移变化，被检测为动态特征点并成功剔除；而右侧人物保持静止，其特征点在多帧中具有高度一致性，被判定为静态特征点。类似地，在图 3-10 (f)中也表现出相同的动态点检测效果。图 3-10 (b)至图 3-10 (f)展示了场景中人物与相机同时处于连续运动状态，此时所有与人物相关的特征点均被准确检测为动态特征点。图 3-10 (g)和图 3-10 (h)则是无动态物体的静态背景场景，此时所有特征点均被保留为静态特征点，进一步验证了系统在静态场景下的鲁棒性。

第三行展示了 sitting 序列中的低动态场景。在图 3-10(i)至图 3-10(k)中，两个人物坐在椅子上，仅手部或上半身存在小幅度的运动。系统能够精准检测到这些局部动态区域的特征点，并将其标记为动态特征点，而静止不动的部分则被保留为静态特征点。图 3-10(l)展示了完全静止的场景，两个人物均无明显运动，所有特征点均被正确识别为静态特征点。

实验结果表明，DSS-SLAM 通过结合全景分割和运动分析等多重机制，精确剔除动态特征点，避免了传统方法因过度剔除静态特征而导致的性能下降。与传统方法不同，DSS-SLAM 并不简单地依赖于先验动态属性来剔除特定类别的物体，而是根据特征点在多帧中的实际运动状态进行动态性判断。这种基于运动信息与几何一致性的动态特征点剔除机制，能够有效避免因过度剔除静态特征而导致的性能损失，同时提升了系统在高动态和低动态场景下的鲁棒性和适应性。

图 3-10 TUM 数据集中的动态特征点过滤示意图

Fig. 3-10 Schematic diagram of dynamic feature point filtering in the TUM dataset

此外，图 3-11 展示了动态物体剔除的可视化实例，可以看到，DSS-SLAM 能够精准剔除那些正在运动的物体，而不会误将静止物体或由于相机运动造成的表观动态误判为动态物体。通过这种精细的处理，系统有效地保留了场景中的关键静态特征点，从而提升了地图构建与位姿估计的精度和鲁棒性。

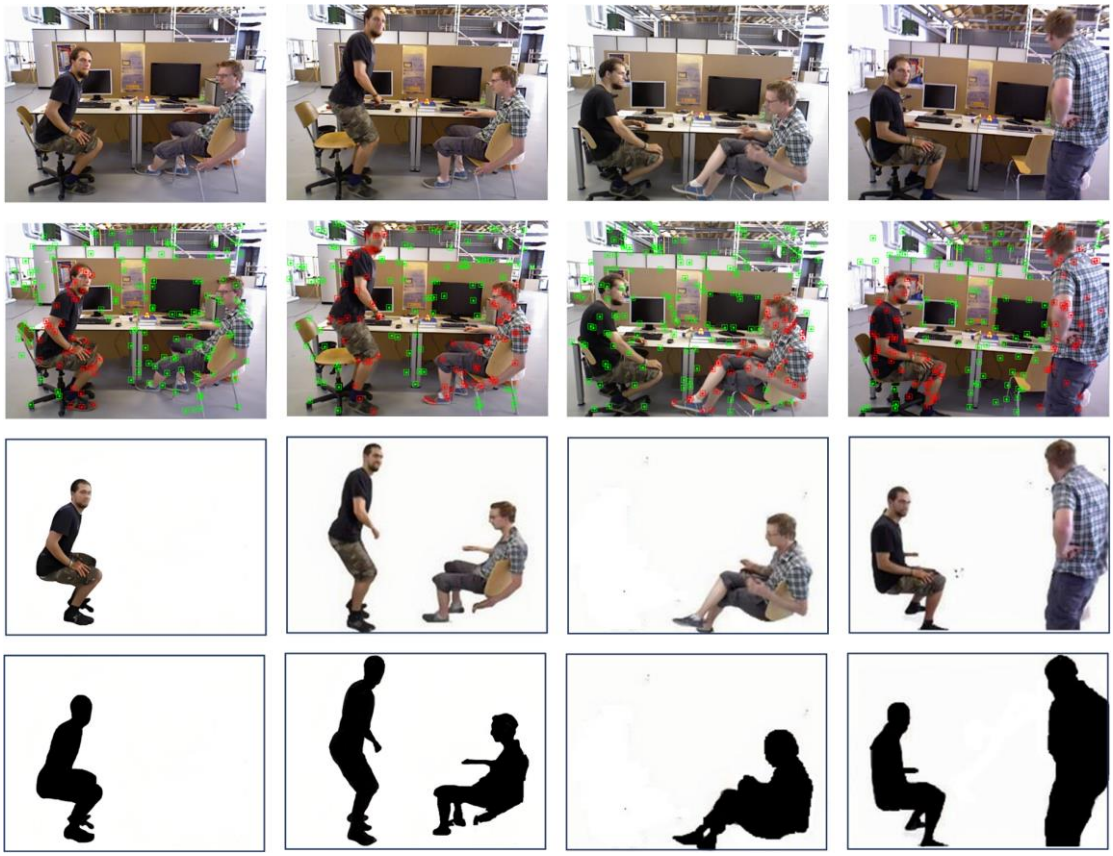


图 3-11 TUM 数据集中的动态掩码示意图

Fig. 3-11 Schematic diagram of dynamic mask in TUM dataset

3.3.5 消融实验

为了验证 DSS-SLAM 算法中各个模块的独立贡献及其对整体性能的影响，本实验设计了一系列消融实验。实验基于 TUM 数据集中的高动态场景(fr3_walking_xyz)和低动态场景(fr3_sitting_static)，对比分析了完整 DSS-SLAM 算法与各种消融版本的表现。

移除全景分割模块，仅依赖基础矩阵的几何一致性验证进行动态特征点的初步分类，结果如表 3-7 所示。在高动态场景中，移除全景分割模块后，系统的绝对轨迹误差显著增加，RMSE 从 0.0110m 增加到 0.0312m，提升了 183.64%。在低动态场景中，ATE 的 RMSE 从 0.0053m 增加到 0.0098m，提升了 84.90%。这一误差激增现象源于几何验证机制的固有缺陷。在缺乏全景掩码先验约束的条件下，基础矩阵估计极易受到动态区域的干扰，这充分表明全景分割模块在动态物体检测和掩码生成过程中发挥着至关重要的作用，且在高动态场景中的重要性更为凸显。

表 3-7 全景分割模块消融实验

Table 3-7 Panoramic segmentation module ablation experiment

序列	DSS-SLAM(m)	移除全景分割(m)	误差增加幅度/%
fr3_walking_xyz	0.0110	0.0312	183.64
fr3_sitting_static	0.0053	0.0098	84.90

移除运动分析模块，仅依赖全景分割模块生成的动态掩码进行特征点剔除，结果如表 3-8 所示。在高动态场景中，系统的 ATE 的 RMSE 从 0.0110m 增加到 0.0245m，提升了 122.72%。在低动态场景中，ATE 的 RMSE 从 0.0053m 增加到 0.0085m，提升了 60.37%。该结果表明，单靠分割无法有效区分表观运动与真实运动。凸显了运动分析模块中光流估计和几何一致性验证的重要性，它们在动态特征点的分类和剔除过程中起到了关键作用。

表 3-8 运动分析模块消融实验

Table 3-8 Motion analysis module ablation experiment

序列	DSS-SLAM(m)	移除运动分析(m)	误差增加幅度/%
fr3_walking_xyz	0.0110	0.0245	122.72
fr3_sitting_static	0.0053	0.0085	60.37

移除了深度辅助验证模块，仅依赖光流和几何一致性验证进行动态特征点分类，结果如表 3-9 所示。在高动态场景中，系统的 ATE 的 RMSE 从 0.0110m 增加到 0.0185m，提升了 68.18%。在低动态场景中，ATE 的 RMSE 从 0.0053m 增加到 0.0078m，提升了 47.16%。这表明在低纹理区域，光流无法有效区分动态物体与视差变化，深度信息的补充对动态特征点的分类至关重要，尤其是在高动态场景中，它提供了必要的补充信息，有效增强了系统的鲁棒性。

表 3-9 深度辅助验证模块消融实验

Table 3-9 Ablation experiment of depth-assisted verification module

序列	DSS-SLAM(m)	移除深度信息(m)	误差增加幅度/%
fr3_walking_xyz	0.0110	0.0185	68.18
fr3_sitting_static	0.0053	0.0078	47.16

实验中，还对 DSS-SLAM 的不同变体与 ORB-SLAM3 进行了实时性分析，如表 3-10 所示。实验均在桌面级计算平台上进行，硬件配置包括 Intel i7 CPU 与 Nvidia GTX 3090GPU。在单帧处理时间方面，DSS-SLAM 的整体处理时间为 70.98ms，高于 ORB-SLAM3 的 33.32ms。这一差异主要源于 DSS-SLAM 在动态

场景处理中引入了如全景分割、运动分析和深度辅助验证等模块，这些模块虽然增加了计算量，但也显著提升了系统的鲁棒性和精度。在系统帧率方面，ORB-SLAM3 在动态场景下的帧率为 30.01FPS，而 DSS-SLAM 的帧率为 14.10FPS。尽管 DSS-SLAM 的帧率较低，但其通过动态特征点剔除、光流计算和深度信息辅助分析等优化手段，确保了在复杂动态环境下的稳定性和鲁棒性。在实际复杂动态环境中，DSS-SLAM 在处理动态物体干扰时的效果优于 ORB-SLAM3。

在移除全景分割模块后，DSS-SLAM 的单帧处理时间降低至 43.26ms，帧率提升至 22.13FPS，提升幅度为 56.95%。这一结果表明，全景分割模块在 DSS-SLAM 中的计算开销较大，其负责物体的分割与掩码生成，造成的密集计算会增加 GPU 负担，移除使得算法的实时性有所提升，但会牺牲一部分定位精度和鲁棒性。在移除运动分析模块的实验中，DSS-SLAM 的处理时间为 52.03ms，帧率为 19.22FPS，提升幅度为 36.31%。最后，在移除深度辅助验证模块后，DSS-SLAM 的单帧处理时间为 65.50ms，帧率为 15.53FPS，提升幅度为 10.14%，这表明在运动分析中，深度信息对动态特征点分类和剔除具有重要作用，而且对实时性的影响相对较小。

表 3-10 实时性分析实验

Table 3-10 Real-time analysis experiment

系统变体	单帧处理时间(ms)	系统帧率(FPS)	帧率变化幅度/%
ORB-SLAM3(基准)	33.32	30.01	--
DSS-SLAM(完全体)	70.98	14.10	--
移除全景分割	43.26	22.13	56.95
移除运动分析	52.03	19.22	36.31
移除深度验证	65.50	15.53	10.14

3.4 基于四足机器人的动态视觉 SLAM 实验分析

为了验证 DSS-SLAM 在四足机器人实际场景中的有效性和稳定性，本实验基于 2.4.1 章节搭建的实验平台，进行真实场景的验证，包含位姿误差和动态点剔除。如图 3-12 所示，在某园区进行了实验，实验包括车辆，行人，以及楼梯区域。DSS-SLAM 在实验中表现出色，能够准确地生成动态掩码，将动态物体与静态背景区分开来，有效剔除真实运动的物体，如正在移动中的车辆和行人，并没有因为机器人的视角移动和车辆具有的动态属性而产生误剔除。在楼梯区域，四足机器人在爬楼过程中存在视角变换和重复纹理，但系统依然保持稳定，展现出很强的鲁棒性。

本文还在室内环境中进行了验证，如图 3-13 所示。实验环境包含低纹理，遮

挡与低动态场景，DSS-SLAM 同样表现出色。对于相邻帧真实运动的目标进行了准确的分割以及特征点的分类，展现了有效处理动态遮挡或低动态复杂场景的能力。

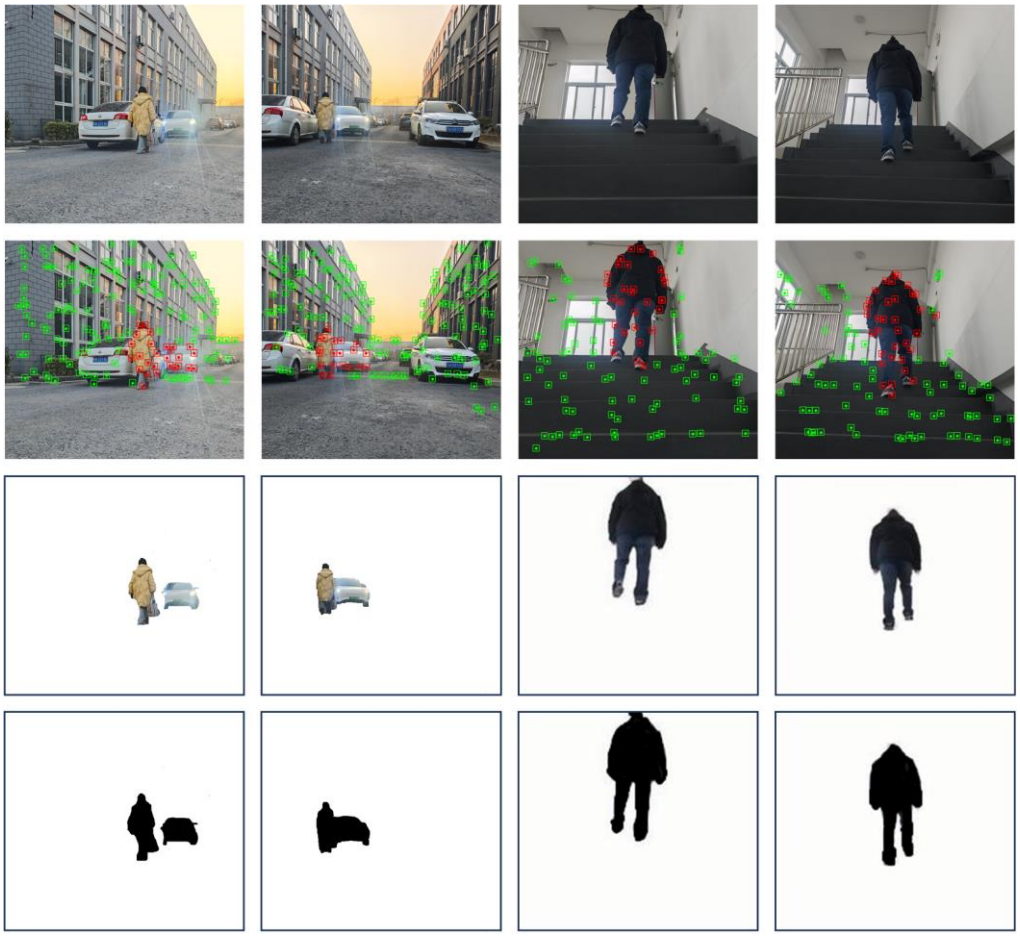


图 3-12 室外场景的动态 SLAM 实验

Fig.3-12 Dynamic SLAM Experiments in Outdoor Scenes

图 3-13 室内场景的动态 SLAM 实验

Fig.3-13 Dynamic SLAM experiments in indoor scenes

为了进一步验证 DSS-SLAM 在真实环境中的性能，将其与 ORB-SLAM3 进行了对比实验，如表 3-11 所示。实验结果表明，DSS-SLAM 在室外和室内场景中的 ATE 和 RPE 均显著低于 ORB-SLAM3。在室外高动态场景中，DSS-SLAM 的 ATE 和 RPE 误差分别降低了 81.73%和 70.14%%，旋转误差降低了 79.62%，这表明 DSS-SLAM 在动态环境中的定位精度和鲁棒性显著优于 ORB-SLAM3。

表 3-11 真实环境对比实验

Table 3-11 Real environment comparison experiment

场景	方法	ATE	RPE(m/f)	RPE(°/f)
室外	ORB-SLAM3	0.6217	0.3372	6.5643
	DSS-SLAM	0.1136	0.1007	1.3378
室内	ORB-SLAM3	0.4011	0.1872	2.7470
	DSS-SLAM	0.1024	0.0858	0.2027

3.5 本章小结

本章面对四足机器人复杂运动场景提出了基于全景分割和运动分析的 DSS-SLAM 算法。通过分割算法实现动态场景分割，运动分析来确定物体真实运动来源并过滤动态特征点，有效克服复杂环境的干扰，提升视觉 SLAM 系统的稳定性和鲁棒性。最后在 TUM、KITTI 等数据集以及真实环境中进行了实验，结果表明无论是在高动态环境还是低动态环境，本章算法与其他方法相比均取得优异结果。

第 4 章 基于深度特征匹配的视觉重定位算法研究

4.1 引言

视觉重定位作为计算机视觉领域的重要研究方向，旨在精确估计相机在场景中的位置。这一技术对于四足机器人在复杂环境中的自主导航至关重要，特别是在室内无 GPS 或动态非结构化环境中。基于结构的视觉重定位方法通过图像检索与特征匹配技术，依靠较强的泛化能力在各种场景中展现出优异的性能。这类方法通过在场景中检索与查询图像相似的图像，并建立图像间的匹配关系，完成视觉定位。其中特征匹配的准确性是决定其定位效果的核心因素。然而，在四足机器人任务中，由于运动引起的视角抖动，动态遮挡以及光照变化或低纹理环境等，特征匹配算法的鲁棒性与精度面临更大的挑战。

为此，本章节借鉴人类匹配图像的直观行为，构建一种有效的深度特征匹配器，以突破现有方法的局限性，从而提升四足机器人在视觉重定位任务中的性能。

4.2 基于特征匹配的重定位算法框架

基于深层特征匹配器的视觉重定位网络，其整体流程如图 4-1 所示，该定位算法整体上由离线部分和在线部分组成。

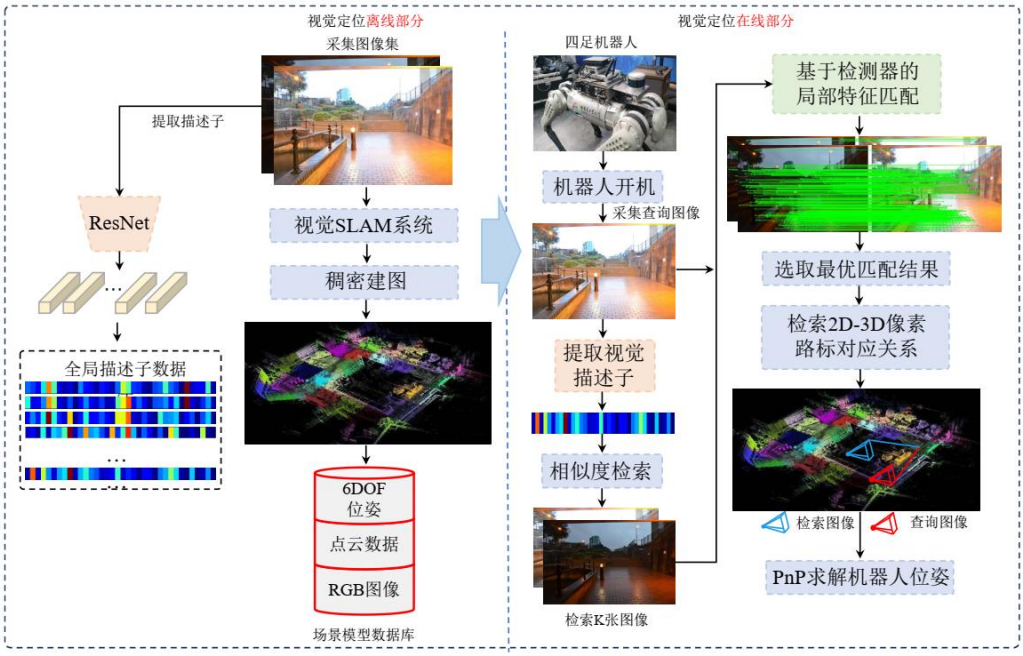


图 4-1 基于深层特征匹配的视觉重定位框架

Fig.4-1 Visual relocalization framework based on deep feature matching

离线部分的任务为在线查询提供可靠的场景信息支持。具体包括以下步骤：首先，进行数据集采集，获取目标场景的高质量图像数据；然后，构建全局描述子数据库，通过提取图像的全局特征，生成用于图像检索的高维描述符索引，便于快速匹配；接着，进行场景的三维重建，获取每帧图像对应的相机位姿；最后，基于三维点云及图像数据，建立场景模型数据库，其中包含图像数据，相机位姿以及三维地图点云。

在线部分的任务是实时估计输入查询图像的相机位姿，该部分的核心是特征匹配。首先，通过图像检索，利用全局描述子数据库，从场景模型中检索出与查询图像最相似的 K 张候选图像，为后续特征匹配缩小搜索范围。然后来进行特征匹配，通过深层特征匹配器建立查询图像与检索图像之间二维像素点的对应关系，并通过已知的检索图像三维路标点信息，建立查询图像的二维像素点与三维路标点的关联。最后，采用基于 RANSAC 的 PnP 算法求解相机的位姿。

4.2.1 基于余弦相似度的图像检索

图像检索的核心在于对图像特征的高效提取与识别。随着深度学习技术的快速发展，神经网络在特征编码能力上显著优于传统的基于内容的图像检索方法。神经网络能够有效提取高维的全局图像特征，从而显著提升检索的准确性和效率。本研究中，采用了在 ImageNet 上训练的 ResNet50 模型对图像进行高维特征提取，并结合余弦相似度实现全局位置的图像检索。

具体而言，本研究构建的全局描述子数据库为 $V_{DB}=\{d_1,d_2,d_3,...,d_N\}$，其中 d_i 表示第 i 张数据库图像的高维特征向量。对于查询图像，通过 ResNet50 提取其全局特征向量 q。随后，计算查询图像特征 q 与数据库中每个描述子 d_i 的余弦相似度，其定义为：

$$S(q,d_i)=\frac{q^{\mathrm{T}}d_i}{\|q\|\|d_i\|} \tag{4-1}$$

其中，$q^{\mathrm{T}}d_i$ 表示向量间的点击，$\|q\|$ 和 $\|d_i\|$ 分别为 q 和 d_i 的欧几里得范数，用以归一化特征向量长度。通过计算 $S(q,d_i)$ 的值，可以度量查询图像与数据库中每幅图像的相似度。根据计算结果，选取余弦相似度最高的 K 张图像作为检索结果，即：

$$Top_K=\underset{i\in\{1,2,3,...,N\}}{\arg\max}\ S(q,d_i) \tag{4-2}$$

4.2.2 基于自适应权重共享特征匹配网络

特征匹配技术是视觉重定位中的核心环节，其作用在于建立图像特征点之间的对应关系，匹配的数量与精度对机器人视觉重定位的效果具有决定性影响。现有的特征匹配技术侧重优化网络架构来提高匹配性能，与之不同，本研究提出了Ada-Matcher，一种基于检测器的深层 Transformer 特征匹配网络。Ada-Matcher 通过自适应权重共享机制构建深层注意力图神经网络，模仿人类匹配行为，以较低的计算和存储负担实现更高的匹配精度，如图 4-2 所示。理论上，堆叠更多的 Transformer 层可以增强匹配性能，但不可避免地会导致模型参数成比例增长。为了解决该问题，Ada-Matcher 使用自适应权重共享机制来探索更深的 Transformer 层，该策略允许 Transformer 层共享一部分参数，大大减少了模型参数量。随着 Transformer 层的增加，当网络学习更高级的语义信息时，会导致特征崩溃。为此，轻量级特征变换被应用于每个 Transformer 层，以进一步丰富特征多样性并防止特征崩溃。得益于自适应权重共享机制和特征变换，Ada-Matcher 可以堆叠大量的 Transformer 层，同时保证合理的参数数量和特征多样性。此外，Ada-Matcher 依赖于本研究中设计的掩模注意力技术，通过生成稀疏注意力掩模以过滤掉相关性较低的信息，降低冗余信息的负面影响从而提高模型效率。

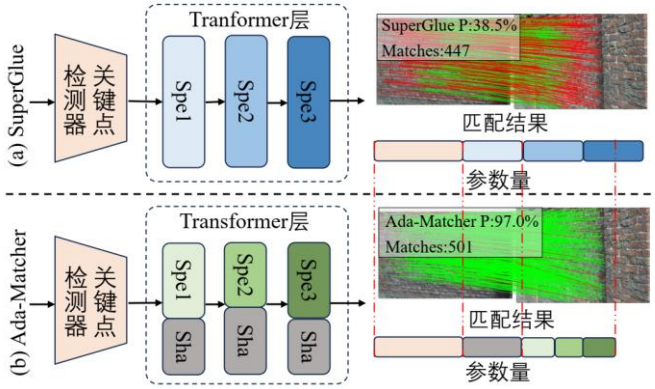


图 4-2 Ada-Matcher 与 SuperGlue 的对比

Fig.4-2 Comparison between Ada-Matcher and SuperGlue

4.2.2.1 Ada-Matcher 总体结构设计

人类在匹配图像时会反复观察图像，观察的次数越多，匹配结果就越精确。借助这一观点，Ada-Matcher 构建了一个深层 Tranformer 特征匹配网络，如图 4-3 所示。给定两个输入图像，Ada-Matcher 使用特征检测器提取关键点和特征描述符，将它们联合描述为局部特征。随后，为了增强这些局部特征的特异性，利用

关键点编码器将关键点和描述符映射到统一的向量中。接下来，该向量被输入到本研究设计的深层 Transformer 架构中，用于图像内和图像间上下文特征聚合。具体来说，在 Transformer 层中，Ada-Matcher 提出了一种掩码注意力技术(Mask-Attention Technology, MAT)，通过计算稀疏掩码注意力矩阵来减少冗余信息交换。随后，它连接前馈网络以提取聚合特征。此外，在每层内应用特征变换(Feature Transformation, FT)以增强特征多样性并提高特征的表达能力。最重要的是本文利用自适应权重共享机制(Adaptive Weight Sharing Mechanism, AWSM)，允许连续 Transformer 层之间动态共享权重，从而降低模型的存储开销。最后，集成最佳匹配层，该层根据聚合特征计算得分矩阵，在图像之间建立准确可靠的匹配。

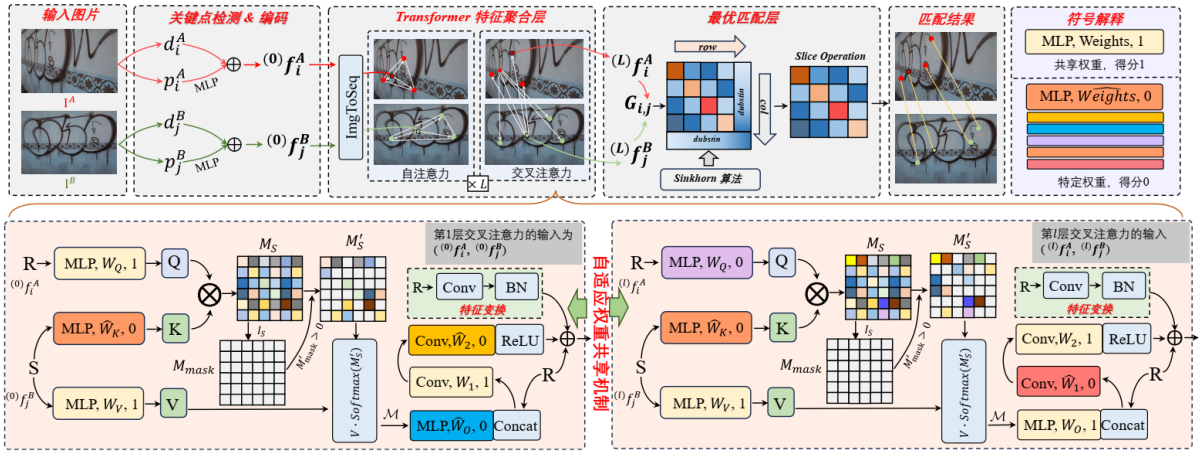


图 4-3 Ada-Matcher 网络结构示意图

Fig.4-3 Ada-Matcher network structure diagram

4.2.2.2 特征检测与编码

在 Ada-Matcher 中，关键点检测器负责从输入的图像中提取关键点和描述符。对于任意一对图像 (I^A, I^B)，每张图像包含一组关键点位置信息 d 和关描述符 p，将它们联合表示为局部特征 (p,d)。对于 I^A，可以将位置信息和描述符联合表示为局部特征 $f_i^A=\{(p_i^A,d_i^A)\,|\,i=1,2,3,...,m\}$，同理图像 I^B 的局部特征可以表示为 $f_j^B=\{(p_j^B,d_j^B)\,|\,j=1,2,3,...,n\}$。

关键点编码器进一步处理这些特征，通过结合视觉描述符与关键点的空间位置，提升特征的区分能力。Ada-Matcher 集成了一个多层感知机(Multilayer Perceptron, MLP)，专门用于将关键点的位置信息编码为高维向量，并将这些位置向量与相应的视觉描述符结合，从而增强其特异性。编码过程可以表达为：

$$^{(0)}f_i = p_i + \mathrm{MLP}(d_i) \tag{4-3}$$

其中 $^{(0)}f_i \in \boldsymbol{R}^D$ 表示生成的融合外观和位置信息的初始局部特征。该特征让深层特

征聚合网络能够更有效地整合图像内外的特征信息，使 Ada-MAtcher 在特征匹配中可以更准确地处理复杂的视觉信息。

4.2.2.3 深层 Transformer 特征聚合网络

在局部特征匹配任务中，除了聚合视觉描述符和关键点位置信息外，远程上下文信息的结合进一步放大了其独特性。Transformer 在整合全局上下文方面表现出了令人印象深刻的能力，借助全局感受野，Transformer 不仅可以整合全局信息，还可以有效消除图像块之间视觉相似性造成的歧义。然而，在标准 Transformer 中冗余信息的过多交互阻碍了其在特征匹配任务中的实际适用性。为了解决这个问题，本文引入了掩码注意力技术，它使用掩模来过滤信息，并将其与前馈网络相结合以提取聚合特征。

(1) 掩码注意力技术。在构建的 Transformer 架构中，每个 Transformer 层由多头注意力层、归一化层和前馈网络组成。Transformer 层的输入特征，表示为 R 和 S。给定 R 和 S，通过线性投影生成查询、键和值矩阵，表示为 Q,K,V。过程如下所示：

$$Q = RW_Q + b_Q, K = SW_K + b_K, V = SW_V + b_V \tag{4-4}$$

其中 W_Q,W_K,W_V 是由三个 MLP 实现的可学习参数。之后计算查询 Q 和键 K 之间的点积注意力分数：

$$M_S = \frac{QK^{\mathrm{T}}}{\sqrt{d}} \tag{4-5}$$

其中 d 是每个注意力头中查询向量的维度。尽管利用注意力矩阵可以从值向量中检索有价值的上下文信息并促进信息集成，但同时也引入了冗余信息的传播，对特征匹配的性能产生负面影响。为此，MAT 构建了一个掩码稀疏注意力矩阵，选择性地关注关键信息，最大限度地减少冗余数据的干扰。具体来说，首先初始化掩码矩阵 $M_{mask} \in R^{C\times C}$，与得分矩阵 M_S 具有相同的维度。随后，创建索引 I_S 以从分数矩阵中选择前 k 个条目，这些条目将在后续屏蔽过程中使用。该步骤如下：

$$M_{mask} \leftarrow 0, I_S = \mathrm{Topk}(M_S, k) \tag{4-6}$$

M_{mask} 被初始化为全零矩阵。$\mathrm{Topk}(\cdot)$ 操作选择注意力矩阵中 $k = 0.75C$ 个得分最高的条目，C 是注意力矩阵的维度。根据获得的索引，将掩码矩阵中的相应位置设置为 1，以将这些分数标识为重要的。然后，构建修正得分矩阵 M'_s。在此矩阵中，由掩码标记的分数保持不变，而所有其他分数设置为负无穷大。这样，未选择的分数不会对后续操作产生影响，确保模型只关注重要信息。其过程如下所示。

$$M'_{mask} \leftarrow (I_S, 1), M'_s = \mathrm{where}(M'_{mask} > 0, M_S, -\infty) \tag{4-7}$$

经过筛选和处理后，得到稀疏注意力得分矩阵 M_s'，用于从值向量 V 中检索信息，该步骤如下：

$$\bar{M} = V \cdot \text{Softmax}(M_S') \tag{4-8}$$

其中 $\bar{M}$ 是根据索引和值矩阵 V 计算得到的全局传输信息。接着，从 $\bar{M}$ 中检索信息，得到最终的全局信息 M。该过程可以表示为式(4.9)：

$$M = \frac{1}{3}\sum_{i}^{k} \bar{M}_i \tag{4-9}$$

此外，另一个 MLP 被用于检索消息：$M' = MW_O$。通过多头注意力层进行特征聚合后，使用前馈网络提取深度聚合特征。前馈神经网络由两个线性变换层与一个非线性激活函数组成，其表达式为：

$$\hat{M} = \gamma([M' \| R]W_1)W_2 \tag{4-10}$$

W_1 和 W_2 是两个线性层的参数矩阵，$[\|]$ 是沿通道维度的串联操作，γ 表示非线性激活函数 ReLU。

(2) 自适应权重共享机制。Ada-Matcher 借鉴 SuperGlue 将注意力图神经网络作为核心架构。该架构是一个综合的多重图，其中节点是来自两个图像的关键点，并且存在两种类型的无向边：自注意边和交叉注意边。具体来说，包含自注意力层的 Transformer 用于聚合图像内的远程特征，而具有交叉注意力层的 Transformer 则有助于图像之间的远程特征聚合。对于具有自注意力层，输入特征 R 和 S 相同（${}^{(l)}f_i^A$ 或 ${}^{(l)}f_j^B$），对于具有交叉注意层，输入特征 R 和 S 来自两个图像（$R = {}^{(l)}f_i^A, S = {}^{(l)}f_j^B$ 或 $R = {}^{(l)}f_j^B, S = {}^{(l)}f_i^A$），具体的表达式为：

$$\begin{aligned}
\hat{R} &= \text{SA}(R, R; \theta) + R \\
\hat{S} &= \text{SA}(S, S; \theta) + S \\
\hat{R} &= \text{CA}(R, S; \theta) + R \\
\hat{S} &= \text{CA}(S, R; \theta) + S
\end{aligned} \tag{4-11}$$

$\hat{R}$ 和 $\hat{S}$ 是更新后的特征，SA(·)是图像内的特征聚合操作，CA(·)是图像间的特征聚合操作。θ 表示 Transformer 层的权重。可以将等式(4-11)表示为输入特征 ${}^{(l)}f_i^A$ 和 ${}^{(l)}f_j^B$ 的完整 Transformer 层，第 l 层的特征聚合过程可以由式(4-12)表示：

$$\begin{aligned}
{}^{(l)}f_i^A &= \text{Trans}({}^{(l)}f_i^A, {}^{(l)}f_i^A; \theta) \\
{}^{(l)}f_j^B &= \text{Trans}({}^{(l)}f_j^B, {}^{(l)}f_j^B; \theta) \\
{}^{(l+1)}f_i^A &= \text{Trans}({}^{(l)}f_i^A, {}^{(l)}f_j^B; \theta) \\
{}^{(l+1)}f_j^B &= \text{Trans}({}^{(l)}f_j^B, {}^{(l)}f_i^A; \theta)
\end{aligned} \tag{4-12}$$

最终堆叠 L 层可以获得增强特征 ${}^{(L)}f_i^A, {}^{(L)}f_j^B$。其中 $\theta = \{W_Q, W_K, W_V, W_O, W_1, W_2\}$。

随着 Transformer 层的堆叠，导致匹配器在有限的资源下难以有效运行。鉴于参数集θ涵盖了大部分模型参数，传统方法是在 Transformer 层之间共享θ的每个参数，这可以有效降低模型规模。然而，这种方法限制了模型对不同输入数据关系的模拟能力，从而损害其捕获信息和学习层次结构的能力。为此，本文没有采用固定的参数共享范式，而是提出了一种自适应权重共享机制。采用该策略在连续的 Transformer 层之间动态共享参数，并且在每次参数共享后进行特征转换。过程如下：

$$^{(l+1)}f_i^A, {}^{(l+1)}f_j^B = \text{Trans}({}^{(l)}f_i^A, {}^{(l)}f_j^B; \hat{\theta}, \tilde{\theta}_l) \tag{4-13}$$

其中$\hat{\theta}$表示共享权重，$\tilde{\theta}_l$表示特定权重。从技术上讲，对于 L 个单独的 Transformer 层，开发了三组参数用于自适应权重共享，即共享的任务参数：$\hat{\theta}=\{\hat{W}_Q,\hat{W}_K,\hat{W}_V,\hat{W}_O,\hat{W}_1,\hat{W}_2\}$，特定的任务参数：$\{\tilde{\theta}_l\}_l^L=\{\tilde{W}_Q^l,\tilde{W}_K^l,\tilde{W}_V^l,\tilde{W}_O^l,\tilde{W}_1^l,\tilde{W}_2^l\}$，可学习的得分：$\{S_l\}_l^L=\{S_Q^l,S_K^l,S_V^l,S_O^l,S_1^l,S_2^l\}$。如图 4-4 所示。通过建立一个指示函数来确定 Transformer 参数在当前迭代中是否共享，具体如下所示，：

$$\Pi(S)=\begin{cases}1 & if\ S>=\lambda\\ 0 & \text{其他}\end{cases} \tag{4-14}$$

其中λ表示预设的阈值。例如，对于第l次迭代任务种的权重W_Q^l可以表示为：

$$W_Q^l=\Pi(S_Q^l)\hat{W}_Q+(1-\Pi(S_Q^l))\tilde{W}_Q^l \tag{4-15}$$

如果得分S_Q^l大于预设阈值λ，则将指示函数设置为 1，指示激活共享权重参数$\hat{W}_Q$。否则，特定任务权重$\tilde{W}_Q^l$被激活。上述过程应用于θ中的所有权重参数，动态地实现参数共享。

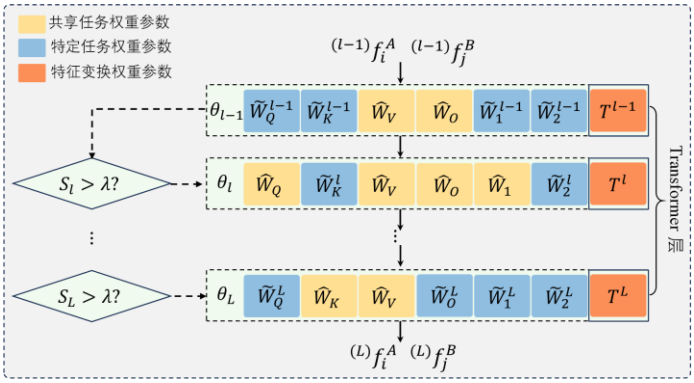


图 4-4 自适应权重共享算法

Fig.4-4 Adaptive Weight Sharing Algorithm

(3) 特征变换。为了应对 Transformer 层增加而带来的特征崩溃问题并增强特征多样性，Ada-Matcher 采用了一种改进的特征变换技术，通过传统的残差连接和

额外的参数化投影，不仅强化了各层之间输入与输出的直接联系，还增加了额外的参数空间，以适应更为复杂的数据模式。因此，特征聚合方程(4.11)可以重新表示为：

$$\begin{aligned}\hat{R} &= \mathrm{SA}(R,R;\theta)+R+R\mathcal{T}_l \\ \hat{S} &= \mathrm{SA}(S,S;\theta)+S+S\mathcal{T}_l \\ \hat{R} &= \mathrm{CA}(R,S;\theta)+R+R\mathcal{T}_l \\ \hat{S} &= \mathrm{CA}(S,R;\theta)+S+S\mathcal{T}_l\end{aligned} \tag{4-16}$$

其中$\mathcal{T}_l$是第l次迭代时的卷积层和归一化层的序列。通过这种特征增强操作，每次更新网络后特征的多样性都会不同，模型的表达能力得到进一步增强。

4.2.2.4 最优匹配层

一个有效的特征匹配方法应该既能够找到特征之间的正确匹配，同时识别并排除错误的匹配。作为最后的匹配模块，遵循 SuperGlue 的方法，最优匹配层主要用于生成部分软分配矩阵$P\in[0,1]^{m\times n}$。对于一般的图匹配过程，可以通过在约束条件下计算得分矩阵$G\in\boldsymbol{R}^{m\times n}$并通过最大化得分获得部分分配矩阵$P$。$G$的表达式：

$$G_{i,j}=<{}^{(L)}f_i^A,{}^{(L)}f_j^B> \tag{4-17}$$

其中$<\cdot>$为内积，${}^{(L)}f_i^A,{}^{(L)}f_j^B$表示经过$L$次特征聚合的局部特征。此外，还引入了“垃圾箱”来容纳无法匹配的关键点。因此，部分软分配矩阵表示为：$\hat{P}\in[0,1]^{(m+1)\times(n+1)}$。图像$I^A$中的每个关键点都被分配给图像$I^B$中的相应关键点或垃圾箱。该矩阵受以下约束：

$$\hat{P}1_{n+1}=[1_m\ n]\ ,\quad \hat{P}^{\mathrm{T}}1_{m+1}=[1_n\ m] \tag{4-18}$$

其中1_m是长度为m的全 1 向量。一般来说，图像I^A中的每个特征点在图像I^B中只有一个匹配点，但是对于图像I^A的垃圾箱来说，它可能与图像I^B中的任意特征点匹配，即有n种可能性。具体地，在式(4.20)指定的约束下，利用函数F可以得到$\hat{P}$。F的表达式为：

$$F=argmax\sum_{i=1}^{m+1}\sum_{j=1}^{n+1}\hat{G}_{i,j}\hat{P}_{i,j} \tag{4-19}$$

将得分矩阵$G_{i,j}$添加一行一列，得到$\hat{G}_{i,j}\in\boldsymbol{R}^{(m+1)\times(n+1)}$。等式(4-19)旨在最大化总体分数，可以使用“Sinkhorn 算法”在 GPU 上有效求解，该算法包括沿行和列迭代归一化。经过多次迭代，删除添加的垃圾箱，恢复软分配矩阵$P=\hat{P}_{1:m,1:n}$。

4.2.3 基于 PnP 的机器人位姿求解

在机器人视觉定位过程中，机器人采集的查询图像 I_A 与数据库中检索到的最相似的 K 张检索图像 I_B 通过 Ada-Matcher 进行局部特征匹配，从匹配结果中选择效果最好的检索图像。之后，根据检索图像的 2D 像素点 P_B^f 与 3D 路标点 P_w 间的对应关系，进一步推导查询图像的 2D 像素点 P_A^f 与 3D 路标点间的对应关系。最后，利用结合 RANSAC 的 PnP 算法求解机器人位姿，整个流程如图 4-5 所示。

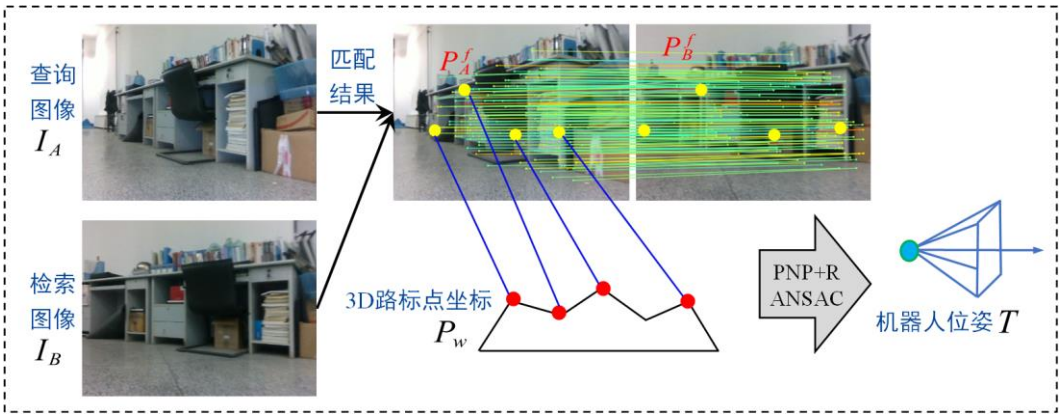


图 4-5 基于 PnP 的机器人位姿求解流程

Fig.4-5 Robot pose solution process based on PnP

4.3 实验结果与分析

4.3.1 单应性估计

精确评估匹配的质量不仅仅涉及在图像之间建立可靠的几何关系，匹配的分布和密度同样至关重要，本文进行了一项专注于单应性估计的实验来评估 AdaMatcher 的性能。实验使用了 HPathces 数据集[66]进行单应性估计测试，并选取其中的 56 个视点变化的序列和 52 个光照变化的序列来进行评估。评估指标采用 Patch2Pix[67]中提出的配置和平均角度误差，即 1/3/5 像素内正确估计的单应性的占比。此外，本文还使用与 SuperGlue 相同的指标来计算精度(Precision)，召回率(Recall)。

如表 4-1 所示，AdaMatcher 在基于两种检测技术的特征匹配方法中均展现了卓越的性能。具体来说，当使用 SIFT 作为特征检测器时，在 1 和 3 像素阈值下，Ada-Matcher 与基线方法相比性能显著增强，这突显了该方法在匹配精细尺度和细微特征方面卓越的准确性。采用 SuperPoint 作为特征检测器时，不同光照条件下，Ada-Matcher 在 1、3 和 5 像素阈值下的性能提升分别为 9%、2%和 2%，均优

于基线方法。同样，在极端视角变化的条件下，与基线 SuperGlue 相比，Ada-Matcher 表现出 1%，3%和 1%的优势。整体变化条件下，Ada-Matcher 仍然具有很强的竞争力，特别是在 1 像素的低阈值下进步显著，提升了 5%。同时在 3 和 5 像素阈值下的性能与当前最先进的方法不相上下，分别提升了 2%，2%。这进一步证实了 AdaMatcher 在各种阈值和复杂场景下的鲁棒性。如图 4-6 所示，从 HPatches 数据集中选择了四对图像，分别包含光照和视点的变化。Ada-Matcher 和 SuperGlue 的匹配结果进一步证实了 Ada-Matcher 的鲁棒性。

表 4-1 单应性估计实验

Table 4-1 Homography estimation experiment

局部特征	匹配方法	整体	光照变换	视角变化
		精度(ε<1/3/5 像素)		
SIFT	SuperGlue	0.43/0.77/0.88	0.52/0.86/0.95	0.34/0.69/0.81
	Ada-Matcher	0.46/0.79/0.87	0.57/0.89/0.94	0.37/0.69/0.81
SuperPoint	NN	0.46/0.78/0.85	0.57/0.92/0.97	0.35/0.65/0.74
	SuperGlue	0.52/0.83/0.89	0.62/0.93/0.97	0.45/0.73/0.83
	SGMNet	0.52/0.85/0.91	0.59/0.94/0.98	0.46/0.74/0.84
	ClusterGNN	0.52/0.84/0.90	0.61/0.93/0.98	0.44/0.74/0.81
	LightGlue	0.53/0.84/0.89	0.61/0.94/0.98	0.46/0.73/0.83
	GlueStick	0.52/0.83/0.90	0.60/0.93/0.98	0.45/0.74/0.81
	Ada-Matcher	0.58/0.85/0.91	0.71/0.95/0.99	0.46/0.76/0.84

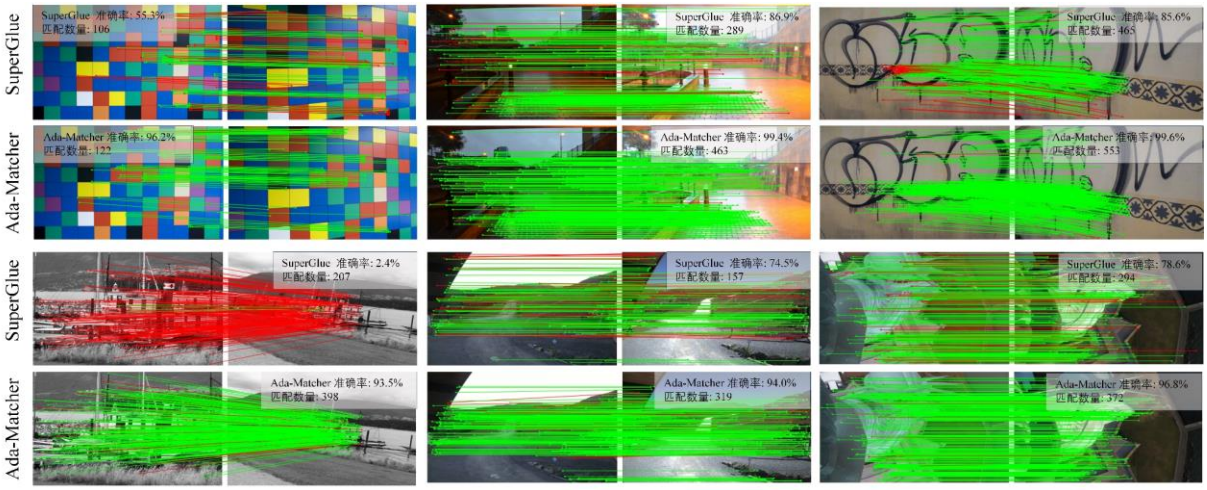


图 4-6 SuperGlue 和 Ada-Matcher 在 HPatches 数据集上的可视化

Fig.4-6 Visualization of SuperGlue and Ada-Matcher on the HPatches dataset

表 4-2 通过精度和召回率证明 Ada-Matcher 在单应性估计中具有显著的优势。首先，SuperGlue*的精度和召回率分别为 66.62%和 67.17%。Ada-Matcher 则达到了 85.83%的精度和 86.40%的召回率，分别比基线提升了 19.21%和 19.23%。此外，

Ada-Matcher 在角度误差方面同样表现出色，在 3、5 和 10 像素阈值下分别超出基线 11.8%、13.62%和 14.43%。

表 4-2 不同像素阈值下角度误差以及精度和召回率

Table 4-2 Angle error, precision and recall rate under different pixel thresholds

局部特征	匹配方法	AUC			精度	召回率
		3 像素	5 像素	10 像素		
SuperPoint	SuperGlue	32.11	45.44	61.33	66.62	67.17
	Ada-Matcher	43.91	59.06	75.76	85.83	86.40

4.3.2 图像匹配

除了单应性估计之外，图像匹配在各种基本视觉任务中也具有至关重要的意义。因此，本文对图像匹配进行了实验评估，以验证 Ada-Matcher 的有效性。Ada-Matcher 使用 HPatches 数据集进行评估，并采用平均匹配精度(Mean Matching Accuracy MMA)作为主要性能指标，该指标量化了每对图像中识别出的正确对应关系的平均百分比，从而提供了一个直观的标准。结果如图 4-7 所示。

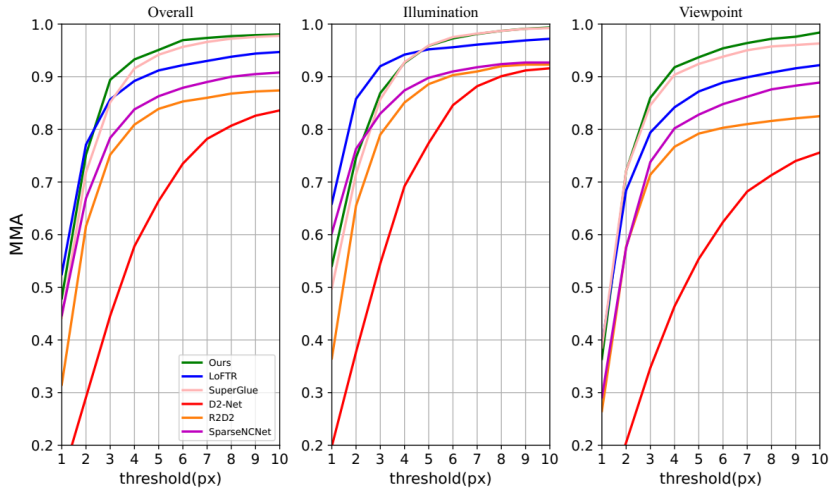


图 4-7 HPatches 数据集上的图像匹配实验

Fig.4-7 Image matching experiment on the HPatches dataset

将 Ada-Matcher 与当前主流图像匹配方法进行了对比，包括无检测器方法(如 LoFTR)和基于检测器的方法(如 SuprGlue)。具体来说，从整体表现来看，Ada-Matcher 在多个精度阈值下均展现出了卓越的性能，展现了其面对复杂环境的可靠性和稳定性。在光照变化条件下，Ada-Matcher 表现出处理细微特征匹配的能力，在低阈值下匹配精度显著高于基线 SuperGlue。当处理具有显著视角变化时，Ada-Matcher 的性能优势尤为突出。在此类复杂条件下，Ada-Matcher 凭借深层特

征匹配结构的设计，有效应对了特征崩溃和冗余干扰问题，实现了更高的匹配精度和鲁棒性。

4.3.3 位姿估计

真实环境的复杂性给局部特征匹配带来了巨大的挑战，包括光照的变化、天气条件的波动和多样化的场景。为验证 Ada-Matcher 在面临这些艰巨挑战时的性能，进行了室外场景中的相对姿势估计实验。该实验采用 MegaDepth 数据集[68]，该数据集包含 196 个场景中捕获的 100 万张互联网图像，每个图像都附有通过 COLMAP 生成的相应稀疏 3D 点云。在本实验评估中，使用曲线下面积(AUC)来评估不同阈值$(5^{\circ},10^{\circ},20^{\circ})$下的姿态误差。

如表 4-3 所示，Ada-Matcher 采用 SIFT 和 SuperPoint 两种局部特征检测和描述方法，均表现了出色的性能。具体来说，当使用 SIFT 作为特征检测器时，基线 SuperGlue 在 AUC$(5^{\circ},10^{\circ},20^{\circ})$阈值下的位姿估计精度为 36.2%，52.30%和 66.50%，Ada-Matcher 在相同评价指标下提高了 2.2%，2.4%和 2.1%。当使用 SuperPoint 进行特征检测时，AdaMatcher 的性能继续优于基线方法，精度为 44.53%、61.66%和 75.60%，显著高于基线的 41.31%，58.26%和 71.22%。此外，与 LightGlue 和 GlueStick 等最先进的方法相比，Ada-Matcher 也表现出了令人瞩目的位姿估计精度。图 4-8 直观地描述了 Ada-Matcher 和基线方法在 MegaDepth 数据集上的匹配性能，进一步证明了所提方法的有效性。

表 4-3 室外位姿估计实验

Table 4-3 Outdoor pose estimation experiment

局部特征	匹配方法	位姿估计(%)		
		5°	10°	20°
SIFT	SuperGlue	36.20	52.30	66.50
	Ada-Matcher	38.40	54.70	68.60
SuperPoint	NN	31.68	46.82	60.05
	SuperGlue	41.31	58.26	71.22
	SGMNet	40.50	59.00	73.60
	ClusterGNN	44.19	58.54	70.33
	LightGlue	42.94	60.89	73.13
	GlueStick	42.70	61.24	72.20
	Ada-Matcher	44.53	61.66	75.60

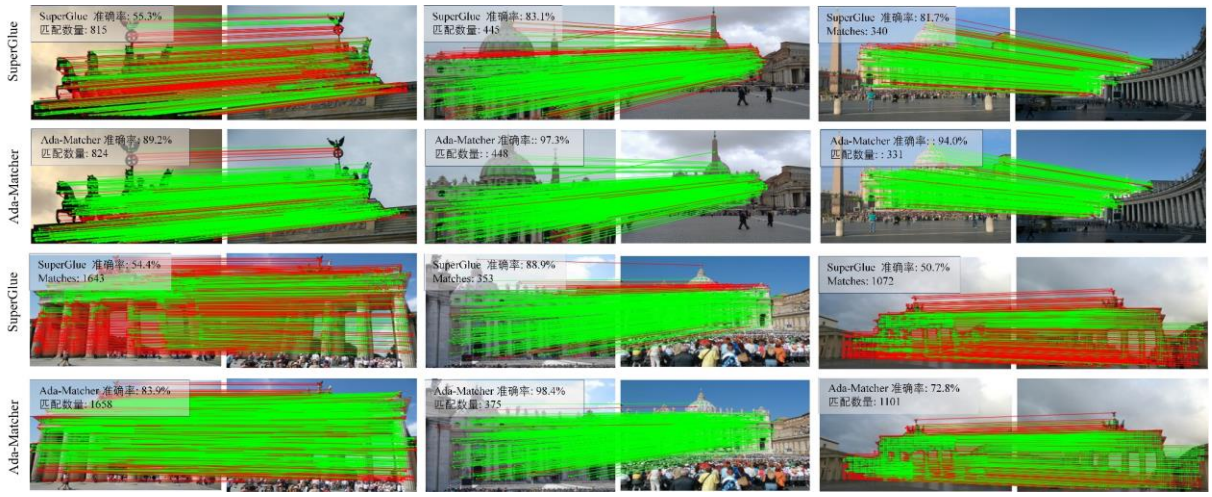


图 4-8 Ada-Matcher 和 SuperGlue 在 MegaDepth 数据集上的匹配结果

Fig. 4-8 Matching results of Ada-Matcher and SuperGlue on MegaDepth dataset

4.3.4 室外视觉定位

作为视觉定位的基础组成部分，特征匹配的性能直接影响定位系统的整体质量和稳定性。室外条件下视觉定位会面临多种的复杂情况，例如常见的昼夜、天气变化等，为了验证 Ada-Matcher 在室外环境下的视觉定位性能，进行了室外视觉定位实验。该实验采用 Aachen Day-Night 数据集[69]，它由 4328 张拍摄 Aachen 的图像以及 922 张手机摄像头拍摄的查询图像组成，其中包括 824 张白天图像和 98 张夜间图像，广泛应用于长期室外视觉定位评估[71]。将 Ada-Matcher 集成到官方 Hloc 视觉定位流程中，并根据官方基准报告不同阈值下的视觉定位精度。结果如表 4-4 所示。

表 4-4 室外视觉定位实验

Table 4-4 Outdoor visual positioning experiment

局部特征	匹配方法	白天(%)	夜晚(%)
		(0.25m, 2°)/(0.5m, 5°)/(1m, 10°)	
SuperPoint	NN	85.4/93.3/97.2	75.5/86.7/92.9
	SuperGlue	88.2/95.5/98.7	86.7/92.9/100
	SGMNet	86.8/94.2/97.7	83.7/91.8/99.0
	ClusterGNN	89.4/95.5/98.5	81.6/93.9/100
	LightGlue	89.2/ 95.4/98.5	87.8/93.9/100
	GlueStick	89.3/95.2/98.4	86.8/93.7/99.0
	Ada-Matcher	89.6/95.6/98.8	86.5/93.9/100

Ada-Matcher 在多个指标上均表现出了显著的优势，白天场景下，Ada-Matcher 在三个阈值条件下的定位精度分别达到了 89.6%，95.4%和 98.8%，与其他方法相比，保持了较高的精度。在夜晚场景下，Ada-Matcher 的定位精度为 86.5%，93.9%和 100%，同样表现出卓越的鲁棒性，尤其在最高阈值条件下达到了 100%的准确率，充分说明了其在复杂环境下的适应能力。相比其他方法，Ada-Matcher 的优越性尤其体现在跨场景的泛化性能上。例如，与 SuperGlue 和 LightGlue 等经典方法相比，Ada-Matcher 在白天场景中实现了更高的精度，同时在夜晚场景中的表现同样领先于大多数对比方法，充分证明了 Ada-Matcher 在动态场景和光照变化条件下的鲁棒性。

4.3.5 室内视觉定位

通常，室内视觉定位任务会受到遮挡，视角变换和低纹理区域等的影响。为了验证 Ada-Matcher 在室内视觉定位方面的有效性，进行了相关的实验分析。本实验在 InLoc 数据集[70]上进行，该数据集包含有遮挡、无纹理场景以及视点和照明变化的图像，已被广泛应用于室内视觉定位和导航研究。实验参数与官方 Hloc 方法保持一致，将 Ada-Matcher 集成到该框架中，并在长期视觉定位[71]基准上评估其性能。如表 4-5 所示，统计定位误差低于预定义阈值的图像比例。

表 4-5 室内视觉定位实验

Table 4-5 Indoor visual positioning experiment

局部特征	匹配方法	DUC1 场景(%)	DUC2 场景(%)
		(0.25m, 10°)/(0.5m, 10°)/(1m, 10°)	
SuperPoint	NN	40.4/58.1/69.7	42.0/58.8/69.5
	SuperGlue	49.0/68.7/80.8	53.4/77.1/82.4
	SGMNet	41.9/64.1/73.7	39.7/62.6/67.2
	ClusterGNN	47.5/69.7/79.8	53.4/77.1/84.7
	LightGlue	49.0/68.2/79.3	55.0/74.8/79.4
	GlueStick	48.9/68.1/79.8	54.7/73.0/79.2
	Ada-Matcher	49.0/71.2/80.3	56.5/74.8/83.2

Ada-Matcher 在不同阈值条件下均表现出了显著的竞争力，与当前先进方法相比展现了卓越的定位性能。具体来说，在场景 DUC1 中，它在 $(0.25m,10^{o})$ 和 $(0.5m,10^{o})$ 阈值下获得了峰值定位精度，在 $(1m,10^{o})$ 的指标中，Ada-Matcher 继续展现出卓越的性能。在场景 DUC2 中，特别是在 $(0.25m,10^{o})$ 标准下，Ada-Matcher 超出了基线 3.1%。在其他阈值条件下，该方法仍然与当前最先进的方法相当，突出了其跨不同场景的鲁棒性和泛化能力。值得注意的是，像 SGMNet 这样的方法虽

然实现了高精度，但受到对初始种子匹配的依赖的限制，如果这些种子没有得到最佳选择，可能会导致效率低下。类似地，尽管 LightGlue 针对速度进行了优化，但它可能面临速度和准确性之间的权衡，特别是由于其早期停止和修剪机制，这可能导致在需要更广泛计算的场景中过早终止。这一系列结果间接表明更深的 Transformer 层和自适应权重共享策略的集成对于增强匹配性能至关重要。

4.3.6 效率分析

为了验证 Ada-Matcher 的效率，该实验将 Ada-Matcher 与基线方法和几种先进的无检测器方法在参数、GFLOPs 和推理速度方面进行比较，以确定其计算成本和存储消耗。实验在单个 NVIDIA RTX 3090 GPU 上进行，输入图像大小调整为 840×840，如表 4-6 所示。

表 4-6 效率分析实验

Table 4-6 Efficiency analysis experiment

匹配方法	参数量(MB)	GFLOPs	运行时间(s)
基于检测器的特征匹配方法			
SuperPoint+SuperGlue	18.2	291.88	0.144
SuperPoint+Ada-Matcher	12.0	212.17	0.150
无检测器的特征匹配方法			
LoFTR	11.06	729.54	0.176
QuadTree	13.21	748.17	0.296
MatchFormer	22.37	1087.95	0.826
ASpanFormer	15.05	1466.15	0.475
OAMatcher	15.29	885.64	0.252

Ada-Matcher 的参数量仅为 12MB，显著低于 SuperGlue 的 18.2MB，展示了其内存资源利用效率。就 GFLOPs 而言，Ada-Matcher 值为 212.17，比 SuperGlue 的 291.8 降低了 27.3%。虽然 Ada-Matcher 的推理速度为 0.150s，比 SuperGlue 稍慢，但差异很小。此外，与 LoFTR、MatchFormer 等无检测器方法相比，基于检测器的方法在 GFLOPs 和运行时间上都表现出显著的优势，证明了 Ada-Matcher 在需要高效计算的场景中具有强大的竞争力。这种优势的产生主要是因为无检测器方法直接处理全局特征和像素信息，进行全局上下文建模和像素级密集匹配，从而导致更高的计算复杂度和时间消耗。相比之下，基于检测器的方法通过专注于提取局部特征来显着减少计算量。此外，得益于自适应权重共享机制，Ada-Matcher 与列出的无检测器方法相比，在参数计数方面保持了强大的竞争力。

4.3.7 消融实验

在本节中，为了验证所提出的每个模块的有效性，本文对 Ada-Matcher 的不同变体进行了实验。如表 4-7 所示，所有模块都对 Ada-Matcher 的卓越性能做出了贡献。(i)代表重新训练的基线方法，以确保公平性。在 AUC(5°,10°,20°)下获得相对位姿估计精度 41.31%，58.26%和 71.22%，其参数大小为 18.2MB。(ii)仅使用掩码注意力，位姿估计精度提高到 42.12%，59.08%和 72.45%，表明 MAT 有效地关注关键的图像内/图像间信息，增强了模型的匹配性能。(iii)实施自适应权重共享，表中结果显示模型参数显着减少了 8MB，与基线相比减少了 45%。此外，模型的匹配精度较 SuperGlue 提高了 2.43%，2.61%和 2.9%。这证实了这种动态共享方法不仅减少了模型的占用空间，而且还保留了每一层的独特参数，增强了模型表达特征的能力。(iv)引入特征变换，从表中可以看出，在 AUC(5°,10°,20°)下，相对位姿估计精度有显着提高，从 43.74%,，60.87%和 74.12%提升到 44.53%，61.66%和 75.60%，其中模型参数总数仅增加 1.8MB。这突显了所提出的 FT 的重要性，有效缓解了网络深度过大导致的特征崩溃。

表 4-7 Ada-Matcher 消融实验

Table 4-7 Ada-Matcher ablation experiment

匹配方法	参数量(MB)	位姿估计精度(%)		
		5°	10°	20°
(i)基准方法	18.2	41.31	58.26	71.22
(ii)+掩码注意力	18.2	42.12	59.08	72.45
(iii)+自适应权重共享	10.2	43.74	60.87	74.12
(iv)+特征变换	12.0	44.53	61.66	75.60

掩码注意力消融分析。为了验证掩膜覆盖率(Mask Coverage Ratios, MCR)对模型性能的影响，如表 4-8 所示，分别使用了 0.25C、0.50C 和 0.75C 下的不同掩模覆盖率进行比较实验，其中 C 表示序列长度。在 MCR 为 0.25C 时，该模型在 AUC(5°,10°,20°)阈值下表现出最低的位姿估计精度，分别为 43.45%，60.54%和 73.88%，这表明覆盖率过低会导致捕获过多冗余信息带来不必要的噪声，从而影响整体性能，随着 MCR 增加到 0.50C，位姿估计精度有所上升也证明了这一观点。当掩膜覆盖率增加至 0.75C 时，AUC(5°,10°,20°)阈值下精度达到最高，分别为 44.53%、61.66%和 75.60%。这表明合理的覆盖率使模型能够跟有效地捕获基本特征，排除冗余信息干扰，平衡计算效率和模型性能，从而提高预测准确性。

表 4-8 掩模覆盖率消融实验

Table 4-8 Mask coverage ablation experiment

掩膜覆盖率	位姿估计精度(%)		
	5°	10°	20°
0.25C	43.45	60.54	73.88
0.50C	44.11	61.35	74.97
0.75C	44.53	61.66	75.60

自适应权重共享机制消融分析。AWSM 通过在相邻 Transformer 层之间动态共享权重，减少存储消耗，同时保留每层的独特权重以增强特征表示。为了验证 AWSM 设计的有效性，使用固定权重共享的 Ada-Matcher 变体进行了对比实验，如表 4-9 所示。使用 AWSM 策略的变体的总参数数为 12.0MB，略高于使用固定权重共享策略的 10.1MB 用。然而，在 AUC$(5^{\circ},10^{\circ},20^{\circ})$阈值下，位姿估计精度明显提高。这表明 AWSM 策略虽然增加了参数数量，但其动态权重共享机制保留了更多的独立权重，有效增强了模型的特征表示能力，提高了位姿估计精度。这种权重共享机制有效地平衡了模型复杂度和性能，特别是在资源受限的应用场景中。

表 4-9 不同权重共享策略的消融实验

Table 4-9 Ablation experiments of different weight sharing strategies

共享策略	参数量(MB)	位姿估计精度(%)		
		5°	10°	20°
固定权重共享	10.1	43.56	60.65	73.90
自适应权重共享	12.0	44.53	61.66	75.60

为了探究更多层共享权重对性能的影响，本文对 AWSM 的各种变体进行了分析，如表 4-10 所示，尽管三层和四层共享策略在参数减少方面具有一定优势，但两层共享策略在所有阈值下都表现出了最佳位姿估计精度，分别达到 44.53%，61.66%和 75.60%。这表明，两层共享策略在保持层之间较高程度的独立性的同时，有效地利用了权重共享的优势。当共享层数增加时，模型可能会丧失一些捕捉细节的能力，这不利于学习复杂的特征，尤其是在需要高精度的应用中。

表 4-10 权重共享消融实验

Table 4-10 Weight sharing ablation experiment

共享策略	参数量(MB)	位姿估计精度(%)		
		5°	10°	20°
共享三层	10.8	44.19	61.21	74.87
共享四层	10.1	43.75	60.84	74.12
共享两层	12.0	44.53	61.66	75.60

4.4 基于四足机器人的视觉重定位实验分析

为了验证基于深度特征匹配器的视觉重定位算法在四足机器人实际场景中的性能表现，进行了实际场景中的实验。实验平台采用文章 2.4.1 章节介绍的宇树科技 B1 四足机器人，视觉传感器使用 Realsense D455。实验通过采集测试集中的图像来确定对应四足机器人的 6D 位姿。场景 1 包含遮挡，重复纹理和视角变换等场景。如图 4-9 中所示，基于 Ada-Matcher 的视觉重定位算法(蓝色线条)更加接近于真实轨迹，在楼梯位置，Ada-Matcher 的匹配密度和质量远高于 SuperGlue。场景 2 包含无纹理或弱纹理等场景，如图 4.9 所示，基于 Ada-Matcher 的视觉重定位算法(蓝色线条)定位轨迹依然领先于 SuperGlue，在无纹理的走廊位置，Ada-Matcher 的匹配数量是 SuperGlue 的 3 倍以上，且更加准确。这强调了深层特征匹配方法在提升视觉重定位性能方面的重要性，突显了 Ada-Matcher 的显著优势。

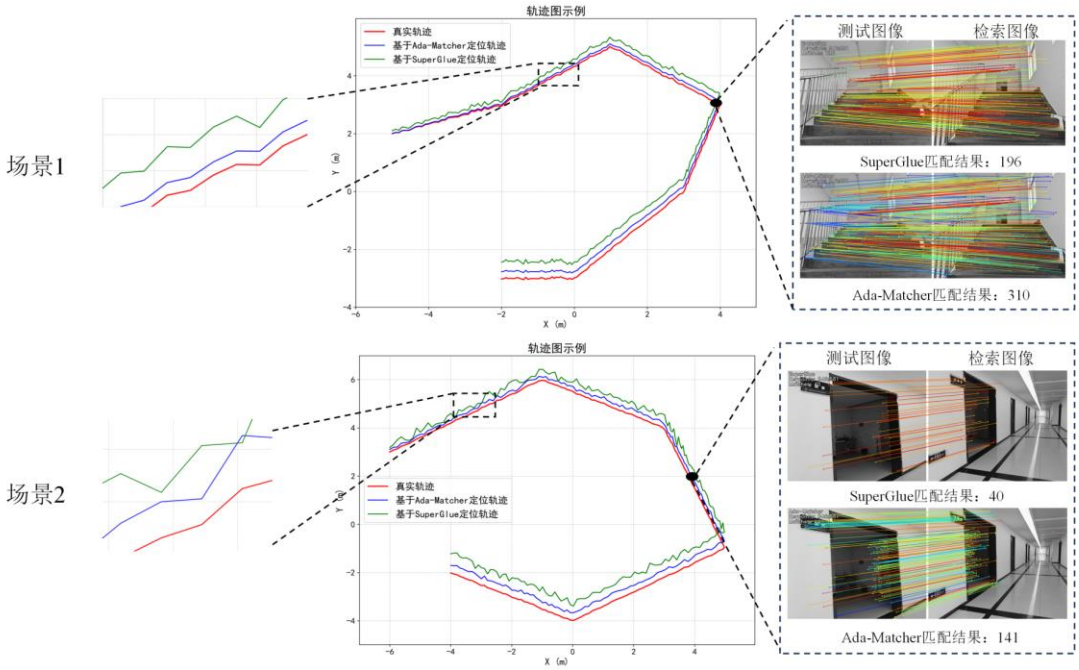


图 4-9 定位轨迹示意图

Fig.4-9 Positioning trajectory diagram

为了更加准确的评估视觉重定位算法性能，本实验还采用了测试图像预测位姿的平移误差中值 ΔT 和旋转误差中值 ΔR 来进行验证，如表 4-11 所示。基于 Ada-Matcher 的视觉重定位方法在各种复杂场景下定位精度均超越了基于 SuperGlue 的方法。相较于基线 SuperGlue，场景 1 中，基于 Ada-Matcher 的视觉重定位方法平移误差和旋转误差分别提升了 37.65%和 33.33%，场景 2 中则分别提高了 30.5%和 52.2%。这一对比再次证明了 Ada-Matcher 在提高视觉定位方面的显著优势。

表 4- 11 实际场景重定位实验

Table 4-11 Actual scene relocation experiment

场景序列	方法	ΔT(m)	ΔR(°)
场景 1	基于 Ada-Matcher 的视觉重定位算法 (本章算法)	0.053	0.14
场景 2	基于 Ada-Matcher 的视觉重定位算法 (本章算法)	0.114	0.44
场景 1	基于 SuperGlue 的视觉重定位算法	0.085	0.21
场景 2	基于 SuperGlue 的视觉重定位算法	0.164	0.92

此外，本文还绘制了两种算法在实际场景中的累计误差曲线来更加直观的展现所提方法的优势，如图 4-10 所示。在两个场景中，基于 Ada-Matcher 的视觉重定位方法的误差累计曲线(红色)始终高于基线 SuperGlue(蓝色)，证明了在相同指标下，本文的方法定位准确率更高，特别是在具有挑战的场景中，Ada-Matcher 可以实现更加精准的定位。

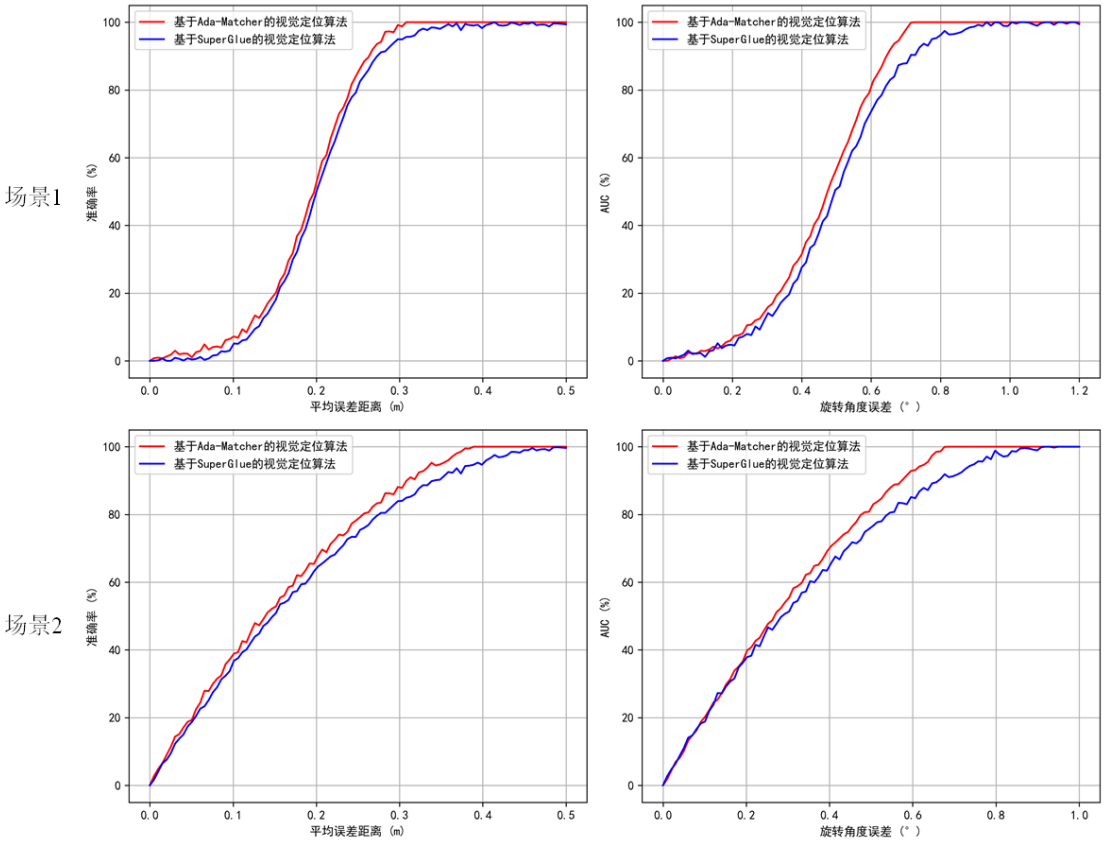


图 4-10 累计误差分布曲线

Fig.4-10 Cumulative error distribution curve

4.5 本章小结

本章介绍了一种基于深度特征匹配器的视觉重定位方法，核心是设计了基于 Transformer 的 Ada-Matcher 特征匹配网络，结合深层 Transformer 与自适应权重共享机制，增强匹配鲁棒性。实验证明，在单应性估计、室内外定位任务中，基于 Ada-Matcher 的定位方法精度与效率均优于其他先进方法，解决了四足机器人因视角变化与动态遮挡导致的定位难题，证明了基于深层特征匹配器的视觉定位方法的有效性。

第 5 章 总结与展望

5.1 研究总结

随着科技进步和社会需求增长，四足机器人凭借其卓越的适应性和灵活性，在工业巡检、物流运输、军事等领域展现出显著的使用价值。然而，由于四足机器人运动特性复杂，以及任务环境多变，如何确保其在动态场景中准确感知环境并确定自身位置，仍是一项重要挑战。本文围绕四足机器人在复杂动态环境下的定位与视觉感知问题展开研究，针对传统视觉 SLAM 和重定位方法在动态场景中的不足，提出了两种创新算法，并通过理论分析，算法设计及实验验证，显著提升了四足机器人在动态环境中的导航与自主探索能力。具体研究如下：

(1) 提出了一种基于动态场景分割的视觉 SLAM 算法 DSS-SLAM，采用全景分割技术生成动态掩码，并利用基础矩阵实现对掩码的优化与目标分类；结合光流估计和深度辅助验证，区分物体的真实运动和相机运动造成的表观运动；最后，利用动态特征点过滤算法有效剔除干扰特征点，优化了位姿估计的精度与鲁棒性。实验在 TUM、KITTI 和 EuRoC 数据集上表明，该算法无论是在高动态场景还是低动态场景中，相较于 ORB-SLAM3 等传统方法具有明显优势。在真实四足机器人平台测试中，DSS-SLAM 有效应对了动态目标，视角变换和动态遮挡等问题，确保了系统的鲁棒性和稳定性。

(2) 提出了一种基于深度特征匹配器的视角重定位方法。本研究的核心是基于自适应权重共享的特征匹配器 Ada-Matcher，通过构建深层 Transformer 架构，结合掩码注意力、特征增强和自适应权重共享技术，该方法在显著提高特征匹配的鲁棒性与精度的同时，有效降低模型规模。在 HPatches、MegaDepth 等数据集的测试中，该方法在光照变化、视角变换等复杂条件下，对于位姿估计和视觉定位等任务均表现出色。通过离线与在线的视觉重定位框架，基于四足机器人平台的实验进一步验证了其在实际环境中的适用性，轨迹误差显著降低，重定位精度得到提升。

5.2 未来展望

本文所提出的四足机器人定位及视觉感知算法在多个数据集以及真实环境中取得了一定的进步，证明了算法的有效性。但仍有一定的改进空间，未来可以深入研究：

(1) 提出的动态视觉 SLAM 尽管算法鲁棒性提升，但新增的分割与运动分析

增加了计算负担，导致帧率低于传统方法。未来可探索轻量化网络设计或硬件加速技术，在保持精度的同时提升实时性能。

(2) 当前研究主要依赖视觉传感器，而单一传感器在面对极端条件下显现出局限性。随着多传感器技术的发展，未来可融合 IMU、激光雷达通过多源信息互补增强算法的适应性和可靠性。

参考文献

[1] Buehler M, Playter R, Raibert M. Robots step outside[C]. Int. Symp. Adaptive Motion of Animals and Machines (AMAM), Ilmenau, Germany, 2005: 1-4.

[2] 刘京运.从 Big Dog 到 Spot Mini:波士顿动力四足机器人进化史[J].机器人产业,2018,(02):109-116.

[3] Bledt G, Powell M J, Katz B, et al. Mit cheetah 3: Design and control of a robust, dynamic quadruped robot[C]. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). IEEE, 2018: 2245-2252.

[4] 张玮奇,王嘉,张琳,等.SUI-SLAM:一种面向室内动态环境的融合语义和不确定度的视觉 SLAM 方法[J].机器人,2024,46(06):732-742.

[5] 李锋,薛梅,詹勇,等.基于分割一切模型 SAM 的实景三维场景语义分割[J].测绘通报,2024,(12):101-105.

[6] 李军侠,苏京峰,崔滢,等.基于语义调制的弱监督语义分割[J/OL].软件学报,2025,1-15.

[7] 徐淑萍,卫浩波,孙洋洋,等.基于模板更新和重检测的长时目标跟踪研究[J].计算机工程与科学,2024,46(12):2196-2204.

[8] 陈吉清,车宇翔,田小强,等.动态场景下基于 3D 多目标追踪的实时视觉 SLAM 方法研究[J].汽车工程,2024,46(05):776-783.

[9] Klein G, Murray D. Parallel tracking and mapping for small AR workspaces[C]. 2007 6th IEEE and ACM international symposium on mixed and augmented reality. IEEE, 2007: 225-234.

[10]Engel J, Schöps T, Cremers D. LSD-SLAM: Large-scale direct monocular SLAM[C]. European conference on computer vision. Cham: Springer International Publishing, 2014: 834-849.

[11]赵玏洋,闫利.移动机器人 SLAM 位姿估计的改进四元数无迹卡尔曼滤波[J].测绘学报,2022,51(02):212-223.

[12]Xie T, Sun Q, Sun T, et al. DVDS: A deep visual dynamic slam system[J]. Expert Systems with Applications, 2025, 260: 125438.

[13]Bescos B, Fácil 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.

[14]Bescos B, Campos C, Tardós J D, et al. DynaSLAM II: Tightly-coupled multi-object tracking and SLAM[J]. IEEE robotics and automation letters, 2021, 6(3): 5191-5198.

[15]Smith R C, Cheeseman P. On the representation and estimation of spatial uncertainty[J]. The international journal of Robotics Research, 1986, 5(4): 56-68.

[16]Davison A J, Reid I D, Molton N D, et al. MonoSLAM: Real-time single camera SLAM[J]. IEEE transactions on pattern analysis and machine intelligence, 2007, 29(6): 1052-1067.

[17]Mur-Artal R, Montiel J M M, Tardos J D. ORB-SLAM: a versatile and accurate monocular SLAM system[J]. IEEE transactions on robotics, 2015, 31(5): 1147-1163.

[18]Mur-Artal R, Tardós J D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras[J]. IEEE transactions on robotics, 2017, 33(5): 1255-1262.

[19]Campos C, Elvira R, Rodríguez J J G, et al. Orb-slam3: An accurate open-source library for visual, visual–inertial, and multimap slam[J]. IEEE Transactions on Robotics, 2021, 37(6): 1874-1890.

[20]Newcombe R A, Lovegrove S J, Davison A J. DTAM: Dense tracking and mapping in real-time[C]. 2011 international conference on computer vision. IEEE, 2011: 2320-2327.

[21]Forster C, Pizzoli M, Scaramuzza D. SVO: Fast semi-direct monocular visual odometry[C]. 2014 IEEE international conference on robotics and automation (ICRA). IEEE, 2014: 15-22.

[22]Engel J, Koltun V, Cremers D. Direct sparse odometry[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(3): 611-625.

[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]Sun K, Mohta K, Pfrommer B, et al. Robust stereo visual inertial odometry for fast autonomous flight[J]. IEEE Robotics and Automation Letters, 2018, 3(2): 965-972.

[25]Tateno K, Tombari F, Laina I, et al. Cnn-slam: Real-time dense monocular slam with learned depth prediction[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 6243-6252.

[26]Dusmanu M, Schönberger J L, Pollefeys M. Multi-view optimization of local feature geometry[C]. Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16. Springer International Publishing, 2020: 670-686.

[27]DeTone D, Malisiewicz T, Rabinovich A. Superpoint: Self-supervised interest point detection and description[C]. Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2018: 224-236.

[28]Tyszkiewicz M, Fua P, Trulls E. DISK: Learning local features with policy gradient[J]. Advances in Neural Information Processing Systems, 2020, 33: 14254-14265.

[29]Dusmanu M, Rocco I, Pajdla T, et al. D2-net: A trainable cnn for joint description and detection of local features[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2019: 8092-8101.

[30]Revaud J, Weinzaepfel P, De Souza C, et al. R2D2: repeatable and reliable detector and descriptor[J]. arXiv preprint arXiv:1906.06195, 2019.

[31]Luo Z, Zhou L, Bai X, et al. Aslfeat: Learning local features of accurate shape and localization[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 6589-6598.

[32]Zhao X, Wu X, Miao J, et al. Alike: Accurate and lightweight keypoint detection and descriptor extraction[J]. IEEE Transactions on Multimedia, 2022.

[33]Ng P C, Henikoff S. SIFT: Predicting amino acid changes that affect protein function[J]. Nucleic acids research, 2003, 31(13): 3812-3814.

[34]Rublee E, Rabaud V, Konolige K, et al. ORB: An efficient alternative to SIFT or SURF[C]. 2011 International conference on computer vision. IEEE, 2011: 2564-2571.

[35]Bloesch M, Czarnowski J, Clark R, et al. Codeslam—learning a compact, optimisable representation for dense visual slam[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 2560-2568.

[36]Czarnowski J, Laidlow T, Clark R, et al. Deepfactors: Real-time probabilistic dense monocular slam[J]. IEEE Robotics and Automation Letters, 2020, 5(2): 721-728.

[37]Yang N, Stumberg L, Wang R, et al. D3vo: Deep depth, deep pose and deep uncertainty for monocular visual odometry[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 1281-1292.

[38]Wang S, Clark R, Wen H, et al. Deepvo: Towards end-to-end visual odometry with deep recurrent convolutional neural networks[C]. 2017 IEEE international conference on robotics and automation (ICRA). IEEE, 2017: 2043-2050.

[39]Zhou H, Ummenhofer B, Brox T. Deeptam: Deep tracking and mapping[C]. Proceedings of the European conference on computer vision (ECCV). 2018: 822-838.

[40]Teed Z, Deng J. Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras[J]. Advances in neural information processing systems, 2021, 34: 16558-16569.

[41]Bescos B, Fácil J M, Civera J, et al. DynaSLAM: Tracking, map**, and inpainting in dynamic scenes[J]. IEEE Robotics and Automation Letters, 2018, 3(4): 4076-4083.

[42]Yu C, Liu Z, Liu X J, et al. DS-SLAM: A semantic visual SLAM towards dynamic environments[C]. 2018 IEEE/RSJ international conference on intelligent robots and systems (IROS). IEEE, 2018: 1168-1174.

[43]Zhong F, Wang S, Zhang Z, et al. Detect-SLAM: Making object detection and SLAM mutually beneficial[C]. 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). IEEE, 2018: 1001-1010.

[44]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.

[45]Ye W, Yu X, Lan X, et al. DeFlowSLAM: Self-supervised scene motion decomposition for dynamic dense SLAM[J]. arXiv preprint arXiv:2207.08794, 2022.

[46]Sattler T, Leibe B, Kobbelt L. Efficient & effective prioritized matching for large-scale image-based localization[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 39(9): 1744-1756.

[47]Liu L, Li H, Dai Y. Efficient global 2d-3d matching for camera localization in a large-scale 3d map[C]. Proceedings of the IEEE International Conference on Computer Vision. 2017: 2372-2381.

[48]Arandjelovic R, Gronat P, Torii A, et al. NetVLAD: CNN architecture for weakly supervised place recognition[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2016: 5297-5307.

[49]Chen H, Luo Z, Zhang J, et al. Learning to match features with seeded graph matching network[C]. Proceedings of the IEEE/CVF international conference on computer vision. 2021: 6301-6310.

[50]Sun J, Shen Z, Wang Y, et al. LoFTR: Detector-free local feature matching with transformers[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 8922-8931.

[51]Wang Q, Zhang J, Yang K, et al. Matchformer: Interleaving attention in transformers for feature matching[C]. Proceedings of the Asian Conference on Computer Vision. 2022: 2746-2762.

[52]Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need[J]. Advances in neural information processing systems, 2017, 30.

[53]Sarlin P E, DeTone D, Malisiewicz T, et al. Superglue: Learning feature matching with graph neural networks[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 4938-4947.

[54]Chen H, Luo Z, Zhang J, et al. Learning to match features with seeded graph matching network[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2021: 6301-6310.

[55]Shi Y, Cai J X, Shavit Y, et al. Clustergnn: Cluster-based coarse-to-fine graph neural network for efficient feature matching[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2022: 12517-12526.

[56]Lindenberger P, Sarlin P E, Pollefeys M. Lightglue: Local feature matching at light speed[C]. Proceedings of the IEEE/CVF International Conference on Computer Vision. 2023: 17627-17638.

[57]Chen Y, Huang D, Xu S, et al. Guide local feature matching by overlap estimation[C]. Proceedings of the AAAI conference on artificial intelligence. 2022, 36(1): 365-373.

[58]Sarlin P E, Cadena C, Siegwart R, et al. From coarse to fine: Robust hierarchical localization at large scale[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019: 12716-12725.

[59]Taira H, Okutomi M, Sattler T, et al. InLoc: Indoor visual localization with dense matching and view synthesis[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 7199-7209.

[60]Yang L, Shrestha R, Li W, et al. Scenesqueezer: Learning to compress scene for camera relocalization[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022: 8259-8268.

[61]宁小娟, 李洁茹, 高凡, 等. 基于最佳几何约束和 RANSAC 的特征匹配算法[J]. 系统仿真学报, 2022, 34(4): 727.

[62]王静,王一博,郭铖,等.基于视觉的相机位姿估计方法综述[J].计算机应用研究,2024,41(08):2241-2251.

[63]Hu J, Huang L, Ren T, et al. You only segment once: Towards real-time panoptic segmentation[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 17819-17829.

[64]Whelan T, Salas-Moreno R F, Glocker B, et al. ElasticFusion: Real-time dense SLAM and light source estimation[J]. The International Journal of Robotics Research, 2016, 35(14): 1697-1716.

[65]Kong L, Shen C, Yang J. Fastflownet: A lightweight network for fast optical flow estimation[C]. 2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2021: 10310-10316.

[66]Balntas V, Lenc K, Vedaldi A, et al. HPatches: A benchmark and evaluation of handcrafted and learned local descriptors[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2017: 5173-5182.

[67]Zhou Q, Sattler T, Leal-Taixe L. Patch2pix: Epipolar-guided pixel-level correspondences[C]. Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021: 4669-4678.

[68]Li Z, Snavely N. Megadepth: Learning single-view depth prediction from internet photos[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 2041-2050.

[69]Toft C, Maddern W, Torii A, et al. Long-term visual localization revisited[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, 44(4): 2074-2088.

[70]Taira H, Okutomi M, Sattler T, et al. InLoc: Indoor visual localization with dense matching and view synthesis[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2018: 7199-7209.

[71]Yan S, Liu Y, Wang L, et al. Long-term Visual Localization with Mobile Sensors[C]. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2023: 17245-17255.

致 谢

时光荏苒，转眼间硕士阶段的学习与研究已画上圆满的句号。这段充实而难忘的学术旅程，既是我个人成长的重要阶段，也离不开诸多师长、同学、同事以及家人朋友的关怀与支持。在此，我谨向所有帮助过我的人致以最诚挚的谢意。

首先，我要衷心感谢我的导师曹雏清博士。在研究过程中，导师以渊博的学识和严谨的治学态度为我指点了迷津。从课题选定到实验设计，再到论文撰写，每一步都离不开导师的悉心指导。曹老师既以严谨的要求鞭策我追求卓越，又以鼓励和包容的心态陪伴我克服困难，让我在学术道路上不断成长。此外，导师还为我提供了先进的实验设备和技术支持，确保我能够顺利开展四足机器人定位与视觉感知的相关研究。曹老师的谆谆教诲和无微不至的关怀，不仅是本论文完成的重要基石，更是我未来学术道路上的宝贵财富。

同时，我衷心感谢课题组的每一位成员。特别感谢罗海南师兄和张靖师姐在实习期间给予我的热心帮助，他们的经验分享让我在工作与学习中受益匪浅。感谢同门张紫阳等同学在实验中的协助与讨论，他们的宝贵建议和无私支持让我在研究中少走了许多弯路。还要感谢哈尔滨工业大学的谢涛、戴崑、蒋志强等博士，他们在论文发表过程中给予我的大力帮助，让我深刻体会到团队协作在科研中的重要性。

此外， 我要感谢我的家人和朋友。感谢父母始终如一的支持与鼓励，他们是我前行的动力；感谢我的女朋友在生活和学习中的陪伴， 她的理解与宽容让我在繁忙的研究中找到片刻的放松。正是有了你们的默默支持，我才能心无旁骛地投入到学术探索之中。

最后，感谢评审专家和校内外同行们提出的宝贵意见，这些建议帮助我完善了论文内容，提升了研究质量。

硕士研究虽已告一段落，但学术之路尚且漫长。未来，我将带着这份感恩之心，继续努力钻研，追求更高的学术目标。本论文的完成不仅是个人努力的结晶，更凝聚了所有帮助过我的人的关怀与支持。在此，向你们致以最深切的谢意！

尚德教学　唯实惟新