# 基于云的视觉 SLAM 回环检测


## 摘要

近年来，智能机器人在生活的各行各业扮演着越来越重要的角色。在被称为“第四次工业革命”前夕的时代，同步定位和建图（Simultaneous Localization and Mapping 简称 SLAM）和传感器融合技术得到飞速发展，各种服务机器人，智能交通机器人相继出现，如：扫地机器人，家庭服务机器人，无人仓库搬运，无人驾驶等。但是在复杂环境下，目前很多 SLAM 算法表现依然不够鲁棒和实时，基于此，本文探讨过去惯性视觉 SLAM 的发展现状，并在 VINS-Mono[1]的基础上，提出基于云的回环检测和全局优化的思路，同时改进优化机制和噪声点检测方法，构建一个适合在复杂环境下运行的惯性视觉 SLAM 系统。数据集测试和算法对比的结果显示，在复杂环境下本文提出的 SLAM 系统能保持较高的精度运行，性能优于许多惯性视觉 SLAM。



**关键词**：惯性视觉 SLAM；传感器融合；优化算法；云计算；

# Cloud based visual SLAM loop detection


## Abstract

In recent years, intelligent robots play an increasingly important role in all walks of life. At the eve of the "fourth industrial revolution", simultaneous localization and mapping (SLAM) and sensor information fusion technology have developed rapidly. Various service robots and intelligent transportation robots have emerged one after another, such as floor sweeper, home service robot, unmanned warehouse, unmanned driving, etc. However, in the complex environment, the performance of many slam algorithms is still not robust. Based on this situation, this paper will discuss the development of inertial vision slam in the past. Moreover, we propose an idea of cloud-based loop detection and global optimization, improve the optimization mechanism and noise point detection method, to build a visual inertial SLAM system for complex environment. Finally, we test our system by using Euroc data set. The result shows that the slam system proposed in this paper can keep high precision in complex environment, and its performance is better than many visual inertial SLAM algorithms.



**Keywords**：Visual Inertial SLAM；Sensor Information Fusion；Optimization；Cloud；

# 目录

# 1. 绪论

## 1.1 课题背景与研究意义

21 世纪以来，由于计算机技术的日益成熟和控制技术，统计学等学科理论的不断完善，人工智能与机器人得到了爆发式地发展，越来越多智能机器人进入人们的视野。在生产制造业中，机器人的智能化减少了大量的社会生产人力成本，提升社会的生产力；在日常生活中，机器人让人们的生活更加便捷和丰富多彩；在军事方面，人工智能与机器人促进军事武器和设备的升级，使作战任务呈现多元化。战略意义之大不言而喻，因此，近年来国家极力支持人工智能和机器人的发展，出台大量政策，其中包括国务院引发的《新一代人工智能发展规划》、工信部《促进新一代人工智能产业发展三年行动计划（2018-2020 年）》。

实现精准定位和构建高精度地图，是移动机器人走向真正智能化的必要条件，也是移动机器人技术的基础。如无人驾驶，家庭服务机器人，仓库搬运机器人必须在精确定位，不发生碰撞的前提下才可以正常工作。因此鲁棒的 SLAM 技术在近十几年一直是学术界和工程界的研究热点，特别是在视觉 SLAM 和激光 SLAM 中，已经发展出非常成熟的理论，这些算法在静态和少噪声的环境下表现良好。但是机器人应用的真实环境往往是复杂多变的，许多算法无法在如此复杂的环境下实现精确定位，而且由于机器人的设备计算能力有限，无法实现实时而高精度的运算，这也是许多移动机器人尚未真正投入生产应用的重要原因。针对这两个问题，本文致力于改进 VINS-Mono 算法，提出基于云端的惯性视觉 SLAM 系统，同时在系统前端引入去除大噪声点方法和权值优化机制，使机器人能够在下雨等复杂环境下，实现较高精度的定位。

## 1.2 国内外研究现状

### 1.2.1 SLAM 研究现状

由于核心传感器不同，SLAM 可以分为视觉 SLAM 和激光 SLAM，前者是使用摄像头获取视觉信息，通过特征提取和特征匹配，恢复相机位姿；后者通过激光扫描得到点云，根据这些点云做状态估计，计算机器人位姿。由于精度高，实时性好，激光 SLAM 在无人驾驶等领域有着广泛的应用。但是激光传感器昂贵的成本阻碍了 SLAM 全面推广到实际应用的步伐。因此，成本更低，信息更丰富的视觉 SLAM 成为一大研究热点。本章节主要介绍视觉

SLAM 近十几年的研究状况：

2007 年 Davison 和 Andrew J.提出了第一个实时的视觉 SLAM 系统 MonoSLAM[2]，该算法利用单目摄像头，生成稀疏的特征点地图，估计位姿，并提出单目特征初始化和特征方向估计的方法。首次实现在纯视觉下，实时恢复相机位姿和构建稀疏地图，该算法被认为是现代视觉 SLAM 的开山之作。

同年, Klein 等人提出了 PTAM[3]，首次将视觉 SLAM 系统分出前端和后端，具体地，前端负责利用图像特征提取和估计相机运动，后端负责非线性优化和建图，优化特征点和相机位姿。其优点在于，运算量小的前端保证了 SLAM 在计算相机位姿的实时性，而运算量大的后端可以在另外的线程慢慢优化，同时保证整个 SLAM 系统的精度。

2014 年，Engel 等人提出了在大尺度环境下的单目 SLAM 系统 LSD-SLAM[4]。与传统的特征点法不同，该系统的前端使用直接法，即在不做特征提取的情况下直接通过像素信息估计位姿，实时重建三维环境。其优点在于，直接法省去了特征提取的时间，也避免了特征丢失的情况出现。

2015 年 Mur-Artal 等人提出了 ORB-SLAM[5]。该算法首次将 ORB 特征应用到 SLAM 当中，ORB 特征提取的快速和准确，有效地提高 SLAM 系统的性能。除此之外，该系统还引入了重定位机制，特征点和关键帧的优胜劣汰策略。它的精确性和普适性，使它受到了学术界工业界广泛的关注，并且得到了极大地推广。

为了提高 SLAM 的鲁棒性，近几年部分学者将思路转移到语义 SLAM 上，传统视觉 SLAM 的特征点属于低级几何信息，容易受到环境干扰而出现特征丢失或特征漂移等情况出现，针对这一缺陷，语义 SLAM 在图像的基础上提取更高级的语义信息，为系统提供增加鲁棒稳定的特征信息。除此之外，也有研究人员将传感器融合应用到 SLAM 中，不同种类的传感器测量者不同属性的信息，如视觉，距离等，多传感器融合为 SLAM 提供了更加多元的信息，在面对大噪声的情况下，系统有着充足而多元的信息来保证状态估计的精确性。其中惯性视觉 SLAM 更是在近几年得到巨大发展。接下来的章节我们将讨论惯性视觉的研究现状。

### 1.2.2 惯性视觉里程计研究现状

视觉里程计作为 SLAM 前端，主要负责估计相机位姿，以及局部优化地图特征点。而惯性视觉里程计（Visual Inertial Odometry 简称 VIO）则是在前者的基础上额外引入了 IMU 传感器信息，利用一定的传感器融合算法，让 SLAM 更加鲁棒。视觉携带的信息非常丰富，不会存在漂移，但是受环境影响大，在相机快速旋转或抖动的过程中容易造成特征点

丢失而引起定位失败。IMU 传感器可以直接测量物体自身的加速度和角速度，受外部环境影响小，但是测量数据会存在漂移，无法长时间准确估计相机状态。VIO 的目的在于将相机和 IMU 互补，以提高 SLAM 的鲁棒性：相机在短时间内的抖动和旋转可以通过 IMU 保持位姿估计，而 IMU 的漂移可以通过视觉信息修正。目前 VIO 广泛应用于 VR，AR 和无人机定位上，并在过去十几年得到快速发展，期间出现大量优秀的 VIO 框架。根据融合方式不同，目前主要有两个分支：松耦合(loosely-coupled)和紧耦合(tightly-coupled)。

所谓松耦合，即在计算纯视觉里程计后再融合 IMU 数据，这样可以避免图像高维度的特征加到状态向量中，以减少数据的耦合性并节省运算时间。2010 年 Jones[6]等人使用滤波方法，在线计算该模型状态的同时估计模型参数，其中包括重力和摄像机到 IMU 坐标系的转换关系。2016 年[7] Munguía 等人提出在滤波的框架下融合视觉，IMU 和 GPS 的信息实现稳定的状态估计，并应用在移动机器人的导航与测绘系统。

紧耦合则是将图像的特征加入到状态向量中与 IMU 的数据同时优化，数据间的耦合性增强以提高里程计的精度。2007 年 Mourikis[8]等人提出了著名的 MSCKF，全称 Multi-State Constraint Kalman Filter（多状态约束下的 Kalman 滤波器），在短时间图像纹理缺失和快速抖动的情况下依然保持较高精度，可在嵌入式系统下运行。2014 年 Leutenegger[9]等人考虑到连续两帧数据的相似性较高，逐帧计算冗余而费时间，因此在 VIO 上引入 KeyFrame 的概念，算法负责维护稀疏的 KeyFrame 中的状态量，在保证精度最优的同时节省大量的运算时间。2016 年 Mur-Artal[10]等人在 ORB-SLAM 的基础上，使用紧耦合融合 IMU 和视觉信息，该算法继承了 ORB-SLAM 的实时和鲁棒，而且弥补纯视觉当中存在的特征丢失问题。2017 年港科大无人机团队结合前人的研究基础提出 VINS-Mono[1]，相较于之前的工作，该算法在准确性和实时性方面提高了一个层次，成为当时视觉惯导领域的顶尖算法，不久该团队又设计出 VINS-Mono 的拓展版本 VINS-Fusion，VINS-Mobile，前者支持 GPS，单目双目，IMU 多传感器的融合，后者可应用于手机端。近两年 VIO 领域的主要研究点有：如何稳健地初始化[11]，引入深度学习方法神经网络[12][13]，如何在动态环境下运行[14]等。

### 1.2.3 云机器人发展现状

2010 年，James Kuffner 教授在某个会议上首次提到了云机器人的概念[15]，他指出机器人的决策程序可以通过网络上传到云端计算，而云端则发送指令远程控制机器人运作。如此一来，机器人的决策效率将不再受到自身的硬件设备限制，任何具备网络通信功能的

机器，在云端的加持下都可以成为强大的智能体。这种将决策和执行分离开的思想，受到了工业界和学术界的广泛关注。不久后 RoboEarth 的构想在英国诞生[16]，它描绘了日后机器人生态的蓝图：通过互联网来建立一个开源巨大的网络数据库，让全世界的机器人接入并调用服务，共享信息。这对机器人的任务协作，知识共享有着深远的意义。Liu 等人[17]提出基于云端的机器人导航学习结构：终身联合强化学习（LFRL）。机器人通过远程调用云端的模型实现导航功能，执行导航过程中使用强化学习修正模型并上传到云端供其他机器人使用。Limosani 等人在 2016 提出了一个基于云的机器人导航系统[18]，该系统的构想是将环境被划分为子地图，所有必要的信息和世界的拓扑表示都存储在远程云基础设施中。机器人通过特定的环境标签调用云服务，并能够动态、自动地更新其导航配置。

## 1.3 本文的主要研究内容

本文针对目前 SLAM 在复杂环境中的定位准确性较低，甚至出现定位失败的问题提出了自己的想法和解决方案，致力于设计一种基于云计算，在复杂环境下依然保持较高精确性和稳定性的系统。在 VINS-Mono 的基础上增加新颖的噪声点检测技术和权值优化机制，并在数据集上模拟雨天等复杂场景，衡量改进前后算法的精度，与前人研究工作做对比。本文内容安排如下：

第一章主要介绍课题背景及研究意义，并简单分析了课题中涉及的 SLAM 和惯性视觉里程计的相关工作，引出研究该课题的必要性。

第二章详细介绍 VINS-Mono 的算法原理和工作机制，主要包括：算法流程，里程计初始化，IMU 与视觉信息的紧耦合优化，以及相关公式的原理等。

第三章主要分析了现有算法在复杂环境下运行存在的问题和产生原因，并且提出基于云的惯性视觉 SLAM 系统，详细阐述权值优化机制和大噪声点检测的思路，和实现方法。

第四章主要介绍测试算法性能的实验方法，对比实验的设置，通过实验结果分析改进后算法性能的提升，以及对比各个算法之间的优劣。

第五章是对全文的工作总结以及本课题的未来展望与研究方向。

# 2. 惯性视觉 SLAM VINS-Mono 的算法原理

## 2.1 传感器测量数据的处理

### 2.1.1 特征提取和特征匹配实现

人类可以通过自己的视觉估计自身的运动状态，只要在视野中找到合适的参照物，我们就可以正确推测自身的运动方向还有速度。例如行走的时候看见周围树木往后移动，那么可以推断自身是往前行走的。学者受此启发，将参照物估计运动状态的思想引入到机器人上，通过特征提取和特征匹配，分别寻找环境中的参考点和构建前后两帧图像参考点的对应关系，进而通过这些匹配的特征点估计运动。在 VINS-Mono 中，算法使用 Harris 角点检测和 LK 光流法分别做特征提取和特征匹配，角点的灰度梯度明显，在提取和匹配过程更利于算法保持稳定的追踪，光流法省去特征点描述子的匹配和计算，直接通过灰度变化计算匹配特征点，具体原理如下：

（1） Harris 角点检测

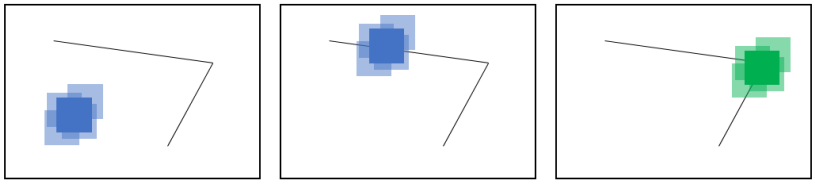

图 2.1 Harris 角点检测

图像的角点一般具有以下特性：一，轮廓之间的交点；二，该点附近的梯度变化明显；三，在同样的场景下即使视角改变，角点附近梯度依然保持稳定。Harris 角点检测的具体思路如图 2-1 所示：使用一个方框在图像上以任意方向试探，并比较移动前后方框内的灰度变化程度，对于在空白区域，方框在任意方向上的梯度变化较小；对于边缘区域，方框在沿着边缘方向的梯度变化小，而对于角点区域，方框在任意方向移动后的灰度变化较大，因此可断定在该方框内存在角点。

（2） LK 光流法

LK 光流法有三个假设条件：

a. 灰度不变性，即图像像素位置随着时间变化，但是他们的灰度值是保持不变的。这是所有光流法都必须满足的条件。

b. 运动小，即相邻的两帧图像变化小，这样才能保证灰度位置变化可以用泰勒近似，

并估计运动。

c. 空间位置不变，即当前帧的两个相邻像素点，在下一帧中也保持相邻。

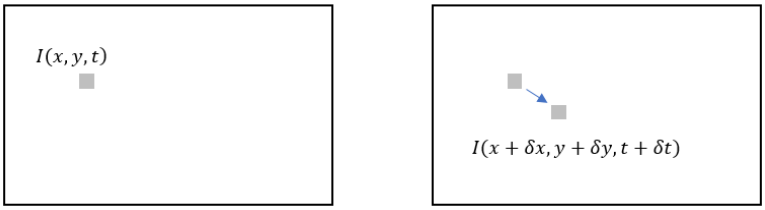


图 2-2 光流法示意图

在上述的三个假设成立的情况下，才能保证 LK 光流法正确求解，具体流程如图 2-2 所示：假设在 t 时刻有一像素点$I(x,y,t)$,其中$x,y$表示该像素在图像中的坐标，$t$表示时间，$I(*)$表示某个像素的灰度。在$t+\delta t$时刻，该像素点位置发生改变，为$I(x+\delta x, y+\delta y, t+\delta t)$。根据灰度不变性假设，像素点的灰度值恒定：

$$I(x,y,t) = I(x+\delta x, y+\delta y, t+\delta t) \tag{2,1}$$

根据小运动假设，上式可用泰勒展开，其中高阶项 H 可直接忽略为 0：

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t + H \tag{2,2}$$

由式（2,1）和（2,2）可得到方程，其中$\delta x$和$\delta y$为未知量：

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 \tag{2,3}$$

简化可写成：

$$I_x\delta x + I_y\delta y = -I_t\delta t \tag{2,4}$$

为了求解该方程，我们取以该像素为中心的 3X3 区域的九个像素点，根据空间位置不变假设，这九个像素在两帧中的相对位置不变，这意味着这九个像素的$\delta x$和$\partial y$是一致的。因此，九个像素构建九条方程，并利用最小二乘法估计$\delta x$和$\partial y$，从而得到该像素的光流。

$$\begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ \cdots & \\ I_{x8} & I_{y8} \\ I_{x9} & I_{y9} \end{bmatrix} * \begin{bmatrix} \delta x \\ \delta y \end{bmatrix} = - \begin{bmatrix} I_{t1}\delta t \\ I_{t2}\delta t \\ \cdots \\ I_{t8}\delta t \\ I_{t9}\delta t \end{bmatrix} \tag{2,5}$$

### 2.1.2 IMU 数据测量模型与预积分

IMU 测量的数据主要包括加速度和角速度，需要通过积分才能得到平移和旋转信息。根据 Joan Solà 的 IMU 测量模型[19]，物体加速度和角速度的测量值由真实值，偏置还有噪声组成，具体见式（2,6）（2,7）：

$$a_m = a_t + b_{a_t} + R_{b_t}^{w} g^{w} + n_a \tag{2,6}$$

$$\omega_m = \omega_t + b_{\omega_t} + n_\omega \tag{2,7}$$

其中$a_m$，$\omega_m$为 IMU 测量值，$a_t$，$\omega_t$为物体真实加速度和角速度，$b_{a_t}$和$b_{\omega_t}$分别为加速度偏置和角速度偏置，$R_{b_t}^{w}$表示世界坐标系到 t 时刻 IMU 坐标系下的旋转，$g^{w}$项代表世界坐标系下的重力分量，$n_a$和$n_\omega$为噪声项，值得注意的是，以上所有项都是在 IMU 坐标系下的值。假设已知$b_{a_t}$，$b_{\omega_t}$和$R_{b_t}^{w} g^{w}$，我们可以通过式（2,8），(2,9）估计真实值：

$$\hat{a}_t = a_m - b_{a_t} - R_{b_t}^{w} g^{w} \tag{2,8}$$

$$\hat{\omega}_t = \omega_m - b_{\omega_t} \tag{2,9}$$

为了方便计算，$\hat{a}_t$用世界坐标系表示，而$\hat{\omega}_t$用 IMU 坐标系表示：

$$\hat{a}_t = R_w^{b_t}(a_m - b_{a_t}) - g^{w} \tag{2,10}$$

$$\hat{\omega}_t = \omega_m - b_{\omega_t} \tag{2,9}$$

最后通过积分$[t_k, t_{k+1}]$内的角速度和加速度，物体在世界坐标系下的平移和旋转可由（2，11）(2,12)（2,13）获得：

$$\hat{v}_{t_{k+1}} = \hat{v}_{t_k} + \int_{t_k}^{t_{k+1}} \hat{a}_t \, dt \tag{2,11}$$

$$\hat{p}_{t_{k+1}} = \hat{p}_{t_k} + \hat{v}_{t_k} \Delta t_k + \iint_{t_k}^{t_{k+1}} \hat{a}_t dt^2 \tag{2,12}$$

$$\hat{q}_{t_{k+1}} = \hat{q}_W^{b_{t_k}} \otimes \int_{t_k}^{t_{k+1}} \frac{1}{2} \Omega(\hat{\omega}_t) q_{b_t}^{b_{t_k}} \, dt \tag{2,13}$$

其中$\Omega(\omega) = \begin{bmatrix} 0 & -\omega_z & \omega_y & \omega_x \\ \omega_z & 0 & -\omega_x & \omega_y \\ -\omega_y & \omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix}$，$q_b^t$ 表示 IMU 在 $t_k$时刻到 t 时刻的旋转。

在实际运行中，IMU 传感器的测量频率往往比相机要快很多，而在 VINS-Mono 中生成 KeyFrame 往往要更慢，如图 2-3 所示。为了获得两图像帧 $t_k$和 $t_{k+1}$之间的位姿和速度的关系，我们可以利用公式（2，11）(2,12)（2,13）对 $t_k$和 $t_{k+1}$之间 IMU 数据进行积分。但是在更新速度时，积分项需要知道每个 IMU 时刻的世界坐标系到 IMU 坐标系的旋转$R_w^{b_t}$；在更新平移时，出了$R_w^{b_t}$外，积分项还要计算每个 IMU 时刻的速度。对于每秒几百帧的 IMU 数据而言，每帧都要更新$R_{bw}^{t}$和$\hat{v}_t$显然并不合理，因此VINS-Mono 引入 C Forster[20]等人提出的

预积分概念。

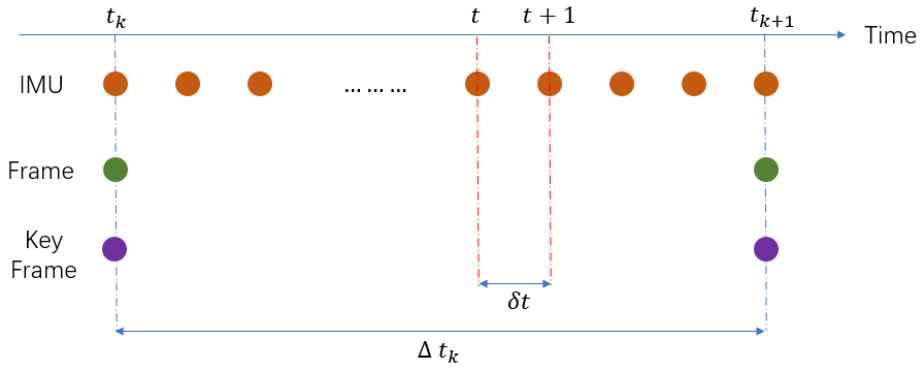


图 2-3 IMU 和图像接收频率示意图

预积分的核心思想在于，将复杂的积分项转化为增量形式，每当有新的 IMU 数据接收时，直接通过增量式方法更新位姿和速度，由此省略重复而繁杂的$R_{bw}^{t}$和$\hat{v}_t$计算。具体原理如下：首先将（2,10）（2,9）代入（2,11）（2,12）（2,13）中，得到：

$$\hat{v}_{t_{k+1}} = \hat{v}_{t_k} - g^w \Delta t_k + \int_{t_k}^{t_{k+1}} R_w^{b_t}(a_m - b_{a_t})\, dt \tag{2,14}$$

$$\hat{p}_{t_{k+1}} = \hat{p}_{t_k} + \hat{v}_{t_k}\Delta t_k - \frac{1}{2} g^w \Delta t_k^2 + \iint_{t_k}^{t_{k+1}} R_w^{b_t}(a_m - b_{a_t}) dt^2 \tag{2,15}$$

$$\hat{q}_{t_{k+1}} = \hat{q}_w^{b_{t_k}} \otimes \int_{t_k}^{t_{k+1}} \frac{1}{2}\Omega(\omega_m - b_{\omega_t}) q_b^t\, dt \tag{2,16}$$

将上述三式从世界坐标系转化到$t_k$时刻IMU 坐标系：

$$R_{b_{t_k}}^{w}\, \hat{v}_{t_{k+1}} = R_{b_{t_k}}^{w}(\hat{v}_{t_k} - g^w \Delta t_k) + \alpha_{t_k}^{t_{k+1}} \tag{2,17}$$

$$R_{b_{t_k}}^{w}\, \hat{p}_{t_{k+1}} = R_{b_{t_k}}^{w}(\hat{p}_{t_k} + \hat{v}_{t_k}\Delta t_k - \frac{1}{2} g^w \Delta t_k^2) + \beta_{t_k}^{t_{k+1}} \tag{2,18}$$

$$q_{b_{t_k}}^{w} \otimes \hat{q}_{t_{k+1}} = \gamma_{t_k}^{t_{k+1}} \tag{2,19}$$

其中$\alpha_{t_k}^{t_{k+1}}$，$\beta_{t_k}^{t_{k+1}}$，$\gamma_{t_k}^{t_{k+1}}$表示积分项：

$$\alpha_{t_k}^{t_{k+1}} = \int_{t_k}^{t_{k+1}} R_{b_t}^{b_{t_k}}\left(a_m - b_{a_t}\right) dt \tag{2,20}$$

$$\beta_{t_k}^{t_{k+1}} = \int_{t_k}^{t_{k+1}} R_{b_t}^{b_{t_k}}(a_m - b_{a_t})\, dt^2 \tag{2,21}$$

$$\gamma_{t_k}^{t_{k+1}} = \int_{t_k}^{t_{k+1}} \frac{1}{2}\Omega(\omega_m - b_{\omega_t}) q_{b_t}^{b_{t_k}}\, dt \tag{2,22}$$

（2,14）和（2,15）等式右边由积分项和非积分项组成，其中非积分项只与$t_k$的状态有关，而积分项主要是包含了$[t_k, t_{k+1}]$之间的 IMU 测量值和旋转，因此只需把上述积分项转化为增量式，最终可得到预积分的完整形式，其中$t$和$t+1$表示在$[t_k, t_{k+1}]$之间的某一时刻：

$$\alpha_{t+1}^{t_{k+1}} = \alpha_t^{t_{k+1}} + R\left(\gamma_t^{t_{k+1}}\right)(a_m - b_{a_t})\,\delta t \tag{2,23}$$

$$\beta_{t+1}^{t_{k+1}} = \beta_t^{t_{k+1}} + \alpha_t^{t_{k+1}}\,\delta t + R\left(\gamma_t^{t_{k+1}}\right)(a_m - b_{a_t})\,\delta t^2 \tag{2,24}$$

$$\gamma_{t+1}^{t_{k+1}} = \gamma_t^{t_{k+1}} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}(\omega_m - b_{\omega_t})\delta t \end{bmatrix} \tag{2,25}$$

## 2.2 惯性视觉里程计

### 2.2.1 基于松耦合的初始化

一个鲁棒的惯性视觉里程计离不开稳健精确的初始化，其中包括估计系统的重力分量，尺度，IMU 偏置等参数，和摄像头与 IMU 之间的刚体变换。VINS-Mono 的初始化主要有三个步骤：1，纯视觉恢复运动；2，惯性视觉校准；3，优化重力分量。

（1） 纯视觉恢复运动

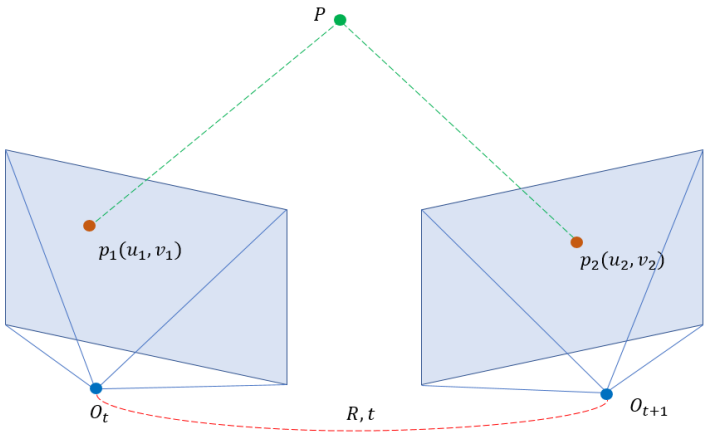


图 2-4 对极几何

在初始化完成之前，算法尚未对 IMU 和摄像头进行校准，无法做传感器融合，因此只能使用纯视觉里程计恢复运动。对于每一帧图像，算法首先使用 Harris 角点检测和 LK 光流法做特征匹配和特征提取，再利用对极几何求解运动。假设空间中有一静止点$P$，在 t 时刻相机位于$O_t$，观测到空间点$P$位于图像$p_1(u_1, v_1)$位置，经过一次运动$R,t$到达$O_{t+1}$的位置，并观测观测到空间点$P$位于图像$p_2(u_2, v_2)$位置，如图 2-4 所示。现在已知$p_1$，$p_2$，相机内参$K$，求$R,t$以及$P$空间坐标，根据相机投影模型，空间中一点投影到图像平面上的坐标为:

$$p_1 = KP \tag{2,26}$$

$$p_2 = K(RP + t) \tag{2,27}$$

根据对极几何约束：

$$K^{-1}p_1 = P = R^{-1}(K^{-1}p_2 - \mathrm{t}) \tag{2,28}$$

变形得到：

$$p_2^T K^{-T} t \hat{} R K^{-1} p_1 = 0 \tag{2,29}$$

上述方程的未知量$R,t$一共有 6 个自由度，而一对匹配特征点$p_1 p_2$可以构建一个约束，为了求解$R,t$，只需在图像中找到足够多的匹配特征点提供约束，构建超定方程，通过最小二乘法恢复运动$R,t$。

（2） 惯性视觉校准

单目对极几何无法解决尺度问题，因为方程（2,29）乘以任意尺度常数依然成立，因此运动中的平移量真实尺度无法确定。因此在惯性视觉校准中，通过 IMU 的测量值恢复纯视觉里程计运动中的尺度，同时估计 IMU 的偏置，以及 IMU 和摄像头之间的刚体变换。

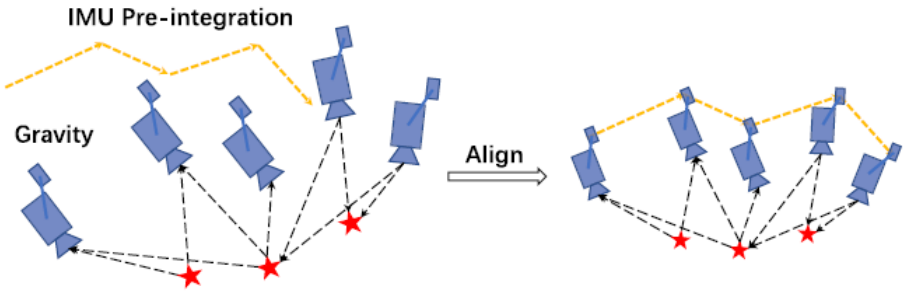


图 2-5 惯性视觉校准

如图 2-5 所示，在纯视觉里程计估计运动的同时，算法将 IMU 接收的信息做预积分。运行一段时间后，算法收集到了丰富的信息，最后通过最小化旋转误差和最小化重投影误差估计 IMU 的角速度偏置，各个时刻的运动速度，尺度信息和重力分量。

（3） 优化重力分量

重力分量通过扰动模型优化，具体如图 2-6 所示。固定重力分量的模长$|g|$，在重力与单位圆交点的相切面上，给两个向量$b_1$，$b_2$加上任意的扰动量。更新得到重力分量为：$\mathrm{g}(\hat{g} + \delta g)$，其中$\delta g = w_1 b_1 + w_2 b_2$，并利用最小化相机重投影误差将$w_1$，$w_2$与惯性视觉校准参数一同优化。最后利用$w_1$，$w_2$更新得到最优重力分量。

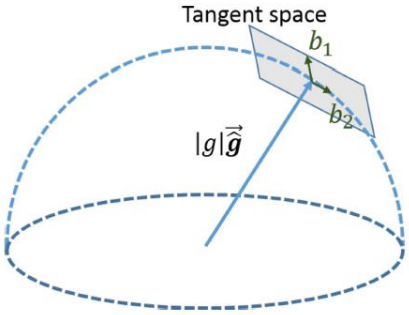


图 2-6 重力分量扰动模型

### 2.2.2 紧耦合单目 VIO

VIO 的初始化估计重力分量，IMU 偏置，尺度以及相机和 IMU 之间的刚体变换等参数，为传感器融合提供基础。初始化成功后，算法将直接利用 IMU 和相机数据的紧耦合做状态估计，本章节将详细介绍 VIO 紧耦合的具体原理。

所谓紧耦合，其实是传感器融合中数据关联的一种形式，与松耦合分而治之的思想不同，紧耦合具体是把各个传感器参数一同添加到状态向量上，通过状态向量与测量数据的关系构建约束并计算误差目标函数，通过最小化目标函数得到最优解。在 VINS-Mono 中状态向量$X$由三部分组成，分别为各个特征点的深度$\{\tau_0,\tau_1,\tau_2 \ldots\ldots \tau_i\}$，各帧 IMU 状态$\{x_0,x_1,x_2 \ldots\ldots x_k\}$其中包括各帧的速度$v_k$，位姿$p_k,q_k$和 IMU 偏置$b_{ak},b_{\omega k}$，以及 IMU 和相机之间的刚体变换$\{x_c\}$，具体见公式（2,32）：

$$x_k = \{p_k, v_k, q_k, b_{ak}, b_{\omega k}\} \tag{2,30}$$

$$x_c = \{p_c, q_c\} \tag{2,31}$$

$$X = \{x_0, x_1, x_2 \ldots\ldots x_k, x_c, \tau_0, \tau_1, \tau_2 \ldots\ldots \tau_i\} \tag{2,32}$$

状态向量$X$作为整个算法的目标估计值，其精度取决于两个因素：一是 IMU 测量数据和相机测量数据的质量，二是状态向量和测量值之间的数据关联。前者主要受外部环境和传感器本身质量的影响，后者则与算法本身的设计相关，因此构建数据关联也成为 VIO 的关键步骤。所谓数据关联，即使用具体公式描述测量值与状态之间的关系，并且利用该公式计算测量误差用以优化，而 VIO 主要包括 IMU 测量误差和视觉测量误差。

（1） IMU 测量误差

在$[t_k, t_{k+1}]$时间段，由公式（2，20），(2,21），(2,22）可得到该时间段的 IMU 预积分值 $\alpha_{t_k}^{t_{k+1}}$，$\beta_{t_k}^{t_{k+1}}$，$\gamma_{t_k}^{t_{k+1}}$，这些值直接通过 IMU 测量数据得到，因此可作为观测值。而根据

公式（2,17），（2,18），（2,19）所描述的状态值与观测值之间的关系，我们可以通过状态值估计得到预积分的预测值$\hat{\alpha}_{t_k}^{t_{k+1}}$，$\hat{\beta}_{t_k}^{t_{k+1}}$，$\hat{\gamma}_{t_k}^{t_{k+1}}$。现在IMU 误差可被定义为状态预测值与观测值之间的差值：

$$\delta\alpha_{t_k}^{t_{k+1}} = \hat{\alpha}_{t_k}^{t_{k+1}} - \alpha_{t_k}^{t_{k+1}} \tag{2,33}$$

$$\delta\beta_{t_k}^{t_{k+1}} = \hat{\beta}_{t_k}^{t_{k+1}} - \beta_{t_k}^{t_{k+1}} \tag{2,34}$$

$$\delta\gamma_{t_k}^{t_{k+1}} = \hat{\gamma}_{t_k}^{t_{k+1}} - \gamma_{t_k}^{t_{k+1}} \tag{2,35}$$

除此之外，IMU 偏置误差直接定义前后两帧$b_{a_k}$的差值：

$$\delta b_{a_k} = b_{a_{k+1}} - b_{a_k} \tag{2,36}$$

$$\delta b_{\omega_k} = b_{\omega_{k+1}} - b_{\omega_k} \tag{2,37}$$

（2） 视觉测量误差

同样考虑在$[t_k, t_{k+1}]$时间段，空间中有一个特征点$P$，相机在$t_k$时刻观测到该特征点在图像的坐标为$p_k(u_k, v_k)$，在$t_{k+1}$时刻观测到该特征点在图像的坐标变成$p_{k+1}(u_{k+1}, v_{k+1})$。根据相机模型和坐标变换公式，我们利用$p_k$坐标和状态值预测$t_{k+1}$时刻的图像坐标$\hat{p}_{k+1}$，并和观测值$p_{k+1}$作比较，如图 2-7 所示：

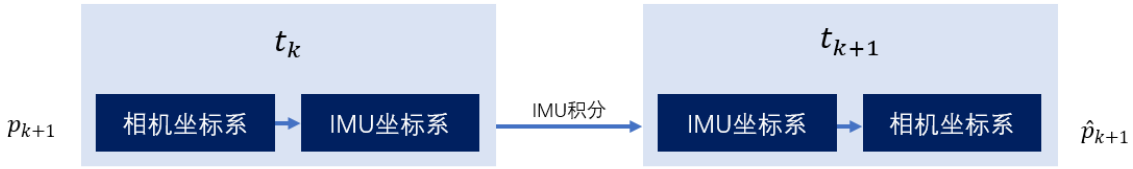


图 2-7 预测$t_{k+1}$时刻特征点的图像坐标过程

首先根据$t_k$时刻特征点图像坐标以及相机内参，计算特征点在相机坐标系下的三维坐标，其中$\tau_k$为特征点深度：

$$P_{c_k} = K^{-1} * p_k * \tau_k \tag{2,38}$$

随后利用 IMU 和相机之间的刚体变换，计算在$t_k$时刻 IMU 坐标系下特征点的三维坐标$P_{imu_k}$：

$$P_{imu_k} = T_c * P_{c_k} \tag{2,39}$$

根据 IMU 在$[t_k, t_{k+1}]$时刻的积分信息，我们计算得到 IMU 在$t_k$和$t_{k+1}$之间的变换，并将$P_{imu_k}$转化为在$t_{k+1}$时刻 IMU 坐标系下特征点的三维坐标$P_{imu_{k+1}}$：

$$P_{imu_{k+1}} = T_{t_k}^{t_{k+1}} P_{imu_k} \tag{2,40}$$

将$P_{imu_{k+1}}$转换为在 $t_{k+1}$时刻相机坐标系下特征点的三维坐标：

$$P_{c_{k+1}} = T_c^{-1} * P_{imu_{k+1}} \tag{2,41}$$

最后根据相机模型预测 $t_{k+1}$时刻观测到特征点在图像的坐标，并将其归一化：

$$\hat{p}_{k+1} = \frac{KP_{c_{k+1}}}{z_{k+1}} \tag{2,42}$$

定义某个特征点的视觉测量误差为：

$$\delta p_{k+1} = \hat{p}_{k+1} - p_{k+1} \tag{2,43}$$

### 2.2.3 基于关键帧和滑动窗口法的局部优化

虽然基于紧耦合的 VIO 精度更高，但是状态向量保存着巨大的信息量，并随着程序的运行逐渐累积，若对每一帧图像同时优化，会导致 VIO 计算复杂性增加，耗时长，更新越来越慢。针对上述问题，VINS-Mono 引入关键帧和滑动窗口法的思想，在减少运算复杂性的同时保持 VIO 的高精度特性。

（1）关键帧

由于相邻两帧的信息有较大的冗余，VINS-Mono 算法规定检测前后两帧的视差，只有当视差大于某一特定的值时，才将其作为关键帧，VIO 只负责对各个时刻的关键帧做优化，图 2-8 显示了关键帧的稀疏性。和逐帧优化不同，引入关键帧的优势在于使优化变量变得稀疏，避免出现状态向量维度过高的情况出现。

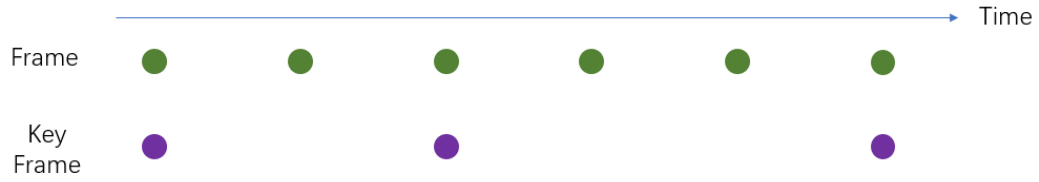


图 2-8 关键帧的稀疏性

（2）滑动窗口法

虽然引入关键帧能减少状态向量的维度，但是随着传感器数据的不断输入，关键帧的数量也不可避免地增加，导致程序运行越来越慢。滑动窗口法针对该问题提出了解决方案：VIO 只维护更新最新 n 个关键帧的数据，忽略过去久远的信息，如图 2-9 所示。

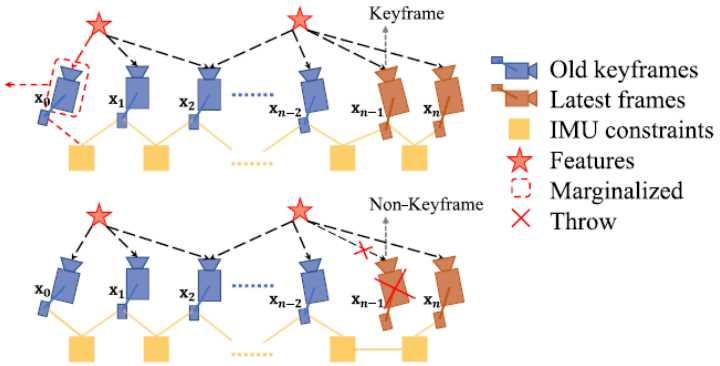


图 2-9 滑动窗口法示意图

滑动窗口主要包含三类信息：一是特征点数据，二是相机位姿，三是 IMU 的积分信息。VIO 在此滑动窗口中，根据上一章节的紧耦合方法做数据关联，构建一张由节点（优化变量）和边（约束）组成的图，通过局部优化得到窗口内各帧的最优状态。其中滑动窗口的大小 n 即为关键帧的最大数量。每当有新的一帧图像数据输入时，首先计算最新一帧图像与前一关键帧的视差，若大于某特定值则将最新一帧作为关键帧添加到滑动窗口中，而最旧的一帧数据则被丢弃。否则将最新一帧的视觉信息丢弃，保留预积分信息，继续等待图像数据的输入。

## 2.3 本章小结

本章主要分为两个章节介绍了惯性视觉里程计的具体原理，第一章节介绍了摄像头和 IMU 的基本数据预处理方式，其中包括 IMU 预积分，角点检测和光流追踪。在此基础上，第二章节详细介绍了惯性视觉里程计的实现步骤，并针对各个步骤的特点，分析其解决的问题的核心思想。具体有惯性视觉里程计的初始化，紧耦合 VIO 状态估计，以及基于关键帧和滑动窗口法的局部优化。

# 3. 基于云的惯性视觉 SLAM 系统

## 3.1 复杂环境下传统 VIO 的问题

### 3.1.1 问题描述

无论是纯视觉还是视觉 IMU 融合，传统 SLAM 算法都是在环境是静态的假设上进行定位的，即机器人周围的环境固定不动，不存在自主移动的物体，如道路上的路标，围栏，还有建筑物等。但是实际的应用场景中，机器人身边往往充斥着大量的动态物体，如图 3-1 所示，包括道路上行驶的汽车，行人，动物，以及被移动的手推车等。这些动态物体会对机器人定位产生极大的干扰。很容易理解，当某个人在道路上行走，并以道路两侧的树木作为参考系，他可以根据树木往后移动的现象确认自身正在往前走的事实，但是如果把快速移动的汽车作为参考系，则得到的结果是自身相对于汽车是往后运动的。从该现象可以看出，选择不同的参考系会估计得到不同的运动结果。在实际应用中，机器人一般是基于大地参考系估计运动，上一章介绍的特征提取本质上，就是机器人在环境中选择梯度明显易分辨的角点作为参考系并估计自身运动。但是机器人无法确认在这些参考点中哪一些是属于静态，哪一些属于动态。

图 3-1 静态场景（左）和动态场景（右）示意图

除此之外，机器人应用场景不乏存在恶劣的环境条件，如下雨和雾天等天气现象。从图 3-2 可以看到，由于图像棱角变模糊的影响，Harris 角点检测算法的噪声会大大增加，特征点位置产生偏移，导致定位误差变大甚至产生严重漂移。另一方面，图像受到雨滴的影响，像素间的相对位置会随时间发生巨大改变的，光流法第三假设将不再成立。导致光流追踪失败。以上两因素综合在一起，会严重破坏 SLAM 系统的定位精度。

图 3-2 道路暴雨天气

### 3.1.2 产生原因和相关解决方法

本章节主要分析机器人在动态环境下定位失败的原因，通过公式证明动态物体和噪声环境对机器人定位的影响。假设在$t$时刻，相机位于$O_t$观测到空间中有一个属于动态物体的点$P_t$，位于图像像素点点$p_1$上，经过一个时间单位的运动，$t+1$时刻该点移动到了$P_{t+1}$，相机在$O_{t+1}$位置观测到该点位于图像像素点$p_2$，如图 3-3 所示。

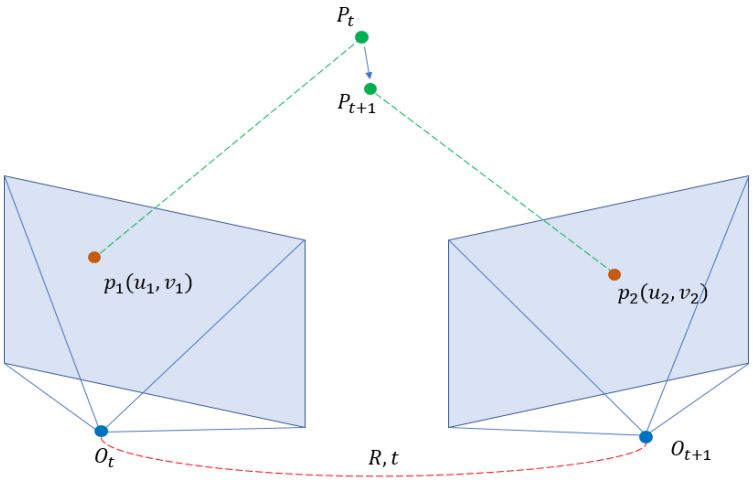


图 3-3 动态物体破坏了对极几何约束

可以看到，由于动态物体发生了运动，相机在前后两个时刻所观测到的像素点不在空间中的同一位置，这意味着对极几何的约束被破坏，即公式（2,28）的等式不再成立，并出现式（3,1）（3,2）的情况。显而易见，等式两边的误差与动态点的运动速度有关，当$P_t$在该时间单位下运动较小，$P_t$可以近似等于$P_{t+1}$；若$P_t$发生了较大的运动，$P_t$和$P_{t+1}$之间的距离变大，由对极约束估计的运动就会有非常大的误差。

$$P_t \neq P_{t+1} \tag{3,1}$$

$$K^{-1}p_1 \neq R^{-1}(K^{-1}p_2 - \mathrm{t}) \tag{3,2}$$

针对该问题，现有的解决方法主要是通过 RANSAC 去除这些大的噪声点。 RANSAC 的核心思想是：在样本集中采样并拟合模型，衡量样本集中符合该模型的数量，经过重复迭代得到最优的模型。RANSAC 在小噪声的情况下表现良好，并能准确剔除动态物体上的噪声点，但是对大噪声处理表现欠佳。除此之外，也有学者使用深度学习提取语义信息，分割场景的动态物体，对动态特征点进行剔除，但是缺点深度学习的需要较大的运算成本，使系统计算耗时增加，与 SLAM 系统的实时性相违背。

图 3-4 语义分割（左）和 RANSAC 消除离群点（右）

## 3.2 基于云的惯性视觉 SLAM 系统

### 3.2.1 系统架构

本文提出的系统架构主要分为两个部分：云端和机器人端。其中机器人端主要负责接收传感器信息，对 IMU 数据和图像数据分别进行预积分和特征提取，更新维护滑动窗口内的状态，并且在引入权值优化和噪声点检测机制，提高算法在复杂环境中的鲁棒性，具体细节将在下两章节介绍。机器人端的程序必须做到实时响应，但是基于嵌入式的机器人算力往往受限于自身的硬件设备，所以我们需要借助云端的算力，将实时性要求不高并且运算量大的回环检测和全局优化程序放到云端计算。受益于云端的储存能力和计算能力，回环检测的词袋模型可以拓展的更加庞大，这有助于提升回环检测的准确性。同时，这种基于中央式处理的架构有利于词袋模型的复用。在此结构下，即使网络条件差导致云端与机器人端连接失败，机器人定位程序理论上依然可以继续运行，等到重新连接上云端后，再调用全局优化和回环检测的服务优化机器人的位姿。

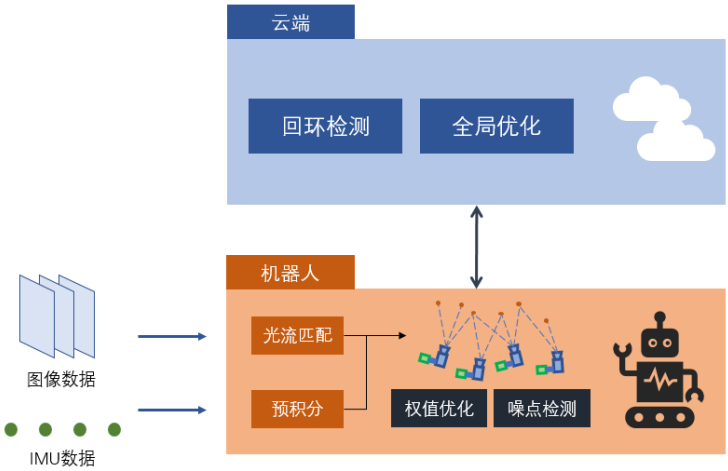


图 3-5 基于云的惯性视觉 SLAM 系统

### 3.2.2 大噪声特征点检测

为了解决 VIO 在动态环境下容易受到噪声点干扰的问题，算法必须充分发挥多传感器融合的优势，IMU 在短时间内能较准确地表征物体的运动，而根据大噪声点所求解的运动模型与 IMU 运动存在较大冲突。基于此事实，本文提出一种借助 IMU 的运动信息辅助检测图像噪声点的方法。与 RANSAC 语义分割的方法相比，该方法在保持高实时性的同时，在大噪声的环境下依然表现稳定，通过简单地评估各个特征点在 IMU 运动模型下的拟合程度，判断特征点中是否存在大噪声点。2018 年 Babu[21]等人提出了一种相似的方法，同样使用 IMU 运动信息检测噪声点，解决 IMU 与特征点运动。但不同的是本文提出的思路是根据 IMU 的方差动态调整阈值，确保系统在最大程度检测噪声点的同时，避免因为阈值过于严苛而削减数据多样性，导致漂移情况的发生。

如图 3-5 所示，相机在$O_t$观测到特征点$P_t$在图像$p_1$上，经过一次运动，特征点移动到$P_{t+1}$，相机在$O_{t+1}$观测到$P_{t+1}$在图像$p_2$上。现在问题可描述为：已知$P_t$的三维坐标，$O_t$的世界坐标系下的位姿，IMU 在$[t, t+1]$之间的积分信息，和$p_1$，$p_2$坐标，判断$P_t$是否属于动态特征点。根据公式（2，14）（2，15）（2，16），首先利用 IMU 积分信息估计$O_{t+1}$相机位姿。假设$P_t$属于静态点，那么算法利用相机模型（2,27），即可预测相机在$O_{t+1}$观察$P_t$在图像中的位置$p_2'$。对比预测值$p_2'$和实际观测值$p_2$发现，由于动态物体移动的影响，两者往往存在较大的差别，我们规定当两点距离大于某一阈值时，则把该点归类为动态特征点。

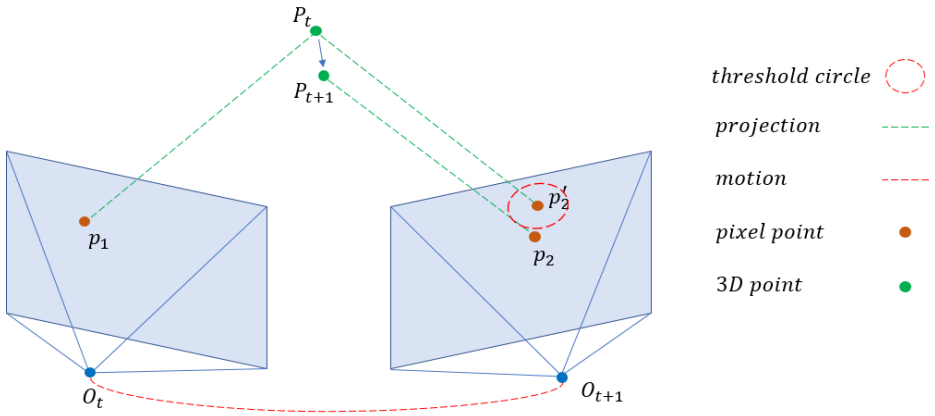


图 3-5 IMU 检测噪声点示意图

除此之外，阈值的设定也是一个非常重要的步骤，若阈值太小，则大量的特征点被剔除，失去了数据的多样性，无法校正 IMU 的偏移；若阈值太大，则会引入额外的误差。因此本文还提出一种设定自适应阈值的思路，即根据 IMU 数据的协方差考察不确定性，并动态设定阈值。

IMU 测量值并不完美，数据往往伴随着噪声和偏移，预积分值的不确定性随着 IMU 测量值的传入而增加，如图 3-6 所示。在图像数据到达之前，每当接收到 IMU 数据，系统状态做一次前向传播，此过程中方差会不断增加。直到接收到图像数据，系统在滑动窗口内做一次局部优化，得到最优估计值。

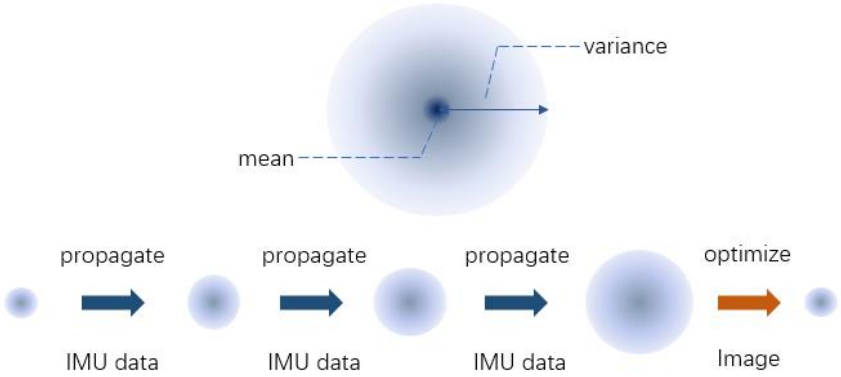


图 3-6 状态不确定性在前向传播中扩散

在判断某特征点是否属于大噪声点时，我们将 IMU 状态的不确定性引入到图像去噪阈值中。具体如图 3-7 所示，将 IMU 状态的协方差映射到二维图像平面上，得到该特征点在图像最有可能的区域$p$，通过与实际观测到的位置$p_m$作比较。映射过程涉及较多的计算公式和繁琐的协方差传递步骤，具体可参考[19]的第五章。

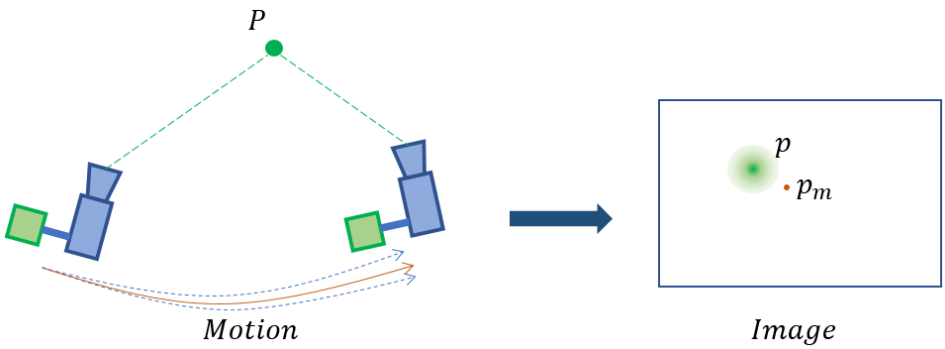


图 3-7 IMU 误差到图像误差的映射

### 3.2.3 权值优化机制

去除大噪声点使算法在复杂环境中运行保持稳健性，但是它也有明显的缺点：复杂环境下提取特征点数量相对较少，加上主动去除噪声点后，剩下的视觉特征点数据不足以校正 IMU 传感器本身的误差，久而久之系统会出现比较大的漂移，位姿误差甚至比去除噪声点之前更大，这显然不符合该方法的初衷。因此，算法有必要研究在数据多样性不足的情况下如何保持准确定位的问题。通过代入场景直观地分析，我们发现人在估计自身运动的过程中，不会在盲目而随机地选取参照物，反而是优先选择更有参照价值的物体。换言之，在人的视野中各个参照物存在一个置信度，它衡量的是根据该参照物所评估的运动是否可信。正是利用这种性质，本文提出权值优化的概念：给各个特征点分配一个权值，在优化过程中优先调整参数，以减少权值大的特征点误差。该方法的优点在于，即使环境的噪声干扰导致特征点数量不足，但只要有若干个质量好的特征点，即可恢复精确的定位。该方法主要分为分配权值和构建目标函数两步。

（1） 分配权值

该方法首先评估特征点对过去运动模型的拟合程度，给各个特征点分配权值，拟合程度越高代表该特征点的数据越精确，相应地所分配的权值越大。第二章的单目紧耦合 VIO 中提到，已知运动模型，特征点三维坐标和前后两帧特征点图像坐标，可通过公式（2-38）至（2-43）求解得到特征点的观测误差。但是值得注意的是，这里要评估的是特征点对优化后结果的拟合程度，因此该已知的运动模型不是 IMU 测量的先验值，而是优化后的后验值。利用式（3，3）计算得到各个特征点的权值$c_i$，其中$c_i$保持在$(0,1]$之间：

$$c_i = \exp(-\|\delta p_i\|^2) \qquad (3，3)$$

除此之外，根据实际特征点的数量n，还需分配一个整体的特征点权值，以增加所有视觉信息在优化时的置信度。具体可通过式（3，4）得到：

$$C = N/n \quad (3，4)$$

其中$N$为特征点可提取的最大数量，最后将式（3，3）（3，4）相乘得到最终的特征点权值，见式（3,5）：

$$\rho_i = C * c_i \quad (3，5)$$

（2） 构建目标函数

传统的视觉惯性 SLAM 目标函数如式（3,6）所示，相机观测值$z_i^c$和 IMU 观测值$z_j^{IMU}$作为状态$X$的约束，在优化过程中通过调整状态向量$X$的各个参数，使视觉测量误差$r_c$和 IMU 测量误差$r_b$达到最小，估计得到最优的状态。该目标函数各个误差项的系数均为 1，这意味着每个 IMU 和相机的观测值对优化同等重要。在特征点数量缺失的情况下，视觉观测误差对整个系统误差的比重变小，状态优化更偏信于 IMU 的测量值，导致系统无法校正 IMU 的误差，从而产生漂移现象。

$$\min_X\{\sum_{i=0}^{N} r_c(z_i^c, X) + \sum_{j=0}^{M} r_b\left(z_j^{IMU}, X\right)\}$$

因此我们利用上一小节得到的权值重新构建该目标函数，具体是在相机观测误差前乘以一个系数， 如式（3,7）所示。该权值的作用一是保证视觉测量误差在系统整体误差的比重不会过低，避免优化过于偏信于 IMU 测量值，二是提高质量高的特征点比重，有利于用最小成本，估计得到最优结果。

$$\min_X\{\sum_{i=0}^{N} \rho_i * r_c(z_i^c, X) + \sum_{j=0}^{M} r_b\left(z_j^{IMU}, X\right)\}$$

## 3.3 本章小结

本章主要阐述了传统的视觉惯性 SLAM 在复杂环境下容易定位失败的现象和分析原因，并列举了当前解决该问题的方法，以及各自的优缺点。在此基础上，本章提出了一个基于云的惯性视觉 SLAM 系统，并引出了本文提高 VIO 在复杂环境下鲁棒性的核心思想：大噪声点检测和权值优化机制。两种方法分别解决了复杂环境的噪声引入问题和特征点数量缺失导致的定位漂移问题，通过公式推导和举例详细说明了这两种思路的可行性。

# 4. 动态复杂环境下的 VIO 实验

## 4.1 实验数据集

本文使用 EuRoc[22]数据集对算法进行评估和测试，该数据集通过无人机采集了工业厂房内的图像，这些序列具有光线明暗变化大，运动旋转较剧烈的特点，常被用于评估 VIO 的鲁棒性。采集数据的无人机上装载的传感器主要有摄像头 Aptina MT9V034 global shutter，惯性传感器 ADIS16448，以及用于获取真实值的运动捕捉标记点。飞行时同步采集 IMU 数据和双目摄像头的图像，但值得注意的是，IMU 数据和图像数据的采集频率不同，前者为 200Hz，后者为 20Hz。

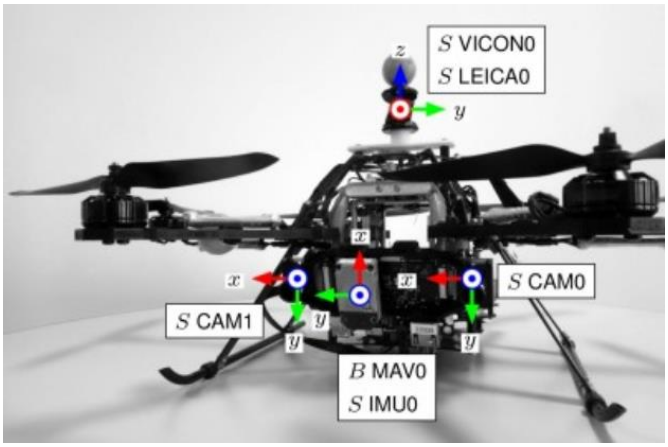


图 4-1 无人机传感器安装位置

图 4-2 实验真实环境

但是该数据集没有动态物体等大噪声点，难以测试本文所改进算法的性能，因此本文使用图像处理技术模拟了下雨的场景，通过设置信噪比调节雨量大小。实验测试在不同信噪比下各算法在没有回环检测的帮助下的路径精度和累积误差。定义累积误差为某时刻系

统求解位姿$p_{km}$和真实位姿$p_{kt}$的欧氏距离，需要注意的是，$p_{kt}$和$p_{km}$必须进行时间对齐：

$$r_k = distance(p_{kt}, p_{km})$$

（1） 实验一

设置环境噪声于信号比为0，测试VINS-Mono，ICE_BA[23]和改进算法的累计误差和路径精度，该实验作为后续实验的对照组。

（2） 实验二

设置环境噪声于信号比为0.001，并在系统初始化完成后引入噪声，测试VINS-Mono，ICE_BA和改进算法的累计误差和精度。该组测试环境存在噪声点情况下，检测噪声点和权值优化的方法对系统鲁棒性的提高。

（3） 实验三

设置环境信噪声于信号比为0.003，并在系统初始化完成后引入噪声，测试VINS-Mono，ICE_BA和改进算法的累计误差和精度。

（4） 实验四

设置环境信噪声于信号比为0.005，并在系统初始化完成后引入噪声，测试VINS-Mono，ICE_BA和改进算法的累计误差。

为了必将三个算法的前端累计误差，以上四组实验均未开启后端的回环检测和全局优化。同时为了控制变量，所有实验均在同一设备下运行，该设备处理器为Intel(R) Core(TM) i7-4710MQ CPU @ 2.50GHz，运行内存8G 。程序使用ROS平台的节点订阅与发布机制实现进程之间的数据交换，以及使用ROS自带软件rviz实现数据可视化。

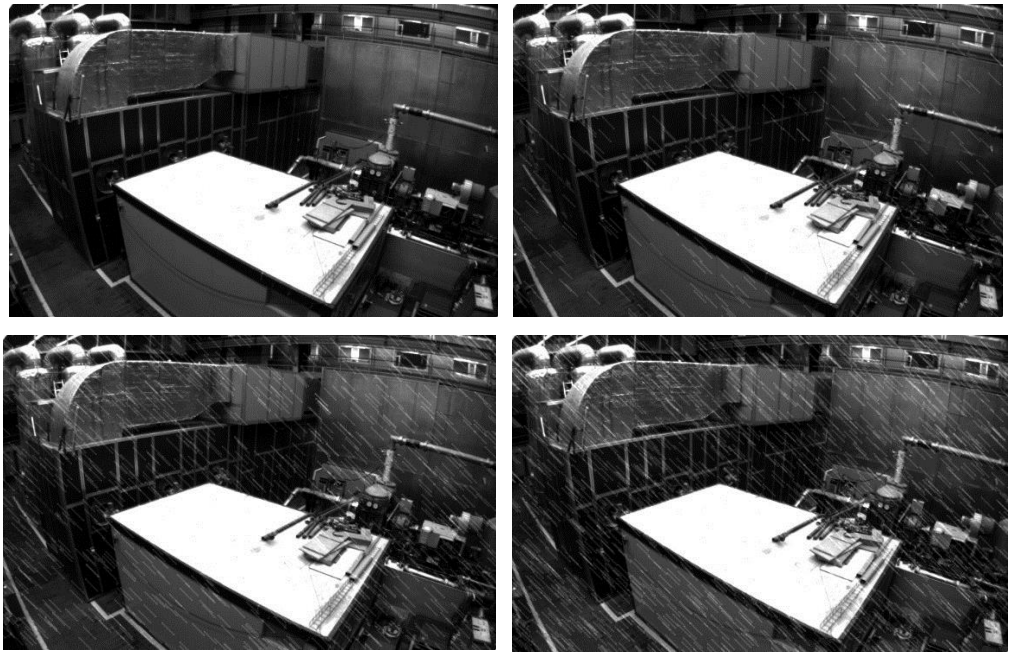

图4-3 模拟下雨场景示意图

## 4.2 实验结果分析

从四个实验测试得到的算法轨迹图可得到：在没有引入雨滴等噪声情况下，各个算法的轨迹与真实值都很接近，即使没有后端的全局优化和回环检测，精度依旧满足定位需求；在噪声与信号比值为 0.001 情况下，ICE_BA 开始出现较大的定位误差，VINS-Mono 与本文改进系统的轨迹仍与真实值接近；

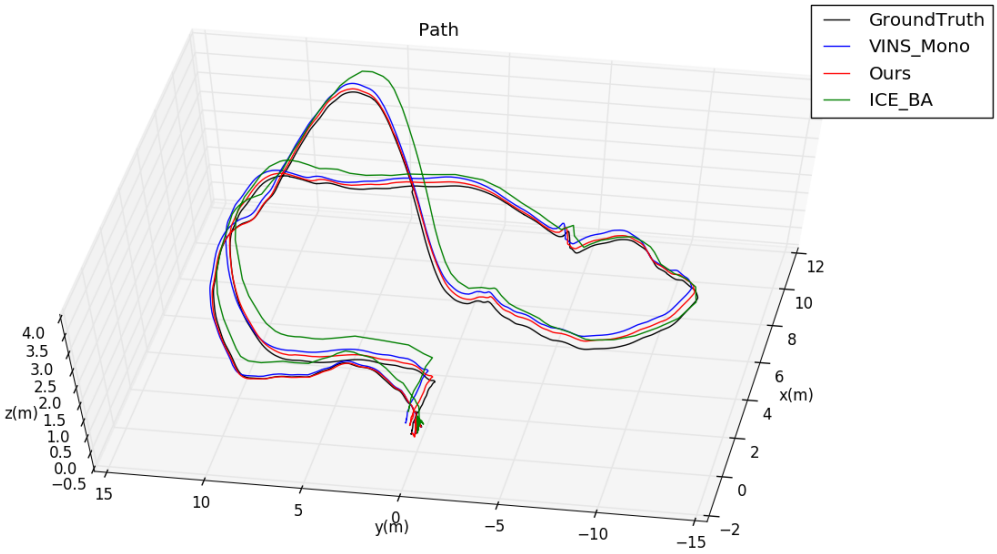


图 4-4 实验一中各算法轨迹图

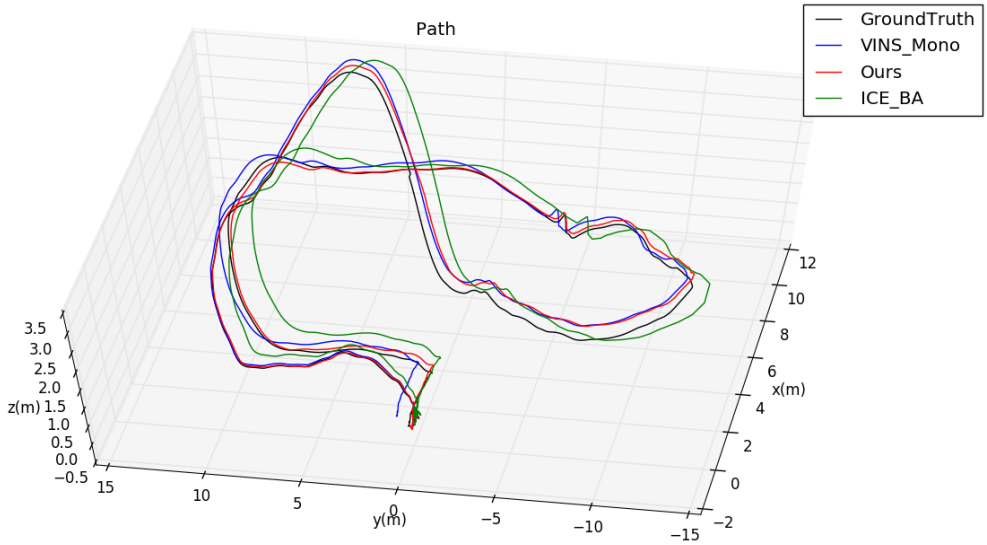


图 4-5 实验二中各算法轨迹图

当噪声与信号比值为 0.003 时，各个算法的特征点开始出现大的噪声，VINS-Mono 和 ICE_BA 的误差在程序最后都超过 1m，而本文所改进的系统依旧保持高精度定位；最后在噪声与信号比值为 0.005 的极端情况下，VINS-Mono 已经出现超过 3m 的定位误差，ICE_BA 运

行期间的最大误差甚至超过了 4m，而本文改进的系统虽然在运行中有些许抖动，但是仍然保持较高的定位精度。综合四个实验可以看到，相较于目前前沿的惯性视觉 SLAM 算法，本文所改进的算法在复杂和动态环境下有更好的鲁棒性。

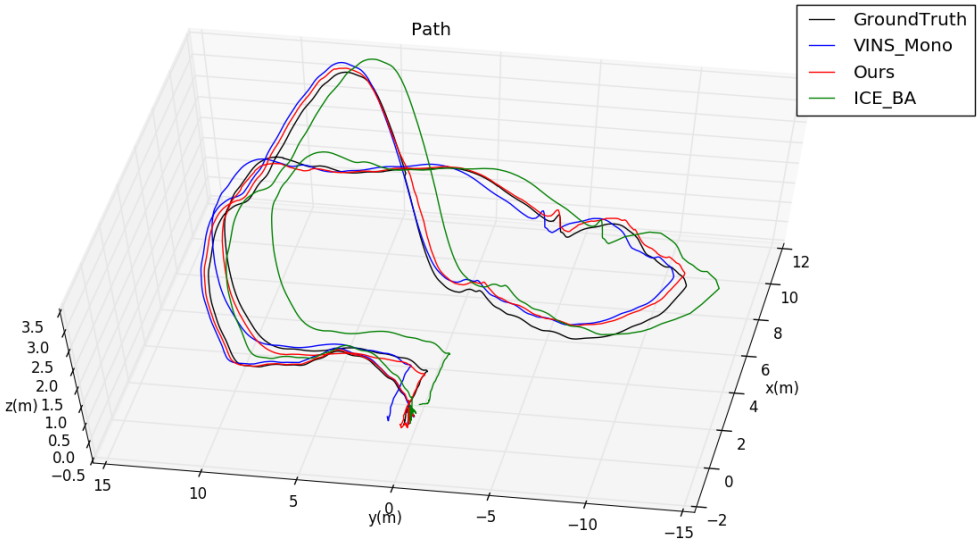


图 4-6 实验三中各算法轨迹图

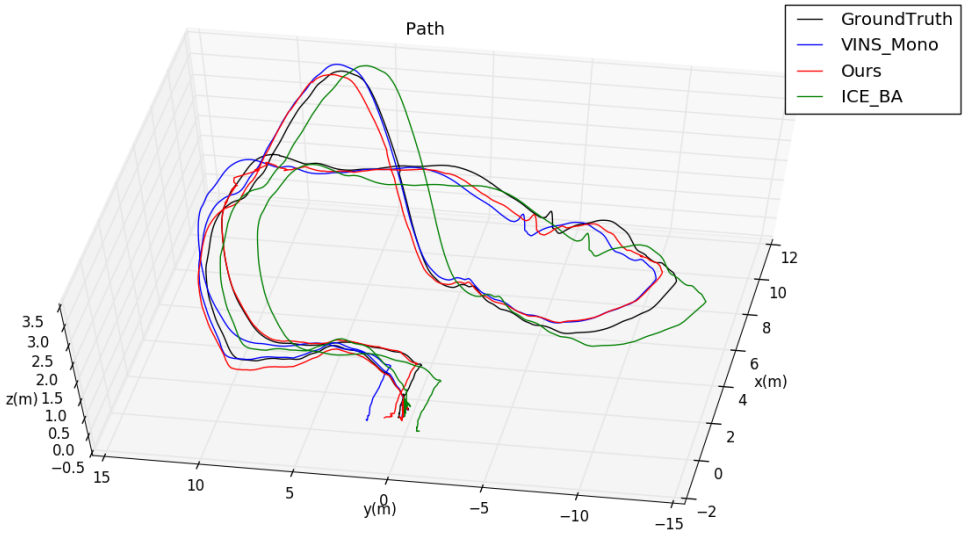


图 4-7 实验四中各算法轨迹图

为了更好地展现各个算法在运行过程中的误差变化情况，我们统计出了四个实验下各个算法每一帧的定位误差如图 4-8 所示。实验一中各个算法的定位误差均不超过 1m，但从曲线看出，本文改进的算法的误差最小。实验二 ICE_BA 算法运行中最大误差达到了 1m 以上，VINS-Mono 最大误差接近 1m，而本文提出的方法误差保持在 0.5m 以下。实验三 ICE_BA 算法最大定位误差超过 4m，VINS-Mono 定位误差则达到以上，本文提出的方法误差仍在 0.5m 以下。实验四 VINS-Mono 误差增大到了 3m 以上，ICE_BA 误差仍然超过 4m，本文方法误差则限制在 1m 左右。

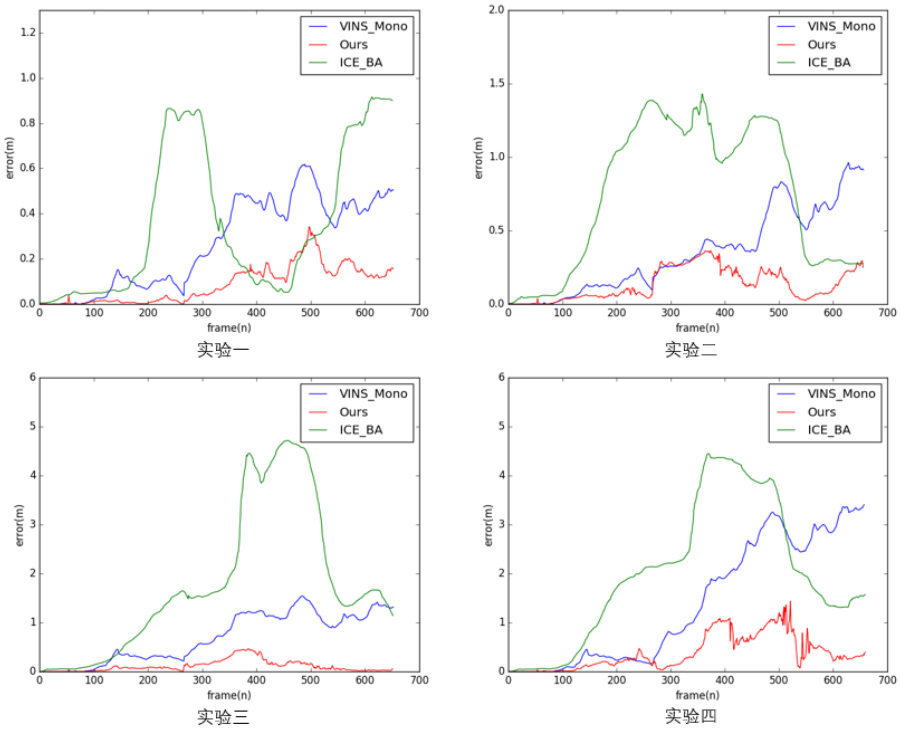


图 4-8 四个实验中各算法定位误差变化

最后我们统计四个实验中各个算法的平均误差，得到表 4-1。从表格看出，随着环境噪声的增大，各个算法的误差也随之增大，但是通过纵向比较发现，本文的所提出的系统对噪声的敏感度更加低，即使噪声达到 0.005，其平均定位误差依然限制在 0.5m。

| | 0.000 | 0.001 | 0.003 | 0.005 |
|---|---|---|---|---|
| ICE_BA | 0.356 | 0.744 | 1.799 | 1.994 |
| VINS_Mono | 0.265 | 0.346 | 0.698 | 1.389 |
| Ours | 0.085 | 0.134 | 0.111 | 0.373 |

表 4-1 各算法的平均误差（m）

## 4.3 本章小结

本章使用 Euroc 数据集设置了四组实验，分别测试 ICE_BA，VINS-Mono，以及本文改进的算法在大噪声环境下的定位精度。实验结果显示我们改进的算法的前端定位精度最高。通过四个实验的纵向与横向比较，我们分析各个算法的在不同噪声程度下的表现，并得出结论：本文所改进的算法在恶劣的环境条件下的鲁棒性更高。

# 5. 总结与展望

## 5.1 全文工作总结

本文致力于提出一种基于云,并且在复杂的环境下能进行精确定位的惯性视觉 SLAM 系统。本文首先介绍了过去视觉 SLAM 的发展现状，列举近几年优秀的 SLAM 算法和分析他们的特点。随后详细讲解惯性视觉 SLAM 领域中前沿的 VINS-Mono 算法，从传感器数据处理到紧耦合，逐步分析各个步骤的功能和存在的必要性。接着针对目前 SLAM 系统所存在的问题，本文提出了基于云的惯性视觉 SLAM 框架，同时发挥传感器融合的优势，提出大噪声检测和权值优化机制，最后通过实验证明了本文基于 VINS-Mono 改进的算法更具鲁棒性。本文的主要贡献有以下三点：

（1）提出基于云的视觉 SLAM 回环检测框架，将 SLAM 的全局优化和回环检测程序放入云端计算，增大回环检测词汇模型，减少假阳性情况的出现，同时减少了机器人本地的计算成本。即使由于网络故障导致连接中断，机器人也能在复杂环境下，通过前端的程序做精确的定位。

（2）提出噪声点检测的思路，利用 IMU 信息作为先验知识，通过计算特征点的重投影误差，将大噪声点剔除，减少优化过程中误差的引入。另外，在设定阈值过程中考虑 IMU 信息的不确定性，以避免过于偏信 IMU 数据导致漂移现象的产生。

（3）提出权值优化机制。在优化过程中我们通过比较各个特征点的重投影误差，选择性地加入权值，评估哪些特征点对位置估计的价值更高。在特征点数量少的情况，我们额外分配更多权值，以达到校正 IMU 数据的效果，该机制可以缓解特征点数量过少导致定位错误的问题，同时提高整个系统的定位精度。

## 5.2 未来研究展望与方向

本文所提出的算法也存在下列不足之处：一，在大噪声环境中定位容易抖动，连续性不够好，从原理上分析我认为这是由于环境中噪声影响，特征点的权值变化过于剧烈，导致各帧优化的结果有较大的偏差；二，根据 IMU 检测噪声点是基于 IMU 数据准确的假设上建立的，可是实际情况中 IMU 也有一定误差，部分好的特征点也会被剔除，同时还容易造成特征点多样性丢失，无法校正 IMU 的误差。对于上述问题，以后的工作主要有以下两个方向：

（1）使用深度学习提取图像中高维的语义。目前大部分 SLAM 的特征点提取和匹配仅仅局限在几何信息上，属于像素级别的 SLAM。而更高级的语义 SLAM 目前才刚刚开始发展。在计算力强大的云服务器和 5G 通信的加持下，机器人只需上图片到云端，调用神经网络模型提取语义信息辅助视觉 SLAM 定位，如图像分割检测动态物体，利用目标检测建立基于物体级别的 SLAM 系统。

（2）引入边缘计算，构建多层级的分布式 SLAM 系统。机器人与云服务器之间存在信息的传输成本，多个机器人的传感器信息同时传入到云端会造成信息拥堵，导致系统实时性降低。因此可以把机器人附近的服务器作为边缘设备，用以分担部分计算任务，减少运输成本和云服务器的负担。

# 致谢

回顾整个毕业设计的历程，我受到身边人很多的帮助，首先我要感谢范衡教授。在毕业设计定题时，他鼓励我研究前沿课题，并给予许多新的思路。另外在实验器材方面，他所成立的实验室为我提供了很大的帮助。给我提供了很多本着学术严谨性的态度，他对我的项目各个细节都进行严格把关，让我深刻明白作为一个学术研究者，如何设计对比实验，写一篇合格的论文。在此我由衷向范衡教授表达感谢！

其次我要感谢陈文钊，游煜根，朱贵杰师兄多年来的指导。从大二开始，我加入了 NEO 机器人俱乐部，并在这家庭中度过了两年多的光阴。期间，在师兄们的带领下，我们共同研究了许多前沿课题，如机器人导航，SLAM，深度学习，机器人探索等，并收获了许多研究成果。而我的科研能力有此过程中得到快速增长，为我继续深造打下了坚实的基础。

最后我还要感谢我的家人和女朋友在背后给我的关怀，她们给了我许多精神上的鼓励和支持，让我能全身心地投入到项目中来，在面对困难时让我有足够的决心继续走下去。无论何时何地，她们都是我生活中最重要的人。

本文在撰写及设计实验时参考了大量文献及书籍，并引用了部分资料。谨向这些书刊资料的作者表示衷心的感谢！

## 参考文献:


[1]. Tong, Qin , L. Peiliang , and S. Shaojie . "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator." IEEE Transactions on Robotics (2018):1-17.

[2]. Davison, Andrew J., et al. ”MonoSLAM: Real-time single camera SLAM.” IEEE transactions on pattern analysis and machine intelligence 29.6 (2007): 1052-1067.

[3]. Klein, Georg, and David Murray. ”Parallel tracking and mapping for small AR workspaces.” Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on. IEEE, 2007.

[4]. Jakob Engel, Thomas Schps, and Daniel Cremers. "LSD-SLAM: Large-Scale Direct Monocular SLAM." European Conference on Computer Vision Springer, Cham, 2014.

[5]. Mur-Artal, Raúl, J. M. M. Montiel, and J. D. Tardós. "ORB-SLAM: A Versatile and Accurate Monocular SLAM System." IEEE Transactions on Robotics 31.5(2015):1147-1163

[6]. Jones, Eagle S., and Stefano Soatto. "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach." The International Journal of Robotics Research 30.4 (2011): 407-430.

[7]. Munguía, Rodrigo, et al. "A visual-aided inertial navigation and mapping system." International Journal of Advanced Robotic Systems 13.3 (2016): 94.

[8]. Mourikis, A.I.; Roumeliotis, S.I. A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation. In Proceedings of the IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007; pp. 3565–3572.

[9]. Leutenegger, S., et al.Keyframe-Based Visual-InertialSLAM using Nonlinear Optimization. In Proceedings of the Robotics: Science and Systems, Berkeley, CA,USA, 12–16 July 2014; pp. 789–795.

[10]. Mur-Artal, R.; Tardós, J.D. Visual-Inertial Monocular SLAM with Map Reuse. IEEE Robot. Autom. Lett. 2017,2, 796–803.

[11]. Campos, Carlos, Montiel José MM, and Juan D. Tardós. "Fast and Robust Initialization for Visual-Inertial SLAM." 2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019.

[12]. Chen, Changhao, et al. "Selective sensor fusion for neural visual-inertial odometry." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019.

[13]. Han, Liming, et al. "DeepVIO: Self-supervised deep learning of monocular visual inertial odometry using 3d geometric constraints." arXiv preprint arXiv:1906.11435 (2019).

[14]. Babu, Benzun Pious Wisely, et al. "Detection and resolution of motion conflict in visual inertial odometry." 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018.

[15]. Kuffner J J, Lavalle S M. Space-filling trees: A new perspective on incremental search for motion planning[C]//Proceedings of IEEE International Conference on Intelligent Robots and Systems. IEEE, 2011:2199-2206.

[16]. Waibel M, Beetz M, Civera J, et al. RoboEarth - A World Wide Web for Robots[J]. 2011, 18(2):69-82.

[17]. Liu, Boyi , et al. "Lifelong Federated Reinforcement Learning: A Learning Architecture for Navigation in Cloud Robotic Systems." (2019).

[18]. Raffaele Limosani , et al. "Enabling Global Robot Navigation Based on a Cloud Robotics Approach." International Journal of Social Robotics 8.3(2016):371-380.

[19]. Sola, Joan. "Quaternion kinematics for the error-state Kalman filter." arXiv preprint arXiv:1711.02508 (2017).

[20]. Forster, Christian, et al. "On-Manifold Preintegration for Real-Time Visual--Inertial Odometry." IEEE Transactions on Robotics 33.1 (2016): 1-21.

[21]. Babu, B. P. W., Cyganski, D., Duckworth, J., & Kim, S. (2018). Detection and Resolution of Motion Conflict in Visual Inertial Odometry. 2018 IEEE International Conference on Robotics and Automation (ICRA). doi:10.1109/icra.2018.8460870.

[22]. Burri, Michael , et al. "The EuRoC micro aerial vehicle datasets." The International journal of robotics research 35.10(2016):1157-1163.

[23]. Liu, Haomin , et al. "ICE-BA: Incremental, Consistent and Efficient Bundle Adjustment for Visual-Inertial SLAM." IEEE/CVF Conference on Computer Vision & Pattern Recognition IEEE, 2018.