

哈爾濱工業大學

本科毕业论文（设计）中期报告

题 目：基于 GPS 数据融合的开阔场景下
双目视觉里程计研究

专 业 机器人工程

学 生 李明谦

学 号 8200880133

指导教师 佟明斯

日 期 2024 年 3 月 15 日

哈尔滨工业大学教务处制

目 录

1. 开题报告预定的研究内容及进度安排.....	3
1.1 研究内容	3
1.1.1 基于 ORB 算法的特征点检测与跟踪的方法研究.....	3
1.1.2 基于滤波的双目视觉惯性里程计的实现方法研究.....	3
1.1.3 基于双目视觉惯性里程计和 GPS 数据融合的方法研究	4
1.2 进度安排	4
2. 已完成的研究工作及成果	4
2.1 ORB 特征点的检测与跟踪.....	4
2.1.1 角点检测.....	6
2.1.2 特征点描述.....	8
2.1.3 双目相机特征点.....	9
2.1.4 实验效果.....	10
2.2 基于滤波的双目视觉惯性里程计.....	10
2.2.1 问题定义.....	11
2.2.2 传感器因子.....	12
2.2.3 优化.....	15
2.2.4 边缘化.....	15
2.2.5 实验效果.....	16
3. 后期拟完成的研究工作和进度安排.....	17
4. 存在的问题与困难.....	17
5. 论文按时完成的可能性.....	17

1. 开题报告预定的研究内容及进度安排

1.1 研究内容

1.1.1 基于 ORB 算法的特征点检测与跟踪的方法研究

无人机在飞行过程中需要较高的定位精度，所以对于特征点的检测与跟踪直接关系到无人机飞行过程中定位结果的好坏。在本课题的研究中，由于开阔场景下无人机飞行姿态的变化，所以考虑在现有 ORB（Oriented FAST and Rotated BRIEF）算法的基础上，通过引入一些识别和优化算法，以帮助无人机飞行姿态变化中检测和描述在不同角度，不同方向和不同尺度上的特征点，同时 ORB 算法本身就对图像噪声和一定程度的遮挡具有较强的鲁棒性，这也能够帮助无人机在飞行过程中对不同视角的物体进行更好的识别和跟踪；此外，由于无人机在工作时飞行速度较快，会导致检测到的特征点出现尺度上的一些变化，描述子出现噪声过大等问题。本课题打算对 ORB 算法使用的 FAST（Features from Accelerated Segment Test）角点检测算法和 BRIEF（Binary Robust Independent Elementary Features）二进制描述子进行一些优化，例如在使用 FAST 角点检测时构建高斯金字塔使其具有尺度不变性，在使用 BRIEF 二进制描述子时可以使用高斯卷积核进行滤波降噪，再加之这些描述子本身计算效率高和空间紧凑的特性，可以极大地减少存储和匹配成本，有利于无人机在飞行过程中出色地计算实时定位，从而得到较高定位精度的视觉里程计。

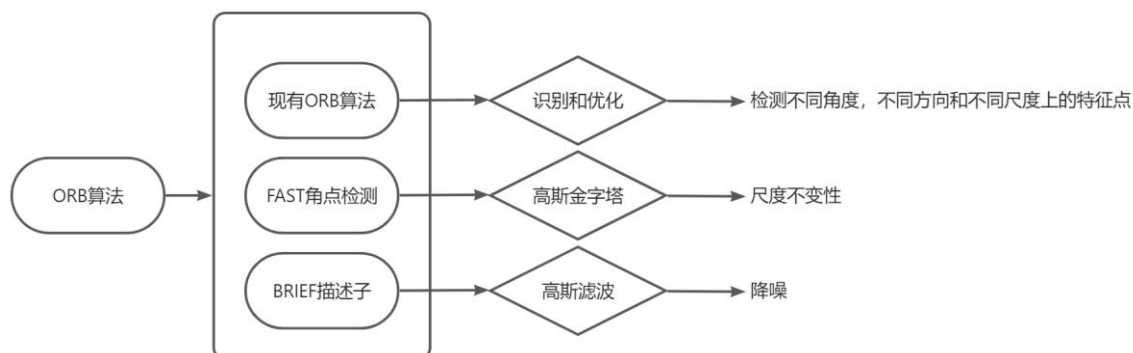


图 1-1 ORB 算法研究内容过程图

1.1.2 基于滤波的双目视觉惯性里程计的实现方法研究

无人机在开阔场景下的飞行过程中，由于飞行姿态会不断发生变化，仅依赖双目相机的数据是不足以实现对无人机自身位姿的准确估计的。这种情况可能导致无人机在自身定位方面出现偏差。然而，通过将无人机上搭载的 IMU（惯性测量单元）传感器的数据与双目相机获取的数据进行融合，采用基于扩展卡尔曼滤波方法，可以实现更为精确的自身位姿状态估计，从而获得更高的定位精度，实现了在无人机上应用双目视觉惯性里程计的目标。

在这个融合过程中，IMU 传感器提供了关于无人机三轴加速度和角速度的信息，视觉里程计则通过 1.1.1 中的研究已经达到了较高的定位精度。通过将这两种不同类型的数据结合起来，在实际操作过程中则可以通过不断改变扩展卡尔曼滤波算法的参数，使其可以

更好地校正由于飞行姿态变化引起的误差，从而提供更准确的位姿估计，帮助无人机更稳定地执行自主导航，避障，目标跟踪等任务，从而更好地完成电力输电巡检作业。

1.1.3 基于双目视觉惯性里程计和 GPS 数据融合的方法研究

在获取了搭载在无人机上的双目视觉惯性里程计数据之后，需要考虑到本研究所涉及的无人机操作环境的特殊性，即户外广阔环境。在这种环境下，仅仅依靠 1.1.2 研究中得到的双目视觉惯性里程计可能会带来一些挑战。这是因为无人机在户外环境中的飞行速度可能会非常快，而所覆盖的空间范围也相对较大，这导致双目相机所能够检测和跟踪的特征点数量相对较少。从而降低双目视觉惯性里程计的准确性。

因此，在这种情况下，本研究考虑采用一种更综合的方法，将双目视觉惯性里程计数据与 GPS（全球定位系统）数据相结合，利用扩展卡尔曼滤波技术来融合这两种不同类型的数据源。这种数据融合的方式可以帮助实现更高精度的定位结果，特别是在户外环境中，其中 GPS 通常提供了相对准确的位置信息。通过将双目视觉惯性里程计和 GPS 数据结合起来，无人机可以更可靠地确定自身位置，克服了双目相机在特征点有限的情况下定位不精准的限制。这种融合方法可以提高无人机的导航精度，确保它可以在广阔的户外环境中完成电路巡检任务，而不会受到位置不确定性的干扰。

1.2 进度安排

- （1）2023.10—2023.12：收集资料，完成文献查阅；
- （2）2023.12—2024.1：确定 ORB 算法的特征点检测与跟踪的实现方法，初步完成 ORB 特征点检测与跟踪的代码编写；
- （3）2024.1—2024.2：确定双目视觉惯性里程计的实现方法，初步完成双目视觉惯性里程计的代码编写；
- （4）2024.2—2024.3：确定双目视觉惯性里程计和 GPS 数据融合的方法，初步完成数据融合的代码编写；
- （5）2024.3—2024.5：在无人机平台上运行相关算法和代码，并根据运行效果不断改进算法和代码，实现最终效果和各部分的优化
- （6）2024.5—2024.6：整理结果，完成毕业论文。

2. 已完成的研究工作及成果

2.1 ORB 特征点的检测与跟踪

ORB（Oriented FAST and Rotated BRIEF）算法是一种用于特征点检测和描述的计算机视觉算法。它于 2011 年由 Ethan Rublee 等人在论文《ORB: An efficient alternative to SIFT or SURF》中提出。ORB 算法主要用于在图像中检测关键点，并计算这些关键点的描述符，从而实现图像特征匹配、目标检测、图像拼接等任务，其主要包括以下几个步骤：

（1）FAST（Features from Accelerated Segment Test）特征点检测器

FAST 算法的核心在于检测图像中的角点。如果一个像素点与其周围足够多的像素点

相比，其灰度值存在明显的差异，那么这个像素点就可能被认为是一个角点。FAST 算法通过设定一个阈值，并比较像素点与其周围像素点的灰度值，从而确定哪些点可能是角点。这种方法的优点是速度快，但缺点是缺乏尺度不变性。ORB 算法通过构建高斯金字塔，并在每一层金字塔图像上进行 FAST 角点检测，从而实现了尺度不变性。

（2）BRIEF（Binary Robust Independent Elementary Features）描述符

BRIEF 描述符是一种二进制描述符，它将关键点周围的像素对比作为特征向量。其核心思想在于利用随机选取的像素对来构建特征点的描述子。具体而言，BRIEF 算法通过在特征点附近随机选取一组像素对，并根据这些像素对的灰度值大小生成一个二进制串作为特征描述子。这种方法的显著优势在于其高效的速度，因为它可以通过简单的灰度比较操作来生成特征描述子，从而迅速地完成任务的检测和描述。

然而，尽管 BRIEF 算法具有速度快的优点，但其仍然存在一些显著的缺陷。首先，BRIEF 算法缺乏旋转不变性，这意味着当图像发生旋转时，同一物体在不同角度下可能会具有不同的特征描述子，导致特征匹配的困难。其次，BRIEF 算法对图像中的噪声相当敏感，这是由于特征描述子的生成过程中所选取的像素对是随机的，而噪声可能导致选取的像素对不稳定，进而影响特征描述子的质量。此外，BRIEF 算法也不具备尺度不变性，因为特征描述子是在固定尺度上生成的，因此对于尺度变化较大的图像，同一物体可能会具有不同的特征描述子，进而影响特征匹配的准确性和可靠性。ORB 算法通过构建高斯金字塔，在保留 BRIEF 算法高效性的同时，增加了尺度不变性等特性；本次研究通过高斯滤波进行降噪处理，使得 BRIEF 算法在一定程度上缓解了噪声敏感问题。

（3）快速近似匹配

在 ORB（Oriented FAST and Rotated BRIEF）算法中，快速近似匹配是一项关键技术，用于加速特征点的匹配过程。该算法采用了汉明距离作为衡量两个特征点之间相似性的度量标准。通过计算两个特征点的描述子之间的汉明距离，ORB 算法可以确定它们之间的相似程度，进而进行特征点的匹配。

为了提高匹配的 efficiency，ORB 算法采用了一种快速的近似匹配方法。这种方法旨在在保证匹配质量的同时，尽可能地减少匹配时间。在进行近似匹配时，ORB 算法通常会采用一些优化策略，例如使用哈希表或树结构来组织特征描述子，以便快速地搜索最相似的特征点。此外，ORB 算法还可能对描述子进行降维或采样，以进一步提高匹配的速度。

通过快速近似匹配方法，ORB 算法能够在处理大规模图像数据时实现高效的特征点匹配。这种匹配方法不仅能够满足实时性的要求，同时也能够保证匹配的准确性和可靠性，使得 ORB 算法在各种计算机视觉应用中都能够取得良好的性能表现。

总的来说，ORB 算法是一种高效的特征点检测和描述算法，它综合了 FAST 特征点检测器和 BRIEF 描述符，并引入了旋转不变性来提高算法的鲁棒性。在本节完成的研究工作中，主要参考了前人提出的 ORB 算法，同时结合自身研究项目的实际需求，在 ORB 算法的基础上进行了一定程度的修改，使得其在实际运行过程中能更精准，稳定地检测和跟踪特征点，为之后视觉里程计的准确性和稳定性奠定基础。

2.1.1 角点检测

在这一部分中，主要通过 FAST 算法和 Harris 角点检测算法结合来进行角点检测。

(1) Harris 角点

Harris 角点是计算机视觉中用于检测图像局部特征点的一种方法。角点定义为图像中灰度变化剧烈的点，或者在两个主方向上的特征点。在图像处理中，这些点通常对应于物体的边缘或角落，具有明显变化的像素梯度。

Harris 角点检测算法的基本思想是使用一个固定窗口在图像上进行任意方向上的滑动，比较滑动前与滑动后窗口中像素的灰度变化程度。如果存在任意方向上的滑动都使得窗口中像素的灰度发生了较大的变化，那么这个窗口的中心就被认为是一个角点。

Harris 角点检测算法的具体步骤如下：

①、计算图像的梯度：通过对图像进行梯度计算，获取图像中每个像素的梯度幅值和方向。

②、计算自相关矩阵：对于每个像素，计算其周围窗口内的梯度幅值和方向的自相关矩阵。

③、计算角点响应函数：使用自相关矩阵计算每个像素的角点响应函数，该函数度量了像素周围区域的角点特征，其中角点响应函数公式为：

$$R = \det M - \alpha(\text{trace}M)^2 \quad (2-1)$$

④、非极大值抑制：对于每个像素，比较其角点响应函数与周围像素的响应函数，保留局部最大值作为角点。

Harris 角点检测算法的优点在于其稳定性和对噪声的鲁棒性，能够准确地定位图像中的角点，并且具有较高的计算能力。因此，在本部分研究工作中将其与 FAST 算法结合共同进行角点检测，以提高系统检测的精确性和鲁棒性。

(2) FAST 算法与 Harris 角点检测算法结合

在这部分研究工作中，首先利用 FAST 算法来初步检测图像中的潜在角点，作为候选特征点。然而，仅仅依靠 FAST 算法检测出的特征点可能并不足以满足后续应用的需求。然后再利用 Harris 角点的度量方式，通过计算图像中每个像素点的角点响应值，来衡量该点是否具备角点的特性。具体来说，Harris 角点响应函数考虑了像素点周围灰度值的变化情况，如果周围灰度值变化剧烈，则该点的响应值就会较高，表明其更有可能是一个角点。通过 Harris 角点响应函数计算出每个候选特征点的响应值，然后按照响应值的大小对候选特征点进行排序，并筛选出响应值最大的前 N 个特征点作为最终的输出。通过这种方式，能够确保系统提取出的特征点不仅数量适中，而且具有较高的代表性和稳定性，为后续的特征描述和匹配提供了坚实的基础。

针对 FAST 检测特征点不具备尺度不变性的难题，本次研究工作引入了尺度空间理论。尺度空间理论是一种描述图像在不同尺度下特性的方法，它通过对图像进行不同尺度的平滑处理，构建出一个图像的金字塔结构，每一层金字塔都代表了图像在不同尺度下的表示。通过这种方式，我们可以同时获取图像在不同尺度下的特征信息，从而实现尺度不变性。

本次研究工作中，首先构建了图像的高斯金字塔。高斯金字塔是图像处理中一个重要的概念，它帮助我们以一种结构化的方式观察和分析图像在不同尺度下的特征。这一过程是通过对原始图像进行一系列精心设计的操作来实现的，其中最为关键的就是高斯滤波和下采样。

首先，高斯滤波是构建高斯金字塔的基础步骤之一。高斯滤波是一种线性平滑滤波，其核心思想是对图像中的每一个像素点，根据其邻域内像素的灰度值分布，进行加权平均，从而得到一个新的像素值。这种滤波方式能够有效地去除图像中的高频噪声和细节，使得图像变得更为平滑。在高斯金字塔的构建中，原始图像会进行多次高斯滤波，每次滤波都会使图像变得更加模糊，从而得到一系列不同模糊程度的图像。

接下来，下采样是构建高斯金字塔的另一个关键步骤。下采样是通过减少图像的尺寸来实现的，通常是通过每隔一定数量的像素取一个像素值，或者对一定区域内的像素值进行平均等方式来实现。下采样的结果是图像的尺寸变小，细节信息减少，但主要的结构和特征仍然得以保留。在高斯金字塔中，每一次高斯滤波后的图像都被进行了下采样操作，从而得到一系列尺寸逐渐减小的图像。

通过高斯滤波和下采样这两个步骤的交替进行，最终可以得到一个高斯金字塔，如下图 2-1 所示。

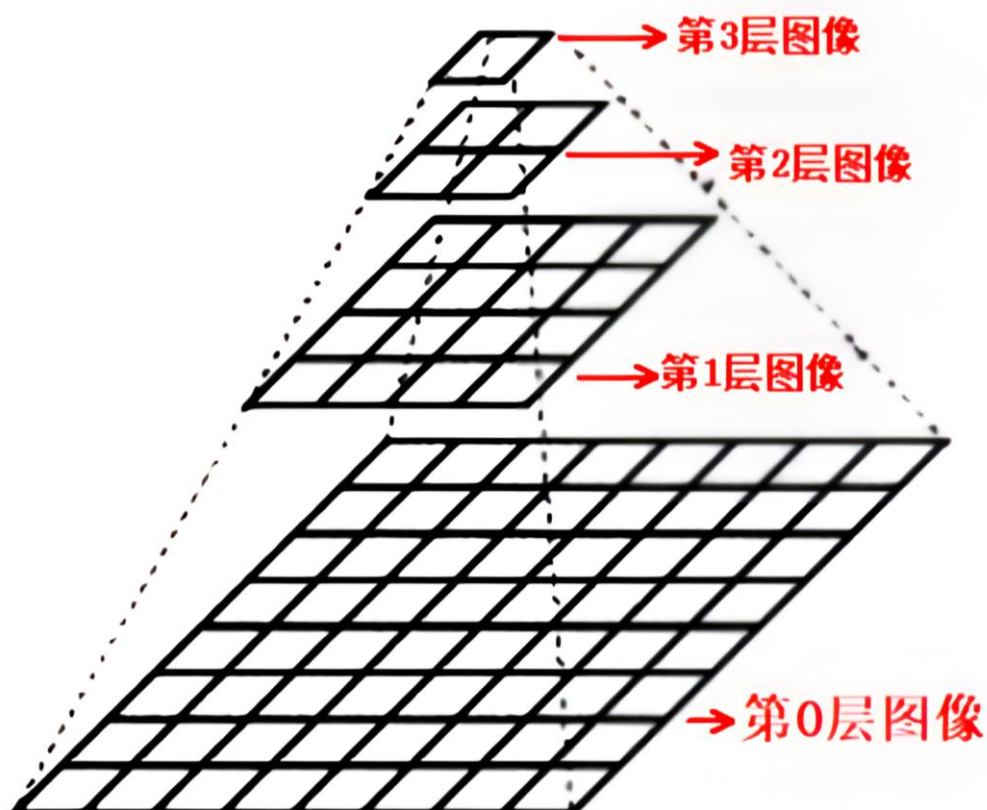


图 2-1 高斯金字塔

这个金字塔由多层图像构成，每一层都代表了原始图像在不同尺度下的表示。随着金字塔层数的增加，图像的尺寸逐渐减小，细节信息逐渐模糊，但主要的结构和特征仍然得以保留。

有了高斯金字塔之后，我们就可以在每一层图像上分别应用 FAST 算法和 Harris 角点检测算法结合进行角点检测。由于高斯金字塔的每一层都代表了图像在不同尺度下的表示，因此，在每一层上应用 FAST 算法和 Harris 角点检测算法检测到的角点都具有尺度不变性。因此无论图像的尺寸或分辨率如何变化，都能够在高斯金字塔的相应层上找到稳定的特征点。

除了尺度不变性外，FAST 特征点还存在另一个问题，即不具备旋转不变性。这意味着当图像发生旋转时，FAST 算法检测到的特征点可能会发生变化，导致特征匹配失败。为了解决这一问题，原论文中提出了一种利用灰度质心法来确定特征点方向的方法，从而实现了带有方向的 FAST 角点，即 oFAST。

灰度质心法的基本思想是，对于每一个 FAST 特征点，我们首先要确定一个半径范围，这个范围通常根据图像的具体情况和特征点的性质来设定。然后，在这个半径范围内，计算所有像素的灰度值的加权平均位置，即灰度质心。通过计算得到的灰度质心，实际上代表了该特征点周围像素灰度值的分布中心。由于灰度质心是根据灰度值进行计算的，因此它对于图像中的亮度变化非常敏感。这使得灰度质心法能够准确地反映出特征点周围像素的灰度分布特性。对于任意一个特征点 p 来说，定义 p 的邻域像素的矩为：

$$m_{pq} = \sum_{x,y \in \mathcal{O}_r} x^p y^q I(x,y) \quad (2-2)$$

其中 $I(x,y)$ 表示像素灰度值， $\mathcal{O}_r$ 是半径为 r 离散化的 Bresenham 圆形窗口，零阶矩 m_{00} 即图像领域窗口内所有像素的灰度和， m_{10} 和 m_{01} 分别是相对于 x 与 y 的一阶矩，因此图像局部邻域的中心矩或者质心可定义为： $C = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right)$ 。

在找到灰度质心后，继续计算从特征点到灰度质心的向量。这个向量不仅包含了特征点与质心之间的位置关系，还隐含了灰度分布的方向信息。因此，我们可以将这个向量定义为该特征点的主方向。主方向确定后，即使图像发生了旋转，只要特征点的主方向也随之旋转，仍然可以通过比较不同特征点的主方向来判断特征点之间的对应关系。oFAST 特征点的方向矢量定义为特征点的中心 O 与质心 C 形成的向量 $\overrightarrow{OC}$ 与 X 轴的夹角，即：

$$\theta = \arctan(m_{01}, m_{10}) \quad (2-3)$$

2.1.2 特征点描述

BRIEF 特征描述子是一种对已检测到的特征点进行表示和描述的特征描述方法，和传统的利用图像局部邻域的灰度直方图或梯度直方图提取特征的方式不同，BRIEF 是一种二进制编码的特征描述子，经过上一步的角点检测后，利用 BRIEF 算法建立特征描述符。其具体算法为：在特征点周围的邻域（通常称为 Patch）内随机选择若干对像素点。这些点对的选择并非任意，而是遵循一定的统计规律，以确保它们能够充分反映邻域内的图像纹理和结构信息。随后，算法会比较每一对点的灰度值大小，根据比较结果生成二进制编码。一般而言，如果一对点中第一个点的灰度值大于第二个点，则编码为 1；反之，则编码为 0。灰度测试 τ 定义如下：

$$\tau(p; \mathbf{x}, \mathbf{y}) \doteq \begin{cases} 1 & p(\mathbf{x}) < p(\mathbf{y}) \\ 0 & p(\mathbf{x}) \geq p(\mathbf{y}) \end{cases} \quad (2-4)$$

式中 $p(\mathbf{x})$ 是图像块上点 $X = (u, v)^T$ 处的灰度值。通过这种方式，BRIEF 算法为每个特征点生成了一个独特的二进制字符串，即特征描述符。这种描述符不仅具有极小的存储开销，而且由于其二进制特性，使得特征匹配过程变得极为高效。

在匹配阶段，算法利用汉明距离来衡量不同特征描述符之间的相似性。汉明距离是指两个等长二进制字符串在对应位置上不同字符（通常是 0 和 1）的个数，非常适合于快速比较二进制特征描述符。其计算过程简单直接，对于两个等长的二进制字符串，我们只需逐一比较它们对应位置上的字符。每发现一个不同的字符，就将计数器加 1。最终，计数器的值就是这两个字符串之间的汉明距离。这种逐位比较的方式不仅易于实现，而且能够确保结果的准确性。在特征匹配中，汉明距离的应用使得系统可以快速找到与给定特征描述符最为接近的其他描述符。通过设定一个阈值，我们可以筛选出汉明距离小于该阈值的所有候选匹配项，从而进一步缩小匹配范围，提高匹配的准确性和效率。本节研究通过对所有的 n 个点进行比较，生成了一个 n 长的二进制串：

$$f_n(p) \doteq \sum_{1 \leq i \leq n} 2^{i-1} \tau(p; \mathbf{x}_i, \mathbf{y}_i) \quad (2-5)$$

对于 n 的取值可以设为 128、256 或者 512，在本次研究中经过测试 n 取值为 128、256 和 512 均对实验结果无明显影响。

在实际应用中，BRIEF 特征描述子凭借其高效和紧凑的特性具有明显的速度优势，但正如前文所说，BRIEF 描述子对噪声具有较高的敏感性。噪声是图像处理中常见的问题，它可能来源于图像采集设备的物理限制、环境干扰或者传输过程中的误差。这些噪声可能会改变图像中像素的灰度值，进而影响到特征点的检测和描述。对于 BRIEF 来说，噪声的存在可能导致随机选择的点对灰度值比较结果的不稳定，进而影响到最终生成的二进制特征描述符的准确性。针对此问题，本次研究中采用了高斯滤波技术对图像进行预处理。高斯滤波其基本原理是通过卷积操作对图像进行平滑处理。本次研究中将高斯核与图像的每个像素及其邻域进行卷积操作，这个过程实质上是将图像中的每个像素值替换为其邻域内所有像素的加权平均。由于高斯核的形状特性，靠近中心位置的像素会被赋予更高的权重，而远离中心的像素权重则逐渐减小，这种加权平均的方式使得图像中的噪声得到平滑处理，同时保留了图像的主要结构和边缘信息。通过以上操作，图像中的每个像素值被替换为其邻域内像素的加权平均，从而有效地平滑图像并减少噪声。这种降噪处理不仅保留了图像的主要结构和特征，还在一定程度上缓解了 BRIEF 特征描述子对噪声的敏感问题，提升了特征点检测与跟踪的准确性和稳定性，为后续的特征匹配和目标识别提供了更坚实的基础。

2.1.3 双目相机特征点

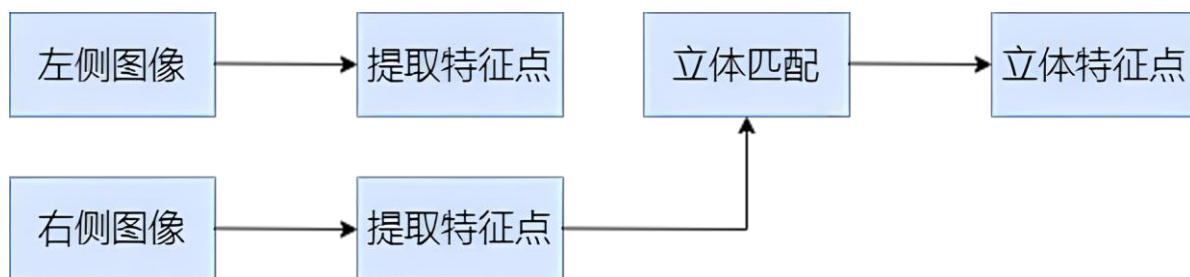


图 2-2 特征点流程图

首先双目相机获取图像，然后系统会对这两幅图像进行角点检测和特征描述。通过以上操作，对于双目相机传入的两幅图像，只要角点的特征描述相似，我们就可以认为它们是匹配的。之后系统处理这些立体特征点。立体特征点由三个坐标 $X_s = (u_L, v_L, u_R)$ 定义，其中 (u_L, v_L) 是左图像上的坐标， u_R 是右图像中的坐标。本节研究从两个图像中提取特征点，并且对于每个左侧特征点，在右侧图像中搜索匹配项。这一过程基于特征点描述的相似性来进行，确保了匹配的精确性。完成匹配后，通过使用左侧 ORB 的坐标和右侧匹配的坐标来生成立体特征点，这些特征点不仅包含了二维图像信息，还融入了深度信息，为我们提供了更全面的场景理解。由于双目相机的结构特点，研究中可以仅从一帧获取深特征点信息，不需要像单目相机那样从运动初始化中获得特定结构。系统启动时，利用第一帧创建一个关键帧，将其姿势设置为原点，并从所有立体特征点创建初始地图并获得其初始位置。

2.1.4 实验效果

通过以上操作，最终特征点检测与跟踪的实现效果如图所示。图 2-3 为利用 FAST 算法和 Harris 角点检测算法对 KITTI 数据集一张图像处理之后的结果，图 2-4 为手持双目相机在实验室内检测的结果：



图 2-3 KITTI 数据集上检测到的特征点

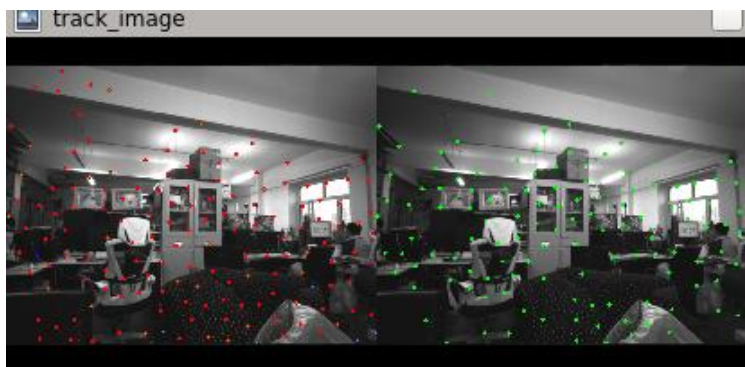


图 2-4 实验室内检测到的特征点

2.2 基于滤波的双目视觉惯性里程计

2.2.1 问题定义

(1) 状态定义

首先需要估计的状态包括无人机中心的三维位置和姿态，相机的三维位置或者估计视觉路标的深度以及 IMU 的变加速度计偏置和陀螺仪偏置，因此对于视觉和惯性传感器，整个状态定义如下：

$$\begin{aligned}\mathcal{X} &= [\mathbf{p}_0, \mathbf{R}_0, \mathbf{p}_1, \mathbf{R}_1, \dots, \mathbf{p}_n, \mathbf{R}_n, \mathbf{x}_{cam}, \mathbf{x}_{imu}] \\ \mathbf{x}_{cam} &= [\lambda_0, \lambda_1, \dots, \lambda_l] \\ \mathbf{x}_{imu} &= [\mathbf{v}_0, \mathbf{b}_{a_0}, \mathbf{b}_{g_0}, \mathbf{v}_1, \mathbf{b}_{a_1}, \mathbf{b}_{g_1}, \dots, \mathbf{v}_n, \mathbf{b}_{a_n}, \mathbf{b}_{g_n}]\end{aligned}\quad (2-6)$$

其中， $\mathbf{p}$ 和 $\mathbf{R}$ 为基本的系统状态，它们对应于世界坐标系下表示的机体位置和姿态。 $\mathbf{x}_{cam}$ 为与相机相关的状态，它包括第一帧中观测到的每个特征的深度 λ 。 $\mathbf{x}_{imu}$ 为与 IMU 相关的变量，它由速度 $\mathbf{v}$ 、加速度计偏置 $\mathbf{b}_a$ 和陀螺仪偏置 $\mathbf{b}_g$ 组成。传感器中心到机体中心的平移假定是已知的，它通过离线标定获得。为了简化符号，将 IMU 表示为机体中心。

(2) 代价函数

状态估计是对机器人的当前状态（如位置、姿态等）进行最优估计。其本质为一个最大似然估计（MLE）问题。MLE 是一种统计方法，它基于观测数据来估计模型参数，使得观测数据出现的概率最大。在机器人的状态估计中，MLE 问题通常涉及一段时间内机器人位姿的联合概率分布。这个联合概率分布包含了机器人在各个时间点的可能位置和姿态信息，以及它们之间的关联性。通过 MLE 方法，我们可以根据观测数据（如传感器读数、视觉特征点等）来推断机器人的最可能状态序列。在所有测量数据是独立的假设下，问题通常推导为：

$$\mathcal{X}^* = \underset{\mathcal{X}}{\operatorname{argmax}} \prod_{t=0}^n \prod_{k \in \mathbf{S}} p(\mathbf{z}_t^k | \mathcal{X}) \quad (2-7)$$

其中 $\mathbf{S}$ 为一组测量，它来自相机、IMU 和其它传感器。为了方便对其进行更好地数学处理，本节研究中假设测量的不确定性服从高斯分布， $p(\mathbf{z}_t^k, \mathcal{X}) \sim \mathcal{N}(\bar{\mathbf{z}}_t^k, \Omega_t^k)$ 。因此，上述公式的负似然对数可以写成：

$$\begin{aligned}\mathcal{X}^* &= \underset{\mathcal{X}}{\operatorname{argmax}} \prod_{t=0}^n \prod_{k \in \mathbf{S}} \exp\left(-\frac{1}{2} \|\mathbf{z}_t^k - h_t^k(\mathcal{X})\|_{\Omega_t^k}^2\right) \\ &= \underset{\mathcal{X}}{\operatorname{argmin}} \sum_{t=0}^n \sum_{k \in \mathbf{S}} \|\mathbf{z}_t^k - h_t^k(\mathcal{X})\|_{\Omega_t^k}^2\end{aligned}\quad (2-8)$$

因此状态估计被转换为一个非线性最小二乘问题，也被称为捆集调整（BA）。捆集调整（Bundle Adjustment）是一种在三维重建和计算机视觉中广泛使用的优化方法。它主要用于调整和优化相机参数（如内参和外参）以及三维空间点的位置，以最小化重投影误差。具体来说，捆集调整首先将所有的观测数据（包括图像特征点的像素坐标和相机的内外参数）进行整理，形成一个大的优化问题。然后，利用非线性优化算法（如 Levenberg-Marquardt 算法）对这个优化问题进行迭代求解，逐步调整相机参数和三维点坐标，直到达到一个满足条件的收敛状态。在本节双目视觉里程计的研究中，会通过多个视

角的图像来确定当前相机所处的位置。这些图像之间会存在大量的同名特征点，即在不同图像中都能识别到的同一物理点。捆集调整的目标就是对这些特征点的三维坐标以及相机的内外参数进行全局优化，使得这些特征点在三维空间中的投影与它们在图像中的位置尽可能一致。

2.2.2 传感器因子

(1) 相机因子

每个相机的内参和相机之间的外参变换都可以通过标定进行获取，这些参数的精确性直接关系到从图像中恢复出的场景三维结构的准确性。在相机获取的图像中，角点特征具有稳定性，使得它们在不同的视角和光照条件下都能保持相对稳定。

为了追踪这些特征点在连续帧之间的变化，我们采用了 KLT 跟踪器。KLT 跟踪器基于光流法，能够在前一帧中准确跟踪到当前帧的对应特征点。这种跟踪方法不仅高效，而且能够在复杂环境下保持较高的跟踪精度。对于双目相机系统，还需要进行特征匹配，即需要在左图像和右图像之间找到对应的特征点。通过双目匹配可以进一步确定特征点的三维位置，进而获得特征点的轨迹来构建相机因子。具体来说，相机因子本质上执行的是一个重投影过程。这个过程起始于对某个特征的首次观测。一旦我们捕获到了这个特征在某帧中的确切位置，就可以利用已知的相机参数和三维空间模型，尝试将这个特征投影到后续的帧中。这本质上是一个逆向操作，将三维空间中的点映射回二维图像平面。在实际运行过程中，由于各种因素（如相机抖动、光照变化、遮挡等）的影响，实际观测到的特征位置与通过重投影得到的特征位置往往存在偏差。这个偏差就被称为重投影误差。重投影误差的大小直接反映了我们当前相机参数和三维点位置的准确性。

为了减小这个误差，本次研究中通过迭代的方法对相机参数和三维点位置进行优化。在每一次迭代中，我们都会根据当前的重投影误差调整相机参数和三维点位置，然后重新计算重投影误差。这个过程会不断重复，直到重投影误差达到规定的阈值范围内，或者优化算法达到收敛状态。通过以上迭代优化过程，我们可以使得实际观测到的特征位置与通过重投影得到的特征位置更加接近，从而提升三维重建的精度和鲁棒性。

考虑图像 i 中首次观测到的特征 l ，其在之后图像 t 中的观测残差定义为：

$$\begin{aligned} \mathbf{z}_t^l - h_t^l(\mathcal{X}) &= \mathbf{z}_t^l - h_t^l(\mathbf{R}_t, \mathbf{p}_t, \mathbf{R}_i, \mathbf{p}_i, \lambda_l) \\ &= \begin{bmatrix} u_t^l \\ v_t^l \end{bmatrix} - \pi_c(\mathbf{T}_c^{b^{-1}} \mathbf{T}_t^{-1} \mathbf{T}_i \mathbf{T}_c^b \pi_c^{-1}(\lambda_l, \begin{bmatrix} u_i^l \\ v_i^l \end{bmatrix})) \end{aligned} \quad (2-9)$$

其中， $\begin{bmatrix} u_i^l \\ v_i^l \end{bmatrix}$ 为出现在图像 i 中的特征 l 的首次观测。 $\begin{bmatrix} u_t^l \\ v_t^l \end{bmatrix}$ 为相同特征在图像 t 中的观测。

π_c 和 π_c^{-1} 为投影和反投影函数，它们依赖于相机模型（本次研究中选用的相机模型为针孔模型）。 $\mathbf{T}$ 为 4×4 的齐次变换矩阵： $\begin{bmatrix} \mathbf{R} & \mathbf{p} \\ 0 & 1 \end{bmatrix}$ 。为了简化表示，在这里忽略了一些齐次项。

$\mathbf{T}_c^b$ 为从机体中心到相机中心的外参变换，它通过离线标定获取。投影误差的协方差矩阵 $\mathbf{\Omega}_t^l$ 为像素坐标下的常值，它来自相机的内参标定结果。

该因子对于左相机和右相机都是通用的，因此能够在不同时间的空间中将左图像中的特征投影到另一左图像，也能够同一时间的空间中将左图像中的特征投影到右图像。

(2) IMU 因子

IMU 传感器具有一个三轴的陀螺仪和三轴加速度计，能够测量物体在三维空间中的角速度和加速度，并以此解算出物体的姿态变化。但是 IMU 传感器的数据会受到噪声的影响，如式(2-10)、(2-11)所示，通常将旋转角速度测量和加速度测量建模成测量值等于在真值的基础上叠加了高斯白噪声以及测量偏置的随机游走噪声。式(2-10)、(2-11)中 w_m 、 a_m 分别表示旋转角速度和加速度的测量值， w 、 a 表示真实值， b 、 n 分别表示传感器测量的随机游走噪声和高斯白噪声。

$$w_m = w + b_w + n_w \quad (2-10)$$

$$a_m = a + b_a + n_a \quad (2-11)$$

在三维空间中，通常将刚体运动描述为位置(Position)、速度(Velocity)、旋转(Rotation/Quaternion)等参数随时间的变化。如式(2-12)、(2-13)、(2-14)所示，分别表示在 t 时刻下，IMU 在三维空间的位置 p_{wb_t} 、速度 v_t^w 、四元素 q_{wb_t} 对时间的导数。其中加速度和旋转角速度可以由 IMU 测量真值 a 、 w 来表示。

$$\dot{p}^{wb_t} = v_t^w \quad (2-12)$$

$$\dot{v}_t^w = a_t^w \quad (2-13)$$

$$\dot{q}_{wb_t} = q_{wb_t} \otimes \begin{bmatrix} 0 \\ \frac{1}{2} w^{b_t} \end{bmatrix} \quad (2-14)$$

因此，对于每一时刻 IMU 的旋转角度和加速度的测量值使用中值积分，可以得到 IMU 传感器累积的位置、速度、旋转的变化。如式(2-15)、(2-16)、(2-17)所示，根据 k 时刻的状态，以及 k 时刻和 $k+1$ 时刻的测量，计算 $k+1$ 时刻的状态。如式(2-18)、(2-19)所示， a 、 w 分别表示 k 时刻和 $k+1$ 时刻的加速度和旋转角速度测量的平均值。其中注意加速度的测量要去掉自然重力加速度的影响。

$$p_{wb_{k+1}} = p_{wb_k} + v_t^w \Delta t + \frac{1}{2} a \Delta t^2 \quad (2-15)$$

$$v_{k+1}^w = v_t^w + a \Delta t \quad (2-16)$$

$$q_{wb_{k+1}} = q_{wb_k} \otimes \begin{bmatrix} 0 \\ \frac{1}{2} w \Delta t \end{bmatrix} \quad (2-17)$$

$$a = \frac{1}{2} [q_{wb_k} (a^{b_k} - b_{a_k}) - g^w + q_{wb_{k+1}} (a^{b_{k+1}} - b_{a_k}) - g^w] \quad (2-18)$$

$$w = \frac{1}{2} [(w^{b_k} - b_{w_k}) + (w^{b_{k+1}} - b_{w_{k+1}})] \quad (2-19)$$

IMU 传感器可以方便地测量运动物体的旋转角速度和加速度，通过积分可以直接得到

物体位置、速度、旋转的变化，但是因为 IMU 的测量存在着高斯白噪声和测量偏置的随机游走噪声，随着时间的增加，IMU 积分计算的结果会存在严重的漂移，所以经常需要将 IMU 传感器的测量与其他传感器相融合，产生更加准确的估计结果。在本节研究中，计算两个连续两帧图像之间的 IMU 的积分增量，从而将视觉数据与 IMU 数据对齐，这个过程称之为预积分(Pre-Integration)。一方面，因为 IMU 传感器的测量频率很高，通常是图片帧率的 10 倍左右(例如 IMU200Hz, Image20Hz)，采用预积分可以保证每一时刻的视觉帧都有一个与之对应的 IMU 预积分增量；另一方面，预积分只是连续视觉帧之间的 IMU 测量增量，与起始时刻的位姿无关，所以当后端非线性优化更新状态量时，并不会改变预积分增量的值，并且两个连续的视觉帧的时间间隔很短，避免了严重的累积漂移问题。具体的预积分增量的推导如公式(2-20)、(2-21)、(2-22)所示。

$$p_{wb_j} = p_{wb_i} + v_i^w \Delta t - \frac{1}{2} g^w \Delta t^2 + q_{wb_i} \iint_{t \in [i, j]} (q_{b_i b_t} a^{b_t}) \delta t^2 \quad (2-20)$$

$$v_j^w = v_i^w - g^w \Delta t + q_{wb_i} \int_{t \in [i, j]} (q_{b_i b_t} a^{b_t}) \delta t \quad (2-21)$$

$$q_{wb_j} = q_{wb_i} \int_{t \in [i, j]} q_{b_i b_t} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} w^{b_t} \end{bmatrix} \delta t \quad (2-22)$$

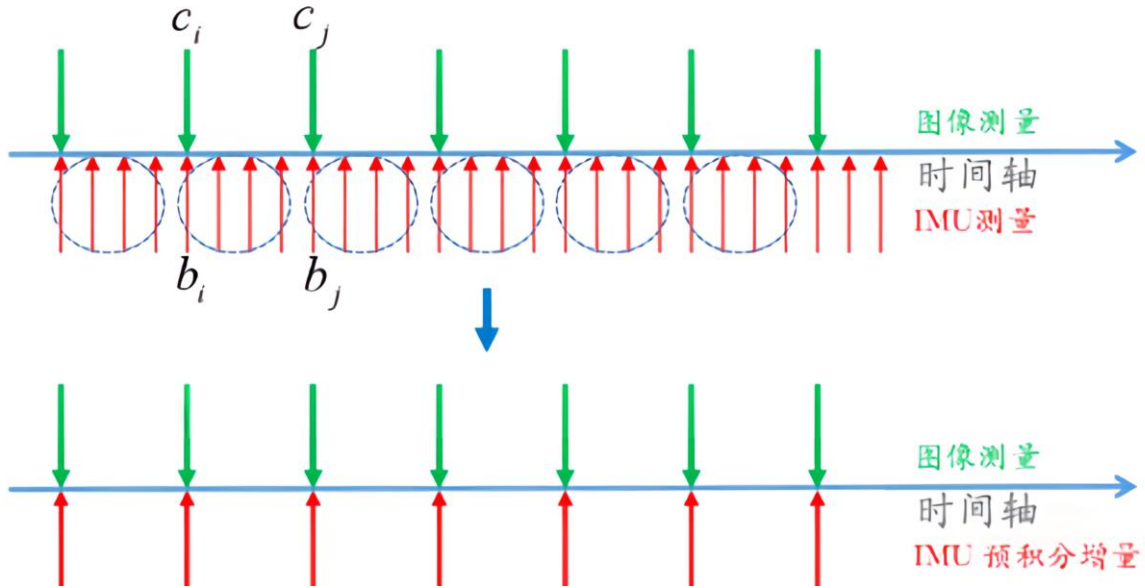


图 2-5 IMU 预积分增量描述示意图

如公式(2-20)、(2-21)、(2-22)和图 2-5 所示，两个连续的视觉图像数据 c_i 、 c_j 中间对应着 $b_i \dots b_j$ 多帧 IMU 测量数据 p_{wb_i} 、 v_i^w 、 q_{wb_i} 分别表示在 i 时刻 IMU 在世界坐标系下的位置、速度、四元素，公式(2-20)、(2-21)、(2-22)分别表示以 i 时刻 IMU 的位置、速度、四元素为参考，对 c_i 、 c_j 间隔内的 $b_i \dots b_j$ 多帧 IMU 测量数据积分，计算出 j 时刻 IMU 在世界坐标系下的位置、速度、四元素 p_{wb_j} 、 v_j^w 、 q_{wb_j} 。其中 i 、 j 时刻所对应的预积分增量为 $\alpha_{b_i b_j}$ 、 $\beta_{b_i b_j}$ 、 $\gamma_{b_i b_j}$ ，具体形式如公式(2-23)、(2-24)、(2-25)所示。

$$\alpha_{b_i b_j} = \iint_{t \in [i, j]} (q_{b_i b_t} a^{b_t}) \delta t^2 \quad (2-23)$$

$$\beta_{b_i b_j} = \int_{t \in [i, j]} (q_{b_i b_j} a^{b_i}) \delta t \quad (2-24)$$

$$\gamma_{b_i b_j} = \int_{t \in [i, j]} q_{b_i b_j} \otimes \begin{bmatrix} 0 \\ \frac{1}{2} w^{b_i} \end{bmatrix} \delta t \quad (2-25)$$

将预积分公式表示成离散形式，则两个连续 IMU 测量之间位置、速度、旋转的 IMU 预积分增量分别表示为式(2-29)、(2-30)、(2-27)。公式(2-26)、(2-27)中 a 、 w 分别表示以 i 时刻为基准，从 k 时刻和 $k+1$ 时刻的加速度和旋转角速度测量的平均值。

$$w = \frac{1}{2} [(w^{b_k} - b_{w_k}) + (w^{b_{k+1}} - b_{w_{k+1}})] \quad (2-26)$$

$$\gamma_{b_i b_{k+1}} = \gamma_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2} w \delta t \end{bmatrix} \quad (2-27)$$

$$a = \frac{1}{2} [q_{b_i b_k} (a^{b_k} - b_{a_k}) + q_{b_i b_{k+1}} (a^{b_{k+1}} - b_{a_{k+1}})] \quad (2-28)$$

$$\alpha_{b_i b_{k+1}} = \alpha_{b_i b_k} + \beta_{b_i b_k} \delta t + \frac{1}{2} \alpha \delta t^2 \quad (2-29)$$

$$\beta_{b_i b_{k+1}} = \beta_{b_i b_k} + a \delta t \quad (2-30)$$

2.2.3 优化

非线性最小二乘问题可以通过 Gauss-Newton 或者 Levenberg-Marquardt 方法求解。代价函数相对于初始状态估计 $\hat{\chi}$ 线性化。代价函数等价于：

$$\underset{\delta \mathcal{X}}{\operatorname{argmin}} \sum_{t=0k \in S}^n \|\mathbf{e}_t^k + \mathbf{J}_t^k \delta \mathcal{X}\|_{\Omega_t^k}^2 \quad (2-31)$$

其中， $\mathbf{J}$ 为每个因子相对于当前状态 $\hat{\chi}$ 的雅可比矩阵。在线性化近似之后，该代价函数具有 $\delta \mathcal{X}$ 的闭式解。本次毕业设计研究中采用的是 Gauss-Newton 法，推导解如下：

$$\underbrace{\sum_t \sum_k \mathbf{J}_t^{kT} \Omega_t^{k-1} \mathbf{J}_t^k}_{\mathbf{H}} \delta \mathcal{X} = - \underbrace{\sum_t \sum_k \mathbf{J}_t^{kT} \Omega_t^{k-1} \mathbf{e}_t^k}_{\mathbf{b}} \quad (2-32)$$

最后，当前状态 $\hat{\chi}$ 由 $\hat{\chi} \oplus \delta \mathcal{X}$ 更新，其中 $\oplus$ 为流形上旋转的加法运算。该过程迭代若干次，直到收敛。在实际运行过程中采用 Ceres 求解器来求解该问题。

2.2.4 边缘化

在处理连续帧的图像数据时，随着时间的推移，需要追踪的状态数量（如相机姿态、三维点位置等）会不断增加。这导致了优化问题的规模逐渐扩大，进而使得计算复杂度呈平方级增长。这种增长趋势不仅加大了计算资源的消耗，还可能影响到实时处理的性能，在无人机飞行过程中会对无人机的定位产生较大干扰。为了有效限制计算复杂度，同时又不损失有用的信息，本次研究工作中引入了边缘化技术。边缘化是一种在优化问题中处理信息冗余和降低计算量的有效方法。在三维重建和计算机视觉中，边缘化过程实际上是将先前的测量数据转化为一个先验项，这个先验项包含了过去帧中的有用信息，但不再作为

独立的变量参与优化。具体来说，边缘化过程涉及到将先前帧中的状态变量从优化问题中移除，同时保留这些状态变量与当前帧中状态变量之间的约束关系。这些约束关系被转化为一个先验项，以矩阵的形式嵌入到新的优化问题中。通过这种方式，我们能够在不丢失有用信息的前提下，显著减小优化问题的规模，从而降低计算复杂度。

需要边缘化的一组状态被表示为 $\mathcal{X}_m$ ，剩余的一组状态被表示为 $\mathcal{X}_r$ 。通过累加所有的边缘化因子，获得一个新的 $\mathbf{H}$ 和 $\mathbf{b}$ 。在重新排列状态的顺序之后，获得如下关系：

$$\begin{bmatrix} \mathbf{H}_{mm} & \mathbf{H}_{mr} \\ \mathbf{H}_{rm} & \mathbf{H}_{rr} \end{bmatrix} \begin{bmatrix} \delta\mathcal{X}_m \\ \delta\mathcal{X}_r \end{bmatrix} = \begin{bmatrix} \mathbf{b}_m \\ \mathbf{b}_r \end{bmatrix} \quad (2-33)$$

使用舒尔补来执行边缘化，如下：

$$\underbrace{(\mathbf{H}_{rr} - \mathbf{H}_{rm}\mathbf{H}_{mm}^{-1}\mathbf{H}_{mr})}_{\mathbf{H}_p} \delta\mathcal{X}_r = \underbrace{\mathbf{b}_r - \mathbf{H}_{rm}\mathbf{H}_{mm}^{-1}\mathbf{b}_m}_{\mathbf{b}_p} \quad (2-34)$$

来获取剩余状态新的先验 $\mathbf{H}_p$ 、 $\mathbf{b}_p$ 。有关边缘化状态的信息被转化为先验项，而没有任何损失。具体而言，在系统中保留十个空间相机帧。当传入新的关键帧时，边缘化视觉和惯性因子，它们与第一帧的状态有关。

在使用贝叶斯法则获得有关当前状态的先验信息之后就能够计算后验，即：似然与先验的乘积： $p(\mathcal{X}|\mathbf{z}) \propto p(\mathbf{z}|\mathcal{X})p(\mathcal{X})$ 。接着，状态估计变成一个最大后验（MAP）问题。注意，在滑动窗口内保留从时刻 m 到时刻 n 的状态。在时刻 m 之前的状态被边缘化，并且转化为一个先验项。因此，MAP问题被写成：

$$\begin{aligned} \mathcal{X}_{m:n}^* &= \underset{\mathcal{X}_{m:n}}{\operatorname{argmax}} \prod_{t=m}^n \prod_{k \in \mathbf{S}} p(\mathbf{z}_t^k | \mathcal{X}_{m:n}) p(\mathcal{X}_{m:n}) \\ &= \underset{\mathcal{X}_{m:n}}{\operatorname{argmin}} \sum_{t=m}^n \sum_{k \in \mathbf{S}} \|\mathbf{z}_t^k - h_t^k(\mathcal{X}_{m:n})\|_{\Omega_t^k}^2 \\ &\quad + (\mathbf{H}_p \delta\mathcal{X}_{m:n} - \mathbf{b}_p) \end{aligned} \quad (2-35)$$

2.2.5 实验效果

下图 2-6 为手持 d435i 相机在实验室内环绕一圈所得到的实验结果。其中绿线为相机的运动轨迹图，从图中可以看出，相机环绕一圈后的末位置与初位置的距离差在误差范围允许内（两者基本重合），此双目视觉惯性里程计精确度和稳定性满足实验要求。

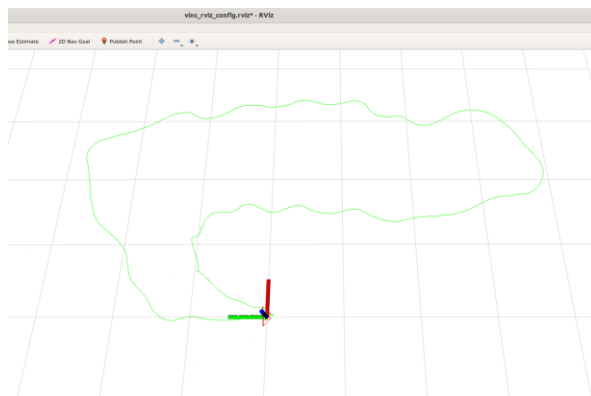


图 2-6 双目视觉惯性里程计

3. 后期拟完成的研究工作和进度安排

(1) 2024.3: 确定双目视觉惯性里程计和 GPS 数据融合的方法, 初步完成数据融合的代码编写;

(2) 2024.4: 在室外开阔环境下, 利用无人机运行相关算法和代码, 并根据运行效果不断改善算法和代码, 实现最终效果和各部分的优化

(3) 2024.5: 整理结果, 完成毕业论文。

4. 存在的问题与困难

(1) 在无人机上运行代码时可能由于工作环境的改变而导致相关代码无法运行;

(2) 在数据融合的过程中可能会出现数据处理错误导致结果异常;

(3) 在无人机上操作时可能会由于缺乏经验而导致实验结果与预期结果偏差较大。

5. 论文按时完成的可能性

已经完成了指定的计划中的目标, 现已开始进行下一阶段的内容, 指导教师经验丰富, 经费充足, 因此可以保证毕设的顺利完成。