

利用 pixloc 实现位姿估计的复现及改进

摘要

摘要: 已知场景中的相机姿态估计是最近由多种学习算法处理的 3D 几何任务。许多从输入图像中回归精确的几何量，如姿势或 3D 点。这要么无法泛化到新的视点，也无法将模型参数与特定场景联系起来。在本文中，我们回到了特征：我们认为深度网络应该专注于学习鲁棒和不变的视觉特征，而几何估计应该留给有原则的算法。本文提出了一种名为 PixLoc 的可训练算法，通过将图像与场景的显式 3D 模型对齐，利用由卷积神经网络提取的密集特征来定位图像的姿态，这是一种与场景无关的神经网络，可根据图像和 3D 模型估计准确的 6-DoF 姿态。基于多尺度深度特征的直接对齐，将相机定位转换为度量学习。本文复现结果良好，并在室内场景进行了实验，识别出来的准确率也非常的高。

关键词：PixLoc 算法；相机姿态估计；卷积神经网络

1 引言

相机姿态估计是在已知场景中为给定图像估计相机位置和方向的问题，解决这个问题是迈向真正自主机器人（如自动驾驶汽车）的关键一步，也是增强现实和虚拟现实系统的先决条件。对比以往的和现在的视觉定位方法，传统的大多是基于特征去估计位姿如 RANSAC 和 PNP [1]，这些 2D-3D 对应关系传统上是通过匹配局部图像特征来计算的。最新的定位系统可以处理具有复杂几何形状和外观随时间变化的大型场景。他们利用深度神经网络来学习提取这些特征，匹配它们，并过滤异常值对应关系 [2]。本文的动机是通过利用现有的姿态先验和场景几何信息，将相机姿态估计的问题分解为特征学习和几何优化两个部分。深度网络应该专注于学习鲁棒的视觉特征，而将几何估计交给原则性算法。通过这种方式，可以实现对新场景的出色泛化能力，并提高稀疏特征匹配的准确性。

2 相关工作

传统的视觉定位包括边缘方向直方图特征提取、LBP 算法、HOG 特征提取算法和 SIFT 特征提取等。这些图像特征提取算法的目标是增强图像检索的鲁棒性和准确性，因此它们通常需要进行复杂的计算和处理，以提取有关图像内容的详细信息。这确实可以提高图像检索的性能，但也引入了计算复杂性的增加，这可能在某些情况下难以满足实时性要求。

2.1 边缘方向直方图特征提取

边缘方向直方图 [8] 是一种用于描述图像中边缘特征分布的方法，它建立在边缘检测的基础之上。该方法首先执行边缘检测，以提取图像中的边缘特征，并计算每个边缘点的方向。然后，它将各个方向上的边缘点数量进行统计，构建边缘方向直方图，将其作为图像的形状特征描述。边缘方向直方图具有不受光照变化影响、简单且计算高效等优点。但是由于只关注图像的边缘信息，因此对于全局信息的描述有限。并且为构建直方图，边缘方向信息被量化，从而导致信息的损失，这影响全局特征的精确性。

2.2 LBP 算法

局部二值模式 (Local Binary Pattern, 简称 LBP) [7] 是一种基于纹理特征提取方法。LBP 算法的基本思想是对图像中的每个像素点周围的邻域进行编码，以描述该像素点的纹理特征。通过比较中心像素与邻域像素的灰度值，将这些比较结果编码为二进制模式，从而得到局部二值模式。LBP 具有计算速度快、对图像亮度变化具有一定不变性等优点，但也有一些限制，例如对于光照变化较大的情况可能不太稳定。

2.3 HOG 特征提取算法

方向梯度直方图 (Histogram of Oriented Gradients, 简称 HOG) [9] 特征提取算法是一种用于描述图像中边缘和纹理信息的计算机视觉方法。HOG 计算图像中每个像素点的梯度，将这些梯度方向信息组合成一个直方图，用来表示图像的纹理特征。HOG 是在图像的局部方格单元上操作，对图像几何的和光学的形变都能保持良好的不变性。HOG 特征需要计算每个像素的梯度，再构建直方图，在大型的图像中，计算的复杂度较高，导致在大规模图像处理和实时应用中的性能下降。

2.4 SIFT 特征提取

尺度不变特征变换 (Scale-Invariant Feature Transform, SIFT) [6] 是一种用于图像特征提取的算法，它可以检测和描述图像中的局部特征，具有尺度和旋转不变性。通过构建关键点并计算这些关键点周围区域的特征描述符来实现。SIFT 特征是图像的局部特征，其对旋转、尺度缩放、亮度变化保持不变性，对视角变化、仿射变化、噪声也保持一定的稳定性。SIFT 特征需要构建高斯金字塔、检测关键点等步骤，所以使用 SIFT 计算相对复杂，计算开销比较大。

3 本文方法

3.1 本文方法概述

PixLoc 是一种可训练的视觉定位算法，它基于由 CNN 提取的密集特征将图像与场景的显式 3D 模型对齐。该网络不需要学习姿态回归本身，只需要提取合适的特征，使算法准确且与场景无关。该方法通过展开直接对齐并仅监督姿态来进行端到端训练，从像素到姿态。与复杂的最新方法相竞争，即使后者是针对每个场景进行训练的。PixLoc 还可以作为轻量级的

后处理步骤来改进任何现有方法估计的姿态。PixLoc 根据场景已知的 3D 结构对齐查询图像和参考图像进行定位。对齐由几个步骤组成，这些步骤可以最小化 CNN 从输入图像中预测的深度特征的误差（图1）。CNN 和优化参数由真实姿态端到端训练得到。使用 PixLoc 进行姿态估计。给定稀疏的 3D 模型和粗略的初始姿态 (R_0, t_0)，PixLoc 提取具有像素置信度的多级特征，用于查询和参考图像。然后，Levenberg-Marquardt [3] 优化根据 3D 点在置信度的指导下，从粗略到精细水平对齐相应的特征。只监督每个级别预测的姿势。

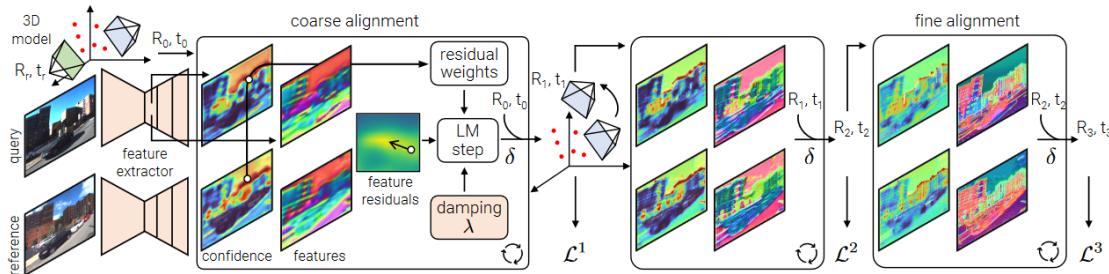


图 1. pixloc 方法示意图

3.2 三维重建模块

三维重建主要使用工具箱 Hierarchical-Localization (hloc) [4] 生成稀疏点云，hloc 主要使用 SfM(Structure-from-Motion) [5]，SfM 算法是从不同视角拍摄的一系列图像重建 3D 结构的过程，其流程如图2：

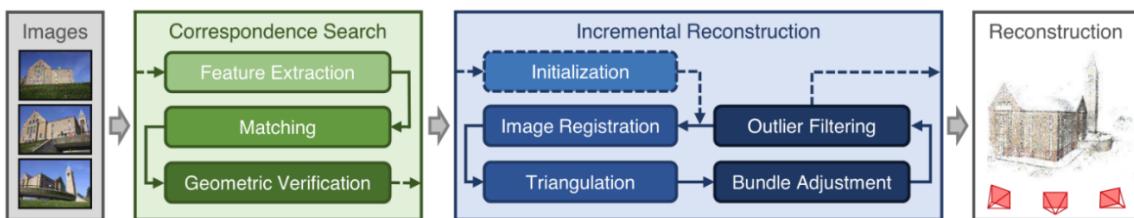


图 2. colmap 方法示意图

3.2.1 对应搜索 (Correspondence Search)

第一阶段是对应搜索，在输入图像中识别场景的重叠部分，并且识别重叠图像中相同点的投影。输出是一组经过几何验证的图像对和每个点的投影图像。

- 特征提取 (Feature Extraction): 提取图片中的特征，并为每个特征计算特征描述符。系统利用 Superpoint 提取特征。Superpoint 是一个利用无监督方式训练的一个用于提取特征以及特征描述子的网络。
- 特征匹配 (Matching): 对不同图像直接的特征点进行匹配，输出是一组潜在的重叠图像及其关联特征。该部分用的是 Superglue，这是一种基于图卷积神经网络的特征匹配算法。
- 几何验证 (Geometric Verification): 由于匹配到的数据往往被异常值污染，即匹配到的特征点不一定完全准确对应空间中的点，因此需要稳定的估计技术。通过 RANSAC 等

方法去除失配点使本征矩阵更精确。该阶段的输出为场景图像为节点，经过几何验证的匹配对为边的图。

3.2.2 增量式重建 (Incremental Reconstruction)

第二阶段是增量式重建，输入为目标场景图片，输出为注册照片的相机位姿以及重建后场景的点的集合。

- 初始化 (Initialization): 从相机重叠度高的密集位置进行初始化往往能得到更稳健和更准确的重建
- 图像配准 (Image Registration): 从度量重建开始，新的图像通过已经配准好的图像中使用与三角点的特征来解决 PnP 问题
- 三角化 (Triangulation): 通过三角化扩展点的集合来增加场景覆盖。只要至少有一张图像，也覆盖了新的场景部分，但从不同的视角被配准，一个新的场景点就可以被三角测量，并添加到点集中。图像配准与三角剖分存在共生关系，图像只能配准到已有的场景结构中，场景结构只能从配准的图像中进行三角剖分。增加了现有模型的稳定性，并通过提供额外的 2D-3D 对应关系来实现新图像的配准。
- 光束法平差 (Bundle Adjustment): 光束法平差是摄像机参数和实际点参数的非线性优化，其最小化重投影误差。Bundle Adjustment 公式¹

$$E = \sum_j \rho(\|\pi(P_c, X_k) - x_j\|_2^2) \quad (1)$$

使用将场景点投影到图像空间的 π 函数和损失函数 ρ 来降低异常值的权重，Levenberg-Marquardt 算法是用来做光束法平差的首选方法。

3.3 定位算法 pixloc 流程

图像检索采用的是深度学习方法 NetVlad，这是一个弱监督的用于场景识别的 CNN 网络。图像检索是给定 query image q ，然后通过特征提取器 f 得到一个固定维度的特征向量 $f(q)$ ，通过计算 $f(q)$ 和数据库中以提前计算的特征向量之间的距离，从而判断 q 所属的类别。具体步骤如下：

1. 首先将待测照片进行下采样。利用网络提取包含更多信息的特征。经过检索后，在已有的数据库中匹配到最相似的图像，列为参考图像。利用 Unet 提取不同尺度的特征图。
2. 然后用的是一个 EDTER(Edge Detection with Transformer)，这是一个基于 Transformer 的边缘检测器，得到一个关于边缘的灰度图。边缘特征图进行归一化以后，融入到不确定图当中，借此增强不确定图的边缘特征，进一步提高定位精度。得到参考图片和待测图片共计六张特征图和六张不确定图后，进行下一步的定位。通过定义残差公式²:

$$r_k^i = F_q^l [p_q^i] - F_k^l [p_k^i] \in \mathbb{R}^D \quad (2)$$

公式4是 i 在给定其当前姿态估计值的查询中的投影，其中 $[\cdot]$ 表示亚像素差值操作。

$$p_q^i = \prod(RP_i + t) \quad (3)$$

N 个观测值的总误差为公式4，公式2中 w_k^i 表示特征图中每个像素的不确定度。

$$E_i(R, t) = \sum_{i,k} W_k^i (\|r_k^i\|_2^2) \quad (4)$$

损失函数公式5：通过比较在每个水平 (R_l, t_l) 上估计的姿态与其基本事实 (\bar{R}, \bar{t}) 来训练的。将3D点的重投影误差降至最低，其中 γ 是Huber成本。这种损失自适应地加权了每个训练样本的旋转和平移监督，并且与场景的规模无关，因此可以使用SfM生成的数据进行训练。为了防止困难的例子使精细特征变得平滑，我们仅在当前一个成功使姿势足够接近地面真实值时才在给定水平上应用损失。否则，将忽略后续损失项。

$$L = \frac{1}{L} \sum_l \sum_i \left\| \prod(R_l + t_l) - \prod(\bar{R}P_i + \bar{t}) \right\|_\gamma \quad (5)$$

3. 初始输入为图像检索到最相似图像。通过多尺度优化，由粗到细，得到图像的最终位姿。

4 复现细节

4.1 与已有开源代码对比

使用了作者的pixloc算法来计算位姿，其中特征提取模块使用的是Unet模型，然后用Transformer的边缘检测器，来提高检索的准确度。本次实验流程中具体操作中，科技楼的地下车库的数据集是通过用自己的手机采集的。手机内参通过标定板标定，通过matlab计算得出。借助生成的模型写了一个脚本，来计算输入图片的大概位姿。从数据采集到相机标定，以及计算检索出来的图片的位姿都是本人完成的。同时，最后还给模型可视化定位结果来方便观察。

4.2 实验环境搭建

我采用的是WSL2(Windows for Linux)下搭建的环境，用的是Ubuntu系统，主要实验环境如下：torch $>=1.7$ 、torchvision $>=0.8$ 、numpy、opencv-python、tqdm、matplotlib、scipy、h5py、omegaconf、tensorboard。

4.3 创新点

本次实验的创新点相对较少，主要是对作者现有工作的复现。作者主要是针对室外环境进行实验，本次复现我是采集学校科技楼地下停车场的数据，对室内环境进行实验验证，并分析晚上的定位效果，测试作者方法的准确度。作者在论文中也提到了，本方法可以在室外定位也可以在室内定位。并写入了几个脚本来验证定位出来的位姿准确度，结果也证明了作者的方法准确度很高。

5 实验流程及结果分析

5.1 数据采集

采集了学校科技楼地下车库的照片，进行三维重建。采用的是 Superpoint（提取特征）+Superglue（特征匹配）[5] 的方法进行 colmap。一共采集了 799 张照片来进行三维重建如图3a，采集了 12 张图片来定位如图3b。

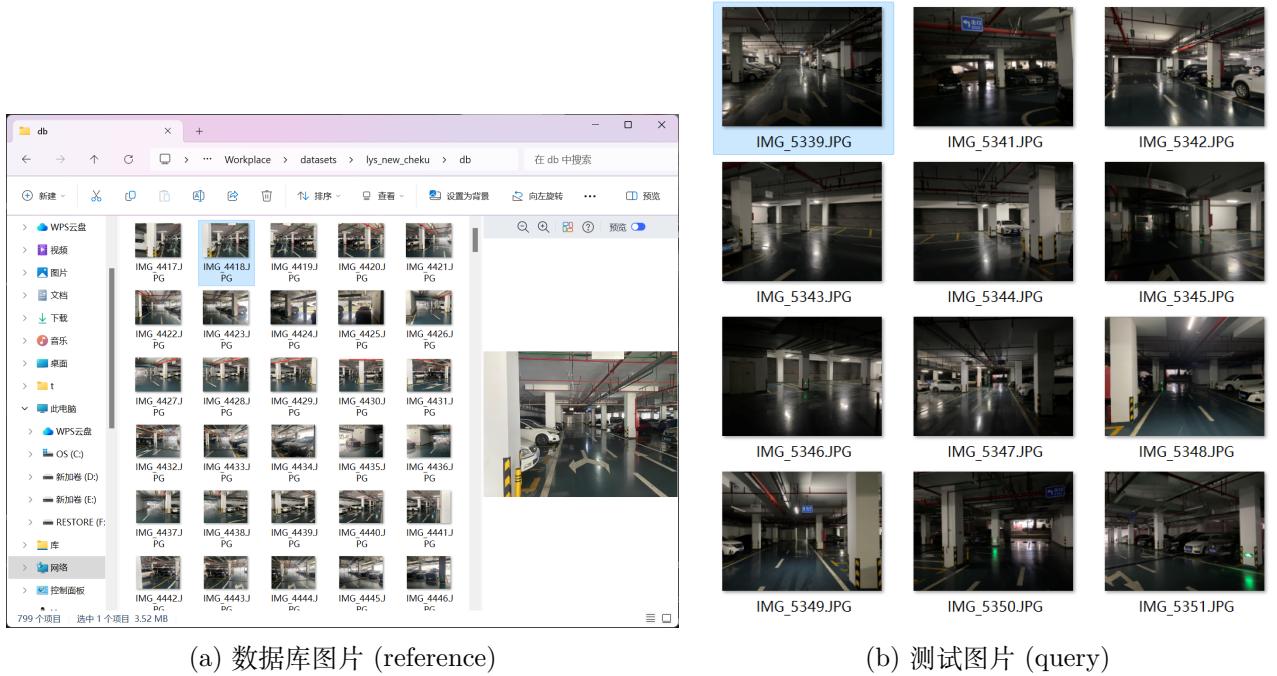


图 3. 科技楼地下车库照片采集

5.2 相机标定

相机标定是指确定相机内部参数和外部参数的过程，以建立相机图像中像素与现实世界中物体点之间的关系。相机标定的目标是找到相机的内部参数，包括焦距、主点坐标以及畸变参数，以及外部参数，即相机在世界坐标系中的位置和朝向。这些参数可以通过观察已知几何形状（例如棋盘格）的图像，并根据它们在图像中的位置关系来解算。标定的具体过程通常包括以下步骤：

1. 准备一张棋盘格并将其贴在一个平面上。
2. 调整拍摄设备的位置，拍摄一组不同角度和距离的照片用于标定物。
3. 从照片中提取棋盘格角点。
4. 估算在理想无畸变情况下的相机内参。
5. 利用最小二乘法估算实际存在径向畸变下的畸变系数。
6. 通过应用极大似然法对估计进行优化，以提高估计的精度。

5.3 三维重建结果

对采集到的数据进行 colmap，数据集生成的稀疏点云，并将其放入 CloudCompare 中的效果，如图4。colmap 的结果与实际地下车库比较好，但还是有些地方比较稀疏，可能是那个位置的图片比较少，采集的时候没有注意到。

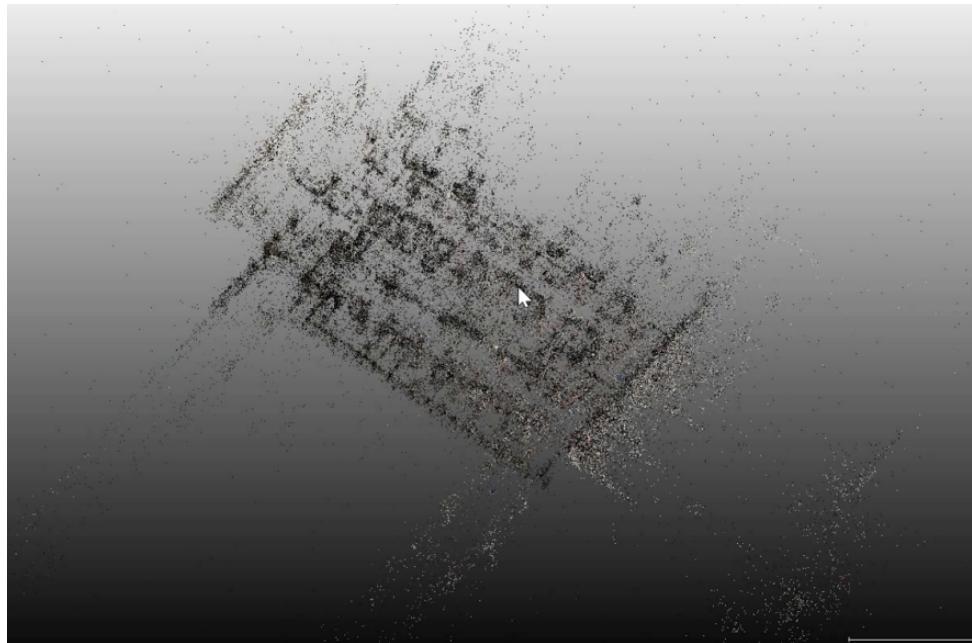


图 4. 地下车库三维重建结果

5.4 实现单张图片定位

从 query 中挑选一张图片来进行定位，能够从数据库 reference 中匹配一张与测试图片相同位姿的照片，能够输出 pixloc 提取具有像素置信度的多级特征。如图5是匹配结果和从粗鲁到精细水平对齐的特征，上图是数据集中的一张 reference 图片，下图是 query 测试图片。图像匹配的结果也是比较精确的，reference 图片与 query 图的位置大致匹配。

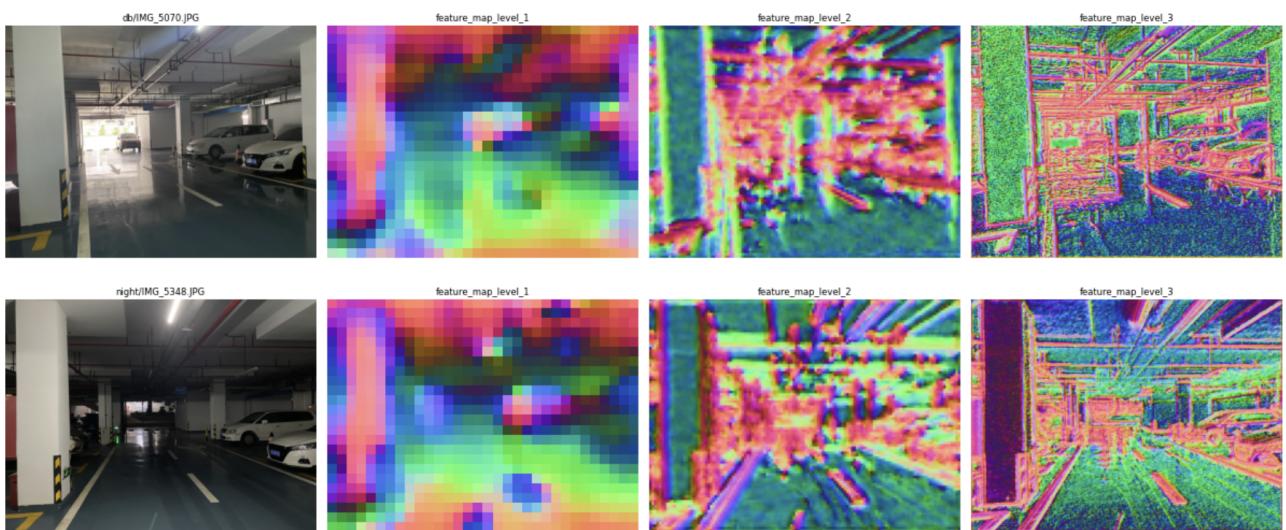


图 5. 定位匹配图和图片置信度

5.5 位姿可视化

对位姿可视化，对 query 和 reference 图像进行可视化操作，可以看出图像位姿的变化与角度的关系。可以通过调整方向、点云大小以及角度来查看。如图6。



图 6. 位姿可视化

5.6 计算图片位姿

编写了计算图像定位 (localization) 的 Python 脚本，通过相机标定获得的内参数以及利用模型的参数来计算出图像的位姿 (position) 和旋转矩阵 (rotation)。并将输入的位姿单独做一个点云放入到 CloudCompare 中如图7，图中画红圈的地方就是通过计算定位到的位姿，与实际的位姿去对比，也可以发现计算的比较准确。

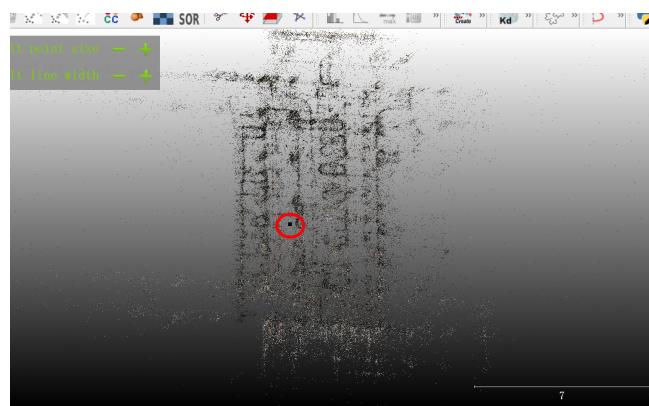


图 7. 单张图片的位姿

6 总结与展望

本次复现内容存在不足，没有进一步的提高定位精度，实现更好的定位效果。由于基础比较薄弱，复现过程比较复杂，每个部分需要花费一定的时间去弄明白，时间比较紧张，希

望能在未来继续突破，继续深入了解。可以在未来通过大语言模型去不断改进算法，提高算法的效率以及减少算法图像检索匹配时间。

参考文献

- [1] Cenek Albl, Zuzana Kukelova, and Tomas Pajdla. Rolling shutter absolute pose problem with known vertical direction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3355–3363, 2016.
- [2] Aritra Bhownik, Stefan Gumhold, Carsten Rother, and Eric Brachmann. Reinforced feature points: Optimizing feature detection and description for a high-level task. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4948–4957, 2020.
- [3] Donald W Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.
- [4] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12716–12725, 2019.
- [5] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.
- [6] 吴锐航, 李绍滋, and 邹丰美. 基于 sift 特征的图像检索. *计算机应用研究*, 25(2):478–481, 2008.
- [7] 张笃振. 基于颜色特征与 lbp 的图像检索算法研究. *微计算机应用*, 30(6):35–38, 2009.
- [8] 申海洋, 李月娥, and 张甜. 基于边缘方向直方图相关性匹配的图像检索. *计算机应用*, 33(7):1980–1983, 2013.
- [9] 邹北骥, 郭建京, 朱承璋, 杨文君, 吴慧, and 何骐. Bow-hog 特征图像分类. *浙江大学学报(工学版)*, 51(12):2311–2319, 2017.