arXiv:2303.03281v2 [cs.CV] 9 Aug 2023

视觉地点识别：教程

Stefan Schubert†、Peer Neubert‡、Sourav Garg††、Michael Milford‡‡ 和 Tobias Fischer‡‡

I. 引言

定位是移动机器人必备的能力，使其能够构建对环境的全面表征，并有效地与环境交互以实现目标。该领域中一个迅速发展的研究方向是视觉地点识别（VPR），即仅凭图像识别世界上先前见过的地点的能力。多年来，关于 VPR 的已发表研究数量显著且持续增长：2006年，标题中含有“视觉地点识别”的论文有两篇，含有“地点识别”的论文有七篇；而在 2022[1] 中，这两类论文的数量分别达到 65 篇和 163 篇。多篇综述和基准测试论文探讨了视觉地点识别领域面临的挑战、尚未解决的问题及取得的成果 [1], [2], [3], [4], [5], [6]。

本文是首篇关于视觉地点识别的教程论文。它统一了视觉地点识别的术语，并从两个重要方向补充了既有研究：

1）为该领域的新手提供系统的介绍，涵盖视觉位置识别问题的表述、通用算法流程、视觉地点识别方法的评估方法，以及视觉地点识别面临的主要挑战和可能的应对方式。

2）作为对熟悉视觉位置识别问题的研究人员的贡献，本文考察了不同类型的视觉位置识别问题在输入（数据库或查询集）、数据处理（在线或离线）和输出（每幅查询图像对应一个或多个匹配结果）方面的复杂之处。本教程还讨论了视觉地点识别算法评估中的细微差别，例如，评估视觉地点识别系统时，系统需要为每个查询找到所有 匹配的数据库图像，而非仅找到一个 匹配结果。

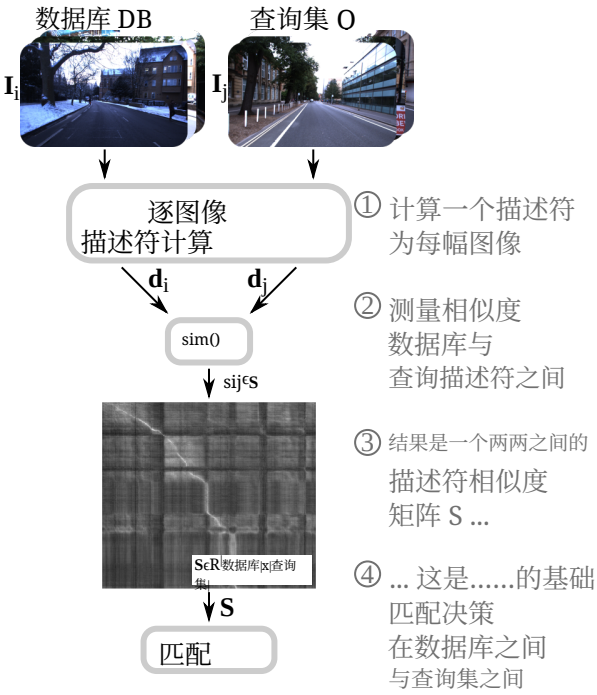


图 1：本图展示了第一节概述的视觉地点识别（VPR）的关键步骤和组成部分。从历史上看，第 4 步中的匹配决策曾用于同步定位与建图（SLAM）中的闭环检测；第二节概述了视觉位置识别问题的发展历史及其相关性。虽然本图展示了一种常见用例，即将查询集中的输入图像与数据库进行比较，但第III节基于这一流程区分了不同的视觉地点识别问题类别，并将其与视觉地点识别应用场景联系起来。第四节将讨论图中各个计算步骤的细节，随后第五节将介绍视觉位置识别流程的评估。（照片来自 [7]）

† S. S. 任职于德国开姆尼茨工业大学。电子邮箱：stefan.schubert@etit.tu-chemnitz.de

‡ P. N. 任职于德国科布伦茨大学。电子邮箱：neubert@uni-koblenz.de

†† S. G. 目前任职于澳大利亚阿德莱德大学（相关工作是在澳大利亚昆士兰科技大学期间完成的）。电子邮箱：sourav.garg@adelaide.edu.au

‡‡ M. M. 和 T. F. 任职于澳大利亚昆士兰科技大学机器人中心。电子邮箱：{michael.milford, tobias.fischer}@qut.edu.au

S. S. 感谢德国联邦经济部的支持事务与气候行动。S. G.、M. M. 和 T. F. 感谢昆士兰科技大学机器人中心提供的支持、澳大利亚研究委员会桂冠学者奖学金 FL210100156（授予 M. M.）提供的资金，以及英特尔实验室向 T. F. 和 M. M. 提供的资助。作者感谢 Mark Zolotas 博士、Alejandro Fontan 博士和 Somayeh Hussaini 对论文草稿提出的宝贵见解。[1]来源：谷歌学术，查询标题：*“title”*

Python 实际代码示例向潜在从业者和研究人员展示了如何实现和评估视觉地点识别（VPR）。相应的源代码可在线获取，此外还列出了文献中的其他开源实现：https://github.com/ stschubert/VPR_Tutorial。该链接还提供了一个用 Python 编写的 Jupyter Notebook，介绍基本的视觉位置识别流程。用户可以借此尝试其他图像描述符、基准数据集和评估指标。

以下第 I-A 节将对视觉位置识别问题作基本介绍。第 I-B 节随后概述本教程其余部分的结构。

A. 视觉地点识别（*VPR*）基础

视觉地点识别涉及匹配一个或多个图像集，以确定哪些图像展示了现实世界中的相同地点。这些图像集通常由移动设备（如手机或 AR/VR 头显）记录，也可能由安装在各种平台上的相机记录，例如机器人、无人飞行器、汽车、公共汽车、火车、自行车或船。

本质上，视觉地点识别（VPR）是一个图像检索问题，其目标是识别之前见过的地点。这一背景提供了超出一般图像检索设置的额外信息和结构。许多视觉地点识别方法会利用这些背景信息，在各种环境中匹配展示相同地点的图像，包括外观和视点差异显著的环境。例如，一项经常被利用的额外信息是：安装在汽车上的相机拍摄的连续 图像所呈现的地点，在空间上彼此相近。

图1概述了基本视觉位置识别流程中的典型步骤和组成部分。给定由已知地点的数据库图像$\mathbf{I}_i \in DB$ 组成的参考集，以及一个或多个查询图像$\mathbf{I}_j \in Q$，目标是在这两个集合之间找到匹配结果，即查询集中的图像 j 与数据库中的图像 i 展示了同一地点的情况。为了找到这些匹配结果，必须为每幅图像计算一个或多个描述符$\mathbf{d}$ ——展示同一地点的图像，其描述符应当相似；而展示不同地点的图像，其描述符则应当不同。描述符通常表示为数值向量（例如，128 维或 4,096 维）。从概念上讲，我们可以将数据库图像与查询图像之间所有成对描述符相似度 s_{ij} 构成的矩阵$\mathbf{S}$ 视为决定哪些图像应当匹配的依据。在实践中，我们必须谨慎选择用于计算和比较图像描述符$\mathbf{d}$的算法，并充分考虑当前视觉位置识别问题所面临的具体挑战和背景。本教程的其余部分将进一步详细介绍这些方面，并从更广泛的理论和实践视角探讨视觉位置识别问题。

B. 本教程所阐述的视觉位置识别问题及其细节

本教程第二节将概述视觉地点识别问题的相关性与发展历史，以及它与其他领域的关系，尤其是其对于拓扑同时定位与建图（SLAM）的重要性。在这种方法中，数据库 DB对应于地图中先前访问过的地点集合。事实上，视觉地点识别研究最初的推动因素之一，是为SLAM系统生成闭环，也就是说，在重访某个地点时（例如，沿环路行进），识别该地点，并将当前观测与地图中已有的观测关联起来（即完成闭环） [8]。这类拓扑SLAM系统最早的示例之一是FAB-MAP [9]，也称为“纯外观SLAM”；它仅依据外观（即图像）生成闭环，因此不同于ORB-SLAM [10] 等三维/度量SLAM系统，后者的地图和视觉地标以三维形式表示。

“地点”的定义是视觉地点识别的一个重要方面。在本教程中，我们采用这样的定义：两幅图像必须具有一定的视觉重叠，即共享部分图像内容例如相同的建筑物，才能被认为是“在同一地点拍摄的” [2]。这一定义使我们能够进一步估计匹配图像之间的相机变换，以用于视觉定位、建图或同步定位与建图等任务——事实上，所需的视觉重叠程度取决于具体应用。我们注意到，另一种定义（尤其是一些研究人员 [11], [12] 所采用的定义）认为，两个地点是否匹配完全取决于它们的位置，而不考虑方向，因而也不考虑视觉重叠。第III节将介绍视觉地点识别的不同应用，并讨论由可用输入数据、所需数据处理和所请求输出的差异所产生的各种视觉地点识别问题子类型。

视觉地点识别算法通常针对特定应用的具体属性量身定制。第IV节将详细介绍一种通用视觉地点识别流程，为各种实际场景及其独特特征提供共同基础。从本节开始，本教程将提供使用 Python 编写的实用代码示例。

需要注意的是，并非所有视觉地点识别算法都针对同一个视觉位置识别问题，例如每个查询所要求的图像匹配数量。这一点对于评估和比较不同视觉地点识别算法的性能尤为重要。第V节将说明并讨论评估流程，这些流程涉及各种数据集、真实值方面的细微差别以及不同的性能指标。

底层数据的属性会显著影响视觉位置识别问题的难度，以及特定算法的适用性。第VI节将讨论一些挑战，例如光照或天气条件变化导致的严重外观变化、同一地点图像之间巨大的视点变化，以及感知混淆，即在两个不同地点拍摄的图像可能看起来极为相似。本节还将介绍应对这些挑战的常见方法，以提高鲁棒性、性能、运行时间和内存效率。这些方法包括对通用流程进行方法上的扩展，其中部分扩展借鉴了机器人场景（例如，沿轨迹录制为视频的图像集）；在这类场景中，视觉地点识别不同于纯粹的图像检索。这通常可以利用额外的知识和信息，例如时空序列（即数据库 DB 和查询集 Q 中的连续图像在现实世界中也彼此相邻）或集合内相似度（即在 DB 或 Q内的相似度）。

II. 历史、相关性与相关领域

视觉地点识别（VPR）的研究可追溯至视觉SLAM、视觉地理定位以及应用于地点图像的图像检索方面的进展 [13]。在机器人学文献中，VPR 历来被称为回环检测，主要用于视觉SLAM中的这一目的 [13]。随着早期基于全局和局部束调整技术的度量SLAM方法只能处理规模有限的环境，VPR 在该领域逐渐受到更多关注，从而为基于词袋方法的拓扑SLAM技术铺平了道路 [14],

例如 FAB-MAP [9]。除了在同步定位与建图流程中的重要性之外，VPR 仍然是仅进行定位的流程中的关键组成部分，此类流程中的地图是预先可用的。

早期的视觉地点识别研究主要关注在环境条件恒定或略有变化时的地点识别。到了 2000 年代后期，研究开始应对昼夜周期或季节变化等更剧烈的条件变化所导致的外观变化。这些方法例如依赖局部特征匹配 [15] 或持续更新基于外观的地图 [16]。此后，在具有挑战性的条件下开展视觉地点识别研究的数量稳步增加，例如应对棘手的昼夜变化 [17]。近年来，研究人员广泛使用自 2012 [1], [3] 年以来出现的、涵盖条件变化的数据集。2014年，深度学习被引入视觉地点识别 [18]，以处理具有挑战性的数据；此后证明了它在变化环境中的有效性 [19]。除了图像和图像描述符，视觉地点识别研究还探索了利用序列、集合内相似度、较弱的全球定位系统信号或里程计等附加信息来提升性能 [20]。

关于视觉地点识别与其他领域之间的关系，我们推荐以下教程： [13] 概述了概率式同步定位与建图，并包含一个关于回环检测的章节，不过自该教程发布以来已有 15 年以上，视觉地点识别领域已经取得了大量进展。[8] 专门探讨同步定位与建图中的回环检测问题。 [21] 通过示例代码介绍了如何使用机器人操作系统（ROS）进行同步定位与建图。 [22]讨论了视觉里程计，即根据视觉输入估计智能体自身运动的过程。因此，视觉里程计与视觉地点识别互为补充；在构建同步定位与建图系统时，两者可以结合起来检测闭环。不过，需要注意的是，本教程的范围仅限于为读者提供关于视觉地点识别及其核心概念的通俗入门介绍。将视觉地点识别方法整合到完整的同步定位与建图或重定位系统中等内容，不在本教程的讨论范围内；探讨这些内容还需涉及许多其他方面，例如与视觉地点识别没有直接关系的批量优化。

除回环检测之外，如果全球定位传感器（例如全球导航卫星系统（GNSS），如全球定位系统（GPS）、伽利略卫星导航系统或北斗卫星导航系统）不可用或精度不足，则视觉地点识别也是必要的。在城市环境中，建筑物或其他结构可能形成“城市峡谷”，阻挡视距卫星信号，造成遮挡，使全球导航卫星系统接收机无法获取准确的位置信息。除了遮挡之外，GNSS 信号从建筑物和其他结构反射所形成的所谓非视距信号，也会进一步降低 GNSS 的精度。这一问题并不局限于城市环境，因为类似的遮挡和反射也可能发生在自然环境中，例如山谷或峡谷。同样，室内环境和洞穴也会因墙壁对卫星信号的吸收或反射而阻碍 GNSS 信号。

或者，视觉地点识别可以作为自主系统中的冗余组件，用于容错以及应对全球导航卫星系统的常见中断，例如卫星服务中断、性能退化或位置/时间异常。值得注意的是，所有全球导航卫星系统都可能遭到中央机构的攻击或封锁，无法用于非军事用途。其他系统可能出于成本或安全方面的考虑而未配备全球导航卫星系统接收机。对于机器人外星任务，安装全球导航卫星系统可能成本过高或耗时太长。

III. 视觉地点识别问题类别与用例

在定位与建图文献中，视觉地点识别的使用方式各不相同，这取决于其构式的三个关键属性：输入，涉及参考图像和查询图像的提供方式（即单会话与多会话）；数据处理，定义运行模式（即在线与批处理）；以及输出，决定预期输出的类型（即单一最佳匹配与多重匹配）。下面的第 III-A 节将更详细地解释这些问题类别。随后，第 III-B 节将使用这些类别介绍不同的视觉地点识别应用场景。表 I 总结了这些用例，以及它们所需的输入和数据处理。请注意，这些类别可能存在例外和偏离情况，例如 [23] 使用多个互不相交的序列作为参考。然而，我们认为，所提出的分类体系可以作为未来研究组织各种视觉地点识别应用场景的良好起点。

A. 视觉地点识别问题类别

我们区分视觉地点识别问题可能变化的三个主要维度，由此形成不同的视觉地点识别问题类别或子类型：

1) 输入 – 单会话视觉位置识别与多会话视觉位置识别：是否有两个独立的输入集，一个用于数据库 DB，另一个用于查询集 Q，还是只有一个与自身进行比较的图像集？单会话视觉位置识别是在单个图像集内进行图像匹配，因此查询集Q 等于数据库 DB （即 $Q = DB$）。在这种情况下，一个实际需要考虑的问题是抑制与最近获取的图像之间的匹配结果——完整SLAM系统通常依靠运动模型来进行此类抑制，而独立的视觉地点识别系统则通常采用启发式方法。相比之下，多会话视觉位置识别是对两个互不相交的图像集进行匹配（即 $DB \cap Q = \varnothing$）；这些图像集是在不同时间（例如夏季和冬季）或由不同平台（例如移动机器人和手机）记录的。

2) 数据处理——在线视觉地点识别与批处理视觉位置识别：图像是逐张依次获取和处理，还是从一开始就以单个批次的形式全部可用？在线视觉地点识别需要处理不断增长的集合Q （即， $Q \neq$ const）以及一个 DB 集合，该集合要么是给定的（即， $DB =$const），要么也在不断增长（即， $DB \neq$const）。相比之下，批处理视觉位置识别可以基于完整的 Q集合（即， $Q =$ const）和 DB 集合（即， $DB =$ const）进行处理。在线视觉地点识别中的图像集不断增长，这限制了可行方法的数量。例如，基于所有图像描述符统计信息的描述符标准化 [24]，或相似度矩阵分解 [25] 等方法，未经修改便无法使用。第VI节将对此作进一步讨论。请注意，对于单会话视觉地点识别，

查询集是不断增长还是固定不变，取决于视觉地点识别是在线执行还是以批处理模式执行。

3) 输出——单最佳匹配视觉位置识别与多匹配视觉位置识别：对于查询图像，预期的输出是数据库中一幅展示同一地点的图像，还是请求该地点的所有图像？单最佳匹配视觉位置识别针对每幅查询图像仅返回最佳匹配的数据库图像 $I_i^* \in DB$。相比之下，多匹配视觉位置识别旨在为每幅查询图像找到所有 匹配的数据库图像。实际上，单最佳匹配视觉位置识别与多匹配视觉位置识别之间的区别，通常归结为：是在查询图像与所有数据库图像之间寻找最大相似度，还是寻找所有高于某一阈值的相似度，如第 IV-E 节所示。识别所有匹配图像通常比仅找到一个正确匹配更具挑战性，因为这要求针对每幅数据库图像明确判断其是否展示了与查询图像相同的地点 [1]。

让我们通过一个示例来说明这些问题类别：确定建筑物内手机的粗略位姿 [x、y、航向、楼层]，例如，以引导人员前往指定地点。为此，我们首先需要对建筑物进行建图，然后人员才能使用手机在建筑物内进行定位。对于建图的第一步，我们可以使用一台配备相机的人工控制移动机器人，采集一个图像查询集 Q^{mapping}，以及一些额外的传感器数据，如里程计。获得所有地点的这些图像后，我们可以运行一种建图算法，处理所有图像和其他数据，从而获得建筑物的度量地图，并将 Q^{mapping} 中的所有图像与度量位姿关联起来。这一建图过程的一部分是用于回环检测的单会话批处理多匹配视觉位置识别，它将整个图像集 Q^{mapping}（批处理视觉位置识别）与其自身进行比较（单会话视觉位置识别），以为每幅图像找到所有 闭环（多匹配视觉位置识别）。在这里，对整个集合 Q^{mapping} 进行批处理，可以应用计算成本高但精确的算法。

建图后（可能是数年之后），第二步是利用手机的相机流进行实际定位。定位时，我们将机器人建图所得的查询集 Q^{mapping} 作为数据库 DB^{loc}，并将其与来自手机相机的查询图像 Q^{loc} 进行比较。为了确定手机的位置，可以使用多会话在线单一最佳匹配视觉位置识别，在线（在线视觉位置识别）将查询图像流 Q^{loc} 与固定数据库 DB^{loc}（多会话视觉位置识别）进行比较，以找到最佳匹配的数据库图像（单一最佳匹配视觉位置识别）及其对应的位姿信息。

总之，视觉地点识别可用于多种不同的用例，详情将在下文第 III-B 节讨论，并在表 I 中展示。这里，每种用例通常都需要结合特定的输入 （单会话／多会话视觉位置识别）和数据处理 （在线／批处理视觉位置识别）类别。输出 类别（单一最佳匹配／多匹配视觉位置识别）的选择也取决于用例，尤其取决于视觉地点识别之后所采用的算法。例如，对于纯粹的（重）定位， [26], 可能只

表 I：视觉地点识别的输入 和数据处理类别（参见第 III-A 节）与相应用例（参见第 III-B 节）的组合。在单会话视觉位置识别中，只有一个输入集，并将其与自身进行比较。例如，在线同步定位与建图就是这种情况：机器人探索环境时，该集合会不断增长（在线视觉位置识别）；而在建图中，预先记录的数据则以批处理方式进行处理。如果输入由两个集合组成，即数据库DB 和查询集 Q （多会话视觉位置识别），则同样可以区分两种情况：数据在采集时在线处理（在线视觉位置识别），或像多会话地图构建那样以批处理方式处理。在多会话在线视觉地点识别中， DB 可以是不断增长的（如多机器人地图构建），也可以是固定的（如视觉（重）定位）。



需要单一最佳匹配，因此输出类别的选择主要取决于用例；而对于基于图的同步定位与地图构建，所需的输出类别还取决于视觉地点识别后的后处理，如下文所述。在基于图的同步定位与地图构建中， [13], 每个节点都编码了 Q中一幅图像的位姿。相连节点之间对应的边表示它们之间的变换。一条边既可以建立在时间上连续的节点之间（使用里程计），也可以建立在被视觉地点识别判定为闭环检测的节点之间。在此，单最佳匹配视觉位置识别可用于匹配并融合两个对应于同一地点的节点，从而使每个地点始终仅由一个节点表示。或者，多匹配视觉位置识别可用于在同一地点的所有现有节点之间创建多条边。当我们无法保证图中每个地点只有一个节点时，或者当我们采用鲁棒优化方法对位姿进行批量优化，并可在处理潜在离群匹配的同时受益于多重匹配提供的额外信息时，这一点尤其有用。

B. 视觉地点识别应用场景

视觉地点识别是多种机器人应用中的关键组成部分，包括自动驾驶、农业机器人技术和机器人包裹配送，也可用于创建元宇宙。视觉地点识别的一些常见任务包括：

1) 六自由度视觉定位的候选选择 [26]：六自由度（DoF）视觉定位（也称为城市尺度/自然地理定位）旨在估计相机在特定环境中的六维位姿（位置和方向）。基于



图 2：视觉地点识别用例概览。上：视觉地点识别用于从数据库中筛选出一小组候选图像，以进行视觉（重）定位；随后使用计算开销较大的六自由度位姿估计方法处理这些图像。中：视觉地点识别可用于在同步定位与建图流程中检测闭环（绿色箭头），也可用于在定位错误后或机器人被移动时（绑架机器人）重新定位。下：在多会话地图构建中，视觉地点识别同样可用于回环检测，但这次采用批处理方式，因为所有图像都预先已知。在多机器人地图构建中，视觉地点识别通过检测多个机器人都曾到访的地点来合并地图。（地图数据来自 OpenStreetMap）

固定的 DB 用于选出与当前查询图像$\mathbf{I}_i \in DB$ 相似度最高的候选项$\mathbf{I}_j \in Q$（参见图 2 上方）。随后，可使用局部图像描述符和更复杂的算法，对这些候选项进行计算量较大的六维位姿估计；若对完整的 DB 集执行此操作，则将难以实现。例如， [27] 中使用地点识别方法 NetVLAD [28]进行候选选择，然后再利用局部描述符进行位姿估计。

2) 用于在线同步定位与地图构建的回环检测和重新定位 [13]：在线同步定位与地图构建用于在创建环境地图的同时估计相机的当前位姿。单会话在线视觉地点识别 用于回环检测（即识别之前访问过的地点），如图 2（中）所示，以补偿里程计数据中的累积误差并创建全局一致地图。它还用于重新定位，以应对定位错误，或相机/机器人被手动移动的情况（即所谓的绑架机器人问题）。**3) 用于建图的回环检测** [13], [29]：建图（也称为完整 SLAM 或离线 SLAM）涉及一次性估计完整路径，以生成地图。这使得可以使用单会话批处理视觉地点识别 进行回环检测（参见图 2，中），该方法基于运行在高性能硬件上的速度较慢但更鲁棒的算法。

4) 用于多会话地图构建的回环检测 [30]：多会话地图构建将同一环境中随时间反复执行的多个 SLAM 任务的结果结合起来。多会话批处理视觉地点识别用于在所有任务各自的地图之间查找共享地点，以便进行地图合并（参见图 2 下图）。或者，使用给定数据库的多会话在线视觉地点识别可用于检测先前已建图的区域（可能用于闭环检测），并实时将地图中的未探索区域纳入其中。

5) 用于多机器人地图构建的共享地点检测 [31]：多机器人地图构建（也称为分布式 SLAM）是指使用多个机器人对环境进行分布式建图。在此，使用不断扩展的 DB 多会话在线视觉地点识别，用于查找每个机器人各自地图之间的共享地点，以便随后进行地图合并，如图 2（下图）所示。

总而言之，本节概述了视觉地点识别的不同问题类别及其相应子类型，并讨论了视觉地点识别的常见用例。

IV. 视觉地点识别的通用流程

本节概述了视觉地点识别（VPR）的通用流程。该流程包含的步骤如图1所示。流程的输入是两个图像集 DB 和 Q （如第三节所述，在单会话 VPR 中，这两个图像集可以相同）。流程会生成匹配决策，也就是说，对于每个查询图像 $\mathbf{I}_j \in Q$，都可以关联一个或多个数据库图像 $\mathbf{I}_i \in DB$。该流程包含以下中间步骤和组件：1）计算逐图像描述符；2）两两比较描述符；3）生成描述符相似度矩阵 $\mathbf{S}$；4）做出匹配决策。以下各小节将更详细地讨论这些要素。第VI节介绍了对此通用流程的扩展，这些扩展可用于提升性能，并增强应对各种挑战的鲁棒性。

A. 输入：数据库和查询图像集

回顾一下，在视觉地点识别流程中，输入由两组图像构成：数据库集 DB 以及查询集中的一组当前图像 Q。DB 集也称为参考集，代表已知地点的地图，通常是在理想条件下（例如晴朗天气）记录的，或者由不同于 Q 的平台记录（例如另一台机器人）。另一方面，查询集 Q是由不同于 DB 的平台记录的“实时视图”，或者是在 DB 之后记录的视图——可能相隔数天、数月甚至数年。两组图像在地理范围上会有重叠，并且包含部分或全部相同的地点。

```
# 加载 GardensPoint Walking 数据集 [32] ，包含两个
# 图像集：数据库和查询集，以及真实值
来自 datasets.load 导入 GardensPointDataset 数据集
数据集 = GardensPointDataset()
DB, Q, GT, GT^soft =数据集。load()
```

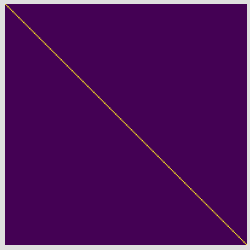
输出：



$\mathbf{I}_1 \in DB$:



$\mathbf{I}_1 \in Q$:



GT:



$\mathbf{GT}^{soft}$:

代码片段 1：在本示例中，VPR 算法的输入是两个互不相交的图像集：数据库 DB 和查询集 Q。我们加载 GardensPoint Walking 数据集 [32] 以及关于对应关系的真实值信息。该真实值仅用于后续评估，在部署算法时既不可用，也不需要。

视觉位置识别问题有不同的类别：仅使用查询集 Q （单会话视觉位置识别），或同时使用 DB 和Q 集（多会话视觉位置识别）。此外，图像集可以在处理前预先指定（批处理视觉位置识别），也可以在在线运行过程中不断增长（在线视觉位置识别）。

代码片段 1 提供了加载包含两个图像集 Q 和 DB 的数据集的示例代码，以及真实值矩阵 $\mathbf{GT}$ 和 $\mathbf{GT}^{soft}$。简而言之，$\mathbf{GT}$ 是一个逻辑矩阵，用于指示对应图像是否展示相同或不同的地点；而 $\mathbf{GT}^{soft}$ 是 $\mathbf{GT}$ 的膨胀版本，用于处理具有低视觉重叠的图像对，从而避免对这类情况下的匹配结果进行惩罚。我们将在第 V-B 节详细介绍这些矩阵。

B. 逐图像描述符计算

本节介绍图像描述符的计算过程。图像描述符是对图像的抽象表示，从原始像素中提取特征，以便更好地抵抗外观和视点变化（图1中的步骤 1，另见代码片段 2）。本教程涵盖两种主要类型的图像描述符：

1) **整体描述符**（也称为全局描述符）用一个向量 $\mathbf{I}_i \in DB$, Q $\mathbf{d}_i \in \mathbb{R}^d$ 表示一幅图像（参见代码片段 2）。这样便能以较短的运行时间高效地进行成对描述符比较。需要注意的是，当使用穷举 k 近邻搜索（kNN）为相似数据库描述符的候选选择获取最近邻时，执行时间会按比例增长

与描述符维度以及数据库中包含的图像数量均呈线性关系。

2) **局部描述符**将图像$\mathbf{I}_i$ 编码为一组 $D_i = \{\mathbf{d}_k \mid k = 1, \ldots, K\}$ 位于感兴趣区域的向量$\mathbf{d}_k \in \mathbb{R}^d$。它们通常比整体描述符性能更好，但需要计算开销较大的局部特征匹配方法，例如左右一致性检查（也称为互匹配） [33], 、单应性估计 [34],、极线约束计算 [15],或深度学习匹配技术，如 SuperGlue[35]。因此，局部描述符通常用于分层流水线：首先使用整体描述符检索排名前K 的匹配结果，然后再通过局部描述符匹配进行重新排序。

从整体描述符和局部描述符的角度对视觉地点识别流程进行抽象，是许多定位、建图和同步定位与建图（SLAM）方案的基础。其他方法包括地点分类 [12],、区域描述符 [6],和增量二进制词袋 [36]。此外，我们在第VI节列出了这一视觉地点识别流程的常见缺点及其应对方法。

要将单幅图像中的一组局部描述符转换为整体描述符，可以采用**局部特征聚合**方法，例如视觉词袋（BoVW） [37],、局部聚合描述符向量（VLAD） [38] 或超维计算（HDC） [33]。在分层流水线中，这样便可同时利用局部描述符进行候选选择（聚合后）和验证（使用原始局部描述符）。

由于描述符计算是视觉地点识别流程中的首批步骤之一，因此会显著影响后续步骤的性能以及视觉地点识别系统的整体性能。用于获取描述符的算法决定了描述符对特定环境、视点变化程度或环境条件变化类型的适用性。例如，基于卷积神经网络的整体描述符，如 AlexNet-conv3 [19]或 HybridNet [39] ，在视点变化较小或几乎没有视点变化的情况下表现良好，但在视点变化较大时表现不佳。另一方面，基于 VLAD 的 [38]描述符（如 NetVLAD [28] ）在视点变化较大的环境中往往表现更好。

此外，深度学习描述符所使用的具体训练数据会影响其在不同环境中的性能。例如，某些描述符在城市环境中的表现可能更好，而其他描述符在自然环境 [40] 或特定地理区域（如西方城市） [41]中可能更有效。

C. 两幅图像之间的描述符相似度

要比较两幅图像的图像描述符，必须计算相似度或距离度量（参见图1和代码片段3中的步骤2）。此过程比较图像 i 和 j的描述符 $\mathbf{d}_i$ 和 $\mathbf{d}j$ （整体），或 D_i 和 D_j （局部）。请注意，相似度 s_{ij} 和距离 $dist_{ij}$ 可以通过反比例函数相关联，例如

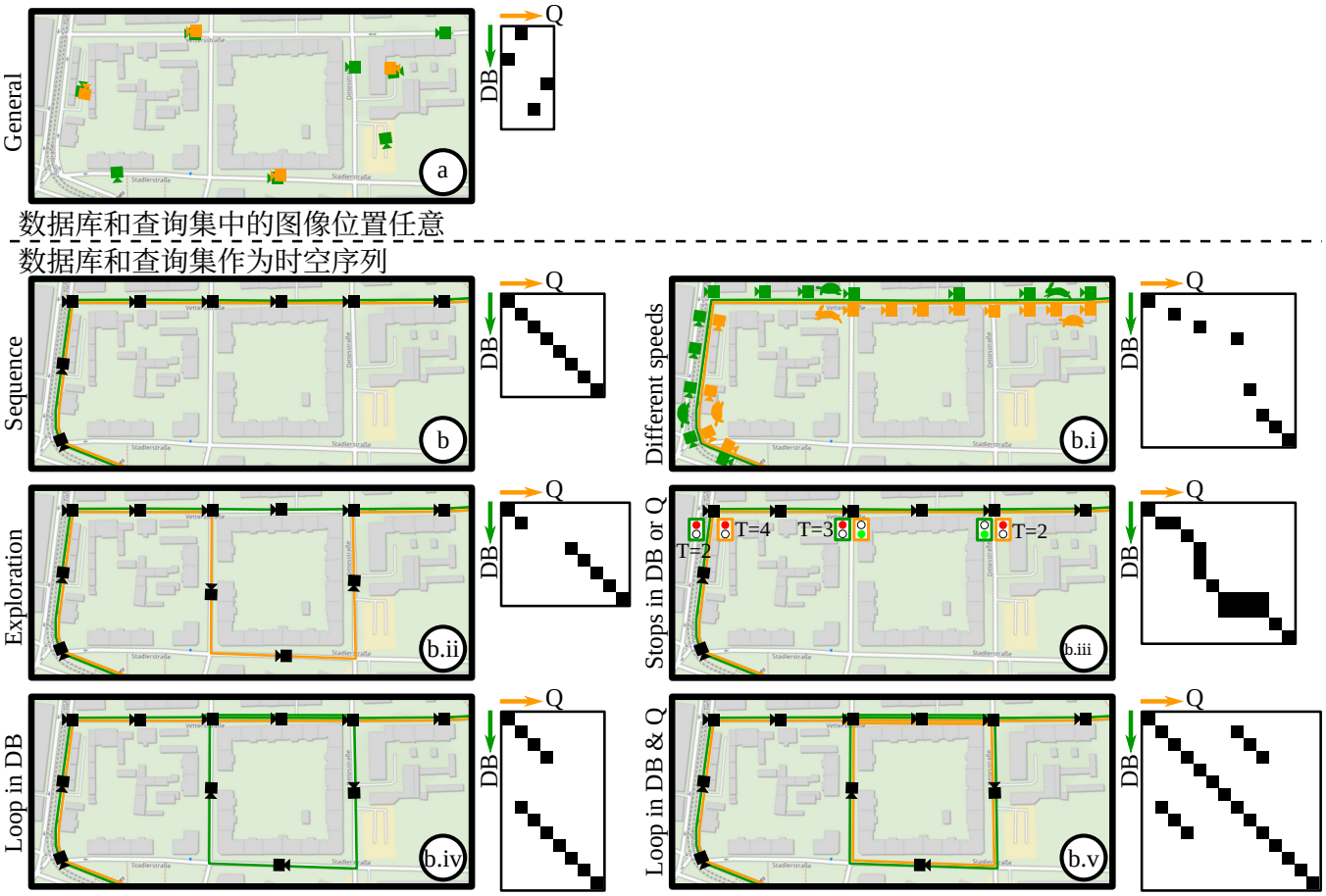


图3：数据库和查询运行期间，相似度矩阵S 与轨迹之间的关系。绿色和橙色摄像机分别表示 DB 和 Q中的图像。绿色和橙色线条表示图像沿轨迹以视频形式录制（也称为时空序列）。在 b.i) 中，兔子/乌龟分别表示沿路线行进时的速度较快/较慢；类似地，在 b.iii) 中，交通信号灯表示在 Q (T=2)、 DB (T=3)或两者（T={2,4}）中停顿 T 个时间步。

```
# 计算整体 HDC-DELF 描述符 [33]
从 特征 _提取。\
    整体特征提取器导入HDCDELF
特征_提取器 = HDCDELF()
D^DB =特征_提取器。计算_特征（数据库）
D^Q =特征_提取器。计算_特征（查询集）
```

代码片段2：图像对应关系的主要信息来源是图像描述符。由于整体图像描述符支持高效的成对描述符比较，我们为每幅图像计算一个整体 HDC-DELF [33] 描述符（图1中的步骤1）。

as

$$s_{ij} = -dist_{ij}\,, \tag{1}$$

或其倒数

$$s_{ij} = \frac{1}{dist_{ij}}\,. \tag{2}$$

整体描述符的比较效率高于局部描述符，因为它们只需要简单且计算效率高的指标，例如余弦相似度

$$s_{ij} = \frac{\mathbf{d}_i^T \cdot \mathbf{d}_j}{\|\mathbf{d}_i\| \cdot \|\mathbf{d}_j\|}, \tag{3}$$

或负欧几里得距离

$$s_{ij} = -\|\mathbf{d}_i - \mathbf{d}_j\|. \tag{4}$$

相比之下，如第 IV-B 节前文所述，对局部描述符进行比较需要更复杂且计算开销更大的算法方法。

D. 成对相似度矩阵S

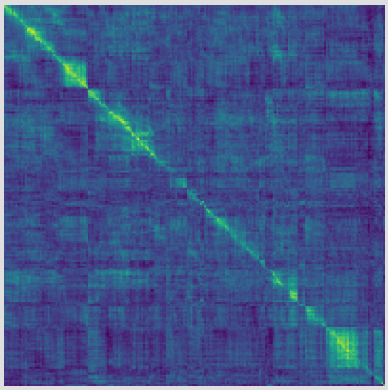
成对描述符相似度矩阵S 是视觉地点识别的关键组成部分。如图1中的步骤3所示，S包含数据库和查询集中图像的描述符之间计算出的所有相似度 s_{ij} 。在单会话视觉位置识别中，S 的维度为 $|Q|\times|Q|$；而在多会话视觉位置识别中，S 的维度为 $|DB|\times|Q|$。根据所采用的方法，S 可以是稠密的（即比较所有描述符），也可以是稀疏的（即使用近似最近邻搜索 [42]或基于序列的比较策略 [43]，仅比较部分描述符）。

```
# 使用余弦相似度比较所有描述符
# 归一化描述符
D^DB = D^DB /
      np.linalg.norm(D^DB, axis=1, keepdims=True)

D^Q = D^Q /
      np.linalg.norm(D^Q, axis=1, keepdims=True)
# 使用矩阵计算描述符之间的点积
# 乘法
S = np.matmul(D^DB, D^Q.转置())
```

输出：

S:



代码片段3：为了比较数据库描述符和查询描述符并获得描述子相似度**S** （图1中的步骤2和3），我们使用它们的余弦相似度（通过归一化描述符向量的内积计算）。虽然在大规模实际应用中，我们可能不需要计算所有可能配对的完整相似度矩阵**S** ，但这对于目视检查很有用。

在采集 Q 和 DB期间，相机轨迹会影响**S** 的整体外观，如图3所示。**S** 中的高相似度模式会显著影响视觉位置识别流程的性能，并可能使某些用于提升性能的算法步骤得以应用或受到限制。可以观察到相机轨迹与**S** 外观之间存在以下关系（参见图3，其中展示了地图中的相应示例）：

a) **一般情况**：如果 DB 和 Q 中的图像是在任意位置、无特定顺序地记录的，那么 **S** 中就没有明显的模式。这是一般视觉定位和全局地理定位的典型情况。

b) **序列**：如果 DB 和 Q 中的图像是沿着轨迹以时空序列的形式记录的（即连续图像在现实世界中也彼此相邻），那么在 **S** 中可能会观察到连续的高相似度线条。这种设置是许多机器人任务的典型情况，包括在线同步定位与地图构建、建图以及多机器人/多会话建图（第III节）。在这种设置下，可以使用基于序列的方法来提升性能（参见第VI节）。相机的轨迹会以下列方式影响 **S** ：

i) **速度**：如果相机在 DB 和 Q中的相同位置以相同速度移动，将观察到具有高相似度的线

其斜率为 $45°$ 。否则，斜率会发生变化。

ii) **探索**：如果查询图像 Q 中显示的地点未出现在 DB 中，那么高相似度线条就会中断。

iii) **停顿**：如果相机在数据库运行或查询运行期间暂时停止（速度为零），就会在另一组中产生多个连续的匹配结果。

- **数据库运行中的停顿 DB**：数据库运行中的停顿会在**S**中形成一条垂直的（位于同一列内）高相似度线。
- **查询运行中的停顿 Q**：查询运行中的停顿会在**S** 中形成一条水平的（位于同一行内）高相似度线。
- **停顿 DB 与 Q**：如果相机在数据库运行和查询运行中都停留在同一地点，那么在**S**中会观察到一块高相似度区域。

iv) **数据库中的循环**： DB 中的循环可能导致Q中的单个查询图像与多个数据库图像匹配。与停顿不同，由循环导致的多个匹配图像在其图像集中并不连续。

v) **数据库与查询集中的循环**： DB 和 Q 中的循环可能导致 DB中的单个数据库图像与额外的查询图像匹配。这会使**S**中的高相似度结构变得更加复杂。

E. 输出：匹配决策

$m_{ij} \in \mathbb{B}^{|Q|\times|Q|}$ 视觉地点识别系统的输出是一组匹配决策 $\in \mathbb{B}^{|DB|\times|Q|} \in$ **M** （图1和代码片段4中的步骤4），其中**M**（单会话视觉地点识别）或**M** （多会话视觉地点识别）用于指示第 i个数据库/查询图像与第 j个查询图像显示的是同一地点（$m_{ij} = true$）还是不同地点（$m_{ij} = false$）。现有匹配技术包括为每个查询选择最佳匹配，或对成对描述符相似度$s_{ij} \in$ **S** 进行简单的阈值处理，也包括通过比较场景的空间布局（例如使用极线约束）或语义布局来进行几何验证。例如，在代码片段4中，M_1 通过为每个查询图像选择最佳匹配的数据库图像来计算，即**S** $\in \mathbb{R}^{|DB|\times|Q|}$ 中每列的最大相似度 s_{ij} （单一最佳匹配视觉地点识别）。另一个例子是代码片段4中 M_2 的计算，其中对**S**应用相似度阈值 θ：如果 $s_{ij} \geq \theta$，则假定第 i个和第j个图像显示同一地点（多匹配视觉地点识别）。下一节将讨论这些输出的性能评估。

V. 性能评估

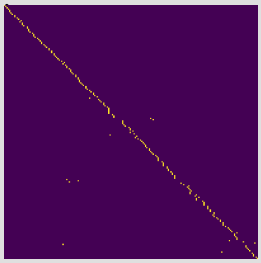
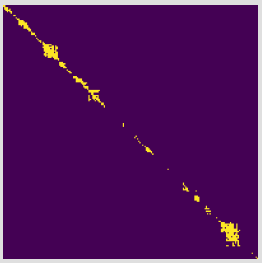
本节主要讨论匹配决策 **M** 或成对相似度 **S** 的评估，这使得不同视觉地点识别方法之间可以进行比较。这需要数据集、相应的真实值和性能指标。下面，我们将概述用于评估的这些组成部分，并讨论它们的特性以及潜在问题。

```
# 根据 S 匹配图像
从 匹配 导入匹配

# 在 S 中为每个查询寻找最佳匹配
# 单最佳匹配视觉位置识别
M1 =匹配。最佳_匹配_每个_查询(S)

# 使用自动调优的阈值 S>=thresh 查找匹配结果
# 多匹配视觉位置识别的阈值
M2 =匹配。阈值处理(S, 阈值='auto')
```

输出：

$\mathbf{M}_1$:  $\mathbf{M}_2$: 

来自$\mathbf{M}_2$的正确与错误匹配示例：



代码片段4：视觉位置识别流程的输出通常是一组离散的匹配结果，即查询图像与数据库图像的配对。要根据相似度矩阵为查询图像获取匹配结果（图1中的步骤4），我们可以找出单个最佳匹配的数据库图像（$\mathbf{M}_1$），也可以尝试找出数据库中所有与查询图像显示同一地点的图像（$\mathbf{M}_2$）。

A. 数据集

对于视觉地点识别而言，数据集由一个或多个图像集组成，这些图像集必须进行匹配，以找出共享地点。例如，广受欢迎的 Nordland 数据集 [44] 提供了四个图像集，每个季节各一个，即春季、夏季、秋季和冬季。这些图像集可以任意组合用于视觉地点识别，但一种典型选择是使用 夏季 作为 DB，并使用 春季、秋季 或冬季 作为 Q。

现有数据集在环境类型以及外观和视点变化的类型与程度上各不相同：**环境类型**包括室内环境（例如，Amsterdam-XXXL [45]）、城市环境（例如，Oxford RobotCar [7]）、郊区环境（例如，StLucia Various Times of the Day [46]），以及乡村等自然环境（例如，Nordland [44]）、森林（例如，SFU Mountain [47]）或湖泊（例如，Symphony Lake [48]）。**外观变化**可能由动态物体（如行人；例如，GardensPoint Walking [32]）、一天中的时间及其带来的光照变化和移动阴影（例如，StLucia Various Times of the Day [46]）、昼夜变化（例如，Alderley [17]）、晴朗、多云、阴天、下雨、有雾或下雪等天气（例如，Oxford RobotCar [7]）、春季、夏季、秋季和冬季等季节变化或旱季与雨季变化（例如，Nordland [44]）、经过的时间所带来的道路施工、施工现场、新建或拆除的建筑物，乃至现代影像与历史影像之间的差异（例如，AmsterTime [49]），或地震后的灾难场景等因素造成。**视点变化**指同一地点的图像之间从几乎像素对齐（例如，Nordland [44]）到人行道的左右两侧（例如，GardensPoint Walking [32]）、街道的左右两侧（例如，SouthBank Bicycle [32]）、全景对齐图像到单幅图像（例如，Tokyo 24/7 [50]）、全景对齐图像之间（例如，Pittsburgh250k [51]）、从自行车道到街道（例如，Berlin Kurfürstendamm [52]）、从空中到地面（例如，Danish Airs and Grounds [53]），或从室内到室外（例如，Amsterdam- XXXL [45]）的变化。有关现有数据集的全面概述，请参阅 [1], [3]。

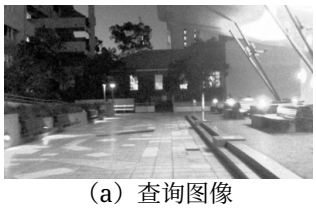
B. 真实值

真实值数据告诉我们，数据集中哪些图像对展示的是相同地点，哪些展示的是不同地点。评估地点识别方法的结果离不开这些数据。真实值可以直接以元组集合的形式给出，用于指出数据库 DB和查询集 Q 中的哪些图像属于同一地点；也可以根据 GNSS 坐标或位姿，并通过其最大允许距离来确定。此外，有些数据集经过采样，使得每个图像集中的同一索引图像展示相同地点（例如 Nordland [44] 和 GardensPoint Walking [32]）。

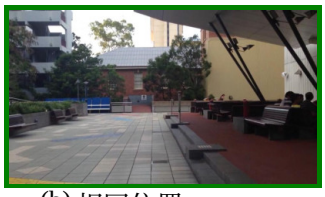
真实值的定义：要评估视觉地点识别结果，需要定义逻辑真实值矩阵 $\mathbf{GT}$。该矩阵与 $\mathbf{S}$ 和 $\mathbf{M}$的维度相同，即 $\mathbf{GT} \in \mathbb{B}^{|Q|\times|Q|}$ 或$\mathbf{GT} \in \mathbb{B}^{|DB|\times|Q|}$。元素 $gt_{ij} \in \mathbf{GT}$用于定义 Q 或 DB 中的第 i幅图像与 Q 中的第 j幅图像是否展示同一地点（$gt_{ij} = true$）或不同地点（$gt_{ij} = false$）。这些元素的值根据数据集中的真实值匹配进行设置。

研究人员采用的另一种视觉地点识别性能评估方法是软真实值矩阵$\mathbf{GT}^{soft}$。软真实值矩阵解决了这样一个问题：我们不期望视觉位置识别方法匹配视觉重叠度极低的图像，即 $gt_{ij} = false$，如图4所示。不过，如果某种方法确实匹配了这些重叠度较低的图像，我们会通过设置 $gt^{soft}_{ij} = true$ 来避免惩罚。完全没有视觉重叠的图像对也会被标记为 $gt^{soft}_{ij} = false$。因此，$\mathbf{GT}^{soft}$ 是$\mathbf{GT}$的膨胀版本，也就是说，它包含$\mathbf{GT}$中包含的所有真实值，以及针对视觉重叠度较低的图像对设置的额外真实值。如果

$$\mathbf{GT} = true\ . \tag{5}$$



（a）查询图像



(b) 相同位置，高视觉重叠，gt^{soft}_{ba}=true gt_{ba}=true



(c) 相同位置，低视觉重叠，真实值 $_{ca}$=错误，真实值$^{soft}_{ca}$=正确



(d) 不同位置，没有视觉重叠，真实值 $_{da}$=错误，真实值$^{soft}_{da}$=错误

图 4：查询图像 (a) 与参考图像 (b-d) 的视觉重叠程度及其对应真实值 gt_{ij} 和 gt^{soft}_{ij} 之间的关系。由于 (c) 与 (a) 仅存在较低的视觉重叠，我们不期望视觉位置识别方法同时匹配这两幅图像，因此设置 $gt_{ca} = false$。但是，如果视觉位置识别方法确实匹配了这两幅图像，我们也通过设置 $gt^{soft}_{ca} = true$ 来避免惩罚。

如果满足以下条件，图像对可以匹配，但不一定必须匹配：

$$\neg\mathbf{GT} \wedge \mathbf{GT}^{soft} = true \ . \tag{6}$$

请注意，我们使用 $\neg$ 表示逻辑否定运算符。这些通常在评估期间被忽略。如果

$$\mathbf{GT}^{soft} = false \ . \tag{7}$$

以下将介绍$\mathbf{GT}$ 和$\mathbf{GT}^{soft}$ 实际如何用于评估。

C. 指标

本节介绍用于评估视觉地点识别方法的既定指标，包括精确率和召回率、精确率-召回率曲线、精确率-召回率曲线下面积、100%精确率下的最大召回率以及 recall@K [5]。所有指标均已在相关代码仓库中实现（见第 I 节）。这些指标基于

- 成对描述符相似度 $s_{ij} \in \mathbf{S}$ （参见第 IV-D 节）或图像匹配结果 $m_{ij} \in \mathbf{M}$ （参见第 IV-E 节）
- 以及相应的真实值 $gt_{ij} \in \mathbf{GT}$ （在使用软真实值的情况下为$gt^{soft}_{ij} \in \mathbf{GT}^{soft}$ ）。

对于单一最佳匹配视觉位置识别，每个查询的评估仅考虑相似度最高的最佳匹配图像对 s_{i^*j}：

$$i^* = \arg\max_i s_{ij} \ . \tag{8}$$

精确率和召回率：精确率 P 和召回率 R 是信息检索领域中的重要指标 [54,第 781] 页。在视觉地点识别中，**精确率** P 表示正确匹配的相同地点图像数量占匹配图像总数的比例，其中

$$P = \frac{\#TP}{\#TP + \#FP} \ . \tag{9}$$

召回率 R 表示同一地点的正确匹配图像数量与真实正例（GTP）总数之比：

$$R = \frac{\#TP}{\#GTP} \ . \tag{10}$$

对于单一最佳匹配视觉地点识别，真实正例的数量是指存在真实匹配的查询图像总数，即：

$$\#GTP = \sum_{\forall j} \begin{cases} 1, & \text{if } \exists i : gt_{ij} \\ 0, & \text{otherwise} \end{cases} \ , \tag{11}$$

而对于多匹配视觉地点识别，真实正例的数量是指实际匹配的图像对数量，即：

$$\#GTP = \sum_{\forall i,j} \begin{cases} 1, & \text{if } gt_{ij} \\ 0, & \text{otherwise} \end{cases} \ . \tag{12}$$

$\#TP$ 和$\#FP$ 分别表示正确匹配和错误匹配的图像对数量。更具体地说，**真正例**TP是指被归类为匹配结果的实际匹配图像对：

$$\#TP = \sum_{\forall i,j} \begin{cases} 1, & \text{if } gt_{ij} \wedge m_{ij} \\ 0, & \text{otherwise} \end{cases} \ . \tag{13}$$

对于单一最佳匹配视觉地点识别，对于每个查询图像，式（8）中的 i^* 仅在式（13）中进行评估。后续的式（14）同样如此。

假正例 FP 是指被错误归类为匹配结果的不匹配图像对：

$$\#FP = \sum_{\forall i,j} \begin{cases} 1, & \text{if } \neg gt_{ij} \wedge m_{ij} \\ 0, & \text{otherwise} \end{cases} \ . \tag{14}$$

请注意，使用软真实值时，在计算 $\#TP$和 $\#FP$时会忽略带有 $\neg gt_{ij} \wedge gt^{soft}_{ij} = true$ （参见公式 (6)）的图像对。虽然**假负例**FN 会间接参与召回率 R的计算，但由于视觉地点识别的分类问题通常存在类别不平衡，**真负例** TN 通常不会被评估，其中 $\#TN \gg \#TP, \#FP, \#FN$。

精确率-召回率曲线：精确率-召回率曲线可用于避免实际的匹配决策；这些决策通常是在视觉地点识别之后，借助计算成本高昂的验证算法做出的。其思路是在一系列阈值 $\boldsymbol{\theta} = \{\min(\mathbf{S}) \ldots, \max(\mathbf{S})\}$范围内，通过 $\mathbf{M} = \mathbf{S} \geq \theta_k$来做出匹配决策。对于

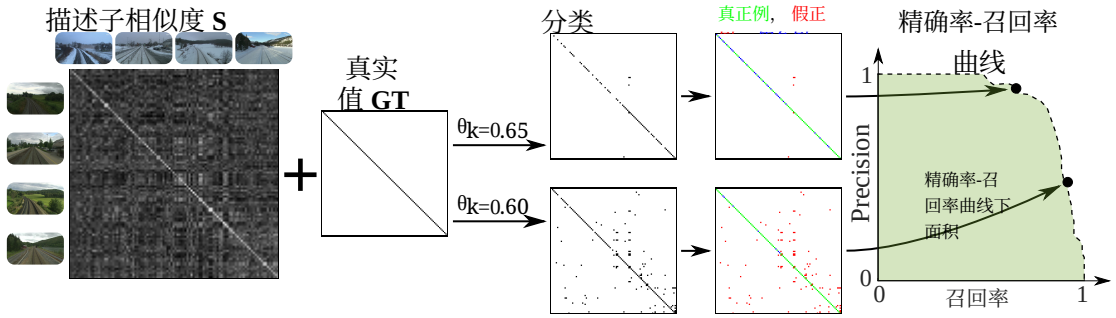


图 5：多匹配视觉位置识别的评估流程，包括精确率-召回率曲线及其曲线下面积（精确率-召回率曲线下面积）。给定相似度矩阵 $\mathbf{S}$ 和真实值 $\mathbf{GT}$ 以及 $\mathbf{GT}^{soft}$ （参见第 V-B 节），将一系列阈值$\theta_k \in \{\min(\mathbf{S}) \ldots, \max(\mathbf{S})\}$ 应用于 $\mathbf{S} \geq \theta_k$，以获得二元匹配决策 $m_{ij} \in \mathbf{M}_k$，针对每个 θ_k。结合真实值，这些决策可标记为真正例（TP）、假正例（FP）、假负例（FN）或真负例（TN），并转换为精确率-召回率曲线及其曲线下面积（精确率-召回率曲线下面积）。

例如，针对某个特定的$\theta_k \in \boldsymbol{\theta}$，真正例的数量 $\#TP_k$ 可通过以下方式计算：

$$\#TP_k = \sum_{\forall i,j} \begin{cases} 1, & \text{if } gt_{ij} \wedge (s_{ij} \geq \theta_k) \\ 0, & \text{otherwise} \end{cases}. \qquad (15)$$

根据式（9）和式（10），可得到两个分别表示精确率和召回率的向量 $P(\theta_k)$ 和 $R(\theta_k)$，二者共同构成**精确率-召回率曲线**。精确率-召回率曲线的完整计算流程如图5所示，代码见代码片段 5。

精确率-召回率曲线下面积（*AUPRC*）：**精确率-召回率曲线下面积**（也称平均精确率，avgP）可用于将精确率-召回率曲线压缩为单个数值，如代码片段6所示。图5中以绿色区域表示精确率-召回率曲线下面积。

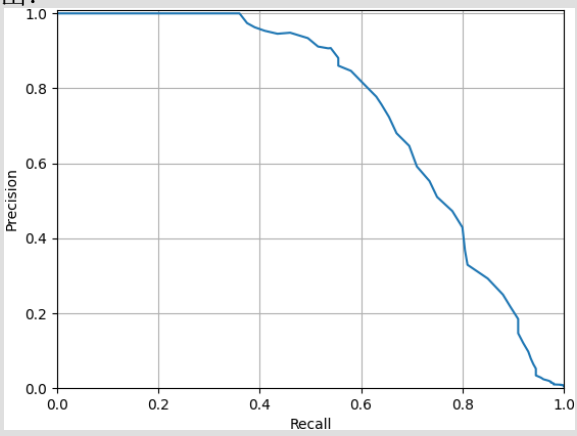
*100%*精确率下的最大召回率：100%精确率下的最大召回率（简称*R@100P*）表示在 $P = 1$ (100%) 时的最大召回率，即没有假正例时的最大召回率 FP（参见式（14））。过去，这项指标对于评估同步定位与建图（SLAM）中的回环检测视觉地点识别方法十分重要。将精确率保持在 $P = 1$ 可以避免错误闭环，从而避免地图构建错误 [55]。然而，随着面向同步定位与建图（SLAM）的鲁棒图优化技术问世 [56]，避免错误闭环已不再那么重要。采用鲁棒图优化时，找到足够多的正确闭环 (TP) 比避免错误闭环 (FP) 更重要。因此，对于此类应用，应优先采用多匹配视觉位置识别来识别所有闭环，而不是针对 R@100P 进行调优。

如果精确率始终未达到 $P = 1$，则 100% 精确率下的最大召回率未定义。因此，也可以改用 99% 或 95% 精确率下的最大召回率。

召回率@K：召回率@K （也称*top-K* 成功率）是图像分类器评估中常用的指标 [57,p. 225]。对于地点识别，其定义如下：对于每个查询图像，给定相似度最高的 K 个数据库图像 K s_{ij}，召回率@K 衡量

```
# 精确率-召回率曲线性能评估
# 用于多匹配视觉位置识别
来自evaluation.metrics导入 createPR
P, R = 创建PR(S, GT, 真值^soft, 匹配='多')
```

输出：



代码片段 5：为了评估相似度矩阵S的质量，我们可以应用一系列递减的阈值 $\boldsymbol{\theta}$，以匹配越来越多的图像对。结合真值信息，每个阈值都会产生一组不同的真正例、假正例、真负例和假负例，进而在精确率-召回率曲线上提供一个点。在本例中，我们为多匹配视觉地点识别创建精确率-召回率曲线。

至少有一张实际匹配的数据库图像的查询图像所占的比例。这意味着，对于每个查询图像，数据库中都必须至少有一张匹配图像，因此该指标适用于不涉及探索的典型定位场景。对于包含新访问地点的建图场景，该指标没有定义。在这种情况下，recall@K 的实现可以简单地忽略所有没有匹配数据库图像的查询图像——然而，这种权宜之计无法评估某种方法处理探索的能力或不足

```
# 使用曲线下面积评估性能
导入numpy 作为np
AUPRC = np.trapz(P, R)

输出:
精确率-召回率曲线下
面积：0.742
```

代码片段6：最后，为了用一个数概括地点识别质量，我们可以使用精确率-召回率曲线下面积（AUPRC）。

在查询运行期间，即不属于数据库集的新地点。

召回率@K 尤其适用于视觉定位任务：此时会检索最相似的 K 个数据库图像或查询图像，以便随后进行几何验证。请注意，对于无需探索的定位场景中的视觉地点识别（即所有查询图像都至少有一个匹配的参考图像），召回率@1与召回率为100%时的精确率相同。

平均性能、最佳情况性能和最差情况性能：为了全面了解视觉位置识别方法在不同环境以及不同类型的外观和视点变化下的表现，最佳做法是使用多个数据集进行评估。上述指标衡量的是单个数据集上的性能。通过考察平均性能、最佳情况性能和最差情况性能，可以更精炼地了解整体表现。平均性能有助于快速比较其他已评估方法。最佳情况性能显示了可达到的最高性能，并揭示了方法的潜在优势：如果某方法的最佳情况性能高于所比较的方法，则该方法非常适合实现这一最佳性能时所对应的条件。最差情况性能则揭示了方法的弱点，以及它对特定条件或轨迹的敏感程度（参见图3）。例如，如果某方法的最差情况性能低于所比较方法的最差情况性能，则表明该方法的鲁棒性较差，并且难以应对至少一个受评估数据集的特定属性。

值得指出的是，评估视觉地点识别方法时还有各种其他指标，包括将计算时间纳入考量的指标。感兴趣的读者可参阅 [5] ，其中全面概述了相关内容，还列举了同一算法在多个指标下的性能评估示例。

VI. 挑战及应对挑战的常见方法

前几节介绍了视觉地点识别（VPR）的一般流程，以及如何评估此类流程。本节将超越这一基本流程，列举研究人员在视觉地点识别领域面临的典型挑战，以及以往工作应对这些挑战的方法。

A. 可扩展性

视觉地点识别（VPR）面临的一项主要挑战是如何扩展系统，以处理数据库或查询集中的大量图像。如第 IV-B 节所述，整体图像描述符可实现快速检索。为减少描述符比较所需的计算量，研究人员提出了高斯、二值、符号或稀疏随机投影等降维技术 [58],[59], [60],，以及主成分分析（PCA） [61] 。

然而，地点识别的计算时间通常仍与数据库 DB 或查询集 Q中的图像数量成正比。为进一步提高效率，可以采用近似最近邻搜索 [42] （例如像 DELF [34]那样结合 KD 树 [62] 和乘积量化 [63] ），而不是对所有数据库描述符进行线性搜索，从而实现次线性时间复杂度。此外，利用较弱 GPS 信号提供的粗略位置数据，可以缩小搜索空间 [64]，提高效率。

最后，为弥补整体描述符准确率降低的问题，可以采用分层地点识别。这种方法通过使用局部图像描述符进行几何验证，对整体描述符检索到的排名前-K 匹配结果重新排序 [9], [65]。

B. 外观变化

机器人重访某个地点时，其当前图像观测的外观往往会发生显著变化（如第 V-A 节所述），这可能会对性能产生负面影响。为缩小查询观测与数据库中存储的观测之间的差异，可以采用光照不变图像 [66],、阴影去除 [67], [68], 、外观变化预测 [69], [70],、线性回归 [71], 以及基于生成对抗网络的深度学习方法 [72], [73] ，将所有图像转换为参考条件 [24]。此类技术需要通过人工监督或条件分类器（例如：数据库为夏季，查询为冬季）提供每幅图像与其实际条件之间的对应关系。

为避免采用这类仅针对特定条件进行训练或设计的特定条件方法（例如，Night-To-Day-GAN：夜间与白天；阴影去除：一天中的不同时段），可以使用按条件进行描述符标准化的方法，从而显著提升在广泛条件下的性能 [24]：这种标准化方法会将来自某一条件的描述符的每个维度归一化为零均值和单位标准差（例如，对夏季的数据库执行一次，对冬季的查询集执行一次）。此外，如果外观变化不仅发生在查询集与数据库遍历路线之间（例如，数据库：晴朗，查询：下雨），还发生在同一遍历路线内部（例如，数据库：晴朗→多云→阴天→下雨，查询：晴朗），则可以先对描述符进行聚类，再按聚类进行标准化。除了应对上述单独的外观变化挑战之外，近期研究的一个共同趋势是，在大规模、多样化数据集 [41]上训练深度架构，以获得对外观变化具有鲁棒性的全局 [74] 和局部 [75] 描述符

描述符。或者，也可以简单地将多个描述符连接起来（这会将它们的维度相加），或使用诸如超维计算之类的技术将它们组合起来（这种方法会限制维度），从而结合多个描述符的优势 [33]。

C. 视点变化

机器人可能会从不同视点重访某个地点。对于无人机，这种变化可能是由于六自由度位姿的变化；对于道路车辆，则可能是由于车道和行驶方向的变化。除了需要从不同视点识别局部特征或区域，还需要应对从不同视点拍摄的图像对之间通常有限的视觉重叠。当视点变化同时受到外观变化的影响时，问题会变得更具挑战性，因为外观变化扩大了感知混淆的可能范围（即不同地点看起来非常相似的问题，已在第 I 节讨论，并在例如 [2] 中详细说明）。应对视点变化的一种常用方法是以置换不变的方式聚合局部特征来学习整体描述符，也就是说，不依赖这些特征的像素位置，例如 NetVLAD [28]。

D. 提升性能

除了前面提到的方法外，还可以利用任务特定知识，通过多种方式提升视觉地点识别性能。

基于序列的方法利用数据库和查询集中的序列，从而在相似度矩阵$\mathbf{S}$ 中形成连续的高相似度线（参见第 IV-D 节和图3）。我们可以将这些方法分为两类：**基于相似度的序列方法**利用相似度 $s_{ij} \in \mathbf{S}$ 查找高相似度的线性片段，例如 SeqSLAM [17] 、SeqConv [20], ，或斜率可能变化的连续高相似度线，例如基于流网络 [76] 或隐马尔可夫模型[77]的方法。SMART [78] 等方法还会利用可用的里程计信息来查找斜率不断变化的序列。另一方面，可以将一系列整体图像描述符合并为单个向量。因此，为地点定义的**序列描述符**能够纳入前序图像帧 [79], [80]中观测到的视觉信息。通过比较数据库和查询集中的这些序列描述符，可以获得地点匹配假设。现有方法包括 ABLE [81]、MCN [82]、增量描述符 [83]、SeqNet [79],以及各种基于深度学习的方法[84], [80]。

除了利用数据库与查询集之间的描述子相似度 $\mathbf{S}$ *between* ，数据库与查询集内的相似度 **intra-database and intra-query sim-ilarities** $\mathbf{S}^{DB}$ **and** $\mathbf{S}^{Q}$（即数据库和查询集内部的描述子相似度）也可用于提升性能。例如，在 [20], 中，集合内相似度 $\mathbf{S}^{DB}$ 和 $\mathbf{S}^{Q}$与 $\mathbf{S}$及序列信息结合，构建了一个因子图，并通过优化该因子图来细化 $\mathbf{S}$中的相似度。在该图中，集合内相似度用于连接数据库或查询集内的图像；由于集合内相似度较高，这些图像很可能展示相同或不同的地点，这取决于其相似度是高还是低的集合内相似度。例如，假设第 l 个查询图像与第 i 个和第 j 个数据库图像之间的相似度分别为 s_{il} 和 s_{jl}，且都较高。进一步假设，其与第 k 个数据库图像之间的相似度 s_{kl} 较低，尽管第 i、 j 和 k 个数据库图像之间的数据库内相似度 s_{ij}^{DB}、s_{ik}^{DB} 和 s_{jk}^{DB} 均较高。随后，图优化会检测到，第 k 个数据库图像与第 l 个查询图像之间的相似度 s_{kl} 也很可能较高。

在机器人**经常重访相同地点**的情况下，可以使用经验地图 [85] 和共现地图 [86]等方法。这些“基于记忆”的方法会持续观察每个地点，并在每次外观发生变化时创建一个描述符。在将新的查询描述符与这一聚合“地图”进行比较时，只需匹配一个处于相似条件下的描述符即可识别该地点，从而减少对条件不变描述符 [87]的需求。一些方法进一步超越了这些基于记忆的技术，通过对时空动态进行建模，预测特征持久性 [88]、预期的室外条件 [89], 或地图占用情况 [90]。

在已知地点的机器人定位情况下（即，每个访问过的查询地点都保证存在于数据库中，并且不会在该地图区域之外进行探索），视觉地点识别可以受益于**地点专用分类器**，这种分类器能够在减少地图存储或检索时间的同时提高准确率 [91], [92]、[93], [94]。一种类似的方法是训练一个基于深度学习的**地点分类器**，直接为给定图像输出地点标签 [12], [95]，或者创建环境专用描述符 [96]。另一种方向是利用已知的地点类型进行地点类型匹配，以限制数据库与查询集之间潜在匹配结果的数量 [19]。例如，如果查询图像是在森林中拍摄的，就无需搜索所有数据库图像；这种语义分类会将数据库图像限制为同样在森林中拍摄的图像。

VII. 结论

视觉地点识别（VPR）是一个成熟的问题，在计算机视觉和机器人学领域都广受关注并得到广泛应用。在本教程中，我们介绍了视觉地点识别任务，包括其不同的问题类别和子类型、典型用例，以及通常的实现和评估方式。此外，我们还讨论了多种可用于应对视觉地点识别中常见挑战的方法。

仍有许多尚待解决的挑战，例如系统集成、内容更丰富的参考地图、视图合成，以及设计一种“适用于所有情况”的解决方案。尽管本教程未讨论这些挑战，我们仍建议感兴趣的读者参阅 [1], [2]。

参考文献

[1] S. Schubert 和 P. Neubert，《是什么让视觉地点识别变得容易或困难？》*arXiv 2106.12671*，2021年。[2] S. Garg、T. Fischer 和 M. Milford，《你的地点在哪里，视觉地点识别？》，载于国际人工智能联合会议（*IJCAI*），2021年，第 4416–4425 页。[3] C. Masone 和 B. Caputo，《深度视觉地点识别综述》，*IEEE Access*，第 9 卷，第 19 516–19 547 页，2021年。

[4] S. Lowry、N. S¨underhauf、P. Newman、J. J. Leonard、D. Cox、P. Corke 和 M. J. Milford，《视觉地点识别：综述》，IEEE机器人学汇刊（TRO），第 32 卷，第 1–19 页，2016年。[5] M. Zaffar、S. Garg、M. Milford、J. Kooij、D. Flynn、K. McDonald-Maier 和 S. Ehsan，《VPR-Bench：一个具有可量化视点和外观变化的开源视觉地点识别评估框架》，《国际计算机视觉杂志》（IJCV），第 1 39 页，2021年 – 。[6] X. Zhang 、 L. Wang 和 Y. Su，《视觉地点识别：从深度学习视角出发的综述》，模式识别，第 113 卷，2021年。[7] W. Maddern、G. Pascoe、C. Linegar 和 P. Newman，《1 年，1000 千米：Oxford RobotCar 数据集》，《国际机器人研究期刊》（IJRR），第 36 卷，第 1 期，第 3–15 页，2017年。[8] K. A. Tsintotas、L. Bampis 和 A. Gasteratos，《同步定位与建图中的重访问题：视觉回环检测综述》，IEEE智能交通系统汇刊（T-ITS），第 23 卷，第 11 期，第 19 929–19 953 页，2022年。[9] M. Cummins 和 P. Newman，《FAB-MAP：外观空间中的概率定位与建图》，《国际机器人研究期刊》（IJRR），第 27 卷，第 6 期，第 647–665 页，2008年。[10] R. Mur-Artal、J. M. M. Montiel 和 J. D. Tard´os，《ORB-SLAM：一种灵活且精确的单目SLAM系统》，IEEE机器人学汇刊（TRO），第 31 卷，第 5 期，第 1147–1163 页，2015年。[11] P. Yin、S. Zhao、I. Cisneros、A. Abuduweili、G. Huang、M. Milford、C. Liu、H. Choset 和 S. Scherer，《通用地点识别综述：迈向现实世界自主性的时代》，arXiv 2209.04497，2022年。[12] T. Weyand、I. Kostrikov 和 J. Philbin，《PlaNet——利用卷积神经网络进行照片地理定位》，载于欧洲计算机视觉会议（ECCV），2016年，第 37–55 页。[13] H. Durrant-Whyte 和 T. Bailey，《同步定位与建图：第一部分》，IEEE机器人与自动化杂志（RA-M），第 13 卷，第 2 期，第 99–110 页，2006 年。[14] D. G´alvez-L´opez 和 J. D. Tardos，《用于图像序列快速地点识别的二进制词袋》，IEEE机器人学汇刊（TRO），第 28 卷，第 5 期，第 1188–1197 页，2012年。[15] C. Valgren 和 A. J. Lilienthal，《SIFT、SURF 与季节：室外环境中基于外观变化的长期定位》，《机器人与自主系统》，第 58 卷，第 2 期，第 149–156 页，2010年。[16] F. Dayoub 和 T. Duckett，《用于移动机器人长期拓扑定位的自适应外观地图》，载于 IEEE/RSJ智能机器人与系统国际会议（IROS），2008 年，第 3364–3369 页。[17] M. J. Milford 和 G. F. Wyeth，《SeqSLAM：适用于晴朗夏日和暴风雪冬夜的视觉路线导航》，载于 IEEE机器人与自动化国际会议（ICRA），2012年，第 1643–1649 页。[18] Z. Chen、O. Lam、A. Jacobson 和 M. Milford，《基于卷积神经网络的地点识别》，澳大拉西亚机器人与自动化会议（ACRA），2014年。[19] N. S¨underhauf、S. Shirazi、F. Dayoub、B. Upcroft 和 M. Milford，《卷积网络特征在地点识别中的性能》，载于 IEEE/RSJ智能机器人与系统国际会议（IROS），2015年，第 4297–4304 页。[20] S. Schubert、P. Neubert 和 P. Protzel，《通过 ICM 实现视觉地点识别中的快速且高效内存的图优化》，载于机器人学：科学与系统（RSS），2021年。[21] M. F. Aslan、A. Durdu、A. Yusefi、K. Sabanci 和 C. Sungur，《教程：移动机器人、SLAM、贝叶斯滤波器、关键帧光束法平差与 ROS 应用》。施普林格国际出版，2021年，第 227–269 页。[22] D. Scaramuzza 和 F. Fraundorfer，《视觉里程计 [教程],》，IEEE机器人与自动化杂志（RA-M），第 18 卷，92，2011年 4，80第。页 – 。[23] O. Vysotska 和 C. Stachniss，《使用多序列地图实现有效的视觉地点识别》，IEEE机器人与自动化快报（RA-L），第 4 卷，第 2 期，第 1730–1736 页，2019年。[24] S. Schubert、P. Neubert 和 P. Protzel，《离散变化和连续变化环境中的视觉地点识别无监督学习方法》，载于 IEEE机器人与自动化国际会议（ICRA），2020年，第 4372–4378 页。[25] K. Ho 和 P. Newman，《利用场景序列检测回环》，《国际计算机视觉杂志》（IJCV），第 74 卷，第 3 期，第 261–286 页，2007 年。[26] T. Sattler 、 W. Maddern 、 C. Toft、A. Torii、L. Hammarstrand、E. Stenborg、D. Safari、M. Okutomi、M. Pollefeys、J. Sivic、F. Kahl 和 T. Pajdla，“在变化条件下对六自由度户外视觉定位进行基准测试”，载于 IEEE计算机视觉与模式识别会议 (CVPR)，2018年，第 8601–8610 页。[27] M. Dusmanu、I. Rocco、T. Pajdla、M. Pollefeys、J. Sivic、A. Torii 和 T. Sattler，“D2-Net：一种用于联合描述和检测局部特征的可训练卷积神经网络”，载于 IEEE计算机视觉与模式识别会议 (CVPR)，2019年，第 8092–8101 页。[28] R. Arandjelovic、P. Gronat、A. Torii、T. Pajdla 和 J. Sivic，“NetVLAD：用于弱监督地点识别的卷积神经网络架构”，载于 IEEE计算机视觉与模式识别会议 (CVPR)，2016年，第 15 297–15 307 页。[29] U. Frese、R. Wagner 和 T. R¨ofer，“从用户视角概述同步定位与建图”，《德国人工智能期刊》，第 24 卷，第 3 期，第 191–198 页，2010年。[30] J. McDonald、M. Kaess、C. Cadena、J. Neira 和 J. J. Leonard，“使用锚节点的六自由度多会话视觉同步定位与建图”，载于欧洲移动机器人会议 (ECMR)，2011年，第 69–76 页。[31] T. Cieslewski、S. Choudhary 和 D. Scaramuzza，“数据高效的分布式视觉同步定位与建图”，载于 IEEE机器人与自动化国际会议 (ICRA)，2018年，第 2466–2473 页。[32] A. Glover，“白天与夜晚，左边与右边”，2014年。[33] P. Neubert 和 S. Schubert，“将超维计算作为系统化聚合图像描述符的框架”，载于 IEEE/CVF计算机视觉与模式识别会议 (CVPR)，2021年，第 16 938–16 947 页。[34] H. Noh、A. Araujo、J. Sim、T. Weyand 和 B. Han，“利用注意力深度局部特征进行大规模图像检索”，载于 IEEE国际计算机视觉会议 (ICCV)，2017年，第 3456–3465 页。[35] P.-E. Sarlin、D. DeTone、T. Malisiewicz 和 A. Rabinovich，“SuperGlue：利用图神经网络学习特征匹配”，载于 IEEE计算机视觉与模式识别会议 (CVPR)，2020年，第 4938–4947 页。[36] E. Garcia-Fidalgo 和 A. Ortiz，“iBoW-LCD：一种使用增量二进制词袋的基于外观的回环检测方法”，《IEEE机器人与自动化快报》，第 3 卷，3057，2018 4，第 3051 页。[37] Sivic 和 Zisserman，“Video Google：一种用于视频中目标匹配的文本检索方法”，载于 IEEE国际计算机视觉会议 (ICCV)，2003年。[38] H. J´egou、M. Douze、C. Schmid 和 P. P´erez，“将局部描述符聚合为紧凑图像表示”，载于 IEEE计算机视觉与模式识别会议 (CVPR)，2010 年，第 3304– 3311 页。[39] Z. Chen 、 A. Jacobson 、 N. S¨underhauf、B. Upcroft、L. Liu、C. Shen、I. Reid 和 M. Milford，“用于视觉地点识别的大规模深度学习特征”，载于 IEEE机器人与自动化国际会议 (ICRA)，2017年，第 3223–3230 页。[40] J. Garforth 和 B. Webb，“迷失林中？困难森林环境中的导航地点识别”，《机器人与人工智能前沿》，第 7 卷，第 186 页，2020年。[41] F. Warburg、S. Hauberg、M. Lopez-Antequera、P. Gargallo、Y. Kuang 和 J. Civera，“Mapillary街景序列：用于终身地点识别的数据集”，载于 IEEE/CVF计算机视觉与模式识别会议 (CVPR)，2020年，第 2626–2635 页。[42] W. Li、Y. Zhang、Y. Sun、W. Wang、M. Li、W. Zhang 和 X. Lin，“高维数据上的近似最近邻搜索——实验、分析与改进”，IEEE知识与数据工程汇刊，第 32 卷，1488，2020 8，第 1475 页。[43] S. Schubert、P. Neubert 和 P. Protzel，“超越近似最近邻：利用结构知识实现高效地点识别”，载于 IEEE机器人与自动化国际会议 (ICRA)，2021年，第 5861–5867 页。[44] N. S¨underhauf、P. Neubert 和 P. Protzel，“我们到了吗？在横跨四季、全长 3000 公里的旅程中挑战 SeqSLAM”，载于 IEEE机器人与自动化国际会议 (ICRA) 长期自主性研讨会，2013年。[45] S. Ibrahimi、N. van Noord、T. Alpherts 和 M. Worring，“由内而外的视觉地点识别”，载于英国机器视觉会议 (BMVC)，2021年。[46] A. J. Glover、W. P. Maddern、M. J. Milford 和 G. F. Wyeth，“FAB-MAP + RatSLAM：用于一天中不同时段的基于外观的同步定位与地图构建”，载于 IEEE机器人与自动化国际会议 (ICRA)，2010年，第 3507–3512 页。

[47] J. Bruce、J. Wawerla 和 R. Vaughan，“SFU 山地数据集：变化环境条件下的半结构化林地小径”，载于 IEEE机器人与自动化国际会议 (ICRA) 变化环境中的视觉地点识别研讨会，2015年。[48] S. Griffith、G. Chahine 和 C. Pradalier，“Symphony 湖泊数据集”，《国际机器人研究期刊》(IJRR)，第 36 卷，第 11 期，第 1151–1158 页，2017年。[49] B. Yildiz、S. Khademi、R. M. Siebes 和 J. van Gemert，“AmsterTime：用于严重域偏移的视觉地点识别基准数据集”，载于国际模式识别会议 (ICPR)，2022年，第 2749–2755 页。[50] A. Torii、R. Arandjelovi´c、J. Sivic、M. Okutomi 和 T. Pajdla，“通过视图合成实现全天候地点识别”，载于 IEEE/CVF计算机视觉与模式识别会议 (CVPR)，2015年，第 1808–1817 页。[51] A. Torii、J. Sivic、M. Okutomi 和 T. Pajdla，“利用重复结构进行视觉地点识别”，IEEE《模式分析与机器智能汇刊》(TPAMI)，第 37 卷，第 11 期，第 2346–2359 页，2015年。[52] N. S¨underhauf、S. Shirazi、A. Jacobson、F. Dayoub、E. Pepperell、B. Upcroft 和 M. Milford，“利用卷积网络地标进行地点识别：视点鲁棒、条件鲁棒、免训练”，载于机器人学：科学与系统 (RSS)，2015年。[53] A. Vallone、F. Warburg、H. Hansen、S. Hauberg 和 J. Civera，“Danish Airs and Grounds：用于从空中到街道层级地点识别与定位的数据集”，IEEE机器人与自动化快报 (RA-L)，第 7 卷，第 4 期，第 9207–9214 页，2022年。[54] C. Sammut 和 G. I. Webb 编，《机器学习百科全书》。马萨诸塞州波士顿：Springer 美国，2010年。[55] M. Magnusson、H. Andreasson、A. Nuchter 和 A. J. Lilienthal，“使用正态分布变换从三维激光数据中进行基于外观的回环检测”，载于 IEEE机器人与自动化国际会议 (ICRA)，2009年，第 23–28 页。[56] N. S¨underhauf 和 P. Protzel，“用于鲁棒位姿图同步定位与地图构建的可切换约束”，载于 IEEE 国际智能机器人与系统会议 (IROS)，2012年，第 1879–1884 页。[57] O. Russakovsky、H. Deng、Jia Su、J. Krause、S. Satheesh、S. Ma、Z. Huang、A. Karpathy、A. Khosla、M. Bernstein、A. C. Berg 和 L. Fei-Fei，“ImageNet 大规模视觉识别挑战赛”，《国际计算机视觉杂志》(IJCV)，第 115 卷，第 3 期，第 211–252 页，2015年。[58] S. Dasgupta，“随机投影实验”，载于人工智能不确定性会议，2000年。[59] D. Achlioptas，“数据库友好型随机投影：使用二元随机变量实现 Johnson-Lindenstrauss 变换”，《计算机与系统科学杂志》，第 66 卷，第 4 期，第 671–687 页，2003年 06 月。[60] P. Li、T. J. Hastie 和 K. W. Church，“利用边际信息改进随机投影”，载于计算学习理论国际会议，2006年，第 635–649 页。[61] Y. Liu 和 H. Zhang，“使用紧凑图像描述符进行视觉回环检测”，载于 IEEE 国际智能机器人与系统会议 (IROS)，2012年，第 1051–1056 页。[62] J. L. Bentley，“用于关联搜索的多维二叉搜索树”，《ACM 通讯》，第 18 卷，517，1975年 9 月，第 509 页。[63] H. Jegou、M. Douze 和 C. Schmid，“用于最近邻搜索的乘积量化”，IEEE《模式分析与机器智能汇刊》(TPAMI)，第 33 卷，第 1 期，第 117–128 页，2010年。[64] O. Vysotska、T. Naseer、L. Spinello、W. Burgard 和 C. Stachniss，“利用 GPS 先验，在显著外观变化下高效且有效地匹配图像序列”，载于 IEEE机器人与自动化国际会议 (ICRA)，2015年，第 2774–2779 页。[65] S. Hausler、S. Garg、M. Xu、M. Milford 和 T. Fischer，“Patch-NetVLAD：用于地点识别的局部-全局描述符多尺度融合”，载于 IEEE/CVF计算机视觉与模式识别会议 (CVPR)，2021年，第 14 141–14 152 页。[66] W. Maddern、A. Stewart、C. McManus、B. Upcroft、W. Churchill 和 P. Newman，“ 光照不变成像：在自动驾驶车辆鲁棒视觉定位、建图和分类中的应用”，载于机器人与自动化国际会议 (ICRA) 变化环境中的视觉地点识别研讨会，2014年。[67] C. McManus、W. Churchill、W. Maddern、A. D. Stewart 和 P. Newman，“暗处交易：利用光照不变性实现鲁棒的长期视觉定位”，载于 IEEE机器人与自动化国际会议 (ICRA)，2014年，第 901–906 页。[68] P. Corke、R. Paul、W. Churchill 和 P. Newman，“应对阴影：捕捉场景固有外观以实现基于图像的室外定位”，载于 IEEE 国际智能机器人与系统会议（IROS），2013年，页码 2085–2092。[69] P. Neubert、N. S¨underhauf 和 P. Protzel，“基于超像素的外观变化预测：实现跨季节的长期导航”，《机器人与自主系统》，第69卷，页码 15–27，2015年。[70] P. Neubert、N. S¨underhauf 和 P. Protzel，“跨季节长期导航的外观变化预测”，载于欧洲移动机器人会议（EC MR），2013年，页码 198–203。[71] S. Lowry 和 M. J. Milford，“用于变化环境中视觉地点识别的有监督与无监督线性学习技术”，IEEE机器人学汇刊（TRO），第32卷，613，2016年 3，600期，页码 –。[72] H. Porav、W. Maddern 和 P. Newman，“恶劣条件下的对抗训练：利用外观迁移实现鲁棒度量定位”，载于 IEEE机器人与自动化国际会议（ICRA），2018年，页码 1011–1018。[73] A. Anoosheh、T. Sattler、R. Timofte、M. Pollefeys 和 L. Van Gool，“用于基于检索的定位的夜间到白天图像转换”，载于 IEEE机器人与自动化国际会议（ICRA），2019年，页码 5958–5964。[74] G. Berton、C. Masone 和 B. Caputo，“重新思考大规模应用中的视觉地理定位”，载于 IEEE/CVF计算机视觉与模式识别会议（CVPR），2022年，页码 4878–4888。[75] B. Cao、A. Araujo 和 J. Sim，“融合深度局部与全局特征以实现图像搜索”，载于欧洲计算机视觉会议（ECCV），2020年，页码 726–743。[76] T. Naseer、L. Spinello、W. Burgard 和 C. Stachniss，“利用网络流实现跨季节的鲁棒视觉机器人定位”，载于 AAAI人工智能会议，2014年，页码 2564–2570。[77] P. Hansen 和 B. Browning，“使用 HMM序列匹配进行视觉地点识别”，载于 IEEE 国际智能机器人与系统会议（IROS），2014年，页码 4549–4555。[78] E. Pepperell、P. I. Corke 和 M. J. Milford，“借助 SMART 实现全环境视觉地点识别”，载于 IEEE机器人与自动化国际会议（ICRA），2014年，页码 1612–1618。[79] S. Garg 和 M. Milford，“SeqNet：用于基于序列的分层地点识别的描述符学习”，IEEE机器人与自动化快报（RA-L），第6卷，第3期，页码 4305–4312，2021年。[80] R. Mereu、G. Trivigno、G. Berton、C. Masone 和 B. Caputo，“学习用于基于序列的视觉地点识别的序列描述符”，IEEE机器人与自动化快报（RA-L），第7卷，第4期，页码 10 383–10 390，2022年。[81] R. Arroyo、P. F. Alcantarilla、L. M. Bergasa 和 E. Romera，“通过高效匹配图像中的二进制序列实现终身视觉定位”，载于 IEEE机器人与自动化国际会议（ICRA），2015年，页码 6328–6335。[82] P. Neubert、S. Schubert 和 P. Protzel，“一种受神经学启发的序列处理模型，用于移动机器人地点识别”，IEEE机器人与自动化快报（RA-L），第4卷，3207，2019年 4，3200期，页码 –。[83] S. Garg、B. Harwood、G. Anand 和 M. Milford，“增量描述符：基于变化的地点表示，实现鲁棒视觉定位”，IEEE机器人与自动化快报（RA-L），第5卷，5127，2020年 4，5120期，页码 –。[84] J. M. Facil、D. Olid、L. Montesano 和 J. Civera，“条件不变的多视图地点识别”，2019年。[85] W. Churchill 和 P. Newman，“基于经验的长期定位导航”，《国际机器人研究期刊》（IJRR），第32卷，第14期，页码 1645–1661，2013年。[86] E. Johns 和 G. Yang，“特征共现图：全天候的基于外观的定位”，载于 IEEE机器人与自动化国际会议（ICRA），2013年，页码 3212–3218。[87] T. L. Molloy、T. Fischer、M. Milford 和 G. N. Nair，“通过贝叶斯选择性融合智能筛选视觉地点识别的参考数据”，IEEE机器人与自动化快报（RA-L），第6卷，第2期，页码 588–595，2020年。[88] D. M. Rosen、J. Mason 和 J. J. Leonard，“迈向半静态环境中的终身特征建图”，载于 IEEE机器人与自动化国际会议（ICRA），2016年，页码 1063–1070。[89] C. Linegar、W. Churchill 和 P. Newman，“巧干而非苦干：回忆相关经验，以实现超大规模但受时间限制的

定位”，载于 *IEEE* 国际机器人与自动化会议（*ICRA*），2015年，页码 90–97。[90] T. Krajn´ık、J. P. Fentanes、J. M. Santos 和 T. Duckett，“FreMEn：频率地图增强，实现变化环境中的长期移动机器人自主性”，*IEEE*机器人学汇刊（*TRO*），第33卷，第4期，页码 964–977，2017年。[91] C. McManus、B. Upcroft 和 P. Newman，“学习用于长期基于视觉定位的地点依赖特征”，*Autonomous Robots*，第39卷，页码 363–387，2015年。[92] P. Gronat、G. Obozinski、J. Sivic 和 T. Pajdla，“学习并校准用于视觉地点识别的每地点分类器”，载于 *IEEE*计算机视觉与模式识别会议（*CVPR*），2013年，页码 907–914。[93] C. McManus、B. Upcroft 和 P. Newmann，“场景签名：用于定位的局部化无点特征”，载于 机器人学：科学与系统（*RSS*），2014年。[94] N. V. Keetha、M. Milford 和 S. Garg，《用于视觉地点识别的环境与地点特定效用分层双重模型》，*IEEE*机器人与自动化快报 *(RA-L)*，第 6 卷，第 4 期，第 6969–6976 页，2021年。[95] S. Hussaini、M. Milford 和 T. Fischer，《通过加权神经元分配实现视觉地点识别的脉冲神经网络》，*IEEE*机器人与自动化快报 *(RA-L)*，第 7 卷，4101，2022 2，第 4094 页–。[96] P. Neubert 和 S. Schubert，《SEER：图像描述符的无监督且样本高效的环境专门化》，载于机器人学：科学与系统 *(R:SS)*，2022年。