
ORB-SLAM: A Versatile and Accurate Monocular SLAM System

摘要：本文提出一个实时的基于特征法的单目同步定位与视觉算法，可以应用于不同尺度大小的室内和室外环境。整个系统对剧烈运动下的 SLAM 有较好的鲁棒性，还可以用于宽基线的闭环检测，重定位，可以完成自动初始化。我们重新设计了一个 SLAM 系统，本系统结构与基于特征的 SLAM 系统相同，包括跟踪、建图、重定位和闭环检测。利用点云和关键帧进行地图重构，系统具有很好的鲁棒性，生成精简、可追踪的稀疏地图，并且当场景不断变化时，地图构建进程仍旧可以继续运行。对 27 个数据集进行测试，与当前优秀的单目 SLAM 算法相比，ORB-SLAM 算法具有卓越的性能。

1 介绍

BA 提供一个强健的匹配网络和初始估计，可以用于精确求解相机位姿和稀疏三维重构，由于运算量的问题，之前很难做到实时处理，现在随着硬件的发展，可以应用到 VSLAM 中。vSLAM 的目标是在恢复周围环境的同时估计相机的轨迹。VSLAM 中 BA 需要的输入和要求如下。

- (1) 从一系列的关键帧中匹配场景的特征点（地图点云）。
- (2) 随着关键帧数量的增加，进行筛选优化，避免冗余。
- (3) 强健的关键帧和点云的网络配置可以得到精确的结果，换言之通过分布合理的有明显视差的关键帧观测到的点云和良好的闭环检测匹配可以提高精度。
- (4) 关键帧位姿估计和点云定位。
- (5) 构建局部地图时，优化的重点是可扩展性。
- (6) 闭环检测可以实时的进行快速的全局优化（如位姿图）

PTAM 提供了一种高效的方法解决关键帧选取，特征匹配，点云的三角法测量，帧中相机定位，构图和追踪丢失后的重定位。但是没有闭环检测，缺少阻塞处理，特征点云的视图不变性差，在初始化地图时需要人工干预。

ORB 继承了 PTAM 的有点，增加了场景识别，带有尺度感知的闭环检测，利用covisibility 信息进行大场景处理等工作，ORB 贡献如下。

- (1) 所有的任务都使用相同的特征，ORB 特征，具有良好的视角不变性和光照不变性，可以在无 GPU 的环境下实时运行。
- (2) 大场景下可以实时运行。使用covisibility 图，只关注局部的 covisible 区域，与全局地图分离。
- (3) 利用 pose 图（Essential 图）进行实时的闭环检测。根据系统的 spanning tree，闭环检测连接和 covisibility 强连接边，生成 pose graph（Essential graph）
- (4) 利用 ORB 特征的视图和光照不变性实时进行相机重定位
- (5) 根据模型选择判据自动完成平面和非平面地图场景初始化，鲁棒性好

-
- (6) 一种合理的挑选关键帧和点云的方法，该方法具有松宽的创建要求和严苛的筛选要求，舍弃冗余的关键帧保证跟踪进程的鲁棒性和算法运行的持续性。

在当下主流的数据集上做了大量的测试。相机的定位精度高于当下先进的直接法 SLAM,

2 相关工作

A. 位置识别

论文 13 比较了几种基于外形特征的位置识别技术。图像到图像的匹配方法效果比图像到地图和地图到地图的方法好。在基于特征的方法中，词袋算法(基于概率学的 FAB-MAP)因为其效率更高而受关注。DBoW2 利用 BRIEF 描述子和 FAST 特征点获取二进制词袋。相比于用 SURF 和 SIFT 特征产生的词袋，该算法特征提取处理时间降低了一个数量级。使用 BRIEF 描述子虽然高效，鲁棒性强，但是不具备尺度和旋转不变性，限制了系统在平面轨迹和闭环检测时对于相似视角检测的应用。在之前的工作，利用 DBoW2 生成了 ORB 词袋位置模型识别器。ORB 是一种具有旋转不变性和尺度不变性的二进制特征，可以作为高效的视角不变性的识别器。对 ORB 特征在 4 个数据集上进行测试，从 10K 个图像数据集中提取 1 个闭环回路只需要 39ms(包括特征提取)，该识别器鲁棒性好，查全率高。本次采用 ORB 的提高版，利用 covisibility 信息在检索数据库时返回几个假设而不是最佳匹配。

B. 地图初始化

单目无法直接从单幅图像中恢复深度信息，需要创建一个初始化地图。一种解决办法是最初追踪一个已知的目标。在滤波方法中，利用逆深度参数化的方法，该方法初始化的点云深度具有较大的不确定性，之后再与他的实际位置进行融合。LSD 采用了相似的方法，利用高方差的随机数初始化像素的深度。

初始化的方法一般有两种，一种是假设当前观测的局部场景是一个平面，利用单应矩阵恢复相机的相对位姿。另一种是利用五点法计算本征矩阵，根据矩阵构建二维平面或场景模型。当场景中所有的点云太过靠近相机而引起视差太小或者图像模糊时，这两种方法无法得到较好的结果。如果当前场景不在一个平面上时，有足够的视差且图像不模糊。可以利用 8 点法解算基本矩阵，从而建模场景内容。实际上本征矩阵是 $E = \hat{t} \times R$ 。

本文第四章提出了一个方法用于进行地图初始化，自动的根据场景是否处于一个平面选择使用单应矩阵还是基本矩阵。通过考虑选用基本矩阵的风险(近似平面)，倾向于选择单应矩阵。如果求得的解具有二义性，则延迟初始化直到有足够的视差。

C. 单目 SLAM

单目最开始使用滤波方法，每一帧图像都要通过滤波器相互关联并估计地图中特征点的位置和相机位置，导致即使连续帧间只有很小的变化也不得不得耗费很大的计算量重新计算所有帧，并且带有线性的累计误差。另一方面，基于关键帧的方法估计地图不需要考虑帧率，只需要使用关键帧进行 BA 优化获得更高的精度。

最具代表性的基于关键帧法的 SLAM 是 PTAM。PTAM 将特征追踪与重构分成两个并行的线程，并且可以在小场景范围内实时的应用于 AR 中。之后 PTAM 算法被完善，采用边缘特征，在跟踪过程中进行旋转估计，并且优化了重定位方法。利用补丁将 PTAM 的地图点云与 FAST 角点相对应。使得点云不仅可以用于特征跟踪，还可以用于位置识别。但是，PTAM 不能进行大范围闭环检测，重定位是利用关键帧低分辨率的缩略图，视角不变性很差。

St. 提出了一个大范围单目 SLAM 算法，前端是实时运行在 GPU 上的光流，FAST 特征匹配和运动 BA，后端窗口是一个滑动窗口 BA，闭环检测使用相似性约束的位姿图优化，可以校正尺度漂移的偏差。ORB 使用的是 7DOF 的位姿优化并且用于 essential 图。

之后 St. 使用了 PTAM 的前端，仅仅根据 covisibility 图跟踪局部地图。他提出了一种双窗口优化作为后端，可以在内窗口运行 BA，在外窗口运行位姿图优化，然而闭环检测仅仅在外窗口包含全部回路时才有效果。在 ORB 系统中，继承了这种思想。使用基于 covisibility 局部地图并且根据 covisibility 图中建立 pose 图。不同的是将其用于两个重新设计的系统，并且特征跟踪和点云重构中采用位置识别，使重定位和闭环检测有较好的鲁棒性。

LSD-SLAM 使用直接法代替特征 BA 完成大场景下的半稠密地图的构建 LSD-SLAM 在无需 GPU 加速的情况下可以实时运行，相比稀疏地图，对于机器人更有应用前景和价值。然而，完成闭环检测必须依赖基于特征的 FAB-MAP 词袋法，而且定位精度低。

介于直接法和特征法之间的是 SVO，不需要在每一帧进行特征提取，可以运行在高帧率的环境下。但是无法进行闭环检测，并需要下视摄像头。

最后，我们想介绍一下关键帧的选取。对于 SLAM 来说，将全部点云和帧输入 BA 是不可行的，只能在没有冗余关键帧的情况下尽可能保留多的点。PTAM 为了避免引起运算复杂度的过量增长，关键帧的选取策略比较严苛。这导致在复杂环境探索时跟踪会失败。ORB 在复杂场景时，尽可能快的插入关键帧，之后剔除掉冗余关键帧，避免冗余运算。整个算法鲁棒性好。

3 系统概述

A. 特征选取

ORB 算法的核心思想是采用相同的特征完成特征跟踪、地图重建、位置识别，基于帧速率的重定位和闭环检测。提高系统的运行效率，避免之前工作中出现的在 SLAM 特征附近插值获得识别特征的深度。本文的算法，从一帧图像中完成特征提取的花费时间少于 33ms。为了获得一般性的位置识别方法，所选特征需要满足旋转不变性，BRIEF 和 LDB 不具有这样的性质。

本文选择 ORB 特征，ORB 特征是有着 256bit 描述子的多尺度 FAST 角点。具有良好的视角不变性，并且匹配计算的速度非常快。ORB 特征可以在宽基线上进行配准，提高 BA 的精度。

B. 多线程：特征跟踪，局部地图重构和闭环检测

ORB-SLAM 由三个线程并行执行。特征跟踪，局部地图重构和闭环检测。特征跟踪负责通过每帧图像定位机位置并决定何时插入新的关键帧。首先通过前一帧初始化特征匹配并且使用运动 BA 优化位姿。如果跟踪丢失，位置识别模

块开始进行全局重定位。当完成相机位姿初始化估计和特征匹配后，从系统关键帧的 covisibility 图中提取一个可见的局部地图，然后局部地图中匹配到的点云被重投影，相机的位姿通过所有匹配点再次被优化。最后，特征跟踪线程决定是否插入新的关键帧。跟踪算法的详细介绍见第 5 章。地图初始化见第 4 章。

局部地图构建进程插入新生成的关键帧并执行 BA 完成相机位姿附近环境的重构优化。搜索 covisibility 图中与新关键帧相连的关键帧的点云与新加入关键帧中未匹配的 ORB 特征点进行配对，利用三角法求解新的点云。有时尽管已经完成了点云图的构建，为了保证点云图的质量，根据特征跟踪线程中收集到的新信息，可能会采用一个更为严苛的点云筛选规则。局部地图构建进程也负责剔除冗余的关键帧。

闭环检测进程搜索回路中的每一个关键帧，验证是否形成闭环。当一个回路被检测到时，计算相似变换，相似变换包含了该回路的累计漂移。之后回路的两端被连接在一起并且重复的点云被融合。最后，在相似约束下，通过执行位姿图优化完成全局一致性检验。主要的创新点是把这个优化通过 Essential 图完成，这是 covisibility 图的一个稀疏子图。Essential 图优化在第三章 D 节，闭环检测的方法在第 7 章介绍。ORB-SLAM 所有图优化通过开源的 g2o 完成。

C. 点云，关键帧和选取方法

地图中点云储存方式如下

- (1) 在世界坐标系下的 3D 坐标
- (2) 视角向量 n_i ，是所有视角的平均单位向量。（是从观测关键帧的光心指向点云的方向）
- (3) ORB 特征的典型描述子 D_i 与该关键帧中被观测的点云的其他描述子相比，汉明距离最小。
- (4) 根据 ORB 特征的尺度不变性限制，可观测点云的最大和最小距离。

关键帧的存储如下

- (1) 相机的位姿 $T_{i\omega}$ ，是刚体变换的旋转矩阵，从世界坐标系到相机坐标系坐标点变换
- (2) 相机的内参，包括焦距和主点。
- (3) 关键帧中提取的所有 ORB 特征坐标，包括与地图点云关联的和没有关联的。如果给出了相机畸变模型，则 ORB 特征坐标经过矫正

地图点云和关键帧使用一个宽松的规则进行创建。之后使用一个严苛的机制负责剔除冗余的关键帧和匹配错误或无法跟踪的地图点云。这样可以保证在探索时地图有较好的扩展性，当复杂的构图环境下提高了系统的鲁棒性，即使反复探索有限大小的相同环境，也可以保证持续运行。相比 PTAM，ORB 重构的地图包含的点云少，很少出现异常值。

D. Covisibility 图和 Essential 图

关键帧之间的 Covisibility 信息在 ORB 系统中是非常有用的，实际上 covisibility 信息表示了一种无向加权图。每一个节点表示关键帧，如果存在连接两个关键帧的边，则表示两个关键帧共同观测到的相同的点云（不少于 15），权重 θ 表示观测到的相同点云的数量。

为了校正一个回路，运行位姿图优化。pose 图是把回路的误差分散表示在图中，为了剔除 covisibility 图中的边缘（covisibility 的边缘很稠密），采用 Essential 图（保留所有的关键帧节点而只保留很少的边缘）保存一个比较完整而稳固的网络得到精确的结果。系统从初始关键帧开始增量式的构建了一个 spanning 树，这个初始的关键帧连接了最少边的 covisibility 图的子图。当插入新的关键帧时，会把新插入的关键帧和与其有最多相同观测点的关键帧相连。当剔除关键帧时，则会根据这个关键帧的连接关系进行更新。Essential 图包含 spanning 树，covisibility 图高权重的边缘和闭环检测的边从而组成了一个稳定的相机网络。整个图优化后得到的结果精确度很好，完整的 BA 也无法提高其精度。

E. 词袋法位置识别

ORB-SLAM 利用基于 DBoW2 算法，利用位置识别模型的词袋完成闭环检测和重定位。视觉词是一个离散化的描述子空间，用描述子来表示视觉字典。这个字典根据从许多图像中提取的 ORB 特征的描述子离线生成。如果图像的数量足够多，字典可以用于不同的环境而获得很好的效果。整个系统增量式的构建一个数据库，数据库包括一个倒排索引用于储存字典中的每一个词，在数据库中可以查找到关键帧，可以高效的查询数据库。当关键帧被剔除时，数据库会进行更新。

由于关键帧之间会有重复的区域，所以在查询数据库中的关键帧时不会有一个得分特别高的关键帧。传统的 DBoW2 考虑这个问题时，会增加时间上和关键帧接近的图像的分值。然而这并没有包含在不同时间插入的具有相同位置的关键帧。ORB-SLAM 把在 covisibility 图与当前关键帧相连的关键帧进行分类。另外，ORB 的数据库返回关键帧匹配分值高于最好分数的 75% 的关键帧。

另外，当需要计算两组 ORB 特征的对应关系时，可以通过匹配词典树中属于相同节点的特征，提高搜索速度。

4 地图自动初始化

地图初始化的目的是为了计算两帧图像之间的相对位姿，通过三角原理初始化一组点云地图。这个功能应该独立于场景并且不需要人工干预可以选择一组合适的两视图，例如具有明显的视差。我们提出一种算法用于并行计算两种不同的几何模型：利用单应矩阵计算平面场景或利用基本矩阵计算非平面场景。我们利用探索式的算法选择场景模型并且通过一种特定的方法恢复对应场景模型相机的相对位姿。我们的算法只有在两视图的结构可靠，低视差已被检测到或是对二义性已知的前提下才会运行，避免构建一个有缺陷的地图。算法的步骤如下所示。

- (1) 查找初始对应关系：从当前帧 F_c 中提取 ORB 特征（只在合适的尺度上）并且在参考帧 F_r 中搜索匹配点 $x_c \leftrightarrow x_r$ ，如果匹配点的数目不

足，重置参考帧。

- (2) 并行计算两种模型：在并行线程上计算单应矩阵 H_{cr} 和基本矩阵 F_{cr}

$$x_c = H_{cr} x_r \quad x_c^T F_{cr} x_r = 0$$

通过 **RANSAC 算法**，使用归一化的直接线性变换和八点法分别计算矩阵两个。为了使两种模型的流程一致，预设**相同的迭代**次数，每次迭代使用相同的点云。8 个点云用于计算基本矩阵，8 个点云中的 4 个点云用于计算单应矩阵。本次迭代时，给模型 M (F 表示基本矩阵， H 表示单应矩阵)。

$$S_M = \sum_i (\rho_M(d_{cr}^2(x_c^i, x_r^i, M)) + \rho_M(d_{rc}^2(x_c^i, x_r^i, M)))$$

$$\rho_M(d^2) = \begin{cases} \Gamma - d^2 & \text{if } d^2 < T_M \\ 0 & \text{if } d^2 \geq T_M \end{cases}$$

d_{cr}^2 和 d_{rc}^2 是两帧间的对称变换误差。 T_M 是外点的剔除阈值，是基于 χ^2 分布（在 95% 置信区间上， $T_H = 5.99$ $T_F = 3.84$ ，假设测量误差是一个像素点的标准差）。 Γ 等于 T_H ，使两个模型在内点区域对于相同的 d 有同样的分数，使得流程保持一致。保存最高分对应的单应矩阵和基本矩阵，如果模型无法找到（没有足够的内点），从步骤 1 重新开始。

- (3) 模型选择：如果场景是**平面，近似平面或者低视差**，可以使用单应矩阵来描述。然而，这种情况下的基本矩阵是可以找到的，但是试图从基本矩阵中恢复运动时可能会得到错误的结果。应该选择**单应矩阵**作为重构的方法从平面中正确的初始化地图，或在检测到低视差场景时拒绝初始化。另一方面，有着足够时差的**非平面场景**只能够用**基本矩阵**进行描述，但是单应矩阵可以用来描述全部匹配点的子集，这些匹配点在一个平面或者低视差下观察。在这种情况下，应该选择使用基本矩阵。我们利用如下公式的计算结果选择矩阵。

$$R_H = \frac{S_H}{S_H + S_F}$$

如果 $R_H > 0.45$ **表示**捕获的场景是平面场景或者低视差情况，选择单应矩阵。否则选择基本矩阵。

- (4) 运动和运动结构重构：当模型选定时，可以恢复关联运动假说。在**单应矩阵**的情况下，使用 **Fau.** 和 **Lus.** 的方法**恢复 8 种运动假设**。利用 **Cheirality** 测试选择有效解。然而，当低视差情况时点云很容易在相机的前面或者后面产生错误的解导致测试失败。本文在相机前和低重投影误差时，采用**直接三角化 8 个解**并且检测是否在有有时差的条件下有一个解的大多数点云都可以被看到。如果没有一个明确的结果，不进行初始化并且返回步骤 1。这个方法可以消除解的歧义，使在低视差和 **twofold ambiguity** 情况下，初始化算法有很好的鲁棒性。这是提升本文初始化算法的鲁棒性的关键。

对于基本矩阵的情况，将基本矩阵通过**相机标定参数**矩阵 K 变换为

本征矩阵。 $E_{rc} = K^T F_{rc} K$ 。利用文献 2 中介绍的单值分解恢复 4 个运动假设。对于单应性，三角化四个解进行重构。

- (5) 捆集调整：最后，运行全 BA(附录 A)来优化初始重构。NewCollege r 机器人数据集是一个非常有挑战的室外环境初始化地图数据集。图 3 中展示了 PTAM、LSD 和 ORB 是如何初始化一个平面的所有点云。PTAM 和 LSD-SLAM 自动初始化构建了有缺陷的地图，PTAM 在手动选择后可以得到很好的初始化效果。ORB 可以在检测到足够时差后根据基础矩阵自动初始化地图。

5 特征跟踪

在本节中，介绍了特征跟踪线程处理相机每一帧图像的步骤。在几个步骤中提到的，相机的位姿优化由**运动 BA**完成。

A. ORB 特征提取

在 8 层图像金字塔中选用 1.2 的尺度因子提取 FAST 特征。图像的分辨率从 512x384 到 752x480，发现提取 1000 个角点比较合适。对于更高分辨率的 1241x376 的 KITTI 数据集，我们提取了 2000 个角点。为了保证均匀分布，我们将每一层分成网格，对每个网格**最少提取 5 个角点**。然后检测每个网格中的角点，如果角点数量不够时，调节检测阈值。如果一些单元格不包含角点(无纹理特征或低对比度)，每个网格包含角点的数量会随之变化。根据保留的 FAST 角点，计算方向和 ORB 描述子。ORB 描述子被用于系统中的所有特征匹配，这与 PTAM 基于图像块关联性的搜索方法不同。

B. 根据之前帧图像初始化位姿估计

如果最后一帧跟踪成功，采用一个**恒定速率运动模型**预测相机的位姿并且对**最后一帧中地图点云**进行引导搜索。对于最后一帧中的与地图点云关联的 ORG 特征，都会在当前帧中它曾经出现的位置附近找到匹配点。如果没有找到足够的匹配点（比如运动模型失效），则在最后一帧图像地图点云的附近较大范围内进行搜索，根据找到的对应关系进行位姿优化。最后获得一组 3D-2D 的对应关系，可以计算相机位姿，解决 RANSAC 中的 PnP 问题。

C. 通过全局重定位初始化位姿估计

如果跟踪丢失，我们将当前帧变换为词袋然后对比候选关键帧进行重定位数据库查询，完成全局重定位，如果有多个相似场景，会有多个候选帧。我们计算每个**候选帧**中与地图点云相关联的 ORB 特征和**当前关键帧**的**匹配一致**。获得每个候选关键帧的一组 2D-3D 的对应关系，然后对每一候选关键帧进行**RANSAC 迭代运算，使用 PnP**算法尝试确定相机的位姿。我们优化位姿并且引导式的搜索候选关键帧中更多的地图点云的匹配点。最后，再次优化相机位姿，如果有足够的内点则继续执行跟踪线程。

D. 跟踪局部地图

一旦完成了相机位姿估计和一组初始化的特征点，可以把地图投影到帧图像并且寻找更多对应的地图点云。为了限制大地图的复杂性，仅仅投影局部地

图。这个局部地图包含 **关键帧** K_1 ， K_1 和当前帧有共同地图点云，以及在 **covisibility** 图上和 K_1 临近的一组 **关键帧** K_2 。局部地图有一个属于 K_1 的参考帧 K_{ref} ， K_{ref} 和当前帧有最多的相同点云。对于 K_1 和 K_2 中的所有点云，按照如下步骤进行搜索

- (1) 计算地图点云在当前帧的投影 x ，如果投影点位于图像外部则剔除。
- (2) 计算当前视图方向向量与点云图平均视图方向之间的夹角，如果夹角大于 60° 则剔除。
- (3) 计算地图点云到相机光心的距离 d ，如果不在地图额尺度不变区域即 $d \notin [d_{min}, d_{max}]$ 则剔除。
- (4) 计算当前帧图像的尺度因子， d/d_{min}
- (5) 比较地图点云的典型描述子 D 和当前帧图像中未匹配的 ORB 特征点描述子，在估计的尺度因子范围内，点云投影的点集是否在 x 附近，关联地图点云获取最佳匹配。

当前帧中所有特征点与地图点云**配对后**，相机位姿使用 **g2o** 完成**最终优化**。

E. 新关键帧的取舍

最后一步决定是否将**当前帧变成新的关键帧**。在局部地图构建中有一个机制用于剔除冗余的关键帧，我们尽可能快的插入关键帧，从而保证特征跟踪线程对于相机的运动，尤其是旋转运动具有更好的鲁棒性。当需要插入一帧新的关键帧时，需要满足如下的条件。

- (1) 距离上一次重定位的位置，必须**超过 20 帧**图像。
- (2) 局部地图构建是**空闲**状态，或者距离**上一次插入关键帧**已经超过了 20 帧图像
- (3) 当前帧最少跟踪了 **50** 个点。
- (4) 当前帧跟踪的点云少于参考帧的 **90%**

不同于 PTAM 使用与其他关键帧的距离作为判据，本文提出了最少视觉变化的概念(条件 (4))。条件 1 保证了良好的重定位，条件 3 保证了良好的特征跟踪情况。如果在局部地图构建进行时插入关键帧，将会发送一个信号停止局部 BA 来保证尽快的处理新的关键帧。

6 局部地图构建

在本节中，介绍了通过每一个新的关键帧 K_i 构建局部地图的步骤。

A. 关键帧插入

首先，更新 **covisibility** 图，为新关键帧 K_i **增加一个新的节点**并根据与其他关键帧共享的地图点云情况更新连接边缘。然后，在 **spanning** 树中将当前关键

帧 K_i 连接到与 K_i 具有最多相同点云的关键帧。计算表示关键帧的**词袋**，可以在三角化新点云时有助于关联数据。

B. 地图点云筛选

为了保证地图点云的**可跟踪性**和正确的**三角化**，地图点云需要在**点云创建后的前三个关键帧期间**通过一个严格的测试，避免错误的数据关联引起的错误。地图中的一个点云必须满足以下两个条件。

- (1) 在预测可以观测到这些点云的关键帧中，追踪线程必须在超过 25% 的关键帧中可以找到这些点云。
- (2) 如果在地图点云创建期后，已经间隔了超过一个关键帧。则这个点云需要在不少于三个关键帧中可以被观测。

如果一个地图点云通过了测试，只有在可以观测到它的关键帧**小于三个**时才可以被剔除。这种情况一般发生在关键帧被剔除和局部捆集调整删除无效的观测点时。这保证了我们地图只包含很少的奇异点。

C. 地图点云创建

通过三角化不同关键帧的 ORB 特征点可以构建新的地图云点。PTAM 用最邻近的关键帧三角化这些点，这些帧的视差非常小；我们用相邻的**N 个关键帧来三角化这些点**，这些帧具有很多相同的特征点。我们需要约束关键帧的数量，由于地图间高度关联，计算代价会很大。通过**三角化**在 **covisibility** 图中与当前被处理**关键帧 K_i** 相连的**关键帧集合 K_c** 中的特征点 **ORB** 来添加新的地图点云。

对于在新关键帧 K_i 中**未匹配的 ORB** 特征点，搜索别的关键帧中**未匹配的点进行匹配**并且剔除掉不满足极对约束的点。这个匹配方法将在 Section3-E 中介绍。一对 ORB 特征点三角化后，如果要添加为地图点云，要保证在两个相机视图下的**深度都是正值**，检测**视差大小**，**重投影误差**和**尺度一致性**。开始时，一个地图点云只可以在两个关键帧中被观测，但是可以被其他关键帧匹配。因此他可以被映射到其他相连的关键帧给**跟踪算法使用**，点云对应关系的搜索细节见第五章 D 部分。

D. 局部捆集调整

局部捆集调整优化当**前处理关键帧 K_i** ，在 **covisibility** 图中与 K_i **相连的所有关键帧**和所有被这些**关键帧观测到的地图点云**。对于那些可以观测到地图点云但是没有与 K_i 相连的关键帧仍旧在优化线程中，但是保持固定不变。为了解决非线性问题，使用 g2o 中的 L-M 非线性最小二乘，Huber 估计器，完成优化后重新计算每一个点云的视角向量 \mathbf{n} ，尺度不变性距离 d_{\min}, d_{\max} 。在优化过程的中段和后端被标记为外点的观测点云将被抛弃。优化的具体细节请看附录。

E. 局部关键帧筛选

为了保证重建地图的紧凑性，局部地图会检测冗余关键帧并且剔除它。由于 BA 的复杂度会随着**关键帧的数量增加而增加**，这可以降低 BA 的复杂度。另外只要相同场景内容不发生变化的情况下，可以长时间运行防止关键帧数量超

限。当 K_c 中的某关键帧 90 的地图点云被不少于三帧其他关键帧在相同或更好的尺度上观测到时，这一帧关键帧被剔除。尺度条件的限制可以保证关键帧中的地图点云有更好的测量精度。

7 闭环检测

闭环检测线程使用局部地图构建时的最后一帧关键帧 K_i ，进行回路的检测和闭合。

A. 候选回路检测

首先，计算 K_i 关键帧的图像词与它在 `covisibility` 图中临近关键帧（ $\theta_{\min} = 30$ ）的相似性，保存最低分数 s_{\min} 。然后，查询识别数据库剔除分数低于 s_{\min} 的关键帧。这和 DBoW2 中归一化分数的操作类似提高了鲁棒性，不同的是 DBoW2 使用以前的图像，而本文则是利用 `covisibility` 图的信息。另外，所有在 `covisibility` 图中与关键帧 K_i 直接相连的关键帧都会在结果中被剔除。为了获得候选回环，需要连续检测三个候选回环（在 `covisibility` 图中连接的关键帧）的一致性。如果有多个场景与关键帧 K_i 相似，则会有多个候选回环产生。

B. 计算相似变换

在单目 SLAM 中，有七个自由度，分别是三轴位移、三轴转动和尺度因子，这七个自由度会影响地图漂移。因此，为了完成闭环，我们需要计算从当前关键帧 K_i 到回路关键帧 K_l 的相似变换，获得闭环累计误差。相似变换也可以作为闭环的几何验证。

首先计算当前关键帧 K_i 和闭环候选关键帧中与地图点云关联的 ORB 特征一致性，在第三章 E 节中介绍了计算方法。此时，对于每个候选闭环，我们有一个 3D-3D 的对应关系。对每一个候选使用 RANSAC 迭代，使用 Horn 的方法获得相似变换。如果通过足够的内点获得了相似变换 S_{il} ，可以进行优化并且搜索更多的对应关系。如果 S_{il} 有足够多的内点，可以再次优化直到 K_l 作为闭环关键帧。

C. 回环融合

回环矫正的第一步是融合重复的地图点云并且在与闭环检测相关的 `covisibility` 图中插入新的边。首先，当前关键帧的位姿 $T_{i\omega}$ 通过相似变换 S_{il} 进行矫正，并且对关键帧 K_l 临近的所有关键帧都采用这样的方法进行矫正，使回路的两端可以对齐。闭环关键帧可以观测到的所有地图点云和它临近的点云都投影到关键帧 K_i ，在投影区域附近的一个狭小区域搜索它的临近点和匹配点，具体方法见第 5 章 D。将所有匹配的地图点云和在计算 S_{il} 时的内点进行融合。所有参与融合的关键帧在 `covisibility` 图中会更新边缘，创建附加的闭环回路边缘。

D. Essential 图优化

为了有效的进行回路闭环，在 **Essential** 图上运行一个位姿图优化，**将回路闭合误差分散到图中**，具体介绍在第三章 D 节。这个优化通过相似变换矫正尺度漂移。误差项和目标函数见附录。完成优化之后，每一个地图点云都随着观测它的关键帧的校正进行变换。

8 实验

我们对 ORBSLAM 做了大量的实验测试系统性能。使用 NewCollege 大场景机器人数据集评估系统的**整体性能**；使用 TUM 的 16 个室内手持 RGB-D 数据集评估系统的**定位精度，重定位和持续运行能力**；使用 KITTI 的室外汽车数据集评估系统的**大场景实时性，定位精度和位姿图优化**的效率。

我们系统根据数据集采集图像的帧率实时进行运算和处理。我们的实验平台的电脑配置是 Core i7-4700MQ，8GB RAM。ORB 主要有三个线程，它们与 ROS 和操作系统的其他任务并行执行，这会对结果引起一些不可预测的结果，因此在一些实验中，我们展示多次运行后的中值。

A. NewCollege 数据集

NewCollege 数据集是机器人穿过长约 2.2 公里的校园和临近的公园时记录的一个的图像数据集。图像采集使用双目摄像头，分辨率是 512x382，帧率是 20fps。整个数据集包括多个闭环和快速旋转，这对于单目 SLAM 来说是一个相当大的挑战。就我们所知，没有单目系统可完整的处理完整数据集。文献[7]中，尽管可以完成闭环检测和大场景下的运行，但是只完成了整个数据集很小的一部分。

作为一个闭环检测的例子，在图 4 展示了中通过**内点相似变换**进行回路检测。图 5 中展示了闭环检测前后的重构效果。红色表示局部地图在闭环检测之后回路两侧延伸的情况。按照数据集采样的帧率处理完后整个重构地图如图 6 所示。右侧的大回路并没有完美闭合，因为它是从反方向穿过，位置识别没能找到闭合回环。

我们统计了实验中每个线程花费的时间。表 1 显示了追踪和局部地图构建线程的运行时间。跟踪线程运行速度与帧速率一致，25-30Hz，跟踪局部地图任务耗时最多。如果有需要，可以通过限制局部地图包括的关键帧数量来提高运行速度。局部地图线程，局部 BA 任务耗时最久，当机器人在探索新地图或是经过重建好的区域时局部 BA 的运行时间是不同得，这是因为当在探索新地图时，局部 BA 会被新插入的关键帧中断，具体细节见第五章 E 部分。如果没有新的关键帧需要插入，局部 BA 会运行大量已经设置好的迭代。

表二显示了 6 个闭环检测的实验结果。从图中可以看出，闭环检测的关键帧数量是如何次线性增加，这是因为只需要比较具有相同图像词的子集从而有较高的查询效率，由此可以看出使用词袋法的好处。**Essential** 图中包含的点云数仅仅是关键帧数的五倍，由此可见这个是一个稀疏图。

B. TUM RGB-D 数据集定位精度

TUM RGB-D 数据集通过外部运动捕捉系统获得了精确的位置信息，是一种可以用来评估定位精度的优秀的数据集。我们剔除了那些包含强烈旋转，缺

少纹理特征和没有运动的不适合单纯的单目 SLAM 的数据集。

为了方便比较，我们还在 TUM 数据集上运行了当前最先进的直接法 LSD-SLAM 和 PTAM。我们也与一些 RGBD-SLAM 数据集网站上提供的路径轨迹进行了比较。为了根据真实值比较 ORB-SLAM, LSD-SLAM 和 PTAM 的定位精度，我们在尺度因子未知的情况下，使用相似变换对齐关键帧序列测量轨迹的绝对误差。在对比 RGBD-SLAM 时，使用刚体坐标转换对齐了轨迹，采用相似的方法检测尺度信息是否恢复良好。LSD-SLAM 采用随机深度值并在一定时间直内使其收敛来初始化，因此我们提出前 10 帧。对于 PTAM，我们从认为好的初始化环境中手动选择两帧图像进行初始化。表三显示了 16 测试场景运行 5 次的定位精度的中位数。

从表中可以看出，除了 `fr3_nostructure_texture_far(fr3_nstr_tex_far)` 场景，ORB-SLAM 可以处理数据集中的所有场景。`fr3_nstr_tex_far` 是一个平面场景图像序列，相机轨迹有两种可能的解释。我们的初始化算法检测到了视图的歧义为了安全拒绝进行初始化。PTAM 初始化时这个场景图像序列时，有时会选择正确的解，有时会选择错误的解，错误的解可能会导致不能接受的结果。我们没有注意到 LSD-SLAM 的两种不同的重构方法，但在这个场景图像序列中，它的误差非常大。在 TUM 数据集的其他场景图像序列中，PTAM 与 LSD-SLAM 的鲁棒性较差，PTAM 在 8 个场景中发生了跟踪丢失，LSD-SLAM 则在三个场景跟踪丢失。

就精度而言，在开环路径时 ORB-SLAM 和 PTAM 精度相近。当 ORB-SLAM 在 `fr3_nstr_tex_near` 场景序列中检测到大的闭环路径时，会获得更高的精度。令人意外的结果是，PTAM 和 ORB-SLAM 的定位精度超过 LSD-SLAM 和 RGBD-SLAM。可能是因为他们没有使用传感器测量的数据进行位姿图的优化。在 ORB-SLAM 中，使用传感器测量数据利用 BA 和联合优化进行了相机位姿和地图优化，这个算法是解决 SFM 问题的黄金算法。之后我们将在第 9 章 B 节中讨论这个问题。另外，LSD-SLAM 相比 ORB-SLAM 对于动态目标鲁棒性较差。

我们注意到 RGBD-SLAM 在 `fr2` 图像序列中存在尺度偏差。利用 7 自由度进行轨迹对齐可以明显减小误差。最后，在 LSD-SLAM 的论文中提到在 `f2_xyz` 场景图像序列中 PTAM 的精度比 LSD-SLAM 精度低，但是论文中没有详细描述实验过程，我们无法复现这个实验结果。

C. TUM RGB-D 数据集重定位测试

在 TUM RGB-D 数据集设计两个重定位实验。在第一个实验中，我们通过 `fr2_xyz` 数据集前 30s 的图像流重构地图，然后对每个连续帧进行全局重定位，评估重定位后的位姿精度。对 PTAM 进行相同实验进行比较。图 7 表示地图初始化的关键帧，重定位位姿关键帧和这些关键帧对应的真实位置。从图中可以看出，由于 PTAM 采用重定位方法不具有视角不变性，因而只能在关键帧附近进行重定位。表 4 表示 PTAM 和 ORB-SLAM 相对真实值的查全率和的误差。ORB-SLAM 重定位精确重定位的帧数是 PTAM 的两倍。在第二个实验中，使用场景图像序列 `fr3_sitting_xyz` 初始化地图，从 `fr3_sitting_xyz` 场景图形序列中重定位所有关键帧。由于人在场景中移动会产生很大的阻塞，这是一个相当具有挑战的实验。ORB-SLAM 重定位识别了所有帧中 78% 的图像，而 PTAM 没有完成重定位，实验结果见表 4。Fig.8 展示了以上实验中 ORB-SLAM 对较为困难的重定位场景的运行效果。

D. TUM RGB-D 数据集持续运行试验。

之前的重定位实验证明，ORB-SLAM 可以在不同视角下进行重定位并且在相对缓和的动态场景下有较好的鲁棒性。这一特性结合关键帧剔除程序，可以使 ORB-SLAM 在不同视角和动态变化的相同场景中持续运行。

在全静态场景中即使相机从不同的视角观测场景，ORB-SLAM 能够限制关键帧数量在一个合理的范围内。在一个特定场景中进行实验，这个场景在 93s 内都从不同的视角观察同一张桌子。比较 ORB-SLAM 和 PTAM 生成地图对应的关键帧数量，如图 9 所示。从图中可以看出，PTAM 总是不断的插入关键帧而 ORB-SLAM 则使关键帧数量最终收敛。

在静态环境下持续运行是所有 SLAM 系统的基本要求，除此之外更为关注系统在动态环境下的运行状态。我们在 fr3 动态场景下分析 ORB-SLAM 的系统特性，这些场景中相机都是对着同一张桌子但是运动轨迹不同，并且这些场景中人和椅子会不断移动。图 10(a)表示地图对应关键帧数量的变化，图 10 (b) 是每一帧关键帧创建和消亡的过程，表示每一帧关键帧在地图构建时存在的时间。在运行到前两个场景图像序列时可以发现，在场景中所有视图完成后地图的大小在不断增加。在图 10 (b) 中可以发现，前两个场景图像序列中有许多关键帧在整个实验中一直存在。在运行到场景图像序列 `sitting_rpy` 和 `walking_xyz` 时，由于地图被解释的足够完善，所以地图大小基本不变。相反，在运行到最后两个场景图像序列时，由于场景有动态变化导致出现了一些新的场景，因而插入了更多的关键帧。图 10 (c) 直方图表示了关键帧从创建后在场景图像序列中存在时间的分布。从图中可以看出，大部分关键帧在创建不就被删除了，只有一小部分关键帧在创建后一直存在到整个实验结束。这表明，ORB-SLAM 系统有一个宽容的关键帧选择标准，这保证系统对于探索时突然的相机运动足够敏感。另一方面，ORB-SLAM 系统最终只会保留很小一部分具有代表性的关键帧。

整个持续运行实验表明，ORB-SLAM 的地图的大小与场景内容有关，而与运行时间长短无关。另外，ORB-SLAM 可以存储场景的动态变化，这有助于理解场景。

E. KITTI 数据集下的大尺度场景和回环检测实验

KITTI 数据集有 11 个场景图像序列，这些场景图像序列是驾车围绕住宅区采集到的，数据集通过 GPS 和激光雷达获得了精确的标准参考值。整个场景图像序列有快速的旋转，场景中有很多树叶使得数据关联十分困难，并且由于车速很快，图像帧率只有 10fps，这对于单目来说是非常大的挑战。ORB-SLAM 可以根据 KITTI 采集的帧率处理除了场景图像序列 01 之外的所有场景图像序列。场景图像序列 01 是在高速公路上采集的，缺少足够多的可跟踪特征。ORB-SLAM 可以对场景图像序列 00,02,05,06,07 和 09 中的回路完成检测和闭环。场景图像序列 09 中的回路只能通过图像序列的最后几帧进行检测，ORB-SLAM 不是每次都可以检测到。

图 11 和 12 表示 ORB-SLAM 生成的路径轨迹与数据集标准值得定性比较。在 TUM RGB-D 数据集中，我们使用相似变换对齐了 ORB-SLAM 生成的轨迹和数据集标准值轨迹。根据图 11 与图 12 中的场景序列 00,05,06,07 和 08 的结果，与文献 25 的单目 SLAM 运行结果进行定性比较。除了场景图像序列 08 外，ORB-SLAM 有更高的精度。场景图像序列 08 似乎没有进行闭环和尺度漂移校准。

表 5 展示了所有场景图像序列执行 5 次后的轨迹 RMSE 误差的中位数。我们也根据地图的尺寸提供了环境误差，结果表明我们系统精度很高。我们的轨迹误差相对于地图尺寸大约在 1% 左右，最小误差是场景 3 下的 0.3%，最大误差是场景 8 下的 5%。场景 8 中没有回路，并且没有矫正漂移。这也表明系统需要闭环检测来完成高精度重建。

在这个实验中，我们还验证了在增加 20 次迭代的全 BA 对地图重构提升效果。可以发现，一些迭代全 BA 可以轻微提高有回路的轨迹精度。但是对于开环轨迹，会造成更大的误差，这意味着我们的系统的输出已经足够精确。在一些场景中，如果需要特别高的精度，我们的算法可以提供一系列匹配点而建立一个强健的相机网络和初始化估计，减少全 BA 的迭代次数。

最后，我们展示了闭环检测方法和 Essential 图优化的边缘中 θ_{\min} 的效果。我们选择场景图像序列 09(长时间的数据集，在最后有闭环检测)。在相同的运行关键下，我们评估了不同的闭环检测策略。在表 6 中，我们展示了不同方法闭环检测下的轨迹误差和运行时间。闭环策略包括，无闭环、全 BA（不同的迭代次数），位姿图优化 EG（10 次迭代，不同的 θ_{\min} 值）和 BA+EG。图 13 展示了不同算法的输出轨迹，没有闭环时误差很大，即使全 BA 迭代 100 次后，误差依旧很大。另一方面，Essential 图优化精度很高且收敛速度很快，并且可以看出不同的 θ_{\min} 值对精度的影响不是特别明显，但是减少边缘数量可以明显降低运行时间。在完成 EG 后增加一个 BA 可以轻微提高精度，但是增加了运行时间。

9 结果与讨论

A. 结论

本文介绍了一个新的单目 SLAM 系统，并且详细描述了它的系统流程和在各个数据集上详尽的测试结果。实验表明，我们的系统可以用于室内和室外，并且无论是汽车、机器人或是手持摄像头运动都有较高的精度。在室内小场景范围内我们的系统精度可以达到 1cm，对于大场景系统精度则是几个厘米。

当前，PTAM 被认为是最好最精确的实时单目 SLAM 算法。BA 作为解决离线 SFM 问题的经典算法而出现在 PTAM 的后端这绝不是巧合。PTAM 最大的成功之处在于，在机器人 SLAM 领域引入了新的方法并且证明了它的实时性能。本文的主要贡献是完善了 PTAM 中一些难以处理的环境，使 PTAM 可以适应多种环境。出于这个目的，我们重新设计了一个单目 SLAM 系统并且加入了一些新的算法，并且结合过去一些优秀的成果，比如像文献 5 中的闭环检测算法，文献 6, 7 中的回路闭环和 covisibility 图算法，文献 37 中的基于 g2o 框架的优化算法，文献 9 中的 ORB 特征算法。据我们所知，没有其他的单目 SLAM 算法可以在如此多的场景中运行并且达到这样的精度。因而，ORB-SLAM 是当前最为可靠和完整的单目 SLAM 系统。我们的关键帧生成和筛选算法可以每隔几帧就添加一个关键帧，并且最终剔除掉所有冗余的关键帧。这使我们的地图易于扩展并且可以应用于带有旋转和快速运动的复杂环境下的路径探索。在同一个环境下，如果场景没有发生改变，地图大小就不会增加，从而可以保存不同时间段内的场景特征，方便对一个较长时间段内的地图进行提取和分析。

最后，我们展示了 ORB 特征的视角不变性，因而具有很强的位置识别能力。并且，ORB 特征可以快速提取与匹配(无需 GPU 加速)，可以满足实时跟踪与重

构的要求。

B. 基于特征的稀疏地图 vs 基于直接法的稠密地图

最近，一些实时的单目 SLAM 算法比如 LSD-SLAM 和 DTAM 可以通过图像像素亮度直接优化相机位姿，完成半稠密或者稠密的地图重构。这种直接法不需要特征提取，因而避免了相应的运算与处理。他们对于模糊的，具有较少纹理特征和高频纹理特征的环境具有较好的鲁棒性。他们稠密的地图重构，相比于稀疏的地图点云在其他应用方面具有更大的优势。

然而，直接法自身也存在局限性。首先，直接法采用了一种表面反射模型，这与实际反射模型对比时本身就存在偏差。并且光亮度一致性限制了匹配的基线，这比其他特征要窄的多。这对重构精度影响较大，需要较宽的观测基线消除深度的不确定性。如果没有正确的模型，直接法会受到诸如卷帘快门，自动增益和自动曝光的影响（比如在 TUM RGB-D 数据集中）。最后，由于直接法计算两较大，在地图重构时一般是增量式的增加（DTAM）或者是舍弃所有的传感器数据，只使用位姿图进行优化（LSD-SLAM）。

于此相反，基于特征的方法具有较好的视角不变性和亮度不变性，可以在一个更宽的基线上匹配特征。BA 通过传感器测量数据，同时优化相机位姿和地图点云。在环境结构和运动估计中，文献 46 已经指出基于特征的方法优于直接法。在本文的研究中，我们的实验结果表明基于特征的 SLAM 具有更高的精度。我们认为，单目 SLAM 的未来应该对两种方法进行融合。

C. 未来的工作

我们的系统可以通过跟踪无穷远处的点进行融合来提高精度。那些没有足够视差而没能放入地图的点，对于分析相机的旋转很有帮助。

另外一个方向是将稀疏点云重建变为稠密或者更有价值的重构地图。由于我们关键帧的选择机制，关键帧包含了高精度的位姿信息，环境情况的概述以及丰富的 covisibility 信息。因此，ORB-SLAM 稀疏点云图是一个优秀的地图初始化估计和框架，在这基础上可以进行稠密和高精度的场景地图重构，可以参考文献 47