

# ORB-SLAM2

---

## ORB-SLAM2

Introduction

Bundle Adjustment

PTAM

ORB-SLAM

Related Work

Place Recognition

Map Initialization

Monocular SLAM

System Overview

Feature Choice

Three Threads: Tracking, Local Mapping, Loop Closing

Map points, Keyframes and their Selection

Covisibility Graph and Essential Graph

BoW Place Recognition

Automatic Map Initialization

Tracking

Local Mapping

Loop Closing

Appendix

Bundle Adjustment

## Introduction

---

## Bundle Adjustment

- 使用BA来纠正camera pose的误差，提供一个初始解对超定方程求解（Levenberg）
- 实时的SLAM算法需要给BA提供以下几点：
  - 在一个选定的帧子集中图像对应的features（即Map points）
  - 随着keyframe数量增加，对关keyframe的选择应当避免不必要的冗余（删除掉信息冗余的keyframe）
  - 给keyframe和point提供强大的网络结构（可靠），这样才能产生比较准确的结果，即对keyframe观察到points给出一个比较好的扩展集（即以此

keyframe做延伸的集合），具有比较重要的视差、多个loop closure被检测出来了（这样得到的地图是比较稳定的）

- 对于keyframe和point有一个初始解
- 用来集中处理scalability优化的local map（单目相机无法处理尺度问题）
- 能够实时处理闭环的全局优化

## PTAM

- PTAM优点：

- limited to small scale operation??（在小尺度下这些操作不太好）
- simple but effective methods for keyframe selection
- feature matching
- point triangulation
- camera localization for every frame
- relocalization after tracking failure

- PTAM缺点：

- lack of loop closing and adequate handling of occlusion
- low invariance to viewport of the relocalization
- need of human intervention for map bootstrapping

## ORB-SLAM

- ORB-SLAM：

- 使用ORB作为唯一特征点，不用GPU优化，提供对于viewport和illumination好的不变性
- 在大环境中实时。使用covisibility graph，在tracking和mapping时都只在local covisibility graph中进行操作，而不必要全局
- 使用pose graph（Essential Graph）（就是双目视觉中的基础矩阵，只考虑相机外参而不必要考虑相机内参）。它是由系统（？）、回环链、covisibility graph中的强边维护。
- 实时相机重定位（使用的对于viewpoint有较强不变性的特征），这样跟踪丢失之后也可以找回来，而且这样加强了map的重用。
- 建立在model selection基础上的一种新的自动化且鲁棒的初始化过程。它能够创建平面或者非平面图的初始地图
- 大量生成，严格筛选，这样得到一个最合适的map point和keyframe的选择。这样可以增强跟踪的鲁棒性，丢失了冗余的keyframe因此可以增长生命长度（指这个过程可以增长）。（PTAM只是一直严格筛选而不会删除，因

此在定位的时候容易丢失（选的少））

- 这篇文章在我们以前工作的基础上增加了：
  - 初始化方法
  - Essential Graph
  - 很好的融合了所有的方法

## Related Work

---

### Place Recognition

见ORB-SLAM

### Map Initialization

Monocular需要获得一个初始的地图，因为单张图片是无法获取深度的。一种方法是使用一种已知的结构来初始化跟踪（如使用A4纸进行定标，获取深度、尺度信息）。在半稠密图上，使用的方法是对像素都使用高方差的随机值来初始化深度，然后希望通过之后的操作进行还原。

从两张图中初始化，有两种方法，一种是使用Faugeras et.al的方法，将当前的屏幕当做平面，然后使用homography恢复出相关相机姿势。还有一种是计算essential matrix（五点法）。但是这两种方法在视差较小和有两种潜在可能的方法的时候都不行（if all points of a planar scene are closer to one of the camera centers）。当然，另一方面如果一个看到了两张非平面的图而且有视差，那么久可以使用八点法恢复出一个唯一的fundamental matrix，并且能够恢复出相机的姿势。

我们在基于模型选择的方法上设计了一种新的自动的方法，来为平面使用homography，为非平面恢复出fundamental matrix（初始的时候，摄像头可能照射到一个完全的平面，也可能是一个非平面）。Torr et al提出了一种统计方法来进行模型选择。我们基于类似的原因提出了一种将选择fundamental matrix的风险考虑在内的启发式的初始化算法来靠近退化的情况，这样就会使用homography（即提出一个评估函数，计算这个场景的平整情况，不平整时可以计算fundamental matrix，否则视差太小就计算homography进行初始化）。

### Monocular SLAM

- 滤波器方法：

每一次更新信息，整个滤波器都必须重新做一遍来对地图特征和相机姿势进

行评估，计算累计误差，这样十分浪费时间。

- **keyframe-base approach:**

使用BA进行优化，建图不与帧率紧密相关。（哪怕帧率增加，关键帧的个数也不会增加，因此建图所使用的数据并不会增加）

- **PTAM:**

使用patch相关性类匹配FAST角点（映射到Map points），这样他就只能用来做跟踪，而无法识别场景。也因此PATM无法检测大的环，重定位也是基于与低分辨率的keyframe省略图的相关性来做的。

- **Strasdat提出一种用于大尺度modocular SLAM。**前端使用GPU实现光流，FAST特征匹配以及motion-only BA。后端使用sliding-window BA。loop closures使用基于similarity constraints的pose graph optimization。我们从中析取了loop closing的做法（使用 Essential Graph完成）。

- **使用double window的方法，对inner window持续做BA优化，并对outter window的一定尺寸做pose graph的BA优化。**但是只有当outter window的尺寸大到一定程度时，loop closure才能够被有效的探测到。我们基于covisibility建立了local map，并在covisibility graph的基础上建立了pose graph，但是关于前端和后端的使用我们重新进行了设计（？？为啥我感觉是一样的）。

- **CD-SLAM:**

包含loop closing、relocalization、large scale operation、efforts to work on dynamic environment（！！注意这个）的完整的SLAM系统。

- **所有的视觉SLAM都展示出：在所有的点和所有的keyframe上面做BA事不好的。**Strasdat指出花销最有效的方法是：保持尽量多的点，但是只保持不冗余的keyframes（ORB-SLAM的做法就是尽量快的选取，但是碰到冗余时就将那些冗余的去掉）

## System Overview

---

### Feature Choice

选择ORB特征（有方向，多尺度FAST角点，256位描述子），快并且对于viewport有不变形。（详情在ORB特征的paper中有介绍）

### Three Threads: Tracking, Local Mapping, Loop Closing

三个线程并行，分别做跟踪，局部建图，回环检测

- 跟踪：
  - 先使用前一个keyframe做初始的特征匹配，并且使用motion-only BA进行优化。
  - 当跟踪丢掉了，place recognition模块会用来做一次全局重定位。
  - 当找到了一个初始解之后，皆可以获得一个局部的可视图了（从covisibility graph中获取到一个local visible map）
  - 通过匹配进行reprojection，并且使用这些reprojection进行camera pose的优化
  - 优化完毕之后决定是否应当插入一个新的keyframe
- 局部建图：
  - local mapping需要处理新的keyframe并且使用local BA，这样就可以根据它周围的camera pose来对它的pose做优化了。
  - 对于在新的keyframe中没找到匹配的ORB特征，在covisibility graph中找到相连的keyframe，并将其三角化。
  - local mapping还需要挑选出冗余的keyframe。
- 回环处理：
  - 对于每一个新的keyframe都尝试搜索一个新的环。（DBoW+geometry）
  - 检测到环后都对于loop中积累的误差都计算出一个相似转换。
  - 将回环的两端对齐并接上，将重合的点融合（Map point）。
  - 根据全局的连续性，基于相似转换的约束对pose graph做优化。（基于Essential Graph，一个covisibility graph的稀疏子图）

使用g2o提供的levenberg-marquadt算法来做所有需要的优化。

## Map points, Keyframes and their Selection

- Map point  $p_i$ 存有一下的信息：
  - 它在世界坐标系中的三维坐标 $X_{\{x\}}\{i\}$
  - 观察方向 $n_{\{i\}}$ ，是多个观察方向的平均向量。（每个keyframe的光心和这个点组成的方向）
  - 一个与ORB相关的描述子 $D_{\{i\}}$ ，到所有和该map point相关联的keyframe中对应的ORB描述子的hamming距离最短。（用一个描述子D来综合所有观察到这个点的特征描述，这样可以增加viewport invariance）
  - 能够观察到它的最大和最小的距离 $d_{\{max\}}$ 和 $d_{\{min\}}$ ，来描述该ORB特征尺度不变形的极限。

- keyframe  $K_i$  存有以下信息：
  - 相机位姿  $T$ ，这样可以将点从世界坐标系投影到相机坐标系。
  - 相机内参，包括焦距和光心。
  - 所有从帧内抽取的ORB特征，并且将其中某些和Map point关联起来。
- 先正常的将keyframe和map point都建起来，之后再用比较严格的手段将冗余的和错配或者没有跟踪到的map point去掉，这样地图能够在巡回的时候（即拿着摄像头四处走的时候）比较好的扩展地图，这样能够加强在比较艰难的条件下跟踪的鲁棒性，并且当它在同一个环境中反复走时，地图的尺寸不会过度增大（因为冗余的都被去掉了）。

## Covisibility Graph and Essential Graph

- Covisibility是根据两个keyframe之间共视的map point建立的，当共视点数 $\theta$ 超过一个threshold的时候就在两个keyframe之间建边，表征一个局部，并且边的强度即共视的点的个数。
- 在纠正回环的时候对pose graph进行优化，将回环的误差分配到图上。（应该就是BA吧？）为了避免将covisibility graph中的所有边都包含进来，建立一个Essential Graph，它保持更少的边，但是保证整个网络仍然很强壮，使得用这个Essential Graph仍然能够得到比较准确的结果。
- 系统从初始的keyframe中不断维护一颗生成树，每进来一个新的keyframe，直接将它使用最强的那条边连上树即可。当一个keyframe被擦除时，更新与它相关的连接。
- Essential Graph 由生成树+强边（ $\theta$ 不小于100）+loop closure边构成，这样可以对所获取的keyframe形成一个强的网络。（从试验中观察可以得到，使用pose graph优化之后结果已经十分准确了，哪怕最后再做一个完全的BA都没什么提升），当然 $\theta_{min}$ 的选影响很大。

## BoW Place Recognition

详细的看ORB-SLAM

- 离线建立一个词典，分级的分类树，含有正向索引和反向索引用来快速查找具有相同ORB feature的keyframe。
- 由于两个keyframe存在着视野重合，因此获得匹配高分的往往不止一个keyframe，因此是选择以一个group（在covisibility graph中相连的keyframe）的综合分作为匹配分，然后以这个group中分最高的作为candidate，会返回多个candidates。
- 在计算特征点响应的同时还做orientation consistency测试，并且抛弃掉outlier，

这样可以增加匹配的正确性。

## Automatic Map Initialization

地图初始化其目的是为了计算出两帧之间的位姿关系来三角化特征点，得到map points的初始集合。这个方法应当与scene（实在想不出怎样翻译才是准确的）无关（planar or general），也不应当需要人的干预（如选择具有比较好的视差的两个初始帧）。

我们初始时对两帧求homography和fundamental matrix两个解（一个用于planar的特殊情况，一个用于普通的情况），然后利用启发式的评估函数来判决哪个解更好，选择一个模型后我们根据它尝试恢复出两个相机之间的关系。只有当两帧之间的关系是安全的时候我们才会成功初始化，在这里我们会探测低视差、双解决方案的歧义的情况，这样才能避免不好的初始化毁掉整个地图。具体步骤如下：

### 1. Find initial correspondences:

在当前帧 $F_c$ 中抽取ORB特征并且在参考帧 $F_r$ 中搜索匹配，如果找到的匹配不够多就换一个参考帧

### 2. Parallel computation of the two models:

并得到homography的转换 $H$ 和fundamental matrix  $F$ 。homography使用normalized DLT算法，后面使用八点法，并且都使用RANSAC来排除outlier。对于两个模型，我们先定好相同的迭代次数，在每次迭代我们为两个模型都计算分数 $S$ （最小二乘法得到的误差），标准偏差为1像素的卡方测试的95%置信度来判断outlier。

### 3. Model selection:

根据homography和fundamental matrix的结果来判断是不是平面（是的话就应当使用homography做初始化而不是fundamental matrix）或者低视差（如果是的话使用fundamental matrix来计算位姿关系，结果会很差，应当拒绝初始化）。同样，非平面且视差较大时应当使用fundamental matrix而不是homography（可能由于图中有一小块确实组成了平面，这样homography求解也会得到结果）。检测到鲁棒的启发式函数： $RH = SH/(SH+SF)$ ， $RH > 0.45$ 时能够比较好的捕捉到平面或者低视差的情况，否则就选用fundamental matrix。

### 4. Motion and Structure from Motion recovery:

根据不同的模型使用不同的方法，homography使用Faugeras的方法恢复出8中可能的动作，fundamental matrix的模型使用SVD分解得到然后使用标定过的内参 $K$ 得到essential matrix得到可能的4个动作。（然后通过triangulate来重构？？怎么

将solution三角化??)

## 5. Bundle Adjustment

# Tracking

---

这些东西与ORB SLAM中的都一样

- ORB Extraction

ORB SLAM

- Initial Pose Estimation from Previous Frame

成功跟踪的话使用持续速度预测当前位姿，然后根据前一帧搜索map point匹配进行位姿优化

- Initial Pose Estimation via Global Relocalisation

BoW找到candidates、RANSAC进行几何验证排除outlier、PnP计算关系、使用map points的reprojection来优化位姿和匹配。

- Track Local Map

将local map (covisibility graph中相连的那些keyframe) 中的map points投影到当前frame上去来将frame上的ORB features关联到Map Points上面去。做完之后使用在frame中的map points对位姿做优化。

- New Keyframe Selection

成为keyframe的条件

这些东西与ORB SLAM中的都一样。

## Local Mapping

### 1. keyframe insertion

更新covisibility graph，更新Spanning tree，计算BoW

### 2. Recent Map Points Culling

Map points要经过创建后前三个keyframes的测试才能得以保留，这样才能避免假点，并且满足以下两个条件：

- 超过25%预测存在的帧中应当被找到
- 三个keyframes的测试



当一个map point被创建后，只有当看到它的keyframes少于三个才会被删除掉。

### 3. New Map Point Creation

一个keyframe- $K_i$ 和与它相连的一个keyframe- $K_c$ ，两个中找到仍未匹配的ORB feature，经过epipolar geometry constraint、positive depth in both cameras、parallax、reprojection error、scale consistency都检测通过后将它们三角化成同一个map point，然后投影到剩下的相连的keyframe中查找，将它们的ORB features也融入到这个map point中。

### 4. Local BA

利用local BA优化当前处理的keyframe  $K_i$ ，将所有与它相连的keyframe取出构成一个inner window与这个group相连的keyframes都取出来作为outter window，利用两个window优化inner window（outter window的keyframes不发生改变）。优化时抛弃outlier。

### 5. Local Keyframe Culling

去掉一个keyframe-90%的它的map points能够被至少三个keyframes看到

这里和ORB SLAM里面也是一样的

## Loop Closing

---

loop closing单独开一个线程做回环检测。

### 1. Loop Candidates Detection

对于 $K_i$ ，取与之相连的邻居，且( $\theta \geq 30$ )，算出最低的BoW分数 $S_{min}$ 。然后访问数据库要求给出candidates分数比 $S_{min}$ 要高，然后将其中与 $K_i$ 直接相连的去掉（因为是要检测回环），当环境中相似的地方时candidates可能有多个。

### 2. Compute the Similarity Transformation

单目相机有七个自由度：旋转、平移、放缩，因此我们要做相似变换，计算累计误差，并进行几何验证。

先计算绑定到map points上的ORB features的响应，这样我们为每个loop candidates计算了3D-3D点的响应，然后使用RANSAC，再求相似变换。如果某个相似变换 $S_{il}$ 有足够多的inliers，再对其优化（guided search of more correspondences，将map points投影进去，在一个小范围内进行搜索找响应点），然后再使用新找到的响应优化。

### 3. Loop Fusion

将重合的map points进行融合，然后更新covisibility graph（将loop closure闭

合)。

- 先对当前keyframe的位姿 $T_{i|w}$ 使用相似转换的信息 $S_{i|l}$ 进行纠正，并且将这个纠正传播到它的邻居上，使得这些转换都接上，这样loop的两端就都对齐了。
- 再将所有的loop keyframe和它的邻居看到的map points投影到 $K_i$ 和它的邻居中，然后进行匹配和搜搜，对于匹配成功的map points进行融合。
- 对于所有有关的keyframe都更新covisibility graph信息，并且创建好边来完成loop closure

#### 4. Essential Graph Optimization

为了有效的完成loop closing，我们基于Essential Graph上使用pose graph optimization进行优化，将loop closing发现的误差分不到图中。这个优化基于相似变换的信息来纠正尺度漂移。

## Appendix

---

### Bundle Adjustment

使用Huber robust cost function来计算损失函数。