



视觉SLAM：从理论到实践

第八次课 回环检测与建图



主讲人 高翔

清华大学 自动控制与工程 博士
慕尼黑工业大学计算机视觉组 博士后
Email: gao.xiang.thu@gmail.com

2017年冬



第八讲 回环检测与建图



1. 回环检测与词袋
2. 建图
3. 展望

1. 回环检测与词袋

1. 回环检测与词袋

- 回环检测的意义
 - VO和后端都存在误差
 - SLAM的建图与定位是耦合的——误差将会累计
- Loop Closing步骤
 - 检测到回环的发生
 - 计算回环修正帧与当前帧的运动
 - 验证回环是否成立
 - 闭环

1. 回环检测与词袋

- 如何检测回环是否发生——回环检测
- 最简单方法：对任意两个关键帧进行特征匹配
- 基于里程计的方法（Odometry based）
- 基于外观的方法（Appearance based）
 - 外观方法是主流
 - 核心在于衡量图像间的相似性
 - 朴素的想法：灰度相减 $s(A, B) = \|A - B\|$.
 - 显然不好，不好在哪里？

1. 回环检测与词袋

表 12-1 回环检测的结果分类

算法 \ 事实	是回环	不是回环
是回环	真阳性 (True Positive)	假阳性 (False Positive)
不是回环	假阴性 (False Negative)	真阴性 (True Negative)

- 回环检测的指标
 - Perceptual Aliasing
 - Perceptual Variability

- ROC曲线 (Precision-Recall曲线)

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}), \quad \text{Recall} = \text{TP} / (\text{TP} + \text{FN}). \quad (12.2)$$

Precision和Recall通常是矛盾的
P高则R低
R高则P低

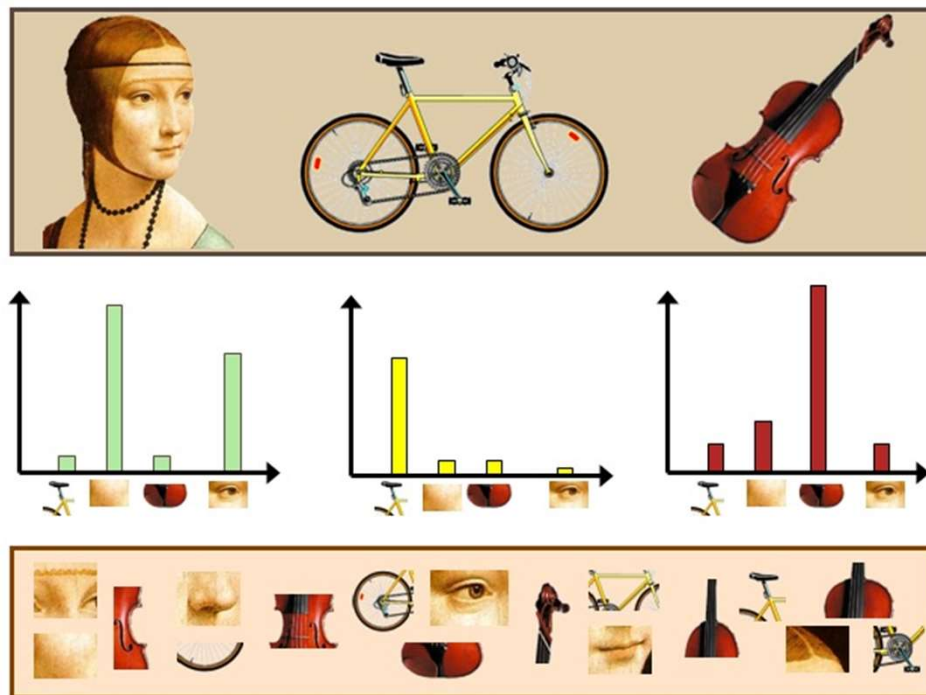


False Positive

False Negative

1. 回环检测与词袋

- Appearance-based主要方法：词袋
- Bag-of-Words, BoW
- BoW是对特征的聚类
 - 特征聚类形成了Word
 - 许多Word组成了Dictionary
 - 图像的相似性=Word的相似性
 - 只看Word的有无，无视Word的顺序



1. 回环检测与词袋

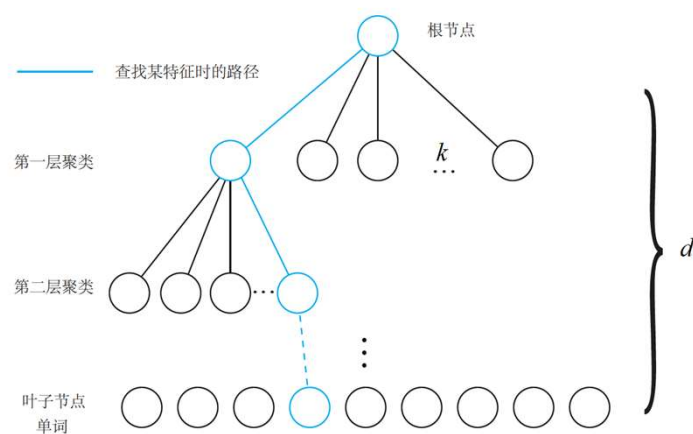
- Word的形成
 - 聚类
 - 经典的k-means聚类 N 个特征点 $\rightarrow k$ 个类

1. 随机选取 k 个中心点: c_1, \dots, c_k 。
2. 对每一个样本, 计算它与每个中心点之间的距离, 取最小的作为它的归类。
3. 重新计算每个类的中心点。
4. 如果每个中心点都变化很小, 则算法收敛, 退出; 否则返回第 2 步。

1. 回环检测与词袋

- 字典的结构

- 当实际拿到一个特征时，需要查询它对应的Word是什么
- 遍历比较： $O(n)$
- 建立字典结构可以加速比较



k-d tree: k分枝树, d层
 k^d 个Word, d次比较, $O(kd)$

1. 回环检测与词袋

- 相似度计算
 - 原则上比较Word即可计算图像相似度
 - 但一些Word很常见，另一些则很罕见
 - TF-IDF (Term Frequency-Inverse Document Frequency)
 - 思路：单词在字典中出现频率越高，则区分度越低/在图像中频率越高-则区分度越高
 - IDF部分可在字典训练过程中计算
 - TF部分则需要对图像的特征进行计算
 - 相似度：

$$A = \{(w_1, \eta_1), (w_2, \eta_2), \dots, (w_N, \eta_N)\} \triangleq \mathbf{v}_A.$$

$$s(\mathbf{v}_A - \mathbf{v}_B) = 2 \sum_{i=1}^N |\mathbf{v}_{Ai}| + |\mathbf{v}_{Bi}| - |\mathbf{v}_{Ai} - \mathbf{v}_{Bi}|.$$

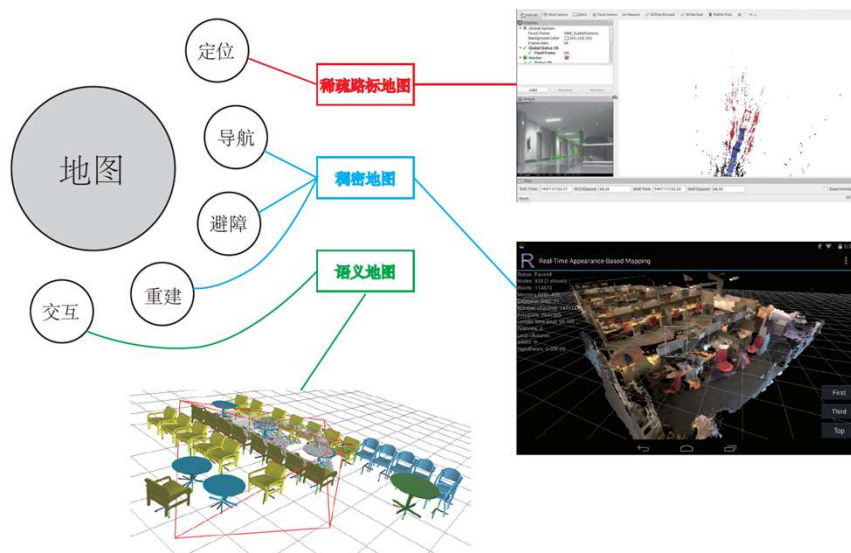
1. 回环检测与词袋

- SLAM中的回环检测
 - 使用相对的评分而非绝对的评分
 - 思路：相邻关键帧是相似的，而回环的相似度应该约等于相邻关键帧
- 检测成功以后：
 - 根据PnP等算法计算运动关系
 - 根据重投影关系验证回环是否成立
 - 利用全局BA或Pose Graph进行优化

2. 建图

2. 建图

- 目前我们谈到的都是特征点地图
- 实际当中地图往往有不同形式，对应不同需求



- 定位仅需匹配特征点
- 导航和避障需要稠密障碍物信息
- 交互需要稠密的物体表面信息
- 高层任务需要语义信息

特征点地图仅满足很小一部分需求

2. 建图

- SLAM中的地图
 - 等价于地图点的位置估计
 - 单目SLAM中可看成地图点的深度估计
 - 批量式 => Bundle Adjustment
 - 递归式 => 极线搜索、深度滤波器

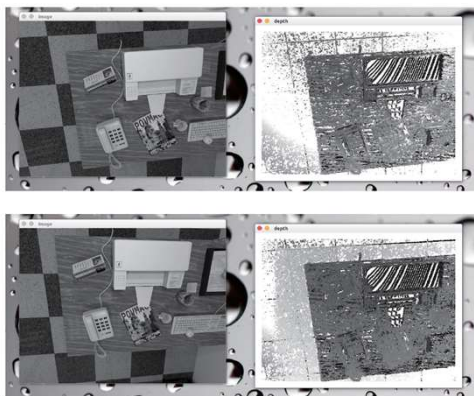


图 13-6 演示程序运行时截图。两图分别是迭代 10 次和 30 次的结果。

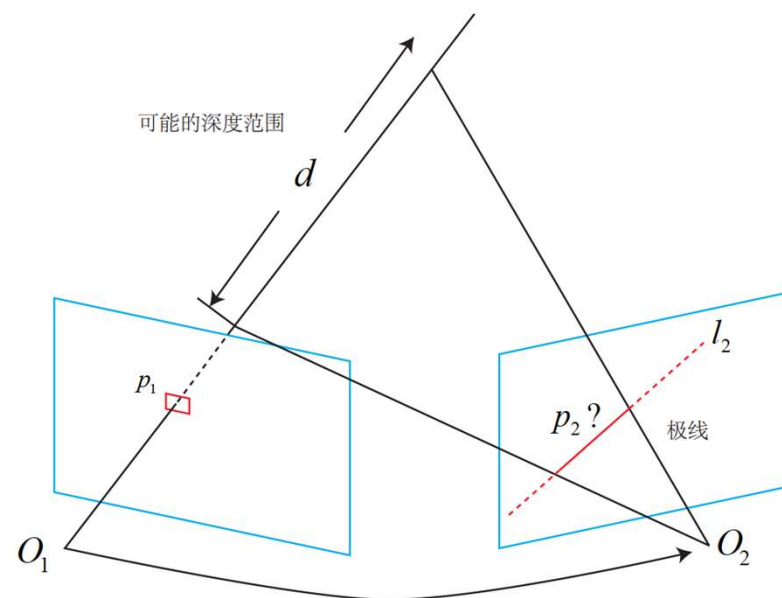
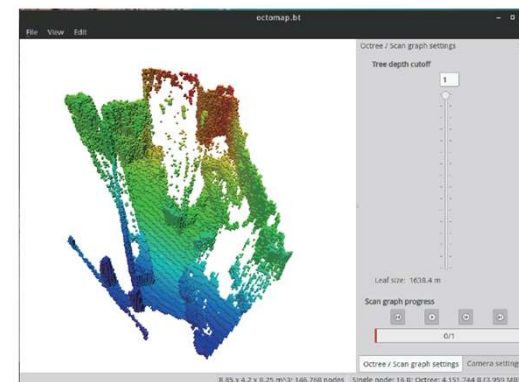


图 13-2 极线搜索示意图。

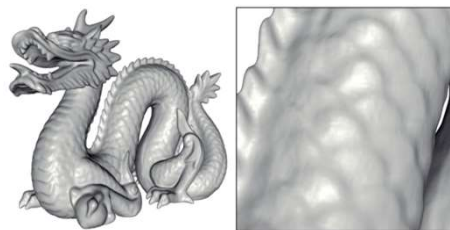
2. 建图

- RGBD稠密建图
 - 点云
 - 网格/面片
 - TSDF
 - 八叉树



八叉树地图 (0.05米分辨率)

泊松重建：从点云到网格 Surfel：以面片为单位的建图



泊松重建示例



Surfel重建示例

2. 建图

- 地图的表达仍在研究中
 - 每一个地图展开谈都是比较大的主题
 - 动态/带人物/长时间的地图
 - 参数化的紧凑表达
 - 语义信息

3. 展望

3. 展望

表 14-1 常用开源 SLAM 方案

- SLAM已有众多的开源方案
 - 开源能够让用户了解其原理, 促进研究的发展
- 今后的方向
 - 多传感器融合 (V+IMU, V+GPS, V+ODOM)
 - SLAM+DL
 - Depth learning
 - Stereo learning
 - Descriptor Learning
 - Semantic mapping/BA
 - Etc.

方案名称	传感器形式	地址
MonoSLAM	单目	https://github.com/hanmekim/SceneLib2
PTAM	单目	http://www.robots.ox.ac.uk/~gk/PTAM/
ORB-SLAM	单目为主	http://webdiis.unizar.es/~raulmur/orbslam/
LSD-SLAM	单目为主	http://vision.in.tum.de/research/vslam/lsdslam
SVO	单目	https://github.com/uzh-rpg/rpg_svo
DTAM	RGB-D	https://github.com/anuranbaka/OpenDTAM
DVO	RGB-D	https://github.com/tum-vision/dvo_slam
DSO	单目	https://github.com/JakobEngel/dso
RTAB-MAP	双目/RGB-D	https://github.com/introlab/rtabmap
RGBD-SLAM-V2	RGB-D	https://github.com/felixendres/rgbdslam_v2
Elastic Fusion	RGB-D	https://github.com/mp3guy/ElasticFusion
Hector SLAM	激光	http://wiki.ros.org/hector_slam
GMapping	激光	http://wiki.ros.org/gmapping
OKVIS	多目 +IMU	https://github.com/ethz-asl/okvis
ROVIO	单目 +IMU	https://github.com/ethz-asl/rovio

谢谢各位对本次课程的关注！

Thank you for attention!