# 2 Bundle Adjustment (5 分，约 3 小时)

## 2.1 文献阅读 (2 分)

### 1. 为何说 Bundle Adjustment is slow 是不对的？

BA is slow 是因为考虑这个问题的时候只考虑了的BA常规形式，没有考虑具体BA问题的结构和稀疏性。

### 2. BA 中有哪些需要注意参数化的地⽅？Pose 和 Point 各有哪些参数化⽅式？有何优缺点

BA问题中点的位移，旋转，甚至相机内参都能参数化。根据具体的形式，我们的状态可能受到奇异性，内部约束和不必要的内部自由度的影响。最合适的参数化应该要做到统一，有限，行为良好且接近当前状态。

对于点来说，可以参数化成XYZ（仿射形式）或者XYZW（射影形式），在距离相机近的点我们使用xyz形式就可以了。但是对于很远的点，因为很远的点和无限远的点本身就不容易区分，还加上噪声问题就可能让我们混肴。在很远的位置，xyz的cost function十分的平坦，steps会变得很大。用XYZW形式的话它会变得自然，有限，条件良好。

对于Pose（位移部分跟点同理）的旋转部分。用Euler angle就有很多数值问题，除非能保证避免奇异性和不平坦空间。可以用单位四元数或者在旋转矩阵R上面加个小扰动来表示。

### 3. \* 本⽂写于 2000 年，但是⽂中提到的很多内容在后⾯⼗⼏年的研究中得到了印证。你能看到哪些 ⽅向在后续⼯作中有所体现？请举例说明

模型要考虑outlier

控制步长

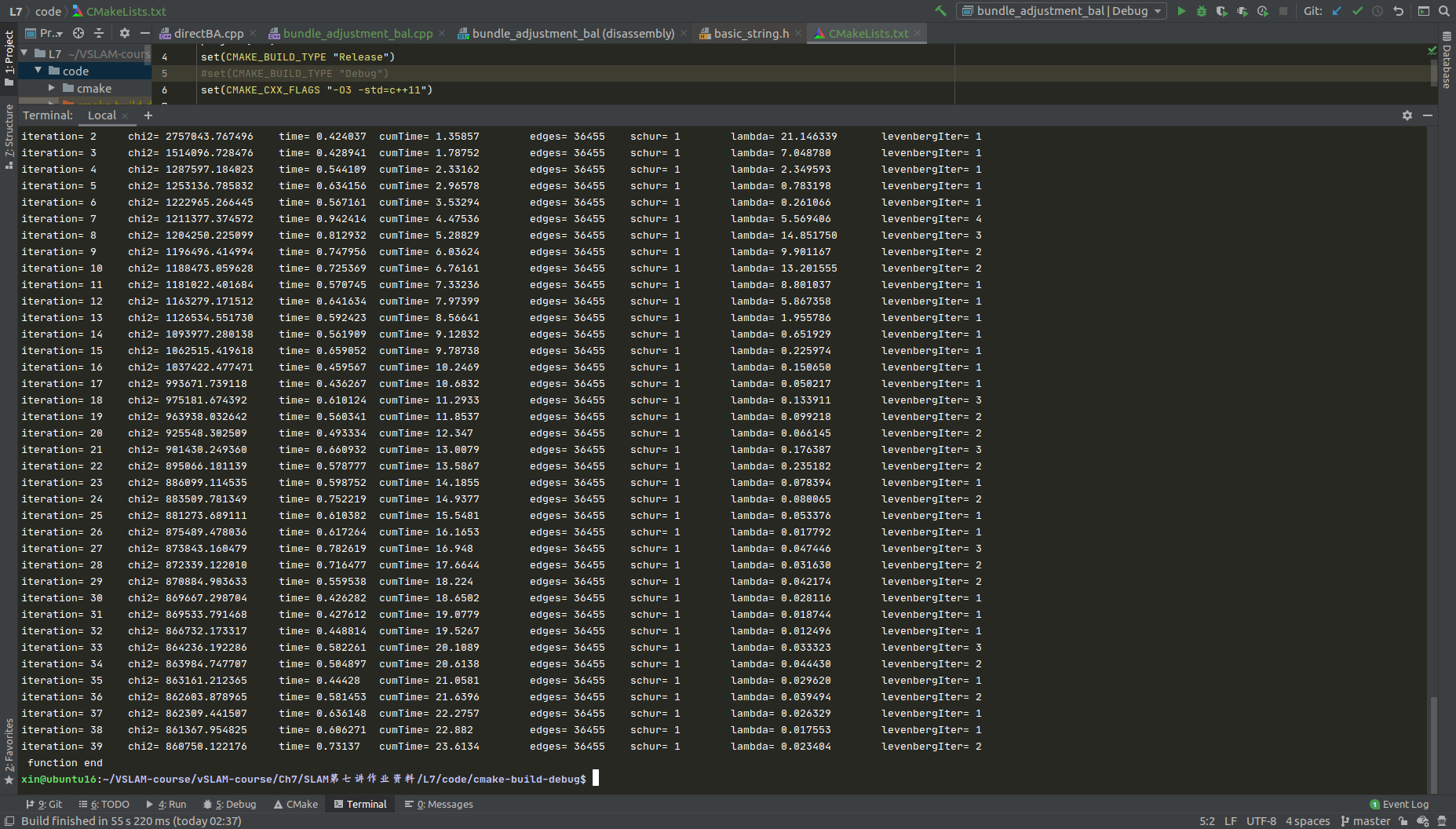
可以探索问题的结构性，善用因式分解，使用稳定的局部参数，缩放和预处理。

## 2.2 BAL-dataset（3 分）

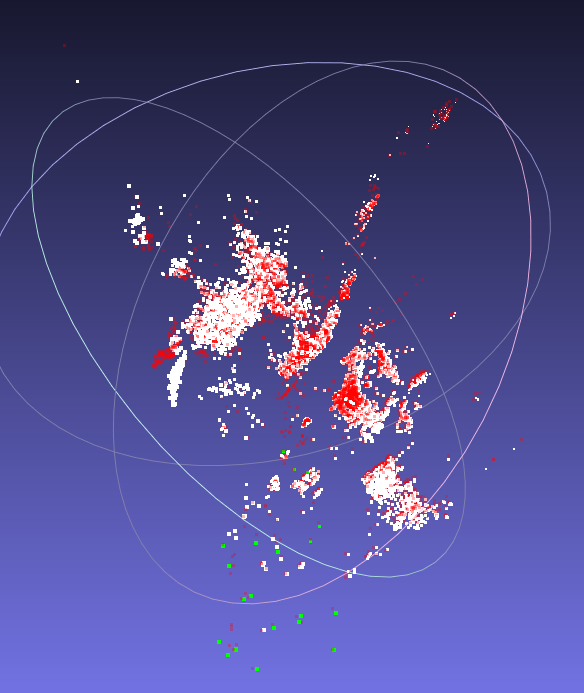
### 请你使⽤ g2o，⾃⼰定义 Vertex 和 Edge书写 BAL 上的 BA 程序，运⾏你的 BA，并给出优化后的点云图。

代码参考了书上示例，详情参见.cpp文件

程序运行结果可见误差在缩小



点云地图，白色为原始位姿，红色为优化后的位姿，主要是同一面上的点云经过优化后都集中到一个平面了。



# 3 直接法的 Bundle Adjustment (5 分，约 3 小时)

## 3.1 数学模型

### 1. 如何描述任意⼀点投影在任意⼀图像中形成的 error？

即任意一点投影到某相机的成像平面后，计算该点（及其patch）的像素和参考像素的差值。

H函数为观测方程

### 2. 每个 error 关联⼏个优化变量？

两个，一个是相机位姿(translation和rotation各3维)

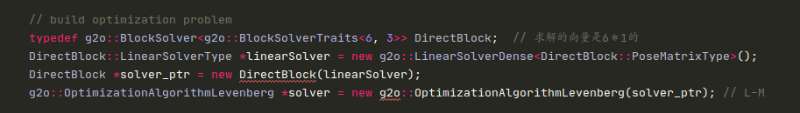
另一个是点的位置（3维）

### 3. error 关于各变量的雅可⽐是什么

此处XYZ为特征点在相机系坐标，R为世界系到相机系的旋转矩阵。

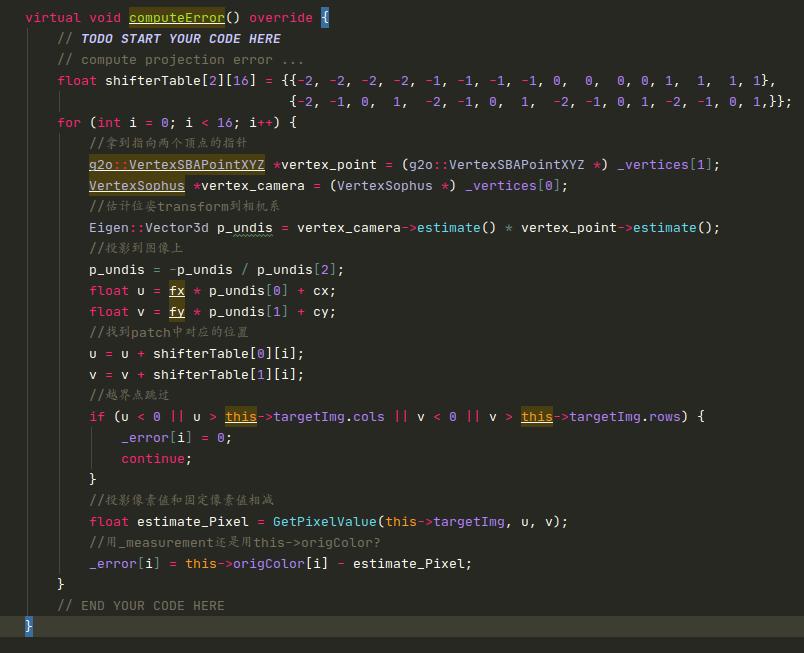
## 3.2 实现-根据上述说明，使⽤ g2o 实现上述优化，并⽤ pangolin 绘制优化结果

这题遇到了error，暂时没有解决。



以下为代码和思路

把估计点投影到相机系统，加上内参放到成像平面，找到patch内对应的位置，判断是否出界，结算该像素值和参考像素值的差。



往optimizer里面加入各个顶点（特征点和位姿），把边和顶点相关联，加入optimizer。



把优化完的数据塞回poses和poitns方便显示函数调用。

### 

### 能否不要以 [x,y,z]T 的形式参数化每个点？

参考阅读题，可以参数化为XYZW形式

### 取 4x4 的 patch 好吗？取更⼤的 patch 好还是取⼩⼀点的 patch 好？

应该跟直接法本身的patch一样，大了小了都不一定更好。

主要是尽量取到跟”局部光度特征”大小差不多的patch最好。

### 从本题中，你看到直接法与特征点法在 BA 阶段有何不同？

Error原理不同，直接法是光度误差，特征点法是特征点的重投影误差。

Error不同之后求偏导的各项都会不一样。

### 4. 由于图像的差异，你可能需要鲁棒核函数，例如 Huber。此时 Huber 的阈值如何选取？

这应该是个经验值吧，考虑到像素变化不应该太大，要我的话可能会取255/5=51。