

## 2 Bundle Adjustment

### 2.1 文献阅读

1. 缓慢是因为没有考虑问题的结构和稀疏性，使用了通用的优化例程。（The claimed slowness is almost always due to the unthinking use of a general-purpose optimization routine that completely ignores the problem structure and sparseness.）
2. 理想情况下，误差模型应该局部接近线性。任何给定的参数化通常只在状态空间相对较小的部分表现良好。因此，状态更新都应该使用基于当前估计增量的稳定的局部参数化来评估。 Pose和Point参数化方式有四元数，欧拉角，变换矩阵。 1) **变换矩阵**：优点：旋转轴可以是任意向量 缺点：旋转其实只需要知道一个向量+一个角度(共4个信息值)，但矩阵却用了16个元素。 2) **欧拉角**：优点：表示更方便，只需要三个值(分别对应x、y、z轴的旋转角度) 缺点：欧拉角不同的顺序会造成不同结果；欧拉角旋转会造成万向锁现象。 3) **四元数**：优点：可以避免万向锁。 缺点：比欧拉旋转稍微复杂了一点，多了一个维度。
3. 文中提到了“Network Structure”，是现在主流的**图优化**，代替EKF这种滤波方法。

## 3 直接法的Bundle Adjustment

### 3.1 数学模型

1.

$$\begin{aligned} p_1 &= \frac{1}{Z_1} KP \\ P' &= TP \\ p_2 &= \frac{1}{Z_2} KP' \\ e(T) &= I_1(p_1) - I_2(p_2) \end{aligned}$$

2. 一个优化变量：变换矩阵T。

3. 对于变换矩阵T：

$$\begin{aligned} \frac{\sigma e}{\sigma T} &= \frac{\sigma I_2}{\sigma p_2} \frac{\sigma p_2}{\sigma P'} \frac{\sigma P'}{\sigma \delta \xi} \delta \xi \\ &= \begin{bmatrix} \frac{\sigma I_2}{\sigma x_2} & \frac{\sigma I_2}{\sigma y_2} \end{bmatrix} \begin{bmatrix} \frac{f_x}{Z} & 0 & -\frac{f_x X}{Z^2} & -\frac{f_x XY}{Z^2} & f_x + \frac{f_x X^2}{Z^2} & -\frac{f_x Y}{Z} \\ 0 & \frac{f_y}{Z} & -\frac{f_y Y}{Z^2} & -f_y - \frac{f_y Y^2}{Z^2} & \frac{f_y XY}{Z^2} & \frac{f_y X}{Z} \end{bmatrix} \end{aligned}$$

### 3.2 实现

1. 我觉得是需要参数化的，图中u-v坐标系下的像素坐标是需要通过[x, y, z]与变换矩阵及相机模型得到的。
2. patch的大小我认为视情况而定吧，纹理弱的需要patch大一点，需要更多的信息进行比对。
3. 误差函数不同，特征点法是重投影误差，直接法是光度误差；雅可比矩阵也不同。
4. Huber的阈值应取比Error集中出现的数值区域大的数值。

