深蓝学院
shenlanxueyuan.com

基于图像的三维模型重建

——双视角SFM

主讲人 隋博士

课程内容

- ✓三角量测(Triangulation)
  - ✓ 直接线性变换法
- ✓ 3D-2D: PnP问题
  - ✓ 直接线性变换法
  - ✓ 非线性优化
- ✓ 捆绑调整Bundle Adjustment

三角量测

深蓝学院 shenlanxueyuan.com

已知**相机参数**和**匹配点**恢复三维点的坐标

第 i 相机投影矩阵：

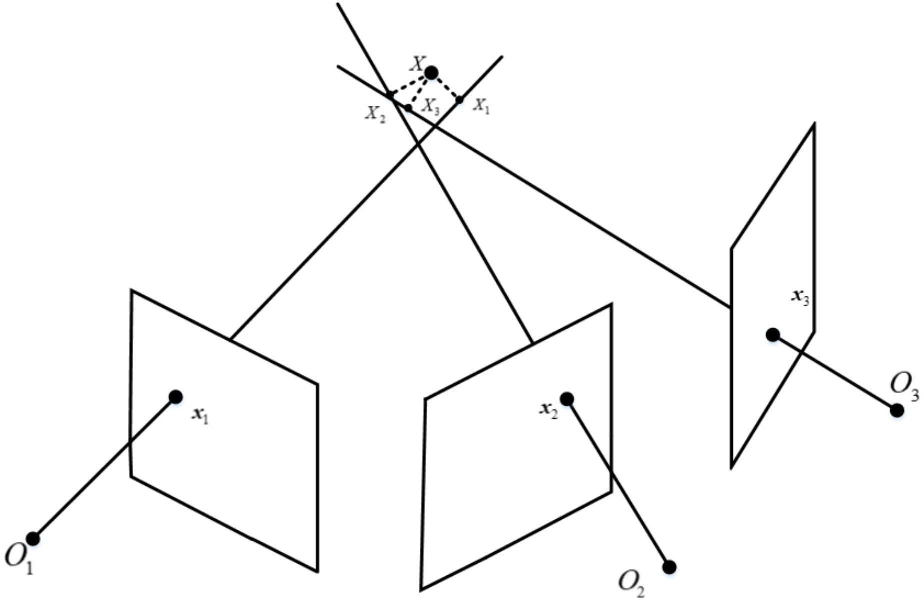
$$\boldsymbol{P}_{\mathrm{i}} = \boldsymbol{K}_i[\boldsymbol{R}_i, \boldsymbol{t}_i] = \begin{bmatrix} \boldsymbol{P}_{i1} \\ \boldsymbol{P}_{i2} \\ \boldsymbol{P}_{i3} \end{bmatrix}$$

三维点点坐标：

$$\boldsymbol{X} = [x, y, z, 1]^T$$

在第 i 个视角中投影的图像坐标为：

$$\mathbf{x}_{\mathrm{i}} = [x_{\mathrm{i}}, y_{\mathrm{i}}, 1]^T$$



三角量测

已知相机参数和匹配点恢复三维点的坐标

根据投影方程可以得到：

$$\mathbf{x}_\mathrm{i} = \boldsymbol{P}_i \boldsymbol{X},$$

上述等式两侧同时叉乘 x_i：

$$\mathbf{x}_i \times (\boldsymbol{P}_i \boldsymbol{X}) = \mathbf{0}$$

即

$$x_i(\boldsymbol{P}_{i3}\boldsymbol{X}) - \boldsymbol{P}_{i1}\boldsymbol{X} = \mathbf{0}$$
$$y_i(\boldsymbol{P}_{i3}\boldsymbol{X}) - \boldsymbol{P}_{i2}\boldsymbol{X} = \mathbf{0}$$
$$x_i(\boldsymbol{P}_{i2}\boldsymbol{X}) - y_i(\boldsymbol{P}_{i1}\boldsymbol{X}) = \mathbf{0}$$

$$\Rightarrow \begin{bmatrix} x_\mathrm{i}\boldsymbol{P}_{\mathrm{i}3} - \boldsymbol{P}_{\mathrm{i}1} \\ y_\mathrm{i}\boldsymbol{P}_{\mathrm{i}3} - \boldsymbol{P}_{\mathrm{i}2} \end{bmatrix} \boldsymbol{X} = \boldsymbol{0}$$

注意 第3个方程与前两个方程**线性相关**

三角量测

已知相机参数和匹配点恢复三维点的坐标

1个观察点提提供2个约束，**X有3**个自由度，至少2对点

$$A\,X=0,$$

$$A=\begin{bmatrix} x_1\boldsymbol{P}_{13}-\boldsymbol{P}_{11} \\ y_1\boldsymbol{P}_{13}-\boldsymbol{P}_{12} \\ \cdots \\ x_{\mathrm{i}}\boldsymbol{P}_{\mathrm{i}3}-\boldsymbol{P}_{\mathrm{i}1} \\ y_{\mathrm{i}}\boldsymbol{P}_{\mathrm{i}3}-\boldsymbol{P}_{\mathrm{i}2} \\ \cdots \\ x_{\mathrm{N}}\boldsymbol{P}_{\mathrm{N}3}-\boldsymbol{P}_{\mathrm{N}1} \\ y_N\boldsymbol{P}_{\mathrm{N}3}-\boldsymbol{P}_{\mathrm{N}2} \end{bmatrix},\boldsymbol{N}\geq\boldsymbol{2}$$

三角量测

思考

如果存在外点（匹配点或姿态）的情况下，如何得到准确的的三维点坐标？

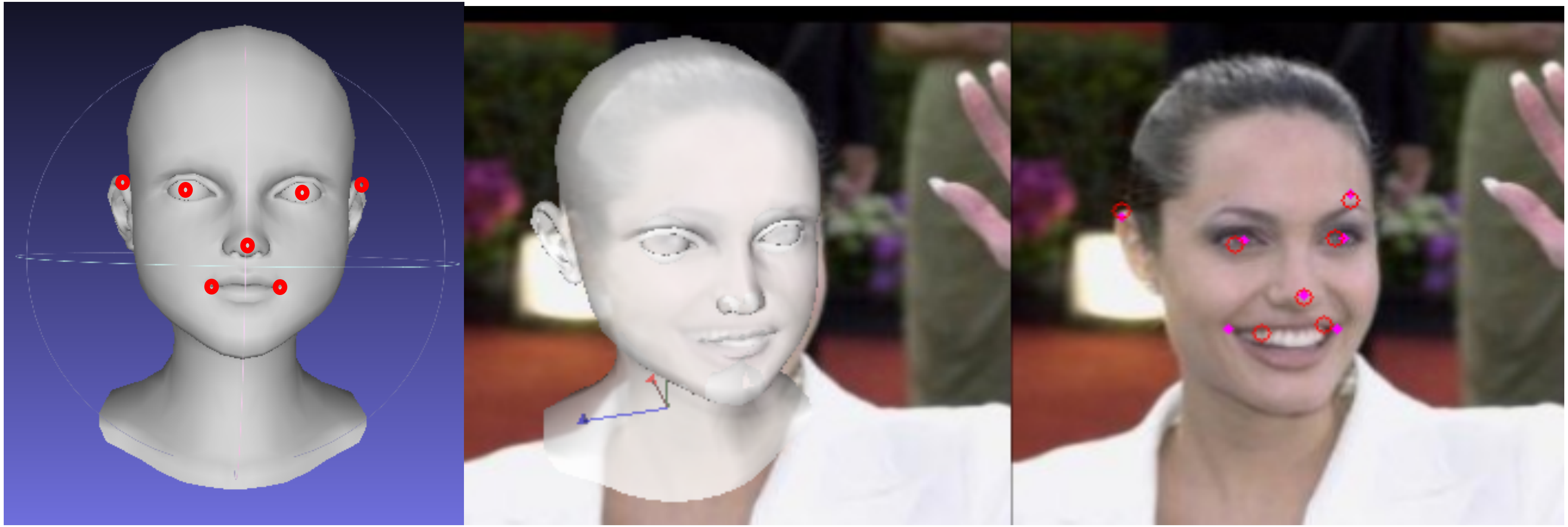
三角量测

Coding-1:

完成task-3/class3_test_triangulation.cc中利用直接线性变化法求三维点坐标

3D-2D:PnP问题-PnP

已知三维点和对应二维点求解相机内外参数



3D-2D:PnP问题-直接线性变换法

已知三维点和对应二维点求解相机内外参数

$$s\begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \boldsymbol{K}\left[\boldsymbol{R}, \boldsymbol{t}\right]\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} T_{11}, & T_{12}, & T_{13}, & T_{14} \\ T_{21}, & T_{22}, & T_{23}, & T_{24} \\ T_{31}, & T_{32}, & T_{33}, & T_{34} \end{bmatrix}\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{pmatrix} \boldsymbol{r}_1^T \\ \boldsymbol{r}_2^T \\ \boldsymbol{r}_3^T \end{pmatrix}\boldsymbol{X}$$

$$u = \frac{T_{11}X + T_{12}Y + T_{13}Z + T_{14}}{T_{31}X + T_{32}Y + T_{33}Z + T_{34}} = \frac{\boldsymbol{X}^T\boldsymbol{r}_1}{\boldsymbol{X}^T\boldsymbol{r}_3}$$

$$v = \frac{T_{21}X + T_{22}Y + T_{23}Z + T_{24}}{T_{31}X + T_{32}Y + T_{33}Z + T_{34}} = \frac{\boldsymbol{X}^T\boldsymbol{r}_2}{\boldsymbol{X}^T\boldsymbol{r}_3}$$

$$\boldsymbol{X}^T\boldsymbol{r}_1 - \boldsymbol{X}^T\boldsymbol{r}_3 u = 0$$

$$\boldsymbol{X}^T\boldsymbol{r}_2 - \boldsymbol{X}^T\boldsymbol{r}_3 v = 0$$

3D-2D:PnP问题–直接线性变换法

深蓝学院 shenlanxueyuan.com

已知三维点和对应二维点求解相机内外参数

$$\begin{pmatrix} X_1^T & 0 & -uX_1^T \\ 0 & X_1^T & -vX_1^T \\ \vdots & \vdots & \vdots \\ X_N^T & 0 & -uX_N^T \\ 0 & X_N^T & -vX_N^T \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \mathbf{0}$$

共需要至少6对3D-2D对应点

$T = [KR, Kt]$ **矩阵QR分解获取K,R,t**

3D-2D:PnP问题-其它常用方法

P3P法：需要4对不共面的点 求出2D点在当前相机坐标系中的3D点，然后进行3D-3D的姿态求解

https://www.cnblogs.com/mafuqiang/p/8302663.html
Complete Solution Classification for the Perspective-Three-Point Problem
A novel parametrization of the perspective-three-point problem for a direct computation of absolute camera position and orientation

ePnP法：需要(>=4不共面或者3对共面) 点进行求解

https://www.cnblogs.com/jian-li/p/5689122.html
EPnP: Efficient Perspective-n-Point Camera Pose Estimation

课程内容

- ✓ 针孔相机模型
  - ✓ 针孔相机模型
  - ✓ 径向畸变
- ✓ 2D-2D:对极几何
  - ✓ 对极约束
  - ✓ 本质/单应矩阵
  - ✓ 直接线性变换法
- ✓ 3D-2D: PnP问题
  - ✓ 三角量测
  - ✓ 直接线性变换法
  - ✓ 非线性优化
- ✓ **捆绑调整Bundle Adjustment**

捆绑调整Bundle Adjustment

深蓝学院 shenlanxueyuan.com

问题阐述 同时对三维点位置和相机参数进行非线性优化

观测点

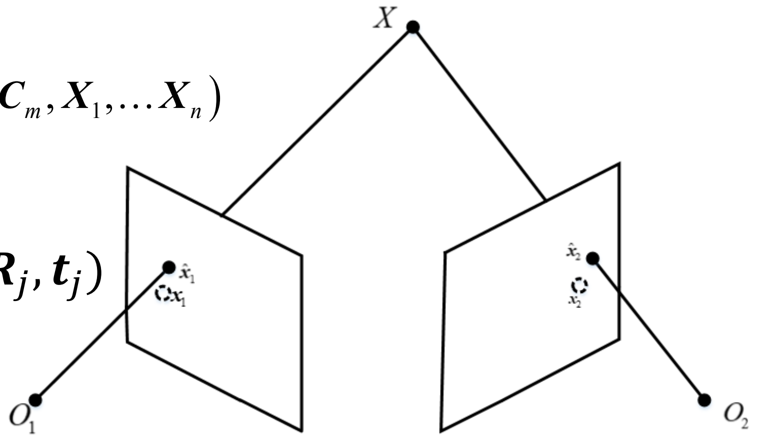
三维点坐标 $\boldsymbol{X}_i = (X_i, Y_i, Z_i)^T$

$$g(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i}^{n}\sum_{j}^{m}\chi_{ij}\left\|\hat{\boldsymbol{u}}_{ij} - \boldsymbol{u}_{ij}\left(\boldsymbol{C}_j, \boldsymbol{X}_i\right)\right\|^2 = \frac{1}{2}\sum_{i}^{n}\sum_{j}^{m}\chi_{ij}e_{ij}, \qquad \boldsymbol{\theta} = (\boldsymbol{C}_1, \ldots, \ \boldsymbol{C}_m, \boldsymbol{X}_1, \ldots \boldsymbol{X}_n)$$

投影点 相机参数 $\boldsymbol{C}_j = (f_j, k_{ij}, k_{2j}, \boldsymbol{R}_j, \boldsymbol{t}_j)$

$\chi_{ij}=1$ 表示第 i 个点在第 j 个相机中可见

$\boldsymbol{\theta} \in \boldsymbol{R}^{9m\times 3n}$ 高维空间的非线性优化



捆绑调整Bundle Adjustment

无约束非线性最小优化问题

$$min\ g(\boldsymbol{\theta}) = min\|\mathbf{x} - f(\boldsymbol{\theta})\|^2,\ \boldsymbol{\theta} \in \boldsymbol{R}^n$$

优化上述问题的最优解通常是指它的**局部最优解**，因此需要一个较好的初始值

捆绑调整Bundle Adjustment

深蓝学院
shenlanxueyuan.com

最速下降法--假设函数一阶可微

假设g($\boldsymbol{\theta}$)在$\boldsymbol{\theta}_t$处可微，则它在$\boldsymbol{\theta}_t$处有Taylor展开式:

$$g(\boldsymbol{\theta})=g(\boldsymbol{\theta}_t)+ (\nabla g(\boldsymbol{\theta}_t))^T \delta\boldsymbol{\theta} + o(\|\delta\boldsymbol{\theta}\|),$$

其中 $\boldsymbol{\theta} = \boldsymbol{\theta}_t + \delta\boldsymbol{\theta}$

- 当 $(\nabla g(\boldsymbol{\theta}_t))^T \delta\boldsymbol{\theta} < 0$时可保证g($\boldsymbol{\theta}$)的值是在下降；
- 当$\delta\boldsymbol{\theta} = -\nabla g(\boldsymbol{\theta}_t)$时，可达到最快的下降速度(略去高阶不计)；

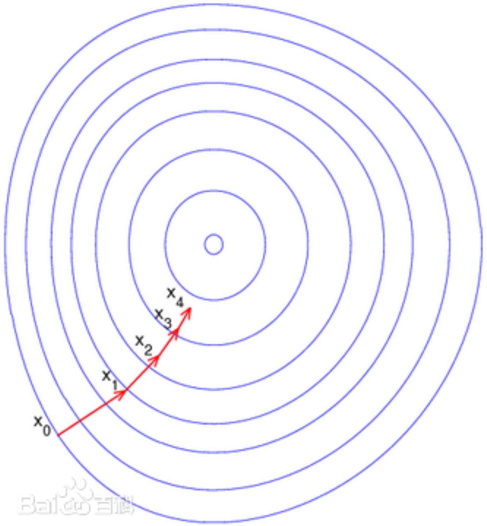
捆绑调整Bundle Adjustment

最速下降法-算法流程

1）给定初始点$\boldsymbol{\theta}_0$，终止控制函数$\epsilon > 0$和步长λ，令$t =$

2）计算$\nabla g(\boldsymbol{\theta}_t)$，若$\|\nabla g(\boldsymbol{\theta}_t)\| \le \epsilon$，停止迭代，输出$\boldsymbol{\theta}_t$，否则进行下一步；

3）取$\boldsymbol{\theta}_{t+1}=\boldsymbol{\theta}_t - \lambda\nabla g(\boldsymbol{\theta}_t)$, t=t+1 转第2）步。



越接近极值速度越慢

捆绑调整Bundle Adjustment

深蓝学院 shenlanxueyuan.com

牛顿法--假设函数二阶可微

假设g($\boldsymbol{\theta}$)在$\boldsymbol{\theta}_t$处二阶可微，且假定二阶导数 ∇^2g($\boldsymbol{\theta}$)总是正定的，则它在$\boldsymbol{\theta}_t$处，以g($\boldsymbol{\theta}$)的二阶近似函数

$$Q(\boldsymbol{\theta})=g(\boldsymbol{\theta}_t)+ (\nabla g(\boldsymbol{\theta}_t))^T \delta\boldsymbol{\theta} + \frac{1}{2}\delta^T \boldsymbol{\theta}\nabla^2 g(\boldsymbol{\theta}_t)\delta\boldsymbol{\theta}, \boldsymbol{\theta} = \boldsymbol{\theta}_t + \delta\boldsymbol{\theta}$$

的极小值点作为下一次迭代点$\boldsymbol{\theta}_{t+1}$。

对上式求梯度并令其等于0，可以得到

$$\nabla Q(\boldsymbol{\theta})=\nabla g(\boldsymbol{\theta}_t) + \nabla^2 g(\boldsymbol{\theta}_t)\delta\boldsymbol{\theta}=0$$

$$\delta\boldsymbol{\theta} = -(\nabla^2 g(\boldsymbol{\theta}_t))^{-1}\nabla g(\boldsymbol{\theta}_t)$$

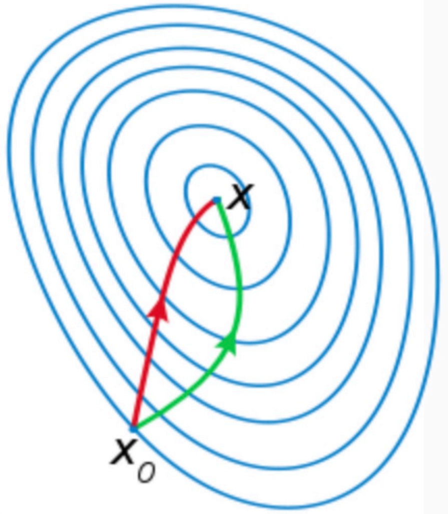
捆绑调整Bundle Adjustment

牛顿法—算法流程

1）给定初始点$\boldsymbol{\theta}_0$，终止控制函数$\epsilon > 0$和步长$\lambda = 1$，令$t = 0$；

2）计算$\nabla g(\boldsymbol{\theta}_t)$，若$\|\nabla g(\boldsymbol{\theta}_t)\| \le \epsilon$，停止迭代，
输出$\boldsymbol{\theta}_t$，否则进行下一步；

3）取$\boldsymbol{\theta}_{t+1}$=$\boldsymbol{\theta}_t$- $(\nabla^2 g(\boldsymbol{\theta}_t))^{-1}\nabla g(\boldsymbol{\theta}_t)$, t=t+1 转第2）步。



牛顿法
最速下降法

捆绑调整Bundle Adjustment

牛顿法

1）速度快

最速下降法是局部平面拟合，牛顿法是局部二次曲面拟合

2）计算量大

需要计算和保存二阶Hessian矩阵的逆矩阵

3）要求初始点离最优点较近

否则无法保证收敛，甚至无法保证下降性

捆绑调整Bundle Adjustment

Levenberg-Marquardt法–原理与优势

原理：

是一种“信赖阈”的方法，当收敛速度较快时，增大信赖域，使算法趋向于牛顿法；当收敛速度较慢时，减小信赖域，使算法趋向于最速梯度法

优势：

- 速度快，只用到一阶矩阵
- 可以在距离初始值较远处得到最优解

捆绑调整Bundle Adjustment

Levenberg-Marquardt法—实现

$$\nabla^2 \mathrm{g}(\boldsymbol{\theta}_t)\delta\boldsymbol{\theta} = -\nabla\, \mathrm{g}(\boldsymbol{\theta}_t),$$

将上述公式中的$\nabla^2 \mathrm{g}(\boldsymbol{\theta}_t)$替换成$\boldsymbol{J}^T(\boldsymbol{\theta}_t)\boldsymbol{J}(\boldsymbol{\theta}_t) + \frac{1}{\lambda}\boldsymbol{I}$,并且$\nabla\, \mathrm{g}(\boldsymbol{\theta}_t)$=$-\boldsymbol{J}^T(\boldsymbol{\theta}_t)(\mathbf{x} - f(\boldsymbol{\theta}_t))$

$$\boxed{(\boldsymbol{J}^T(\boldsymbol{\theta}_t)\boldsymbol{J}(\boldsymbol{\theta}_t) + \frac{1}{\lambda}\boldsymbol{I})\delta\boldsymbol{\theta} = \boldsymbol{J}^T(\boldsymbol{\theta}_t)(\mathbf{x} - f(\boldsymbol{\theta}))}$$ 增量正规方程

,

λ为信赖域半径, $\boldsymbol{J}(\boldsymbol{\theta}_t) = \nabla f(\boldsymbol{\theta}_t)$

当λ趋向于无穷大时, $\boldsymbol{J}^T(\boldsymbol{\theta}_t)\boldsymbol{J}(\boldsymbol{\theta}_t)\delta\boldsymbol{\theta} = -\nabla\, \mathrm{g}(\boldsymbol{\theta}_t)$

当λ趋向于零时, $\delta\boldsymbol{\theta} = -\nabla\, \mathrm{g}(\boldsymbol{\theta}_t)$

捆绑调整Bundle Adjustment

深蓝学院 shenlanxueyuan.com

Levenberg-Marquardt法--算法流程

1. $t=0$ 时, 选取初始点 $\boldsymbol{\theta}_0$, 终止控制常数 ε, 令 $e^0=\left\|\mathbf{x}-f(\boldsymbol{\theta}_0)\right\|^2$, $\lambda_0=10^{-3}$

2. 计算 $J^T(\boldsymbol{\theta}_t)$

3. 构造增量正规方程 $\left(J^T(\boldsymbol{\theta})J(\boldsymbol{\theta})+\lambda_t\boldsymbol{I}\right)\delta(\boldsymbol{\theta})=(\boldsymbol{\theta}_{t+1}-\boldsymbol{\theta}_t)=J^T(\boldsymbol{\theta}_t)\left(\mathbf{x}-f(\boldsymbol{\theta}_t)\right)$

4. 通过求解增量正规方程, 得到 $\delta(\boldsymbol{\theta})$

如果 $\left\|\mathbf{x}-f(\boldsymbol{\theta}_t+\delta\boldsymbol{\theta})\right\|^2<e^k$, 令 $\boldsymbol{\theta}_{t+1}=\boldsymbol{\theta}_t+\delta(\boldsymbol{\theta})$,

如果 $\left\|\delta(\boldsymbol{\theta})\right\|<\varepsilon$, 终止迭代;

否则, 令 $\lambda_{t+1}=0.1\lambda_t$, $t=t+1$, 执行第2步

否则 $\left\|\mathbf{x}-f(\boldsymbol{\theta}_t+\delta\boldsymbol{\theta})\right\|^2\geq e^k$, 令 $\lambda_{t+1}=10\lambda_t$, 执行第3步

捆绑调整Bundle Adjustment

Coding-2:

阅读task-3/class3_test_lm_optimization.cc的代码流程，归纳LM算法的流程，并写成伪代码

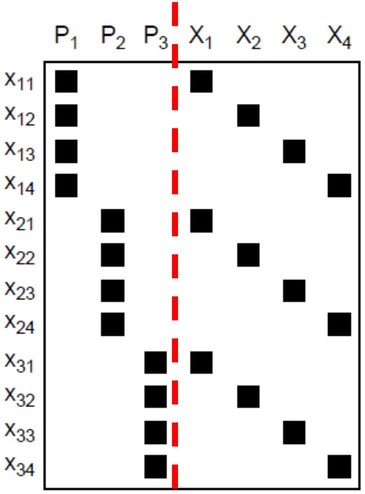
捆绑调整Bundle Adjustment

Levenberg-Marquardt法—增量规方程的求解

对称、稀疏

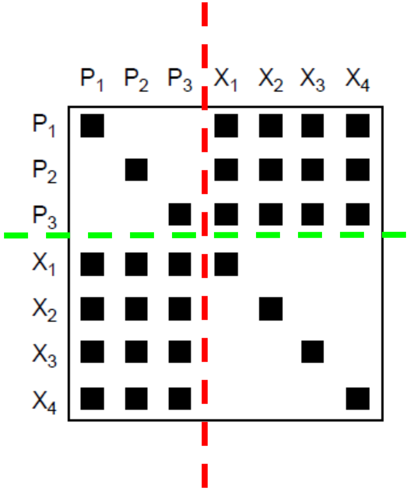
$$\boxed{\left(J^T(\theta)J(\theta)+\lambda_t I\right)}\delta(\theta)=\left(\theta_{t+1}-\theta_t\right)=J^T(\theta_t)\left(\mathbf{x}-f(\theta_t)\right)$$

$$J(\theta_t)=[J_C\ J_X]$$



(a)

$J^T(\theta)$



(b)

$J^T(\theta)J(\theta)$

捆绑调整Bundle Adjustment

深蓝学院 shenlanxueyuan.com

Levenberg-Marquardt法-正规方程的求解

$$\left(J^T(\boldsymbol{\theta})J(\boldsymbol{\theta})+\lambda_t \boldsymbol{I}\right)\delta(\boldsymbol{\theta})=J^T(\boldsymbol{\theta}_t)\left(\mathbf{x}-f(\boldsymbol{\theta}_t)\right)$$

左乘 $\begin{bmatrix} \boldsymbol{I} & -\boldsymbol{J}_{CX}\boldsymbol{J}_{XX}^{-1} \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$

$$\begin{bmatrix} \boldsymbol{J}_{CC} & \boldsymbol{J}_{CX} \\ \boldsymbol{J}_{CX} & \boldsymbol{J}_{XX} \end{bmatrix}\begin{bmatrix} \delta_C \\ \delta_X \end{bmatrix}=\begin{bmatrix} b_C \\ b_X \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{J}_{CC}-\boldsymbol{J}_{CX}\boldsymbol{J}_{XX}^{-1}\boldsymbol{J}_{CX} & \boldsymbol{0} \\ \boldsymbol{J}_{CX} & \boldsymbol{J}_{XX} \end{bmatrix}\begin{bmatrix} \delta_C \\ \delta_X \end{bmatrix}=\begin{bmatrix} b_C-\boldsymbol{J}_{CX}\boldsymbol{J}_{XX}^{-1}b_X \\ b_X \end{bmatrix}$$

$$\left(\boldsymbol{J}_{CC}-\boldsymbol{J}_{CX}\boldsymbol{J}_{XX}^{-1}\boldsymbol{J}_{CX}\right)\boldsymbol{\delta}_C=\boldsymbol{b}_C-\boldsymbol{J}_{CX}\boldsymbol{J}_{XX}^{-1}\boldsymbol{b}_X$$

线性方程，共轭梯度法求解

$$\boldsymbol{J}_{XX}\boldsymbol{\delta}_X=\boldsymbol{b}_X-\boldsymbol{J}_{CX}\boldsymbol{\delta}_C$$

捆绑调整Bundle Adjustment

深蓝学院
shenlanxueyuan.com

雅阁比矩阵的计算

以第 i 个三维点在第j 个相机中的投影为例

$$e_{ij}=\frac{1}{2}\left\|\boldsymbol{e}_{ij}\right\|^2=\frac{1}{2}\left\|\hat{\boldsymbol{u}}_{ij}-\boldsymbol{u}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)\right\|^2=\frac{1}{2}\left(\left(u_{ij}-\hat{u}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)\right)^2+\left(v_{ij}-\hat{v}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)\right)^2\right)$$

$$\boldsymbol{\xi}_{ij}=\begin{bmatrix}\boldsymbol{C}_j\\ \boldsymbol{X}_i\end{bmatrix},\quad \boldsymbol{C}_j=\left(f_j,k_1,k_2,\boldsymbol{R}_j,\boldsymbol{t}_j\right)$$

详细过程见参考文件
BA雅各比矩阵推导.pdf

$$\frac{\partial e_{ij}}{\boldsymbol{\xi}_{ij}^T}=\boldsymbol{e}_{ij}^T\frac{\partial \boldsymbol{e}_{ij}}{\boldsymbol{\xi}_{ij}^T}=-\boldsymbol{e}_{ij}^T\frac{\partial\hat{\boldsymbol{u}}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)}{\boldsymbol{\xi}_{ij}^T}$$

$$\frac{\partial\hat{\boldsymbol{u}}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)}{\partial\boldsymbol{\xi}_{ij}^T}=\begin{bmatrix}\frac{\partial\hat{\boldsymbol{u}}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)}{\partial\boldsymbol{C}_j^T} & \frac{\partial\hat{\boldsymbol{u}}_{ij}\left(\boldsymbol{C}_j,\boldsymbol{X}_i\right)}{\boldsymbol{X}_i^T}\end{bmatrix}=\begin{bmatrix}\frac{\partial u_{ij}(\boldsymbol{C}_i,\boldsymbol{X}_j)}{\partial \boldsymbol{C}_j^T} & \frac{\partial u_{ij}(\boldsymbol{C}_i,\boldsymbol{X}_j)}{\partial \boldsymbol{X}_j^T}\\ \frac{\partial v_{ij}(\boldsymbol{C}_i,\boldsymbol{X}_j)}{\partial \boldsymbol{C}_j^T} & \frac{\partial v_{ij}(\boldsymbol{C}_i,\boldsymbol{X}_j)}{\partial \boldsymbol{X}_j^T}\end{bmatrix}$$

捆绑调整Bundle Adjustment

Coding-3:

仔细推导雅阁比矩阵，并完成task3/test_jacobian.cc中雅阁比矩阵的实现

理解->推导->实现== 真正掌握

双视角运动恢复结构

特征检测与匹配

↓

初始化

- 内参数的获取（Exif图像信息）
- 相机参数的恢复（本质矩阵分解）
- 三维点的坐标（三角量测）

↓

非线性优化(BA)

深蓝学院
shenlanxueyuan.com

三维建模课程地址

感谢各位聆听！

Thanks for Listening