深蓝学院
shenlanxueyuan.com

视觉SLAM进阶课程
从零开始手写VIO-第5期

第4讲 滑动窗口理论与实践 作业讲评

评讲人 吕华龙

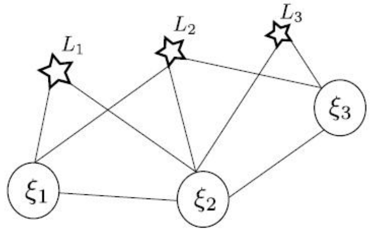
【作业】

深蓝学院 shenlanxueyuan.com

① 设某时刻，SLAM系统中相机和路标点的观测关系如下图所示，其中ξ表示相机姿态，L表示观测到的路标点。当路标点L表示在世界坐标系下时，第k个路标被第i时刻的相机观测到，重投影误差为$r(\xi_i, L_k)$。另外，相邻相机之间存在运动约束，如IMU或者轮速计等约束。

1. 绘制上述系统的信息矩阵 Λ

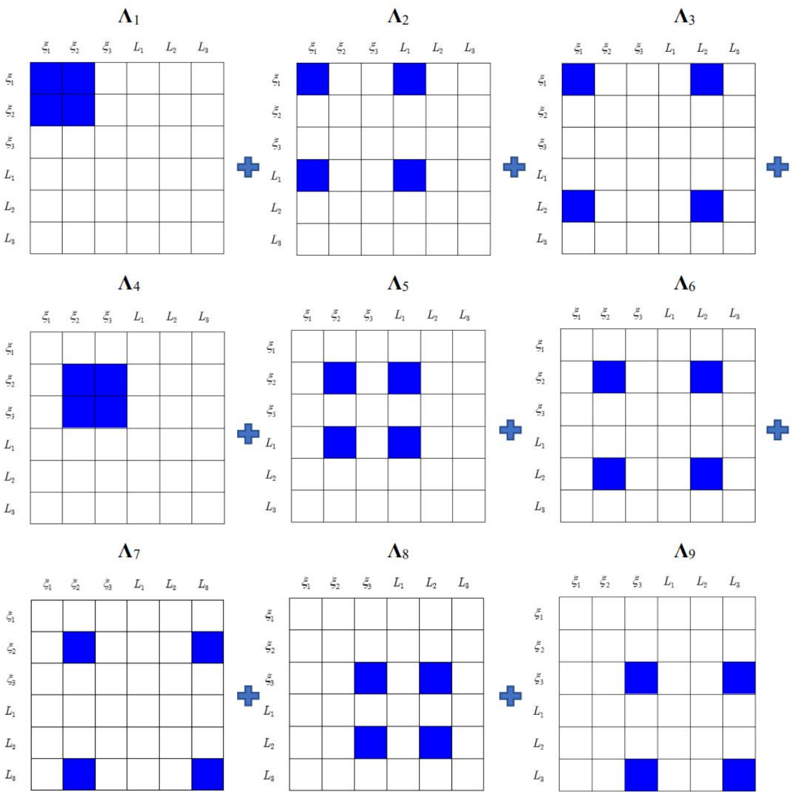
2. 绘制相机 ξ_1 被marg以后的信息矩阵 Λ'

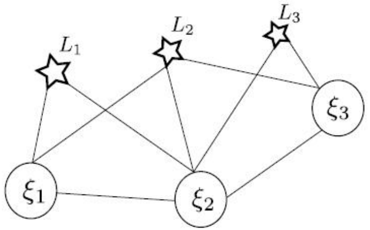


② 阅读《Relationship between the Hessian and Covariance Matrix for Gaussian RandomVariables》。证明信息矩阵和协方差的逆之间的关系。

③ 补充作业代码中单目Bundle Adjustment信息矩阵的计算，并输出正确的结果。正确的结果为：奇异值最后7维接近于0，表明零空间的维度为7

题目一



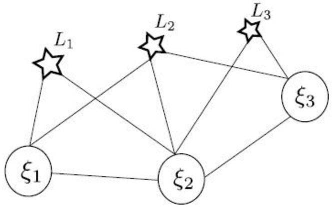


得到当前时刻系统的信息矩阵Λ

	ξ_1	ξ_2	ξ_3	L_1	L_2	L_3
ξ_1	3	1		1	1	
ξ_2	1	5	1	1	1	1
ξ_3		1	3		1	1
L_1	1	1		2		
L_2	1	1	1		3	
L_3		1	1			2

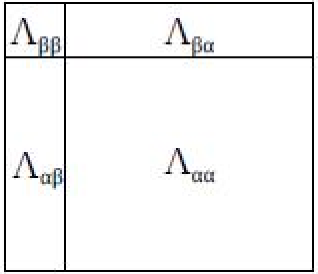
From优秀作业-刘晓燕

题目一



	ξ_1	ξ_2	ξ_3	L_1	L_2	L_3
ξ_1	3	1		1	1	
ξ_2	1	5	1	1	1	1
ξ_3		1	3		1	1
L_1	1	1		2		
L_2	1	1	1		3	
L_3		1	1			2

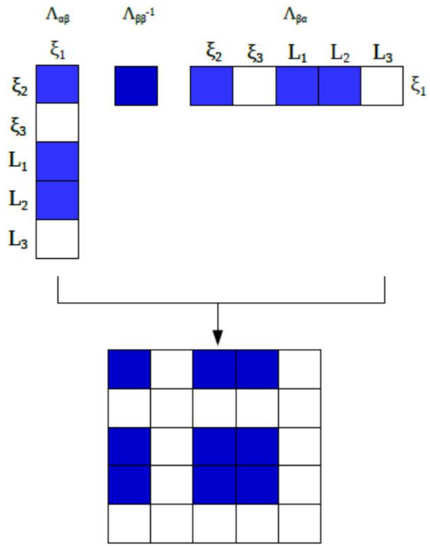
$$\xrightarrow{\Lambda_{\alpha\alpha}-\Lambda_{\alpha\beta}\Lambda_{\beta\beta}^{-1}\Lambda_{\beta\alpha}}$$



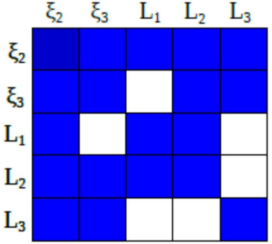
Λ_{aa}

	ξ_2	ξ_3	L_1	L_2	L_3
ξ_2	5	1	1	1	1
ξ_3	1	3		1	1
L_1	1		2		
L_2	1	1		3	
L_3	1	1			2

—



$p(\xi_2, \xi_3, L_1, L_2, L_3)$



得到边缘化之后的信息矩阵：

题目二

阅读《Relationship between the Hessian and Covariance Matrix for Gaussian RandomVariables》证明信息矩阵和协方差的逆之间的关系。

具体的证明过程如下：

假设高斯随机向量θ，均值向量为θ^*，协方差矩阵为Σ_θ，则其联合概率密度函数为：

$$p(\theta)=(2\pi)^{-\frac{N_\theta}{2}}|\Sigma_\theta|^{-\frac{1}{2}}\exp\left[-\frac{1}{2}(\theta-\theta^*)^{\mathrm{T}}\Sigma_\theta^{-1}(\theta-\theta^*)\right] \tag{1}$$

目标函数可以定义为该联合概率密度函数的负对数：

$$J(\theta)=-\ln p(\theta)=\frac{N_\theta}{2}\ln 2\pi+\frac{1}{2}\ln|\Sigma_\theta|+\frac{1}{2}(\theta-\theta^*)^{\mathrm{T}}\Sigma_\theta^{-1}(\theta-\theta^*) \tag{2}$$

从上式可以看出，该目标函数是变量θ的二次函数，通过对θ_l和$\theta_{l'}$求偏导，可以得到在(l, l')上的Hessian矩阵：

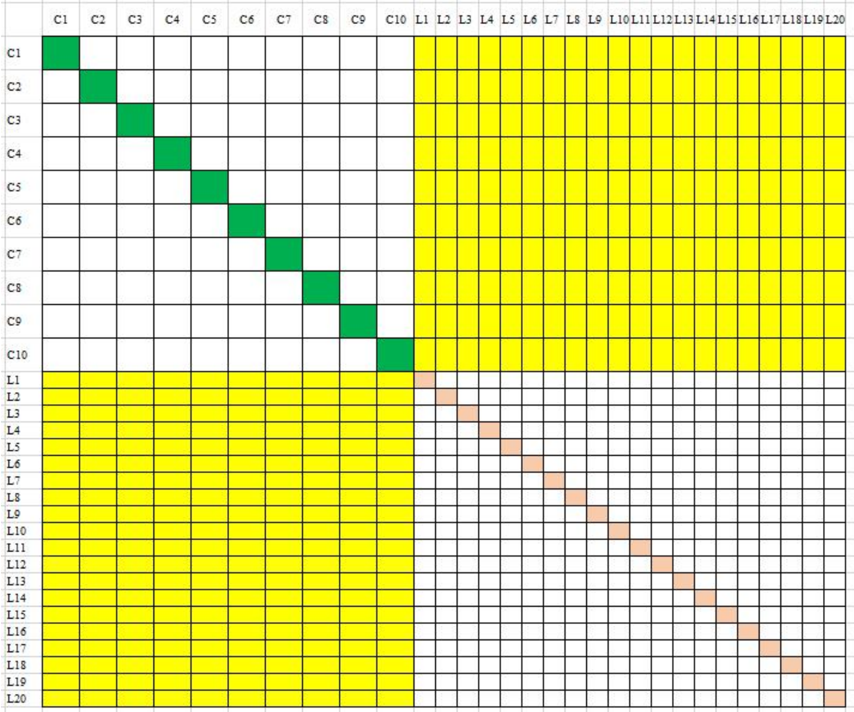
$$H^{(l,l')}(\theta^*)=\left.\frac{\partial^2 J(\theta)}{\partial\theta_l\partial\theta_{l'}}\right|_{\theta=\theta^*}=\left(\Sigma_\theta^{-1}\right)^{(l,l')} \tag{3}$$

所以，Hessian矩阵等于协方差矩阵的逆。

$$H(\theta^*)=\Sigma_\theta^{-1} \tag{4}$$

题目三

补充作业代码中单目Bundle Adjustment信息矩阵的计算，验证信息矩阵零空间维度为7.



程序中仿真的单目模型：

10相机Pose,

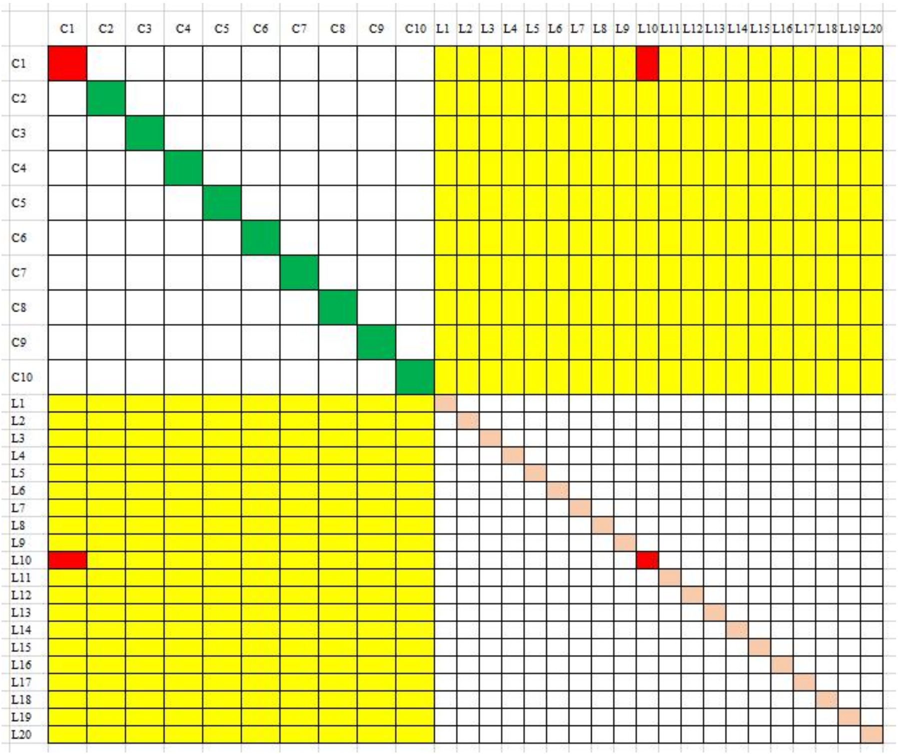
20个Feature Point,

并且每个相机都能观测到所以所有路标点。

相机Pose根据曲线模拟生成，

3D landmark在Pose周围随机生成。

题目三



对于其中一项残差，$r(\xi_i, p_j)$其中$i \in (1 \sim 10), j \in (1 \sim 20)$对于H的贡献可分成四个小块：

$$J_{T_i}^{T} J_{T_i} (6*6阶)$$

$$J_{T_i}^{T} J_{P_j} (6*3阶)$$

$$J_{P_j}^{T} J_{T_i} (3*6阶)$$

$$J_{P_j}^{T} J_{P_j} (3*3阶)$$

题目三

对于残差r(ξ_i, p_j)的Jacibian　$J_i = (\frac{\partial r(\xi_i, p_j)}{\partial \xi_i}, 0, 0 \ldots\ldots \frac{\partial r(\xi_i, p_j)}{\partial p_j}, 0,0,0,0)$

残差对于Pose的导数：$J_{T_i} = \frac{\partial r(\xi_i, p_j)}{\partial \delta \xi_i} = \lim_{\partial \xi \to 0} \frac{r(\xi_i \oplus \partial \xi)}{\partial \delta \xi} = \frac{\partial \mathrm{r}}{\partial P'} \frac{\partial P'}{\partial \delta \xi}$

残差对于landmark的导数：$J_{P_j} = \frac{\partial r(\xi_i, p_j)}{\partial \mathrm{P}} = \frac{\partial \mathrm{r}}{\partial P'} \frac{\partial P'}{\partial P}$

$$\frac{\partial \mathrm{r}}{\partial P'} = \begin{pmatrix} \frac{\partial u}{\partial X'} & \frac{\partial u}{\partial Y'} & \frac{\partial u}{\partial Z'} \\ \frac{\partial v}{\partial X'} & \frac{\partial v}{\partial Y'} & \frac{\partial v}{\partial Z'} \end{pmatrix} = -\begin{pmatrix} f_x \frac{1}{Z'}, 0, -f_x \frac{\mathrm{X}'}{Z'^2} \\ 0, f_y \frac{1}{Z'}, -f_y \frac{\mathrm{Y}'}{Z'^2} \end{pmatrix}$$

$$\begin{aligned} \frac{\partial \mathrm{P}'}{\partial \delta \xi} &= \lim_{\delta \xi \to 0} \frac{\partial r(\xi \oplus \delta \xi)}{\delta \xi} = \lim_{\delta \xi \to 0} \frac{\exp(\xi^\wedge)\exp(\delta \xi^\wedge)P - \exp(\xi^\wedge)P}{\delta \xi} \\ &= \lim_{\delta \xi \to 0} \frac{\exp(\xi^\wedge)(I + \delta \xi^\wedge)P - \exp(\xi^\wedge)P}{\delta \xi} \\ &= \lim_{\delta \xi \to 0} \frac{\exp(\xi^\wedge)\delta \xi^\wedge P}{\delta \xi} \\ &= \lim_{\delta \xi \to 0} - \frac{\exp(\xi^\wedge)P^\wedge \delta \xi}{\delta \xi} \\ &= -\exp(\xi^\wedge)P^\wedge \\ &= \begin{pmatrix} -(RP+t)^\wedge & I \\ 0^T & 0^T \end{pmatrix} \end{aligned}$$

$$J_{T_i} = \frac{\partial \mathrm{r}}{\partial P'} \frac{\partial P'}{\partial \delta \xi} = \begin{pmatrix} -\frac{xyf_x}{z^2} & f_x + \frac{x^2}{z^2} f_x & -\frac{y}{z} f_x & \frac{f_x}{z} & 0 & -x\frac{f_x}{z^2} \\ =(f_y + \frac{y^2}{z^2} f_y) & \frac{xyf_y}{z^2} & \frac{x}{z} f_y & 0 & \frac{f_y}{z} & -y\frac{f_y}{z^2} \end{pmatrix}$$

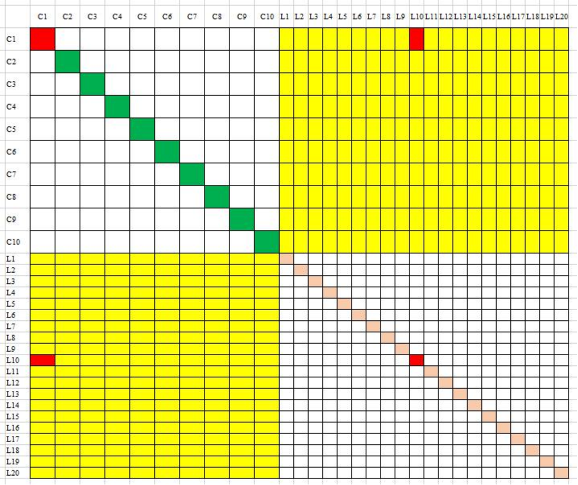
$$J_{P_j} = \frac{\partial r(\xi_i, p_j)}{\partial \mathrm{P}} = \frac{\partial \mathrm{r}}{\partial P'} \frac{\partial P'}{\partial P}$$

$$\frac{\partial \mathrm{r}}{\partial P'} = \begin{pmatrix} \frac{\partial u}{\partial X'} & \frac{\partial u}{\partial Y'} & \frac{\partial u}{\partial Z'} \\ \frac{\partial v}{\partial X'} & \frac{\partial v}{\partial Y'} & \frac{\partial v}{\partial Z'} \end{pmatrix} = -\begin{pmatrix} f_x \frac{1}{Z'}, 0, -f_x \frac{\mathrm{X}'}{Z'^2} \\ 0, f_y \frac{1}{Z'}, -f_y \frac{\mathrm{Y}'}{Z'^2} \end{pmatrix}$$

$$\mathrm{P}' = R_{wc}^{-1}(P_w - t_{wc})$$

$$\frac{\partial \mathrm{P}'}{\partial P} = -R_{wc}^{-1}$$

题目三




```
for (int i = 0; i < poseNums; ++i) {
    Eigen::Matrix3d Rcw = camera_pose[i].Rwc.transpose();
    Eigen::Vector3d Pc = Rcw * (Pw - camera_pose[i].twc);

    double x = Pc.x();
    double y = Pc.y();
    double z = Pc.z();
    double z_2 = z * z;
    Eigen::Matrix<double,2,3> jacobian_uv_Pc;
    jacobian_uv_Pc<< fx/z, 0 , -x * fx/z_2,
                     0, fy/z, -y * fy/z_2;
    Eigen::Matrix<double,2,3> jacobian_Pj = jacobian_uv_Pc * Rcw;
    Eigen::Matrix<double,2,6> jacobian_Ti;

    jacobian_Ti << -x* y * fx/z_2, (1+ x*x/z_2)*fx, -y/z*fx, fx/z, 0 , -x * fx/z_2,
                    -(1+y*y/z_2)*fy, x*y/z_2 * fy, x/z * fy, 0,fy/z, -y * fy/z_2;

    H.block(i*6,i*6,6,6) += jacobian_Ti.transpose() * jacobian_Ti;
    H.block(j*3 + 6*poseNums,j*3 + 6*poseNums,3,3) += jacobian_Pj.transpose() *jacobian_Pj ;
    H.block(i*6,j*3 + 6*poseNums, 6,3) += jacobian_Ti.transpose() * jacobian_Pj;
    H.block(j*3 + 6*poseNums,i*6 , 3,6) += jacobian_Pj.transpose() * jacobian_Ti;
}
```

$J_{T_i}{}^T J_{T_i}$(6*6阶) $J_{T_i}{}^T J_{P_j}$（6*3阶）

$J_{P_j}{}^T J_{T_i}$（3*6阶） $J_{P_j}{}^T J_{P_j}$(3*3阶)

对信息矩阵进行 SVD 分解后，

发现特征值的最后 7 维接近于零，

即表示原始的 H 矩阵零空间维度为7维。

3.21708e-17
2.06732e-17
1.43188e-17
7.66992e-18
6.08423e-18
6.05715e-18
3.94363e-18

感谢各位聆听！

Thanks for Listening