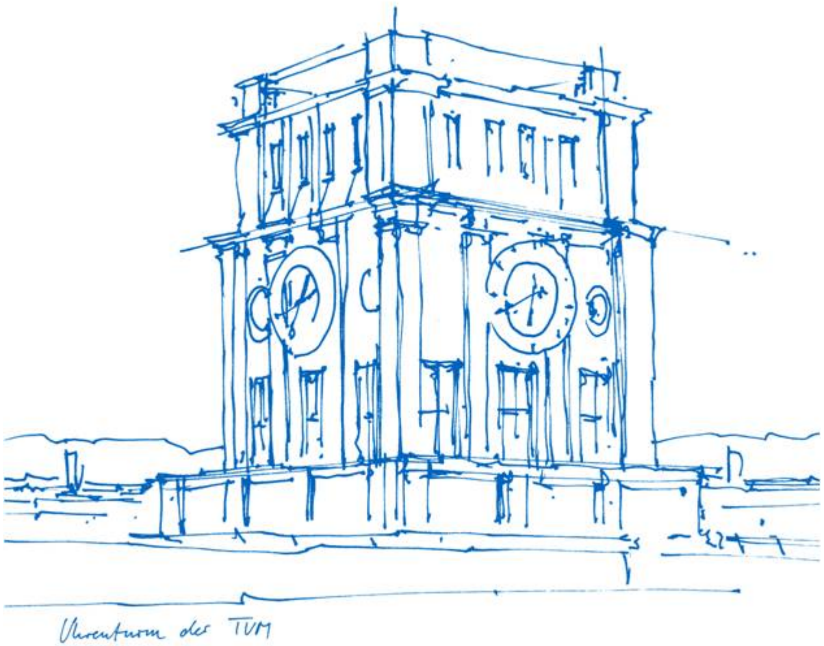
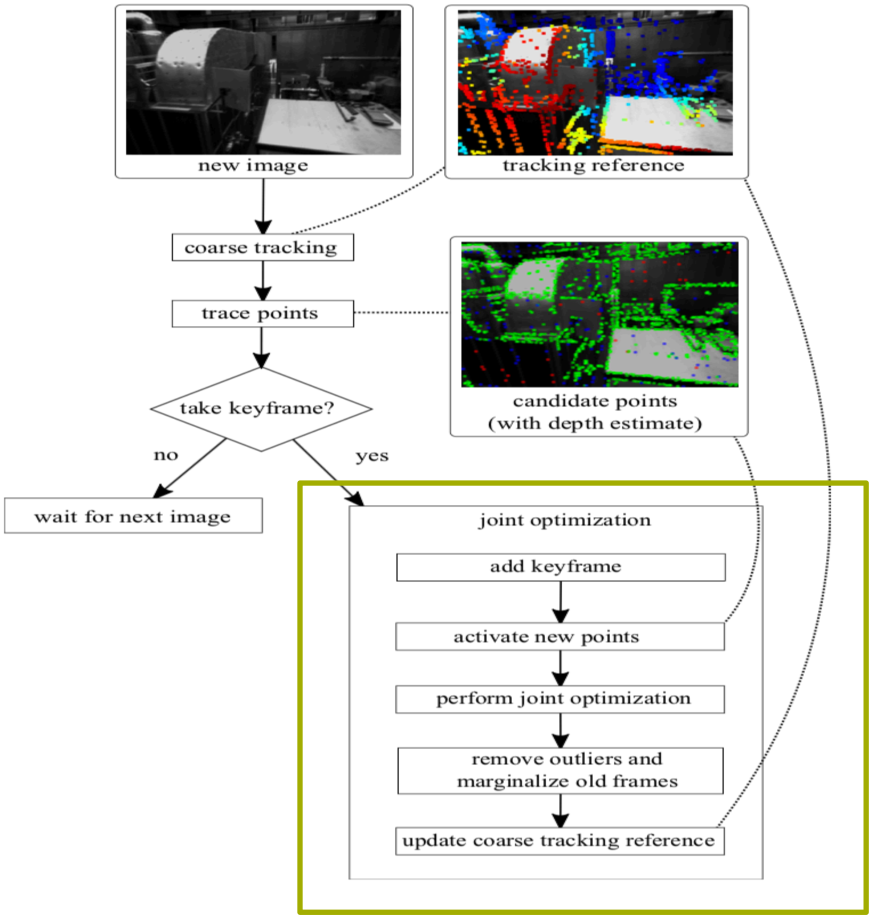
DSO 优化部分讲解

曲星威

2019.10.13



DSO 流程框架



论文：2.3内容

1. Weighted Gauss-Newton 优化方法
2. Marginalization 介绍
3. First Estimate Jacobian介绍

Weighted Gauss-Newton Method

$$\hat{\theta} = \arg\min_{\theta} r^T(\hat{\theta})\Sigma_r^{-1} r(\hat{\theta})$$

$$r(\hat{\theta}) = \left\| I_{ref} - I_{target} \right\|_2$$

$$r(\theta) = r(\hat{\theta}) + J(\hat{\theta} - \theta)$$

$$\frac{1}{2}\frac{\partial(r^T(\theta)\Sigma_r^{-1} r(\theta))}{\partial\theta^T} = J^T\hat{\theta}\Sigma_r^{-1}(J_{\hat{\theta}}\widehat{\Delta\theta} + r(\hat{\theta}) - r(\theta)) = 0$$

Normal equation:

$$M\widehat{\Delta\theta} = m$$

$$M = J_{\hat{\theta}}^T\Sigma_r^{-1}J_{\hat{\theta}}\ ,\ m = J_{\hat{\theta}}^T\Sigma_r^{-1}(r(\theta) - r(\hat{\theta}))$$

We suppose that each state with covariance 1, According to Covariance Propagation

$$\Sigma_r = J_r\Sigma_{\hat{\theta}}J_r^T \approx J_r\Sigma_{\theta}J_r^T$$

假设$\hat{\theta}$ noise
是服从高斯分布的

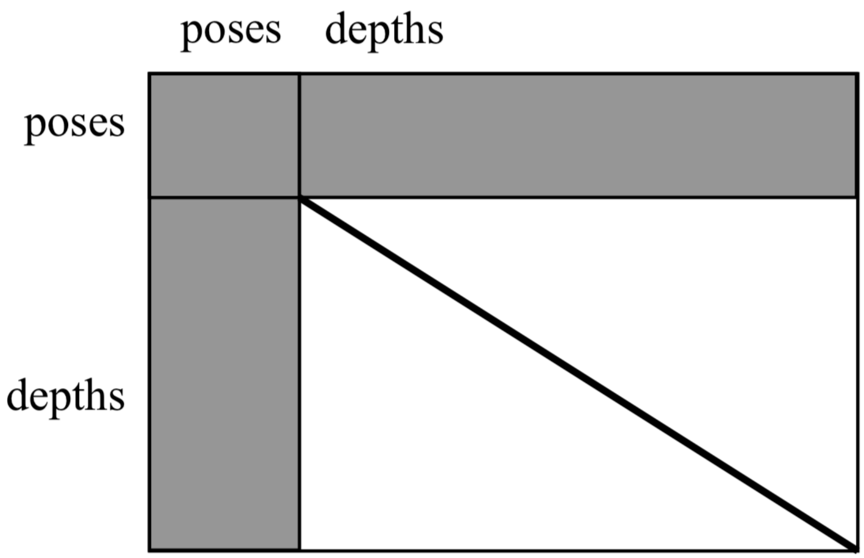
Schur Complementation

$$x + y = 1$$

$$x - y = 1$$

$$x = y + 1 \;\rightarrow\; 2y + 1 = 1 \;\rightarrow\; y$$

$$\begin{bmatrix} \mathbf{H}_{\alpha\alpha} & \mathbf{H}_{\alpha\beta} \\ \mathbf{H}_{\beta\alpha} & \mathbf{H}_{\beta\beta} \end{bmatrix} \begin{bmatrix} \delta \boldsymbol{x}_{\alpha} \\ \delta \boldsymbol{x}_{\beta} \end{bmatrix} = \begin{bmatrix} \boldsymbol{b}_{\alpha} \\ \boldsymbol{b}_{\beta} \end{bmatrix}$$



$$\widehat{\mathbf{H}_{\alpha\alpha}} = \mathbf{H}_{\alpha\alpha} - \mathbf{H}_{\alpha\beta}\mathbf{H}_{\beta\beta}^{-1}\mathbf{H}_{\beta\alpha}$$

$$\widehat{\boldsymbol{b}_{\alpha}} = \boldsymbol{b}_{\alpha} - \mathbf{H}_{\alpha\beta}\mathbf{H}_{\beta\beta}^{-1}\boldsymbol{b}_{\beta}$$

$$\delta \boldsymbol{x}_{\alpha} = \widehat{\mathbf{H}_{\alpha\alpha}}^{-1} \cdot \widehat{\boldsymbol{b}_{\alpha}}$$

$$\mathbf{H}_{\beta\alpha}\delta \boldsymbol{x}_{\alpha} + \mathbf{H}_{\beta\beta}\delta \boldsymbol{x}_{\beta} = \boldsymbol{b}_{\beta} \Leftrightarrow \delta \boldsymbol{x}_{\beta} = \mathbf{H}_{\beta\beta}^{-1}\left(\boldsymbol{b}_{\beta} - \mathbf{H}_{\beta\alpha}\delta \boldsymbol{x}_{\alpha}\right)$$

Schur Complementation and Marginlization

$$\widehat{\mathbf{H}_{\alpha\alpha}} = \mathbf{H}_{\alpha\alpha} - \mathbf{H}_{\beta\alpha}\mathbf{H}_{\beta\beta}^{-1}\mathbf{H}_{\alpha\beta} + \mathbf{H}_{\mathrm{L}} + \mathbf{H}_{\mathrm{M}}$$

DSO维护的矩阵

原始 Camera Hessian

HSC

Prior

Marginalization矩阵

$\widehat{\mathbf{H}_{\alpha\alpha}}$

C1		
	C2	
		C3

C1	D	
D	C2	
		C4

C1	D	
D	C2	
		C4

$\mathbf{H}_M$

0	0
0	0

Margin Points

M1	D	D
D	M2	D
D	D	M3

Margin Frame

M1	D
D	M2

Margin Points

M1	D	D
D	M2	D
D	D	M4

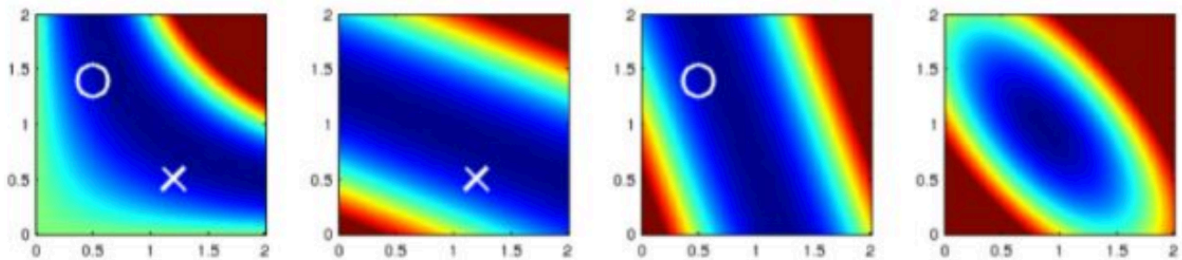
Margin Frame

C2	D2
D2	C4

First Estimate Jacobian

Windowed, real-time optimization: Consistency.

(for now, let's assume we have initializations, and know which points to use and where they are visible.)



$$E = E_1 + E_2 = (xy-1)^2 + (xy-1)^2$$

$E'_1 = E_1$ Lin. around (0.5, 1.4)

$E'_2 = E_2$ Lin. around (1.2, 0.5)

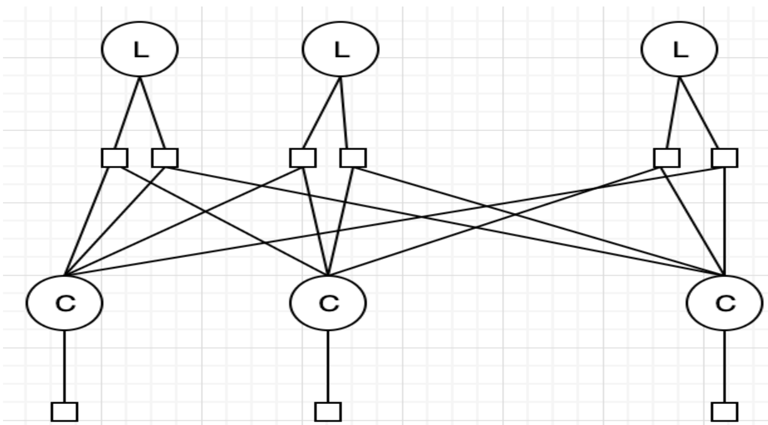
$E' = E'_1 + E'_2$

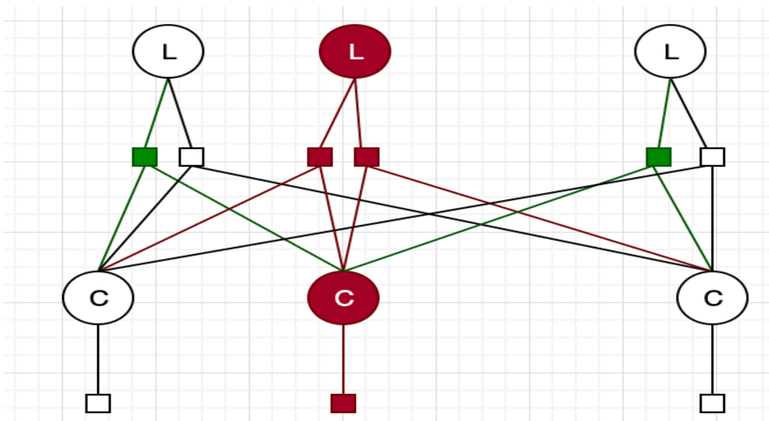
$\boxed{\textit{nullspace disappears!}}$

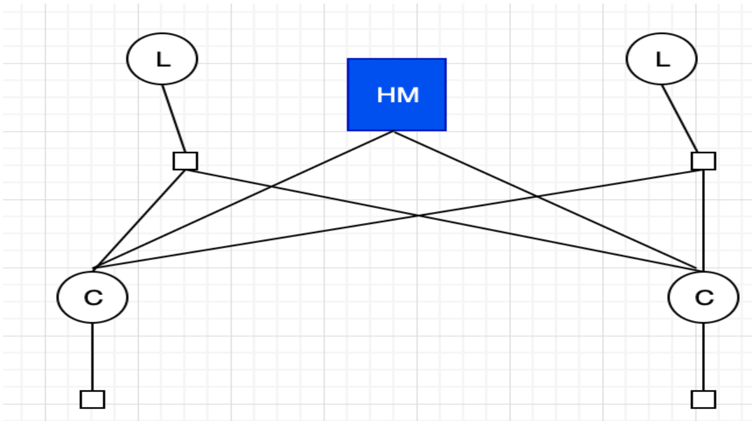
$$xy = 1 \qquad -2.8x + 0.3 = y \qquad -x + 0.3 - 2.4y = 0$$

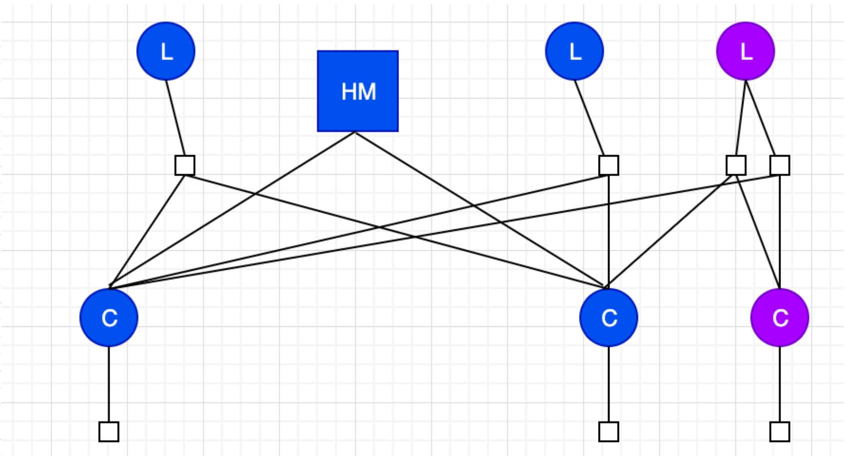
never combine linearizations around different linearization points,
especially in the presence of non-linear nullspaces!
It will render unobservable dimensions observable, and corrupt the system.

Why FEJ









First Estimate Jacobian

在paper里， Jakob说所有的depth，以及poses在加入窗口的时候都要fix住他的线性点，但其实不是 参考论文 “Improving the Accuracy of EKF-Based Visual-Inertial Odometry” 里面其实只有translation 会影响0空间，Rotation 和 depth都不会影响。在代码实现中 depth也一直进行update。

https://intra.ece.ucr.edu/~mourikis/papers/Li2012-ICRA.pdf

Reference

L. V. Stumberg, V. Usenko, and D. Cremers. Direct sparse visual-inertial odometry using dynamic marginalization. In *2018 IEEE International Conference on Robotics and Au- tomation (ICRA)*. IEEE, may 2018.
Li M, Mourikis A I. Improving the accuracy of EKF-based visual-inertial odometry[C]//2012 IEEE International Conference on Robotics and Automation. IEEE, 2012: 828-835.
Engel J, Koltun V, Cremers D. Direct sparse odometry[J]. IEEE transactions on pattern analysis and machine intelligence, 2017, 40(3): 611-625.