



CloudMinds

# 主流VIO技术综述

崔华坤 2019.1.16

# 目录

一、BA和因子图

二、VIO分类

三、各种主流VIO

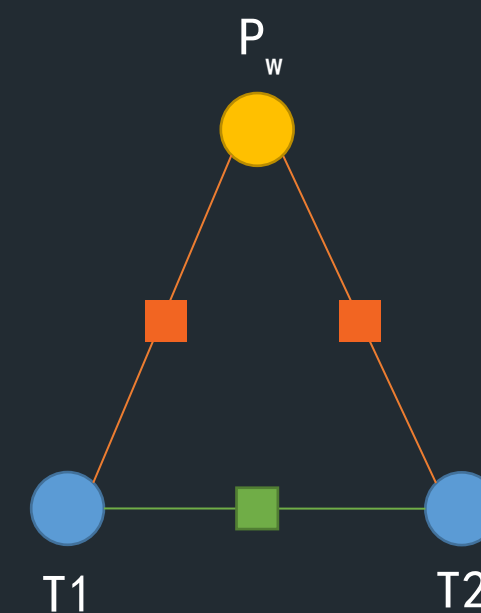
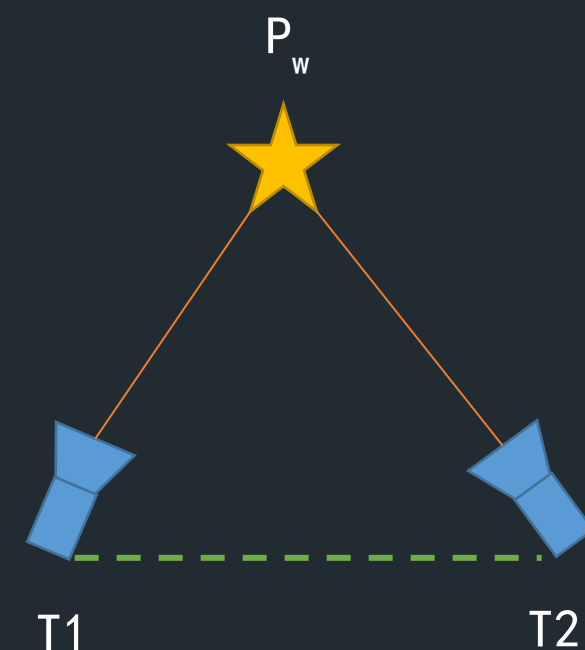
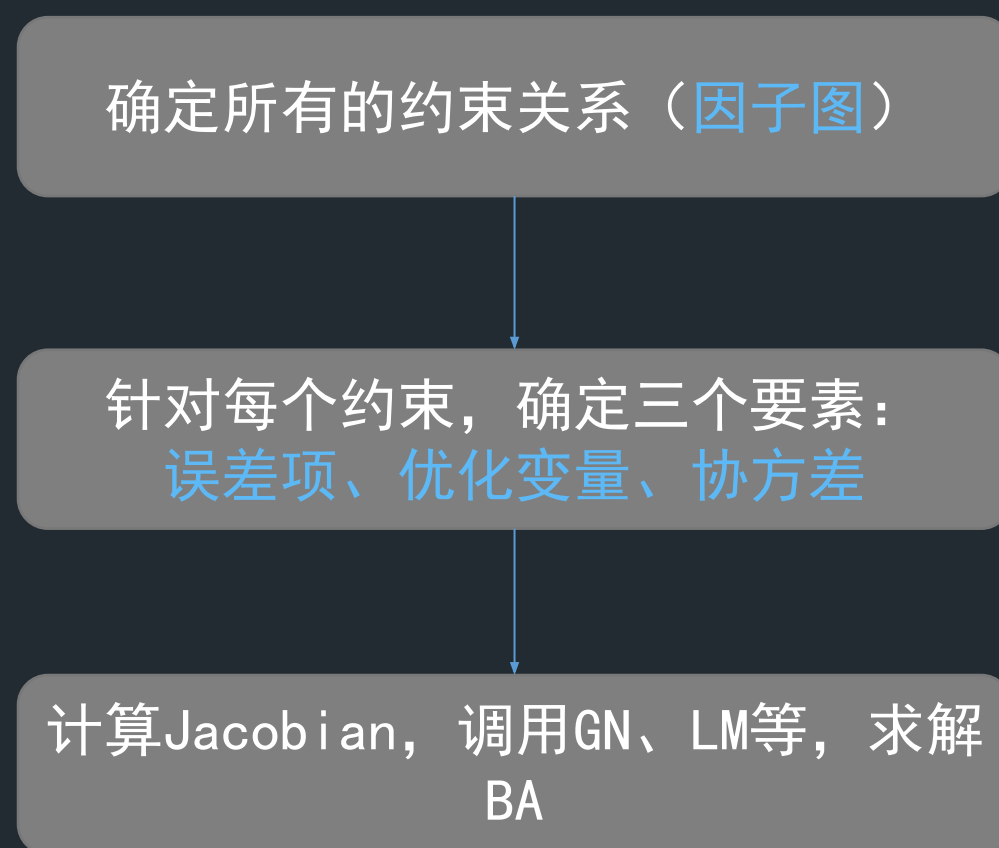
- MSCKF
- ROVIO
- VINS
- VI-ORB
- ICE-BA

**定义：**通过优化相机位姿和特征点的空间位置，使得每条光束都能打到光心；

**作用：**BA是SLAM后端的核心，SLAM其实就是一个BA问题，理解BA对于分析现有问题和建模新问题等都很有帮助；但是理解BA并不表示SLAM效果就好，SLAM最终结果的好坏很大取决于数据关联的好坏；

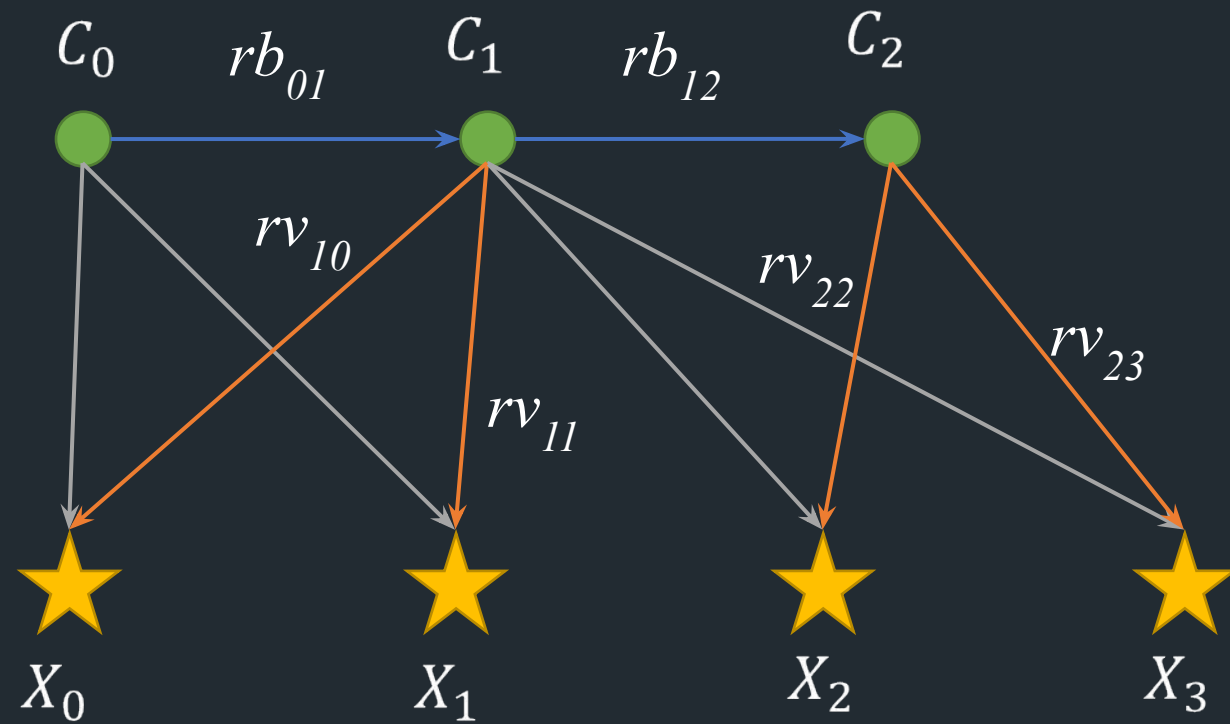
**核心：**误差项、优化变量、协方差

**难点：**李群、李代数、四元数的Jacobian，一通则百通，也可借鉴g2o写好的点和边，但要注意坐标系转换关系。





# 图解BA的求解过程



	T0	M0	T1	M1	T2	M2	x0	x1	x2	x3
rv10										
rv11										
rv22										
rv23										
rb01										
rb12										

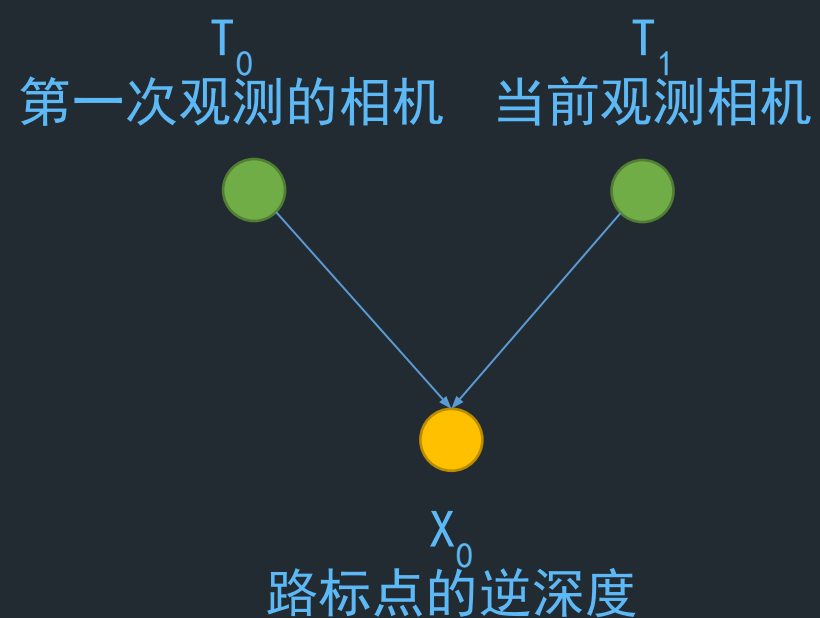
Jacob i an

	T0	M0	T1	M1	T2	M2	x0	x1	x2	x3	b
T0											
M0											
T1											
M1											
T2											
M2											
x0											
x1											
x2											
x3											

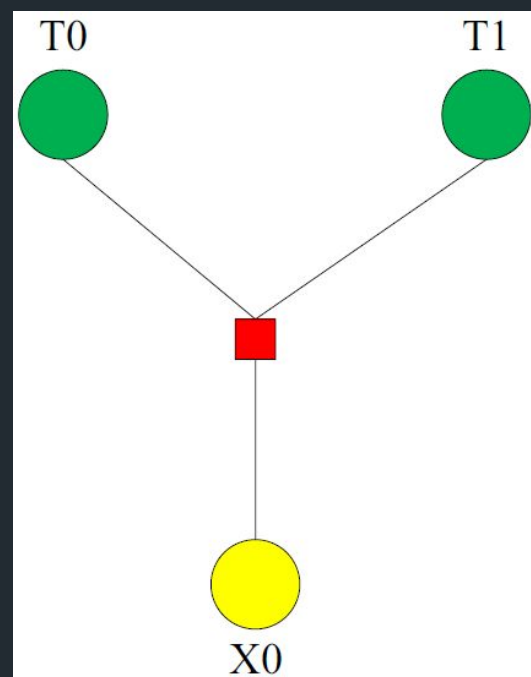
$$H = b \delta x$$

# 因子图

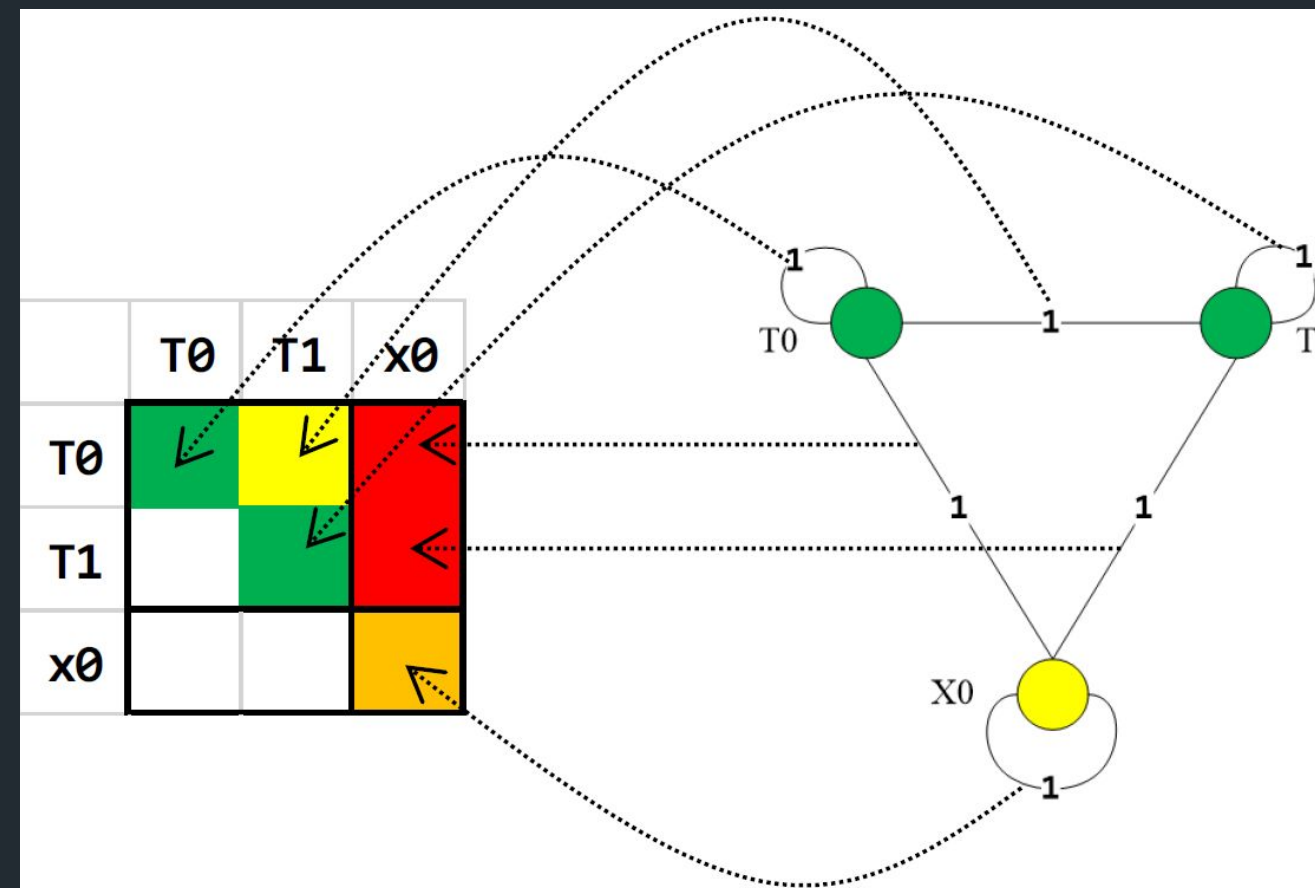
因子图中方块表示因子，圆圈表示优化变量。一个因子表示一个约束，即表示一个误差项，与之相连的圆圈是与该误差项相关的优化变量。



视觉约束：重投影误差



视觉因子



因子图与Hession的对应关系

$$H_{T_0 T_0} = \left( \frac{\partial e}{\partial T_0} \right)^T P^{-1} \left( \frac{\partial e}{\partial T_0} \right)$$

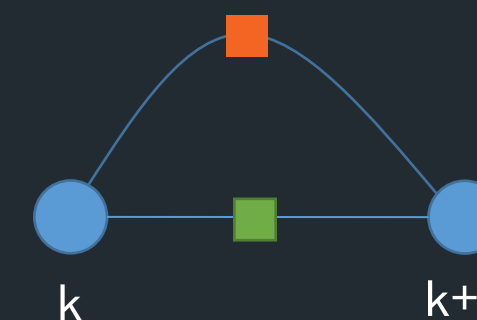
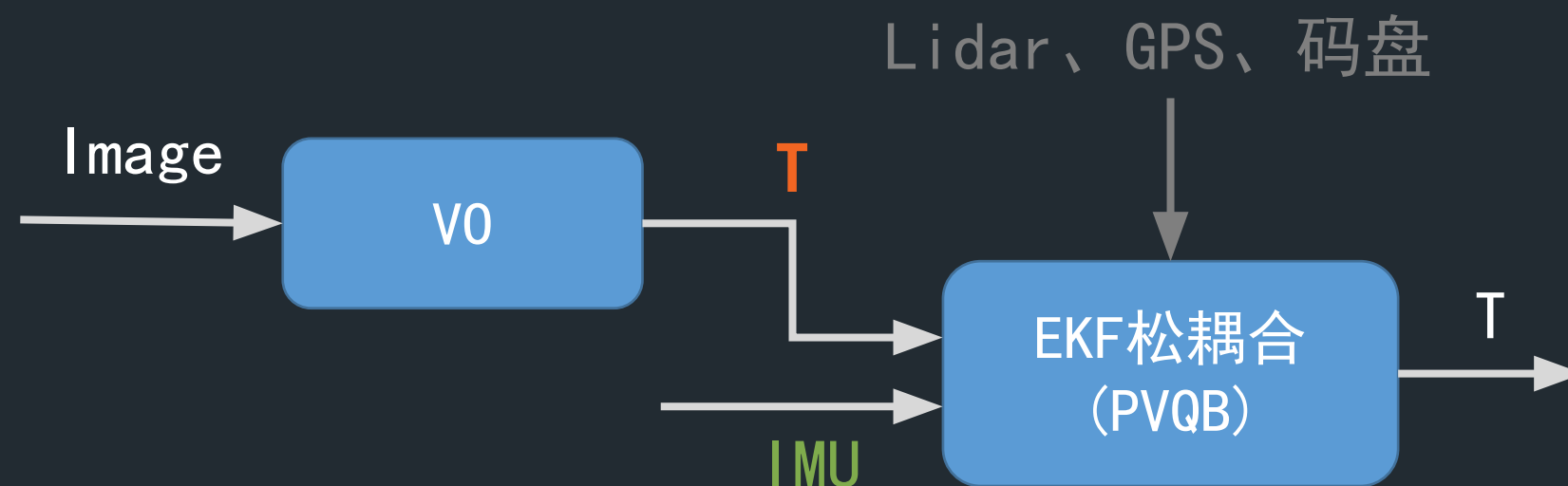


## VIO分类:

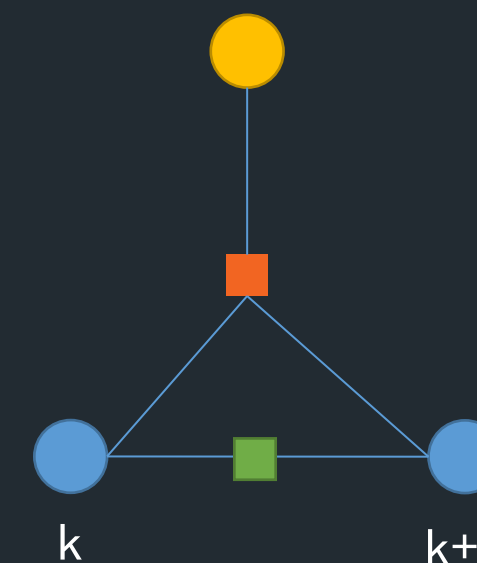
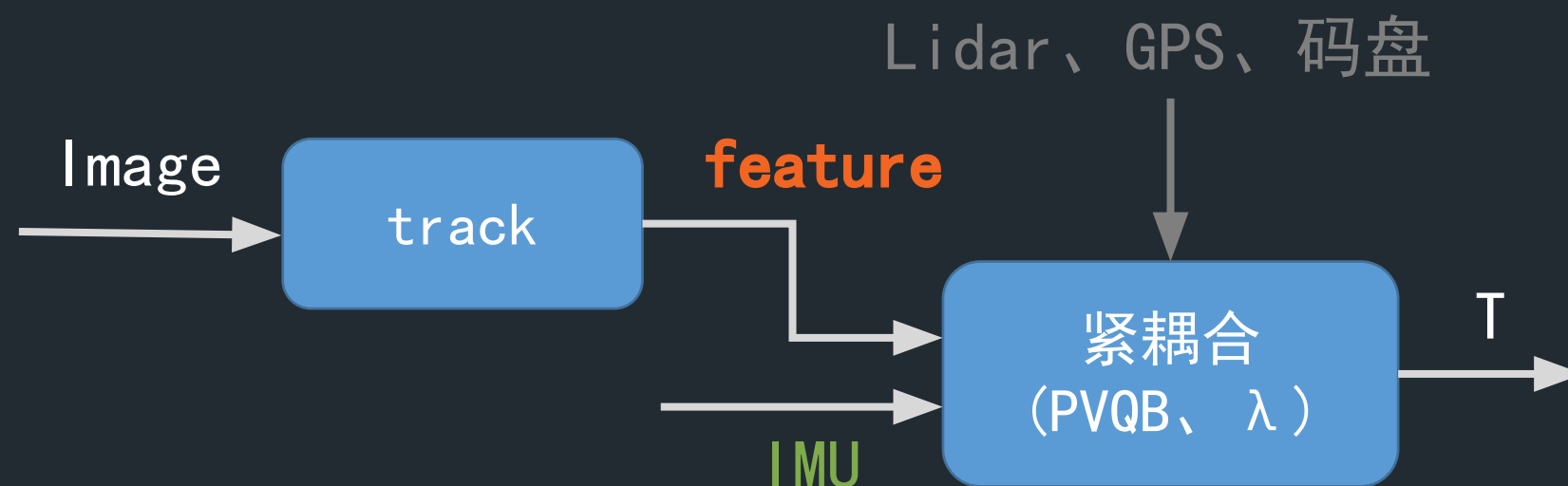
将视觉约束 加入到联合优化是紧耦合

将视觉约束后的位姿加入到联合优化是松耦合

松耦合



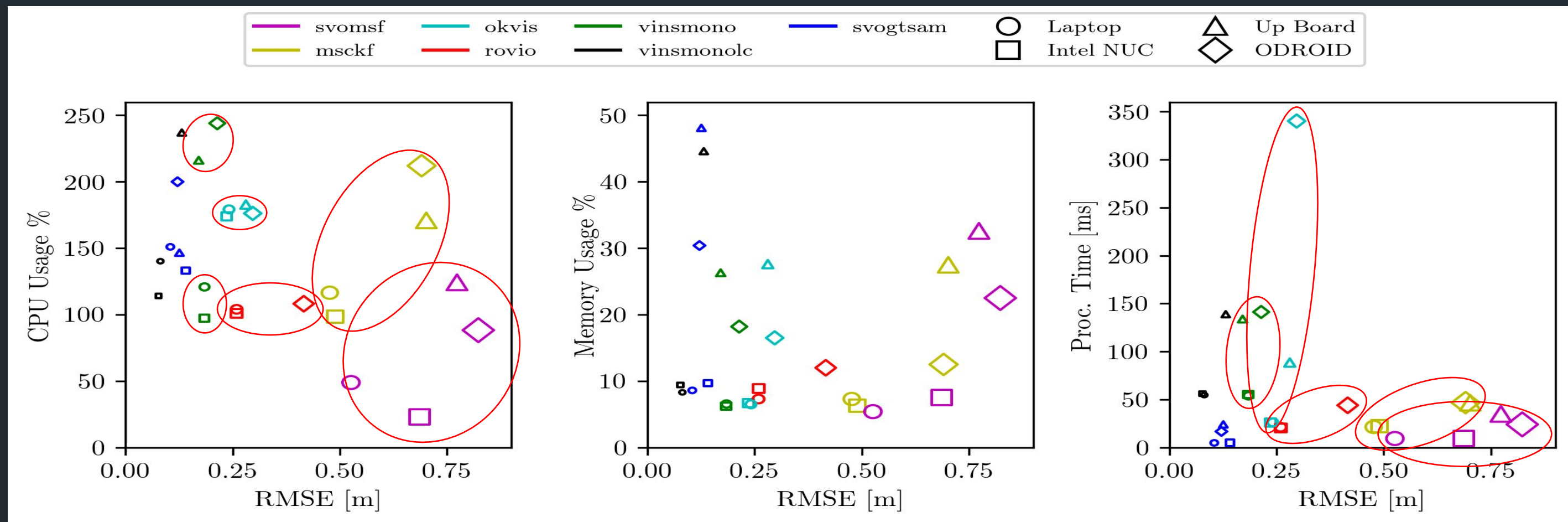
紧耦合





## 主流VIO

VIO框架	耦合方案	后端方案	前端	视觉误差	初始化	回环	精度	效率
MSF	松耦合	滤波EKF	\	\	\	\	最差1	最快5
MSCKF	紧耦合	滤波EKF	fast+光流	重投影	静止	无	较差2	较快4
ROVIO	紧耦合	滤波IEKF	fast+光度	光度	静止	无	一般3	较快4
OKVIS	紧耦合	优化	fast+brisk	重投影	静止	无	较好4	最慢1
VINS	紧耦合	优化	fast+光流	重投影	动态	有	最好5	较慢2
VI-ORB	紧耦合	优化	orb	重投影	动态	有	\	\
ICE-BA	紧耦合	优化	fast+光流	重投影	静止	无	\	\



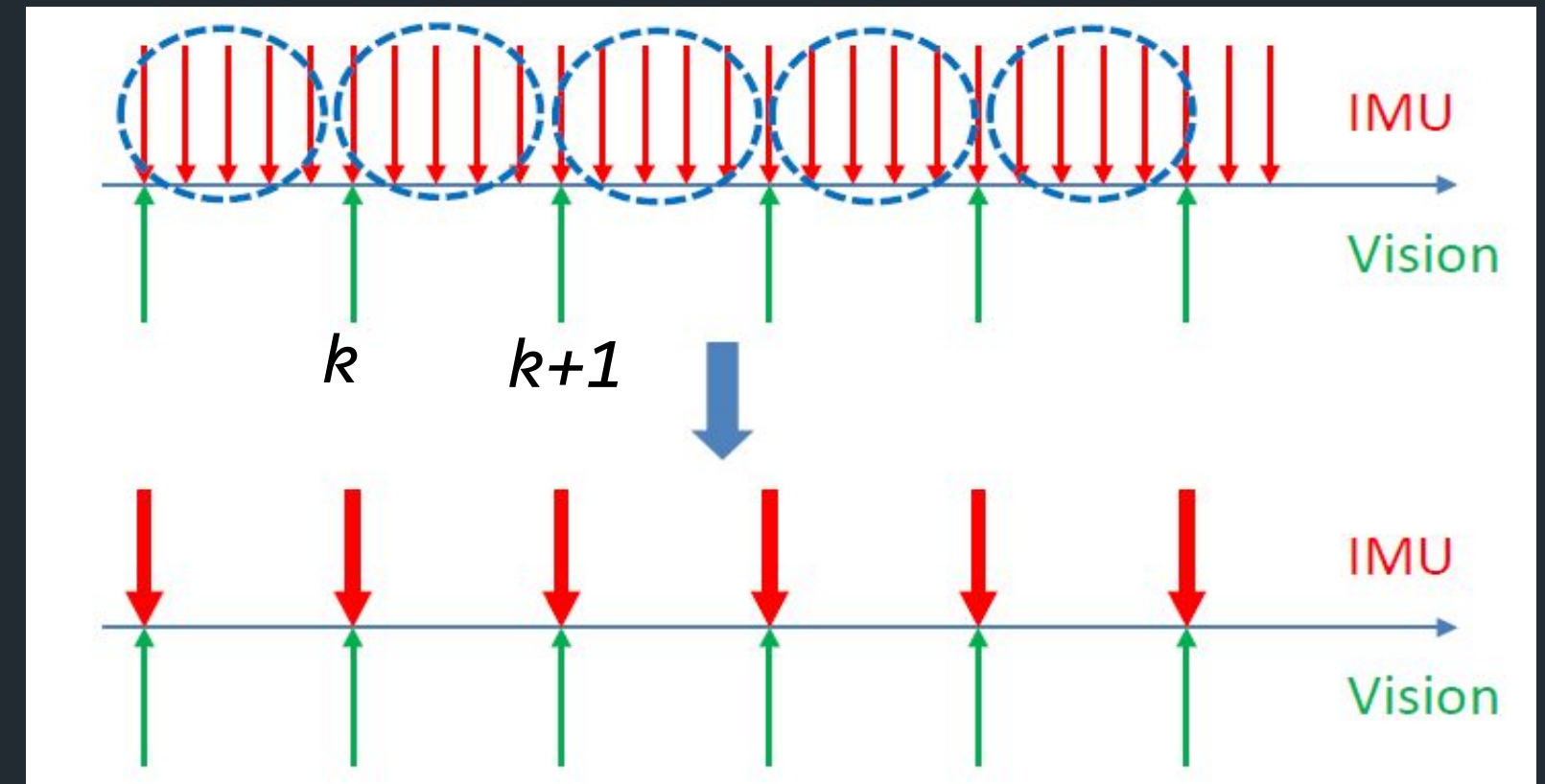


## 滤波vs优化

1. 滤波相当于单次优化，区别是滤波仅考虑上一帧的影响，而优化则考虑所有帧的影响；
2. 滤波比单次优化的精度高，因滤波是通过协方差来计算增量，而优化是GN、LM来计算；
3. 多次优化精度比滤波高，但效率低于滤波，因优化可以迭代多次，不断优化线性化点，使误差最小。

## IMU预积分

1. 积分下一个时刻的PVQ作为视觉初始值；
2. 预积分相邻帧的PVQ变化量，作为IMU约束；
3. 计算IMU的协方差，和Jacobian。



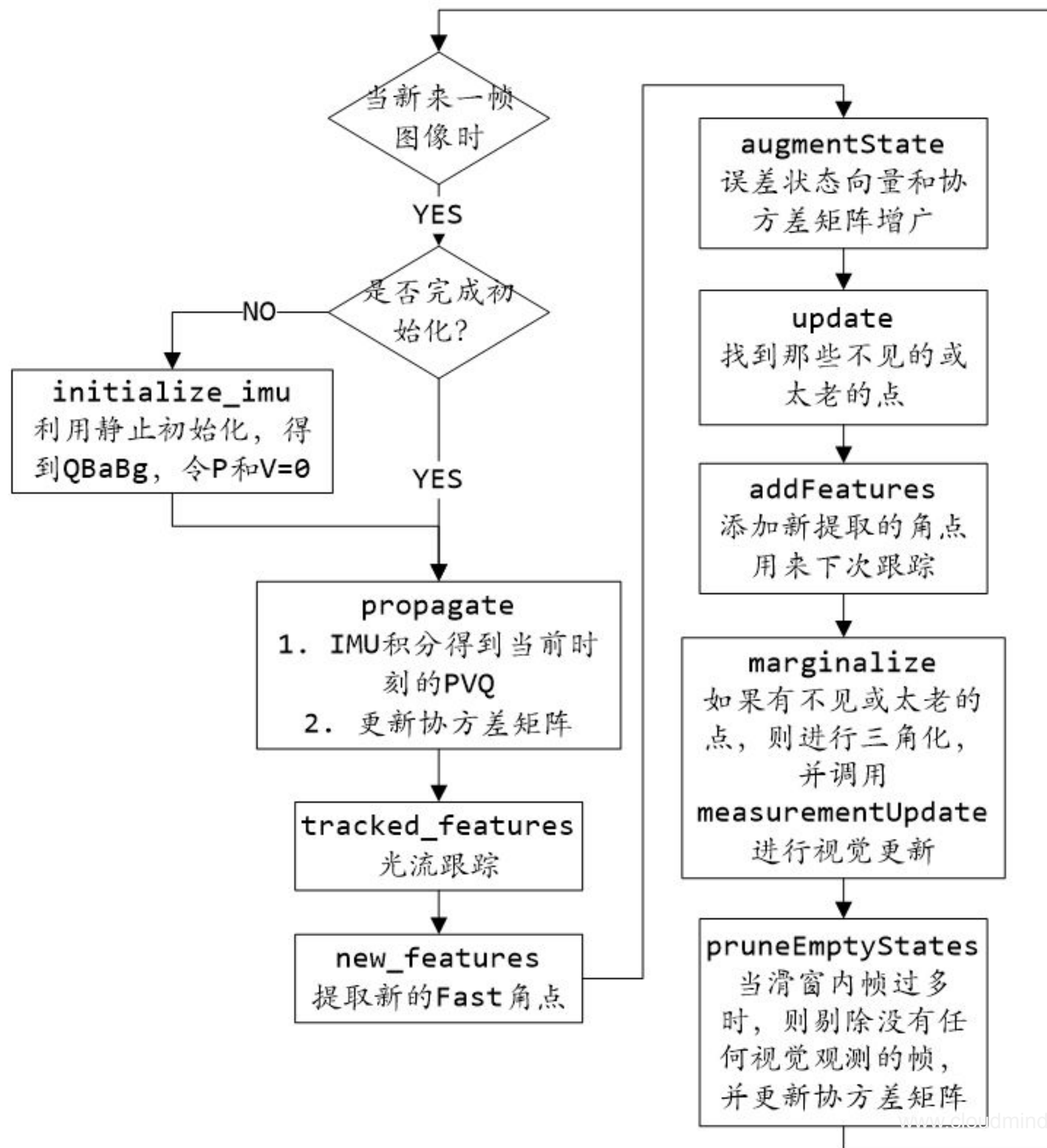


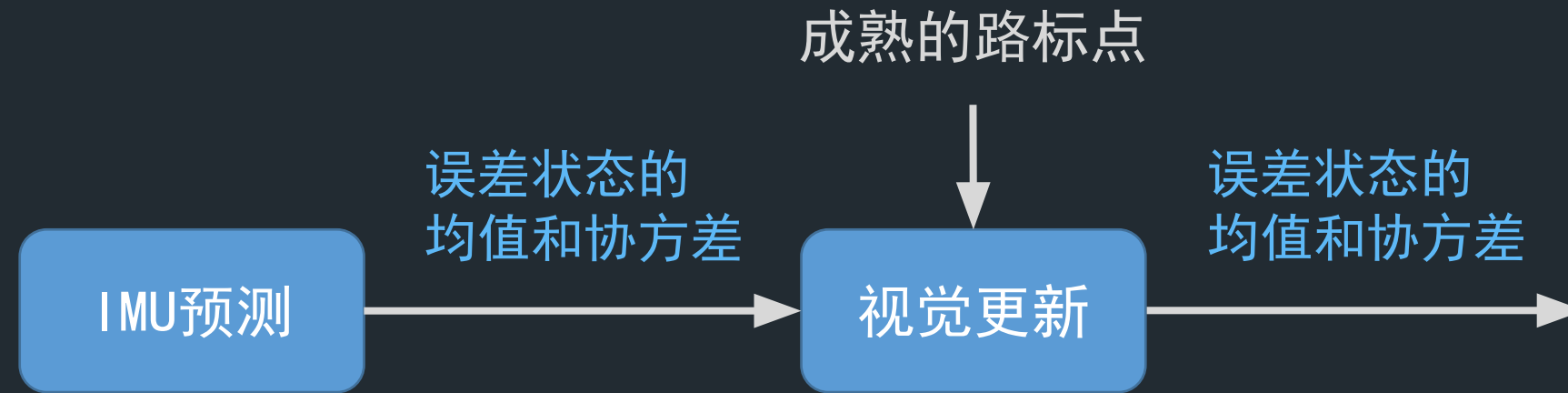


# MSCKF

## 流程:

1. 初始化
2. IMU预测
3. 视觉跟踪
4. 视觉增广
5. 视觉更新—选老点或者看不见的点
6. 视觉更新—边缘化
7. 剔除老帧

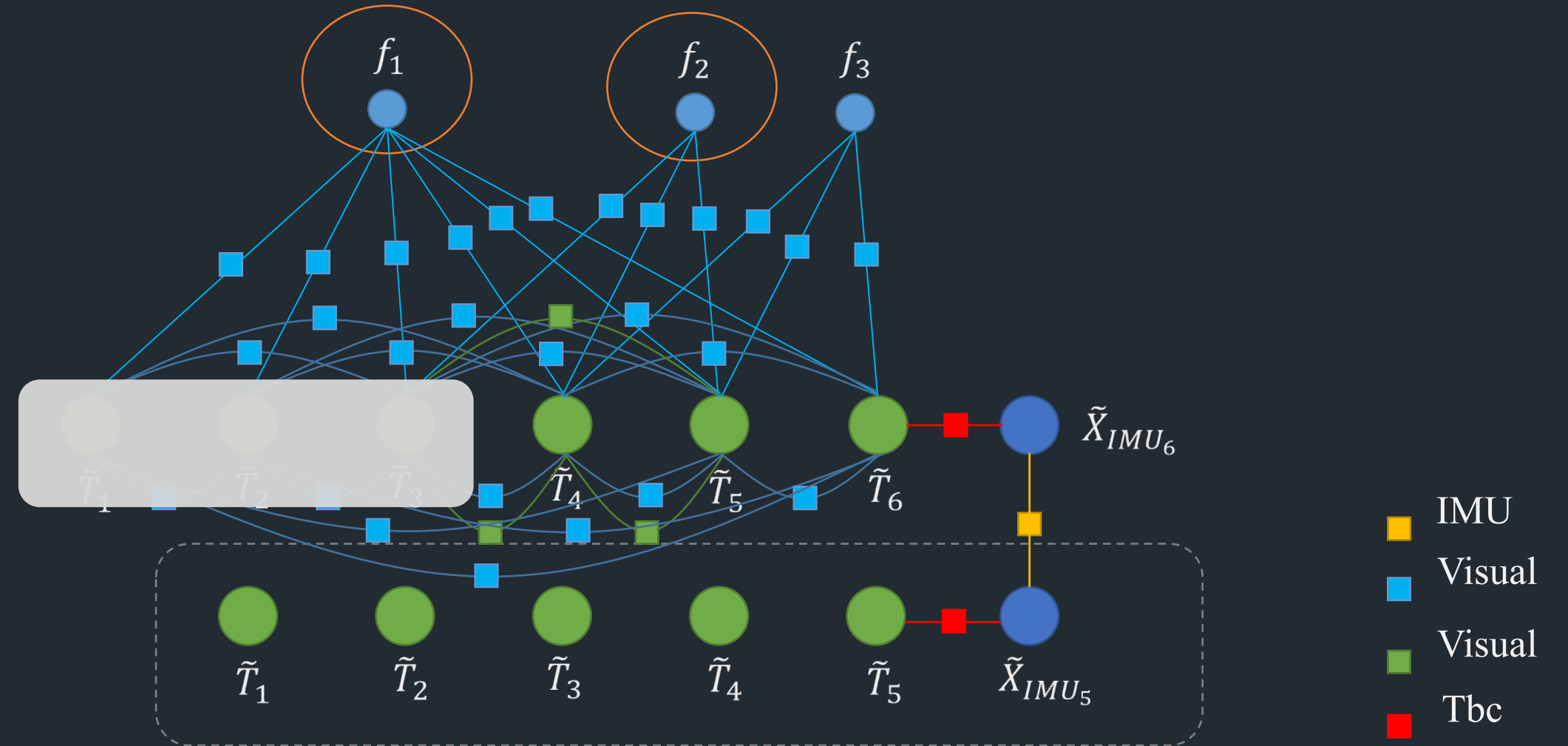




误差状态向量:

$$\tilde{X}_k^{(15+6N) \times 1} = [\tilde{X}_{IMU_k} \ \delta\theta_{C_1} \ ^G\tilde{p}_{C_1} \ \cdots \ \delta\theta_{C_N} \ ^G\tilde{p}_{C_N}]^T$$

$$\tilde{X}_{IMU_k}^{15 \times 1} = [\delta\theta_I \ \tilde{b}_g \ ^G\tilde{v}_I \ \tilde{b}_a \ ^G\tilde{p}_I]^T$$



[1] A. I. Mourikis. A multi-state constraint kalman filter for vision-aided inertial navigation. Proc. ICRA, 2007.

[2] F. Gonzalez. Visual inertial odometry for mobile robotics. 2015.

[3] N. Trawny. Indirect kalman filter for 3D attitude estimation. 2005.

[4] L. Clement. The battle for filter supremacy: a comparative study of the multi-state constraint kalman filter and the sliding window filter. CVR. 2015.

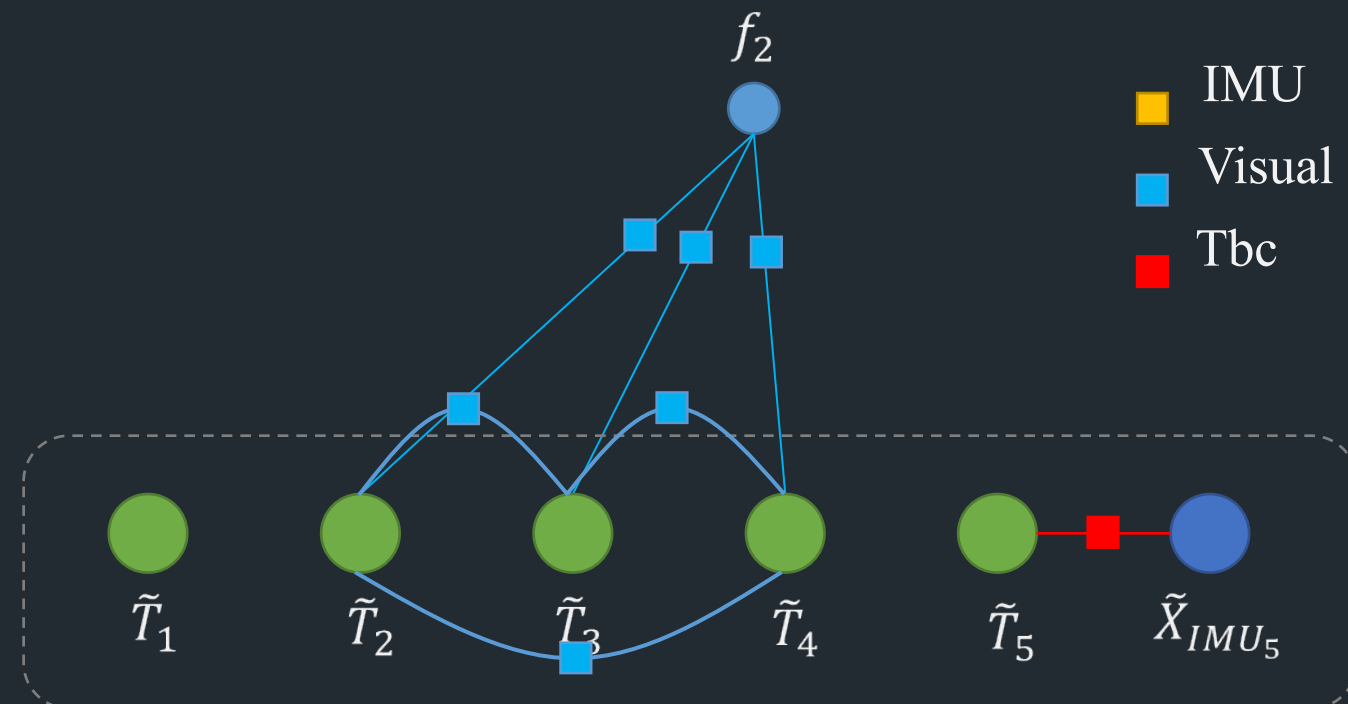
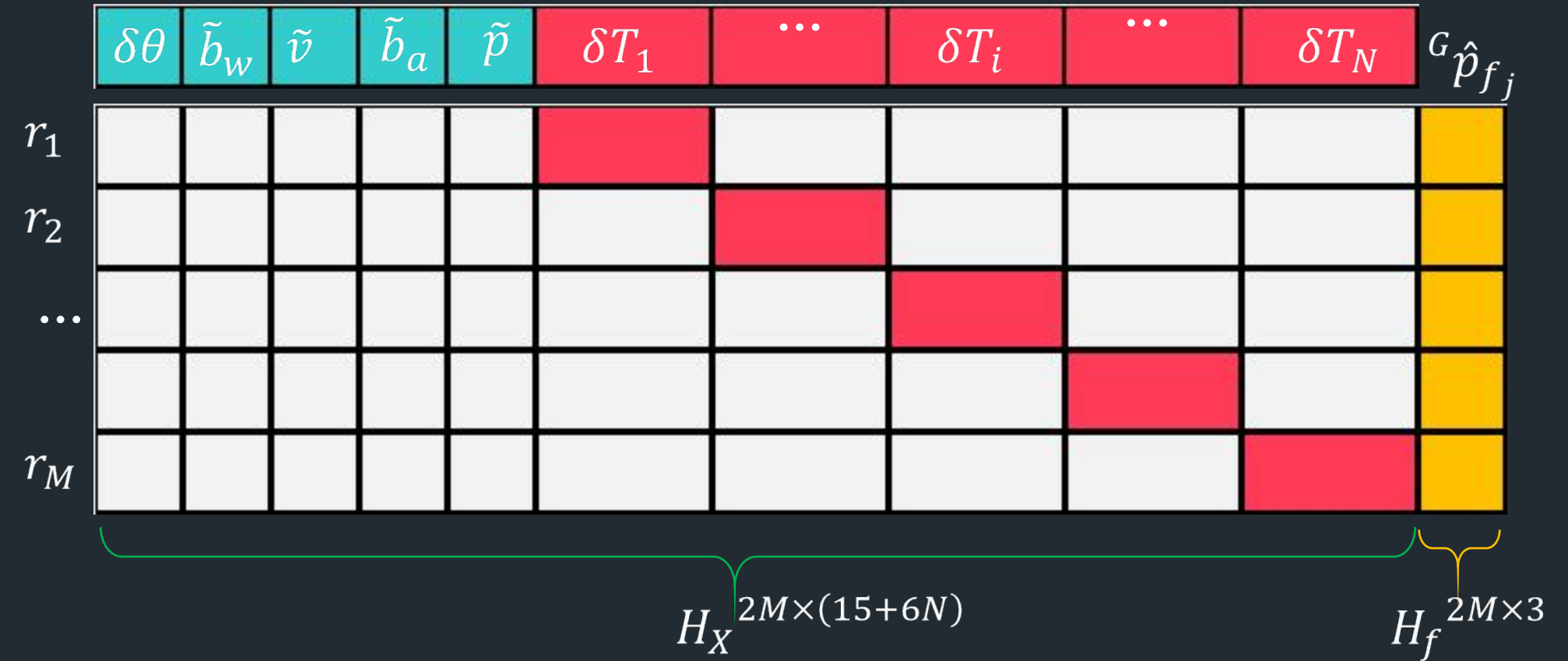
[5] J. Sola. Quaternion kinematics for the error-state Kalman filter. 2017.

第j个路标点的所有视觉误差为

$$\begin{aligned} r^{2M \times 1} &\cong H_X^{2M \times (15+6N)} \tilde{X}^{(15+6N) \times 1} \\ &+ H_f^{2M \times 3} G \hat{p}_{f_j}^{3 \times 1} + n^{2M \times 1} \end{aligned}$$

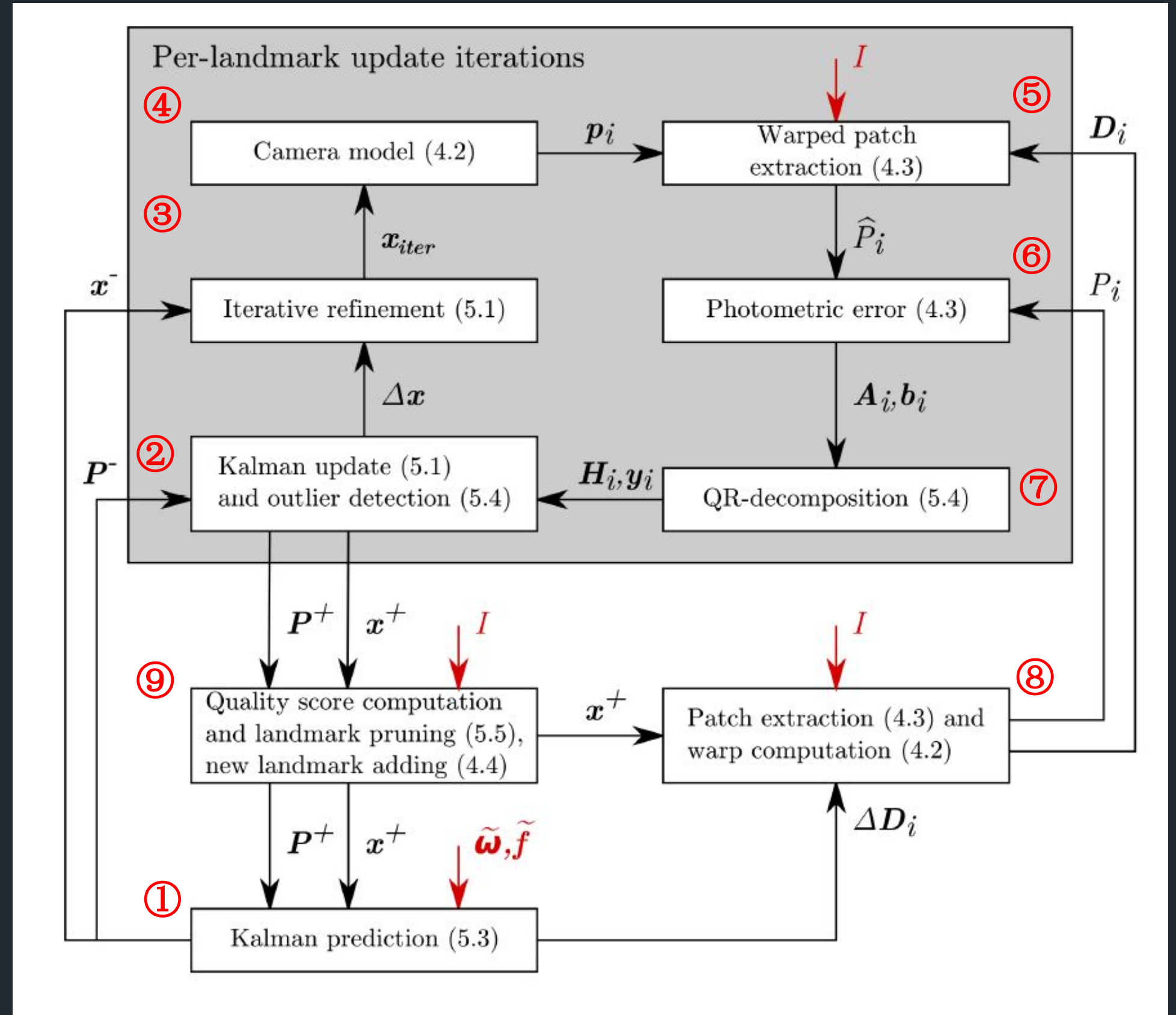
将r投影到 $H_f$ 的左零空间，相当于对路标点进行边缘化，将边缘化约束来优化。

$$\begin{aligned} r_0^{(2M-3M_L) \times 1} &= A^T r^{2M \times 1} \\ &\cong \underbrace{A^T H_X^{2M \times (15+6N)}}_{H_0} \tilde{X}^{(15+6N) \times 1} + \underbrace{A^T n^{2M \times 1}}_{n_0} \end{aligned}$$

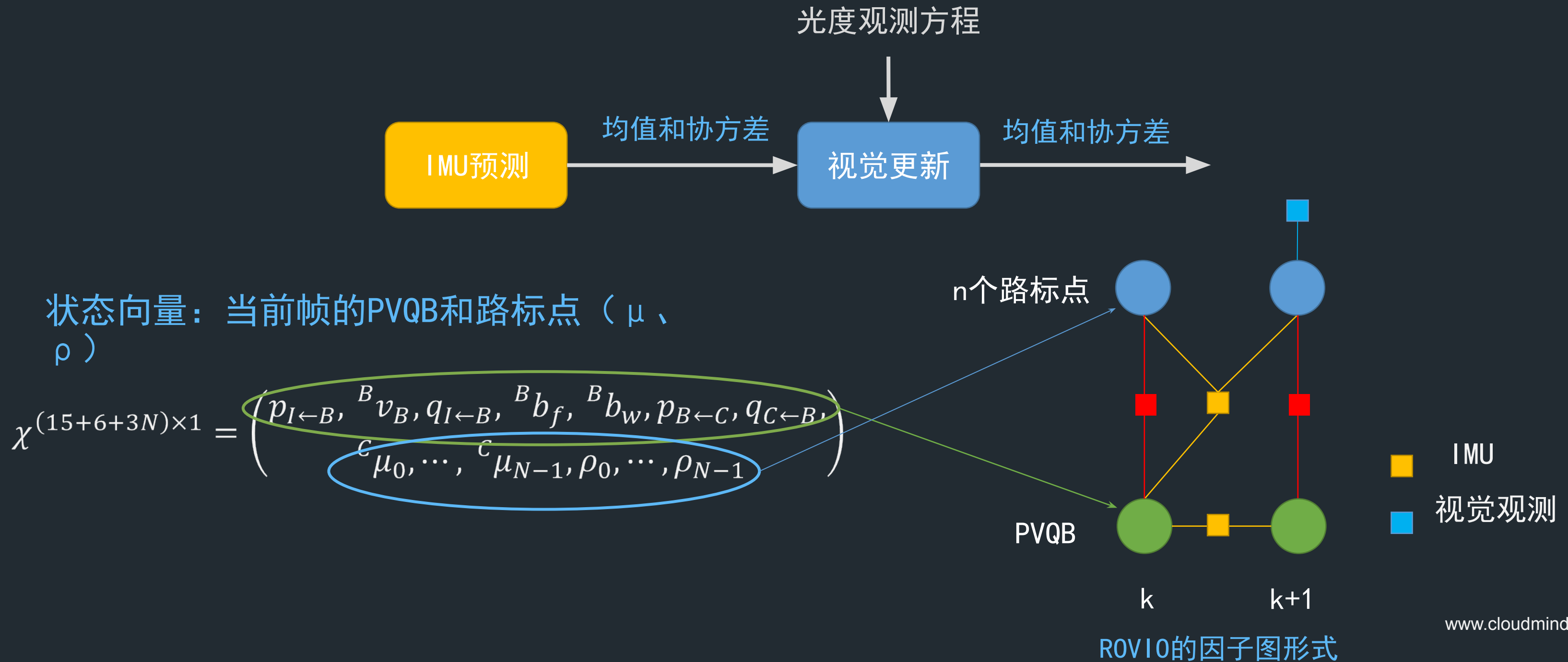


## 流程:

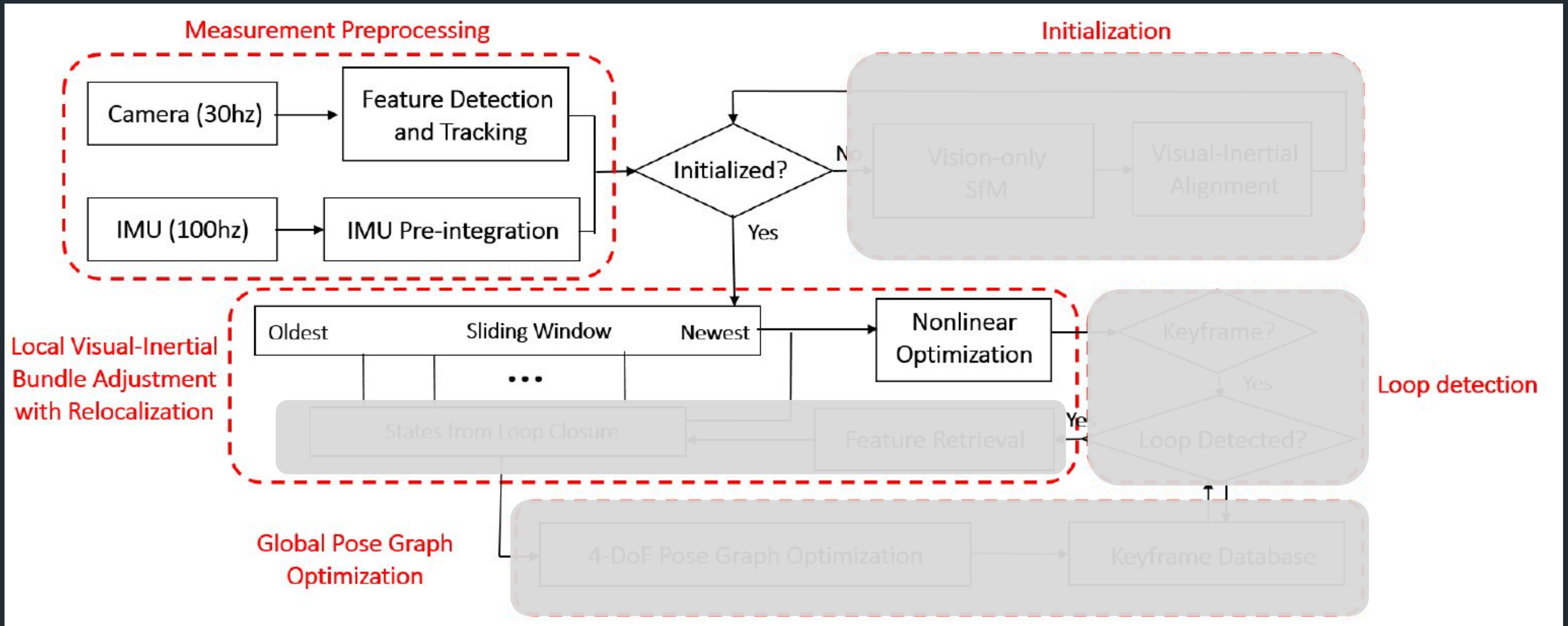
1. IMU预测
2. 视觉更新
3. IEKF
4. 相机模型
5. 像素坐标校正
6. 光度误差
7. QR分解
8. Patch提取及Warp计算
9. 路标点质量评价及维护



**创新点：**路标点用当前帧下的归一化相机系坐标 $P_c$ 和逆深度表示，并作为状态向量进行预测和更新。参数化不同导致预测和更新的细节不同，但是总体还是EKF五大公式。又因 $P_c$ 有边界约束，因此引入bearing vector来使得 $P_c$ 平滑可导。

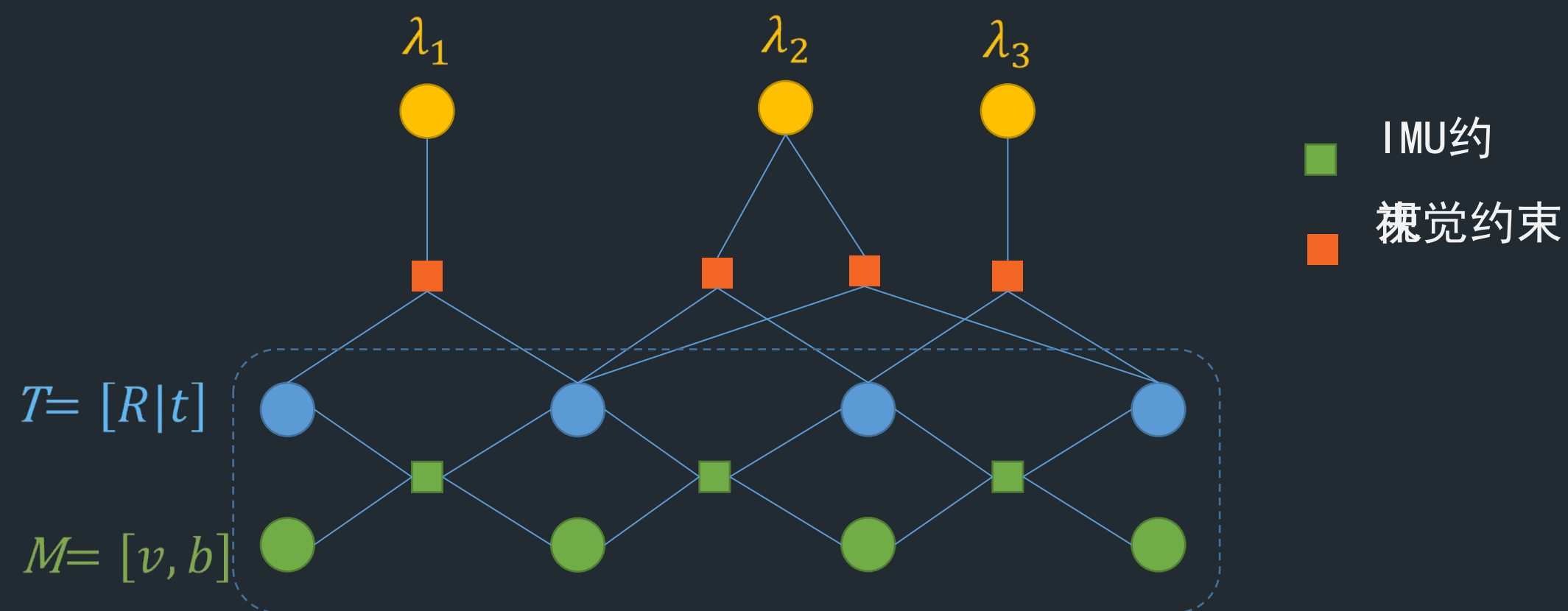






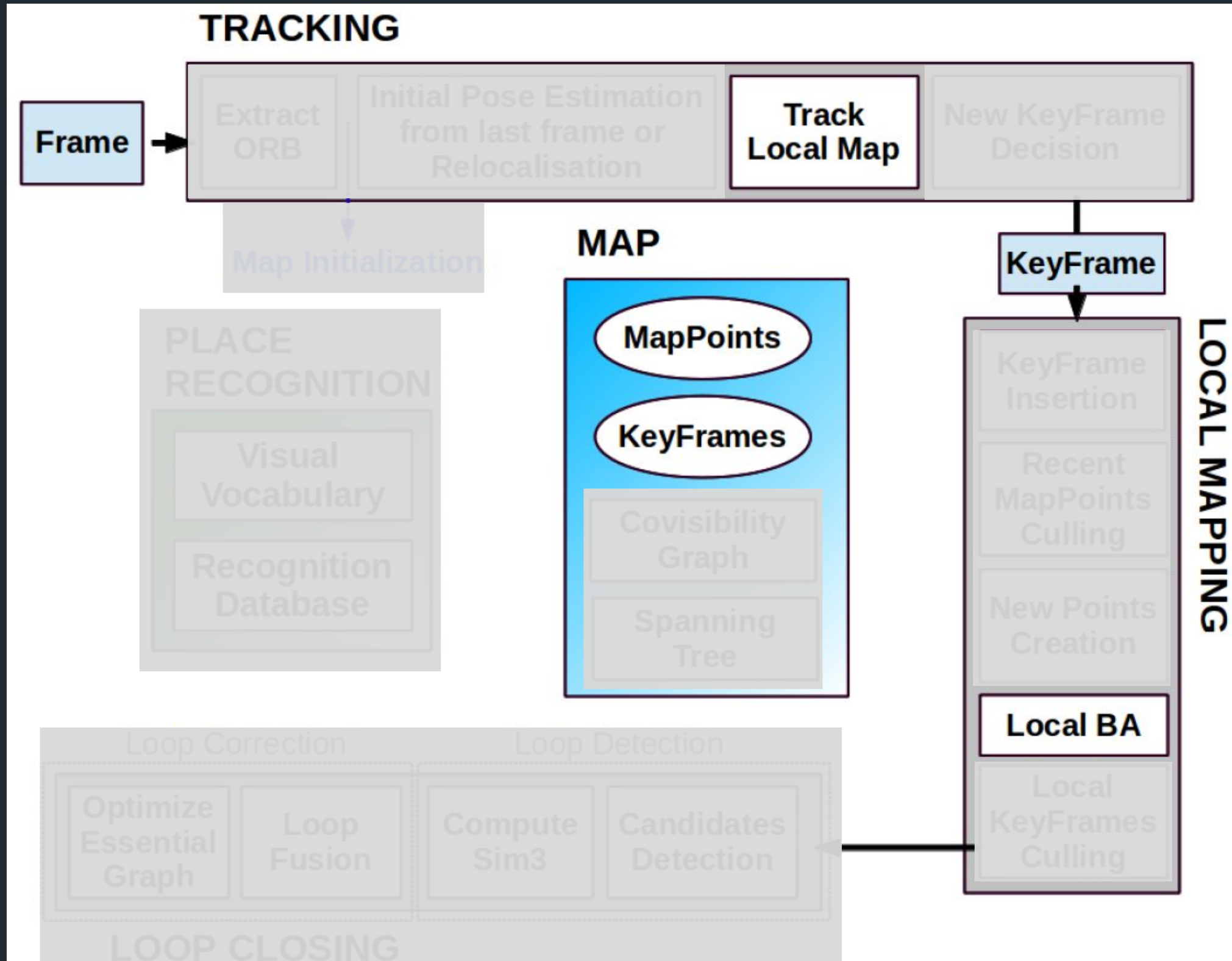
优化向量包括滑动窗口内的 $n$ 个相机状态PVQB、Camera到IMU的外参、 $m$ 个3D点的逆深度：

$$X = [x_0, x_1, \dots, x_n, x_c^b, \lambda_0, \lambda_1, \dots, \lambda_m]$$

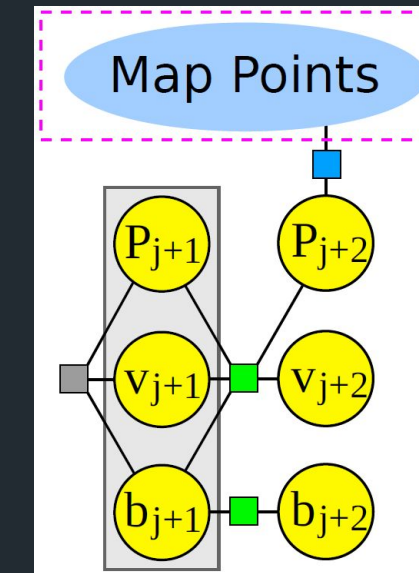


VINS的因子图形式

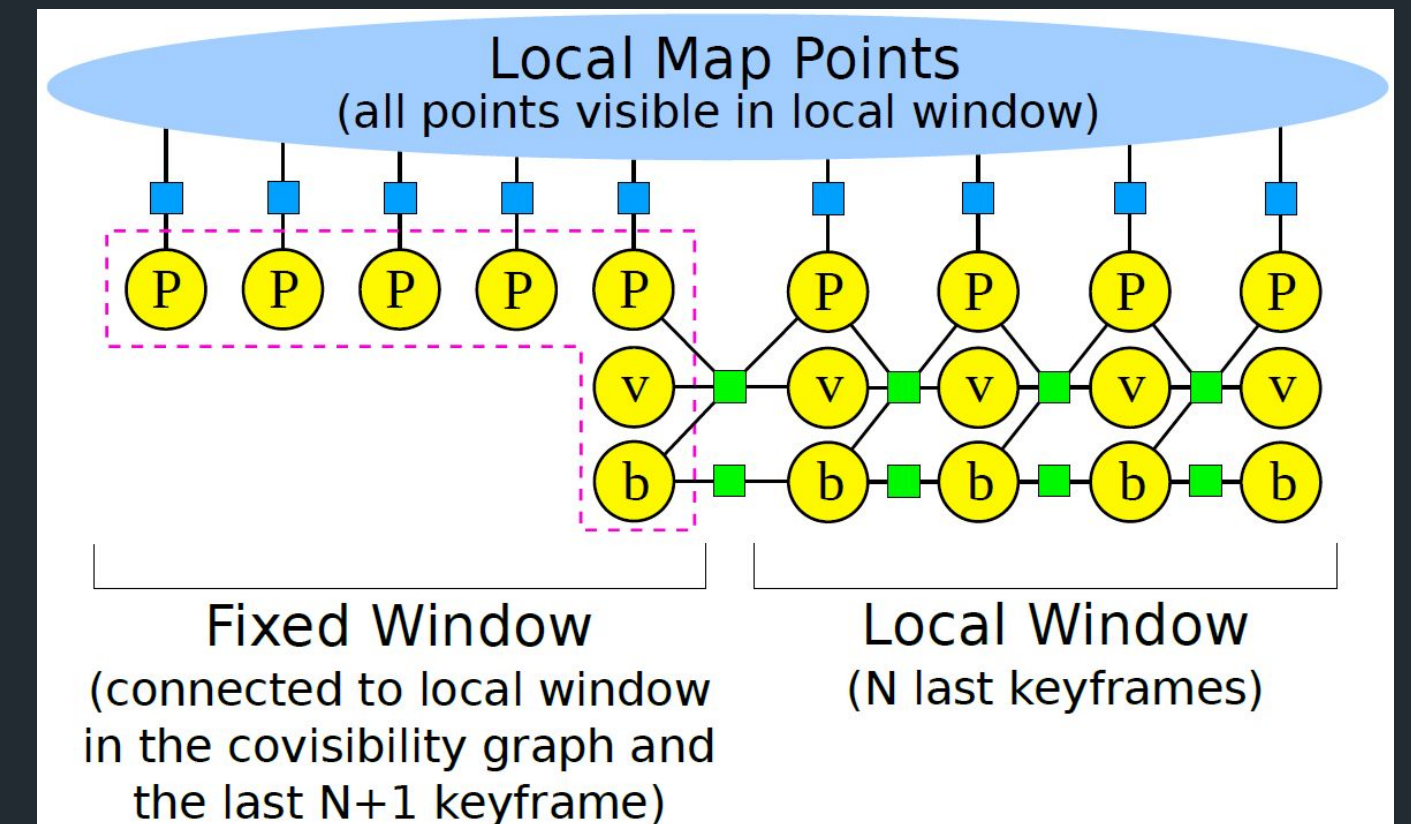
- [1] T. Qin. VINS-Mono: A robust and versatile monocular visual-inertial state estimator. arXiv preprint arXiv: 1708.03852, 2017.
- [2] N. Trawny. Indirect Kalman Filter for 3D Attitude Estimation. 2005.
- [3] Sola. Quaternion kinematics for error-state kalman filter. 2017.
- [4] K. Eickenhoff. Decoupled, Consistent Node Removal and Edge sparsification for graph-based SLAM. 2016.
- [5] J. Engel. Direct Sparse Odometry. 2016.
- [6] Sliding Window Filter with Application to Planetary Landing. 2010.
- [7] S. Leutenegger. Keyframe-Based Visual-Inertial SLAM Using Nonlinear Optimization. 2015.



Track Local Map: 仅优化当前帧

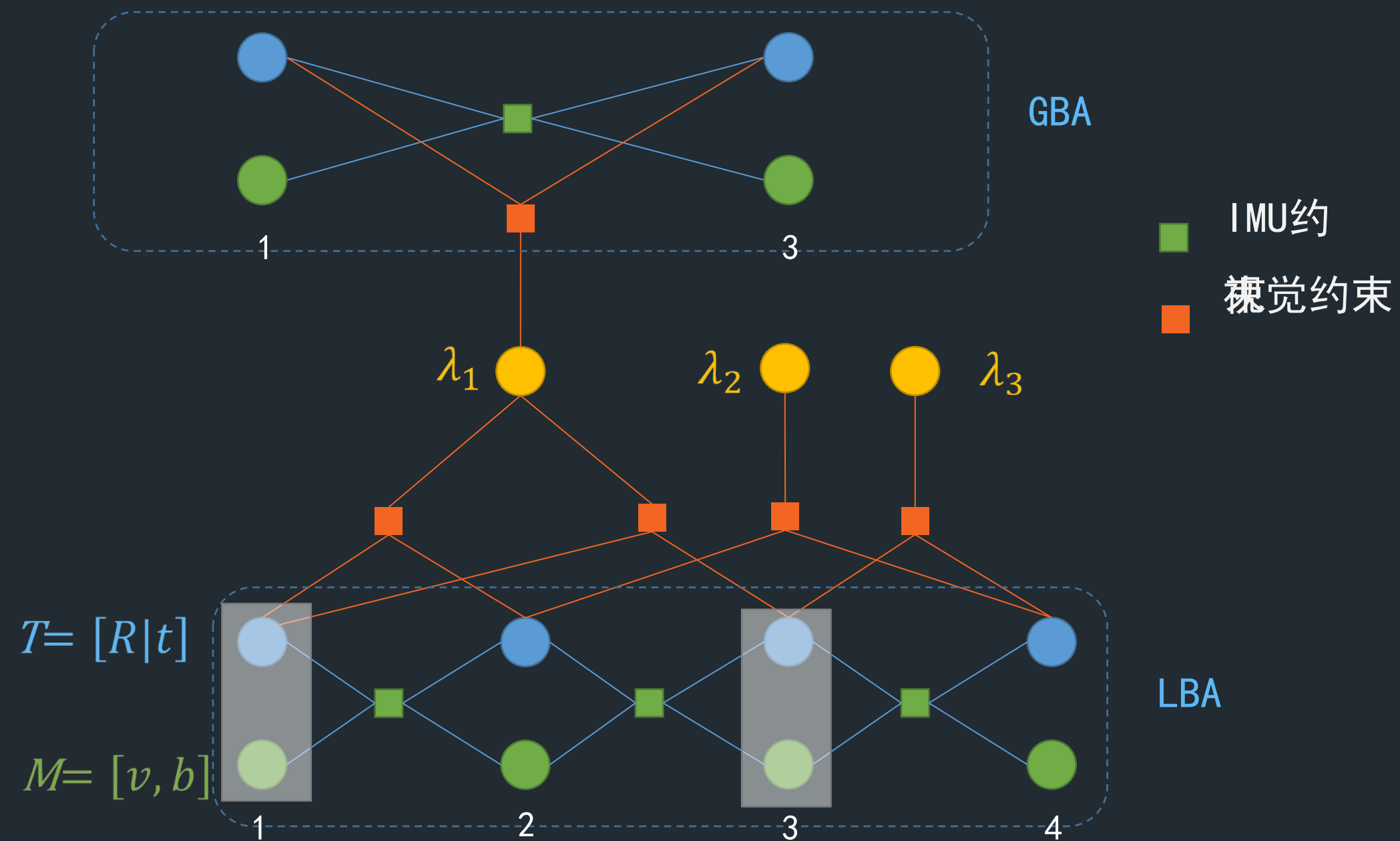


Local BA: 优化滑窗内所有帧PVQB和路标点





通过增量方式加速BA求解，误差为IMU和视觉，LBA为滑窗优化，类似VINS，GBA为所有KF优化



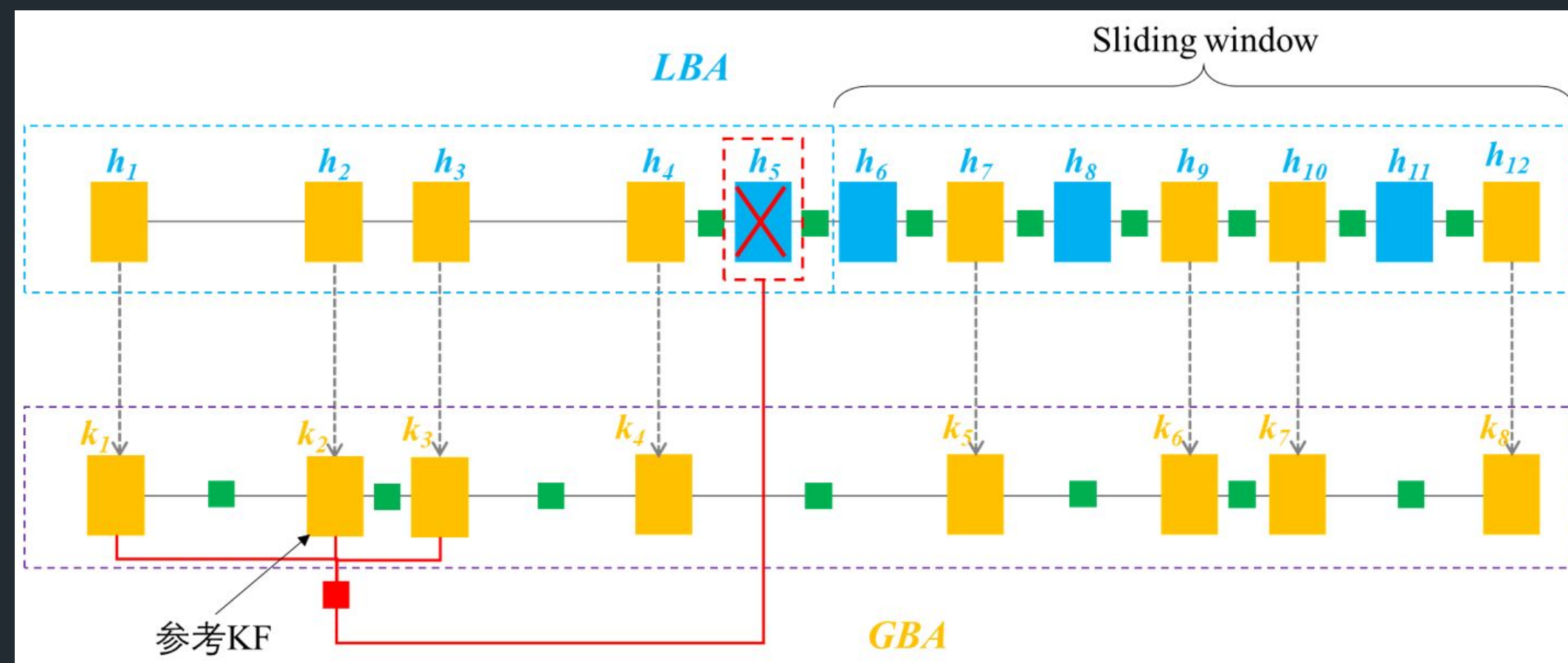
ICE-BA的因子图形式

## ICE-BA的增量思想

ICE-BA认为在建立增量方程时, 对之前已经算过的那些状态向量再次计算Jacobian对精度不会有太大改善, 因此没必要重新线性化(线性化就是计算Jacobian), 因为即使重新计算, 得到的Jacobian也没什么变化。并且, 也没必要对整体 $[H|b]$ 矩阵进行消元, 而是只对变化的进行更新。最后, 只计算变化的路标点即可。对于没有变化的状态向量的临时值则一直保存下来, 避免重新计算。

总结来说, 在ICE-BA中更新Factor有三种情况:

1. 该Factor在本次迭代中没有变化, 则不更新;
2. 该Factor是新Factor, 则在原来基础上 += 新Factor。如新观察到一个新的路标点, 则对此新路标点的观测约束就是一个新的Factor;
3. 该Factor已存在但需要更新, 则先减去旧Factor, 再加上新Factor。



[1] H. Liu. ICE-BA: Incremental, consistent and efficient bundle adjustment for visual-inertial slam. 2018. CVPR.

[2] H. Liu. Robust keyframe-based dense SLAM with an RGB-D camera. 2017.

# Thanks

!

cloudminds