Lecture 15 – Simultaneous Localization & Mapping

Dec 25th, 2018

Danping Zou,

Associate Professor

Institute for Sensing and Navigation

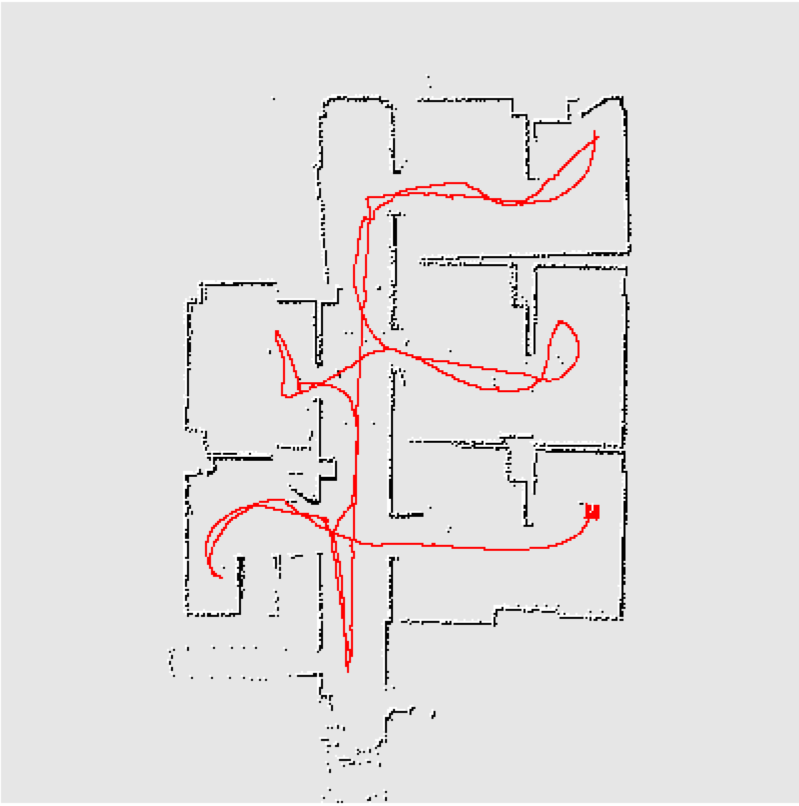
上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

Outline

- Classic SLAM
  - SLAM problem
  - EKF SLAM
  - Visual EKF SLAM – MonoSLAM/StructSLAM
- Visual SLAM systems – A brief survey
  - Filter vs keyframe optimization
  - PTAM/ORB-SLAM
  - Direct visual SLAM
    - DTAM/LSD-SLAM/SVO/DSO

What's SLAM

- **SLAM** - **S**imultaneous **L**ocalization **A**nd **M**apping :
  - To solve localization and mapping problems at the same time.

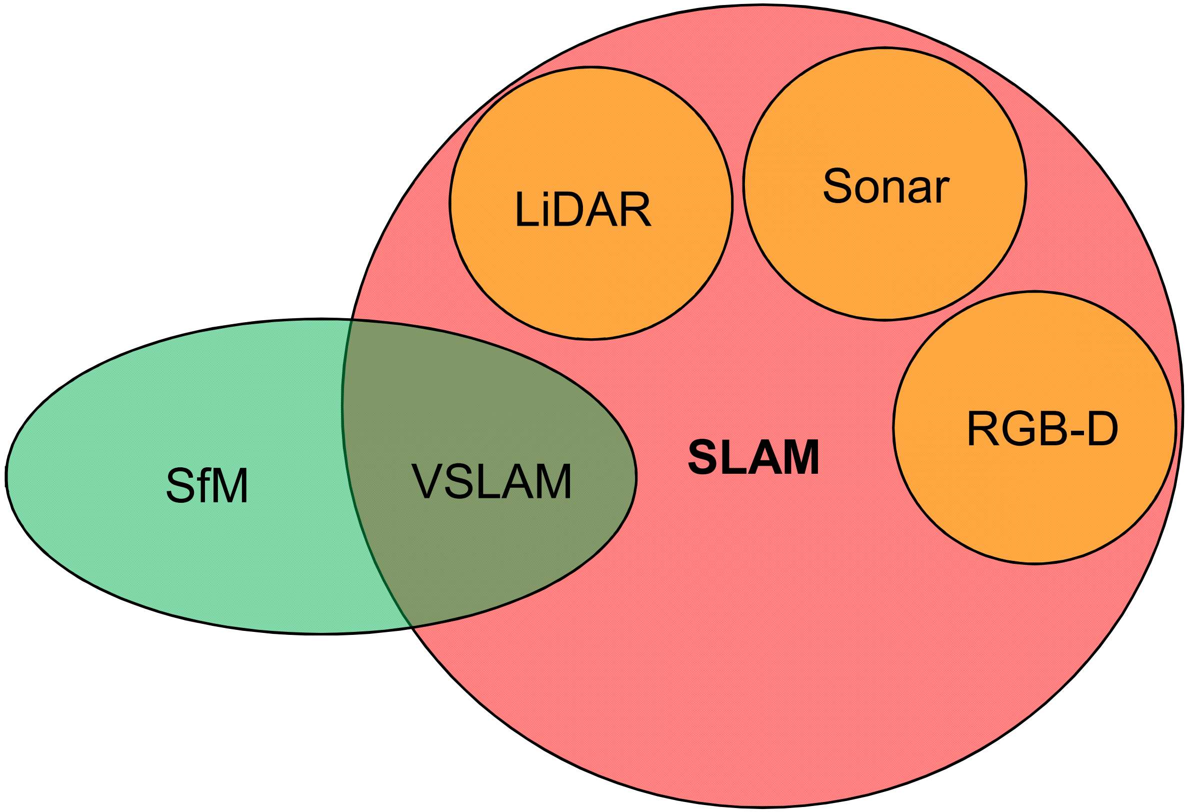


Explore in an **unknown environment**:

- No global positioning systems
- No map

SfM vs SLAM

- Visual SLAM (VSLAM) is a special case of Structure-from-motion (SfM)
  - Online & Real-time



Classic definition of SLAM problem

- Inputs :
  - Time sequence of **observations (sensor measurements)**

$$z_{1:k} = \{z_1, z_2, \cdots, z_k\}$$

  - Robot control **signals**

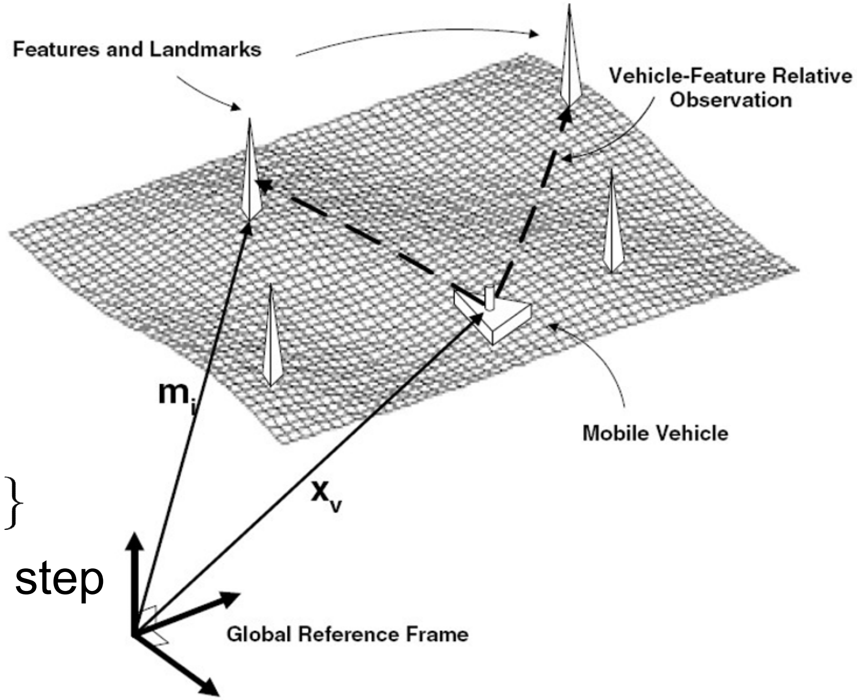
$$u_{1:k} = \{u_1, u_2, \cdots, u_k\}$$

- Outputs:
  - A **map** of the environment

$$m = \{m^{(1)}, m^{(2)}, \cdots, m^{(n)}\}$$

  - Sensor **poses** associated with each step

$$x_{1:k} = \{x_1, x_2, \cdots, x_k\}$$



SLAM problem

- In probabilistic form, the SLAM problem tries to get the state distribution

$$P(\boldsymbol{x}_k, \boldsymbol{m} | \boldsymbol{z}_{1:k}, \boldsymbol{u}_{1:k}, \boldsymbol{x}_0)$$

be computed for all time steps k.

$\boldsymbol{x}_k$: Current pose of the robot

$\boldsymbol{m}$: All the landmarks in the map

$\mathbf{z}_{1:k}$: Observations (current and previous time steps)

$\mathbf{u}_{1:k}$: Control inputs (current and previous time steps)

$\boldsymbol{x}_0$: Initial pose

SLAM in probabilistic form

- Use **Bayesian inference** to estimate the poster probability recursively.

$$P(\boldsymbol{x}_{k-1}, \boldsymbol{m} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k-1}, \boldsymbol{x}_0)$$

$$P(\boldsymbol{x}_k | \boldsymbol{x}_{k-1}, \boldsymbol{u}_k)$$

Prediction : $P(\boldsymbol{x}_k, \boldsymbol{m} | \boldsymbol{z}_{1:k-1}, \boldsymbol{u}_{1:k}, \boldsymbol{x}_0)$

$$P(\boldsymbol{z}_k | \boldsymbol{x}_k, \boldsymbol{m})$$

Correction : $P(\boldsymbol{x}_k, \boldsymbol{m} | \mathbf{z}_{1:k}, \mathbf{u}_{1:k}, \boldsymbol{x}_0)$

Probabilistic SLAM

- Motion model : probability distribution on state transitions

$$P(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{u}_k)$$

- Observation model : the probability of making an observation $\boldsymbol{z}_k$ when the robot location and landmark locations are known.

$$P(\boldsymbol{z}_k|\boldsymbol{x}_k, \boldsymbol{m})$$

SLAM problem

- Usually, we assume the noise model (both motion model and measurement model) is Gaussian.

$$P(\boldsymbol{x}_k, \boldsymbol{m} | \boldsymbol{z}_{1:k}, \mathbf{u}_{1:k}, \boldsymbol{x}_0)$$

$$\Downarrow$$

$$\begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{m} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{x}_{k|k} \\ \boldsymbol{m}_{k|k} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{xx} & \boldsymbol{\Sigma}_{xm} \\ \boldsymbol{\Sigma}_{xm}^{\mathrm{T}} & \boldsymbol{\Sigma}_{mm} \end{bmatrix}_{k|k} \right)$$

$$\Downarrow$$

$$\boldsymbol{X} \sim \mathcal{N}(\boldsymbol{X}_{k|k}, \Sigma)$$

EKF SLAM

- Pipeline of EKF SLAM

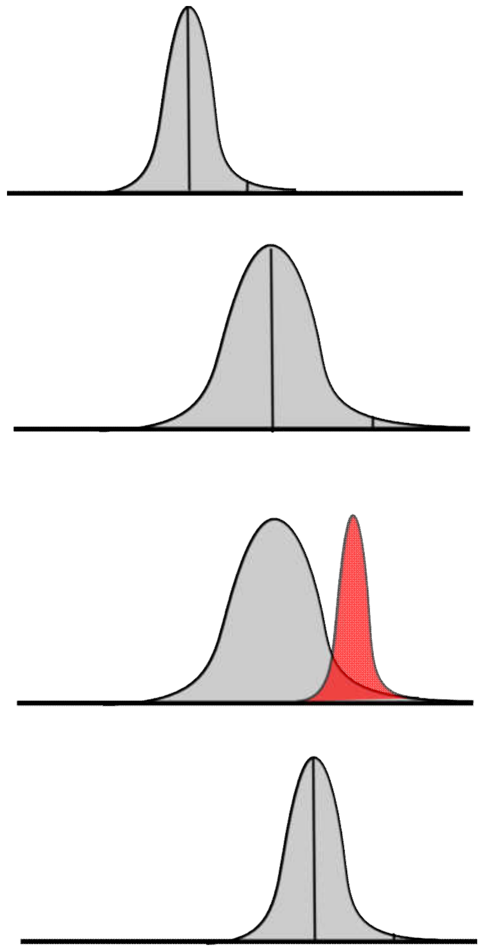
$$\boldsymbol{X}_{k-1|k-1}, \Sigma_{k-1|k-1}$$

Prediction : $\boldsymbol{X}_{k|k-1} = \boldsymbol{f}(\boldsymbol{X}_{k-1|k-1})$

$$\boldsymbol{X}_{k|k-1}, \Sigma_{k|k-1}$$

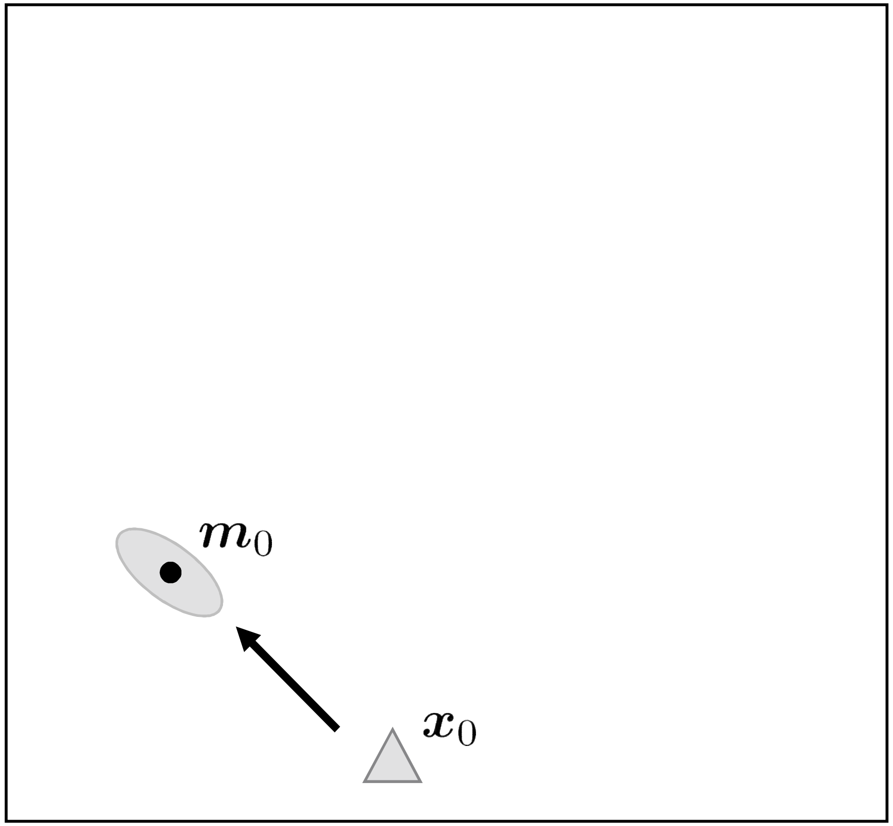
Correction : $\boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{X}_{k|k-1})$

$$\boldsymbol{X}_{k|k}, \Sigma_{k|k}$$



SLAM overview

- Let us assume that the robot uncertainty at its initial location is zero.
- From this position, the robot observes a feature which is mapped with an uncertainty related to the sensor error.



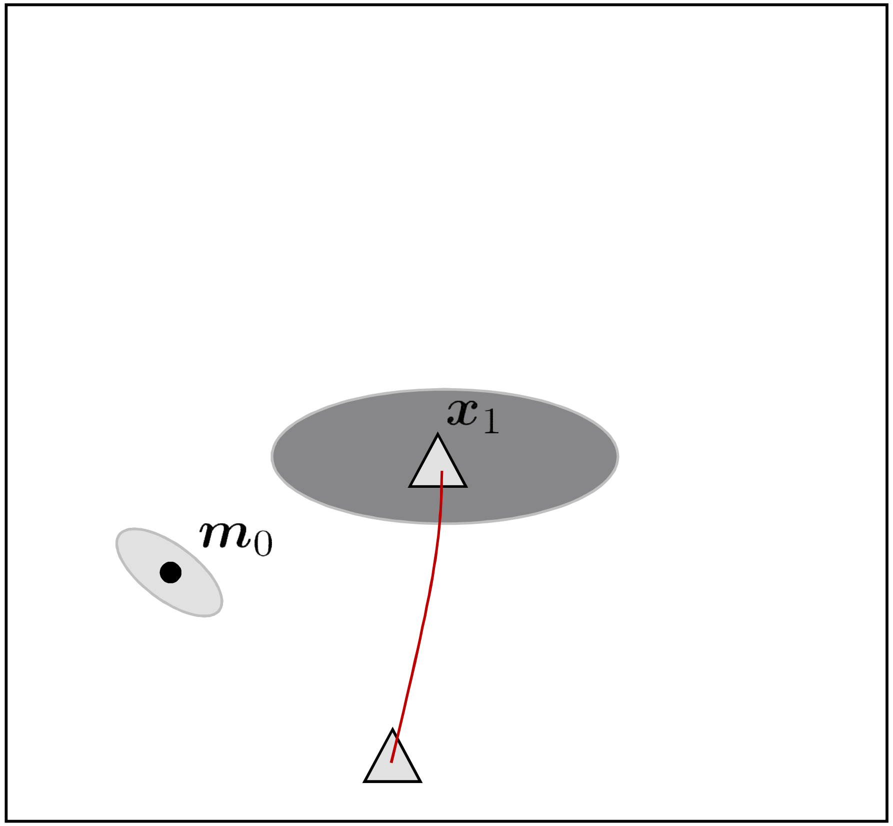
$$\boldsymbol{X}_{0|0} = \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{m}_0 \end{bmatrix} \quad \Sigma_{0|0} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Sigma_{m_0 m_0} \end{bmatrix}$$

SLAM overview

- Prediction： As the robot moves, its pose uncertainty increases under the effect of the errors introduced by the odometry

$$\boldsymbol{x}_1 = \boldsymbol{f}(\boldsymbol{x}_0) + \boxed{\boldsymbol{n}_0}$$

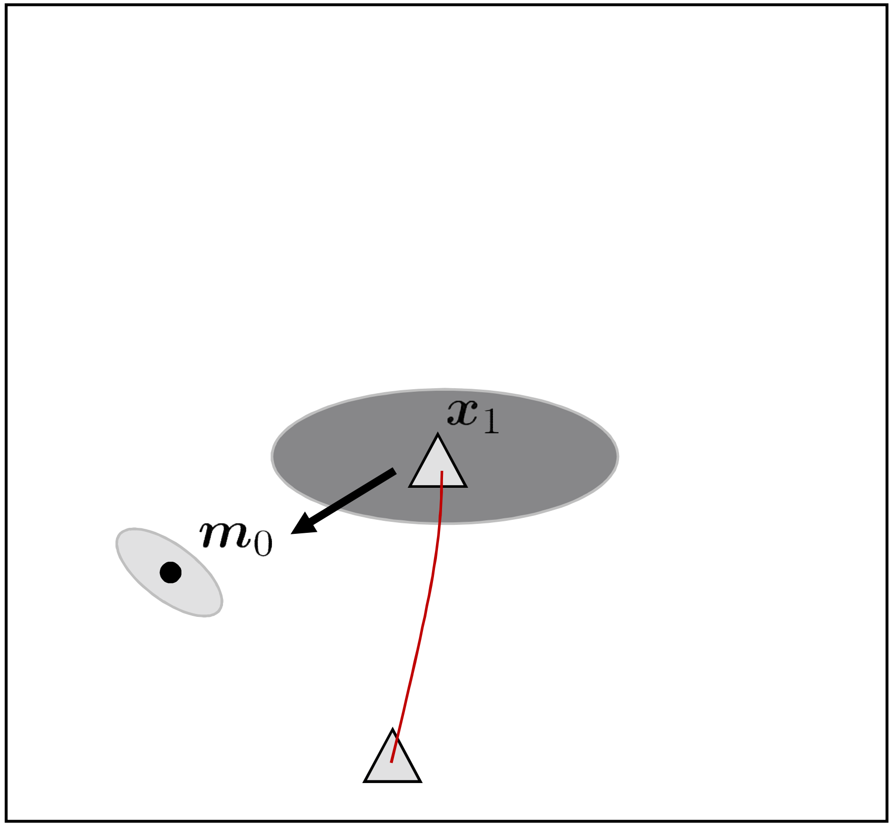
Prediction noise



$$\boldsymbol{X}_{1|0} = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{m}_0 \end{bmatrix} \qquad \Sigma_{1|0} = \begin{bmatrix} \Sigma_{x_1 x_1} & \mathbf{0} \\ \mathbf{0} & \Sigma_{m_0 m_0} \end{bmatrix}$$

SLAM overview

- Update： the robot has a new observation of the landmark from a new viewpoint.



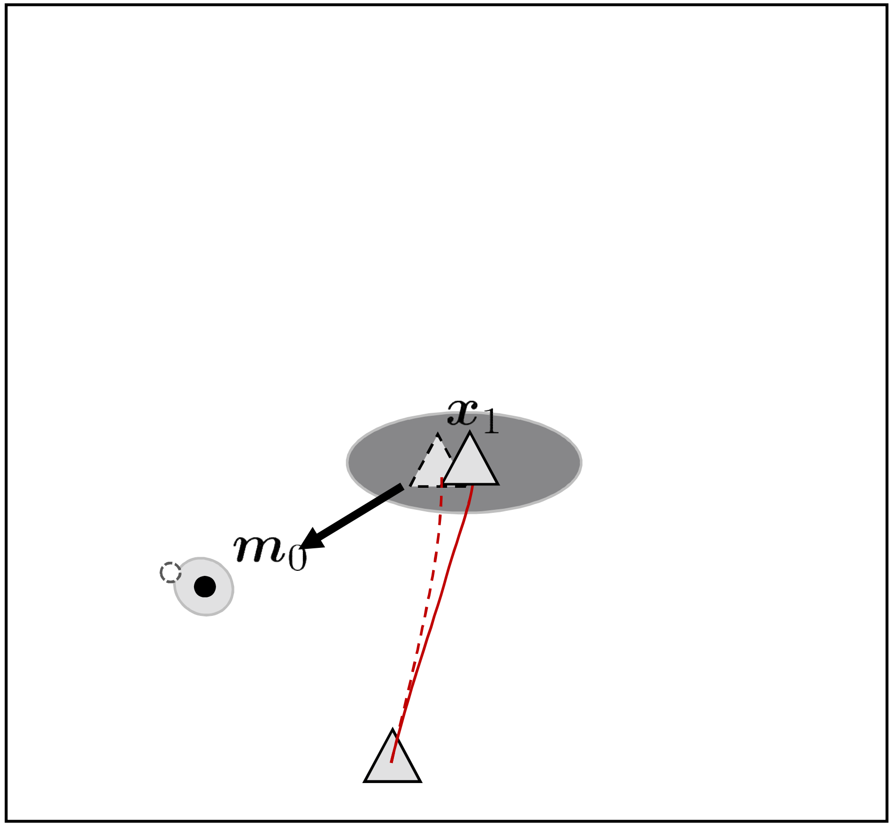
SLAM overview

- Update： the robot has a new observation of the landmark from a new viewpoint.

Prediction noise

$$z_1 = h(X_{1|0}) + \boxed{r_1}$$

$$X_{1|0}, \Sigma_{1|0} \xrightarrow{\text{EKF update}} X_{1|1}, \Sigma_{1|1}$$



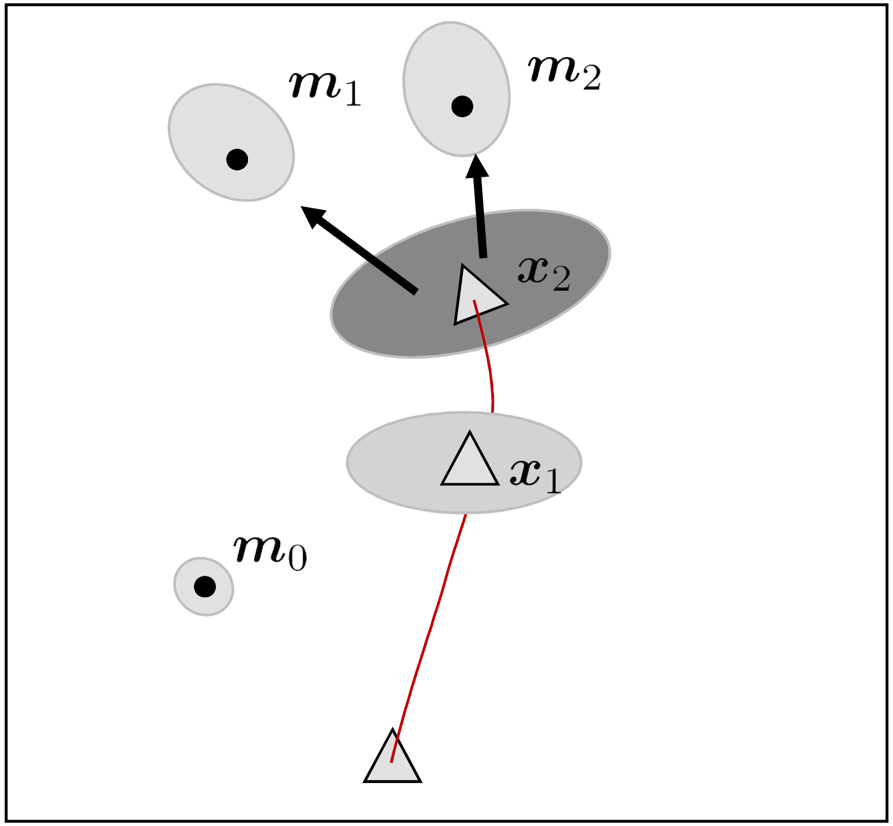
SLAM overview

- **New landmarks** : the robot observes two new features and use them to update the whole state
  - Add them to the state first

$$\boldsymbol{X}_{2|1} \leftarrow \begin{bmatrix} \boldsymbol{x}_2 \\ \boldsymbol{m}_0 \\ \boldsymbol{m}_1 \\ \boldsymbol{m}_2 \end{bmatrix}$$

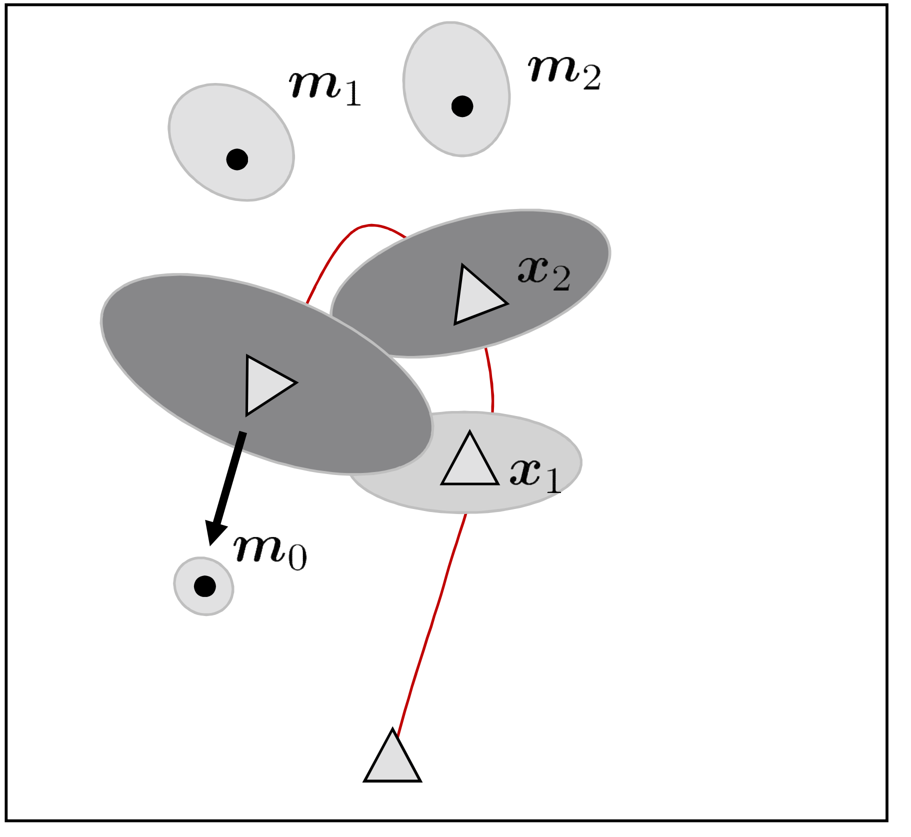
  - As well as the covariance

$$\Sigma_{2|1} \leftarrow \begin{bmatrix} \Sigma_{2|1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Sigma_{m_1 m_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Sigma_{m_2 m_2} \end{bmatrix}$$



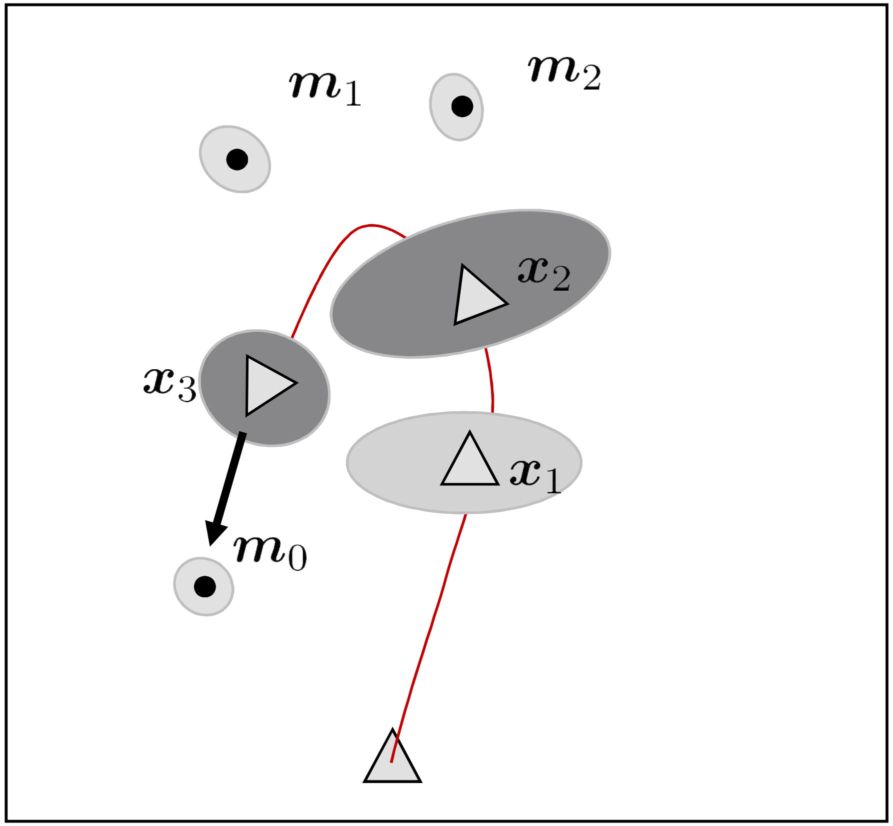
SLAM overview

- Loop closing - the robot observes old landmarks again and update the whole state to remove the drift error.



SLAM overview

- Loop closing - the robot observes old landmarks again and update the whole state to remove the drift error.
- When a loop closure is detected, the robot pose uncertainty shrinks.
- At the same time, the map is updated and the uncertainty of other observed features and all previous robot poses also reduce



EKF SLAM

- **Motion model** : probability distribution on state transitions

$$\boldsymbol{X}_{k|k-1} = \boldsymbol{f}(\boldsymbol{X}_{k-1|k-1}) + \boldsymbol{n}_k \qquad \boldsymbol{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_k)$$

- **Observation model** : the probability of making an observation when the robot location and landmark locations are known.

$$\boldsymbol{z}_k = \boldsymbol{h}(\boldsymbol{X}_{k|k-1}) + \boldsymbol{r}_k \qquad \boldsymbol{r}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$$

$$\begin{tikzcd}
\text{Observation:} & z_1 & z_2 & z_3 & z_4 & \\
\text{State:} & \boldsymbol{x}_1 \arrow[u] \arrow[r] & \boldsymbol{x}_2 \arrow[u] \arrow[r] & \boldsymbol{x}_3 \arrow[u] \arrow[r] & \boldsymbol{x}_4 \arrow[u] \arrow[r] & \cdots
\end{tikzcd}$$

EKF pipeline

Update(Correction)

1. Kalman Gain:

$$\mathbf{G}_k = \hat{\boldsymbol{\Sigma}}_k \mathbf{H}_k^T (\mathbf{H}_k \hat{\boldsymbol{\Sigma}}_k \mathbf{H}_k^T + \mathbf{R}_k)^{-1}$$

2. Residual:

$$\Delta \boldsymbol{z}_k = (\mathbf{z}_k - \mathbf{H}_k \hat{\boldsymbol{\mu}}_k)$$

3. State update:

$$\boldsymbol{\mu}_k = \hat{\boldsymbol{\mu}}_k + \mathbf{G}_k \Delta \boldsymbol{z}_k$$

4. Covariance update:

$$\boldsymbol{\Sigma}_k = (\boldsymbol{I} - \mathbf{G}_k \mathbf{H}_k) \hat{\boldsymbol{\Sigma}}_k$$

Prediction

1. State prediction:

$$\hat{\boldsymbol{\mu}}_k = \mathbf{F}_k \boldsymbol{\mu}_{\boldsymbol{k-1}}$$

2. Covariance propagation:

$$\hat{\boldsymbol{\Sigma}}_k = \mathbf{F}_k \boldsymbol{\Sigma}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k$$

$$\boldsymbol{x}_0, \boldsymbol{\Sigma}_0$$

Initial state & covariance

EKF SLAM

- Further reading about EKF SLAM

Simultaneous localization and mapping with the extended Kalman filter，A very quick guide.

Visual EKF SLAM

Real-Time Simultaneous Localisation and Mapping with a Single Camera,
Andrew J. Davison, ICCV 2003.
Real-Time Localisation and Mapping with Wearable Active Vision,
Andrew J. Davison, Walterio Mayol and David W. Murray, ISMAR 2003
Real-Time 3D SLAM with Wide-Angle Vision,
Andrew J. Davison, Yolanda Gonzalez Cid and Nobuyuki Kita, IAV 2004.

StructSLAM:

Visual SLAM with Building Structure Lines

Hui Zhong Zhou, Danping Zou et al.

Shanghai Key Laboratory of Navigation and Location Based Services
Shanghai Jiao Tong University
Apirl,2014

Visual EKF SLAM

- State definition :
  - $\mathcal{R}$ Camera motion (position, orientation, velocity)
  - $\mathcal{M}$ 3D coordinates of the landmarks

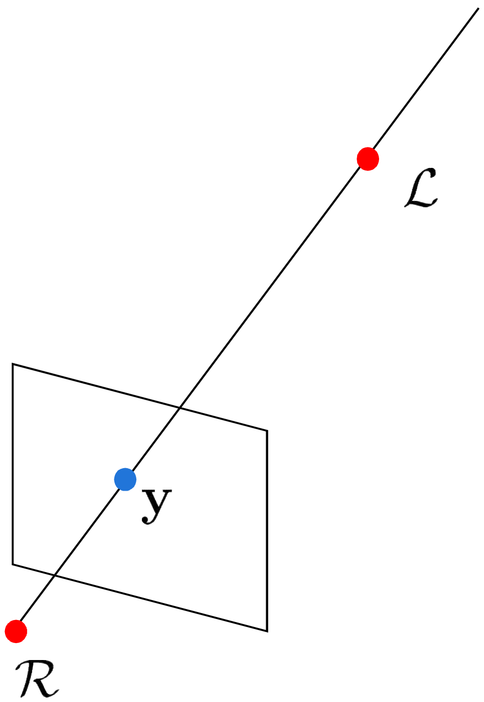
$$\mathbf{x} = \begin{bmatrix} \mathcal{R} \\ \mathcal{M} \end{bmatrix} = \begin{bmatrix} \mathcal{R} \\ \mathcal{L}_1 \\ \vdots \\ \mathcal{L}_n \end{bmatrix} \quad \mathbf{P} = \begin{bmatrix} \mathbf{P}_{\mathcal{R}\mathcal{R}} & \mathbf{P}_{\mathcal{R}\mathcal{M}} \\ \mathbf{P}_{\mathcal{M}\mathcal{R}} & \mathbf{P}_{\mathcal{M}\mathcal{M}} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{\mathcal{R}\mathcal{R}} & \mathbf{P}_{\mathcal{R}\mathcal{L}_1} & \cdots & \mathbf{P}_{\mathcal{R}\mathcal{L}_n} \\ \mathbf{P}_{\mathcal{L}_1\mathcal{R}} & \mathbf{P}_{\mathcal{L}_1\mathcal{L}_1} & \cdots & \mathbf{P}_{\mathcal{L}_1\mathcal{L}_n} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{\mathcal{L}_n\mathcal{R}} & \mathbf{P}_{\mathcal{L}_n\mathcal{L}_1} & \cdots & \mathbf{P}_{\mathcal{L}_n\mathcal{L}_n} \end{bmatrix}$$

Visual EKF SLAM

- **Observation** – Feature points detected on the image
- **Observation model** – perspective projection :

$$\mathbf{y} = \boldsymbol{h}(\mathcal{R}, \mathcal{L}) + \mathbf{v}$$

  - After linearization: $\Delta \boldsymbol{z} = \mathbf{H}_{\mathcal{R}} \delta \mathcal{R} + \mathbf{H}_{\mathcal{L}} \delta \mathcal{L}$



- The observation matrix of the i-th landmark

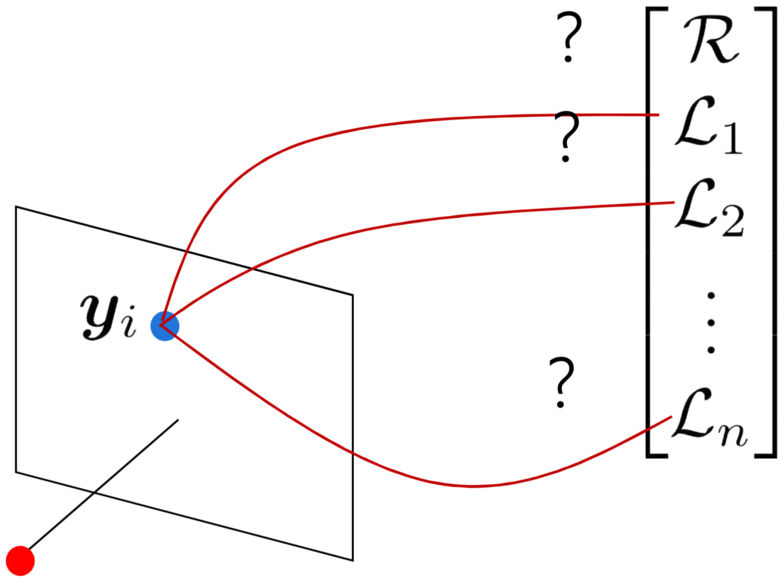
$$\mathbf{H}_i = [\mathbf{H}_{\mathcal{R}}^{(i)} \quad 0 \quad \cdots \quad 0 \quad \mathbf{H}_{\mathcal{L}}^{(i)} \quad 0 \quad \cdots \quad 0]$$

- The whole observation matrix is

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \\ \vdots \\ \mathbf{H}_N \end{bmatrix}$$

Visual EKF SLAM

- **Data association** - Find the correspondences between the observations and state variables (landmarks).
- One-point RANSAC to remove outliers



Visual EKF SLAM

- Prediction model – prediction of camera motion and landmark position
- The landmarks are static, so we have

$$\mathcal{M} \leftarrow \mathcal{M}$$

- For the camera, we use the following prediction

$$\mathcal{R} \leftarrow f_{\mathcal{R}}(\mathcal{R}, \boldsymbol{n})$$

- The covariance is updated by

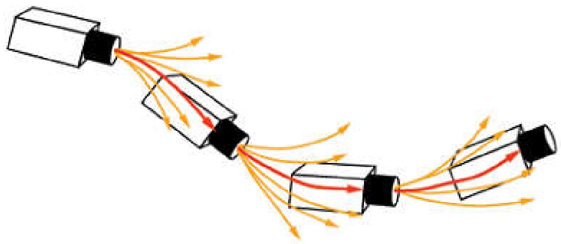
$$\mathbf{P} \leftarrow \mathbf{F_x P F_x^T} + \mathbf{F_n N F_n^T}$$

$$\mathbf{F_x} = \begin{bmatrix} \frac{\partial f_{\mathcal{R}}}{\partial \mathcal{R}} & 0 \\ 0 & \mathbf{I} \end{bmatrix} \qquad \mathbf{F_n} = \begin{bmatrix} \frac{\partial f_{\mathcal{R}}}{\partial \mathbf{n}} \\ 0 \end{bmatrix}$$

Visual EKF SLAM

- Camera prediction model
  - Constant velocity (**MonoSLAM**)

$$\mathcal{R} = \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \\ \mathbf{v} \\ \omega \end{bmatrix} \leftarrow \begin{bmatrix} \mathbf{q} \oplus \mathbf{q}\{\omega \Delta t\} \\ \mathbf{p} + \mathbf{v}\Delta t \\ \mathbf{v} \\ \omega \end{bmatrix}$$



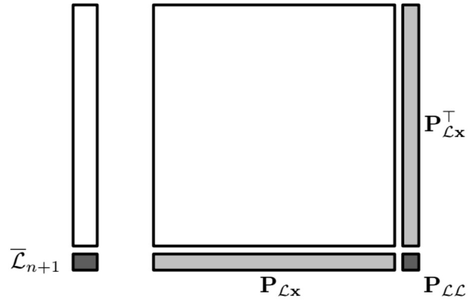
  - Aided with IMU

$$\mathcal{R} = \begin{bmatrix} \mathbf{q} \\ \mathbf{p} \\ \mathbf{v} \end{bmatrix} \leftarrow \begin{bmatrix} \mathbf{q} \oplus \mathbf{q}\{\omega \Delta t\} \\ \mathbf{p} + \mathbf{v}\Delta t \\ \mathbf{v} + (\mathbf{R}_q \mathbf{a} + \mathbf{g})\Delta t \end{bmatrix}$$

  - Aided with odometry

Visual EKF SLAM

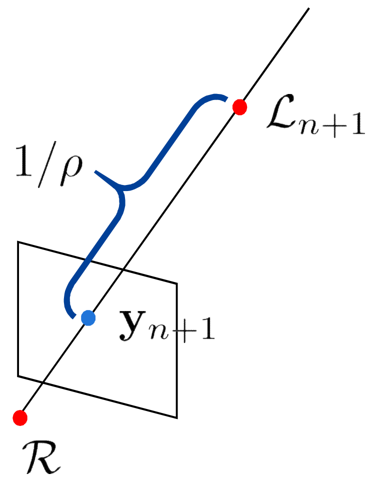
- New landmarks
  - 1. add new variables into the state
  - 2. augment the covariance matrix



- Let the initialization function be

$$\mathcal{L}_{n+1} = g(\mathcal{R}, \mathbf{y}_{n+1}, \rho)$$

  - ρ - inverse-depth
  - $g(\cdot)$ - back projection



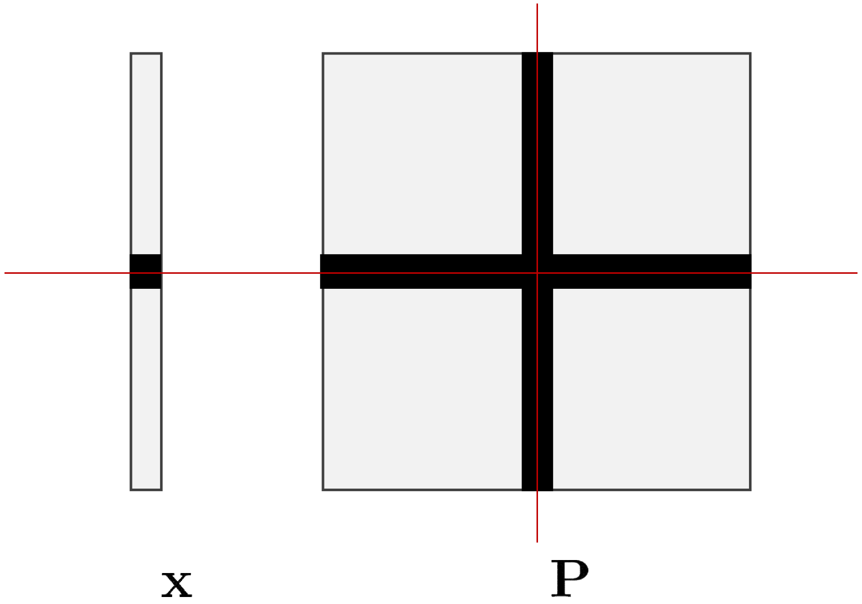
- Update the co-variance and cross-variance

$$\mathbf{P}_{\mathcal{L}_{n+1}\mathcal{L}_{n+1}} = (\frac{\partial g}{\partial \mathcal{R}})\mathbf{P}_{\mathcal{R}\mathcal{R}}(\frac{\partial g}{\partial \mathcal{R}})^T + (\frac{\partial g}{\partial \mathbf{y}_{n+1}})\mathbf{R}(\frac{\partial g}{\partial \mathbf{y}_{n+1}})^T$$

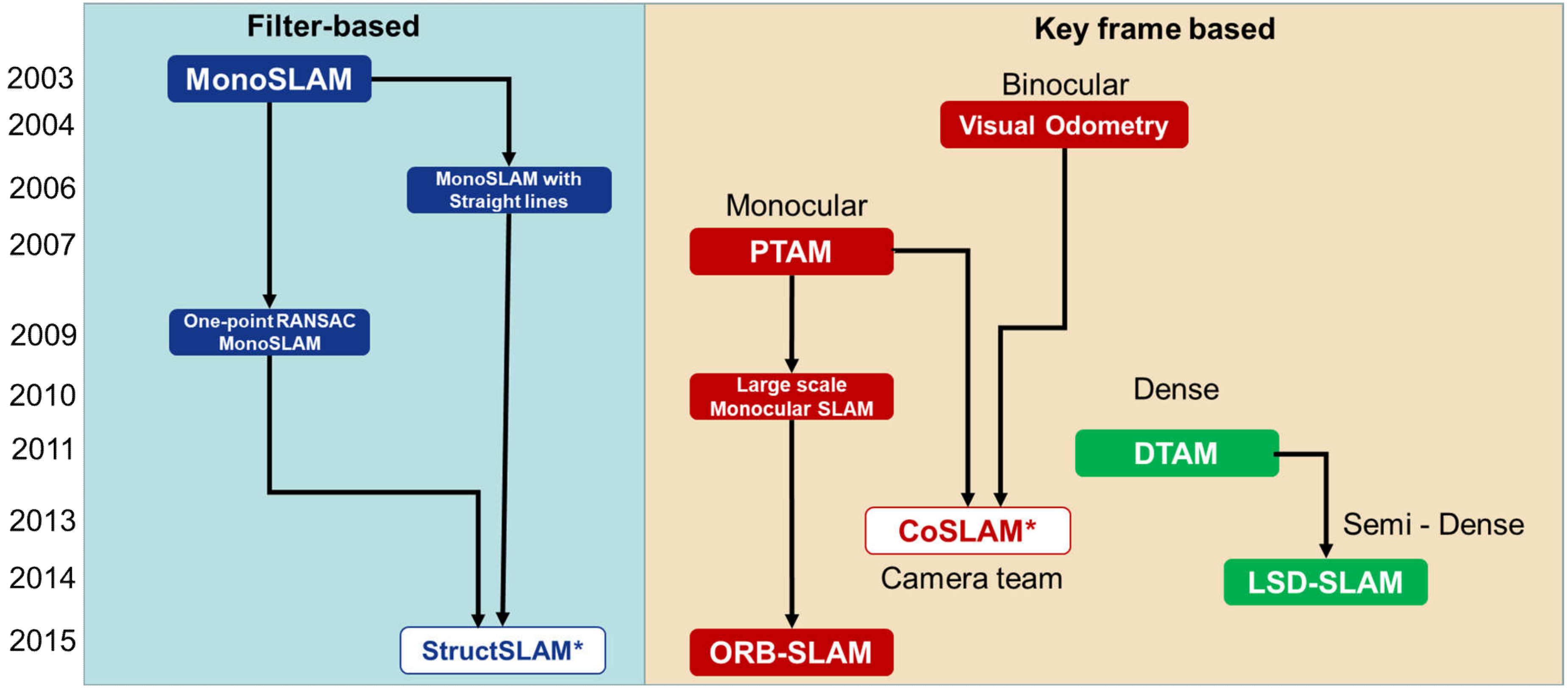
$$\mathbf{P}_{\mathcal{L}\mathbf{x}} = (\frac{\partial g}{\partial \mathcal{R}})\mathbf{P}_{\mathcal{R}\mathbf{x}} = (\frac{\partial g}{\partial \mathcal{R}})[\mathbf{P}_{\mathcal{R}\mathcal{R}}\ \mathbf{P}_{\mathcal{R}\mathcal{M}}]$$

Visual EKF SLAM

- Remove old landmarks from the state and the covariance matrix to keep the dimension number.



Visual SLAM systems – A brief survey



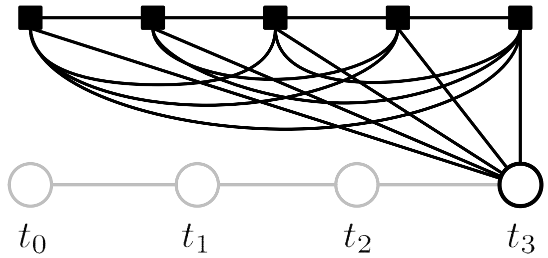
Outline

- ~~Classic SLAM~~
  - ~~SLAM problem~~
  - ~~EKF SLAM~~
  - ~~Visual EKF SLAM – MonoSLAM/StructSLAM~~
- Visual SLAM systems – A brief survey
  - Filter vs keyframe optimization
  - PTAM/ORB-SLAM
  - Direct visual SLAM
    - DTAM/LSD-SLAM/SVO/DSO

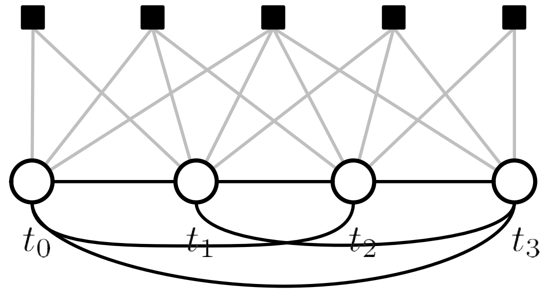
Filter vs Optimization

- Classic EKF SLAM (EKF), Multi-state constrained KF(MSCKF), keyframe Optimization

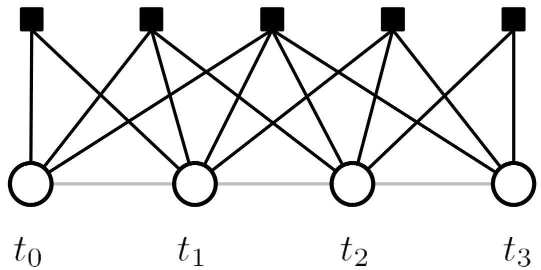
EKF



MSCKF

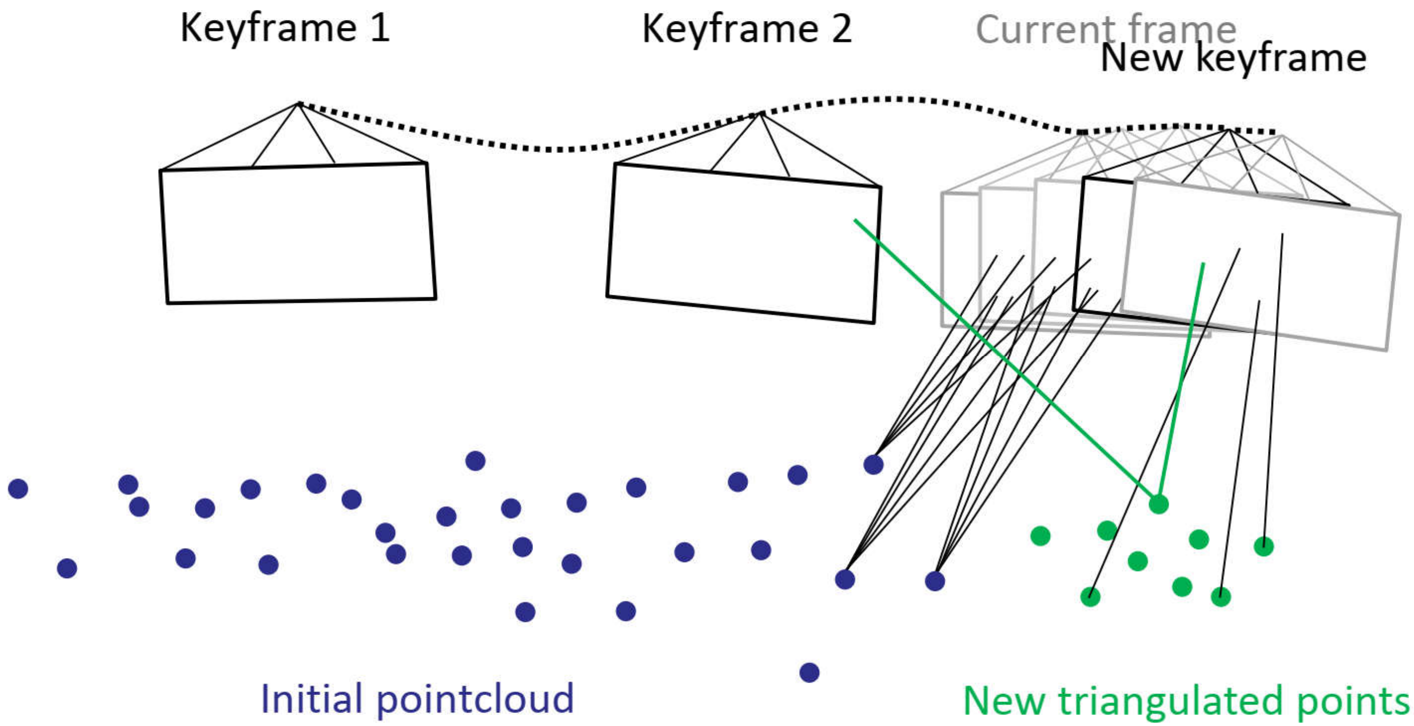


Keyframe Optimization



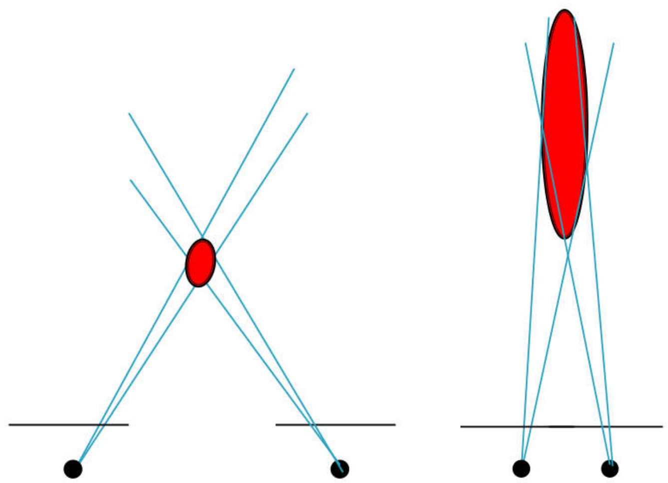
Keyframe-based Visual SLAM

- Similar to incremental Structure-from-Motion



Keyframe-based Visual SLAM

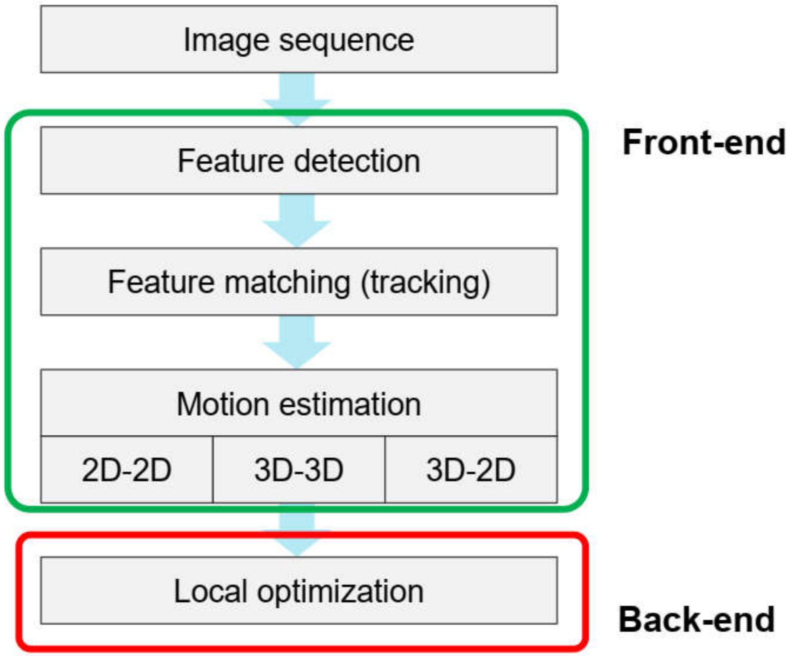
- The key is how to select those key frames :
  - Accuracy
  - Robustness



- A good practice for keyframe selection
  - $\frac{\#\ feature\ points\ tracked}{\#feature\ points\ in\ the\ last\ key\ \ frame} > \text{threshold}\ (e.g.\ 80\%\ )$

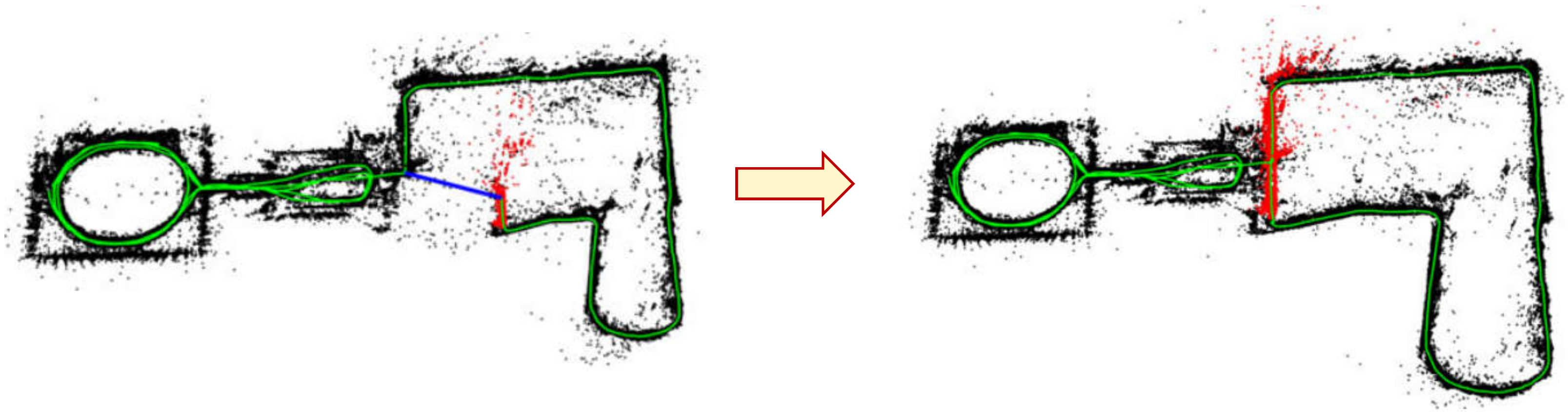
Keyframe-based Visual SLAM

- **Font-End** vs **Back-end**
  - The ***front-end*** is responsible for
    Feature extraction, matching, and outlier removal
    Loop closure detection
  - The ***back-end*** is responsible for the pose and structure optimization (e.g., iSAM, g2o, Google Ceres)



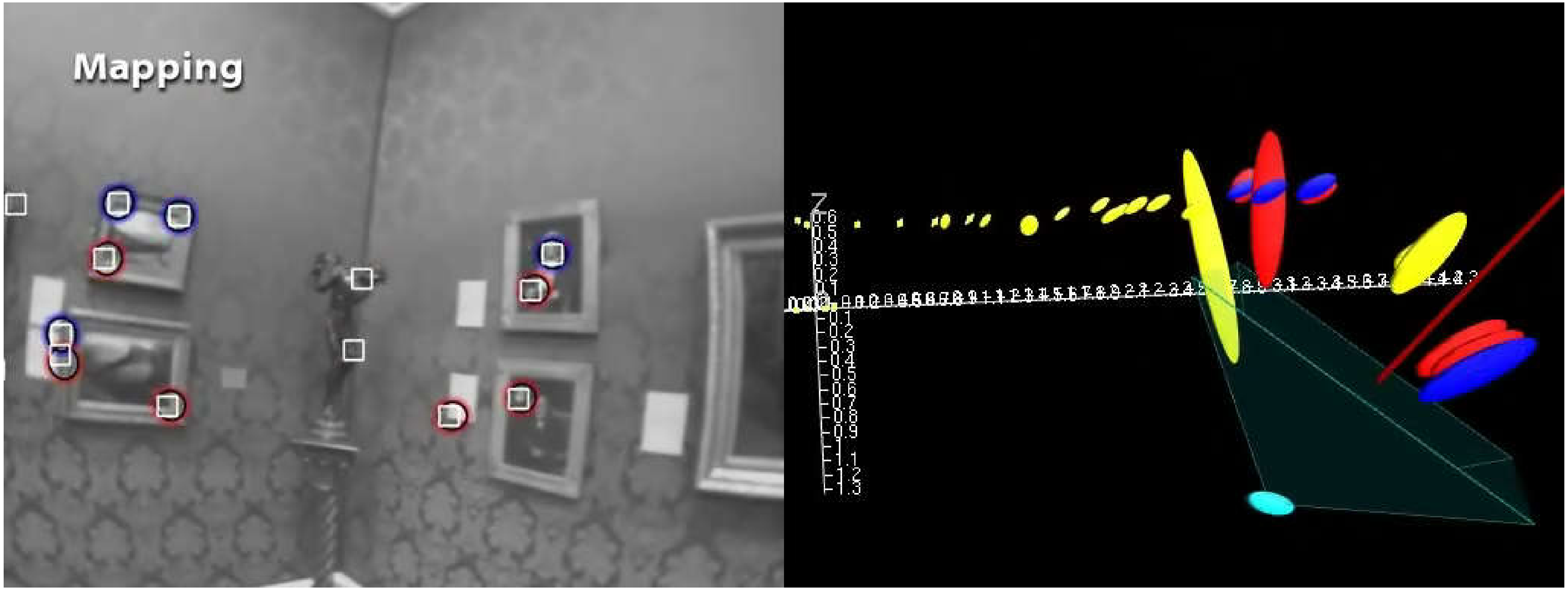
Keyframe-based Visual SLAM

- **Loop closing**
  - Find revisited place (loop closure detection)
  - Eliminate the accumulated error (Pose graph optimization)



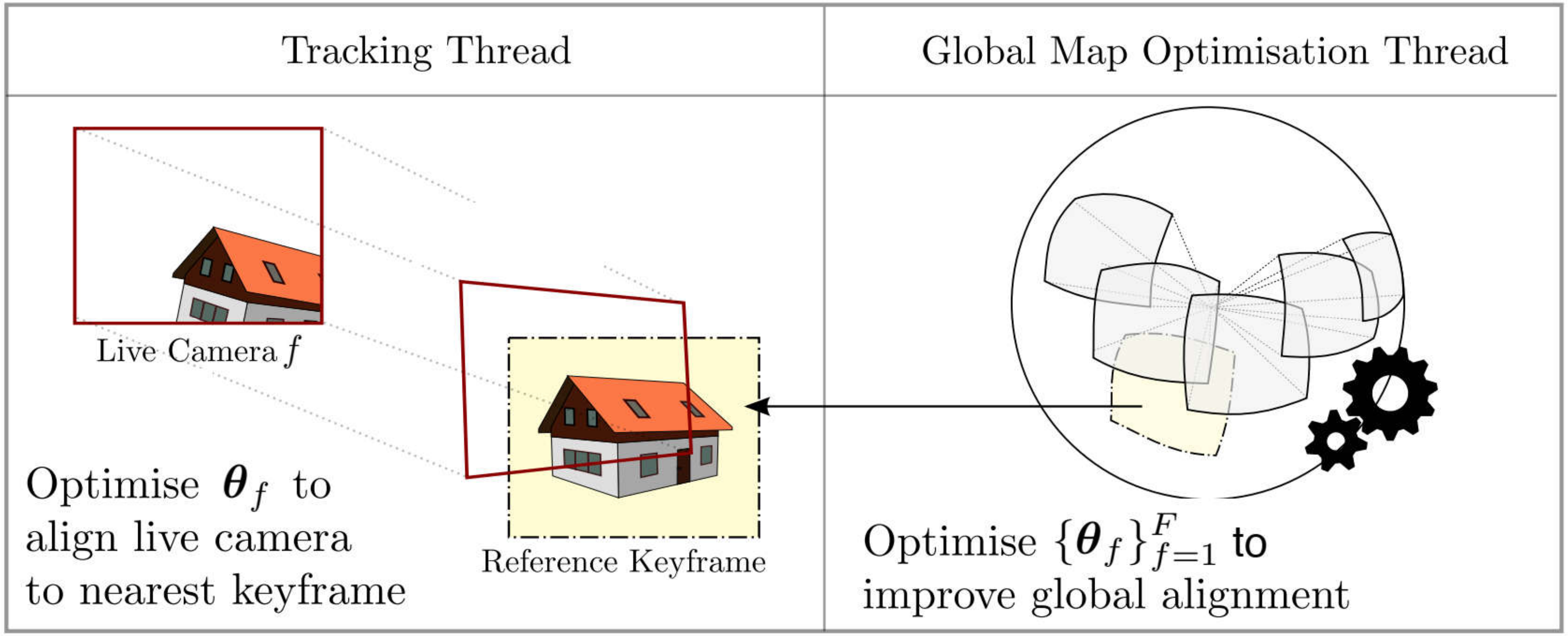
Keyframe-based Visual SLAM

- Other components : **Re-localization**
  - Recover SLAM after occasional failures



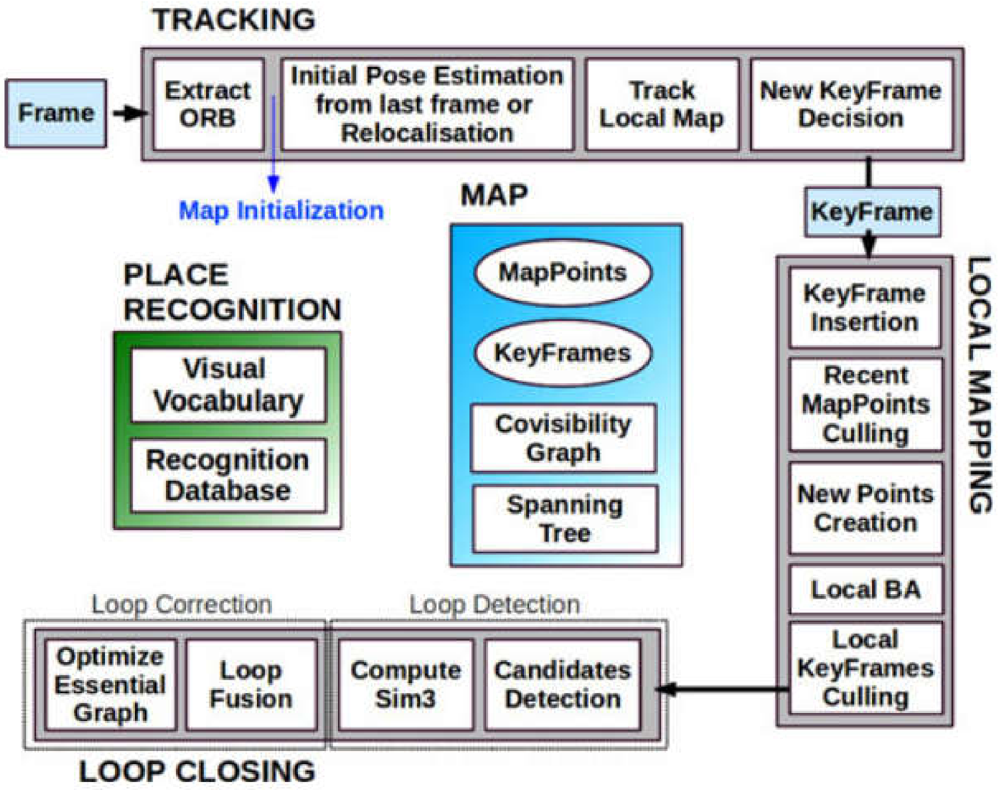
Keyframe-based Visual SLAM

- PTAM(2007)



Keyframe-based Visual SLAM

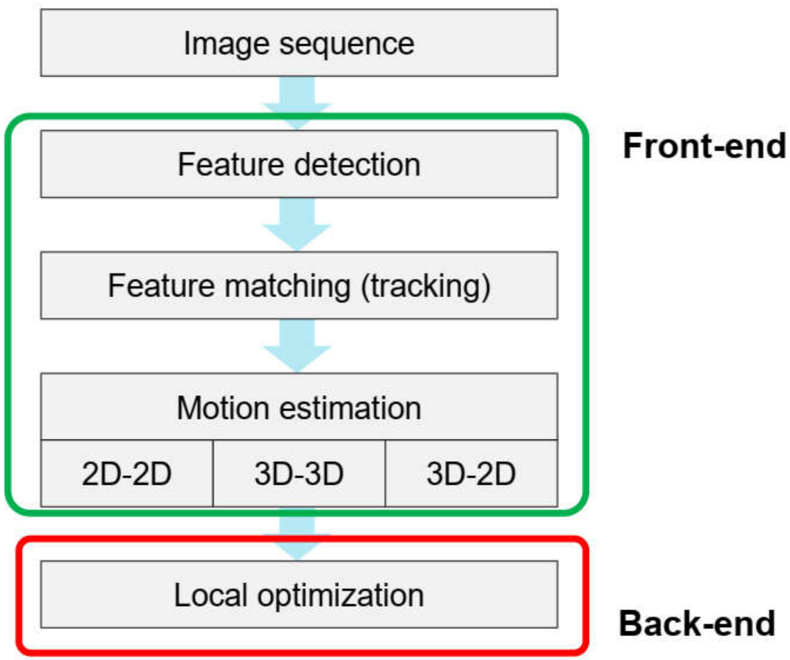
- ORB-SLAM （2015）



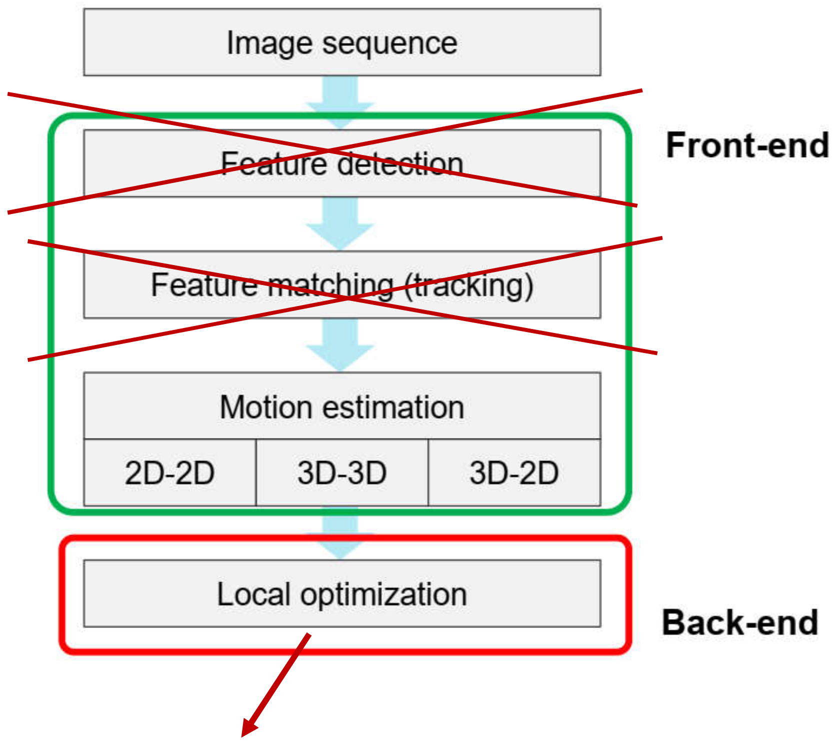
http://webdiis.unizar.es/~raulmur/orbslam/

Direct visual SLAM

- Classic approach - indirect



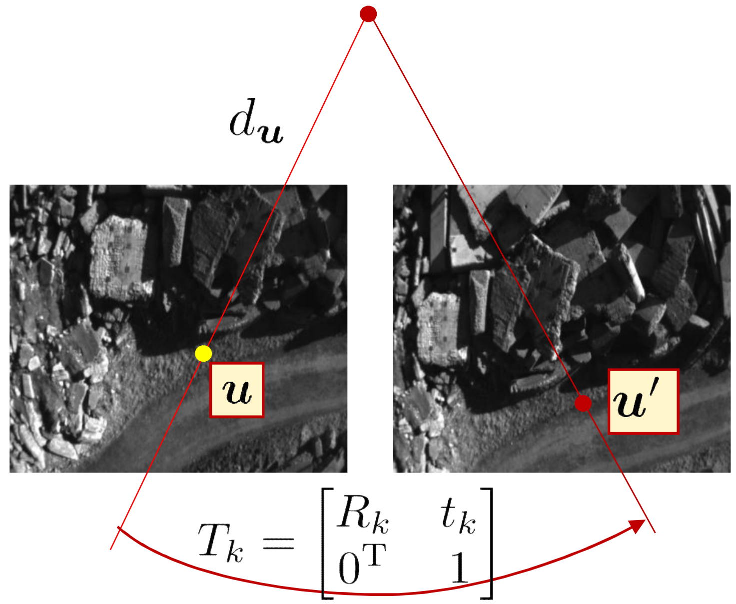
- Direct approach



Without feature points

Direct visual SLAM

- Pose estimation by image alignment - minimize the photometric error.



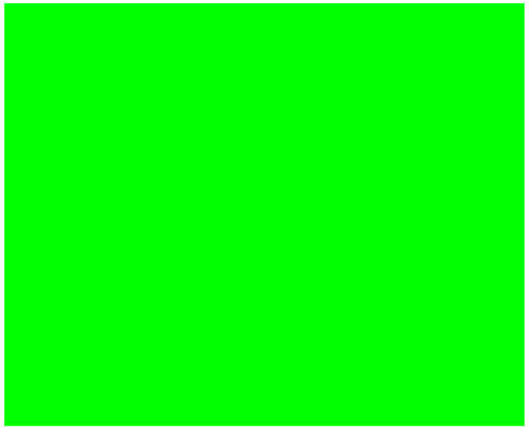
$$T_k = \arg\min_{T} \sum_{u} \rho[I_k(\pi\underbrace{(T \cdot \pi^{-1}(\boldsymbol{u}, d_{\boldsymbol{u}}))}_{\boldsymbol{u}'}) - I_{k-1}(\boldsymbol{u})]$$

Direct visual SLAM

- Pose estimation by image alignment

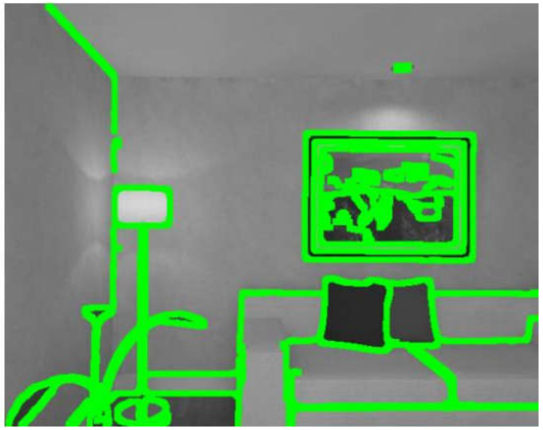
$$T_k = \arg\min_{T} \sum_{u} \rho[I_k(\boldsymbol{u}') - I_{k-1}(\boldsymbol{u})]$$

Dense



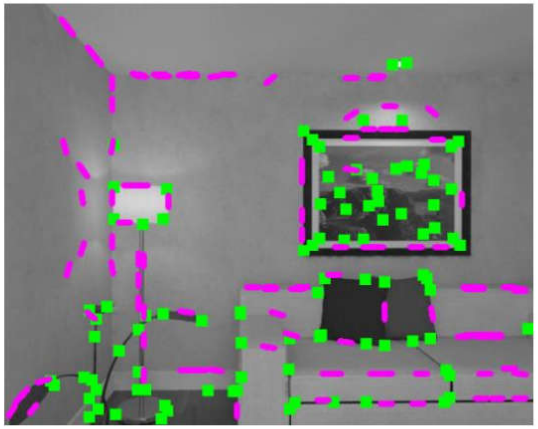
DTAM [Newcombe et al. '11]
300'000+ pixels

Semi-Dense



LSD [Engel et al. 2014]
~10'000 pixels

Sparse



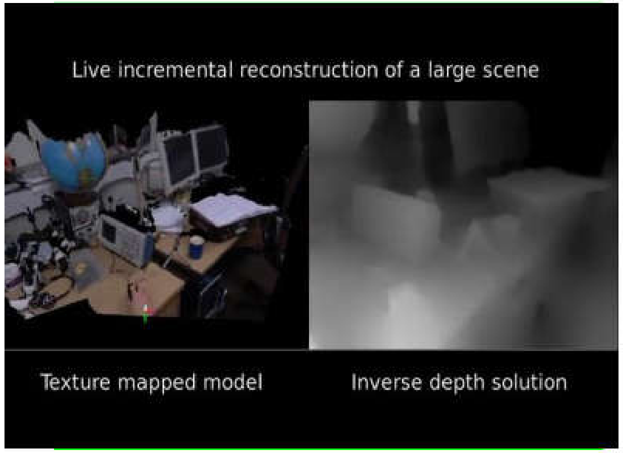
SVO [Forster et al. 2014]
100-200 features x 4x4 patch
~ 2,000 pixels

Direct visual SLAM

- Pose estimation by image alignment

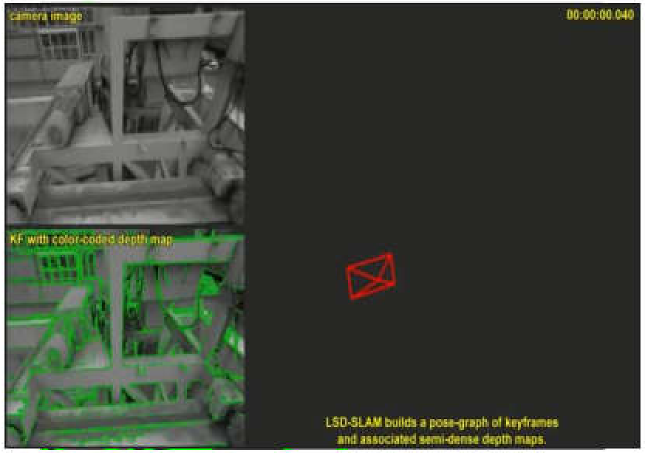
$$T_k = \arg\min_{T} \sum_{u} \rho[I_k(\boldsymbol{u}') - I_{k-1}(\boldsymbol{u})]$$

Dense



DTAM [Newcombe et al. '11]
300,000+ pixels

Semi-Dense



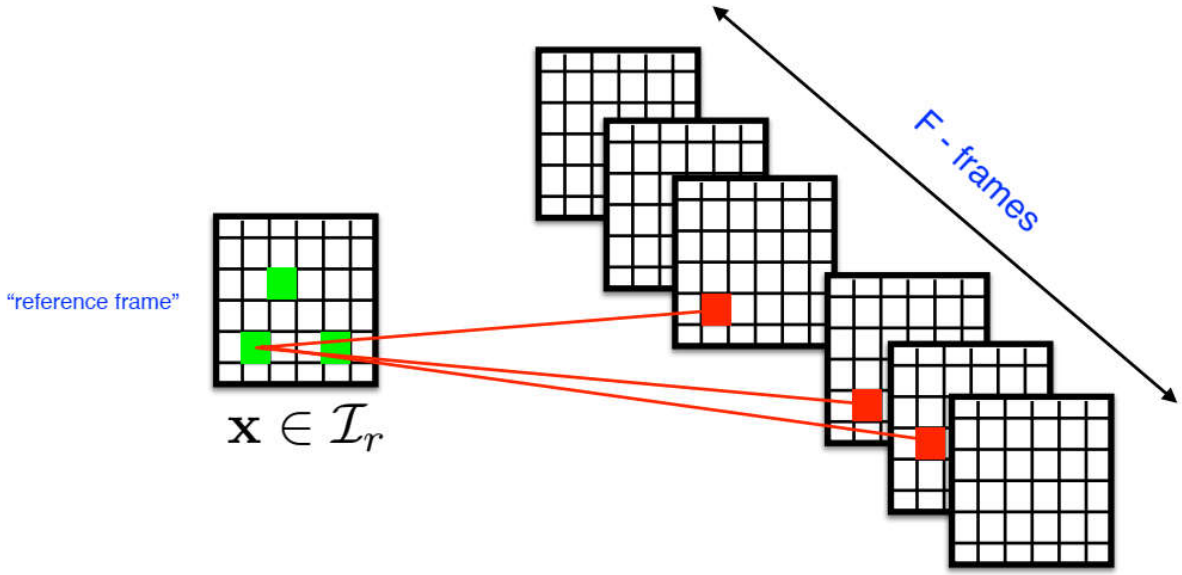
LSD-SLAM [Engel et al. 2014]
~10,000 pixels

Sparse

SVO [Forster et al. 2014]
100-200 features x 4x4 patch
~ 2,000 pixels

Direct visual SLAM

- Back-end : Photometric Bundle adjustment



$$\arg\min_{\boldsymbol{\lambda},\boldsymbol{\theta}} \sum_{r=1}^{F} \sum_{\mathbf{x}\in\mathcal{I}_r} \sum_{f\in\text{obs}(\mathbf{x})} ||\mathcal{I}_r(\mathbf{x}) - \mathcal{I}_f(\mathcal{W}\{\mathbf{x};\boldsymbol{\theta}_f,\lambda_r(\mathbf{x})\})||_2^2$$

DSO - J. Engel, V. Koltun, and D. Cremers. Direct sparse odometry. arXiv preprint arXiv:1607.02565, 2016.
H. Alismail, B. Browning and S. Lucey "Photometric Bundle Adjustment for Vision-based SLAM", ACCV 2016

Direct visual SLAM

- Back-end : Photometric Bundle adjustment