Simultaneous Localization and Mapping

Tsang-Kai Chang
tsangkaichang@ucla.edu

Nov., 2020

Goal

1. present the importance of the SLAM problem
2. highlight the engineering challenges in SLAM systems

Summary

1. Overview of navigation problems
2. Optimization-based SLAM
3. SLAM in visual-inertial systems

Navigation Problems

From the very beginning...

$$\mathbf{s}_{t+1} = f(\mathbf{s}_t, u_t) + \mathbf{w}_t,$$
$$\mathbf{o}_t = h(\mathbf{s}_t, \lambda) + \mathbf{v}_t.$$

- localization:
- SLAM:
- planning:

A simple 2D model

- unicycle model f:

$$\mathbf{s}_{t+1} = \begin{bmatrix} \mathbf{s}_{\theta,t+1} \\ \mathbf{s}_{x,t+1} \\ \mathbf{s}_{y,t+1} \end{bmatrix} = \begin{bmatrix} \mathbf{s}_{\theta,t} + \Delta t(u_{\omega,t} + \mathbf{w}_{\omega,t}) \\ \mathbf{s}_{x,t} + \Delta t \cos(\mathbf{s}_{\theta,t})(u_{v,t} + \mathbf{w}_{v,t}) \\ \mathbf{s}_{y,t} + \Delta t \sin(\mathbf{s}_{\theta,t})(u_{v,t} + \mathbf{w}_{v,t}) \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{s}_{\theta,t} \\ \mathbf{s}_{x,t} \\ \mathbf{s}_{y,t} \end{bmatrix} + \Delta t \begin{bmatrix} 1 & 0 \\ 0 & \cos(\mathbf{s}_{\theta,t}) \\ 0 & \sin(\mathbf{s}_{\theta,t}) \end{bmatrix} \begin{bmatrix} u_{\omega,t} + \mathbf{w}_{\omega,t} \\ u_{v,t} + \mathbf{w}_{v,t} \end{bmatrix}.$$

- bearing-and-range model h:

$$\mathbf{o}_t = \begin{bmatrix} \mathbf{o}_{\phi,t} \\ \mathbf{o}_{r,t} \end{bmatrix} = \begin{bmatrix} \tan^{-1}\left(\frac{\lambda_y - \mathbf{s}_{y,t}}{\lambda_x - \mathbf{s}_{x,t}}\right) - \mathbf{s}_{\theta,t} \\ \sqrt{(\lambda_x - \mathbf{s}_{x,t})^2 + (\lambda_y - \mathbf{s}_{y,t})^2} \end{bmatrix} + \mathbf{v}_t.$$

- We will focus on another more realistic and common model in the second part.

Optimization-based SLAM

Optimization-based SLAM

- SLAM is just an optimization problem!?

HMM Interpretation

- $(\mathbf{s}_{0:T}, \mathbf{o}_{1:T})$ forms a hidden Markov model (HMM).
- The log-likelihood function is now

$$\begin{aligned}&\log p_{\mathbf{s}_{0:T},\mathbf{o}_{1:T}}(s_{0:T}, o_{1:T}; u_{0:T-1}, \lambda) \\ &\quad = \log p_{\mathbf{s}_0}(s_0) + \sum_{t=0}^{T-1} \log p_{\mathbf{s}_{t+1}|\mathbf{s}_t}(s_{t+1}|s_t; u_t) + \sum_{t=1}^{T} \log p_{\mathbf{o}_t|\mathbf{s}_t}(o_t|s_t; \lambda).\end{aligned}$$

- When the noises are zero-mean Gaussian, we have

$$\begin{aligned}\log p_{\mathbf{s}_{t+1}|\mathbf{s}_t}(s_{t+1}|s_t; u_t) &= c_Q - \frac{1}{2}\|s_{t+1} - f(s_t, u_t)\|_Q^2, \\ \log p_{\mathbf{o}_t|\mathbf{s}_t}(o_t|s_t; \lambda) &= c_R - \frac{1}{2}\|o_t - h(s_t, \lambda)\|_R^2,\end{aligned}$$

where Q and R are the covariance matrices of $\mathbf{w}_t$ and $\mathbf{v}_t$, respectively.

Optimization-based SLAM

- Given $o_{1:T}$, the optimal value $(\hat{s}_{0:T}, \hat{\lambda})$ can be considered as the solution of the corresponding maximum likelihood (ML) problem.

- the ML problem = a nonlinear least square (NLS) problem[1]

$$
\begin{aligned}
(\hat{s}_{0:T}, \hat{\lambda}) &= \underset{(s_{0:T},\lambda)}{\arg\max} \log p_{\mathbf{s}_{0:T},\mathbf{o}_{1:T}}(s_{0:T}, o_{1:T}; u_{0:T-1}, \lambda) \\
&= \underset{(s_{0:T},\lambda)}{\arg\max} \sum_{t=0}^{T-1} \log p_{\mathbf{s}_{t+1}|\mathbf{s}_t}(s_{t+1}|s_t; u_t) + \sum_{t=1}^{T} \log p_{\mathbf{o}_t|\mathbf{s}_t}(o_t|s_t; \lambda) \\
&= \underset{(s_{0:T},\lambda)}{\arg\min} \sum_{t=0}^{T-1} \|s_{t+1} - f(s_t, u_t)\|_Q^2 + \sum_{t=1}^{T} \|o_t - h(s_t, \lambda)\|_R^2
\end{aligned}
$$

[1] Or an MAP problem?

Solving NLS problem

- An essential part in early SLAM literature
- We have tools now, g2o or ceres for example.

Graph-based SLAM

- Due to several reasons, λ is not important.
- In Graph SLAM, the log-likelihood of the entire trajectory becomes,

$$\log p_{S_{0:T}, O_{(t,\tau)\in\mathcal{O}_v}}(s_{0:T}, o_{(t,\tau)\in\mathcal{O}_v}; u_{0:T-1}) = \log p_{S_0}(s_0) + \sum_{t=0}^{T-1} \log p_{S_{t+1}|S_t}(s_{t+1}|s_t; u_t) + \sum_{(t,\tau)\in\mathcal{O}_v} \log p_{O_{(t,\tau)}|S_t,S_\tau}(o_{(t,\tau)}|s_t, s_\tau)$$

- And it leads to a NLS problem again:

$$\begin{aligned}\hat{s}_{1:T} &= \arg\max_{s_{1:T}} \log p_{S_{0:T}, O_{(t,\tau)\in\mathcal{O}_v}}(s_{0:T}, o_{(t,\tau)\in\mathcal{O}_v}; u_{0:T-1}) \\ &= \arg\max_{s_{1:T}} \sum_{t=0}^{T-1} \|s_{t+1} - f(s_t, u_t)\|_Q^2 + \sum_{(t,\tau)\in\mathcal{O}_v} \|o_{t,\tau} - h'(s_t, s_\tau)\|_{R'}^2,\end{aligned}$$

About nomenclature

- maybe explicit-landmark and implicit-landmark?

SLAM in Visual-Inertial Systems

SLAM in VI system

- Commonly, SLAM systems use inertial measurement units (IMUs) as proprioceptive sensors, and cameras as exteroceptive sensors.
- I will talk about the essential knowledge to build such SLAM systems.
- Definitely not the only method to do so.

Architecture

- frontent
- backend

Rigid body transformation

- Different measurements are taken in different frames:
  - navigation frame (n) / world frame (w)
  - body frame (b)
  - camera frame (c)

- We have to know how to transform measurements between these frames.

Orientation parameterization

- Euler angle
  - Gimbal lock

- rotation matrix

$$RR^{\mathsf{T}} = R^{\mathsf{T}}R = I_3$$
$$\text{det}R = 1.$$

- unit quaternion

$$q = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^{\mathsf{T}} = \begin{bmatrix} q_0 \\ q_v^{\mathsf{T}} \end{bmatrix} \in \mathbb{R}^4, \quad ||q||_2 = 1.$$

Time dynamic model

- IMU = gyroscope + accelerometers
- IMU measurements

$$u^{nb}_{\omega,t} = \omega^{nb}_t + \delta_{\omega,t} + \eta_{\omega,t},$$
$$u_{a,t} = (R^{nb}_t)^{\mathsf{T}}(a_t - g) + \delta_{a,t} + \eta_{a,t}.$$

- time dynamic model

$$s^{nb}_{q,t+1} = s^{nb}_{q,t} \odot \exp_{\mathsf{q}}\left(\Delta t(u_{\omega,t} - \delta_{\omega,t} - \eta_{\omega,t})\right),$$
$$s_{v,t+1} = s_{v,t} + \Delta t\left(R_{\mathsf{q}}(s^{nb}_{q,t})(u_{a,t} - \delta_{a,t} - \eta_{a,t}) + g\right),$$
$$s_{p,t+1} = s_{p,t} + \Delta t\, s_{v,t} + \frac{\Delta t^2}{2}\left(R_{\mathsf{q}}(s^{nb}_{q,t})(u_{a,t} - \delta_{a,t} - \eta_{a,t}) + g\right).$$

IMU preintegration

- Normally, imu rates are much higher then the observation rate.
- In practice, this causes some numerical difficulties.

$$\Delta R_{t,t+L} \doteq R_t^{\mathsf{T}} R_{t+1} = \prod_{l=0}^{L-1} \exp\left(\Delta t(u_{\omega,t+l} - \delta_{\omega,t+l} - \eta_{\omega,t+l})\right),$$

$$\Delta v_{t,t+L} \doteq R_t^{\mathsf{T}}(v_{t+L} - v_t - g\Delta t_L) = \sum_{l=0}^{L-1} \Delta t\, \Delta R_{t,t+l}\left(u_{a,t_l} - \delta_{a,t+l} - \eta_{a,t+l}\right)$$

$$\begin{aligned}\Delta p_{t,t+L} &\doteq R_t^{\mathsf{T}}\left(p_{t+L} - p_t - v_t \Delta t_L - \frac{1}{2} g \Delta t_L^2\right) \\ &= \sum_{l=0}^{L-1}\left[\Delta t \Delta v_{t,t+l} + \frac{\Delta t^2}{2} \Delta R_{t,t+l}(u_{a,t+l} - \delta_{a,t+l} - \eta_{a,t+l})\right].\end{aligned}$$

Observation model

- pinhole camera model

$$o_{m,t} = \pi(T^{cn}(\lambda_m)) + v_{m,t}$$

$$\pi\left([x, y, z]^\mathsf{T}\right) = \begin{bmatrix} f_u & 0 \\ 0 & f_v \end{bmatrix} \begin{bmatrix} x/z \\ y/z \end{bmatrix} + \begin{bmatrix} p_u \\ p_v \end{bmatrix}.$$

- More realistic models should be used, including distortion.

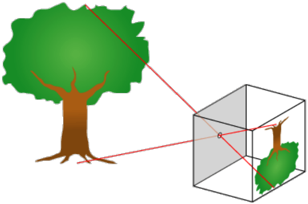
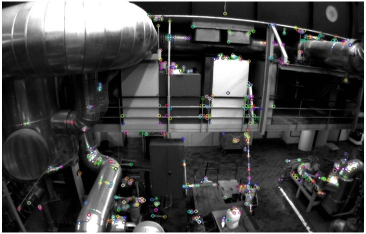


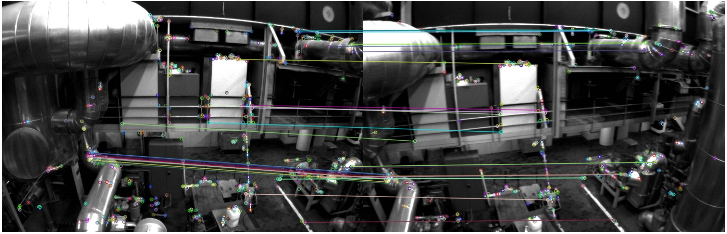
Figure: Pinhole camera model. [Wiki]

Feature extraction and matching

- Extract features first



- then find the matching between features from different images



Optimization over a manifold

- Even though we can use different parameterization, there are only 3 degrees of freedom for orientation.
- We need to provide the Jacobian matrices, which stand for the variation to the global parameterization from the local parameterization.
- different noise modeling
- tedious!

Summary

Summary

- There are several open-source SLAM projects: ORB SLAM, RGB-D SLAM, okvis, etc.
- Build a SLAM system on your own!
- How can SLAM be integrated into the entire navigation system?