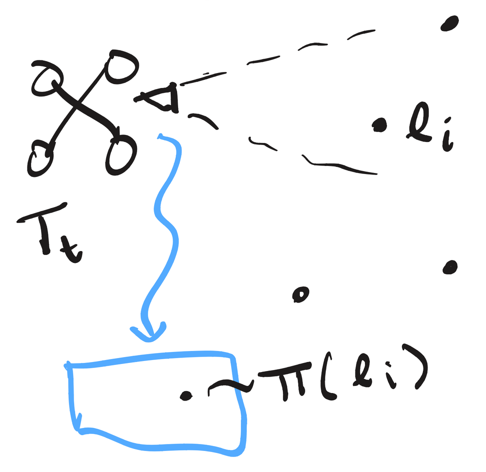
4/9: Factor graph SLAM

Announcements:
- HW 4 out, due 4/23
- project workshop 4/28

Last time
- factor graphs

Today:
- factor graph SLAM

e.g.) (quadrotor SLAM)



states/poses: $T_t \in SE(3)$
measurements:
- IMU: acceleration + angular vel
- camera: $I_t \in \mathbb{R}^{H \times W \times 3}$
- landmarks: $l_i \in \mathbb{R}^3$

Goal: optimize for both T_t, l_i

[ORB-SLAM3]
[GTSAM, Daellart]

SLAM / VIO (visual-inertial odometry)

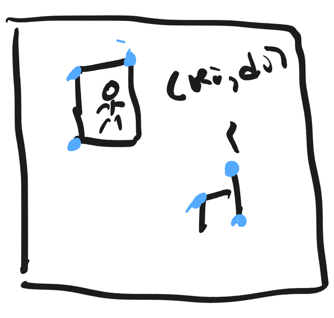
"front end" → "back end"

"front end"
- detects landmarks in the image
- matching landmarks
- map maintenance

"back end"
- sparse Gauss-Newton

Q: How does the front end work?

1) Feature detector



want to find visually distinguishable points
- intensity changes
- edge/corner detectors

in practice: - SIFT, ORB
- superpoint

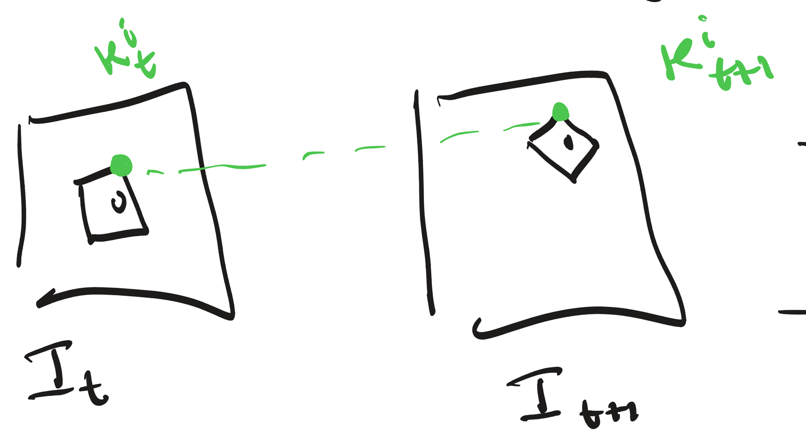
$I_t \mapsto (\kappa_t^i, d_i)$

(u, v) pixel coords; descriptor

what is "good"? between two frames, return the same points, same descriptors

$\kappa_t^i = \underbrace{\pi}_{\text{projection}}(T_t^{-1} \cdot l_i)$ putting landmark into body frame, projecting

2) feature matching



- heuristically using descriptor
- superglue

what this gives us:

given $I_t \mapsto (\kappa_t^i, i)$

keypoints; index (matching)

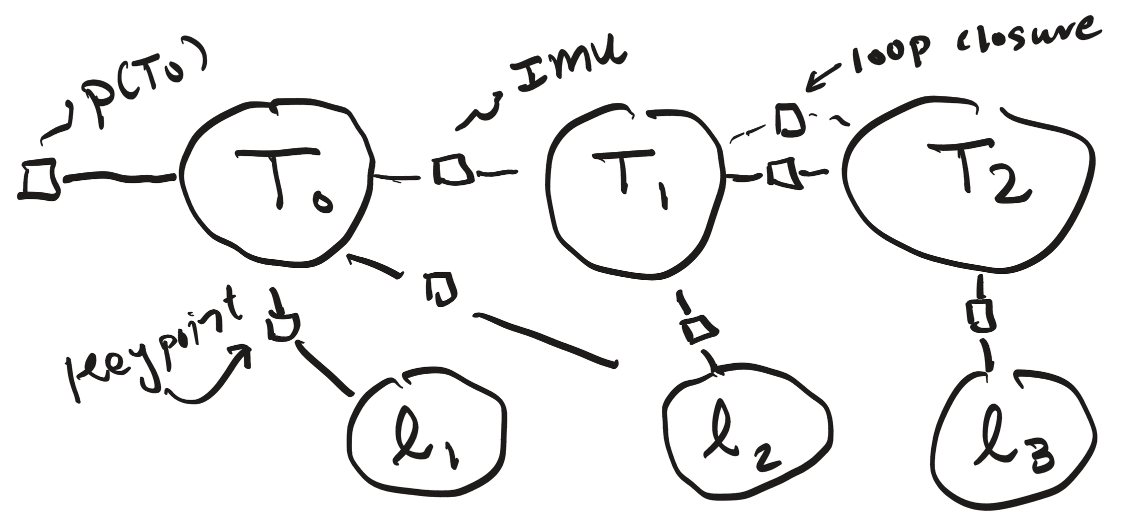
Adversarial cases:
i) Featureless / shiny environments
ii) Repetition

Many other considerations
✓ loop closures
✓ map merges
✓ adding / dropping landmarks

What about the IMU?
one issue: IMU reads much faster than the camera

"IMU preintegration"
aggregates lots of IMU measurements to downsample to camera freq.

$\Delta T_t = \log(T_t^{-1} \cdot T_{t+1})$, Q_t



Factor graph:

Putting it together: The MAP problem

$$\min_{T_{0:H},\, l_{1:N}} \sum_t \| \Delta T_t - \log(T_t^{-1} \cdot T_{t+1}) \|^2_{Q_t^{-1}}$$

IMU

$$+ \sum_{i \in \mathcal{I}_t} \| \kappa_t^i - \pi(T_t^{-1} \cdot l_i) \|^2_{R_t^{-1}}$$

keypoints (reprojection error)

$\|r\|^2_{\Sigma^{-1}} = \frac{1}{2} r^T \Sigma^{-1} r$, weighted norm

$\mathcal{I}_t = \{ i \mid l_i \text{ seen in image } t \}$