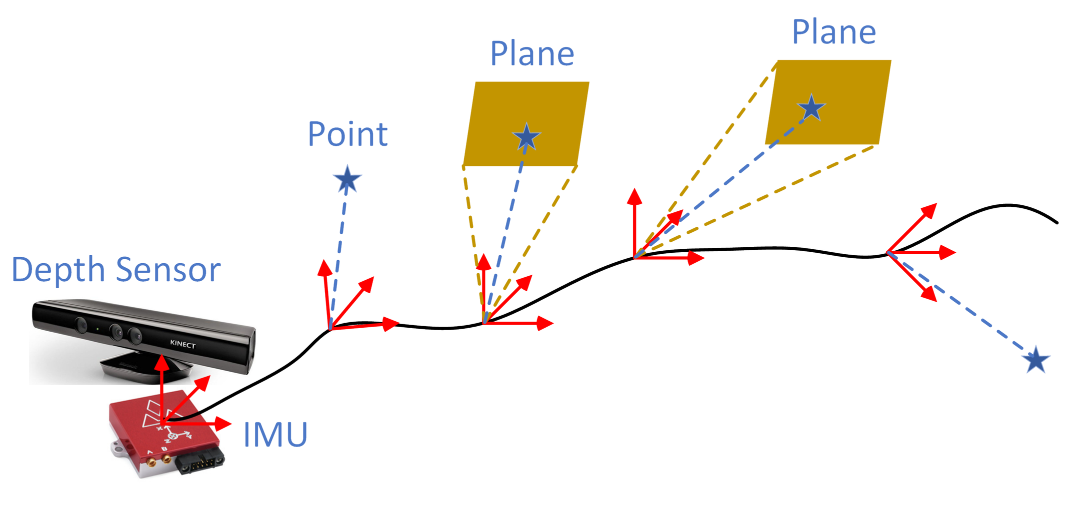
Tightly-Coupled Aided Inertial Navigation with Point and Plane Features

Y. Yang, P. Geneva, X. Zuo,
K. Eckenhoff, Y. Liu and G. Huang

RPNG

Motivation

- Aided inertial navigation is one of the most popular 6DOF pose estimation methods.
- Geometric features (point and plane) should be used for the aided INS.
- Geometric constraints (i.e. point-on-plane) should be exploited to improve estimator.



Aided inertial navigation system with point and plane features. Note that some points are lying on the planes.

Contributions

- A tightly-coupled estimator for aided INS with both point and plane features, applicable to a vision sensor along with a generic depth sensor.
- We detect planar point features and enforce point-on-plane constraints in estimation.
- We introduce a simple but effective plane feature initialization approach for state estimation.

System Model

- State vector:

$$\mathbf{x}_I = \begin{bmatrix} {}^I_G\bar{q}^\top & \mathbf{b}_g^\top & {}^G\mathbf{v}_I^\top & \mathbf{b}_a^\top & {}^G\mathbf{p}_I^\top \end{bmatrix}^\top$$

$$\mathbf{x}_{calib} = \begin{bmatrix} {}^D_I\bar{q}^\top & {}^D\mathbf{p}_I^\top \end{bmatrix}^\top$$

$$\mathbf{x}_{feat} = \begin{bmatrix} {}^G\mathbf{p}_\mathbf{f}^\top & {}^G\mathbf{p}_\pi^\top \end{bmatrix}^\top$$

- Generic point measurement

$$\mathbf{z}_p = \begin{bmatrix} z^{(r)} \\ \mathbf{z}^{(b)} \end{bmatrix} = \begin{bmatrix} \sqrt{{}^I\mathbf{P}_\mathbf{f}^\top {}^I\mathbf{P}_\mathbf{f}} + n^{(r)} \\ \mathbf{h_b}\left({}^I\mathbf{P}_\mathbf{f}, \mathbf{n}^{(b)}\right) \end{bmatrix}$$

- Plane measurement

$$\mathbf{z}^{(\pi)} = {}^D\mathbf{p}_\pi + \mathbf{n}^{(\pi)} = {}^D d_\pi {}^D\mathbf{n}_\pi + \mathbf{n}^{(\pi)}$$

$$\begin{bmatrix} {}^D\mathbf{n}_\pi \\ {}^D d_\pi \end{bmatrix} = \begin{bmatrix} {}^D_I\mathbf{R} & \mathbf{0}_{3\times1} \\ {}^D\mathbf{p}_I^\top {}^D_I\mathbf{R} & 1 \end{bmatrix} \begin{bmatrix} {}^I_G\mathbf{R} & \mathbf{0}_{3\times1} \\ -{}^G\mathbf{p}_I^\top & 1 \end{bmatrix} \begin{bmatrix} {}^G\mathbf{n}_\pi \\ {}^G d_\pi \end{bmatrix}$$

CP Plane Feature

- Plane extraction from point cloud:

$$d_i = \frac{{}^D\mathbf{p}_\pi^\top {}^D\mathbf{p}_{\mathbf{f}mi}}{\left\|{}^D\mathbf{p}_\pi\right\|} - \left\|{}^D\mathbf{p}_\pi\right\|$$

$$\arg\min_{{}^D\mathbf{p}_\pi} \sum_{i=1}^{m} \|d_i\|^2_{\mathbf{R}_{di}^{-1}}$$

- Plane data association:

$$r_m = \left(\tilde{\mathbf{z}}^{(\pi)}\right)^\top \left(\mathbf{H}_\pi \mathbf{P}_{k|k} \mathbf{H}_\pi^\top + \mathbf{R}_\pi\right)^{-1} \tilde{\mathbf{z}}^{(\pi)}$$

- Mahalanobis distance test. If the distance smaller than a lower threshold, this plane will be associated to a current plane. If larger than a high threshold, a new plane will be initialized.

Plane Feature Initialization

- Both the initial estimate and covariance of plane feature need to be initialized for KF algorithm:

$$\mathbf{z}^{(\pi)} = \mathbf{h}\left(\begin{bmatrix} \mathbf{x}_k \\ {}^G\mathbf{p}_\pi \end{bmatrix}\right) + \mathbf{n}^{(\pi)}$$

- An MLE can be formulated as:

$$\min_{\mathbf{x}_k, {}^G\mathbf{p}_\pi} \left\| \mathbf{z}^{(\pi)} - \mathbf{h}\left(\begin{bmatrix} \mathbf{x} \\ {}^G\mathbf{p}_\pi \end{bmatrix}\right) \right\|^2_{\mathbf{R}_\pi^{-1}} + \left\| \begin{bmatrix} \tilde{\mathbf{x}} \\ {}^G\tilde{\mathbf{p}}_\pi \end{bmatrix} \right\|^2_{\mathbf{P}_{k+1|k}^{-1}}$$

- The initialized estimate and covariance:

$$\begin{bmatrix} \tilde{\mathbf{x}} \\ {}^G\tilde{\mathbf{p}}_\pi \end{bmatrix} = \left(\mathbf{P}_{k+1|k+1}\right)^{-1} \begin{bmatrix} \mathbf{H}_x^\top \\ \mathbf{H}_\mathbf{f}^\top \end{bmatrix} \mathbf{R}_\pi^{-1} \tilde{\mathbf{z}}^{(\pi)} = \begin{bmatrix} \mathbf{0} \\ \mathbf{H}_\mathbf{f}^{-1} \tilde{\mathbf{z}}^{(\pi)} \end{bmatrix}$$

$$\mathbf{P}_{k+1|k+1} = \begin{bmatrix} \mathbf{P}_{k|k} & -\mathbf{P}_{k|k}\mathbf{H}_x^\top \mathbf{H}_\mathbf{f}^{-1} \\ -\mathbf{H}_\mathbf{f}^{-\top}\mathbf{H}_x\mathbf{P}_{k|k} & \mathbf{H}_\mathbf{f}^{-1}\left(\mathbf{R}_\pi + \mathbf{H}_x\mathbf{P}_{k|k}\mathbf{H}_x^\top\right)\mathbf{H}_\mathbf{f}^{-\top} \end{bmatrix}$$

Point-on-Plane

- Exploit the structural constraints from the environments: pt-on-plane:

$$\mathbf{g}(\mathbf{x}) := \frac{\mathbf{p}_\mathbf{f}^\top \mathbf{p}_\pi}{\|\mathbf{p}_\pi\|} - \|\mathbf{p}_\pi\| = 0$$

- The constraints can be treated as extra cost term added to the estimator as:

$$\|\mathbf{g}(\mathbf{x})\|^2_{\sigma_g^{-2}}$$

- Pt-on-plane constraints detection are based on the pt to plane distance:

$$d_m = \frac{{}^D\mathbf{p}_{\pi m}^\top {}^D\mathbf{p}_\mathbf{f}}{\left\|{}^D\mathbf{p}_{\pi m}\right\|} - \left\|{}^D\mathbf{p}_{\pi m}\right\|$$

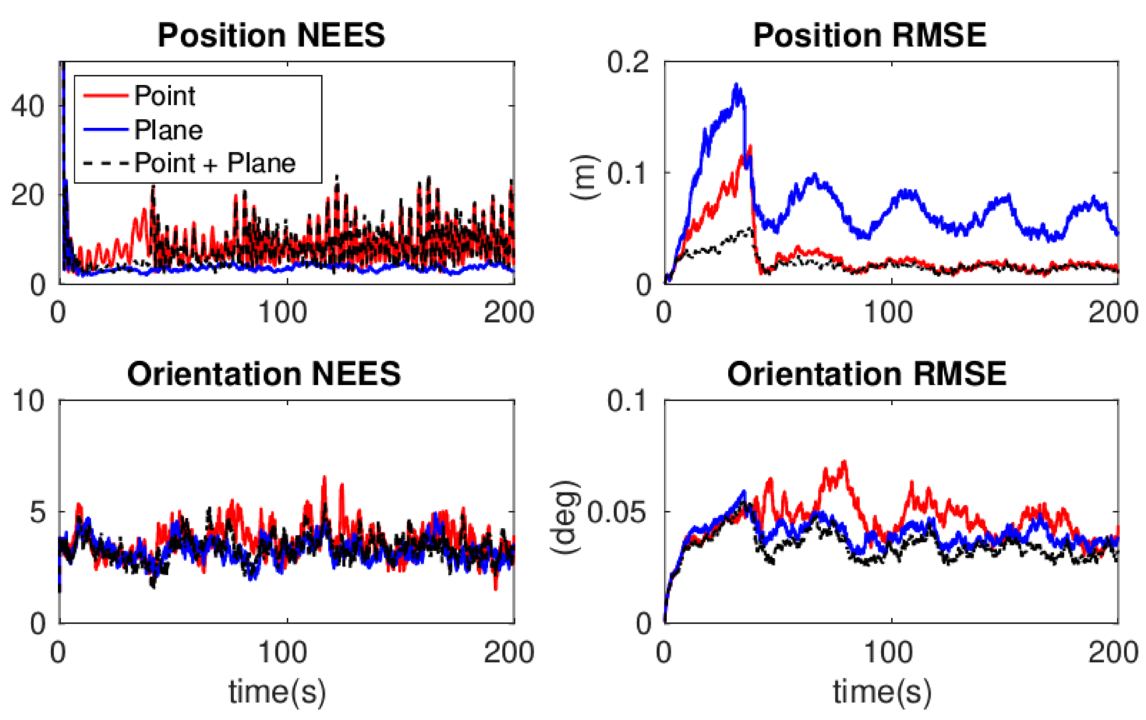
- Mahalanobis distance will be computed based the above equation:

$$r_p = d_m^\top \left(\mathbf{H}_{mx}\mathbf{P}_{k|k}\mathbf{H}_{mx}^\top + \mathbf{H}_{mn}\mathbf{R}_\pi\mathbf{H}_{mn}^\top\right)^{-1} d_m$$

- If the pt-on-plane constraint is accepted, this constraint will be treated as a measurement into the estimator.

Simulation

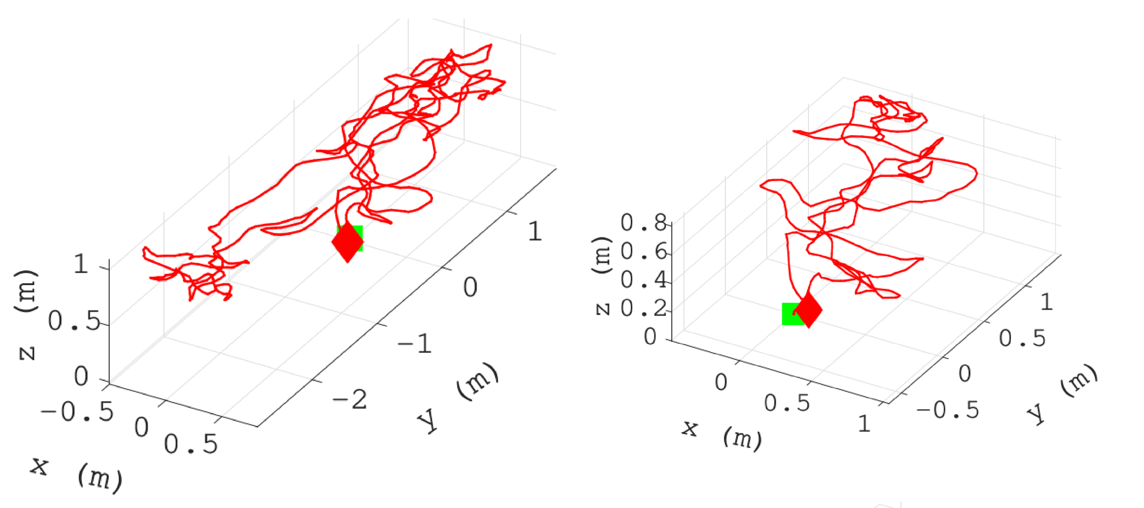
- Monte-Carlo simulations



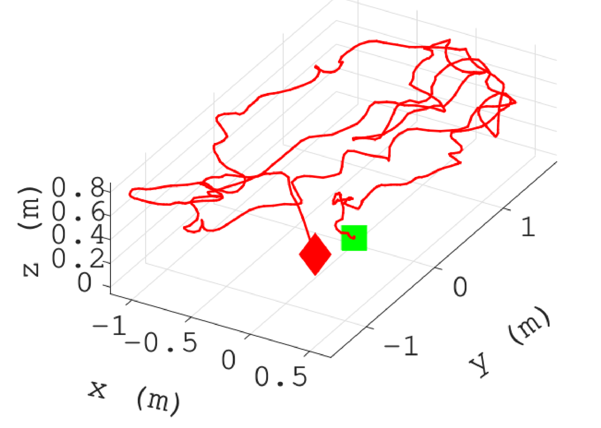
Experiments

Results

Unit (m)	Trajectory 1	Trajectory 2	Trajectory 3
MSCKF+Plane	0.2682	0.2607	0.8432
MSCKF+Pt+Plane	0.0539	0.1113	0.3608
MSCKF+Pt-On-Plane	0.0461	0.1095	0.3363



- Pt+plane is better than plane only.
- Pt-on-plane can improve the accuracy.



Future Work

- Improve the plane matching algorithm.
- Based on observability constrained EKF improve the consistency.