Visual Inertial SLAM

Zhexu Li

Introduction

Simultaneous Localization and Mapping (SLAM) is one of the fundamental problems in robotic navigation. An accurate, fast, and reliable SLAM algorithm has great potential in many industries including robotics, security, and transportation. Despite the current research focus has shifted to deep learning based approaches because of the rapid advance in neural network architectures, traditional extended Kalman filter (EKF) based SLAM algorithms are still valuable because they require less computational resources than deep learning models to train, not to mention understanding the intuitions behind these models will provide a solid foundation for learning more advanced models. In this project, I implement an EKF prediction step based on SE(3) kinematics with IMU measurements and an EKF update step based on the stereo-camera observation model with feature observations to perform localization and mapping. Then I combine the IMU prediction with the landmark update and implement an IMU update step based on the stereo-camera observation model to obtain a complete visual-inertial SLAM algorithm. I deploy the algorithm on an autonomous vehicle dataset to create a map of vehicle trajectory and landmarks. The resulting algorithm could potentially be used in many scenarios including robotic navigation, and autonomous mapping.

Problem Formulation

IMU Localization via EKF Prediction

Let $T_t \in SE(3)$ be the vehicle pose at time t, given z_t (the observation at time t), and u_t (the control input at time t), the goal is to estimate the state of the vehicle T_t using EKF, which assumes the prior pdf $p_{t\,|\,t}$is Gaussian, which means:

$$T_t | \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \Sigma_{t|t})$$

And The goal of the EKF Prediction process is to obtain the predicted mean $\mu_{t+1\,|\,t}$ and the predicted variance $\Sigma_{t+1\,|\,t}$ using the motion model p_f, for this project I use Discrete-time rotation kinematics.

Landmark Mapping via EKF Updates

Given the IMU pose $T_{t+1} := wTi,\ t\ \in\ SE(3)$, the new features observation z_{t+1}, the goal is to estimate the coordinates of landmarks:

$$\mathbf{m} := \begin{bmatrix}\mathbf{m}_1^\top & \cdots & \mathbf{m}_M^\top\end{bmatrix}^\top \in \mathbb{R}^{3M}$$

which generated the observations.

And the EKF updates step returns the updated mean $\mu_{t+1\,|\,t+1}$ and the updated variance $\Sigma_{t+1\,|\,t+1}$.

Visual-Inertial SLAM

Given a sequence of control inputs $u_{0:T}$ and a sequence of features observations $z_{0:t}$, the goal is to estimate the pose of the vehicle:

$$T_t|\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \Sigma_{t|t})$$

and the coordinates of landmarks:

$$\mathbf{m} := \begin{bmatrix}\mathbf{m}_1^\top & \cdots & \mathbf{m}_M^\top\end{bmatrix}^\top \in \mathbb{R}^{3M}$$

over time 0, ……., T, using the prediction and updates steps mentioned above.

<u>Technical Approach</u>

IMU Localization via EKF Prediction

Given z_t (the observation at time t), u_t(the control input at time t), and the prior mean and covariance, the state of the vehicle T_t estimated using EKF is given by:

$$T_t|\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \Sigma_{t|t})$$

where the predicted mean $\mu_{t+1\,|\,t}$ and variance $\Sigma_{t+1\,|\,t}$ are:

$$\boldsymbol{\mu}_{t+1|t} = \boldsymbol{\mu}_{t|t} \exp\left(\tau_t \ddot{\mathbf{u}}_t\right)$$

$$\Sigma_{t+1|t} = \mathbb{E}[\delta\boldsymbol{\mu}_{t+1|t}\delta\boldsymbol{\mu}_{t+1|t}^\top] = \exp\left(-\tau\overset{\curlywedge}{\mathbf{u}}_t\right)\Sigma_{t|t}\exp\left(-\tau\overset{\curlywedge}{\mathbf{u}}_t\right)^\top + W$$

where W is the motion noise.

The implemented algorithm, predict(v, w, tau, uto, sto), intakes linear velocity v, angular velocity w, time discretization tau, mean at time t (uto), and variance at time t (sto), and it returns the predicted mean $\mu_{t+1|t}$ and the predicted variance $\Sigma_{t+1|t}$ calculated using the formulas above. This algorithm is enough for the IMU localization using EKF prediction part of the project.

Landmark Mapping via EKF Updates

Given the IMU pose $T_{t+1} \in SE(3)$, new features observations z_{t+1} , and the prior mean and covariance, the coordinates of landmarks **m** (described above) can be retrieved by estimating the mean $\mu_{t+1} \in \Re^{3M}$ and covariance $\Sigma_{t+1} \in \Re^{3M \times 3M}$ of the landmarks, which are given by:

$$\begin{aligned}
\boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t + K_{t+1}\left(\mathbf{z}_{t+1} - \tilde{\mathbf{z}}_{t+1}\right) \\
\Sigma_{t+1} &= (I - K_{t+1}H_{t+1})\Sigma_t
\end{aligned}$$

where

$$\tilde{\mathbf{z}}_{t+1,i} := K_s \pi\left({}_O T_I T_{t+1}^{-1} \underline{\boldsymbol{\mu}}_{t,j}\right) \in \mathbb{R}^4 \qquad \text{for } i = 1, \ldots, N$$

$$H_{t+1,i,j} = \begin{cases} K_s \frac{d\pi}{d\mathbf{q}}\left({}_O T_I T_{t+1}^{-1} \underline{\boldsymbol{\mu}}_{t,j}\right) {}_O T_I T_{t+1}^{-1} P^\top & \text{if } \Delta_t(j) = i, \\ \mathbf{0}, & \text{otherwise} \end{cases}$$

$$K_{t+1} = \Sigma_t H_{t+1}^\top \left(H_{t+1}\Sigma_t H_{t+1}^\top + I \otimes V\right)^{-1}$$

where Ks is the stereo calibration, oTi is the transformation from IMU frame to Optical frame, and pi is the projection function.

The implemented algorithm, mapping_updates(K, T, fs, trajectory, t_v) intakes stereo camera intrinsics K, transformation T from Optical to IMU, features observations fs, and the trajectory of the vehicle, and it returns the updated coordinates of the landmarks **m** in world frame. The transformation of landmarks from optical frame to world frame is basically the same compared to the HW 2 Q2, except transformation iTo already takes oRr into account so we can directly use it instead. The transformation was implemented in otw(fs, Ks, b, uto, T) which intakes features observations fs, stereo intrinsics K, baseline b, pose uto, and transformation iTo, and it returns the coordinates of landmark in world frame.

Visual-Inertial SLAM

Given a sequence of control inputs $u_{0:T}$ and a sequence of features observations $z_{0:t}$, the pose of the vehicle:

$$T_t | \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \Sigma_{t|t})$$

can be estimated by a combination of the prediction steps:

$$\boldsymbol{\mu}_{t+1|t} = \boldsymbol{\mu}_{t|t} \exp\left(\tau_t \ddot{\mathbf{u}}_t\right)$$

$$\Sigma_{t+1|t} = \mathbb{E}[\delta\boldsymbol{\mu}_{t+1|t}\delta\boldsymbol{\mu}_{t+1|t}^\top] = \exp\left(-\tau\overset{\curlywedge}{\mathbf{u}}_t\right)\Sigma_{t|t}\exp\left(-\tau\overset{\curlywedge}{\mathbf{u}}_t\right)^\top + W$$

and the updates steps:

$$\boldsymbol{\mu}_{t+1|t+1} = \boldsymbol{\mu}_{t+1|t}\exp\left((K_{t+1}(\mathbf{z}_{t+1} - \tilde{\mathbf{z}}_{t+1}))^\wedge\right)$$
$$\Sigma_{t+1|t+1} = (I - K_{t+1}H_{t+1})\Sigma_{t+1|t}$$

where

$$K_{t+1} = \Sigma_{t+1|t}H_{t+1}^\top\left(H_{t+1}\Sigma_{t+1|t}H_{t+1}^\top + I \otimes V\right)^{-1}$$

$$H_{t+1,i} = -K_s\frac{d\pi}{d\mathbf{q}}\left({}_O T_I \boldsymbol{\mu}_{t+1|t}^{-1}\underline{\mathbf{m}}_j\right){}_O T_I\left(\boldsymbol{\mu}_{t+1|t}^{-1}\underline{\mathbf{m}}_j\right)^\odot$$

$$\tilde{\mathbf{z}}_{t+1,i} := K_s\pi\left({}_O T_I\boldsymbol{\mu}_{t+1|t}^{-1}\underline{\mathbf{m}}_j\right)$$

and the coordinates of landmarks:

$$\mathbf{m} := \begin{bmatrix}\mathbf{m}_1^\top & \cdots & \mathbf{m}_M^\top\end{bmatrix}^\top \in \mathbb{R}^{3M}$$

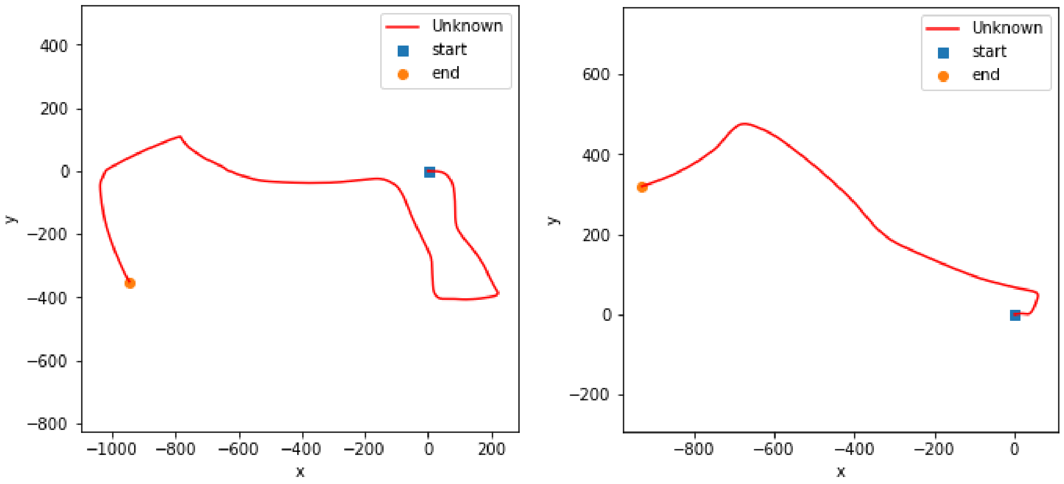
can be estimated by using the same formulas shown in the *Landmark Mapping via EKF Updates*. The implemented algorithm, VIS(v, w, tau, K, iTo, fs, b), intakes linear velocity v, angular velocity w, time discretization tau, stereo camera intrinsics K, transformation iTo from Optical to IMU, features observations fs, and camera baseline b, and it returns the estimated trajectory of the vehicle, and the landmarks locations in the world frame.

Once the algorithm has been implemented, I deployed it on two autonomous vehicle dataset which contains 13289 features observed through 3026 discrete time intervals, the results will be discussed in the results section.

Results

IMU Localization via EKF Prediction

The trajectory map generated by the EKF prediction are:

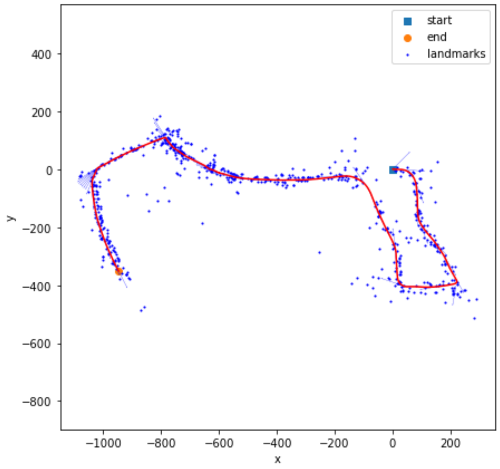


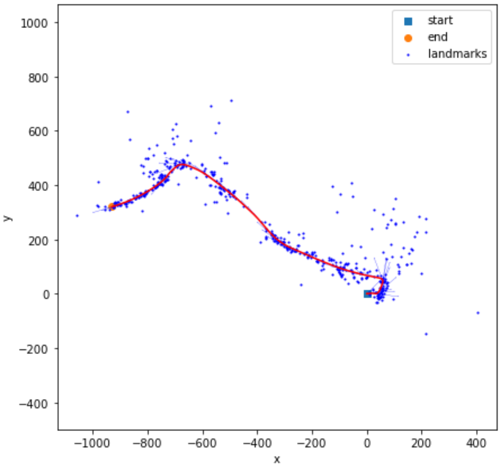
dataset 10, dataset 03

They seems to match the trajectory of the car in the provided video.

Landmark Mapping via EKF Updates

Using 1329 features (1 / 10 total features) for dataset 10 and 511 features (1 / 10 total features) for dataset 3, the landmark mapping results are:

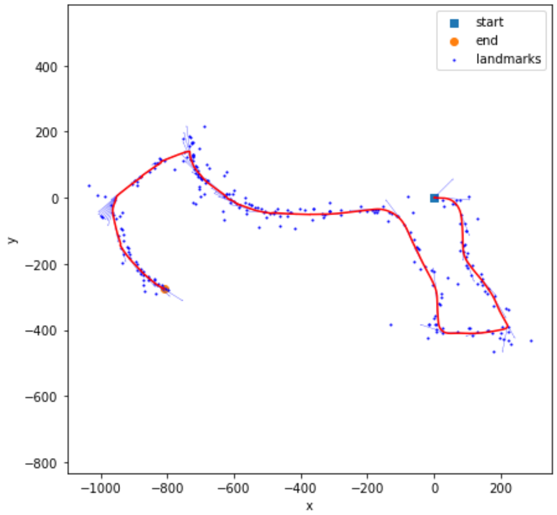


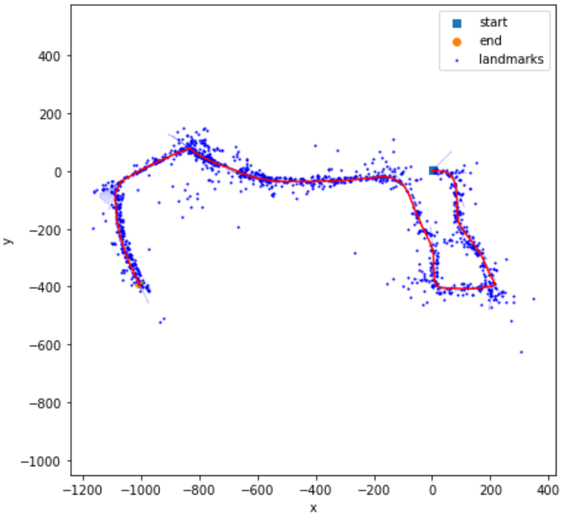


The landmarks positions in these maps seems to make sense based on the provided video.

Visual-Inertial SLAM

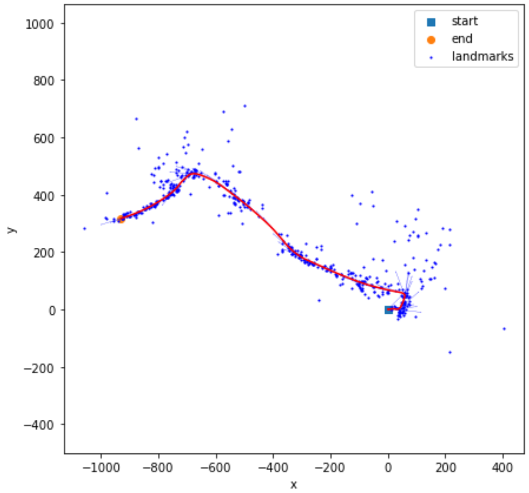
Using 133 features (1 / 100 total features) and 1329 features (1 / 10 total features) for dataset 10, the Visual-Inertial SLAM results are:





We can observe the number of features included affected the quality of the trajectory estimated, and it seems the more features are included, the better the quality is, while the longer it takes to run the algorithm. And for the 1329 features case I reduced the observation noise, which made the estimated trajectory more similar to the prediction only trajectory.

I also mapped the dataset 03 using 511 features, the result is:



Overall, the quality of these SLAM results are better than the prediction only trajectories, which is expected. To improve the algorithm, one can adjust the observation and motion noise (I used the noise suggested by the professor on piazza), I didn't tune the noise because it was time consuming to run the algorithm. One can also increase the number of features, but it requires more computational resources to do so. Based on my experience, it's always important to find a balance between speed and accuracy. And overall, the Visual Inertial SLAM algorithm seems to perform properly.