Visual Inertial SLAM

Amitash Nanda

Electrical and Computer Engineering
University of California San Diego
La Jolla, USA
ananda@ucsd.edu

I. Introduction

Simultaneous Localization and Mapping (SLAM) has become an important research area for a couple of years proving a key technology in solving robotics, autonomous vehicles, and virtual-reality problems. SLAM draws the concept of estimation from sensor data, model comprising the sensors, and the 3D motion of the sensor around its surrounding. Visual SLAM and visual-inertial SLAM has become novel research area with a variety of applications. The advent of smart devices, embedded camera units, and Inertial measurement units (IMU) provide better research capability for visual-inertial SLAM. This topic formulated in 1980's and since then has been an active research subject. SLAM's main objective is to obtain an estimate of an autonomous system path while reconstructing a map of the surrounding environment. Mapping depends on the robot's pose and position while localization requires the environmental map information. Both of them is complex and challenging because of cost and accuracy of the sensors and unknown environment. Therefore, it's important to use a technique where mapping and localization occurs simultaneously.

Navigation and localization are quite important for many indoor applications like autonomous robots, unmanned aerial vehicles or Augmented Reality. In these scenarios GPS doesn't work properly while Wi-Fi or mobile signal are not accurate enough. For such problems visual odometry and inertial navigation systems are usually employed. VO is used in combination with other sensors to obtain a robust performance. IMU measures the acceleration and rotational velocity, and the motion can be obtained by integrating the measurement data. Extended Kalman Filter or EKF is commonly used for data fusion. It's application on SLAM gives an estimate of robot pose and the landmarks location in the environment. It's a linear quadratic estimation that uses a series of measurement observed over time, that contains noise, inaccuracies and provides estimates of unknown variables.

In this project we have implemented visual-inertial simultaneous localization and mapping (SLAM) using an extended Kalman filer (EKF) in python. EKF is used over other popular sparse SLAM algorithm like Particle Filter and Factor Graphs SLAM. This is because EKF uses prediction and update steps to track an autonomous system over time and estimate landmark position. Our project involves three main steps as:

A. IMU Localization via EKF Prediction

B. Landmark Mapping via EKF Update

C. Visual Inertial SLAM

The rest of the paper is organized in the following sections: Section II states the problem formulation, Section III discusses about the technical approaches, Section IV displays the obtained results.

II. Problem Formulation

A. Dataset:

This project has provided us with synchronized measurements from an inertial measurement unit (IMU) of a car. The linear velocity $v_t \in \mathbb{R}^3$ and angular velocity $\omega_t \in \mathbb{R}^3$ measured in the body frame of the IMU. All data is synchronized to time stamp τ_t in UNIX standard seconds-since-the-epoch January 1, 1970. The pixel coordinates $z_t \in \mathbb{R}^{4*M}$ of detected visual features with correspondence between the left and right camera frames are also provided in the data. The landmark m_i that are not observable at time t the returned data is denoted by $[-1 \ -1 -1 -1]^T$. The intrinsic camera calibration contains stereo baseline b which is the distance between camera and intrinsic calibration matrix K.

$$K = \begin{bmatrix} fs_u & 0 & c_u \\ 0 & fs_v & c_v \\ 0 & 0 & 1 \end{bmatrix}$$

B. SLAM Mapping and Localization:

With a given series of controls u_t, sensor observations z_t over discrete time steps t, the SLAM is required to compute an estimate of the robot's position x_t and the positions of landmark m_t in the world frame. So, the objective is to compute a probabilistic model.:

$$P(m_{t+1}, x_{t+1} | \, z_{1:t+1}, u_{1:t})$$

To update the location posteriors sequentially, Bayes' rule requires to be implemented. So, with the transition function $P(x_t|x_{t-1})$ and given map:

$$P(x_t|z_{1:t},u_{1:t},m_t) = \sum_{m_{t-1}} P(z_t|x_t,m_t,u_{1:t}) \sum_{x_{t-1}} P(x_t|x_t,x_{t-1})\, P(x_{t-1}|m_t,z_{1:t-1}\,,u_{1:t})/\eta$$

Also, the map update can be performed through:

$$P(m_t|x_t,z_{1:t},u_{1:t}) = \sum_{x_t}\sum_{m_t} P(m_t|x_t,m_{t-1},z_t,u_{1:t}) P(m_{t-1},x_t|z_{1:t-1},m_{t-1},u_{1:t})$$

Fig 1. Visual features matched across the left-right camera frames(left) and across time (right).

III. TECHNICAL APPROACH

The objective of this project is to use an extended Kalman Filter or EKF to predict and update the car's pose and trajectory across time. The prediction step is based on the SE (3) kinematics with measurements from IMU. While the update step is based on the observation model of stereo camera. Further the feature observation is used to perform localization and mapping.

A. *Extended Kalman Filter or EKF:*

EKF is the non-linear version of the Kalman filter which linearizes about an estimate of the current mean and covariance. The advantage of the EKF over the simpler complementary filter algorithms is that by fusing all available measurements it is better able to reject measurements with significant errors. This makes vehicle less susceptible to faults that affect a single sensor. The EKF equations are stated below:

$x_t|\, z_{0:t}, u_{0:t-1} \sim N(u_{t|t}, \Sigma_{t|t}$ ------ Prior

$x_{t+1} = f(x_t, u_t, w_t),\;\; w_t \sim N(0, W)$ ----- Motion Model

$z_t = h(x_t, v_t), v_t \sim N(0, V)$------- Observation Model

Where x_t defines state at time t. While $z_{0:t-1}$ states the observation from 0 to t and $u_{0:t-1}$ states the observation from 0 to $t-1$. μ is mean and Σ is standard deviation.

Two important steps that we have used in the project for EKF is predication and update. In the given problem statement, having the synchronized measurements of IMU and stereo camera, we used the EKF model by visual-inertial SLAM. The Jacobians for the predicted step, the predicted mean and variance are stated below:

$$F_t \coloneqq \frac{df}{dx}\left(\mu_{t|t}, u_t, 0\right),\;\; Q_t \coloneqq \frac{df}{dw}\left(\mu_{t|t}, u_t, 0\right)$$

$$\mu_{t+1|t} = f(\mu_{t|t}, \mu_t, 0)$$

$$\Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^T + Q_t W Q_t^T$$

In the above equation W is the co-variance of the noise.

Also, the Jacobians for the Update step along with updated mean and co-variances are states below:

$$H_{t+1} \coloneqq \frac{dh}{dx}\left(\mu_{t+1|t}, 0\right),\;\; R_{t+1} \coloneqq \frac{dh}{dv}\left(\mu_{t+1|t}, u_t, 0\right)$$

$$\mu_{t+1|t+1} = \mu_{t+1|t} + K_{t+1|t}(z_{t+1} - h(\mu_{t+1|t}, 0))$$

$$\Sigma_{t+1|t+1} = \Sigma_{t+1|t}\, H_{t+1}^T (H_{t+1}\Sigma_{t+1|t} H_{t+1}^T + R_{t+1} V R_{t+1}^T)^{-1}$$

B. *IMU Localization via EKF prediction:*

In this project we implemented the EKF prediction step based on the SE (3) kinematics. Also used the linear and angular velocity measurements to estimate the pose $T_t \in SE(3)$ of the IMU over time t. To determine the baseline trajectory for SLAM a prediction-only trajectory is obtained initially. Keeping track of the pose $\mu_{t|t} \in$ SE (3) at each time stamp t, the prediction can be calculated as stated below. Rewriting the motion model using the nominal kinematics of $\delta\mu_{t+1|t}$ with time discretization τ is written as:

$$\mu_{t+1|t} = \mu_{t|t} \exp{(\tau \hat{u}_t)}$$

$$\delta\mu_{t+1|t} = \exp\!\left(-\tau \overset{\curlywedge}{u}_t\right) \delta\mu_{t|t} + w_t$$

We derived the following EKF prediction step with the motion model defined above, where $\omega_t \sim \mathrm{N}(0, \mathrm{W})$. Here v_t *and* ω_t are the linear and angular velocities obtained from the IMU. The difference between the present and previous timestamps is given by τ. W is the variance of Gaussian noise ω_t and mean 0.

$$u_t = \begin{bmatrix} v_t \\ \omega_t \end{bmatrix} \in \mathbb{R}^6$$

$$\hat{u}_t \coloneqq \begin{bmatrix} \hat{\omega}_t & v_t \\ 0^T & 0 \end{bmatrix} \in \mathbb{R}^{4\,X\,4}$$

u_t is the concatenation of the linear and angular velocity of the IMU. This is further passed through a hat-map to get the nominal split of the perturbation kinematics. The perturbation scales current position with $\exp(\tau\hat{u}_t)$ in discrete time. This formulation is implemented in the program using the expm from scipy.linalg function.

C. Landmark Mapping via EKF Update:

Feature Matching:

The left and right stereo images are provided to us to find the corresponding features between them. Independent directions resembling a corner is the pixel such that the smallest eigenvalue of the picture is larger than some threshold. While the corner feature detected on the left image and optical flow tracks the feature on right image.

Once the predicted IMU trajectory is obtained, the next step involves the landmark mapping to estimate the positions. In this case, an update only EKF is used, and it can estimate the unknown landmark position m as a state $u_{t|t}$. From the intrinsic calibration matrix K, we can formulate the complete stereo camera calibration matrix M. It is provided by:

$$M := \begin{bmatrix} fs_u & 0 & c_u & 0 \\ 0 & fs_v & c_v & 0 \\ fs_u & 0 & c_u & -fs_u b \\ 0 & fs_v & c_u & 0 \end{bmatrix}$$

We first checked if there are any newly observed features at a specific timestamp, then we implemented EKF. The unknown landmark positions $m \in R^3$ as a state. Provided a new observation let's say z_{t+1}, the predicted observation based on μ_t and δ_{t+1} is given by:

$$\tilde{z}_{t+1,i} = M\pi\left(oT_I T_{t+1}^{-1}\mu_{t,j}\right) \in R^4 \quad for\ i = 1, \ldots, N_{t+1}$$

Considering the perturbation for the position landmark, the projection matrix is shown by:

$$P = [I\ 0] \in R^{3\,X\,4}$$

The projection function and its derivative is given by as follows:

$$\pi(q) = \frac{1}{q3} \in R^4$$

$$\frac{d\pi}{dq}(q) = \frac{1}{q3} = \begin{bmatrix} 1 & 0 & \frac{-q1}{q3} & 0 \\ 0 & 1 & \frac{-q2}{q3} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{-q4}{q3} & 1 \end{bmatrix} \in R^{4X4}$$

After the calculation of the projection function and its derivative we, solved the Jacobian $H_{t+1,i,j}$ of $\tilde{z}_{t+1,i}$ with respect to the landmark m_j. Then we performed the EKF update step after every visual observation, keeping track of mean and covariances. The SLAM update is written as:

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

$$\mu_{t+1|t+1} = \mu_{t+1|t} \exp\left(\left(K_{t+1}(z_{t+1} - \tilde{z}_{t+1})\right)^{\wedge}\right)$$

$$\Sigma_{t+1|t+1} = (I - K_{t+1}H_{t+1})\Sigma_{t+1|t}$$

There is no need to implement the prediction step as the landmarks are assumed to be static. The z coordinates of all landmarks are also assumed to be zero and focused on xy coordinates.

D. Visual Inertail SLAM:

Combining the IMU based prediction with the EKF update landmark mapping, we successfully implemented an IMU update step. It is based on the stereo camera observation model to obtain a complete visual inertial SLAM. New IMU pose are localized using the EKF prediction and the car trajectory is appended. With the successive iteration the car's trajectory and new landmark is observed.

IV. Results

The EKF SLAM created a more accurate map and trajectory compared to just the IMU localization prediction and landmark mapping via EKF update. The algorithm is performed on two datasets provided to us (03.npz and 10.npz). IMU based localization, the trajectory is determined as follows, the dead reckoning Map for both datasets are shown in Figure 1 and Figure 2 respectively.

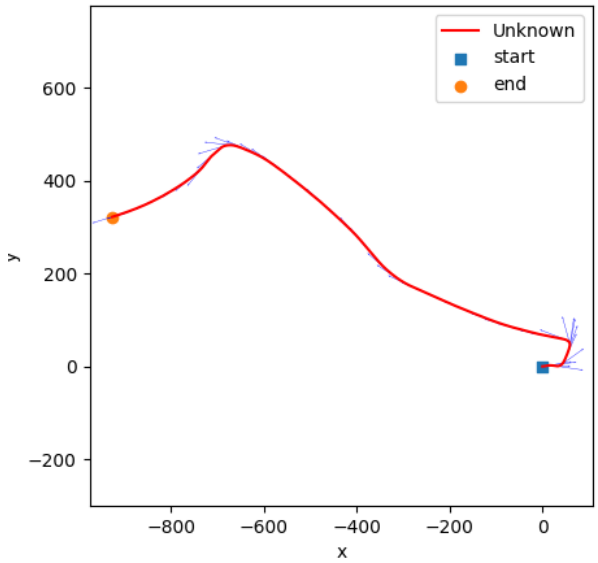

Fig 2. IMU based localization via EKF prediction for 03.npz

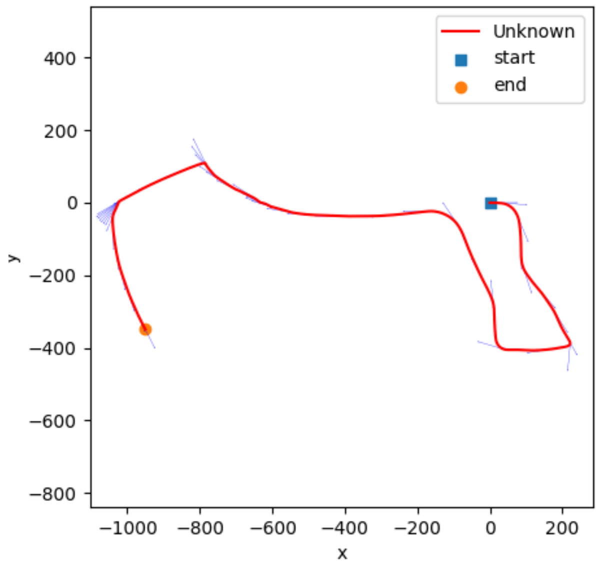

Fig 3. IMU based localization via EKF prediction for 10.npz

The trajectory from the IMU-based localization is assumed to be correct and EKF-update is performed for landmark mapping. The world frame is plotted using all the features. After the successful implementation of the algorithm, the map displays the car trajectory and environment landmarks. The landmark features are tried to down sampled for 10.npz datasets but with full feature the code has ran. While for 03.npz it has been shown till the 1900 iteration. The results of EKF landmark mapping, the, blue points are closely aligned with the trajectory of the car in red colour. With the variation in noise the best map is created as shown below.

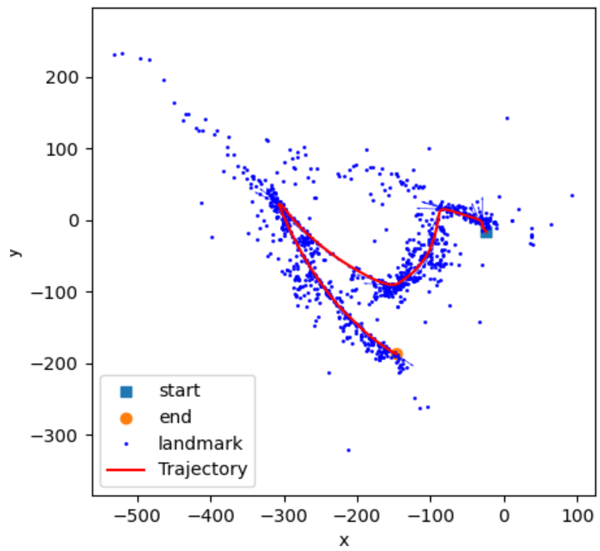

Fig 4. IMU based localization via EKF prediction for 03.npz

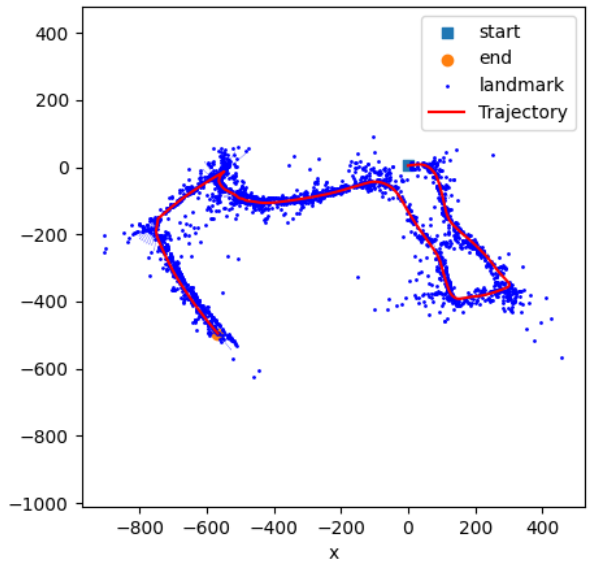

Fig 4. IMU based localization via EKF prediction for 10.npz