Visual-Inertial SLAM

Chung-Pang, Wang
University of California San Diego
Department of Electrical and Computer Engineering

***Abstract*—The goal of this project is to implement visual-inertial simultaneous localization and mapping (SLAM) using an extended Kalman filter (EKF). With synchronized measurements from an inertial measurement unit (IMU) and a stereo camera as well as the intrinsic camera calibration and the extrinsic calibration between the two sensors, specifying the transformation from the left camera frame to the IMU frame.**

***Index Terms*—Extended Kalman Filter, Visual-inertial SLAM**

I. Introduction

In this project, we will first implement an Extended Kalman Filter (EKF) prediction step based on the SE(3) kinematics equations and the linear and angular velocity measurements from the IMU to estimate the pose $T_t \in SE(3)$ of the IMU over time t. Then, we will implement Landmark mapping via EKF update step based on stereo-camera observation model with visual feature observations. Finally, we will combine the IMU prediction step with the landmark update step and implement an EKF update step for the IMU pose $Tt \in SE(3)$, based on the stereo-camera observation model, to obtain a complete visual-inertial SLAM algorithm.

II. Problem Formulation

A. Datasets Overview

The goal of this project is to implement visual-inertial simultaneous localization and mapping (SLAM) using an extended Kalman filter (EKF). We have synchronized measurements from an inertial measurement unit (IMU) and a stereo camera as well as the intrinsic camera calibration and the extrinsic calibration between the two sensors, specifying the transformation from the left camera frame to the IMU frame. For IMU measurements, we have linear velocity $\mathbf{v}_t \in \mathbb{R}^3$ and angular velocity $\omega_t \in \mathbb{R}^3$ of the body with coordinates expressed in the body frame of the IMU. For Visual feature measurements, we have pixel coordinates $\mathbf{z}_t \in \mathbb{R}^{4\times M}$ of detected visual features from M point landmarks with pre-computed correspondences between the left and the right camera frames. For our camera model we have Intrinsic calibration matrix with stereo baseline b in meters, camera calibration matrix:

$$\mathbf{K_s} = \begin{bmatrix} fs_u & 0 & c_u & 0 \\ 0 & fs_v & c_b & 0 \\ fs_u & 0 & c_u & -fs_u b \\ 0 & fs_v & c_v & 0 \end{bmatrix} \tag{1}$$

We also have Extrinsic matrix for Extrinsic calibration: transformation ${}_I T_C \in SE(3)$ from the left camera to the IMU frame.Furthermore, the IMU sensor is placed upside down on the vehicle so the IMU frame is oriented as x = forward, y = right, z = down. When estimating the IMU trajectory, you can leave the IMU frame orientation as is or rotate around the x-axis to obtain a regular coordinate frame.

B. Extended Kalman Filter

In the first part, we are only implementing EKF prediction step to obtain the whole trajectory and also not predicting the covariance of the robot poses since we only need the poses to construct IMU odometry. We have not formulate a way to perform EKF update step to implement landmarks mapping. First, we need to construct EKF in a more general way. Motion model for the continuous-time IMU pose $T(t)$ with noise $\mathbf{w}(t)$

$$\dot{T} = T(\hat{\zeta} + \hat{\mathbf{w}}) \quad \zeta(t) := \begin{bmatrix} \mathbf{v}(t) \\ \boldsymbol{\omega}(t) \end{bmatrix} \in \mathbb{R}^6 \tag{2}$$

To consider a Gaussian distribution over T, express it as a nominal pose $\boldsymbol{\mu} \in SE(3)$ with small perturbation $\hat{\delta\boldsymbol{\mu}} \in \mathfrak{se}(3)$

$$T = \boldsymbol{\mu} \exp(\hat{\delta\boldsymbol{\mu}}) \approx \boldsymbol{\mu}(I + \hat{\delta\boldsymbol{\mu}})$$

Substitute the nominal and perturbed pose in the kinematic equations:

$$\begin{aligned} &\dot{\boldsymbol{\mu}}(I + \hat{\dot{\delta}}\boldsymbol{\mu}) + \boldsymbol{\mu}(\hat{\dot{\delta}}\boldsymbol{\mu}) = \boldsymbol{\mu}(I + \hat{\delta\boldsymbol{\mu}})(\hat{\mathbf{u}} + \hat{\mathbf{w}}), \\ &\dot{\mu} + \dot{\mu}\hat{\delta}\boldsymbol{\mu} + \boldsymbol{\mu}(\hat{\dot{\delta}}\boldsymbol{\mu}) = \boldsymbol{\mu}\hat{\mathbf{u}} + \boldsymbol{\mu}\hat{\mathbf{w}} + \boldsymbol{\mu}\hat{\boldsymbol{\mu}}\hat{\mathbf{u}} + \boldsymbol{\mu}\hat{\boldsymbol{\mu}}\hat{\mathbf{w}}, \\ &\dot{\boldsymbol{\mu}} = \boldsymbol{\mu}\hat{\mathbf{u}}, \quad \boldsymbol{\mu}\hat{\mathbf{u}}\hat{\boldsymbol{\mu}} + \boldsymbol{\mu}(\hat{\dot{\delta}}\boldsymbol{\mu}) = \boldsymbol{\mu}\hat{\mathbf{w}} + \boldsymbol{\mu}\hat{\boldsymbol{\mu}}\hat{\mathbf{u}}, \\ &\dot{\boldsymbol{\mu}} = \boldsymbol{\mu}\hat{\mathbf{u}}, \quad \hat{\dot{\delta}}\boldsymbol{\mu} = \hat{\dot{\delta}}\hat{\mathbf{u}} - \hat{\mathbf{u}}\hat{\boldsymbol{\mu}} + \hat{\mathbf{w}} = (-\hat{\mathbf{u}}\delta\boldsymbol{\mu})^{\wedge} + \hat{\mathbf{w}} \end{aligned} \tag{3}$$

Using $T = \boldsymbol{\mu}\exp(\hat{\mu}) \approx \boldsymbol{\mu}(I + \hat{\delta\boldsymbol{\mu}})$, the pose kinematics $\dot{T} = T(\hat{\mathbf{u}} + \hat{\mathbf{w}})$ can be split into nominal and perturbation kinematics:

$$\begin{aligned} \text{nominal} : \dot{\mu} &= \mu\hat{\mathbf{u}} \\ \text{perturbation} : \quad \dot{\delta\boldsymbol{\mu}} &= -\hat{\mathbf{u}}\delta\boldsymbol{\mu} + \mathbf{w} \\ \hat{\mathbf{u}} : &= \begin{bmatrix} \hat{\omega} & \hat{\mathbf{v}} \\ 0 & \hat{\omega} \end{bmatrix} \in \mathbb{R}^{6\times 6} \end{aligned} \tag{4}$$

In discrete-time with discretization τ_t, the above becomes:

$$\begin{aligned} \text{nominal} : \boldsymbol{\mu}_{t+1} &= \boldsymbol{\mu}_t \exp(\tau_t \hat{\mathbf{u}}_t) \\ \text{perturbation} : \delta\boldsymbol{\mu}_{t+1} &= \exp(-\tau_t \hat{\mathbf{u}}_t)\,\delta\boldsymbol{\mu}_t + \mathbf{w}_t \end{aligned}$$

Motion model can be written as nominal kinematics of $\boldsymbol{\mu}_{t|t}$ and perturbation kinematics of $\delta\boldsymbol{\mu}_{t|t}$ with time discretization τ_t :

$$\begin{aligned} \boldsymbol{\mu}_{t+1|t} &= \boldsymbol{\mu}_{t|t} \exp\left(\tau_t \hat{\mathbf{u}}_t\right) \\ \delta\boldsymbol{\mu}_{t+1|t} &= \exp\left(-\tau_t \hat{\mathbf{u}}_t\right) \delta\boldsymbol{\mu}_{t|t} + \mathbf{w}_t \end{aligned} \tag{5}$$

EKF prediction step with $\mathbf{w}_t \sim \mathcal{N}(0, W)$:

$$\begin{aligned} \boldsymbol{\mu}_{t+1|t} &= \boldsymbol{\mu}_{t|t} \exp\left(\tau_t \hat{\mathbf{u}}_t\right) \\ \Sigma_{t+1|t} &= \exp\left(-\tau \hat{\mathbf{u}}_t\right) \Sigma_{t|t} \exp\left(-\tau \hat{\mathbf{u}}_t\right)^\top + W \end{aligned} \tag{6}$$

where

$$\mathbf{u}_t = \begin{bmatrix} \mathbf{v}_t \\ \boldsymbol{\omega}_t \end{bmatrix} \in \mathbb{R}^6 \quad \hat{\mathbf{u}}_t = \begin{bmatrix} \hat{\boldsymbol{\omega}}_t & \mathbf{v}_t \\ \mathbf{0}^\top & 0 \end{bmatrix} \in \mathbb{R}^{4\times4}$$

$$\hat{\mathbf{u}}_t = \begin{bmatrix} \hat{\boldsymbol{\omega}}_t & \hat{\mathbf{v}}_t \\ 0 & \hat{\boldsymbol{\omega}}_t \end{bmatrix} \in \mathbb{R}^{6\times6}$$

We have EKF prediction step for both mean and covariance. Now we need to further construct the formulations for EKF update steps. Observation model with measurement noise $\mathbf{v}_{t,i} \sim \mathcal{N}(0, V)$:

$$\mathbf{z}_{t,i} = h\left(T_t, \mathbf{m}_j\right) + \mathbf{v}_{t,i} := K_s \pi\left(oT_I T_t^{-1} \underline{\mathbf{m}}_j\right) + \mathbf{v}_{t,i} \tag{7}$$

Homogeneous coordinates for m: $\underline{\mathbf{m}}_j := \begin{bmatrix} \mathbf{m}_j \\ 1 \end{bmatrix}$

Projection function and its derivative:

$$\pi(\mathbf{q}) := \frac{1}{q_3}\mathbf{q} \in \mathbb{R}^4 \quad \frac{d\pi}{d\mathbf{q}}(\mathbf{q}) = \frac{1}{q_3}\begin{bmatrix} 1 & 0 & -\frac{q_1}{q_3} & 0 \\ 0 & 1 & -\frac{q_2}{q_3} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{q_4}{q_3} & 1 \end{bmatrix} \in \mathbb{R}^{4\times4}$$

All observations, stacked as a $4N_t$ vector, at time t with notation abuse:

$$\mathbf{z}_t = K_s \pi\left(oT_I T_t^{-1} \underline{\mathbf{m}}\right) + \mathbf{v}_t \quad \mathbf{v}_t \sim \mathcal{N}(\mathbf{0}, I \otimes V) \tag{8}$$

where

$$I \otimes V := \begin{bmatrix} V & & \\ & \ddots & \\ & & V \end{bmatrix}$$

Prior of the mapping : $\mathbf{m} \mid \mathbf{z}_{0:t} \sim \mathcal{N}\left(\boldsymbol{\mu}_t, \Sigma_t\right)$ with $\boldsymbol{\mu}_t \in \mathbb{R}^{3M}$ and $\Sigma_t \in \mathbb{R}^{3M\times3M}$. We have stereo calibration matrix K_s, extrinsics ${}_o\mathrm{T}_{imu} \in SE(3)$, IMU pose $T_{t+1} \in SE(3)$, new observation $\mathbf{z}_{t+1} \in \mathbb{R}^{4N_{t+1}}$. We can predict observation based on $\boldsymbol{\mu}_t$ and known correspondences Δ_{t+1} :

$$\tilde{\mathbf{z}}_{t+1,i} = K_s \pi\left(oT_I T_{t+1}^{-1} \underline{\mu}_{t,j}\right) \in \mathbb{R}^4 \quad \text{for } i = 1, \ldots, N_{t+1} \tag{9}$$

Jacobian of $\tilde{\mathbf{z}}_{t+1,i}$ with respect to $\mathbf{m}_j$ evaluated at $\boldsymbol{\mu}_{t,j}$:

$$H_{t+1,i,j} = \begin{cases} K_s \frac{d\pi}{d\mathbf{q}}\left(oT_I T_{t+1}^{-1} \underline{\boldsymbol{\mu}}_{t,j}\right) oT_I T_{t+1}^{-1} P^\top, & \text{if } \Delta_t(j) = i, \\ \mathbf{0}, & \text{otherwise} \end{cases} \tag{10}$$

EKF update:

$$\begin{aligned} K_{t+1} &= \Sigma_t H_{t+1}^\top \left(H_{t+1} \Sigma_t H_{t+1}^\top + I \otimes V\right)^{-1} \\ \mu_{t+1} &= \mu_t + K_{t+1}\left(\mathbf{z}_{t+1} - \tilde{\mathbf{z}}_{t+1}\right) \\ \Sigma_{t+1} &= \left(I - K_{t+1} H_{t+1}\right) \Sigma_t \end{aligned} \tag{11}$$

Now, we have all of the formulation of EKF prediction and updates for both Localization and Landmarks Mapping.

C. IMU localization via EKF prediction

With the linear velocity of robot position and angular velocity of robot orientation provided from the IMU sensor data:

$$v_t = \begin{bmatrix} v_x(t) \\ v_y(t) \\ v_z(t) \end{bmatrix} \omega_t = \begin{bmatrix} \omega_x(t) \\ \omega_y(t) \\ \omega_z(t) \end{bmatrix} \tag{12}$$

we can utilize discrete-time pose kinematics to obtain robot pose in world frame at time t ${}_wT_t \in SE3$. We first have generalized velocity $\zeta_t = [v(t), \omega(t)]^T \in \mathbb{R}^6$, then discrete-time pose kinematics:

$$T_{t+1} = T_t exp(\tau_t \hat{\zeta}_t) \tag{13}$$

where $\hat{\zeta}_t$ is the hat mat of generalized velocity, twist matrix:

$$\hat{\zeta}_t = \begin{pmatrix} v_t \\ \omega_t \end{pmatrix} = \begin{bmatrix} \hat{\omega}_t & v_t \\ 0 & 0 \end{bmatrix} \tag{14}$$

With equation (11), we can obtain robot poses for every time step t and further construct the whole trajectory of the robot.

D. Landmark mapping via EKF update

We obtained the robot odometry from previous localization. Now we are assume that the poses of the robot x are known. The environment is represented by M static landmarks, and each of them is characterized by its location in the space denoted as $\mathbf{m}_i, i = 1, \ldots, K$. These landmarks are considered as points in the 3D space and can be specified by three numerical values where $\mathbf{m}_i \in \mathbb{R}^3$ and $\mathbf{m} \in \mathbb{R}^{3\times M}$. In each EFK update step, we will first generate an initial guess of the landmarks location with known poses, then update μ_{t+1} with equation(9), we will explain the details of the process in the following sections. The robot can sense the landmarks at each time step t, where the observation is denoted as $\mathbf{z}_t$. Since the robot can sense more than one landmarks at a single time step, $\mathbf{z}_t$ is a general notation for composed observation from multiple landmarks.

The goal of the mapping problem is then to estimate the locations of landmarks based on the pose of robot x and the observation $\mathbf{z}_t$. In each EFK update step, we will first generate an initial guess of the landmarks location with known poses, then update μ_{t+1} with equation(9), we will explain the details of the process in the following sections.

E. Visual-inertial SLAM

To obtain a better map and odometry, we need to estimate the position of landmarks and the pose of the robot simultaneously, the idea is to merge the predict and update steps of Extended Kalman Filter based visual mapping and visual-inertial odometry. Therefore, instead of implementing EKF prediction and update separately, we need to fist predict the pose and the positions of the landmarks (initial guess) and their corresponding covariance then update the poses and the landmarks and their covariance. However, since the EKF update step formulation we derived previously are based on landmarks $m_j \in \mathrm{R}^3$, the update step formulation for poses $T \in SE(3)$ needed to be reformulated. We will discuss more in the next section.

III. Methods

A. 3D Stereo-Camera Model

Since the feature data are in pixels coordinate observed by the left camera and right camera, we need to utilize stereo camera model to covert it into world coordinate to obtain the position of the landmarks in world frame. Consider stereo camera model:

$$\begin{bmatrix} u_L \\ v_L \\ d \end{bmatrix} = \begin{bmatrix} fs_u & 0 & c_u & 0 \\ 0 & fs_v & c_v & 0 \\ 0 & 0 & 0 & fs_u b \end{bmatrix} \frac{1}{z} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{15}$$

where $d = u_L - u_R = \frac{1}{z} fs_u b$ Since stereo intrinsic matrix is not invertible, we cannot directly take the inverse of K_s to obtain x and y, instead, we derive the solution of x and y:

$$\begin{aligned} x &= (u_L - c_x)\, z/f_x \\ y &= (v_L - c_y)\, z/f_y \end{aligned} \tag{16}$$

where $z = \frac{fsub}{d}$

B. EKF update for SE(3) Poses

Let the elements of $H_{t+1} \in \mathbb{R}^{4N_{t+1}\times 6}$ corresponding to different observations i be $H_{t+1,i} \in \mathbb{R}^{4\times 6}$ The first-order Taylor series approximation of observation i at time $t+1$ using an IMU pose perturbation $\delta\boldsymbol{\mu}$ is:

$$\begin{aligned} \mathbf{z}_{t+1,i} &= K_s \pi \left(oT_I \left(\boldsymbol{\mu}_{t+1|t} \exp(\hat{\delta\boldsymbol{\mu}}) \right)^{-1} \underline{\mathbf{m}}_j \right) + \mathbf{v}_{t+1,i} \\ &\approx K_s \pi \left(oT_I (I - \hat{\boldsymbol{\mu}}) \boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j \right) + \mathbf{v}_{t+1,i} \\ &= K_s \pi \left(oT_I \boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j - oT_I \left(\boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j \right)^{\odot} \delta\boldsymbol{\mu} \right) + \mathbf{v}_{t+1,i} \\ &\approx \underbrace{K_s \pi \left(oT_I \boldsymbol{\mu}_{t+1|t}^{-1} \underline{m}_j \right)}_{\tilde{\mathbf{z}}_{t+1,i}} \underbrace{- K_s \frac{d\pi}{d\mathbf{q}} \left(oT_I \boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j \right) oT_I \left(\boldsymbol{\mu}_{t+1|t}^{-1} \mathbf{m}_j \right)^{\odot}}_{H_{t+1,i}} \delta\boldsymbol{\mu} + \mathbf{v}_{t+1,i} \end{aligned} \tag{17}$$

Where homogeneous coordinates $\underline{\mathbf{s}} \in \mathbb{R}^4$ and $\hat{\boldsymbol{\xi}} \in \mathfrak{se}(3)$:

$$\hat{\boldsymbol{\xi}} = \underline{\mathbf{s}}^{\odot} \boldsymbol{\xi} \qquad \begin{bmatrix} \mathbf{s} \\ 1 \end{bmatrix}^{\odot} := \begin{bmatrix} l & -\hat{\mathbf{s}} \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4\times 6}$$

Predicted observation based on $\boldsymbol{\mu}_{t+1|t}$ and known correspondences Δ_t :

$$\tilde{\mathbf{z}}_{t+1,i} := K_s \pi \left(oT_l \boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j \right) \quad \text{for } i = 1, \ldots, N_{t+1} \tag{18}$$

Jacobian of $\tilde{\mathbf{z}}_{t+1,i}$ with respect to T_{t+1} evaluated at $\boldsymbol{\mu}_{t+1|t}$

$$H_{t+1,i} = -K_s \frac{d\pi}{d\mathbf{q}} \left(oT_I \boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j \right) oT_I \left(\boldsymbol{\mu}_{t+1|t}^{-1} \underline{\mathbf{m}}_j \right)^{\odot} \in \mathbb{R}^{4\times 6} \tag{19}$$

EKF update step:

$$\begin{aligned} &K_{t+1} = \Sigma_{t+1|t} H_{t+1}^{\top} \left(H_{t+1} \Sigma_{t+1|t} H_{t+1}^{\top} + I \otimes V \right)^{-1} \\ &\mu_{t+1|t+1} = \mu_{t+1|t} \exp\left(\left(K_{t+1} \left(\mathbf{z}_{t+1} - \tilde{\mathbf{z}}_{t+1} \right) \right)^{\wedge} \right) \\ &\Sigma_{t+1|t+1} = \left(I - K_{t+1} H_{t+1} \right) \Sigma_{t+1|t} \end{aligned} \tag{20}$$

Where

$$H_{t+1} = \begin{bmatrix} H_{t+1,1} \\ \vdots \\ H_{t+1,N_{t+1}} \end{bmatrix}$$

C. IMU localization via EKF prediction

As mentioned previously, we only implement EKF prediction for IMU localization to obtain the trajectory. For implementation, I used equation (4) to compute all of the predicted poses in the world frame and also the predicted covariance overtime.

D. Landmark mapping via EKF update

With equation (13) and (14), we can obtain landmarks positions in camera frame. Then, we transform it into predicted observation in world frame with (7). With the predicted observation (initial guess), we can update landmark positions in each update step with (9).

E. Visual-inertial SLAM

In Visual-inertial SLAM, we are merging the predict and update steps of Extended Kalman Filter based visual mapping and visual-inertial odometry. For implementation, the setting of the noises and covariance are very important since the SLAM algorithm is very sensitive to the noise setting. Based on the fact that we know our motion will be more accurate than the landmarks prediction and updates, I set motion noise and pose prior to be relatively small around 0.1 and set landmark covariance and measurements noise to be 15 pixels.

IV. Results

A. IMU localization via EKF prediction

Our algorithm is tested with 2 data set, and all of them are collected in real driving scenarios. After verifying the path visually through the provided video, the results of the IMU odometry seem to be consistent with motion of the car in the video as shown in Fig.1.

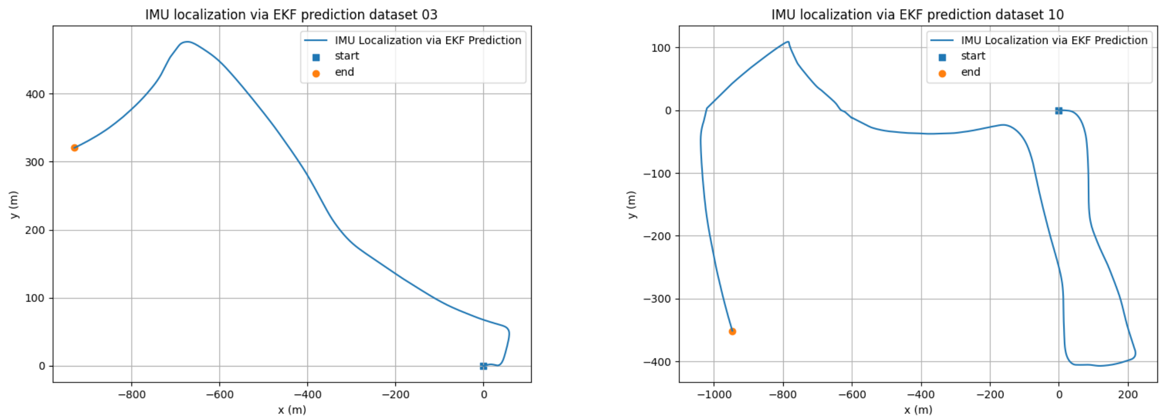

Fig. 1. IMU localization via EKF prediction

B. Landmark mapping via EKF update

Landmark mapping plays a crucial role in ensuring the accuracy of the overall SLAM framework. The initial guess for landmark mapping is shown in Fig. 2. From this visualization, we can observe that the landmarks are distributed consistently along both sides of the vehicle's trajectory, which aligns well with the expected pattern for a car driving in an urban environment. This initial guess serves as a foundation for refining the landmark positions through subsequent updates.

Using the Extended Kalman Filter (EKF) update step, the landmark estimates are progressively refined as new observations are integrated. The EKF update incorporates both the measurement noise and the covariance associated with the landmarks, enabling the system to balance between the observed data and prior estimates. As shown in Fig. 3, the updated landmarks demonstrate a more uniform and even distribution compared to the initial guess, effectively capturing the vehicle's odometry and the surrounding environment.

A key improvement observed after the EKF update is the reduction in the uncertainty associated with landmark positions. By leveraging the sensor data and the vehicle's estimated pose, the EKF update minimizes discrepancies in the landmark estimates, ensuring better consistency and alignment with the actual environment. This is particularly beneficial in dynamic or noisy environments, where raw measurements alone may not be reliable.

Furthermore, the EKF update step highlights the adaptability of the SLAM framework. For instance, landmarks that were initially misaligned or sparsely placed become more evenly distributed after several update cycles. This process demonstrates the robustness of the EKF in handling variations in sensor noise and environmental conditions.

C. Visual-inertial SLAM

As shown in Fig. 4, the trajectory produced by the SLAM algorithm appears significantly different from the localization-only trajectory. This difference is expected, as SLAM incorporates additional information from observed landmarks to correct errors in localization. The scale of the map is represented in meters, and the results highlight a substantial deviation of up to 100 meters at the trajectory's endpoint. Such deviations, while large in real-world scenarios, reflect the challenges of accurately fusing visual and inertial data in the absence of ground truth.

One of the key challenges in implementing visual-inertial SLAM lies in tuning the covariance and noise parameters. The algorithm's sensitivity to these parameters makes it difficult to achieve an optimal balance between over-trusting sensor data (leading to noisy maps) and over-smoothing (resulting in overly simplistic trajectories). Without access to ground truth data, parameter tuning often involves iterative adjustments and visual verification to ensure the trajectory aligns with observed motion in video recordings.

Despite these challenges, the SLAM-generated trajectory demonstrates improvements in capturing fine-grained details of the environment compared to localization alone. For instance, the integration of visual feature observations helps correct drift in odometry, resulting in a trajectory that better reflects the robot's actual path. However, achieving a precise trajectory remains an open challenge, particularly in environments with high sensor noise or dynamic obstacles.

This sensitivity underscores the importance of robust noise models and accurate sensor calibration. Future improvements could involve adaptive noise tuning techniques or the incorporation of ground truth data during testing to refine parameter settings. Additionally, integrating advanced filtering techniques, such as smoothing-based SLAM, could further enhance trajectory accuracy and mapping reliability.

REFERENCES

[1] UCSD ECE276A slides on motion models and observation models https://natanaso.github.io/ece276a/ref/ECE276A_4_MotionAndObservationModels.pdf
[2] UCSD ECE276A slides on locoalization Modelshttps://natanaso.github.io/ece276a/ref/ECE276A_6_LocalizationOdometry.pdf
[3] UCSD ECE276A slides on visual slam Modelshttps://natanaso.github.io/ece276a/ref/ECE276A_12_VI_SLAM.pdf

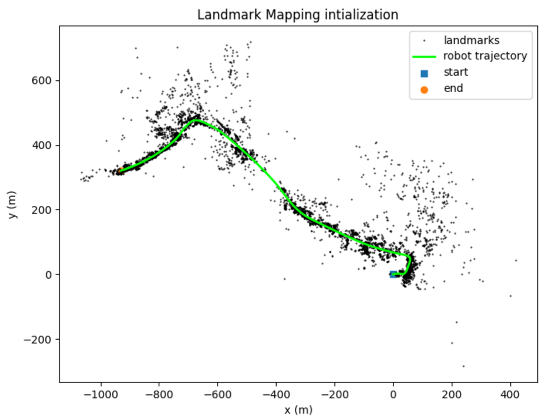

Fig. 2. Landmark Mapping Initial Guess Dataset 03

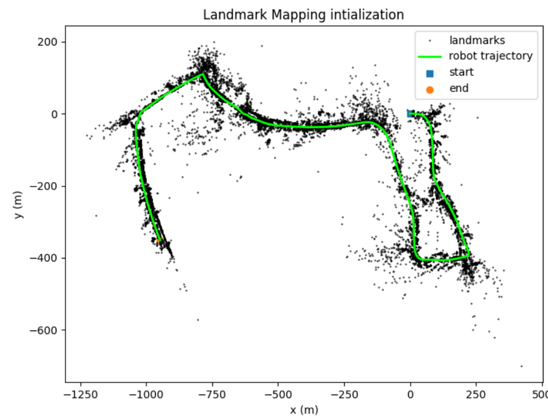

Fig. 3. Landmark Mapping Initial Guess Dataset 10

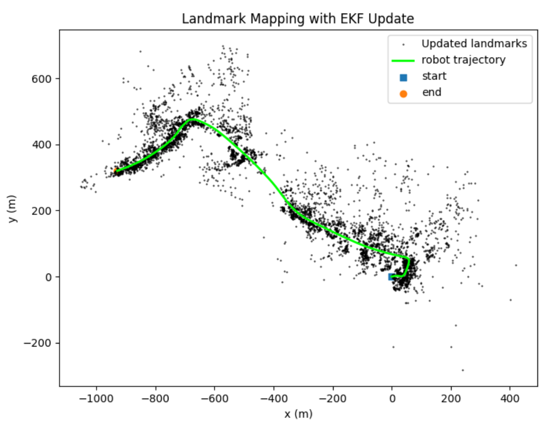

Fig. 4. Landmark Mapping with EKF Update Dataset 03

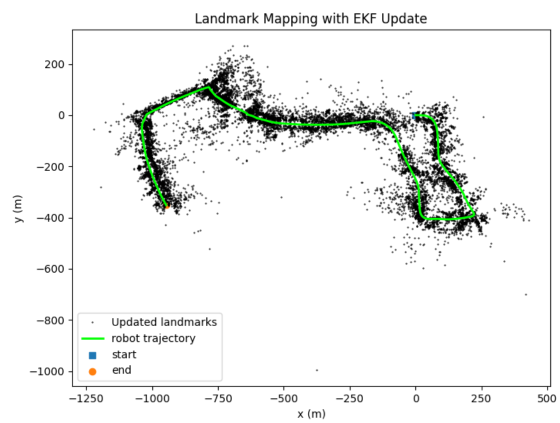

Fig. 5. Landmark Mapping with EKF Update Dataset 10

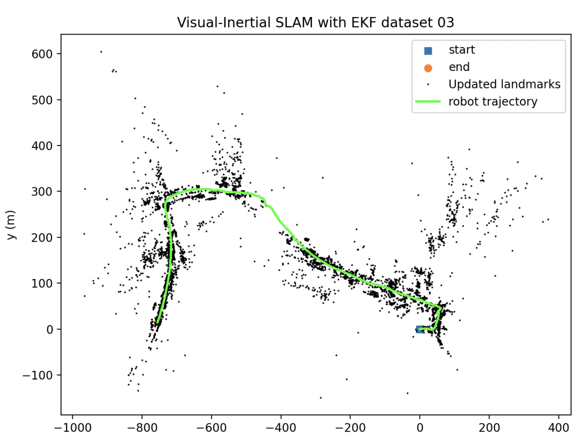

Fig. 6. Visual-Inertial SLAM with EKF Dataset 03

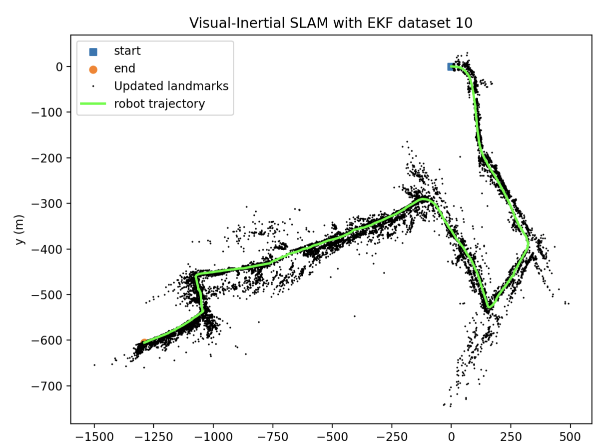

Fig. 7. Visual-Inertial SLAM with EKF Dataset 10