Visual-Inertial SLAM with Extended Kalman Filter

Zejia Wu
ECE Department
UC San Diego
zew024@ucsd.edu

***Abstract*—This project explores the problem of visual-inertial simultaneous localization and mapping (SLAM), a fundamental challenge in robotics essential for enabling autonomous systems to navigate and operate in unstructured or dynamic environments. We present an extended Kalman filter (EKF)-based solution that fuses data from an inertial measurement unit (IMU) and a stereo vision system to estimate both the trajectory of the sensor platform and the structure of the surrounding environment. The proposed framework is divided into two core stages: the first involves predicting the pose of the IMU (hence the robot) over time using $SE(3)$ motion models within the EKF prediction step; the second performs map estimation by incorporating visual observations of environmental features into the EKF update, allowing for accurate recovery of landmark positions. This tightly coupled visual-inertial strategy enables reliable state estimation by leveraging the complementary strengths of inertial and visual sensors. The experimental results confirm the capability of the system to produce consistent localization and mapping outcomes, highlighting the potential of visual-inertial SLAM in advancing robust perception for autonomous agents. Furthermore, this implementation serves as a stepping stone toward more advanced techniques in sensor fusion and robotic state estimation.**

***Index Terms*—Visual-Intertial SLAM, Extended Kalman Filter, SE(3) kinematics**

I. Introduction

A fundamental challenge in advancing robotic autonomy lies in enabling robots to perceive, understand, and interact with complex and dynamic environments. Central to addressing this challenge is Simultaneous Localization and Mapping (SLAM), a foundational technique that allows autonomous agents to construct a map of their surroundings while concurrently estimating their own position within that map. Among the various SLAM methodologies, Visual-Inertial SLAM (VI-SLAM) has gained prominence due to its ability to fuse high-frequency inertial measurements with spatially rich visual data. This sensor fusion enhances robustness and accuracy, mitigating the individual shortcomings of using visual or inertial sensors alone.

The primary goal of this project is to develop a VI-SLAM system based on the Extended Kalman Filter (EKF) framework, utilizing measurements from an inertial measurement unit (IMU) and a stereo vision system. The proposed system predicts the IMU's trajectory using continuous-time kinematic models and refines this estimate through updates informed by visual observations of static environmental features. This two-stage process enables both precise pose estimation and accurate mapping of environmental landmarks—two capabilities that are vital for navigation and decision-making in autonomous systems.

Beyond its theoretical importance, VI-SLAM has wide-ranging real-world applications, including autonomous driving, aerial robotics, and planetary exploration, where reliable perception and mapping are essential. This project seeks to examine the practical efficacy of EKF-based VI-SLAM and contribute to the broader effort of enhancing robotic perception and autonomy. The report details our full approach—from problem formulation and algorithm design to implementation and evaluation—offering insights into both the technical challenges encountered and the strategies employed to address them.

II. Problem Formulation

In this work, we address the problem of localizing a mobile platform equipped with a stereo RGB camera system and an inertial measurement unit (IMU), while simultaneously constructing a map of the surrounding environment. The objective is to estimate the state trajectory $\mathbf{x}_{0:t}$, which encodes the vehicle's pose over time up to the current time step t, given a sequence of control inputs $\mathbf{u}_{0:t-1}$ and sensor observations $\mathbf{z}_{0:t}$. Concurrently, we aim to build a spatial map $\mathbf{m} \in \mathbb{R}^{3\times M}$, representing the three-dimensional positions of M static landmarks observed in the environment. This dual estimation problem—tracking the vehicle's motion while incrementally refining the map—forms the core of the SLAM framework under consideration. We can overwrite the $\mathbf{x} \in \mathbb{R}^{3\times M+6}$ as the concatenation of the robot pose $\mathbf{x}_{0:t} \in \mathbb{R}^6$ and the landmark position $m \in \mathbb{R}^{3\times M}$, since they are all variables that we want to estimate. Essentially, we are solving the MAP optimization problem:

$$\min_{\mathbf{x}_{0:T}} -\log p(\mathbf{x}_0) - \sum_{t=0}^{T} \log p_h(\mathbf{z}_t|\mathbf{x}_t) - \sum_{t=0}^{T-1} \log p_f(\mathbf{x}_{t+1}|\mathbf{x}_t, u_t).$$

To simultaneously estimate both the robot's pose and the positions of environmental landmarks, the system dynamics are modeled as a Markov process. In this framework, the state-to-estimate (of the robot and the landmark) at time t, denoted by $\mathbf{x}_t$, is assumed to depend solely on its previous state $\mathbf{x}_{t-1}$, the control input $\mathbf{u}_t$ for the robot (since we assume that the landmarks are static), and the current observation $\mathbf{z}_t$. This formulation enables the use of a Kalman Filter (Bayes Filter) to recursively estimate the posterior distribution over the state space. The Bayes Filter maintains two primary probabilistic estimates at each timestep:

Prediction Step. This step computes the prior (or predictive) distribution over the future state, given as

$$p_{t+1|t}(\mathbf{x}_{t+1}) = p(\mathbf{x}_{t+1}|\mathbf{u}_t, \mathbf{x}_t),$$

which projects the system's state forward based on the current control inputs, typically using a motion model.

In this project, we assume that the landmarks are static, hence we have

$$p_{t+1|t}(\mathbf{x}_{t+1}[0:3M]) = \mathbf{x}_{t+1}[0:3M] + w_m,$$

which are only affected by noise. The robot pose can be described by $T_t \in SE(3)$, which will be predicted according to its kinematics in the prediction step.

Update Step. Once a set of new observations becomes available, the filter updates its belief about the state through

$$p_{t+1|t+1}(\mathbf{x}_{t+1}) = p(\mathbf{x}_{t+1}|\mathbf{z}_{t+1}),$$

integrating the latest sensor measurements to refine the prediction and reduce uncertainty. In this project, we use pixels in images generated by two cameras as our observation. Assuming that we observe N_t matched features at time t in both images, the observation $z_t \in \mathbb{R}^{4N_t}$ will be the pixel coordinates in both images.

This recursive process exploits the Markov property and conditional independence assumptions, enabling efficient real-time estimation. The prediction step leverages the system's kinematic or dynamic model to infer motion, while the update step corrects the estimate using incoming sensory data, thus maintaining an accurate and consistent belief of the robot's state.

III. Technical Approach

In this section, we will give detailed algorithms and implementations to deal with the problems formulated above.

A. Extended Kalman Filter

The Extended Kalman Filter (EKF) is a widely used extension of the classical Kalman Filter, designed to handle nonlinear system dynamics and observation models. It approximates the true posterior distribution by linearizing the nonlinear functions around the current estimate of the state's mean and covariance, thereby maintaining a Gaussian representation of uncertainty. The EKF operates through two primary stages:

- Prediction Step: The filter propagates the current state estimate and its associated covariance forward in time using the system's nonlinear motion model, predicting the future state before new measurements are received.
- Update Step: Upon receiving a new observation, the EKF updates the predicted state by incorporating the measurement information. This step corrects the prediction, reducing uncertainty and improving the accuracy of the state estimate.

Assuming that at current time step t, the prior distribution of the state $\mathbf{x}_t$ given all past measurements $\mathbf{z}_{0:t}$ and control inputs $\mathbf{u}_{0:t-1}$ follows a Gaussian distribution

$$\mathbf{x}_{t|t} \sim \mathcal{N}(\mu_{t|t}, \Sigma_{t|t}),$$

where $\mathbf{x}_t \in \mathbb{R}^{3M+6}$ contains states for both the robot and landmarks, and $\Sigma \in \mathbb{R}^{3M+6\times 3M+6}$.

The motion model estimates the subsequent state $\mathbf{x}_{t+1}$ based on the current state $\mathbf{x}_t$, the control input $\mathbf{u}_t$, and process noise $\mathbf{w}_t$, which is assumed to be zero-mean Gaussian with covariance $\mathbf{W}$:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, w_t), \qquad w_t \sim \mathcal{N}(0, W).$$

In this project, the landmarks and the robot pose follow different motion models. We will show this in detail in the next two subsections.

To linearize the motion model, EKF uses the first-order Taylor series as an approximation to the nonlinear model, hence Jacobian is needed:

$$F_t = \frac{\partial f}{\partial \mathbf{x}}(\mu_{t|t}, \mathbf{u}_t, 0) \in \mathbb{R}^{(3M+6)\times(3M+6)},$$
$$Q_t = \frac{\partial f}{\partial \mathbf{w}}(\mu_{t|t}, \mathbf{u}_t, 0) \in \mathbb{R}^{(3M+6)\times(3M+6)}.$$

The prediction step can then be performed:

$$\mu_{t+1|t} = f(\mu_{t|t}, \mathbf{u}_t, 0),$$
$$\Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^T + Q_t W_t Q_t^T.$$

Similarly, to perform the update step, we need to consider the observation model and its Jacobians:

$$\mathbf{z}_{t+1} = h(\mathbf{x}_{t+1}, v_t), \quad v_t \sim \mathcal{N}(0, V),$$
$$H_{t+1} = \frac{\partial h}{\partial \mathbf{x}}(\mu_{t+1|t}, 0) \in \mathbb{R}^{4M\times(3M+6)},$$
$$R_{t+1} = \frac{\partial h}{\partial \mathbf{v}}(\mu_{t+1|t}, 0) \in \mathbb{R}^{4M\times 4M}.$$

Then the update equations correct the predicted state mean $\mu_{t+1|t}$ and the covariance $\Sigma_{t+1|t}$:

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

where $K_{t+1} \in \mathbb{R}^{(3M+6)\times 4M}$ is the Kalman gain:

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

which encodes the reliability of our sensor measurements.

Now, the existing two problems are

- What is the motion model and how to calculate its Jacobians?
- What is the observation model and how to calculate its Jacobians?

We will show this in the following to subsections.

B. SE(3) Kinematics and Derivatives

In this project, we assume that the landmarks are static, hence we have

$$f(\mathbf{x}_{t+1}[0:3M]) = \mathbf{x}_{t+1}[0:3M] + w_m,$$

which are only affected by noise. The derivatives with respect to the landmark positions and landmark noise are both identity matrices, which is the part in Q_t and R_t that we do not need to care much.

The only challenging part is the robot kinematics. Since the robot pose belongs to $SE(3)$, given the control input

$$\mathbf{u}_t = \begin{bmatrix} v_t \\ \omega_t \end{bmatrix} \in \mathbb{R}^6,$$

the prediction step can be formulated as

$$\mu_{t+1|t} = \begin{bmatrix} \mu^m_{t+1|t} \\ \mu^r_{t|t} \exp(\tau_t \hat{\mathbf{u}}_t) \end{bmatrix}, \quad \Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^T + W,$$

where

$$F_t = \begin{bmatrix} I^{3M\times 3M} & 0 \\ 0 & \exp(\tau_t \mathbf{u}_t^{\curlywedge}) \end{bmatrix},$$
$$\hat{\mathbf{u}}_t = \begin{bmatrix} \hat{\omega}_t & \hat{v}_t \\ \mathbf{0}^{\mathrm{T}} & 0 \end{bmatrix} \in \mathbb{R}^{4\times 4},$$
$$\mathbf{u}_t^{\curlywedge} = \begin{bmatrix} \hat{\omega}_t & \hat{v}_t \\ \mathbf{0} & \hat{\omega}_t \end{bmatrix} \in \mathbb{R}^{6\times 6}.$$

Now we have $H^m_{t+1|t} \in \mathbb{R}^{4N_t \times 3M}$ and $H^p_{t+1|t} \in \mathbb{R}^{4N_t \times 6}$, which are the Jacobians of the observation model with respect to landmark positions and the robot pose. The observation matrix Jacobian is then

$$H_{t+1|t} = \begin{bmatrix} H^m_{t+1|t} & H^p_{t+1|t} \end{bmatrix} \in \mathbb{R}^{4N_t \times (3M+6)}.$$

Assuming that the noise Jacobian $R_{t+1|t}$ is identity, we now have everything to perform an update step in EKF.

C. Visual Mapping

We have two cameras with known positions with respect to the IMU reference frame, and they have different intrinsic matrices. Both cameras have the same rotation with respect to the IMU frame. Therefore, given the position of the right camera in the left camera optical frame,

$$P^L_R = {}_oT_r \ {}^{left}_{cam}T_{imu} \ {}^{right}_{cam}T^{-1}_{imu} \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^{\mathrm{T}},$$

we have the observation model

$$\begin{bmatrix} u_L \\ v_L \\ u_R \\ v_R \end{bmatrix} = K\Pi({}_oT_r \ {}^{left}_{cam}T_{imu} \ T^{-1} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}),$$

where

$$K = \begin{bmatrix} f^L_u & 0 & c^L_u & 0 \\ 0 & f^L_v & c^L_v & 0 \\ f^R_u & 0 & c^R_u & -f^R_u P^L_R[0] - c^R_u P^L_R[2] \\ 0 & f^R_v & c^R_v & f^R_u P^L_R[1] - c^R_u P^L_R[2] \end{bmatrix}.$$

We need to calculate the Jacobian of the observation model with respect to the state-to-estimate. For those with respect to the landmark positions, we have

$$H^m_{i,j} = \begin{cases} K_s \frac{d\pi}{dq}(T_{trans}\mu^j_{t+1|t}) T_{trans} P^T & \text{if } \Delta_t(j) = i \\ 0 & \text{otherwise} \end{cases},$$

where

$$T_{trans} = {}_oT_r \ {}^{left}_{cam}T_{imu} \ T^{-1},$$

and

$$\frac{d\pi}{dq}(q) = \begin{bmatrix} 1 & 0 & \frac{q_1}{q_3} & 0 \\ 0 & 1 & \frac{q_2}{q_3} & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & -\frac{q_4}{q_3} & 1 \end{bmatrix}.$$

For those with respect to the robot pose, we have

$$H^p_j = -K_s \frac{d\pi}{dq}(T_{trans}\underline{\mu}^j_{t+1|t}) \ {}_oT_r \ ({}^{left}_{cam}T_{imu} \ T^{-1}\underline{\mu}^j_{t+1|t})^{\odot},$$

where the odot operator $\odot$ is defined as

$$\begin{bmatrix} s \\ 1 \end{bmatrix}^{\odot} = \begin{bmatrix} I & -\hat{s} \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4\times 6}$$

IV. Results

In this part, we present the outcomes of the various components involved in the EKF-SLAM pipeline, including the estimated IMU trajectory, landmark mapping results, and the performance of the EKF-SLAM system. A comprehensive analysis and comparison of these results are conducted to gain deeper insights into the behavior and effectiveness of the visual-inertial SLAM framework. Additionally, dynamic visualizations of the map construction process are provided through the link for dataset00, link for dataset01, and link for dataset02.

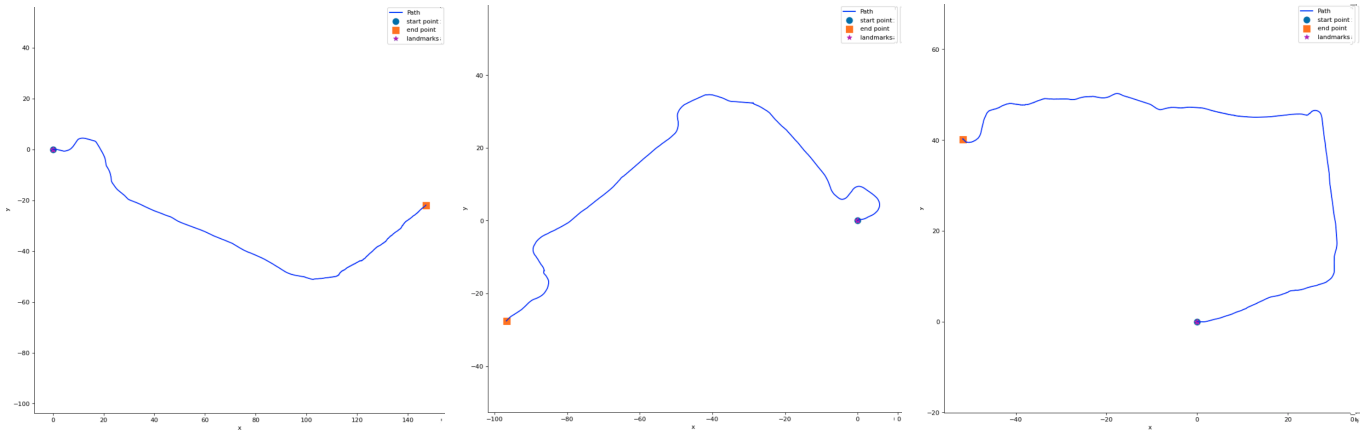
A. IMU Trajectory Prediction and Landmark Visual Mapping

In this phase, $SE(3)$ kinematics were employed to predict the robot's pose, under the assumption that the estimated IMU trajectory accurately reflects the true motion. The results of this trajectory prediction are illustrated in Fig 1. This step served to evaluate the quality of the IMU data, and the resulting trajectory was notably smooth—indicating a consistent and stable estimation of motion over time in the absence of visual corrections. Such smoothness highlights the reliability of the IMU in capturing continuous dynamic motion.

Following the trajectory prediction, the system proceeded to map environmental landmarks using the previously estimated IMU path, as shown in Fig 2. The positions of visual landmarks were evolving as more features were observed. This gradual refinement of landmark estimates illustrates the system's adaptive capability, progressively enhancing its spatial understanding as new information becomes available. The results demonstrate the system's effectiveness in constructing a map of landmarks relative to the robot's path.

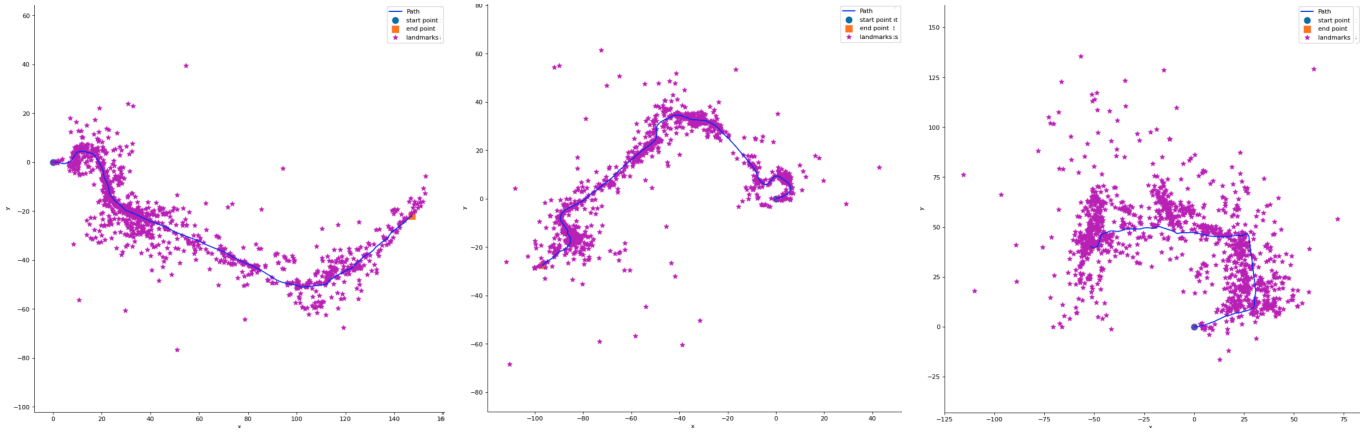
B. SLAM Results

In the second stage of our analysis, we evaluate the performance of the full visual-inertial SLAM system, which integrates IMU-based state prediction with visual landmark updates and pose corrections. We began by tuning the noise parameters through a series of experiments. When a higher observation noise is assumed, the system shows greater confidence in the IMU prediction, resulting in a trajectory that closely resembles the one derived from IMU data alone. In contrast, reducing the observation noise relatively increases the influence of visual measurements, leading to more substantial



(a) Dataset 00 (b) Dataset 01 (c) Dataset 02

Fig. 1: The IMU prediction results for three datasets, where the blue path is the 2D projection of the robot trajectory.



(a) Dataset 00 (b) Dataset 01 (c) Dataset 02

Fig. 2: The mapping results for three datasets, where the blue path is the 2D projection of the robot trajectory, and the pink dots are the estimated locations of detected landmarks.

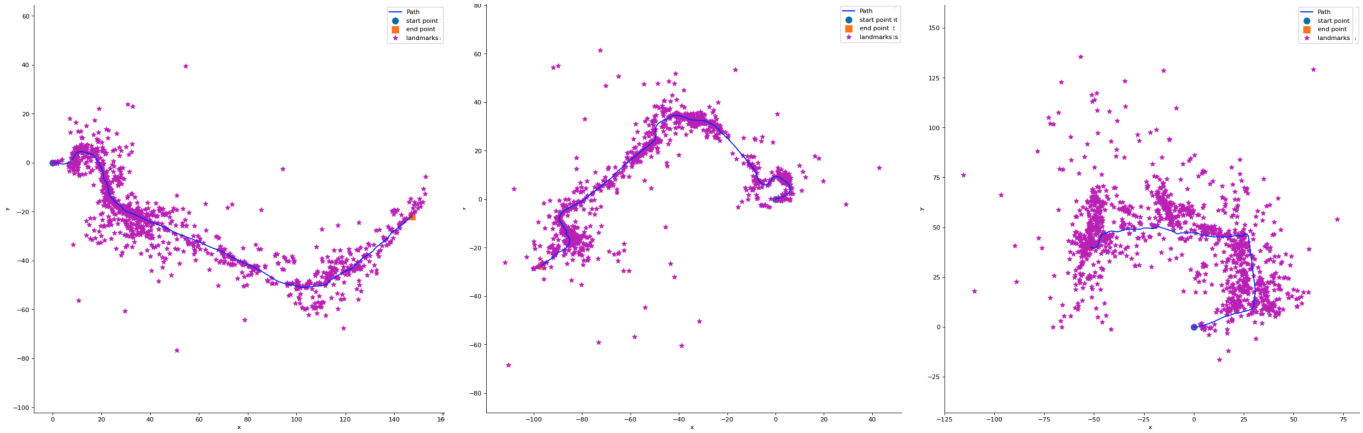
corrections in pose estimates. We found that the motion noise scale $\sqrt{0.0001}$ and measurement noise scale $\sqrt{2}$ can produce the smoothest and most visually consistent (to the video) trajectory for the first two datasets. For dataset02, we used measurement noise scale $\sqrt{4}$.

To filter out the unreliable visual features, we implemented a filtering mechanism to discard landmark observations that are too far away or deviated significantly from predicted values.

Finally, we show the SLAM results for three datasets in Fig 3, in which the robot path is more smooth the previous two images since it is corrected based on the observations. However, the improvement in robot trajectory is not easy to observe without zooming in since the IMU results are already quite accurate. The results highlight the importance of carefully modeling the observation process. Accurate parameter tuning and robust outlier rejection are essential for improving the overall stability and reliability of the visual-inertial SLAM system.

V. Conclusion

In conclusion, this project provided a practical implementation of visual-inertial SLAM using an Extended Kalman Filter (EKF) framework. It demonstrated the fundamental integration of visual and inertial sensor data for simultaneous localization and mapping, while also revealing the inherent challenges of this approach. The results were promising, but they also indicated areas for improvement, particularly in computational efficiency and the accuracy of landmark estimation. Overall, this work establishes a solid foundation for future research and development aimed at enhancing the performance and reliability of SLAM systems in real-world applications.



(a) Dataset 00 (b) Dataset 01 (c) Dataset 02

Fig. 3: The SLAM results for three datasets, where the blue path is the 2D projection of the robot trajectory, and the pink dots are the estimated locations of detected landmarks.