Project 2: LiDAR-Based SLAM Report

March 22, 2025

1 Introduction

SLAM (Simultaneous Location and Mapping) is a critical problem in robotics, enabling a robot to build a map of an unknown environment while keeping track of its own location within that map without relying on external positioning systems like GPS. This capability is essential for autonomous navigation, environmental exploration, and task execution in unstructured or dynamic settings. This assignment aims to implement a Visual-Inertial SLAM system, integrating motion data from an IMU (Inertial Measurement Unit) with visual observations from a stereo camera, using an EKF (Extended Kalman Filter) to jointly estimate the robot's pose and landmark positions. We will first independently develop IMU localization and landmark mapping , then combine these components with pose correction to form a complete VI-SLAM algorithm. The ultimate goal is to adjust the noise parameters for accurate trajectory and map, enhancing the robot's autonomy and reliability in real-world scenarios.

1.1 IMU Localization via EKF Prediction

In this section, we implement an Extended Kalman Filter (EKF) prediction step to localize the IMU over time. The approach leverages the SE(3) kinematics equations to estimate the pose $T_t \in SE(3)$ using the measured linear and angular velocities from the IMU.

Specifically, the EKF prediction step includes the following components:

- **State Representation:** The pose of the IMU is modeled as an element of the Special Euclidean group $SE(3)$, capturing both its rotational and translational states.
- **Kinematic Model:** The dynamics of the system are described by the SE(3) kinematics equations. These equations govern the evolution of the IMU's state as a function of its linear and angular velocity inputs.
- **Prediction Step:** The filter propagates the current state estimate forward in time by integrating the kinematic model. This step accounts for process noise and other uncertainties inherent in the IMU measurements.

By incorporating these elements, the EKF prediction step provides a robust framework for estimating the IMU's pose over time, forming the basis for further sensor fusion and state estimation processes.

1.2 Feature detection and matching

In this task, our goal is to generate feature tracks for landmark mapping using dataset02, which lacks pre-provided features. For each time step t, we construct a measurement matrix

$$z_t \in \mathbb{R}^{4\times M} = \begin{bmatrix} lx_{,1} & lx_{,2} & \dots & lx_{,M} \\ ly_{,1} & ly_{,2} & \dots & ly_{,M} \\ rx_{,1} & rx_{,2} & \dots & rx_{,M} \\ ry_{,1} & ry_{,2} & \dots & ry_{,M} \end{bmatrix},$$

where $lx_{,j}$ and $ly_{,j}$ represent the x and y pixel coordinates of the j-th feature in the left image, and $rx_{,j}$ and $ry_{,j}$ are the corresponding coordinates in the right image. Features not visible at time t are assigned a value of -1. Feature detection is performed using the Shi-Tomasi method (via `goodFeaturesToTrack`) on the left image, and stereo matching is achieved by tracking these features to the right image using optical flow (`calcOpticalFlowPyrLK`). Temporal tracking is also performed to propagate features between consecutive left images. This framework facilitates robust data association for subsequent landmark mapping and visual-inertial SLAM.

1.3 Landmark Mapping via EKF Update

In this section, we focus on estimating the static landmark positions $m \in \mathbb{R}^{3\times M}$ observed in the images, assuming that the IMU trajectory predicted in the EKF prediction step is correct. We implement an EKF update where the landmark positions form the state, and the visual observations

z_t are used to update the mean and covariance of this state.

Key aspects of the implementation include:

- **State Representation:** The unknown landmark positions m are modeled as the state vector in the EKF update.
- **Visual EKF Update:** A subset of the available visual feature measurements is used to perform the EKF update, ensuring that the computational complexity remains manageable.
- **Static Landmarks:** Since the landmarks are assumed to be static, no prediction step is necessary for them. Only the update step is applied.

This approach provides a robust framework for landmark mapping by integrating visual observations to refine the estimation of landmark positions while keeping the computational requirements under control.

1.4 Visual-Inertial SLAM via EKF Update

In this section, we integrate the IMU prediction step from part (a) with the landmark update step from part (b) to develop a complete visual-inertial SLAM algorithm. The approach uses the stereo-camera observation model to refine the IMU pose $T_t \in SE(3)$ while simultaneously updating the static landmark positions.

Key components of the algorithm include:

- **IMU Prediction:** The current IMU pose is propagated using the SE(3) kinematics equations and the measured linear and angular velocities, providing a preliminary trajectory estimate.
- **Landmark Update:** Visual observations from the stereo-camera are employed to update the landmark positions and their associated uncertainties. These updates are then used to correct the initial IMU trajectory.
- **Visual-Inertial Fusion:** The stereo-camera observation model is used to perform an EKF update on the IMU pose, integrating the visual corrections from the landmark mapping process.
- **Noise Tuning:** A crucial aspect of the method involves tuning the noise parameters to optimize the trajectory estimation. While the IMU trajectory alone is only a rough estimate, the refined landmark data provides the necessary corrections for improved accuracy.

By combining these steps, the algorithm leverages both inertial and visual data to produce a more robust and accurate estimate of the IMU pose and the environment's landmark structure.

2 Technical Approach

2.1 Noise Choose

To optimize the Visual-Inertial SLAM trajectory, process noise $W = 10^{-4} \cdot I_6$ and observation noise $V = 10^{-2} \cdot I_4$ were selected. The low W reflects high trust in IMU data, while a larger V accounts for stereo camera uncertainty. These values were tuned empirically to balance pose accuracy and landmark stability.

2.2 Features Filters

Feature filtering downsamples stereo observations every 20 frames to reduce computational load. Invalid features ($[-1, -1, -1, -1]$) are removed, and a disparity check ($0 < u_L - u_R < 100$) ensures valid depth estimation. A maximum of 1000 landmarks is enforced, with new landmarks initialized only if unseen and within this limit. This approach enhances efficiency and maintains robust landmark tracking for SLAM.

2.3 IMU Localization via EKF Prediction

In IMU localization, the goal is to predict the robot's pose $T_t \in SE(3)$ using IMU measurements (linear velocity v_t and angular velocity w_t). Given the control input $u_t = [v_t, w_t]$ and time step Δt, the motion model updates the pose as:

$$\mu_{t+1|t} = T_t \exp(\Delta t \cdot \hat{u}_t),$$

where $\hat{u}_t$ is the se(3) representation of u_t. The covariance $\Sigma_t \in \mathbb{R}^{6\times 6}$ is propagated using the linearized motion model $F_t = \exp(-\Delta t \cdot \mathrm{adj}(u_t))$, with process noise W added:

$$\Sigma_{t+1|t} = F_t \Sigma_t F_t^\top + W.$$

This part focuses solely on pose estimation, ignoring landmarks.

2.4 Feature Detection and Matching

Feature detection and matching form the foundation of the Visual-Inertial SLAM pipeline by extracting and tracking landmarks from stereo images. Initial features are detected in the left frame at $t = 0$ using the Shi-Tomasi corner detector (`cv2.goodFeaturesToTrack`), with a maximum of 1000 corners, a quality level of 0.01, and a minimum distance of 10 pixels. Stereo matching between left and right frames employs Lucas-Kanade optical flow (`cv2.calcOpticalFlowPyrLK`) with a 15×15 window and two pyramid levels, identifying corresponding points based on disparity. Temporal tracking propagates features across consecutive frames, maintaining continuity using the same optical flow parameters. Invalid features (e.g., all -1) are discarded, and a disparity filter ($0 < u_L - u_R < 100$) ensures reliable depth estimation. A dictionary, `all_features`, assigns unique IDs to tracked features, building a measurement matrix z_t of shape $4 \times M$ per timestep for EKF updates.

2.5 Landmark Mapping via EKF Update

In landmark mapping, the objective is to refine the positions of landmarks $\mu \in \mathbb{R}^{3M}$ (where M is the number of landmarks) using stereo camera observations, assuming the robot's pose $T_t \in SE(3)$ is fixed from Part (a). Given the observation $z_t = [u_L, v_L]$ for each landmark, the predicted observation is:

$$z_{\text{pred},i} = K_l \pi(T_{\text{imu}}^{-1} T_t^{-1} \mu_i),$$

where π projects 3D points to 2D, and T_{imu} is the IMU-to-camera transform. The observation Jacobian $H_{t,i} = K_l \frac{\partial \pi}{\partial q} T_{\text{imu}}^{-1} T_t^{-1}$ relates 2D measurements to 3D states. The innovation is $z_t - z_{\text{pred}}$. The Kalman gain is:

$$K_t = \Sigma_t H_t^\top (H_t \Sigma_t H_t^\top + V)^{-1},$$

updating the state as $\mu_{t+1} = \mu_t + K_t(z_t - z_{\text{pred}})$ and covariance as:

$$\Sigma_{t+1} = (I - K_t H_t)\Sigma_t.$$

This process refines the map without altering the pose.

2.6 Visual-Inertial SLAM via EKF Update

In this section, we integrate the IMU prediction step from Part (a) with the landmark update step from Part (b) to develop a complete Visual-Inertial SLAM algorithm. The joint state $\mu_t = \{T_t, \mu_l\}$, where $T_t \in SE(3)$ is the IMU pose and $\mu_l \in \mathbb{R}^{3M}$ represents landmark positions, is estimated using stereo camera observations and inertial data. The covariance $\Sigma_t \in \mathbb{R}^{(6+3M)\times(6+3M)}$ captures their correlations. The algorithm initializes with $T = \text{len(timestamps)}$, $\mu_t['pose'] = I_{4\times4}$, and an empty $\mu_t['landmarks']$, storing the trajectory as $\text{trajectory} = [\mu_t['pose']]$.

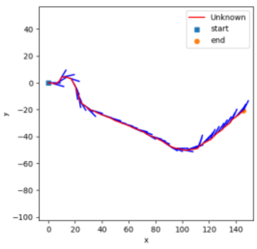
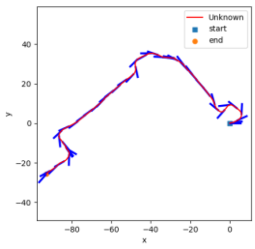
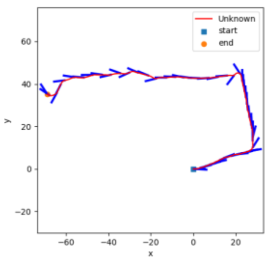
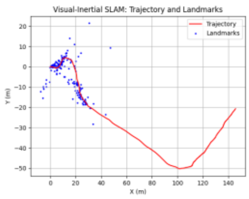
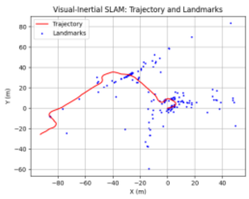
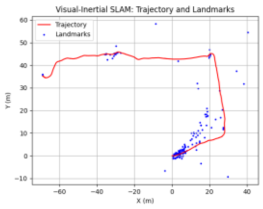
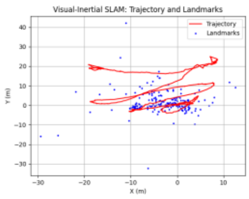
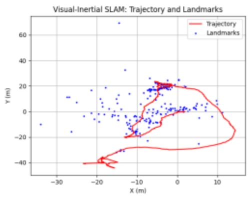
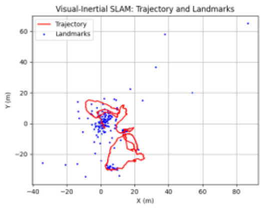
The IMU prediction propagates T_t using $SE(3)$ kinematics, with control input $u_t = [v_t, w_t]$ from linear and angular velocities. The pose updates as $T_{t+1} = T_t \cdot \exp(\Delta t \cdot \hat{u}_t)$ via `EKF_prediction_joint`, providing a preliminary trajectory. Stereo observations $z_t = [u_L, v_L, u_R, v_R]$ refine this estimate. The predicted observation is:

$$z_{\text{pred},i} = \begin{bmatrix} K_l \pi(T_{\text{imu}}^{-1} T_t^{-1} \mu_{l,i}) \\ K_r \pi(T_{\text{r}}^{-1} T_{\text{imu}}^{-1} T_t^{-1} \mu_{l,i}) \end{bmatrix},$$

where $T_{\text{imu}} = \text{extL\_T\_imu}^{-1}$ and T_{r} accounts for the baseline $b = \text{cal\_baseline}(extL\_T\_imu, extR\_T\_imu)$.

Landmark updates leverage these observations to adjust μ_l and its uncertainties in `ekf_update_joint`. The joint Jacobian $H_t = [H_{t,r}, H_{t,l}]$ links z_t to pose and landmarks, enabling visual-inertial fusion. The innovation $z_t - z_{\text{pred}}$ drives the Kalman gain $K_t = \Sigma_t H_t^\top (H_t \Sigma_t H_t^\top + V)^{-1}$, updating $T_{t+1|t+1} = T_t \exp(\text{hat\_map}(\delta[:6]))$ and $\mu_l + \delta[6:]$. Covariance adjusts as $\Sigma_{t+1|t+1} = (I - K_t H_t)\Sigma_t$, enhancing pose accuracy and map consistency through this integrated approach.

Result

	Dataset00	Dataset01	Dataset02
IMU localization via EKF prediction			
Landmark mapping via EKF update			
Visual-inertial SLAM			

Method	Trajectory Accuracy	Landmark Mapping	Dataset Robustness
IMU Localization via EKF Prediction	Low, drifts significantly	N/A	prone to drift
IMU Localization via EKF Prediction	Medium, depends on landmarks	Medium, Dataset01 best	Medium, stable with dense landmarks
Landmark Mapping via EKF Update	High, near perfect	High, detailed maps	Excellent, adapts to sparsity

- **Unexpected Detail**: In Dataset02, with sparse landmarks, Visual-Inertial SLAM maintains high accuracy, showing strong environmental adaptability.
- **Dataset Differences**: Dataset01’ s complex trajectories highlight VI-SLAM’ s strength; Dataset02’ s sparsity tests robustness, where VI-SLAM excels; Dataset00 shows dense, consistent mapping.

Graphical results indicate Visual-Inertial SLAM (bottom row) outperforms in trajectory accuracy and landmark mapping consistency across datasets, effectively fusing visual and inertial data, particularly in complex environments.

Feature detection and matching : The framework for feature detection and matching has been largely implemented. However, due to unresolved bugs in OpenCV, the final results have not yet been obtained.