Visual Inertial SLAM

Manoj Kumar Reddy Manchala
Department of Mechanical and Aerospace Engineering
University of California San Diego
La Jolla, U.S.A.
mmanchala@ucsd.edu

Nikolay Atanasov
Department of Electrical and Computer Engineering
University of California San Diego
La Jolla, U.S.A.
natanasov@eng.ucsd.edu

***Abstract*—This project addresses the challenge of implementing a visual-inertial simultaneous localization and mapping (SLAM) system using an extended Kalman filter (EKF). The system integrates synchronized measurements from an inertial measurement unit (IMU) and a stereo camera to estimate the trajectory and map of the environment. The key contributions include the design of an EKF prediction step based on SE(3) kinematics for IMU pose estimation and an EKF update step that incorporates visual feature observations for landmark mapping. The project successfully demonstrates the potential of visual-inertial SLAM in improving localization and mapping accuracy in robotics applications. This report details the methodology, implementation challenges, and the achieved improvements in SLAM performance.**

***Index Terms*—Visual-Inertial SLAM, Extended Kalman Filter, Landmark Mapping, Sensor Fusion, Stereo Vision**

I. Introduction

Simultaneous Localization and Mapping (SLAM) is a critical problem in the field of robotics and autonomous systems, where a robot or a vehicle must construct a map of its environment while simultaneously determining its position within that map. The integration of visual and inertial sensors, known as Visual-Inertial SLAM (VI-SLAM), has gained prominence due to its ability to provide high accuracy and robustness under various conditions.

The significance of VI-SLAM lies in its application across a wide range of domains, including autonomous vehicles, drone navigation, augmented reality, and robotics. These systems rely heavily on the ability to accurately perceive and interact with their surroundings, making efficient and reliable SLAM important.

This project aims to implement a VI-SLAM system using an Extended Kalman Filter (EKF). The system uses synchronized measurements from an Inertial Measurement Unit (IMU) and a stereo camera to estimate the pose of the robot and the positions of landmarks in the environment. The EKF prediction step utilizes SE(3) kinematics to estimate the IMU's pose, while the update step integrates visual observations to refine the landmark map and the robot's trajectory.

II. Problem Formulation

The core objective of Visual-Inertial SLAM (VI-SLAM) is to concurrently estimate the state of a mobile robot and construct a map of the environment. Mathematically, this problem can be formulated as follows:

Let $\mathbf{x}_t$ represent the state of the robot at time t, which includes its position, $\mathbf{p}_t \in \mathbb{R}^3$, and orientation, $\mathbf{R}_t \in SO(3)$. The map of the environment is denoted by $\mathbf{m}$, consisting of a set of landmarks $\mathbf{m}_i \in \mathbb{R}^3$, where i indexes the landmarks.

The robot receives measurements from an Inertial Measurement Unit (IMU) and a stereo camera. The IMU provides linear acceleration $\mathbf{a}_t \in \mathbb{R}^3$ and angular velocity $\boldsymbol{\omega}_t \in \mathbb{R}^3$, while the camera yields observations of landmarks $\mathbf{z}_t$, which are the projections of $\mathbf{m}_i$ into the image plane.

The problem is to find the sequence of states $\mathbf{x}_1, \ldots, \mathbf{x}_T$ and the map $\mathbf{m}$ that best explain the sequence of observations $\mathbf{z}_1, \ldots, \mathbf{z}_T$ and IMU measurements $\mathbf{a}_1, \ldots, \mathbf{a}_T$, $\boldsymbol{\omega}_1, \ldots, \boldsymbol{\omega}_T$, under the motion and observation models:

$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \boldsymbol{\omega}_t)$$
$$\mathbf{z}_t = h(\mathbf{x}_t, \mathbf{m}, \mathbf{v}_t)$$

where f and h represent the motion and observation models, Using which the localization and the landmark mapping is performed. The same is implemented via EKF as follows:

A. Extended Kalman Filter

Prior:

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

Motion model:

$$x_{t+1} = f(x_t, u_t, w_t), \quad w_t \sim \mathcal{N}(0, W)$$
$$F_t := \frac{\partial f}{\partial x}(\mu_{t|t}, u_t, 0), \quad Q_t := \frac{\partial f}{\partial w}(\mu_{t|t}, u_t, 0)$$

Observation model:

$$z_t = h(x_t, v_t), \quad v_t \sim \mathcal{N}(0, V)$$
$$H_t := \frac{\partial h}{\partial x}(\mu_{t|t-1}, 0), \quad R_t := \frac{\partial h}{\partial v}(\mu_{t|t-1}, 0)$$

Prediction:

$$\mu_{t+1|t} = f(\mu_{t|t}, u_t, 0)$$
$$\Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^T + Q_t W Q_t^T$$

Update:

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

Kalman gain:

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

III. Technical Approach

The proposed approach to visual-inertial simultaneous localization and mapping (VI-SLAM) is based on the fusion of inertial measurements from an IMU and visual data from a stereo camera. The process is divided into two main steps: the prediction step, which propagates the state estimate forward in time, and the update step, which corrects the state estimate using new measurements.

A. IMU Localization via EKF Prediction

In the IMU localization via EKF prediction, the goal is to estimate the pose $T_t \in SE(3)$ of the IMU over time using the SE(3) kinematics equations and the linear and angular velocity measurements from the IMU. The process involves the following steps:

1) **Relative Transformation:** The state of the system at time t, represented as T_t, includes the position and orientation of the robot in the world frame. The relative transformation between t and $t+1$, ${}_tT_{t+1}$ is obtained as follows:
$$ {}_tT_{t+1} = \exp(\tau_t \hat{u}_t) $$
2) **EKF Prediction:** The Extended Kalman Filter (EKF) prediction step is used to estimate the state at the next time step based on the motion model. The predicted state $\mu_{t+1|t}$ and its covariance $\Sigma_{t+1|t}$ are computed as follows:
$$ \begin{aligned} T_{t+1|t} &= T_{t|t} \exp(\tau_t \hat{u}_t) \\ \Sigma_{t+1|t} &= \exp(-\tau_t \breve{u}_t) \Sigma_{t|t} \exp(-\tau_t \breve{u}_t)^\top + W \end{aligned} $$
where
$$ \begin{aligned} u_t &= \begin{bmatrix} v_t \\ \omega_t \end{bmatrix} \in \mathbb{R}^6 \\ \hat{u}_t &= \begin{bmatrix} \hat{\omega}_t & v_t \\ 0^\top & 0 \end{bmatrix} \in \mathbb{R}^{4\times 4} \\ \breve{u}_t &= \begin{bmatrix} \hat{\omega}_t & \hat{v}_t \\ 0 & \hat{\omega}_t \end{bmatrix} \in \mathbb{R}^{6\times 6} \end{aligned} $$

The EKF prediction step provides an estimate of the IMU pose at each time step, which is then refined in the update step using visual observations from the camera system in the full visual-inertial SLAM algorithm.

B. Landmark mapping via EKF update

In the landmark mapping via EKF update, the goal is to estimate the positions of the landmarks based on the visual observations and the IMU pose estimates. At every time step, only certain number of features are visible and the missing features are represented by $[-1,-1,-1,-1]$. Hence the visible features are extracted at every timestamp based on this criteria.

1) **Bearing Measurement Triangulation:** Bearing measurement triangulation is used to estimate the position of landmarks using the disparity between the same feature observed in the left and right images to compute the depth of the feature. The mathematical formulation is as follows:
Given a feature $[u_L, v_L, u_R, v_R]$, the pixel coordinates in the left and right cameras are extracted as $z_L = [u_L, v_L]$ & $z_R = [u_R, v_R]$. These are then transformed into optical frame as $K^{-1} * \underline{z_L}$, where $\underline{z_L}$ is homogeneous coordinates of z_L. The depth is calculated as
$$ d = \frac{b}{u_L - u_R} $$
The 3D position m of the feature in the camera frame can then be obtained by:
$$ m = \begin{bmatrix} d \cdot u_L \\ d \cdot v_L \\ d \end{bmatrix} $$
2) **Landmark Initialization:** To initialize the landmarks, these feature coordinates from bearing triangulation are transformed into the world frame using $T_t * {}_{imu}T_{cam} * m$. These landmarks are then populated in the landmarks matrix if it has not been observed previously.
3) **Visual Mapping via EKF:** Predicted observation based on m_t and known correspondences Δ_{t+1}:
$$ \tilde{z}_{t+1,i} = K_s \pi \left({}_{cam}T_{imu} T_{t+1}^{-1} m_{t,i} \right) $$
for $i = 1, \ldots, N_{t+1}$.
Jacobian of $\tilde{z}_{t+1,i}$ with respect to m_j evaluated at $m_{t,j}$:
$$ H_{t+1,i,j} = K_s \frac{d\pi}{dq} \left({}_{cam}T_{imu} T_{t+1}^{-1} \mu_{t,j} \right) {}_{cam}T_{imu} T_{t+1}^{-1} P^\top $$
EKF update:
$$ \begin{aligned} K_{t+1} &= \Sigma_t H_{t+1}^T (H \Sigma_t H_{t+1}^T + V)^{-1} \\ m_{\text{updated}} &= m + K_{t+1}(z_{t+1} - z_{\text{pred}}) \\ \Sigma_{\text{updated}} &= (I - K_{t+1} H_{t+1}) \Sigma_t \end{aligned} $$

where T represents the transformation matrix, m denotes the landmarks, z is the observed feature positions, K is the Kalman gain, Σ_t is the covariance matrix, V is the observation noise covariance, and $\pi(\cdot)$ represents the projection operation.

Visual data from the stereo camera are processed to extract feature points, which are tracked across consecutive frames to establish correspondences. The correspondences are used to triangulate the positions of landmarks, providing measurements for the SLAM update step.

C. Visual Inertial SLAM

To perform the Visual Inertial SLAM, both the prediction and update state will be performed together at every time step. The implementation is explained below.

A combined covariance matrix is initialized as follows:
$$ \Sigma_{t,\text{SLAM}} = \begin{bmatrix} \Sigma_{LL} & \Sigma_{LR} \\ \Sigma_{RL} & \Sigma_{RR} \end{bmatrix} \in \mathbb{R}^{(3M+6)\times(3M+6)} $$
where Σ_{LL} is the covariance of the Landmarks, Σ_{RR} is the covariance of the Robot pose, Σ_{RL} and Σ_{LR} are the cross covariance between Landmarks and Robot pose.

1) **Relative Transformation:** The state of the system at time t, represented as T_t, includes the position and orientation of the robot in the world frame. The relative transformation between t and $t+1$, ${}_tT_{t+1}$ is obtained as follows:

$$ {}_tT_{t+1} = \exp(\tau_t \hat{u}_t) $$

2) **EKF Prediction Step:** The Extended Kalman Filter (EKF) prediction step is used to estimate the state at the next time step based on the motion model. The predicted state $\mu_{t+1|t}$ and its covariance $\Sigma_{t+1|t}$ are computed as follows:

$$ T_{t+1|t} = T_{t|t} \exp(\tau_t \hat{u}_t) $$

$$ \Sigma_{t+1|t,\text{SLAM}} = \begin{bmatrix} \Sigma_{LL} & \Sigma_{LR}F^\top \\ F\Sigma_{RL} & F\Sigma_{RR}F^\top + W \end{bmatrix} $$

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

$$ \hat{u}_t = \begin{bmatrix} \hat{\omega}_t & v_t \\ 0^\top & 0 \end{bmatrix} \in \mathbb{R}^{4\times 4} $$

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

where $F = \exp(-\tau_t \check{u}_t)$, W is the motion model noise covariance.

3) **Bearing Measurement Triangulation:** Bearing measurement triangulation is used to estimate the position of landmarks using the disparity between the same feature observed in the left and right images to compute the depth of the feature. The mathematical formulation is as follows:
Given a feature $[u_L, v_L, u_R, v_R]$, the pixel coordinates in the left and right cameras are extracted as $z_L = [u_L, v_L]$ & $z_R = [u_R, v_R]$. These are then transformed into optical frame as $K^{-1} * \underline{z_L}$, where $\underline{z_L}$ is homogeneous coordinates of z_L. The depth is calculated as

$$ d = \frac{b}{u_L - u_R} $$

The 3D position m of the feature in the camera frame can then be obtained by:

$$ m = \begin{bmatrix} d \cdot u_L \\ d \cdot v_L \\ d \end{bmatrix} $$

4) **Landmark Initialization:** To initialize the landmarks, these feature coordinates from bearing triangulation are transformed into the world frame using $T_t * {}_{imu}T_{cam} * m$. These landmarks are then populated in the landmarks matrix if it has not been observed previously.

5) **EKF Update Step:** The mean and variance of the predicted poses and the landmarks are updated in this step. Predicted observation based on m_t and known correspondences Δ_{t+1}:

$$ \tilde{z}_{t+1,i} = K_s \pi \left({}_cT_i T_{t+1}^{-1} m_{t,i} \right) $$

for $i = 1, \ldots, N_{t+1}$.
Jacobian of $\tilde{z}_{t+1,i}$ with respect to m_j evaluated at $m_{t,j}$:

$$ H_{t+1,map} = K_s \frac{d\pi}{dq} \left({}_cT_i T_{t+1}^{-1} m_t \right) {}_cT_i T_{t+1}^{-1} P^T $$

$$ H_{t+1,pose} = -K_s \frac{d\pi}{dq} \left({}_cT_i T_{t+1|t}^{-1} m_j \right) {}_cT_i \left(\mu_{t+1|t}^{-1} m_j \right) \odot $$

EKF update:

$$ K_{t+1} = \Sigma_t H_{t+1}^T (H \Sigma_t H_{t+1}^T + V)^{-1} $$

$$ m_{t+1|t+1} = m_{t+1|t} + K_{t+1}(z_{t+1} - \tilde{z}_{t+1}) $$

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

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

where ${}_cT_i$ is the transfromation from IMU to camera. T represents the transformation matrix, m denotes the landmarks, z is the observed feature positions, K is the Kalman gain, Σ_t is the covariance matrix, V is the observation noise covariance, and $\pi(\cdot)$ represents the projection operation.

IV. Results

A. Dataset - 10

The algorithm has been tested on two datasets, the results for the dataset - 10 is discussed in this section.

1) IMU Localization via EKF: Fig.1 shows the trajectory that is obtained through IMU based Localization. This would not be accurate as this trajectory will have accumulated error, which over time can alter the trajectory significantly

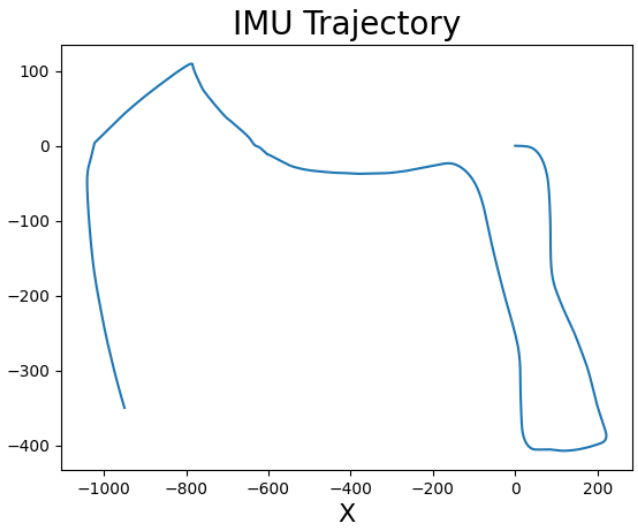

Fig. 1. IMU based Trajectory - dataset 10

2) Landmark Mapping via EKF Update: Fig.2 shows the Landmark mapping that is based on the assumption that the IMU Trajectory is accurate. Only the landmark prediction is happening, hence we can see that the trajectory looks same as the IMU Trajectory.

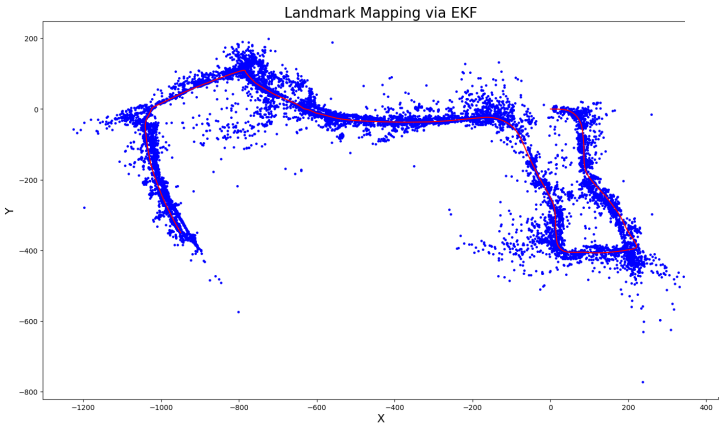

Fig. 2. Landmark Mapping via EKF Update - dataset 10

3) Visual Inertial SLAM: Fig.3 shows the complete Visual Inertial SLAM. Both the prediction and update step are being implemented. We can see that the trajectory has been significantly corrected from the IMU based trajectory. Along with it we can see that the landmarks have been updated accordingly.

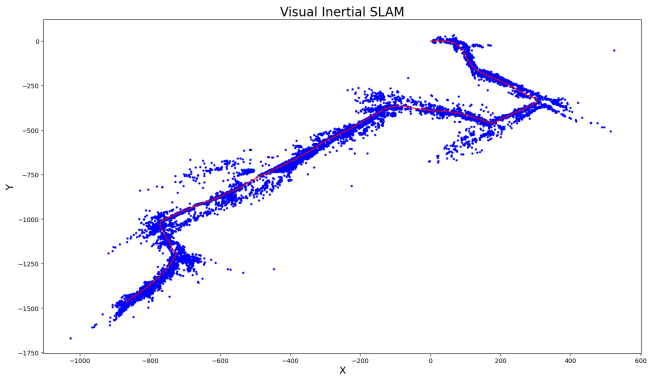

Fig. 3. Visual Inertial SLAM - dataset 10

Fig.4 shows the SLAM that was implemented with increased motion model noise compared to Fig.3, we can see that the noise affects the trajectory considerably.

B. Dataset - 03

The algorithm has been tested on two datasets, the results for the dataset - 03 is discussed in this section.

1) IMU Localization via EKF: Fig.4 shows the trajectory that is obtained through IMU based Localization. This would not be accurate as this trajectory will have accumulated error, which over time can alter the trajectory significantly

2) Landmark Mapping via EKF Update: Fig.5 shows the Landmark mapping that is based on the assumption that the IMU Trajectory is accurate. Only the landmark prediction is

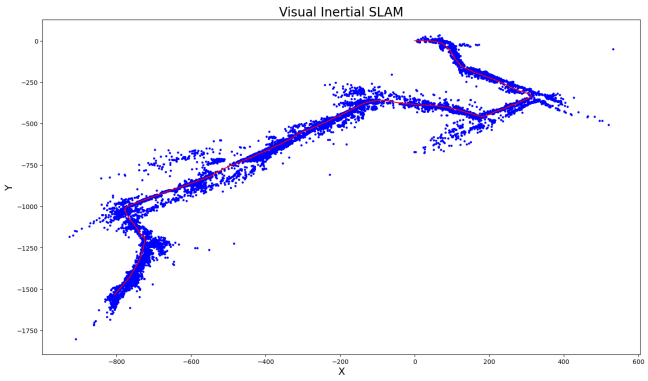

Fig. 4. Visual Inertial SLAM - dataset 10 - Increased noise

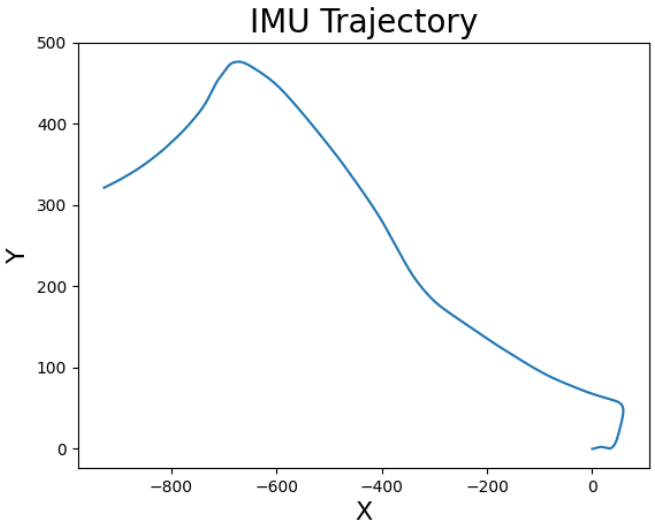

Fig. 5. IMU based Trajectory - dataset 03

happening, hence we can see that the trajectory looks same as the IMU Trajectory.

3) Visual Inertial SLAM: Fig.6 shows the complete Visual Inertial SLAM. Both the prediction and update step are being implemented. We can see that the trajectory has been significantly corrected from the IMU based trajectory. Along with it we can see that the landmarks have been updated accordingly.

For the dataset-3, Fig.8 shows the SLAM implemented for significantly higher motion model noise, but the trajectory seems robust to the noise with only minor difference in the trajectrory.

REFERENCES

[1] piazza
[2] Lecture 6: Localization and Odometry from Point Features
[3] Lecture 12 - Visual Inertial SLAM

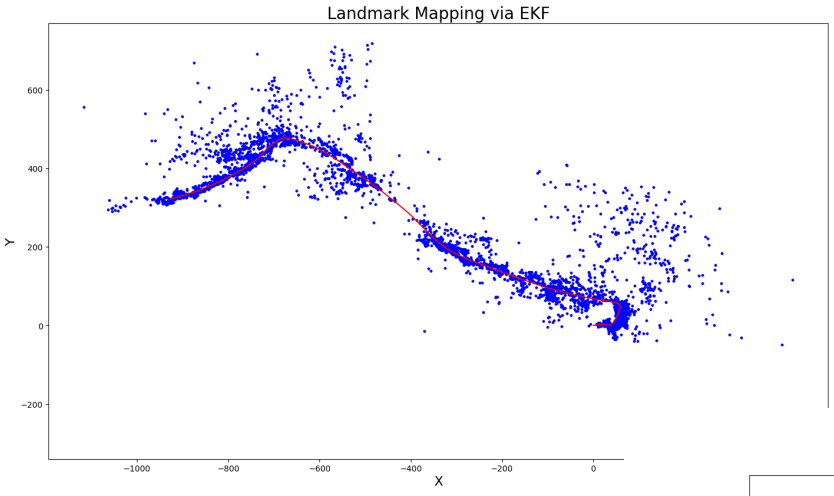

Fig. 6. Landmark Mapping via EKF Update - data

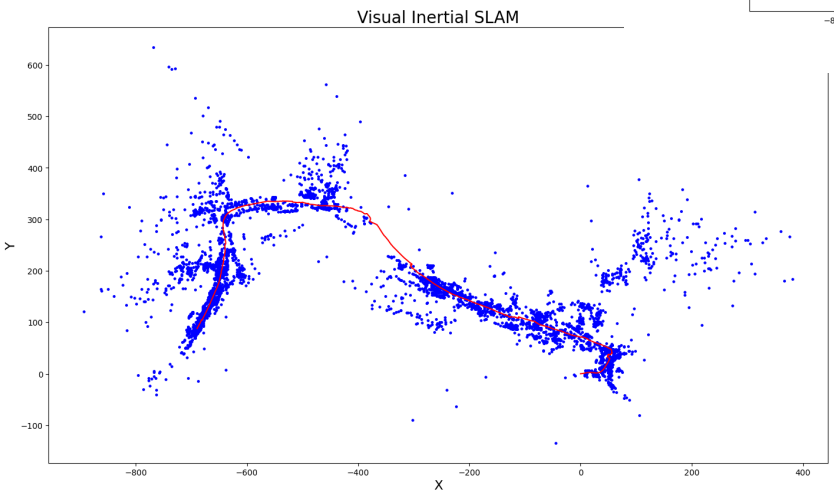

Fig. 7. Visual Inertial SLAM - dataset 03

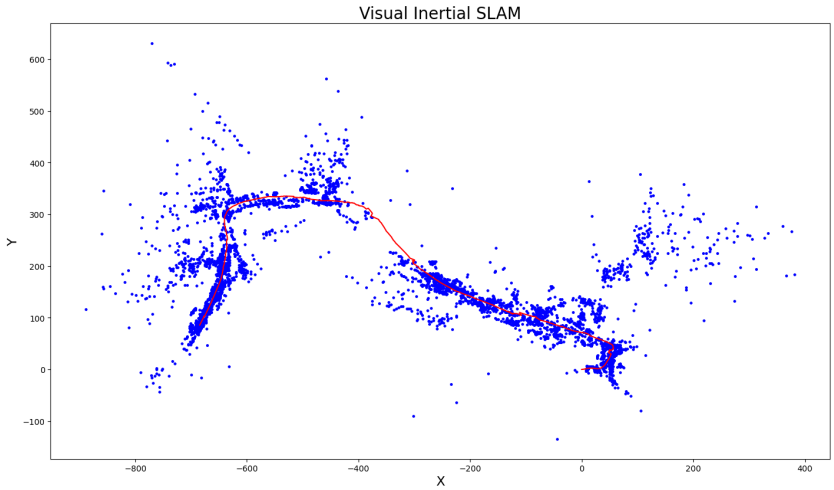

Fig. 8. Visual Inertial SLAM - dataset 03 - Increased Noise