EXTENDED KALMAN FILTER BASED VISUAL INERTIAL SLAM

Chun-Nien Chan
Department of Electrical and Computer Engineering
University of California, San Diego
chc030@eng.ucsd.edu

1. INTRODUCTION

Simultaneous Localization And Mapping (SLAM) is a challenging topic in robotics and has been researched for a few decades. When building a map from the observations of a robot, a good estimate of the robot's location is necessary. However, a robot needs a consistent and reliable map for localization. SLAM is the computational problem to simultaneously estimates a map of the environment and pose of a moving robot relative to that map, without any a-priori information external to the robot except for the observations of the robot. SLAM approaches are capable of building the map online while correcting the errors of its pose estimate as it sensing the surroundings.

Over decades, several approaches for SLAM based on particle filter[1][2], extended Kalman filter[3], and neural networks[4] has been researched. These approaches are designed to build maps in different representations, including landmark-based representation, surfels, polygonal mesh, and occupancy grid. Nowadays, they are widely used for applications such as motion planning of robots in unknown environments [5] and employed in self-driving cars, unmanned aerial vehicles, and autonomous underwater vehicles.

In this paper, we propose a solution for visual-inertial SLAM based on Extended Kalman Filter(EKF) and landmark. We evaluated the proposed solution with real-world measurements from an IMU and a stereo camera installed in a car. It can estimate reliable maps and trajectory on various datasets in a reasonable time.

2. PROBLEM FORMULATION

2.1. Simultaneous Localization And Mapping

The SLAM problem can be described as a probabilistic Markov Chain. Given robot's pose $\mathbf{x}_t$ and control input $\mathbf{u}_t$ at discrete time steps t, the pose in the following time step $t+1$ is a probabilistic function:

$$\begin{aligned} \mathbf{x}_{t+1} &= f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) \sim p_f(\cdot|\mathbf{x}_t, \mathbf{u}_t) \\ \mathbf{w}_t &= \text{motion noise} \end{aligned} \tag{1}$$

which is also known as the motion model.

To map the environment, the robot may sense the environment to get observation at each time step. Denote the observation at time t as $\mathbf{z}_t$ and the environment as $\mathbf{m}$, the sensor observations are governed by a probabilistic law:

$$\begin{aligned} \mathbf{z}_t &= h(\mathbf{x}_t, \mathbf{m}, \mathbf{v}_t) \sim p_h(\cdot|\mathbf{x}_t, \mathbf{m}) \\ \mathbf{v}_t &= \text{observation noise} \end{aligned} \tag{2}$$

which is also known as the observation model.

With motion model Eq.1 and observation model Eq.2, SLAM is the problem to determine the environment $\mathbf{m}$ and robot poses $\mathbf{x}_t$ from observations $\mathbf{z}_0, \ldots, \mathbf{z}_t$ and control inputs $\mathbf{u}_0, \ldots, \mathbf{u}_{t-1}$ at each time step t. The objective to compute can be written in probabilistic form:

$$p(\mathbf{m}, \mathbf{x}_t|\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1}) \tag{3}$$

The relation between $\mathbf{m}$ and $\mathbf{x}_t$ is difficult to determine. However, we can take advantages of the decomposition of the joint probability density function according to the Markov assumptions:

$$\begin{aligned} p(\mathbf{x}_{0:t}, \mathbf{z}_{0:t}, \mathbf{u}_{0:t-1}, \mathbf{m}) =& p_{0|0}(\mathbf{x}_0, \mathbf{m}) \\ & \prod_{i=1}^{t} p_h(\mathbf{z}_i|\mathbf{x}_i, \mathbf{m}) \\ & \prod_{i=1}^{t} p_f(\mathbf{x}_i|\mathbf{x}_{i-1}, \mathbf{u}_{i-1}) \end{aligned} \tag{4}$$

In practical implementations, maximum likelihood estimation (MLE) is used to find the optimal $\mathbf{x}_{0:t}$ and $\mathbf{m}$ in order to determine the poses of robot and the environment. The formulation can be written as:

$$\begin{aligned} &\max_{\mathbf{x}_{0:t}, \mathbf{m}} \sum_{i=0}^{t} \log(p_h(\mathbf{z}_i|\mathbf{x}_i, \mathbf{m})) \\ &+ \sum_{i-=1}^{t} \log(p_f(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{u}_{t-1})) \end{aligned} \tag{5}$$

2.2. Bayes Filtering

Bayes filtering is a probabilistic inference technique for estimating the state $\mathbf{x}_t$ of dynamical systems, like robot, that

combines evidence from control inputs and observations using the Markov assumptions and Bayes rule. The Bayes filter relies on two steps to keep track of $p_{t|t}(\mathbf{x}_t)$ and $p_{t+1|t}(\mathbf{x}_{t+1})$

2.2.1. Prediction Step

Given a prior density $p_{t|t}$ over $\mathbf{x}_t$ and the control input $\mathbf{u}_t$, we use the motion model p_f to compute the predicted density $p_{t+1|t}$ over $\mathbf{x}_{t+1}$ as the following equation:

$$p_{t+1|t}(\mathbf{x}) = \int p_f(\mathbf{x}|\mathbf{s}, \mathbf{u}_t) p_{t|t}(\mathbf{s}) d\mathbf{s} \tag{6}$$

2.2.2. Update Step

Given the predicted density $p_{t+1|t}$ over $\mathbf{x}_{t+1}$ and the measurement $\mathbf{z}_{t+1}$, we use the observation model p_h to incorporate the measurement information and obtain the posterior $p_{t+1|t+1}$ over $\mathbf{x}_{t+1}$ as the following equation:

$$p_{t+1|t+1}(\mathbf{x}) = \frac{p_h(\mathbf{z}_{t+1}|\mathbf{x}) p_{t+1|t}(\mathbf{x})}{\int p_h(\mathbf{z}_{t+1}|\mathbf{s}) p_{t+1|t}(\mathbf{s}) d\mathbf{s}} \tag{7}$$

2.3. Landmark-based Mapping

The landmark-based mapping algorithm aims to address the problem of generating a map of the environment from noisy and uncertain sensor observations $\mathbf{z}$ with the assumption that the poses of the robot $\mathbf{x}$ 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 $\mathbf{m} \in \mathbb{R}^{3\times M}$

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 $\mathbf{x}$ and the observation $\mathbf{z}_t$. Therefore, we can define the observation model as follows:

$$p(\mathbf{z}_t|\mathbf{x}_t, \mathbf{m}, \mathbf{n}_t) \tag{8}$$

$\mathbf{n}_t$ is the index map with $|\mathbf{n}_t| = N_t$, where N_t is the number of observed landmarks at time t and $\mathbf{n}_{i,t}$ is the index of the landmark corresponding to the i_{th} observation $\mathbf{z}_{i,t}$.

2.4. Sensors Configuration

Our proposed solution aims to solve the SLAM problem with observations from an IMU and a stereo camera installed in a car. The IMU observations contain the linear velocity $\mathbf{v}_t \in \mathbb{R}^3$ and angular velocity $\omega_t \in \mathbb{R}^3$ in the frame of the IMU. The stereo camera data are pre-computed to extract the visual features and find the correspondence between left and right camera frames across time steps (data association). Fig.1 shows the visual features extracted by the stero camera. The visual features at time t is denoted as $\mathbf{z}_t \in \mathbb{R}^{4M}$, where the i_{th} column contains the pixel coordinates of landmark i in the left and right camera images. However, for a specific time step t, there might be some landmarks that are not observable. If the i_{th} landmark is not observed at time t, the i_{th} column of observation $\mathbf{z}_t$ would be $[-1, -1, -1, -1]^T$.

In our configuration, we assume that the transformation from the IMU to the camera optical frame ${}_O\mathbf{T}_I \in SE(3)$ (extrinsic parameters) and the stereo camera calibration matrix $\mathbf{M}$ (intrinsic parameters) are known. The camera calibration matrix is defined as the following equation:

$$\mathbf{M} = \begin{bmatrix} fs_u & 0 & c_u & 0 \\ 0 & fs_v & c_b & 0 \\ fs_u & 0 & c_u & -fs_u b \\ 0 & fs_v & c_v & 0 \end{bmatrix} \tag{9}$$

where f is the focal length, s_u, s_v are pixel scaling, c_u and c_v are the principle points, and b is the stereo baseline.

3. TECHNICAL APPROACH

3.1. Extended Kalman Filter

The Extended Kalman Filter (EKF) is a nonlinear version of the Kalman Filter, which linearizes about an estimate of the current mean and covariance with a moment matching approach.

The nonlinear Kalman Filter is a Bayes filter with the following assumptions:

1. The prior pdf $p_{0|0}$ is Gaussian
2. The state $\mathbf{x}_{t+1}$ is affected by Gaussian noise, that is:
$$\mathbf{x}_{t+1} = f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t), \mathbf{w}_t \sim \mathcal{N}(0, \mathbf{W}) \tag{10}$$
3. The observation $\mathbf{z}_t$ is affected by Gaussian noise, that is:
$$\mathbf{z}_t = h(\mathbf{x}_t, \mathbf{v}_t), \mathbf{v}_t \sim \mathcal{N}(0, \mathbf{V}) \tag{11}$$
4. The process noise $\mathbf{w}_t$ and measurement noise $\mathbf{v}_t$ are independent of each other, of the state $\mathbf{x}_t$ and across time
5. The posterior pdf is forced to be Gaussian via approximation

The challenge of the nonlinear Kalman Filter is that the predicted and updated pdfs are not Gaussian and cannot be evaluated in closed form. Using moment matching, we can force the predicted and updated pdfs to be Gaussian by evaluating their first and second moments and approximating them with Gaussians with the same moments.

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

The Extended Kalman Filter uses a first-order Taylor series to approximate the integrals required to implement the nonlinear Kalman Filter. Thus the approximation of the motion model would be as follow:

$$\begin{aligned} f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) &\approx f(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0) + \left[\frac{df}{d\mathbf{x}}(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0)\right](\mathbf{x}_t - \boldsymbol{\mu}_{t|t}) \\ &\quad + \left[\frac{df}{d\mathbf{w}}(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0)\right](\mathbf{w}_t - 0) \\ &\approx f(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0) + \mathbf{F}_t(\mathbf{x}_t - \boldsymbol{\mu}_{t|t}) + \mathbf{Q}_t\mathbf{w}_t \end{aligned} \tag{12}$$

where

$$\begin{aligned} \mathbf{F}_t &= \frac{df}{dx}(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0), \\ \mathbf{Q}_t &= \frac{df}{dw}(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0) \end{aligned} \tag{13}$$

The approximation of the observation model would be as follow:

$$\begin{aligned} h(\mathbf{x}_{t+1}, \mathbf{v}_{t+1}) &\approx h(\boldsymbol{\mu}_{t+1|t}, 0) \\ &\quad + \left[\frac{dh}{dx}(\boldsymbol{\mu}_{t+1|t}, 0)\right](\mathbf{x}_{t+1} - \boldsymbol{\mu}_{t+1|t}) \\ &\quad + \left[\frac{dh}{d\mathbf{v}}(\boldsymbol{\mu}_{t+1|t}, 0)\right](\mathbf{v}_{t+1} - 0) \\ &\approx h(\boldsymbol{\mu}_{t+1|t}, 0) + \mathbf{H}_{t+1}(\mathbf{x}_{t+1} - \boldsymbol{\mu}_{t+1|t}) \\ &\quad + \mathbf{R}_{t+1}\mathbf{v}_{t+1} \end{aligned}$$

$$\text{where } \mathbf{H}_{t+1} = \frac{dh}{d\mathbf{x}}(\boldsymbol{\mu}_{t+1|t}, 0), \text{ and } \mathbf{R}_{t+1} = \frac{dh}{d\mathbf{v}}(\boldsymbol{\mu}_{t+1|t}, 0) \tag{14}$$

Based on the equations above, the models of Extended Kalman Filter will be the following equations:

Prior:

$$\mathbf{x}_t|\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} \sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}) \tag{15}$$

Motion model:

$$\begin{aligned} \mathbf{x}_{t+1} &= f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) \\ \mathbf{w}_t &\sim \mathcal{N}(0, \mathbf{W}) \\ \mathbf{F}_t &:= \frac{df}{d\mathbf{x}}(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0) \\ \mathbf{Q}_t &:= \frac{df}{d\mathbf{w}}(\boldsymbol{\mu}_{t|t}, \mathbf{u}_t, 0) \end{aligned} \tag{16}$$

Observation model:

$$\begin{aligned} \mathbf{z}_t &= h(\mathbf{x}_t, \mathbf{v}_t) \\ \mathbf{v}_t &\sim \mathcal{N}(0, \mathbf{V}) \\ \mathbf{H}_t &:= \frac{dh}{d\mathbf{x}}(\boldsymbol{\mu}_{t|t-1}, 0) \\ \mathbf{R}_t &:= \frac{dh}{d\mathbf{v}}(\boldsymbol{\mu}_{t|t-1}, 0) \end{aligned} \tag{17}$$

Prediction:

$$\begin{aligned} \boldsymbol{\mu}_{t+1|t} &= f(\boldsymbol{\mu}_{t|t}, u_t, 0) \\ \boldsymbol{\Sigma}_{t+1|t} &= \mathbf{F}_t\boldsymbol{\Sigma}_{t|t}\mathbf{F}_t^T + \mathbf{Q}_t\mathbf{W}\mathbf{Q}_t^T \end{aligned} \tag{18}$$

Update:

$$\begin{aligned} \boldsymbol{\mu}_{t+1|t+1} &= \boldsymbol{\mu}_{t+1|t} + \mathbf{K}_{t+1|t}(\mathbf{z}_{t+1} - h(\boldsymbol{\mu}_{t+1|t}, 0)) \\ \boldsymbol{\Sigma}_{t+1|t+1} &= (\mathbf{I} - \mathbf{K}_{t+1|t}\mathbf{H}_{t+1})\boldsymbol{\Sigma}_{t+1|t} \end{aligned} \tag{19}$$

Kalman Gain:

$$\mathbf{K}_{t+1|t} := \boldsymbol{\Sigma}_{t+1|t}\mathbf{H}_{t+1}^T(\mathbf{H}_{t+1t+1|t}\mathbf{H}_{t+1}^T + \mathbf{R}_\mathbf{V}\mathbf{R}_{t+1}^T)^{-1} \tag{20}$$

3.2. EKF-based Visual Mapping

For the landmark-based visual mapping problem, we assume that the inverse IMU pose $\mathbf{U}_t = {}_w\mathbf{T}_{I,t}^{-1} \in SE(3)$ is known. Moreover, as described in Sec.2.3 and Sec.2.4, we assume that the landmarks are static and the data association $\pi_t = \{1, \dots, M\} \rightarrow \{1, \dots, N_t\}$ stipulating which landmarks were observed at each time t is pre-computed by an external algorithm. The observation model can be written with the Gaussian prior and observation noise:
Prior:

$$\begin{aligned} \mathbf{m}|\mathbf{z}_{0:t} &\sim \mathcal{N}(\mu_t, \mathbf{\Sigma}_t) \\ \boldsymbol{\mu}_t &\in \mathbb{R}^{3M} \\ \mathbf{\Sigma}_t &\in \mathbb{R}^{3M \times 3M} \end{aligned} \tag{21}$$

Observation model:

$$\begin{aligned} \mathbf{z}_{t,i} &= h(\mathbf{U}_t, \mathbf{m}_j) + \mathbf{v}_{t,i} \\ &= \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t, \underline{\mathbf{m}}_j) + \mathbf{v}_{t,i} \\ \mathbf{m}|\mathbf{z}_{0:t} &\sim \mathcal{N}(\boldsymbol{\mu}_t, \mathbf{\Sigma}_t) \\ \mathbf{v}_{t,i} &\sim \mathcal{N}(0, \mathbf{V}) \end{aligned} \tag{22}$$

Homogeneous coordinate:

$$\underline{\mathbf{m}} = [\mathbf{m}^T, 1]^T \tag{23}$$

where $\boldsymbol{\mu}_t \in \mathbb{R}^{3M}$ is the expectation of locations of all the landmarks, $\mathbf{\Sigma}_t \in \mathbb{R}^{3M \times 3M}$ is the covariance of the estimate, and $\mathbf{M}$ is the calibration matrix in Eq.(9). The projection function π and its derivative are defined as follows:

$$\begin{aligned} \pi(\mathbf{q}) &= \frac{1}{q_3}\mathbf{q} \in (R)^4 \\ \frac{d\pi}{d\mathbf{q}}(\mathbf{q}) &= \frac{1}{q_3}\begin{bmatrix} 1 & 0 & -\frac{q_1}{q_3} & 0 \\ 0 & 1 & -\frac{q_2}{q_3} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{q_4}{q_3} & 1 \end{bmatrix} \end{aligned} \tag{24}$$

We can stack all the observation $\mathbf{z}_{t,i}$ as a single vector $\mathbf{z}_t \in \mathbb{R}^{4N_t}$ and rewrite the observation model equation as follows:

$$\begin{aligned} \mathbf{z}_t &= \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t\underline{\mathbf{m}}) + \mathbf{v}_t \\ \mathbf{v}_t &\sim \mathcal{N}(\mathbf{0}, \mathbf{I} \otimes \mathbf{V}) \end{aligned} \tag{25}$$

where $\otimes$ is the Kronecker product.

For Extended Kalman Filter, we need to derive the differentiation of the observation model with respect to $\mathbf{m}$ evaluated with $\boldsymbol{\mu}_t$. Consider a small perturbation $\delta\boldsymbol{\mu}_{t,j}$ for the location of landmark j:

$$\mathbf{m}_j = \boldsymbol{\mu}_{t,j} + \delta\boldsymbol{\mu}_{t,j} \tag{26}$$

The first-order Taylor series approximation to observation i at time t can be then written as:

$$\begin{aligned} \mathbf{z}_{t,i} &= \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t(\boldsymbol{\mu}_{t,j} + \delta\boldsymbol{\mu}_{t,j})) + \mathbf{v}_{t,i} \\ &= \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t(\underline{\boldsymbol{\mu}}_{t,j} + \mathbf{P}^T\delta\boldsymbol{\mu}_{t,j})) + \mathbf{v}_{t,i} \\ &\approx \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t\underline{\boldsymbol{\mu}}_{t,j}) + \\ &\quad \mathbf{M}\frac{d\pi}{d\mathbf{q}}({}_O\mathbf{T}_I\mathbf{U}_t\underline{\boldsymbol{\mu}}_{t,j}){}_O\mathbf{T}_I\mathbf{U}_t\mathbf{P}^T\delta\boldsymbol{\mu}_{t,j} + \mathbf{v}_{t,i} \end{aligned} \tag{27}$$

where $\mathbf{P} = [\mathbf{I}, \mathbf{0}] \in \mathbb{R}^{3\times4}$ is the projection matrix.

With the equations mentioned above, the Extended Kalman Filter update steps for landmark mapping are listed as follows:
Predicted observations:

$$\begin{aligned} \tilde{\mathbf{z}}_{t,i} &= \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t\underline{\boldsymbol{\mu}}_{t,j}) \in \mathbb{R}^4 \\ &\text{for } i = 1, \dots, N_t \end{aligned} \tag{28}$$

Observation matrix:

$$\begin{aligned} \mathbf{H}_{t,i,j} &= \begin{cases} \mathbf{M}\frac{d\pi}{d\mathbf{q}}({}_O\mathbf{T}_I\mathbf{U}_t\underline{\boldsymbol{\mu}}_{t,j}){}_O\mathbf{T}_I\mathbf{U}_t\mathbf{P}^T \\ \quad \text{- if observation } i \text{ corresponds to landmark } j \\ \quad \text{at time } t \\ \mathbf{0} \in \mathbb{R}^{4\times3} \\ \quad \text{- otherwise} \end{cases} \\ \mathbf{H}_t &\in \mathbb{R}^{4N_t \times 3M} \end{aligned} \tag{29}$$

EKF Update:

$$\mathbf{K}_t = \mathbf{\Sigma}_t\mathbf{H}_t^T(\mathbf{H}_t\mathbf{\Sigma}_t\mathbf{H}_t^T + \mathbf{I} \otimes \mathbf{V})^{-1} \tag{30}$$

$$\boldsymbol{\mu}_{t+1} = \boldsymbol{\mu}_t + \mathbf{K}_t(\mathbf{z}_t - \tilde{\mathbf{z}}_t) \tag{31}$$

$$\mathbf{\Sigma}_{t+1} = (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\mathbf{\Sigma}_t \tag{32}$$

Since we assume all the landmarks are static, there is no need to perform prediction steps in EKF-based visual-mapping.

For each landmark, we initialize its position at the first time step t we observe it. The following equation with the observation $\mathbf{z}_{t,i}$ is used to compute the landmark initialization:

$$\mathbf{z}_{t,i} = \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t\mathbf{m}_j) \tag{33}$$

3.3. EKF-based Visual-Inertial Odometry

The localization problem aims to estimate the inverse IMU pose of the robot $\mathbf{U}_t = {}_w\mathbf{T}_{I,t}^{-1} \in SE(3)$ given the IMU measurements $\mathbf{u}_t = [\mathbf{v}_t^T, \omega_t^T]^T$, the visual feature observation $\mathbf{z}_{0:T}$, and the landmark coordinates $\mathbf{m} \in \mathbb{R}^{3\times M}$ in the world frame. With the same assumption in the visual mapping problem, data association between observations and landmarks is pre-computed by an external algorithm. The motion model

can be written with the Gaussian prior and process noise:
Prior:

$$\begin{aligned} \mathbf{U}_t|\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} &\sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}) \\ \boldsymbol{\mu}_{t|t} &\in SE(3) \\ \boldsymbol{\Sigma}_{t|t} &\in \mathbb{R}^{6\times 6} \end{aligned} \tag{34}$$

Motion model:

$$\begin{aligned} \mathbf{U}_{t+1} &= \exp(-\tau((\mathbf{u}_t + \mathbf{w}_t))\wedge)\mathbf{U}_t \\ \mathbf{u}_t &= [\mathbf{v}_t^T, \omega_t^T]^T \\ \mathbf{U}_t|\mathbf{z}_{0:t}, \mathbf{u}_{0:t-1} &\sim \mathcal{N}(\boldsymbol{\mu}_{t|t}, \boldsymbol{\Sigma}_{t|t}) \\ \mathbf{w}_{t,i} &\sim \mathcal{N}(0, \mathbf{W}) \end{aligned} \tag{35}$$

where τ is the time difference in seconds between two contiguous frames.

In order to separate the effect of the noise $\mathbf{w}_t$ from the motion of the deterministic part of ${}_w\mathbf{T}_{I,t} = \mathbf{U}_t^{-1}$, we take advantages of the discrete-time perturbation idea in [6][7] and rewrite the motion model in terms of nominal kinematics and zero-mean perturbation kinematics:

$$\begin{aligned} \boldsymbol{\mu}_{t+1|t} &= \exp(-\tau \mathbf{u}_t{}^{\wedge})\boldsymbol{\mu}_{t|t} \\ \delta\boldsymbol{\mu}_{t+1|t} &= \exp(-\tau \mathbf{u}_t{}^{\curlywedge})\delta\boldsymbol{\mu}_{t|t} + \mathbf{w}_t \end{aligned} \tag{36}$$

where

$$\begin{aligned} \mathbf{u}_t &= \begin{bmatrix} \mathbf{v}_t \\ \omega_t \end{bmatrix} \in \mathbb{R}^6 \\ \mathbf{u}_t^{\wedge} &= \begin{bmatrix} \omega_t^{\wedge} & \mathbf{v}_t \\ \mathbf{0}^T & 0 \end{bmatrix} \in \mathbb{R}^{4\times 4} \\ \mathbf{u}_t^{\curlywedge} &= \begin{bmatrix} \omega_t^{\wedge} & \mathbf{v}_t^{\wedge} \\ 0 & \omega_t^{\wedge} \end{bmatrix} \in \mathbb{R}^{6\times 6} \end{aligned} \tag{37}$$

With the motion model defined in Eq.(36), the Extended Kalman Filter prediction steps are listed as follows:

$$\begin{aligned} \boldsymbol{\mu}_{t+1|t} &= \exp(-\tau \mathbf{u}_t^{\wedge})\boldsymbol{\mu}_{t|t} \\ \boldsymbol{\Sigma}_{t+1|t} &= \mathbb{E}[\delta\boldsymbol{\mu}_{t+1|t}\delta\boldsymbol{\mu}_{t+1|t}^T] \\ &= \exp(-\tau \mathbf{u}_t^{\curlywedge})\boldsymbol{\Sigma}_{t|t}\exp(-\tau \mathbf{u}_t^{\curlywedge})^T + \mathbf{W} \end{aligned} \tag{38}$$

The observation model is the same as the definition in the visual mapping problem in Eq.(22). In order to derive the update steps for Extended Kalman Filter, we need the observation model Jacobian $\mathbf{H}_{t+1|t} \in \mathbb{R}^{4N_t\times 6}$ with respect to the inverse IMU pose $\mathbf{U}_t$ evaluated at $\boldsymbol{\mu}_{t+t|t}$. The first-order Taylor series approximation of observation i at time $t+1$ using an inverse IMU pose perturbation $\delta\boldsymbol{\mu}_{t+1|t+1}$ is:

$$\begin{aligned} \mathbf{z}_{t,i} &= \mathbf{M}\pi({}_O\mathbf{T}_I\exp(\delta\boldsymbol{\mu}_{t+1|t}^{\wedge})\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j) + \mathbf{v}_{t+1,i} \\ &\approx \mathbf{M}\pi({}_O\mathbf{T}_I(\mathbf{I} + \delta\boldsymbol{\mu}_{t+1|t}^{\wedge})\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j) + \mathbf{v}_{t+1,i} \\ &= \mathbf{M}\pi({}_O\mathbf{T}_I\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j + {}_O\mathbf{T}_I(\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j)^{\odot}\delta\boldsymbol{\mu}_{t+1|t+1}) + \mathbf{v}_{t+1,i} \\ &\approx \mathbf{M}\pi({}_O\mathbf{T}_I\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j) \\ &\quad + \mathbf{M}\frac{d\pi}{d\mathbf{q}}({}_O\mathbf{T}_I\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j){}_O\mathbf{T}_I(\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j)^{\odot}\delta\boldsymbol{\mu}_{t+1|t+1} + \mathbf{v}_{t+1,i} \end{aligned} \tag{39}$$

where

$$\begin{bmatrix} \mathbf{s} \\ 1 \end{bmatrix}^{\odot} = \begin{bmatrix} \mathbf{I} & -\mathbf{s}^{\wedge} \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4\times 6} \tag{40}$$

Similar to the update steps in EKF-based visual mapping, the update steps for visual-inertial odometry can be composed of the following equations:
Prior:

$$\begin{aligned} \mathbf{U}_{t+1}|\mathbf{z}_{0:t}, \mathbf{u}_{0:t} &\sim \mathcal{N}(\boldsymbol{\mu}_{t+1|t}, \boldsymbol{\Sigma}_{t+1|t}) \\ \boldsymbol{\mu}_{t+1|t} &\in SE(3) \\ \boldsymbol{\Sigma}_{t+1|t} &\in \mathbb{R}^{6\times 6} \end{aligned} \tag{41}$$

Predicted observations:

$$\begin{aligned} \tilde{\mathbf{z}}_{t+1,i} &= \mathbf{M}\pi({}_O\mathbf{T}_I\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j) \\ &\text{for } i = 1, \ldots, N_t \end{aligned} \tag{42}$$

Observation matrix:

$$\begin{aligned} \mathbf{H}_{i,t+1|t} &= \mathbf{M}\frac{d\pi}{d\mathbf{q}}({}_O\mathbf{T}_I\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j){}_O\mathbf{T}_I(\boldsymbol{\mu}_{t+1|t}\mathbf{m}_j)^{\odot} \\ \mathbf{H}_{t+1|t} &= \begin{bmatrix} \mathbf{H}_{1,t+1|t} \\ \mathbf{H}_{2,t+1|t} \\ \vdots \\ \mathbf{H}_{N_{t+1},t+1|t} \end{bmatrix} \in \mathbb{R}^{4N_t\times 6} \end{aligned} \tag{43}$$

EKF Update:

$$\mathbf{K}_{t+1|t} = \boldsymbol{\Sigma}_{t+1|t}\mathbf{H}_{t+1|t}^T(\mathbf{H}_{t+1|t}\boldsymbol{\Sigma}_{t+1|t}\mathbf{H}_{t+1|t}^T + \mathbf{I}\otimes\mathbf{V})^{-1} \tag{44}$$

$$\boldsymbol{\mu}_{t+1|t+1} = \exp((\mathbf{K}_{t+1|t}(\mathbf{z}_{t+1} - \tilde{\mathbf{z}}_{t+1}))^{\wedge})\boldsymbol{\mu}_{t+1|t} \tag{45}$$

$$\boldsymbol{\Sigma}_{t+1|t+1} = (\mathbf{I} - \mathbf{K}_{t+1|t}\mathbf{H}_{t+1|t})\boldsymbol{\Sigma}_{t+1|t} \tag{46}$$

3.4. EKF SLAM

To achieve the goal of estimating position of landmarks and the pose of the robot simultaneously, the proposed idea is to merge the predict and update steps of Extended Kalman Filter based visual mapping and visual-inertial odometry. First of all, the joint estimated state and covariance under the Gaussian assumption are defined as follows:

$$\begin{aligned} \boldsymbol{\mu} &= \begin{bmatrix} \boldsymbol{\mu}_m \\ \boldsymbol{\mu}_p \end{bmatrix} \in \mathbb{R}^{(6+3M)} \\ \boldsymbol{\Sigma} &\in \mathbb{R}^{(6+3M)\times(6+3M)} \end{aligned} \tag{47}$$

where $\boldsymbol{\mu}_m$ is the estimated landmark position in Eq.(21) and $\boldsymbol{\mu}_p$ is the estimated six degrees of freedom of inverse IMU pose in Eq.(34)(41).
The Extended Kalman Filter predict step on the joint estimated state and covariance is derived form the predict step of

visual-inertial odometry in Eq.(36) only because all the landmarks are assumed static. The equations given the IMU measurement $\mathbf{u}_t$ are listed as follows:

$$\begin{aligned}\boldsymbol{\mu}_{t+1|t} &= \begin{bmatrix}\boldsymbol{\mu}_{m,t+1|t}\\ \boldsymbol{\mu}_{p,t+1|t}\end{bmatrix} = \begin{bmatrix}\boldsymbol{\mu}_{m,t+1|t}\\ \exp(-\tau\mathbf{u}_t^{\wedge})\boldsymbol{\mu}_{p,t|t}\end{bmatrix}\\ \boldsymbol{\Sigma}_{t+1|t} &= \mathbf{F}_t\boldsymbol{\Sigma}_{t|t}\mathbf{F}_t^T + \mathbf{W}\\ \mathbf{F}_t &= \begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \exp(-\tau\mathbf{u}_t^{\curlywedge})\end{bmatrix}\\ \mathbf{W} &= \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & \mathbf{W}_p\end{bmatrix}\end{aligned} \tag{48}$$

where $\mathbf{W}_p$ is the process noise covariance in Eq.(36)

The update step is formed by combining the update step of both visual mapping and visual-inertial odometry. The update step equations are listed as follows:
Predicted observations:

$$\begin{aligned}\tilde{\mathbf{z}}_{t,i} &= \mathbf{M}\pi({}_O\mathbf{T}_I\mathbf{U}_t\underline{\boldsymbol{\mu}}_{m,t,j}) \in \mathbb{R}^4\\ &\text{for } i = 1,\ldots,N_t\end{aligned} \tag{49}$$

Observation matrix:

$$\begin{aligned}\mathbf{H}_{t+1|t} &= \begin{bmatrix}\mathbf{H}_{m,t+1|t} & \mathbf{H}_{p,t+1|t}\end{bmatrix} \in \mathbb{R}^{4N_t\times(3M+6)}\\ \mathbf{H}_{m,t+1|t} &: \text{Observation matrix in Eq.(29)}\\ \mathbf{H}_{p,t+1|t} &: \text{Observation matrix in Eq.(43)}\end{aligned} \tag{50}$$

EKF Update:

$$\mathbf{K}_{t+1|t} = \boldsymbol{\Sigma}_{t+1|t}\mathbf{H}_{t+1|t}^T(\mathbf{H}_{t+1|t}\boldsymbol{\Sigma}_{t+1|t}\mathbf{H}_{t+1|t}^T + \mathbf{I}\otimes\mathbf{V})^{-1} \tag{51}$$

$$\begin{aligned}\boldsymbol{\mu}_{t+1|t+1} &= \begin{bmatrix}\boldsymbol{\mu}_{m,t+1|t+1}\\ \boldsymbol{\mu}_{p,t+1|t+1}\end{bmatrix}\\ &= \begin{bmatrix}\boldsymbol{\mu}_{m,t+1|t} + \mathbf{K}_{t+1|t}(\mathbf{z}_t - \tilde{\mathbf{z}}_t)\\ \exp((\mathbf{K}_{t+1|t}(\mathbf{z}_{t+1} - \tilde{\mathbf{z}}_{t+1}))^{\wedge})\boldsymbol{\mu}_{p,t+1|t}\end{bmatrix}\end{aligned} \tag{52}$$

$$\boldsymbol{\Sigma}_{t+1|t+1} = (\mathbf{I} - \mathbf{K}_{t+1|t}\mathbf{H}_{t+1|t})\boldsymbol{\Sigma}_{t+1|t} \tag{53}$$

where V is the observation noise covariance in Eq.(22).

4. RESULTS AND DISCUSSION

The proposed EKF based SLAM algorithm is tested with 3 data set, and all of them are collected in real driving scenarios. The hyper-parameters used in our algorithm are same for all the data set except for the number of landmarks. We want it to match the real-world scenarios that this algorithm runs online and not able to tune parameters for every unknown environment. The hyper-parameters are listed in Table.1.

The results of EKF SLAM are shown in Fig.2-4. The figures show the estimated trajectory and the 2D position of the visual features. The red line in the figure is the estimate robot trajectory, and the green dots are the position of landmarks. For each test case, we compare the result of IMU-based landmark mapping and the visual SLAM. The former case estimates the trajectory with IMU measurement and EKF prediction only and focuses on estimating the position of landmarks, while the latter simultaneously predict and update the position of landmarks and robot pose.

We calibrate the hyperparameters with the second test case. The second test case is the scenario that the car almost drives in a closed loop. The result shows that the beginning of the trajectory almost meets the end of the trajectory, which is exactly what we expected. The estimate position of the visual features also distributes in a reasonable way. On the other hand, the IMU-based landmark mapping does not provide excellent results. The reason is that the trajectory is not corrected based on the observations of visual features. The quality of the estimate position of the visual features also relies on the correct pose of the robot. Thus the landmark estimates are also not accurate.

Due to the lack of ground truth data, we are not able to quantize the performance of our proposed algorithm. For example, we can collect the ground truth of the robot trajectory with a high precision GPS. It may help us measure the performance of our SLAM algorithm and make improvements in the future.

5. REFERENCES

[1] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al., "Fastslam: A factored solution to the simultaneous localization and mapping problem," *Aaai/iaai*, vol. 593598, 2002.

[2] Michael Montemerlo, Sebastian Thrun, Daphne Koller, Ben Wegbreit, et al., "Fastslam 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges," in *IJCAI*, 2003, pp. 1151–1156.

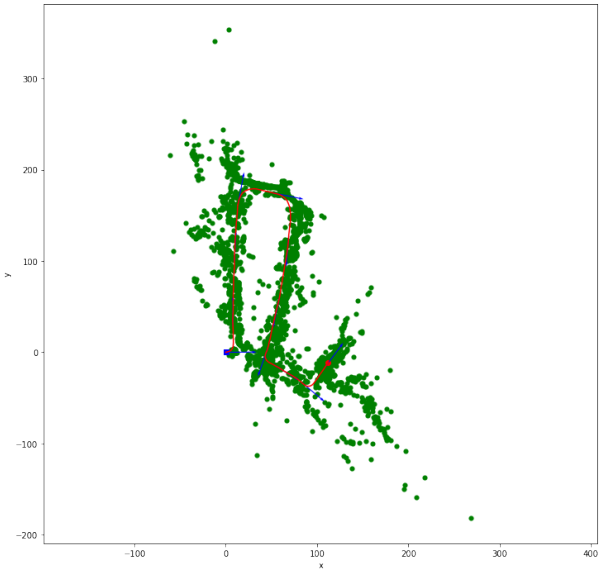
[3] Tim Bailey, Juan Nieto, Jose Guivant, Michael Stevens, and Eduardo Nebot, "Consistency of the ekf-slam algorithm," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2006, pp. 3562–3568.

[4] Keisuke Tateno, Federico Tombari, Iro Laina, and Nassir Navab, "Cnn-slam: Real-time dense monocular slam with learned depth prediction," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6243–6252.

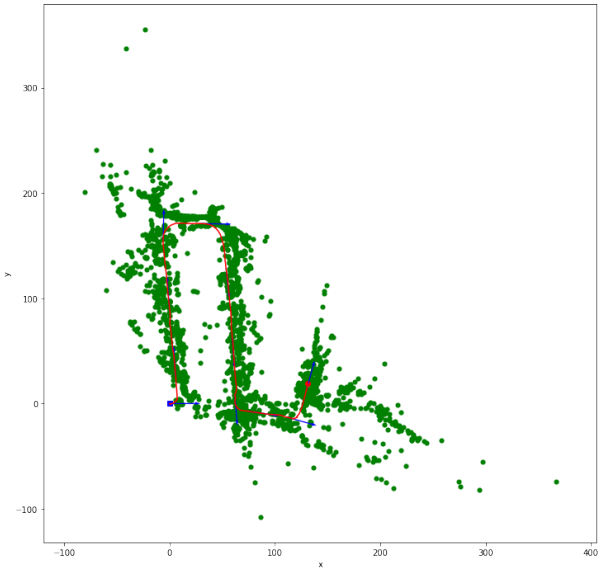
[5] Beril Sirmaccek, Nicolò Botteghi, and Mustafa Khaled, "Reinforcement learning and slam based approach for mobile robot navigation in unknown environments," in *ISPRS Workshop Indoor 3D 2019*, 2019.

Parameter	Description	Value
$\mathbf{\Sigma}_m$	Prior landmark estimate covariance	$0.005 * \mathbf{I} \in \mathbf{R}^{3\times3}$
$\mathbf{\Sigma}_p$	Prior pose estimate covariance	$0.001 * \mathbf{I} \in \mathbf{R}^{6\times6}$
$\mathbf{V}$	Observation noise covariance	$100 * \mathbf{I} \in \mathbf{R}^{4\times4}$
$\mathbf{W}_p$	Process noise covariance	$0.001 * \mathbf{I} \in \mathbf{R}^{6\times6}$

Table 1. Hyper-parameters for EKF Visual Inertial SLAM

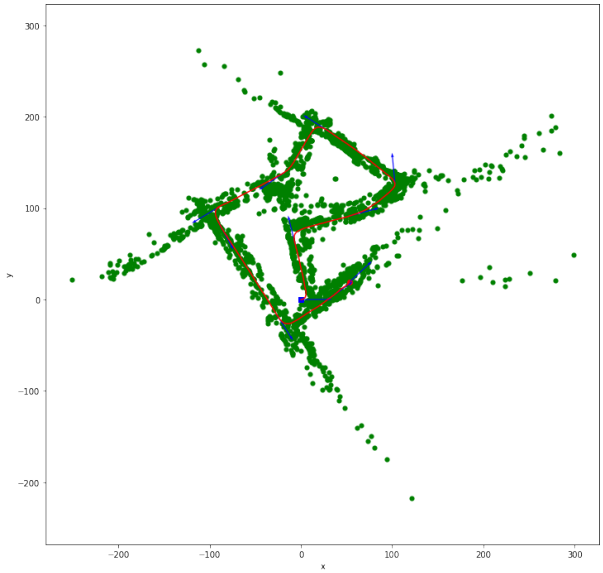


(a) IMU-based landmark mapping

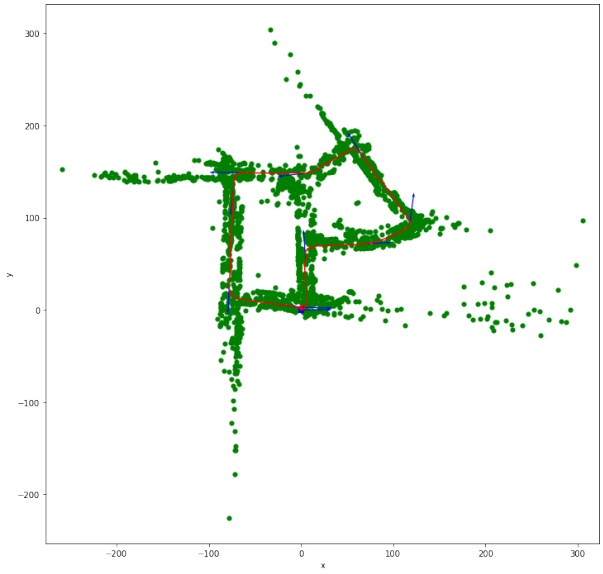


(b) EKF-SLAM

Fig. 2. Trajectory and Visual Features: TestCase 1



(a) IMU-based landmark mapping

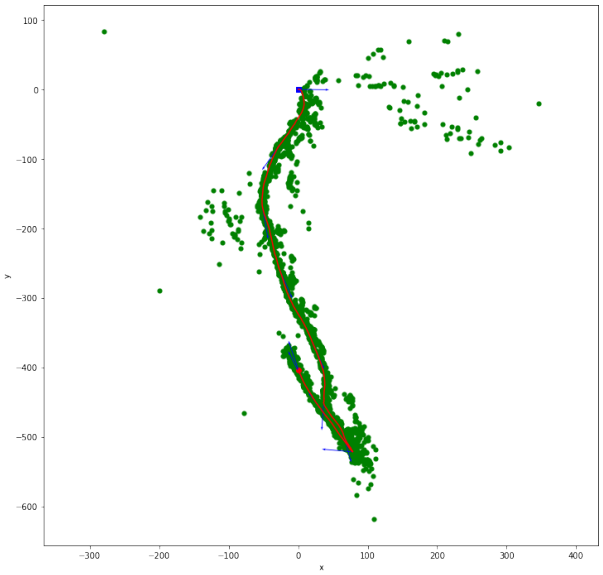


(b) EKF-SLAM

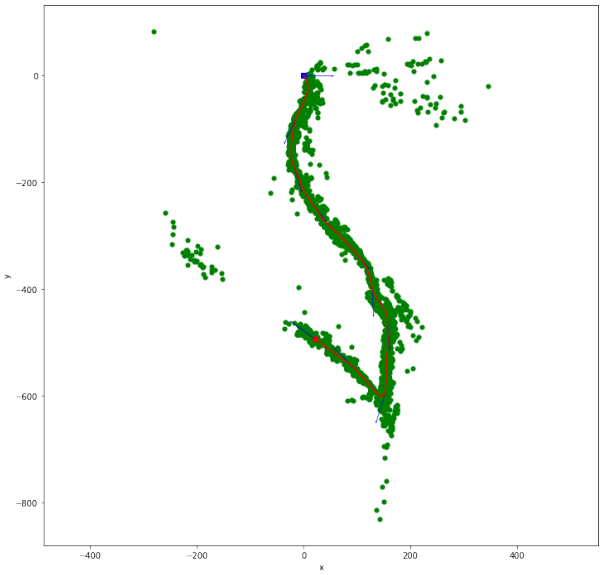
Fig. 3. Trajectory and Visual Features: TestCase 2

[6] Nikolay Atanasov, "Ece276a: Sensing estimation in robotics lecture 13: Visual-inertial slam," 2020.

[7] Timothy D Barfoot, *State estimation for robotics*, Cambridge University Press, 2017.



(a) IMU-based landmark mapping



(b) EKF-SLAM

Fig. 4. Trajectory and Visual Features: TestCase 3