Visual Inertial Simultaneous Localization and Mapping

Steve Kim
Department of Electrical Computer Engineering
University of California, San Diego
stk002@eng.ucsd.edu

***Abstract*—This paper details the process of Visual Inertial Simultaneous Localization and Mapping, aka VI SLAM, with Extended Kalman Filtering. We show how just an RGB camera and an IMU can provide rich details about a vehicle's movements as well as its surrounding environment.**

***Index Terms*—VI SLAM, Extended Kalman Filter**

I. Introduction

There are many ways for a self driving vehicle to map its location and movement over time. One might expect that in order do generate an accurate map, a vehicle would require several complicated sensor devices. But surprisingly, it can be done with just two relatively older technologies - an RGB camera and an Inertial Measurement Unit (IMU). By using the Extended Kalman Filter (EKF), we can combine the data from these two sensors to yield measurements more accurate than either provides alone. In this paper, we review how the EKF can match image and motion data in order for a vehicle to map the environment as well as keep track of its movements. In the Problem Formulation section, we present the mathematical models used to model IMU and camera data together. In the Technical Approach section, we show the key steps and equations in combining the sensor data in order to create a path and mapping of the environment. In the Results section, we present our results as well as commentary on the process in total.

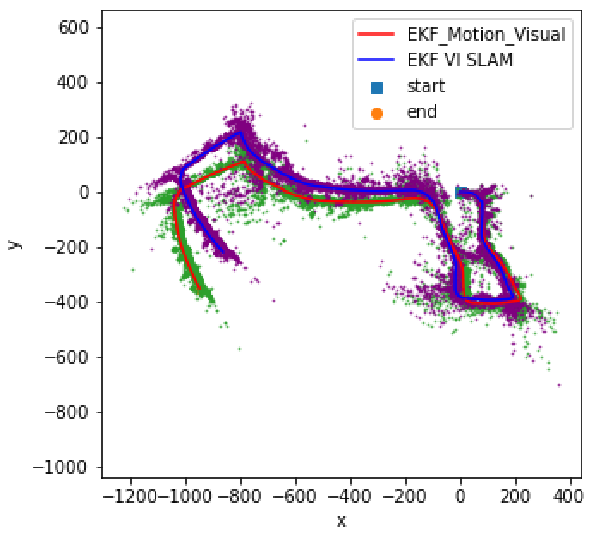
II. Problem Formulation

The purpose of SLAM is two-fold: 1. Find the motion of the vehicle and 2. Map the surrounding environment. In this section, we review how the Kalman filter can be used to conduct SLAM.

A. Extended Kalman Filter

The Kalman Filter is a linear estimation algorithm that is able to measure processes which have Gaussian noise. Since Visual Inertial Simultaneous Localization and Mapping is a non-linear problem, we make use of the Extended Kalman Filter (EKF), which is designed to deal with non-linear cases.

In the EKF, we consider each sensor input as the result of a differential equation, f and h. Let x_t represent our vehicle state, with u_t representing our control inputs(linear and angular velocity). And let z_t represent our observations from the camera, with m_j representing the features in the environment. Let w_t and v_t represent motion noise and observational noise, respectively. The subscript t represents timestep.



$$\boldsymbol{x}_t = f(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t, \boldsymbol{w}_t), w_t \sim \mathcal{N}(0, W) \tag{1}$$

$$\boldsymbol{z}_t = h(x_t, \boldsymbol{v_t}), v_t \sim \mathcal{N}(0, V) \tag{2}$$

Equation 1 represents the motion model, and equation 2 shows the the EKF observational model. The functions f and h can be used to predict the measurements of the motion model and the visual model, but in order to get their covariances, we also need to calculate the Jacobians, which we will show in the Technical Approach section. One thing to make note of is how we're treating the noise of each measurement. We can only treat noise in an additive way if the measurements are independent. This leads us to our next section, where we review the assumptions we've made in order to make these calculations.

B. Gaussian and Markov Assumptions

As powerful and convenient as the Extended Kalman Filter is, we need to make several assumptions in order to justify its use. First, we need to assume that the prior PDF $p_{t|t}$, the motion model noise w_t, and the observation model noise v_t are Gaussian. We also assume that w_t and v_t are independent of each other, as well as independent of state x_t, as well as across timesteps t. This allows us to add the covariances together in EKF. We know that in real life, these assumptions of independence are not often true. However, these assumptions notwithstanding, in the procedure of VI SLAM, we can still obtain accurate results.

$$p(\mathbf{x}_{0:T}, \mathbf{m}, \mathbf{z}_{0:T}, \mathbf{u}_{0:T-1}) = \\ p_{0|0}(\mathbf{x}_0, \mathbf{m}) \prod_{t=0}^{T} p_h(\mathbf{z}_t | \mathbf{x}_t, \mathbf{m}) \prod_{t=1}^{T} p_f(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_{t-1}) \quad (3)$$

Equation 8 shows how we are able to construct a joint PDF using the Markov assumption.

C. Rotating Stereo Camera Input

In order to combine data from the IMU with the RGB camera, we need to make sure that everything is in the same frame of reference. In this project, we have a stereo camera, which means we have a left and right camera input. We know the distance between the left and right lenses, which we call the baseline b. And because we have a stereo camera setup, we can calculate depth z of any point we see from both sensors.

$$\begin{bmatrix} u_L \\ v_L \\ u_R \\ v_R \end{bmatrix} = \underbrace{\begin{bmatrix} fs_u & 0 & c_u & 0 \\ 0 & fs_v & c_v & 0 \\ fs_u & 0 & c_u & -fs_u b \\ 0 & fs_v & c_v & 0 \end{bmatrix}}_{M} \frac{1}{z} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (4)$$

Using the above equation 6, we can calculate the depth z

$$z = \frac{f_{s_u}}{U_L - U_R} \quad (5)$$

With the above equations, we can take a point from the camera input (pixel frame) and map it to a coordinate in the real world in reference to the device itself.

D. Feature Acquisition

In our implementation of VI SLAM, we already had observation features available from our dataset. However, if we had to obtain features on our own, we would implement a corner detecting algorithm, such as finding a Harris Corner. This algorithm can be defined as shown below, where k is a small scalar (0.05), and λ_1 and λ_2 are the eigenvalues of G.

$$\lambda_1 \lambda_2 - k(\lambda_1 + \lambda_2)^2 = det(G) - ktr^2(G) \geq \rho \quad (6)$$

III. Technical Approach

With the above preparation and assumptions made, we are ready to dive into the key calculations made to create VI SLAM.

A. Prediction step

In order to calculate the position of the vehicle, we show the Extended Kalman Filter equations with further details.

The motion state prediction can be calculated as follows:

$$\boldsymbol{\mu}_{t+1|t} = f(\boldsymbol{\mu}_{t|t}, \boldsymbol{u}_t, 0) \quad (7)$$

$$\Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^T + Q_t W Q_t^T \quad (8)$$

The function f is defined as as the first order Taylor series approximation to the motion model:

$$f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{w}_t) \approx f(\mathbf{x}_t, \mathbf{u}_t, \mathbf{0}) + F(\mathbf{x}_t - \boldsymbol{\mu}_{t|t}) + Q\mathbf{w}_t \quad (9)$$

Let $\hat{u}_t$ represent the control input with linear and angular velocities. It is defined as follows:

$$\hat{\mathbf{u}}_t := \begin{bmatrix} \hat{\boldsymbol{\omega}}_t & \hat{\boldsymbol{\lambda}}_t \\ \mathbf{0} & 0 \end{bmatrix} \in \mathbb{R}^{4\times4} \quad (10)$$

And F and Q are defined below:

$$F_t = \frac{df}{dx}(\mu_{t|t}, u_t, 0), \ Q_t = \frac{df}{dw}(\mu_{t|t}, u_t, 0) \quad (11)$$

We construct our pose via nominal and perturbation kinematics of $\delta\mu_{t|t}$ with time discretization τ:

$$\boldsymbol{\mu}_{t+1|t} = exp(\tau\hat{\mathbf{u}}_t)\boldsymbol{\mu}_{t|t} \quad (12)$$

$$\delta\boldsymbol{\mu}_{t+1|t} = exp(-\tau\hat{\mathbf{u}}_t) + w_t \quad (13)$$

Which leads us to our EKF prediction formula:

$$\boldsymbol{\mu}_{t+1|t} = exp(\tau\hat{\mathbf{u}}_t)\boldsymbol{\mu}_{t|t} \quad (14)$$

$$\Sigma_{t+1|t} = exp(-\tau\overset{\curlywedge}{\mathbf{u}}_t)\Sigma_{t|t} exp(-\tau\overset{\curlywedge}{\mathbf{u}}_t)^T + W \quad (15)$$

Where $\overset{\curlywedge}{\mathbf{u}}$ is defined as

$$\overset{\curlywedge}{\mathbf{u}}_t := \begin{bmatrix} \hat{\boldsymbol{\omega}}_t & \hat{\boldsymbol{\lambda}}_t \\ 0 & \hat{\boldsymbol{\omega}}_t \end{bmatrix} \in \mathbb{R}^{6\times6} \quad (16)$$

And for the observation model, we apply the same first-order Taylor series approximation for prediction:

Here is the observational model once again

$$\boldsymbol{z}_t = h(x_t, \boldsymbol{v_t}), v_t \sim \mathcal{N}(0, V) \quad (17)$$

And its prediction equation

$$h(\mathbf{x}_{t+1}, \mathbf{v}_{t+1}) \approx \\ h(\boldsymbol{\mu}_{t+1|t}, \mathbf{0}) + H_{t+1}(\mathbf{x}_{t+1} - \boldsymbol{\mu}_{t+1|t}) + R_{t+1}\mathbf{v}_{t+1} \quad (18)$$

Where H and R are defined below:

$$H_t = \frac{dh}{dx}(\mu_{t|t-1}, 0), \ R_t = \frac{dh}{dv}(\mu_{t|t-1}, 0) \quad (19)$$

B. Update Step

Now that we have our predictions, we can move forward a timestep and update our measurements for our mean pose.

$$\boldsymbol{\mu}_{t+1|t+1} = \mu_{t+1|t} + K_{t+1|t}(z_{t+1} - h(\mu_{t+1|t}, 0)) \tag{20}$$

$$\Sigma_{t+1|t+1} = (I - K_{t+1|t}H_{t+1})\Sigma_{t+1|t} \tag{21}$$

We define K as our Kalman Gain factor as follows:

$$K_{t+1|t} := \Sigma_{t+1}H_{t+1}^T(H_{t+1}\Sigma_{t+1}H_{t+1})^T + R_{t+1}VR_{t+1}^T)^{-1} \tag{22}$$

In the process of driving, the vehicle can see landmarks multiple times over different frames, and when it does, it will provide an update to the landmark coordinates.

Let $\tilde{z}_{t+1,i}$ be the predicted observation

$$\tilde{\mathbf{z}}_{t+1} = M\pi({}_oT_i\mu_{t+1}^{-1}\boldsymbol{m}_j) \tag{23}$$

In the above equation, $\boldsymbol{m}_j$ is the homogenous coordinates equivalent to $[m_j 1]^T$. M is the stereo calibration matrix defined in equation 4. And π is the projection function defined as:

$$\pi(q) = \frac{1}{q_3}q \tag{24}$$

And its derivative is:

$$\frac{d\pi}{d\mathbf{q}} = \frac{1}{q_3}\begin{bmatrix} 1 & 0 & -\frac{q_1}{q_3} & 0 \\ 0 & 1 & -\frac{q_2}{q_3} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{q_4}{q_3} & 1 \end{bmatrix} \tag{25}$$

Let the Jacobian of $\tilde{z}_{t+1,i}$ with respect to T_{t+1} evaluated at $\mu_{t+1|t}$ be equal to:

$$H_{i,t+1|t} = -M\frac{d\pi}{d\mathbf{q}}({}_oT_i\boldsymbol{\mu}_{t+1|t}\underline{\mathbf{m}}_j){}_oT_i(\boldsymbol{\mu}_{t+1|t}\underline{\mathbf{m}}_j)^{\odot} \tag{26}$$

Then the update for the landmark observations will be:

$$K_{t+1|t} := \Sigma_{t+1|t}H_{t+1|t}^T(H_{t+1|t}\Sigma_{t+1|t}H_{t+1|t}^T + I \otimes V)^{-1} \tag{27}$$

$$\mu_{t+1|t+1} = \mu_{t+1|t}exp((K_{t+1}(z_{t+1} - \tilde{z}_{t+1}))\hat{)} \tag{28}$$

$$\Sigma_{t+1|t+1} = (I - K_{t+1}H_{t+1})\Sigma_{t+1|t} \tag{29}$$

III. Dead Reckoning

If we were to show what the landmarks and path looks like purely with data coming from the IMU, it would look like figure 2 above.

The road the vehicle drove on did not vary greatly in the z direction (no major hills or dips), and the IMU provided good data; as a result, the dead reckoning mapping shows approximately correct results.

IV. Results

We can see that at the start, the motion model and the VI SLAM model correspond quite closely. This makes sense because the main issue with IMU-only mapping is that errors grow over time. We can see this occur as the vehicle continues to drive - the motion model and the VI SLAM model drift apart more and more.

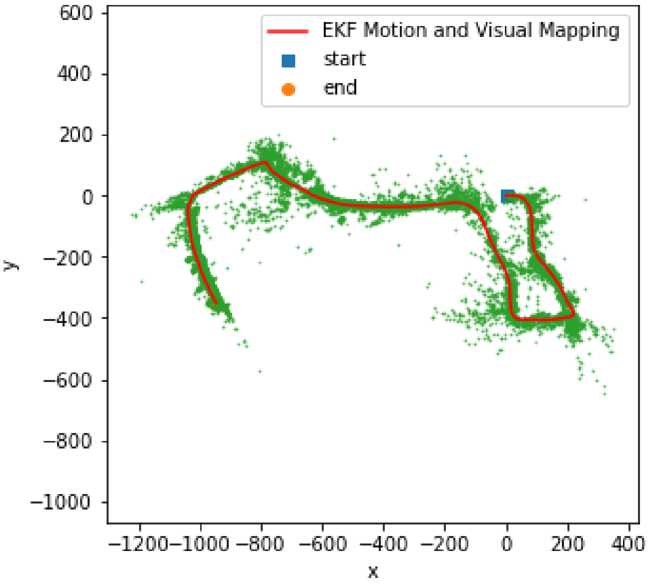


Fig. 2. Mapping path and landmarks with just the motion model

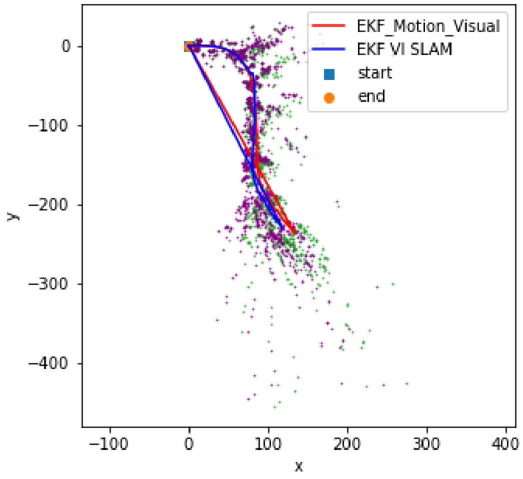


Fig. 3. Motion landmarks are green. VI SLAM landmarks are purple.

V. Collaboration

Collaborated with Hala Abualsaud and Luis Martin Herrera Lezama

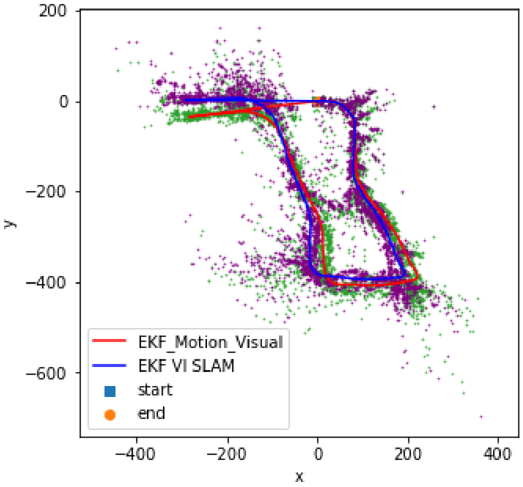


Fig. 4. The motion model and the VI SLAM model are separated after several vehicle turns

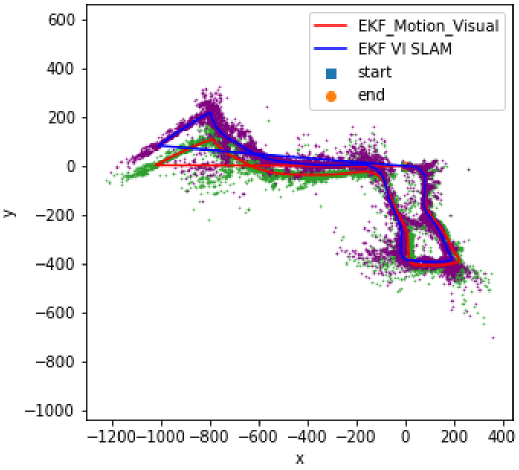


Fig. 5. The difference between the two models stays relatively stable, mostly due to good data from the IMU

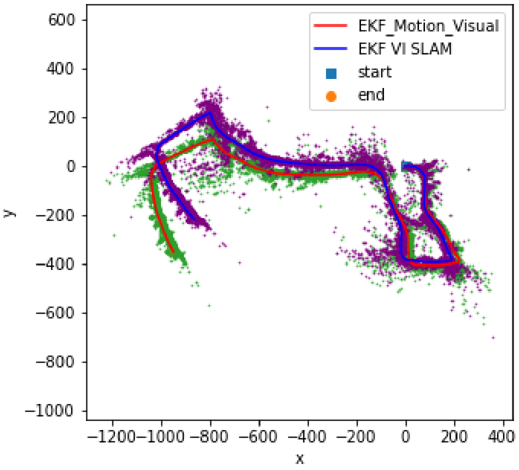


Fig. 6. We see that by the end, the motion model and VI SLAM model have separated more than before. We can expect that if there were more data, we would see the two paths continue to drift apart.