Visual Intertial Simultaneous Localization and Mapping

Srinidhi Kalgundi Srinivas
Department of Electrical and Computer Engineering
University of California San Diego
skalgundisrinivas@ucsd.edu

***Abstract*—This report presents the mathematical formulation of Extended Kalman Filters and their usage in Visual Inertial Simulataneous Localization and Mapping (VI-SLAM). Results of mapping and localizing an autonomus vehicle using data from Inertial Measurement Unit (IMU) and camera is presented**

***Index Terms*—Bayes Filter, SLAM, Prediction, Update, Mapping, Autonomous vehicle, IMU, Stereo Camera**

I. Introduction

Humans use a mental map to get around places. We rely on GPS to follow a global plan and intuition and experience to navigate locally where we cannot rely on GPS. The same is not true for robots, where the robots cannot accurately rely on GPS for positioning. Plethora of other sensors are required by the robots to map the environment and localize itself in the environment. If a robot, such as an autonomous car is given a prior map, localization is only a matter of observing the surrounds and estimating the current pose of the vehicle. In reality, such is not the case. Robots more often than not will have to localize and map simultaneously. This process of simultaneous localization and mapping is commonly referred to as *Simultaneous Localization and Mapping or SLAM.*

Using SLAM, robots build their own map as they move around in an unknown environment. SLAM is a computational problem in that, the robot or the computer tracks the robots' position and maps the environment. It is generally a chicken and egg problem. There has been a lot of research into building computationally efficient and accurate SLAM algorithms over the past few decades. Some of the noteworthy algorithms are, Fast SLAM [1], Rao-Blackwellized Particle Filter SLAM [3], Kalman Filter based SLAM [2], and the simplest yet powerful method based on Bayes Filter [4], Particle Filter SLAM. These SLAM algorithms differ from one another in map representations, use of different probability density functions for representing agent's pose etc. In this report, mathematical formulation, technical approach to solving the problem and results are presented. IMU odometry data combined with observations from stereo camera are to perform EKF SLAM.

II. Problem Formulation

A. SLAM

SLAM at the outset is a parameter estimation problem for $\mathbf{x}_{0:T}$ which is the state or pose of the robot and map $\mathbf{m}$, given a dataset of robot inputs $\mathbf{u}_{0:T-1}$ and observations $\mathbf{z}_{0:T}$.

The joint distribution of the pose or state of the robot, map of the environment, series of observations and control inputs can be decomposed under Markov Assumptions. The joint distribution $p(x_{0:T}, m, u_{0:T-1}, z_{0:T})$ can be written under Markov assumption as:

$$p_0(x_0, m) \prod_{t=0}^{T} p_h(z_t|x_t, m) \prod_{t=1}^{T} p_f(x_t|x_{t-1}, u_{t-1}) \prod_{t=0}^{T-1} p(u_t|x_t) \tag{1}$$

where, p_h is the observation model, p_f is the motion model, p_0 is the prior state of the robot and $p(u_t|x_t)$ is the control policy. In this problem, we assume that the control policy is one.

B. Kalman Filter

[4] Bayes Filter is a probabilistic technique for SLAM problem that combines evidence from control inputs and observations made by the robot. Bayes Filter uses Markov assumptions to predict and update states of the robot. It keeps track of two probability distribution functions as mentioned below:

$$\text{Updated PDF: } p_{t|t}(x_t) := p(x_t, z_{0:t}, u_{0:t-1})$$
$$\text{Predicted PDF: } p_{t+1|t}(x_{t+1}) := p(x_{t+1}, z_{0:t}, u_{0:t})$$

where *t* represent time.

Prediction Step: Bayes filter uses the motion model p_f and prior pdf $p_{t|t}$ of the robot is to compute the pdf of the state of the robot at time *t+1*

$$p_{t+1|t}(x) = \int p_f(x|s, u_t) p_{t|t}(s) ds$$

Update Step: Bayes filter uses the observation model p_h and predicted pdf $p_{t+1|t}$ to obtain $p_{t+1|t+1}$

$$p_{t+1|t+1}(x) = \frac{p_h(z_{t+1}|x) p_{t+1|t}(x)}{\int p_h(z_{t+1}|s) p_{t+1|t}(s) ds}$$

C. Kalman Filter

Kalman Filter based SLAM assumes that the prior PDF $p_{t|t}$ is Gaussian. It also makes an assumption that the motion and the observation models are linear in the state x_t with motion model noise w_t and observation model noise v_t. w_t and v_t are assumed to be independent of each other and of the state x_t.

Motion Model: Motion model for Kalman Filter is as shown below;

$$x_{t+1} = f(x_t, u_t, w_t) := Fx_t + G_u t + w_t, w \sim N(0, W)$$

$$x_{t+1}|x_t, u_t \sim N(Fx_t + Gu_t, W)$$

where F $\in$R$^{d_x x d_x}$, G $\in$R$^{d_x x d_u}$, W $\in$R$^{d_x x d_x}$

Observation Model: Observation model for Kalman Filter is as shown below:

$$z_t = h(x_t, v_t) := Hx_t + v_t, v \sim N(0, V)$$

$$z_t|x_t \sim N(Hx_t, V)$$

$$x_{t+1}|x_t, u_t \sim N(Fx_t + Gu_t, W)$$

where H $\in$R$^{d_z x d_x}$,V $\in$R$^{d_z x d_z}$

One of the major drawbacks of the Kalman Filter is that it assumes that the motion and observation models are linear which are not true in majority of the practical scenarios. Kalman Filter involved calculating values of 5 integrals, 2 in prediction and 3 in the update step. These integrals are approximated using various methods such as Extended Kalman Filter and Unscented Kalman Filter.

D. Extended Kalman Filter

Extended Kalman Filter(EKF) approximates non-linear motion and observation models using Taylor series around their noise means and use these approximations to calculate the prediction and approaximate integrals.

Motion model for EKF is given by;

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

Observation model for EKF is given by;

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

Prediction: Prediction of $\mu_{t+1|t}$ and $\sum_{t+1|t}$ is given by the following equations:

$$\mu_{t+1|t} = f(\mu_{t|t}, u_t, 0)$$

$$\Sigma_{t+1|t} = F_t \Sigma_{t|t} F_t^T + Q_t W Q_t^t$$

Update: Updating $\mu t + 1|t$ and $\Sigma_{t+1|t}$ is performed by the following set of equations:

$$\mu_{t+1|t+1} = \mu t + 1|t + K_{t+1}(z_t - h(\mu_{t+1|t}, 0))$$

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

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

Visual Inertial SLAM is explained with equations in Technical Approach as the line between problem formulation for VI SLAM and technical approach is thin.

In our problem, motion model is obtained from the IMU and observation model is obtained from the stereo camera model.

III. Technical Approach

To avoid computational burden, only a subset of the features were used to run VI-SLAM. Of the given 5105 features in 03.npz data, 200 features were used without the loss of accuracy. As the number of features increased, the covariance matrix grew exponentially resulting in memory issues.

A. Localization by Visual Inertial Odometry

Data provided contains linear and angular velocities measured by the IMU mounted on the vehicle. To localize the car, pose of the IMU, i.e., transformation from IMU frame to world frame T_t is calculated under the assumption that the world frame landmark coordinates and the data association between the landmark and its observation is known.

Data: u_t = $[v_t^T, \omega_t^T]^T$ $\in$R^6 and observations $z_{0:t}$ is given.

Motion Model: Motion model of pose **T** $\in SE(3) with perturbation \delta(\mu_{t+1|t})$ and nominal kinematics $\mu_{t+1|t}$ is given by the following set of equations:

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

$$\delta(\mu_{t+1|t}) = exp(-\tau_t \hat{\hat{u}}_t)\delta_{t|t} + w_t$$

where,

$$u_t = \begin{pmatrix} v_t \\ \omega_t \end{pmatrix} \in \mathbb{R}^6, \hat{u}_t = \begin{pmatrix} \hat{\omega}_t & v_t \\ \mathbf{0}^T & 0 \end{pmatrix} \in \mathbb{R}^{\text{4x4}}, \tag{2}$$

$$\hat{\hat{u}}^t = \begin{pmatrix} \hat{\omega}_t & \hat{v}_t \\ 0 & \hat{\omega}_t \end{pmatrix} \in \mathbb{R}^{\text{6x6}} \tag{3}$$

For the project, ω_t and v_t were taken for every successive step.

Prediction step during localization was performed based on the following equations:

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

$$\Sigma_{t+1|t} = exp(-\tau_t \hat{\hat{u}}_t)\Sigma_{t|t} exp(-\tau_t \hat{\hat{u}}_t)^T + W$$

where $\mu_{t|t} \in \mathbb{R}^3$, $\Sigma_{t|t} \in \mathbb{R}^{6 \text{ x } 6}$ and W is the process pose noise $\in \mathbb{R}^{6 \text{ x } 6}$. It was observed that the EKF update step for the Visual Inertial SLAM was very sensitive to W and the value of W was arrived at empirically by running multiple experiments with different values whose results are discussed in the Results section.

B. Landmark Mapping

The data provided is from stereo camera model where the features z_t for every time step *t* is in $\mathbb{R}^{4xM}$, where M is the total number of landmarks. The landmarks that are not observed at time *t* contain the values [-1, -1, -1, -1].

At every timestep *t*, landmarks that are being observed at that instant was calculated and compared with previously seen landmarks. If the landmarks were not previous seen, those landmarks were marked and initialized to the observed values converted to world frame. If the landmarks were observed before, the EKF update step was performed for the previously observed landmark based on the observation model and update step as shown below.

Observation Model: The stereo camera observation model is as follows:

$$z_{t,i} := K_s\pi({}_oT_IT_t^{-1}m_j) + v_{t,i}$$

where K_s is the intrinsic matrix, i represents the i^{th} pixel coordinates, ${}_oT_I$ is the transformation to camera coordinates and T_t is the pose of the IMU.

Visual Mapping does not involve predicting new values and only the update step of the EKF is performed. Update step for visual mapping is as follows:

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

$$\mu_{t+1|t+1} = \mu_{t+1|t} + K_{t+1}(z_t - \tilde{z}_t)$$

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

where,

$$\mu_{t+1|t+1} \in \mathbb{R}^{3M}, \Sigma_{t+1|t+1} \in \mathbb{R}^{\text{3M x 3M}}, H_{t+1} \in \mathbb{R}^{\text{4N x 3M}}$$

where, N is the number of features being updated.

$$\tilde{z}_t = K_s\pi({}_oT_IT_t^{-1}\mu_j)$$

and

$$H_{t+1,i,j} = K_s\frac{d\pi}{dq}({}_oT_IT_t^{-1}\mu_j)_oT_IT_t^{-1}P^T, P = [I, 0]$$

$$\frac{d\pi}{dq}(q) = \frac{1}{q3}\begin{pmatrix} 1 & 0 & -\frac{q1}{q3} & 0 \\ 0 & 1 & -\frac{q2}{q3} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{q3}{q4} & 1 \end{pmatrix} \tag{4}$$

Observation noise V was initialized as an identity matrix multiplied with a value 100. Choice of the number 100 was arrived at empirically and it was observed that very small values cause the Kalman gain calculation to fail because of singularity.

The world frame coordinated of the features that are observed for the first time are by the following set of linear equations:

$$\begin{pmatrix} u_L \\ v_L \\ d \end{pmatrix} = \begin{pmatrix} fs_u & 0 & c_u & 0 \\ 0 & fs_v & c_v & 0 \\ 0 & 0 & 1 & fs_ub \end{pmatrix}\frac{1}{Z}\begin{pmatrix} X_o \\ Y_o \\ Z_o \\ 1 \end{pmatrix} \tag{5}$$

where X_o, Y_o, Z_o are the optical frame co ordinates of the pixel indices u_L, v_L, c_u, c_v are the values by which the origin is translated and f is the focal length of the camera.

Corresponding world co ordinates are calculated from optical co ordinates using the equation:

$$\begin{pmatrix} X_o \\ Y_o \\ Z_o \end{pmatrix} =_o R_rR^T(m - p) \tag{6}$$

where, ${}_oR_r$ is the optical translation from regular frame to the optical frame, *R* is the rotation matrix from camera to the world frame, *m* is a vector in $\mathbb{R}^3$ representing co ordinates in the real world and *p* is the translation of the camera sensor in the world frame.

C. *Visual Inertial SLAM*

To complete the VI SLAM pipeline, EKF update step based on the combined covariances of landmark observations and pose covariances was implemented. The combined covariance $\Sigma \in \mathbb{R}^{\text{3M+6 x 3M+6}}$ where the diagonal blocks are individual variances and off diagonal elements capture the covariance between landmarks and estimated poses.

Prediction step of VI SLAM is utilizes the same equations mentioned in Section 3A, whereas the update step equations are modified to account for the covariances. Update equations used to implement VI-SLAM are as below;

$$\tilde{z}_t = K_s\pi({}_oT_I\mu_{t+1|t}^{-1}m_j)$$

$$H_{t+1,i,j} = -K_s\frac{d\pi}{dq}({}_oT_I\mu_{t+1|t}^{-1}m_j)_oT_I(\mu_{t+1|t}^{-1}m_j)^{\odot}$$

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

$$\mu_{t+1|t+1} = \mu_{t+1|t}exp((K_{t+1}(z_t - \tilde{z}_t)\hat{)})$$

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

where,

$$\mu_{t+1|t} \in SE(3), \Sigma_{t+1|t} \in \mathbb{R}^{\text{6 x 6}}, H_{t+1} \in \mathbb{R}^{\text{4N x 6}}$$

where, N is the number of features being updated.

These equations are combined with update equations mentioned in Section 3B were implemented.

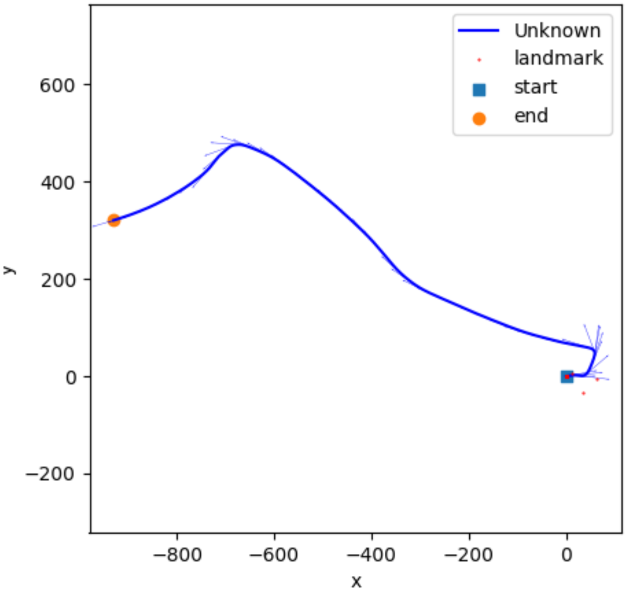
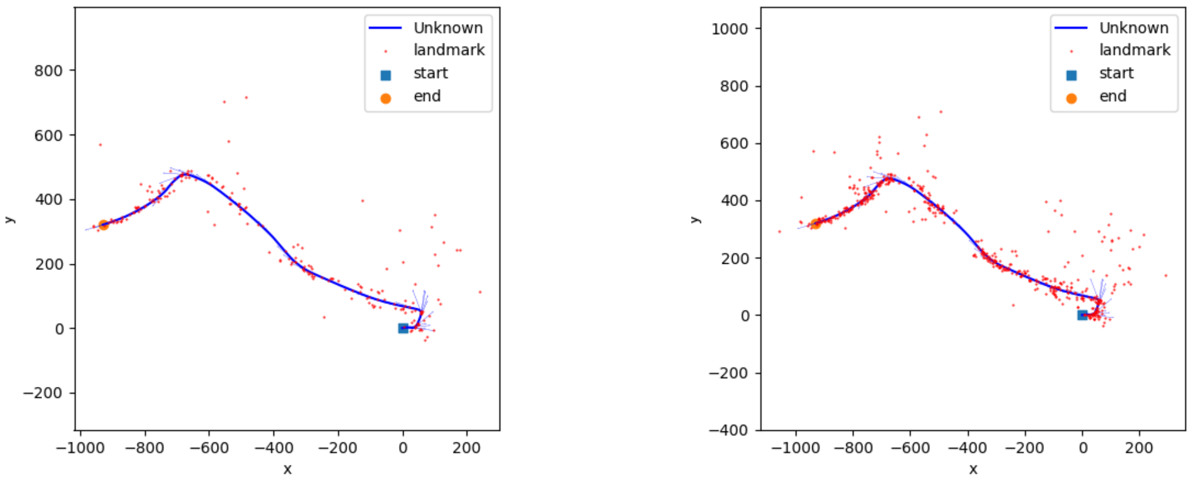


Fig. 1: Trajectory generated by localization



(a) EKF update with 200 features

(b) EKF update with 500 features

Fig. 2: Comparison of EKF update step for landmark mapping using different number of features

IV. Results

A. IMU Localization via EKF prediction

Figure 1 shows the trajectory of the car generated by just the prediction step using linear and angular velocity values provided by the IMU. The predicted path is not dependent on any of the noise values.

B. Landmark Mapping Via EKF Update

Figure 2a shows the landmarks updated using EKF equations when 200 features out of 5105 features were used. Comparision between landmarks updated based on the number of features used is shown in Figure 2. It can be seen in the figure that better results can be obtained when more features are used at the cost of computational efficiency.

C. Visual Inertial SLAM

Figure 3 shows the output of the running full VI SLAM algorithm on 03.npz dataset that captures the relationship between map updates and pose updates. The algorithm is run for 200 features.

Results of running the same algorithm on 10.npz dataset is as showin in 4b. Comparison between trajectory and landmarks for data in 10.npz with and without SLAM is also shown.

The results shown are validated by comparing the trajectory with the videos provided.

During the experiments, it was noted that the output of VI SLAM was very sensitive to the IMU pose noise W. Results for different noise values can be seen in Figure 5. For majority of different values of the noise levels, the trajectory looks like what it needs to be but the orientation changes significantly.

V. Acknowledgements

To all the instructors and students who asked and answered questions on Piazza.

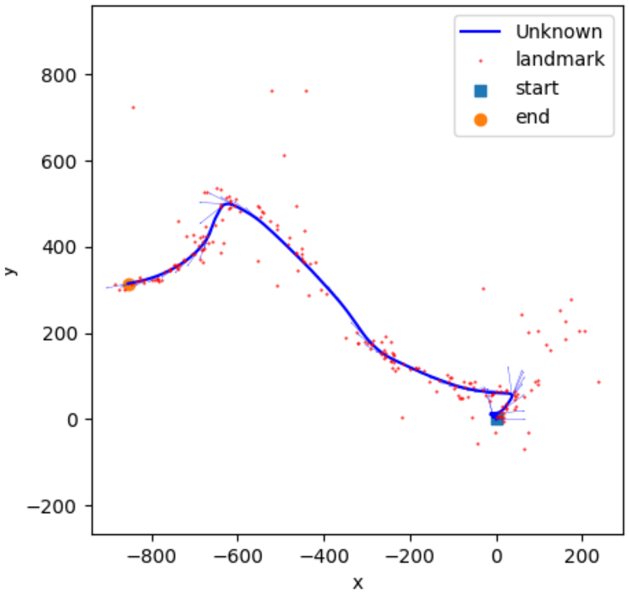


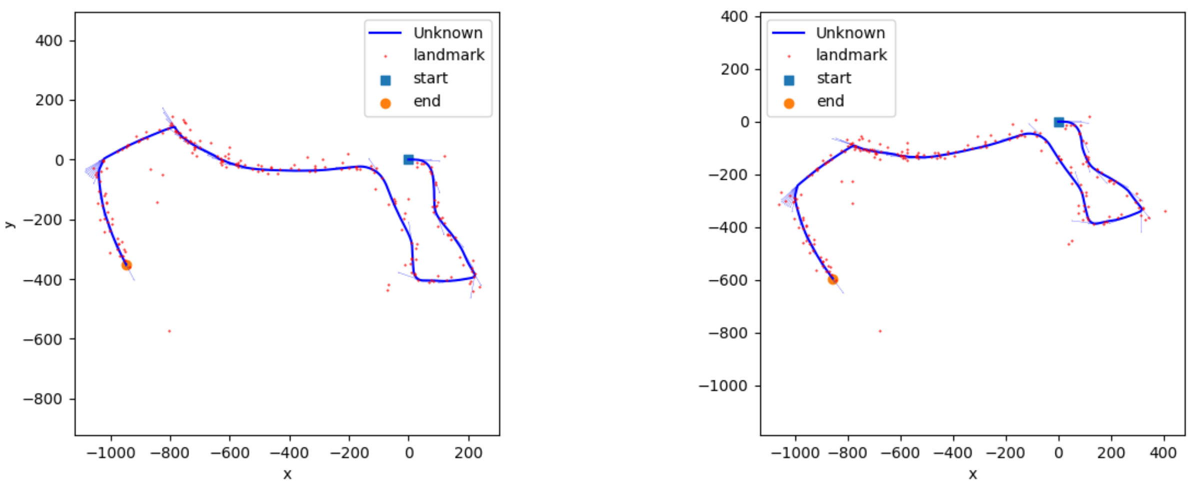
Fig. 3: Output of VI SLAM for 03.npz dataset

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] 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.

[3] Murphy K., Russell S. (2001) Rao-Blackwellised Particle Filtering for Dynamic Bayesian Networks. In: Doucet A., de Freitas N., Gordon N. (eds) Sequential Monte Carlo Methods in Practice. Statistics



(a) EKF update with 200 features for 10.npz

(b) Result of VI-SLAM for 10.npz dataset

Fig. 4: Comparison of results for 10.npz dataset with EKF update and full SLAM

for Engineering and Information Science. Springer, New York, NY. https://doi.org/10.1007/978-1-4757-3437-9_24

[4] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. 2005. Probabilistic Robotics (Intelligent Robotics and Autonomous Agents). The MIT Press.

Please refer to the next page for comparison Figure 5 - VI-SLAM outputs based on different noise values.

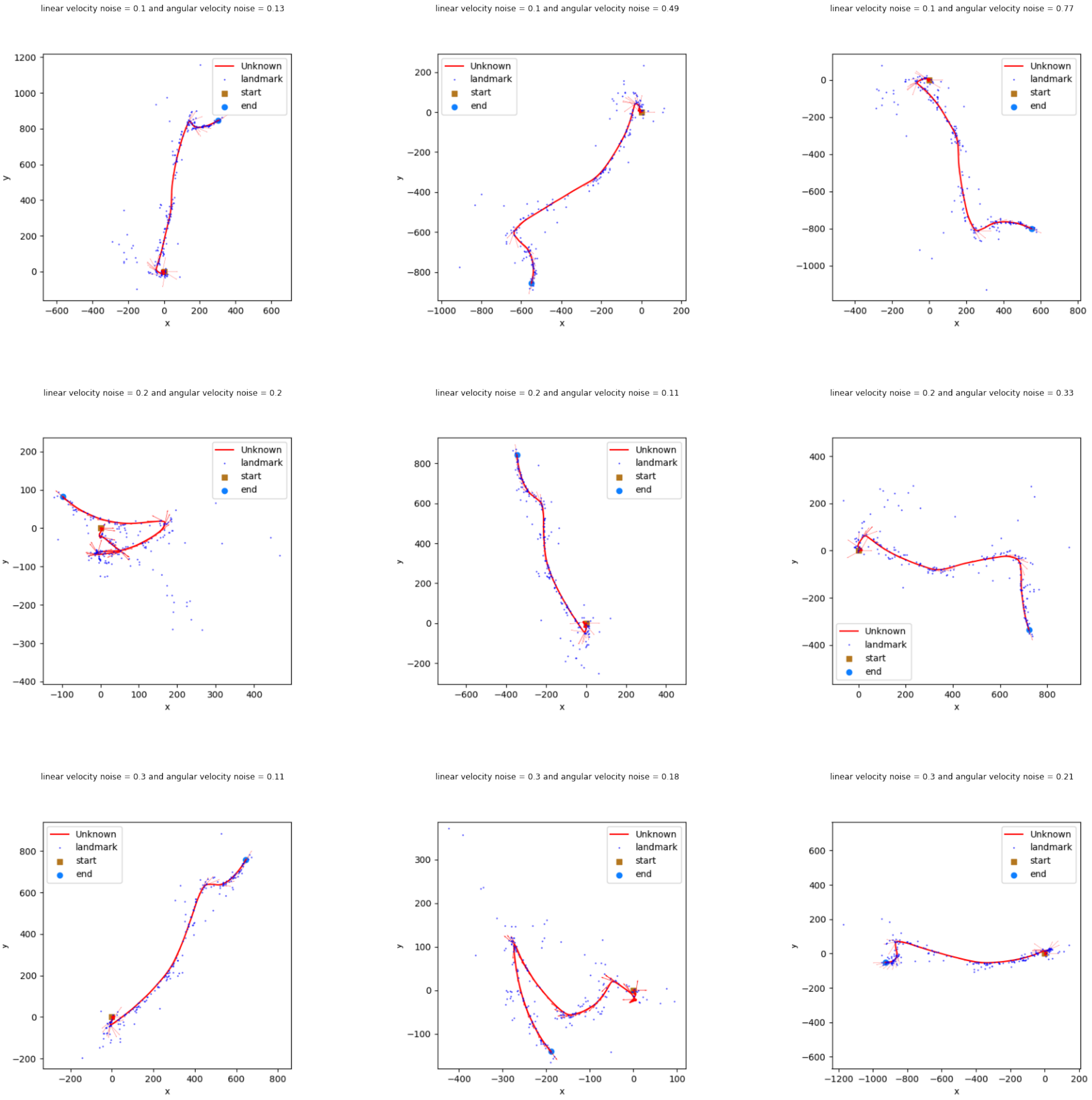


Fig. 5: Output of VI-SLAM affected by pose noise W