Unscented Kalman Filter

We are implementing the Unscented Kalman Filter on the data we are getting from IMU and processed data from the camera on the quadrotor. The data from the camera on the quadrotor is used as measurement data in the Unscented Kalman Filter.

We are using IMU data - angular velocities and linear acceleration for the prediction step in this filter.

There are two steps in this filter Prediction and Update and each part of the project it is implemented in a different way.

In both parts of the project, the prediction step has non-Additive noise

Part 1 – In Part 1 of the project, we are using the pose of the robot (position and angle of the robot with respect to the world frame) as a measurement in the update step.

First, we have to define $x_{aug} = \begin{pmatrix} x \\ q \end{pmatrix}$ with mean $\mu_{aug} = \begin{pmatrix} \mu \\ 0 \end{pmatrix}$ and covariance $\sum_{aug} = \begin{pmatrix} \Sigma & 0 \\ 0 & Q \end{pmatrix}$

Step 1: Compute the Sigma Points

We have to choose a set of 2n’ + 1 sigma points using the i^{th} column of the Cholesky Decomposition of $\sum_{aug}$.

$$\mathcal{X}^{(0)} = \boldsymbol{\mu}_{aug}, \qquad \mathcal{X}^{(i)} = \boldsymbol{\mu}_{aug} \pm \sqrt{n' + \lambda'} \left[\sqrt{\boldsymbol{\Sigma}_{aug}}\right]_i \qquad i = 1, \ldots n'$$

$$\lambda' = \alpha^2 (n' + k) - n' \quad \alpha \text{ and } k \text{ determine sigma points spread}$$

Here α = 0.001, n’ = n+n_q, n = 15 and n_q = 12, k = 1

Step 2: Propagate the Sigma points

We have to create a process model a nonlinear function which will be used to propagate the sigma points to predict the new points.

$$\chi_t^{(i)} = f(\chi_{aug,t-1}^{(i),x}, u_t, \chi_{aug,t-1}^{(i),n}) \qquad i = 0, \ldots 2n'$$

Here Function “f”

$$\dot{\mathbf{x}} = \begin{bmatrix} \mathbf{x}_3 \\ G(\mathbf{x}_2)^{-1}(\omega_m - \mathbf{x}_4 - \mathbf{n}_g) \\ \mathbf{g} + R(\mathbf{x}_2)(\mathbf{a}_m - \mathbf{x}_5 - \mathbf{n}_a) \\ \mathbf{n}_{bg} \\ \mathbf{n}_{ba} \end{bmatrix} = f(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{n}) \longrightarrow \boldsymbol{x}_t = f(\boldsymbol{x}_{t-1}, \boldsymbol{u}_t, \boldsymbol{n}_t)$$

We have to add the results from function to the respective sigma points to calculate the predicted mean.

Step 3: Computing predicted mean and covariance matrix.

$$\bar{\boldsymbol{\mu}}_t = \sum_{i=0}^{2n'} W_i^{(m)} \chi_t^{(i)} \qquad \bar{\boldsymbol{\Sigma}}_t = \sum_{i=0}^{2n'} W_i^{(c)\prime} \left(\chi_t^{(i)} - \bar{\boldsymbol{\mu}}_t\right)\left(\chi_t^{(i)} - \bar{\boldsymbol{\mu}}_t\right)^T$$

$$W_0^{(m)\prime} = \frac{\lambda'}{n' + \lambda'} \qquad W_i^{(m)\prime} = \frac{1}{2(n' + \lambda')}. \qquad i = 1, \dots 2n'$$

$$W_0^{(c)\prime} = \frac{\lambda'}{n' + \lambda'} + (1 - \alpha^2 + \beta) \quad W_i^{(c)\prime} = \frac{1}{2(n' + \lambda')}$$

So, by using above methods we can calculate the predicated mean and covariance which can be used for Update Step.

Update Step

$$\boldsymbol{z_t} = \boldsymbol{C x} + \boldsymbol{\eta} \qquad \boldsymbol{\eta} \sim N(\boldsymbol{0}, \boldsymbol{R})$$

$$\boldsymbol{\mu_t} = \boldsymbol{\bar{\mu}_t} + \boldsymbol{K_t}(\boldsymbol{z_t} - C\ \boldsymbol{\bar{\mu}_t})$$

$$\boldsymbol{\Sigma_t} = \boldsymbol{\bar{\Sigma}_t} - \boldsymbol{K_t}\ \boldsymbol{C}\ \boldsymbol{\bar{\Sigma}_t}$$

$$\boldsymbol{K_t} = \boldsymbol{\bar{\Sigma}_t}\ \boldsymbol{C^T}\ (\boldsymbol{C}\ \boldsymbol{\bar{\Sigma}_t}\ \boldsymbol{C^T} + \boldsymbol{R})^{-1}$$

We have to use the above equations to perform the Update step. Here z_t measurements are coming from the pose data coming from the camera mounted on the quadrotor from Project 2.

Part 2 – In Part 2, we are using optical flow velocity (linear velocity in the camera frame) data as z_t in the measurement update. We are also assuming the angular velocity coming from the z_t measurement and the predicted angular velocity is the same. So, the update step is using the linear velocity of the camera in the world in the camera frame as a measurement update.

In Part 2 the Prediction step is as similar to the prediction step in part 1. i.e. Steps 1 to 3 in part 1 is as the same as in Part 2.

There are the following steps in the Update step. We are doing this step according to the additive noise.

Update Step

$$\boldsymbol{z_t} = g(\boldsymbol{x_t}) + \boldsymbol{r_t} \qquad \boldsymbol{r_t} \sim N(\boldsymbol{0}, \boldsymbol{R_t})$$

Here
```
Rot = [cosd(45) -sind(45) 0 % Rotation matrix representing
      -sind(45) -cosd(45) 0 the body in Camera frame.
       0 0 -1];

Trans = [-0.04
         0         % Translation vector from Camera to Body Frame
        -0.03];
```

Here $g(x_t)$ = $\boldsymbol{R_B^C l(x_{2,}, x_3)} - \boldsymbol{R_B^C S(r_{BC}^B)}\ \ \boldsymbol{R_C^B}\ {}^{C}\omega_C^W$

x_2 = $[\phi, \theta, \psi]^T = [\text{roll}, \text{pitch}, \text{yaw}]^T$

$l(x_2,x_3)$ = Rot × transpose($\begin{bmatrix} c\psi c\theta & c\psi s\theta s\phi - c\phi s\psi & s\phi s\phi s\psi + c\phi s\psi s\theta \\ c\theta s\psi & c\phi c\psi + s\theta s\phi s\psi & c\phi s\theta s\psi - c\psi c\phi \\ -s\theta & c\theta s\phi & c\phi c\theta \end{bmatrix}$) × linear velocity(x_3)

Step 1: Compute Sigma Points

Let the dimensionalities of $\boldsymbol{x}$ be n and let $\boldsymbol{\mu}$ be the expected value of the state

$$\chi_t^{(0)} = \bar{\boldsymbol{\mu}}_t \qquad \chi_t^{(i)} = \bar{\boldsymbol{\mu}}_t \pm \sqrt{n+\lambda}\left[\sqrt{\bar{\boldsymbol{\Sigma}}_t}\right]_i \qquad i = 1, \dots n$$

Step 2: Propagate Sigma Points through the nonlinear measurement model function g

$$Z_t^{(i)} = g\left(\chi_t^{(i)}\right) \qquad i = 0, \dots 2n$$

Step 3: Compute the mean and covariance matrices

$$\boldsymbol{z}_{\mu,t} = \sum_{i=0}^{2n} W_i^{(m)} Z_t^{(i)}$$ Update

$$\boldsymbol{C}_t = \sum_{i=0}^{2n} W_i^{(c)} \left(\chi_t^{(i),} - \bar{\boldsymbol{\mu}}_t\right)\left(Z_t^{(i)} - \boldsymbol{z}_{\mu k}\right)^T \qquad \boldsymbol{S}_t = \sum_{i=0}^{2n} W_i^{(c)} \left(Z_t^{(i)} - \boldsymbol{z}_{\mu k}\right)\left(Z_t^{(i)} - \boldsymbol{z}_{\mu,t}\right)^T + \boldsymbol{R}_t$$

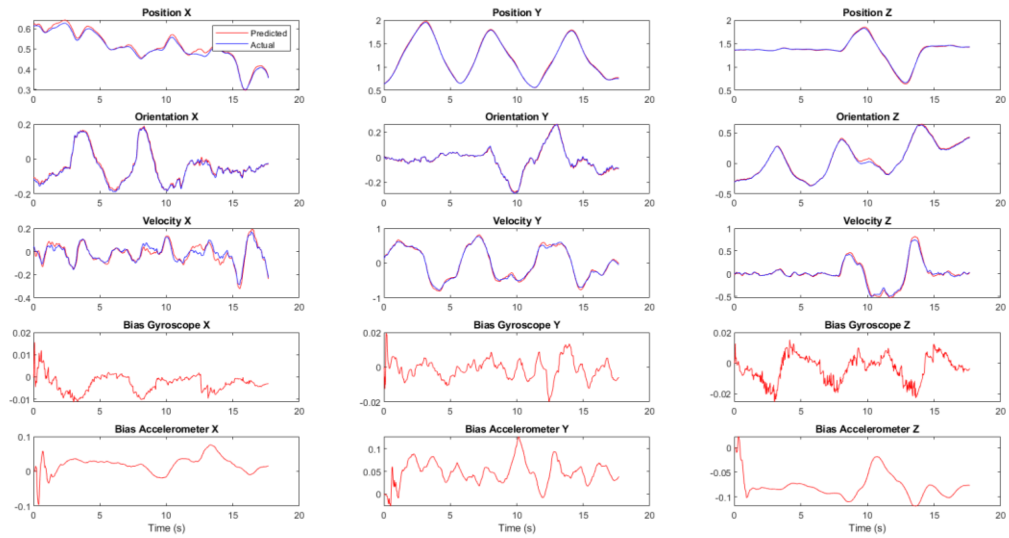
Here z_{uk} is $z_{\mu,t}$

After doing the above three steps, I am computing the filter gain, filter state mean and covariance by using the formulas mentioned below.

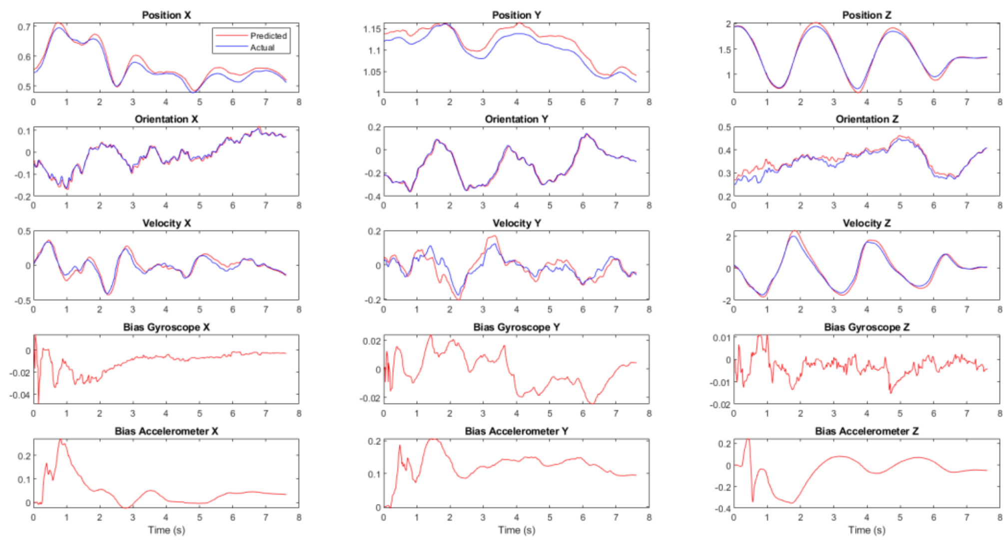
$$\boldsymbol{\mu}_t = \bar{\boldsymbol{\mu}}_t + \boldsymbol{K}_t\,(\boldsymbol{z}_t - \boldsymbol{z}_{\mu,t})$$
$$\boldsymbol{\Sigma}_t = \bar{\boldsymbol{\Sigma}}_t - \boldsymbol{K}_t\,\boldsymbol{S}_t \boldsymbol{K}_t{}^T$$
$$\boldsymbol{K}_t = \boldsymbol{C}_t\,\boldsymbol{S}_t{}^{-1}$$

So, this is how, I am doing the Part 1 and Part 2 of the project.

Results, Plots & Conclusions – Now, I am attaching the graphs coming from the results of both Part 1 and Part 2.



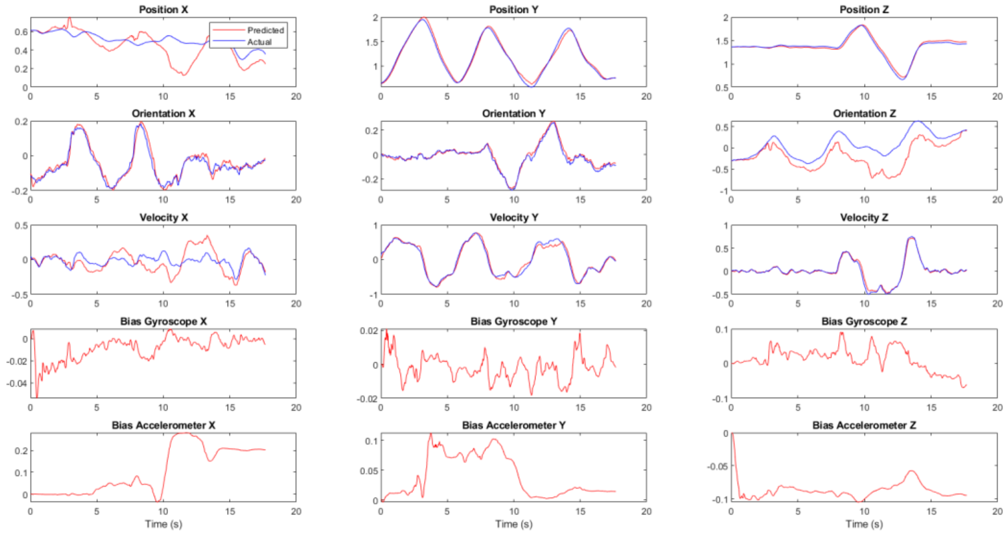
Part 1 Dataset 1



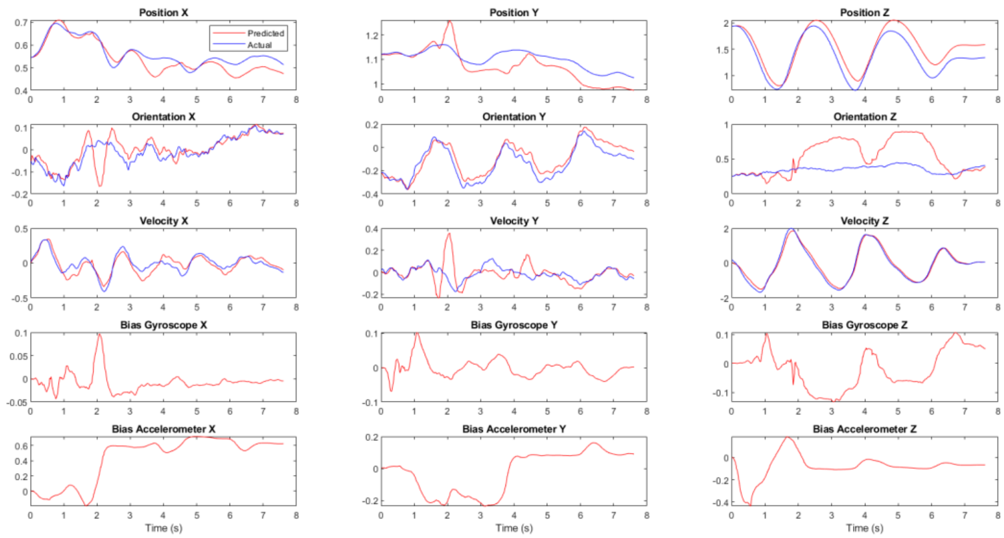
Part 1 Dataset 4

When we look at the plots of both datasets 1 and 4 for part 1 of the project, we will find that the blue and red plots are approximately matching.

For dataset 1, the accuracy of plot matching is quite better however for dataset 4, some plots for position, velocity is not so good. One of the reasons are the parameters like na, ng, nbg, nba and R are not tuned in a better way and there is some error in dataset 4.



Part 2, Dataset 1



Part 2, Dataset 4

When we look at the plots for both Dataset 1 and Dataset 4 of Part 2, we will see that the plots in blue and red are not quite overlapping as compared to Part 1 of the Project. The major reason for this is the tuning of some values of na, ng, nba, nbg, and R. If these variables can be tuned then the plots may overlap with each other, if I had more time, I would have done that, but I tried my best to make both blue and red plots converge by tuning the values na, ng, nba, nbg and R. It was a great learning experience.