Attitude Estimation using Non-stinky Unscented Kalman Filter on 6-DoF IMU

Ankush Singh Bhardwaj
abhardwaj@wpi.edu

Sri Lakshmi Hasitha Bachimanchi
sbachimanchi@wpi.edu

Anuj Pradeep Pai Raikar
apairaikar@wpi.edu

Late Days Used: 2

***Abstract*—This project presents the implementation of Non-stinky Unscented Kalman Filter to estimate the 3D orientation of the IMU (ArduIMU+ V2) by reading the acceleration and the gyroscope values given by it. A Madgwick Filter and a Complementary Filter is also implemented based on the fusion of the values from the accelerometer and the gyroscope to obtain the orientation. The orientation estimated by the filters is then compared with the ground truth from the Vicon motion capture system.**

I. Implementation

A. Reading and Appropriating the Data

The data is obtained from six degree IMU, 3-axis gyroscope and 3-axis accelerometer fitted on a drone. The orientation of the IMU recorded from the VICON motion capture is also provided. The data for each correspondence is provided in a .mat file. The IMU data exists as:

$$[a_x, a_y, a_z, w_z, w_x, w_y]$$

The VICON data meanwhile stores the timestamps **ts** and the orientation in a 3x3 Rotation matrix denoting **Z-Y-X** Euler angles for N time instances. The parameter values for the IMU are also provided in a 2x3 vector containing the Scale and Bias values.

The acceleration has been converted into m/s^2. The formula implemented in the code is:

$$\bar{a} = (a \cdot scale + bias) \cdot g$$

The angular rates have been converted into rad/s^{-1}.

$$\bar{\omega} = \frac{3300}{1023} \cdot \frac{\pi}{180} \cdot (\omega - b_g)$$

The bias for angular rate conversion is obtained by calculating the average of the first couple hundred values of each angular rate from the IMU data.

B. Implementation of the Methods to estimate Orientations

Methodology

We use four methods to estimate the orientation of the IMU. The first method uses only the gyroscope readings for attitude estimation, while the second uses the accelerometer readings. We also implement a Complementary filter and Madgwick filter by fusing the readings from both gyroscope and accelerometer.

$$R = R_z(\alpha)\,R_y(\beta)\,R_x(\gamma) = \overset{\text{yaw}}{\begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}} \overset{\text{pitch}}{\begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}} \overset{\text{roll}}{\begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}}$$
$$= \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix}$$

Fig. 1. 3D Rotation

Rotation Matrix using R-P-Y Euler Angles Method

The attitude is calculated from the roll, pitch and yaw rate by using the 3D rotation matrix formula notation in Fig 1.

1) **Orientation From Gyroscope Measurements:**

   Using Integration:

   - The function iterates through the IMU timestamps and gyroscope measurements obtained from the IMU data.
   - For each timestamp, the time difference between the current and previous timestamps is calculated.
   - The orientations are updated using the roll, pitch, and yaw angles calculated above w_x, w_y, w_z.

2) **Orientation from Accelerometer Measurements**:

   - The second method used only the accelerometer readings for attitude estimation.
   - The orientation is returned as well as the roll, pitch and yaw values stored as a vector are returned.
   - The formula is used keeping in mind IMU is only rotating and that the acceleration due to gravity is in the Z-axis.

$$\text{Roll }, \phi = \tan^{-1}\left(\frac{a_y}{\sqrt{a_x^2 + a_z^2}}\right)$$

$$\text{Pitch }, \theta = \tan^{-1}\left(\frac{-a_x}{\sqrt{a_y^2 + a_z^2}}\right)$$

$$\text{Yaw }, \psi = \tan^{-1}\left(\frac{\sqrt{a_x^2 + a_y^2}}{a_z}\right)$$

3) **Complementary Filter**:

- We low pass the accelerometer measurement data as described below. The alpha for Low Pass filter is taken as 0.2

$$\hat{\mathbf{a}}_{t+1} = (1-\alpha)\mathbf{a}_{t+1} + \alpha\hat{\mathbf{a}}_t$$

Fig. 2. Accelerometer LPF

- We high pass the gyroscope measurement data as described below. The alphs for High Pass filter is taken as 0.2.

$$\hat{\omega}_{t+1} = (1-\alpha)\hat{\omega}_t + (1-\alpha)(\omega_{t+1} - \omega_t)$$

Fig. 3. Gyroscope HPF

- The two measurements are fused into one using a α value that weighs the two. The alpha for complementary filter is taken as 0.5.

- The output orientation is stored for plotting.

4) **Madgwick Filter**:

The newer method used for the above estimated attitudes is **Madgwick** filter. This filter formulates the problem in quaternion space.

- We begin by calculating the readings for orientation obtained only from the gyroscope and only from the accelerometer, in the quaternion notation.
- Normalise the quaternion notation orientation we calculated in the above steps using formula in Fig 2.
- The second step is to perform the orientation increments from the accelerometer. Let our initial estimate begin at the quaternion [1 0 0 0].

$$q_w = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)$$
$$q_x = \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)$$
$$q_y = \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)$$
$$q_z = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right)$$

Fig. 4. Euler Angles to Quaternion Conversion

- We calculate the orientation estimate from the gyroscope using the formula below.

$${}^{I}_{W}\dot{\mathbf{q}}_{\omega,t+1} = \frac{1}{2} W\hat{\mathbf{q}}_{\text{est},t} \otimes \begin{bmatrix} 0 \\ {}^{I}\omega_{t+1} \end{bmatrix} \quad (1)$$

- The function **f** formulated for the compliance of the values and the Jacobian (**J**) are calculated and their product is used to get our gradient from our previous estimate.

$$\nabla f\left({}^{I}_{W}\hat{\mathbf{q}}_{\text{est},t},\ {}^{W}\hat{\mathbf{g}},\ {}^{I}\hat{\mathbf{a}}_{t+1}\right)$$
$$= J^T\left({}^{I}_{W}\hat{\mathbf{q}}_{\text{est},t},\ {}^{W}\hat{\mathbf{g}}\right)\nabla f\left({}^{I}_{W}\hat{\mathbf{q}}_{\text{est},t},\ {}^{W}\hat{\mathbf{g}},\ {}^{I}\hat{\mathbf{a}}_{t+1}\right) \quad (2)$$

where

$$J\left({}^{I}_{W}\hat{\mathbf{q}}_{\text{est},t},{}^{W}\hat{\mathbf{g}}\right) = \begin{bmatrix} -2q_3 & 2q_4 & -2q_1 & 2q_2 \\ 2q_2 & 2q_1 & 2q_4 & 2q_3 \\ 0 & -4q_2 & -4q_3 & 0 \end{bmatrix}$$

and

$$f\left({}^{I}_{W}\hat{\mathbf{q}}_{\text{est},t},{}^{W}\hat{\mathbf{g}},{}^{I}\hat{\mathbf{a}}_{t+1}\right) = \begin{bmatrix} 2\left(q_2q_4 - q_1q_3\right) - a_x \\ 2\left(q_1q_2 + q_3q_4\right) - a_y \\ 2\left(\frac{1}{2} - q_2^2 - q_3^2\right) - a_z \end{bmatrix} \quad (3)$$

- **Update Step:** The normalised gradient is multiplied by β and that product is subtracted from the new gyroscope measurement estimate.

The term **beta** is the trust ratio between the accelerometer and gyroscope values and is set to 0.09.(Manually tuned)

- Calculate use the product of the update step and the difference in the time step.
- The new estimate is given as the sum of the normalised previous estimate and the product above. See the formula below.

$$ {}^{I}_{W}\dot{\mathbf{q}}_{\text{est},t+1} = {}^{I}_{W}\dot{\mathbf{q}}_{\omega,t+1} + {}^{I}_{W}\mathbf{q}_{\nabla,t+1} $$

- Repeat the steps at each time instant.
- Estimates calculated at each time instant are converted to the Euler angle representation.

$$ \sin(pitch) = 2 \cdot (w \cdot y - z \cdot x) $$

$$ pitch = \begin{cases} \frac{\pi}{2} & \text{if } \sin(pitch) > 0 \\ -\frac{\pi}{2} & \text{if } \sin(pitch) < 0 \\ \arcsin(\sin(pitch)) & \text{otherwise} \end{cases} $$

$$ \sin(yaw + pitch) = 2 \cdot (w \cdot z + x \cdot y) $$

$$ \cos(yaw + pitch) = 1 - 2 \cdot (y^2 + z^2) $$

$$ yaw = \arctan 2(\sin(y + p), \cos(y + p)) $$

$$ \sin(roll) = 2 \cdot (w \cdot x - y \cdot z) $$

$$ roll = \begin{cases} \frac{\pi}{2} & \text{if } \sin(roll) > 0 \\ -\frac{\pi}{2} & \text{if } \sin(roll) < 0 \\ \arcsin(\sin(roll)) & \text{otherwise} \end{cases} $$

where, y is yaw and p is pitch.

5) **”Non-Stinky” Unscented Kalman Filter**:

Unlike Complementary filter and Madgwick filter, Unscented Kalman filter (UKF) uses a probabilistic Bayesian approach for orientation estimation and addresses non-linear models. It predicts and updates the estimate based on nonlinear process and measurement models by choosing a set of sigma points. These sigma points capture the distribution of state variables with mean and covariance values associated with them.

a) **Generation of Sigma Points:** Sigma points are calculated based on the current state, the covariance matrix P and the process noise Q as follows.

$$ \begin{aligned} S &= \sqrt{P_{k-1} + Q} \\ \mathcal{W}_{i,i+n} &= \text{ columns } \left(\pm\sqrt{2n \cdot (P_{k-1} + Q)} \right) \end{aligned} \tag{4} $$

The list of 2n sigma points $\mathcal{X}$ generated from nxn covariance matrix is represented as a state consisting of a quaternion and angular velocity.

$$ \mathcal{X}_i = \hat{x}_{k-1} + \mathcal{W}_i \tag{5} $$

$$ \mathcal{X}_i = \begin{pmatrix} q_{k-1} q_{\mathcal{W}} \\ \vec{\omega}_{k-1} + \vec{\omega}_{\mathcal{W}} \end{pmatrix} \tag{6} $$

b) **Process Model:** The process model for state transition computes differential quaternion based on prior angular velocities (assuming that the angular velocity remains constant during the time step Δt) and updates quaternion orientation. The updated state consists of the quaternion and angular velocity.

$$ \begin{aligned} \omega_k &= \omega_{k-1} \\ q_\Delta &= \left[\cos\left(\frac{|\vec{\omega}_{k-1}|\, \Delta t}{2} \right), \frac{\vec{\omega}_{k-1}}{|\vec{\omega}_{k-1}|} \sin\left(\frac{|\vec{\omega}_{k-1}|\, \Delta t}{2} \right) \right] \end{aligned} \tag{7} $$

The sigma points are transformed over the time step Δt using the process model for a new set of sigma points, $\mathcal{Y}$.

$$ \mathcal{Y}_i = A(\mathcal{X}_i, 0) = \begin{pmatrix} q_k q_w q_\Delta \\ \vec{\omega}_k + \vec{\mathrm{w}}_\omega \end{pmatrix} \tag{8} $$

c) **Mean of Sigma Points:** The weighted average of sigma points is estimated by computing the quaternion mean and angular velocity mean. The quaternion mean is estimated using the Intrinsic Gradient Descent method (for ensuring unit quaternion). It is computed using barycentric mean $\vec{e}$ signifying the deviation between the estimated mean $\bar{q}$ and the real mean orientation.

$$ \vec{e} = \frac{1}{2n} \sum_{i=1}^{2n} \vec{e}_i \tag{9} $$

where $\vec{e}_i$ is the error vector giving the relative rotation between set element q_i and the estimated mean of the last iteration $\bar{q}_t$,

$$ \vec{e}_i = q_i \bar{q}_t^{-1} \tag{10} $$

The angular velocity mean is computed by,

$$ \bar{\omega} = \frac{1}{2n} \sum_{i=1}^{2n} \omega_i \tag{11} $$

d) **Priori State Vector Covariance Update:** The Priori State Vector Covariance is computed using the sigma point deviations, $\mathcal{W}_i'$ calculated by subtracting mean state from each sigma point.

$$\mathcal{W}_i' = \begin{pmatrix} q_i \bar{q}^{-1} \\ \vec{\omega}_i - \overline{\vec{\omega}} \end{pmatrix} \tag{12}$$

$$\bar{P}_k = \frac{1}{2n} \sum_{i=1}^{2n} \mathcal{W}_i' \mathcal{W}_i'^T \tag{13}$$

$$\bar{P}_k = \frac{1}{2n} \sum_{i=1}^{2n} [\mathcal{X}_i - \bar{x}] [\mathcal{X}_i - \bar{x}]^T \tag{14}$$

e) **Measurement Model:** The measurement models for gyro and accelerometer measurements, H_1 and H_2 can be used to compute predicted measurements $\mathcal{Z}_i$ from the sigma points and the average predicted measurement $\bar{z}_k$.

$$\begin{array}{ll} H_1: & \vec{z}_{\text{rot}} = \overrightarrow{\text{w}}_{\text{k}} + \overrightarrow{\text{v}}_{\text{rot}} \\ H_2: & \vec{z}_{acc} = \vec{g}' + \overrightarrow{\text{v}}_{acc} \end{array} \tag{15}$$

$$\mathcal{Z}_i = \begin{pmatrix} q_i^{-1} g q_i \\ \vec{\omega}_k \end{pmatrix} \tag{16}$$

$$\bar{z}_k = \frac{1}{2n} \sum_{i=1}^{2n} \mathcal{Z}_i \tag{17}$$

f) **Measurement Estimate Covariance:** The covariance of predicted measurements $\mathcal{Z}_i$ can be computed similarly as equation 13 as follows, where ϕ_i' is the measurement deviation calculated by subtracting predicted measurement from the set.

$$\bar{P}_{zz} = \frac{1}{2n} \sum_{i=1}^{2n} \phi_i' \phi_i'^T \tag{18}$$

where,

$$\phi_i' = \begin{pmatrix} q_i \bar{q}^{-1} \\ \vec{\omega}_i - \overline{\vec{\omega}} \end{pmatrix} \tag{19}$$

$$\bar{P}_{zz} = \frac{1}{2n} \sum_{i=1}^{2n} \left[\mathcal{Z}_i - z_k^-\right] \left[\mathcal{Z}_i - z_k^-\right]^T \tag{20}$$

The difference between the predicted measurement $\bar{z}_k$ and the actual measurement z_k is called Innovation, v_k that predicts the deviation in the UKF's predicted measurements and the observed measurements. The covariance in the Innovation, P_{vv} is computed as,

$$P_{vv} = P_{zz} + R \tag{21}$$

where, R is measurement noise covariance.

g) **Cross Correlation Matrix:** Cross correlation matrix, P_{xz} helps in quantifying effects of changes of predicted state affect on predicted measurements. It is used to calculate kalman gain and helps in weighing the predicted state estimate. Large values of P_{xz} indicates filter giving more weight to the prediction. P_{xz} is calculated as follows,

$$\bar{P}_{xz} = \frac{1}{2n} \sum_{i=1}^{2n} \mathcal{W}_i' \phi_i'^T \tag{22}$$

$$\bar{P}_{xz} = \frac{1}{2n} \sum_{i=1}^{2n} \left[\mathcal{Y}_i - \hat{x}_k^-\right] \left[\mathcal{Z}_i - z_k^-\right]^T \tag{23}$$

and Kalman gain, K_k,

$$K_k = P_{xz} P_{\nu\nu}^{-1} \tag{24}$$

Further, the state and its covariance are updated as the below equations.

$$\hat{x}_k = \hat{\bar{x}} x_k + K_k \nu_k \tag{25}$$

$$P_k = \bar{P}_k - K_k P_{\nu\nu} K_k^T \tag{26}$$

II. Results

The plots of the angles estimated from Complementary Filter, Madgwick Filter and Unscented Kalman Filter are plotted with angles from the VICON data.

Please see the figures attached here. The link to the videos obtained after running the `rotplot.py` and can be referred to with the name of the data set where 1-6 is for the given data set and 7-10 is the test sets.

Link: rotplots

III. References

1 A Quaternion-based Unscented Kalman Filter for Orientation Tracking by Edgar Kraft
2 https://github.com/NitinJSanket/ESE650Project2.pdf
3 https://nitinjsanket.github.io/tutorials/attitudeest/
4 ENAE788M: YouTube

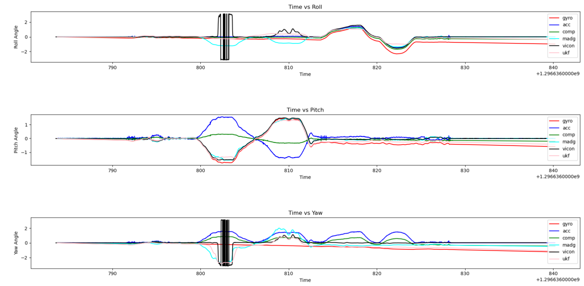

Fig. 5. Plots for Dataset 1

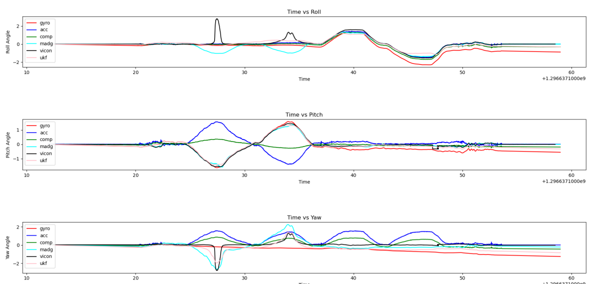

Fig. 6. Plots for Dataset 2

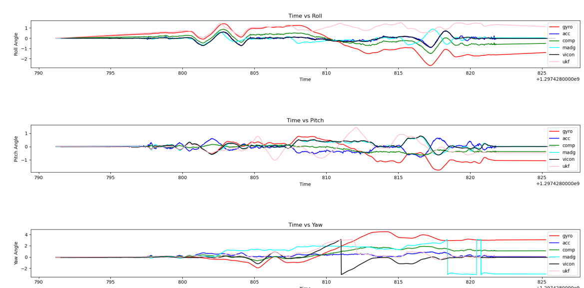

Fig. 7. Plots for Dataset 3

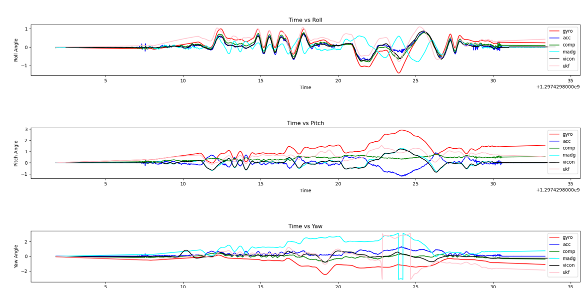

Fig. 8. Plots for Dataset 4

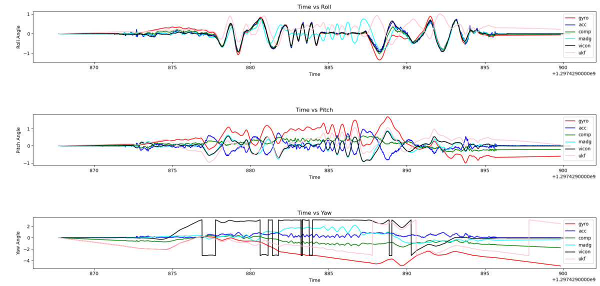

Fig. 9. Plots for Dataset 5

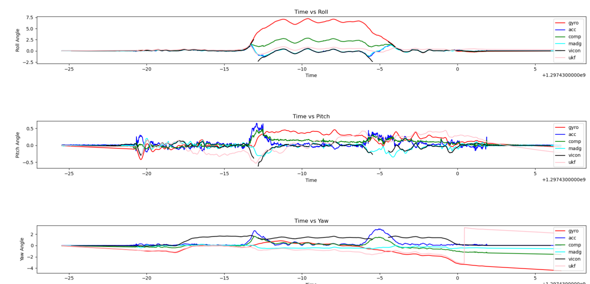

Fig. 10. Plots for Dataset 6

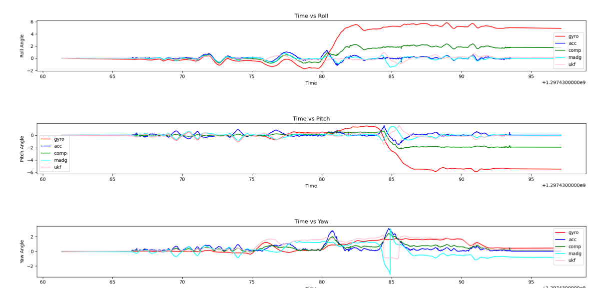

Fig. 11. Plots for Test Dataset 1

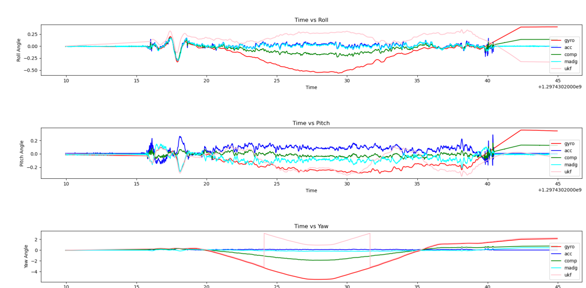

Fig. 12. Plots for Test Dataset 2

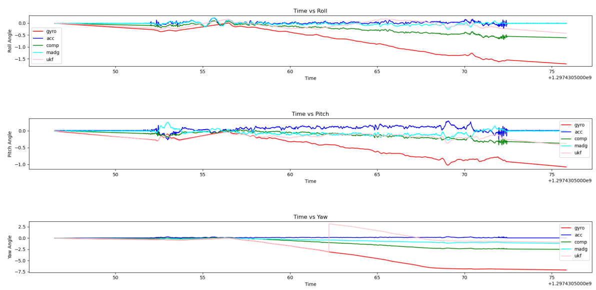

Fig. 13. Plots for Test Dataset 3

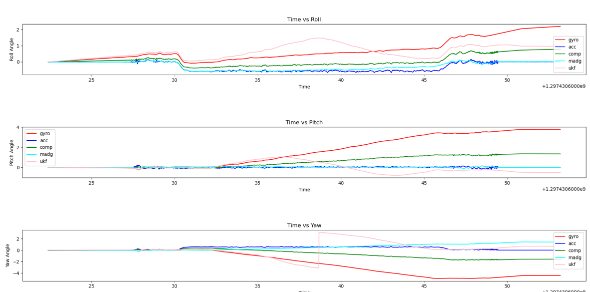

Fig. 14. Plots for Test Dataset 4