P1: Madgwick Filter

Mayank Bansal
Robotics Engineering
Worcester Polytechnic Institute
Email: mbansal1@wpi.edu

Siyuan 'Oliver' Huang
Robotics Engineering
Worcester Polytechnic Institute
Email: shuang4@wpi.edu

Miheer Diwan
Robotics Engineering
Worcester Polytechnic Institute
Email: msdiwan@wpi.edu

***Abstract*—The aim of this project was to estimate the 3D orientation (attitude) of an IMU using four different methods — using an ideal gyroscope, using an ideal accelerometer, using a Complementary filter, and using a Madgwick Filter.**

I. Phase 1

In Phase 1 of this project, we performed IMU attitude estimation using four different methods — using a gyroscope, an accelerometer, a Complementary filter, and a Madgwick filter. The four methods were tested on six different training datasets and four test sets. The results were plotted for analysis against the ground truth.

A. Data Processing

The data was collected with an ArduIMU+ V2, a six degree of freedom Inertial Measurement Unit and the ground truth data was collected using a Vicon motion capture system. Each data file consists of six values and their corresponding timestamps — three linear acceleration values and three angular velocity values in the x, y and z direction, in the form $\begin{bmatrix} a_x & a_y & a_z & \omega_z & \omega_x & \omega_y \end{bmatrix}^T$.

The accelerometer bias and scale parameters were also provided in a separate file in the form of a 2×3 vector where the first row denotes the scale values — $\begin{bmatrix} s_x & s_y & s_z \end{bmatrix}$ and the second row denotes the bias values — $\begin{bmatrix} b_{a,x} & b_{a,y} & b_{a,z} \end{bmatrix}$.

The Vicon ground truth data is extracted in a similar manner. It contains the rotation matrices estimated from ZYX Euler angles in the form of $3 \times 3 \times N$ matrices and their corresponding timestamps.

1) Data Conversion: The IMU data is not in physical units. Therefore, we need to convert it before attitude estimation. Converting acceleration values to ms^{-2}:

$$\tilde{a_x} = \frac{a_x + b_{a,x}}{s_x} \tag{1}$$

where, $\tilde{a_x}$ represents the value of a_x in physical units, ba_x is the bias and s_x is the scale factor of the accelerometer. Converting angular velocities to rad/s:

$$\tilde{\omega} = \frac{3300}{1023} \times \frac{\pi}{180} \times 0.3 \times (\omega - b_g) \tag{2}$$

Here, $\tilde{\omega}$ represents the value of ω in physical units and b_g is the bias. b_g was calculated as the average of the first hundred samples (assuming that the IMU is at rest in the beginning).

2) Time Stamp Alignment Using Slerp: Since the timestamps of the IMU data and the Vicon ground truth data are not aligned, I aligned them using Slerp.

II. Attitude Estimation Using an Ideal Gyroscope

The gyroscope mathematical model is given by:

$$\omega = \hat{\omega} + \mathbf{b}_g + \mathbf{n}_g \tag{3}$$

Here, ω is the measured angular velocity from the gyroscope, $\hat{\omega}$ is the latent ideal angular velocity we wish to recover, $\mathbf{b}_g$ is the gyroscope bias which changes with time and other factors like temperature, and $\mathbf{n}_g$ is the white Gaussian gyroscope noise.We estimate the orientation using only gyroscope data by integration. Since, we cannot perform integration without knowing the initial values, we assume that the initial orientation from Vicon is known. The initial values can also be estimated from other sensors or by starting from rest. We have the angular velocities at timestamp t and we want to estimate the orientations at timestamp $t+1$. The eqn. can be given by:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_{t+1} = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix}_t + \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}_t \times \delta t \tag{4}$$

The computed output for imuRaw1.mat is shown in Fig. 1

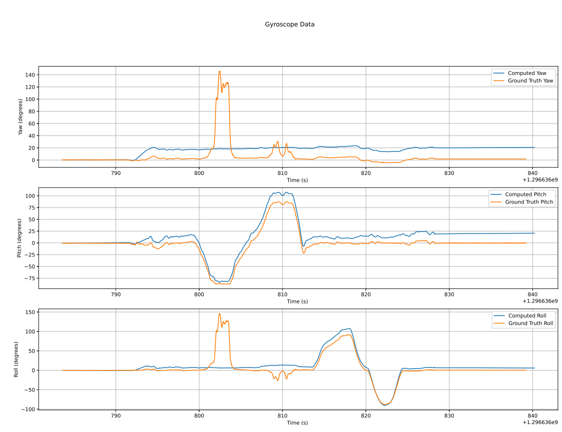

Fig. 1. Gyroscope Attitude Estimation for imuRaw1.mat

III. Attitude Estimation Using an Ideal Accelerometer

The accelerometer mathematical model is given by:

$$\mathbf{a} = {}^{W}R_B{}^{T}(\hat{\mathbf{a}} - \mathbf{g}^{W}) + \mathbf{b}_a + \mathbf{n}_a \tag{5}$$

Here, $\mathbf{a}$ is the measured acceleration from the acc, $\hat{\mathbf{a}}$ is the latent ideal acceleration we wish to recover, $\mathbf{R}$ is the orientation of the sensor in the world frame, $\mathbf{g}$ is the acceleration due to gravity in the world frame, $\mathbf{b}_a$ is the accelerometer bias which changes with time and other factors like temperature and $\mathbf{n}_a$ is the the white gaussian accelerometer noise.

We assume that our IMU is only rotating and we want to estimate the orientations in the next state ($\begin{bmatrix}\phi & \theta & \psi\end{bmatrix}_{t+1}^{T}$) when we have the linear accelerations in x, y and z direction in the previous state ($\begin{bmatrix}a_x & a_y & a_z\end{bmatrix}_{t}^{T}$). We also assume that the world frame is oriented in such a way that the negative z axis coincides with the gravity vector. Now, the only forces acting on the accelerometer are coming from the gravity vector. So,

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

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

It should be noted that since the accelerometer is symmetric with respect to the z axis, it does not provide any information about Yaw. However, since the IMU is never completely vertical, the Yaw can be computed as:

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

The computed output for imuRaw1.mat is shown in Fig. 2

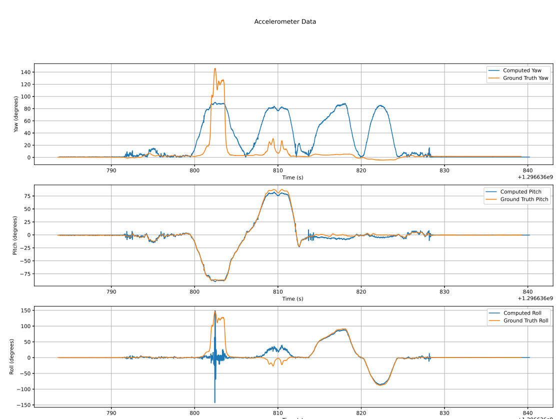

Fig. 2. Accelerometer Attitude Estimation for imuRaw1.mat

IV. Attitude Estimation Using Complementary Filter

In orientation estimation, gyroscope and accelerometer data play crucial roles, each with distinct advantages and limitations. However, gyroscopes are susceptible to bias and drift over time due to varying biases. Accelerometers, on the other hand, offer stability in long-term orientation estimation by utilizing gravity-influenced readings, but they falter in accurately capturing rapid movements. To counter noise in accelerometer data, low-pass filtering is applied, albeit at the cost of introducing a proportional lag represented by the filter's time constant (α). To tackle these challenges, a complementary filter is employed. Gyroscope data is high-pass filtered to alleviate bias-related drift, while accelerometer data is low-pass filtered to minimize noise. These filtered outputs are then weighted, scaled, and combined, harnessing the strengths of both sensors while mitigating their weaknesses. This approach results in accurate orientation estimates that are both responsive to fast changes and resistant to long-term drift.

Low pass filter formula: [α = 0.9]

$$\hat{\mathbf{a}}_{t+1} = (1-\alpha)\mathbf{a}_{t+1} + \alpha\hat{\mathbf{a}}_t \tag{6}$$

High pass filter formula: [α = 0.003]

$$\hat{\omega}_{t+1} = (1-\alpha)\hat{\omega}_t + (1-\alpha)(\omega_{t+1} - \omega_t) \tag{7}$$

Complementary Filter formula: [α = 0.5]

$$Ang_{t+1} = (1-\alpha)(Ang_t + \omega_{t+1}dt) + \alpha\mathbf{a}_{t+1} \tag{8}$$

The complementary filter is shown in Fig. 3.

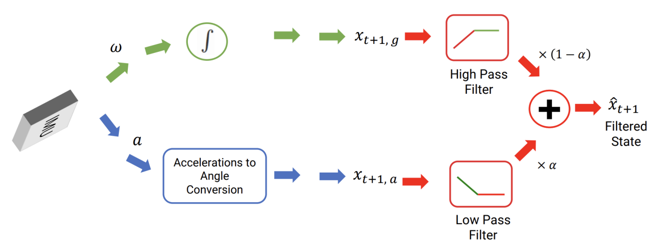

Fig. 3. Complementary Filter

The computed output for imuRaw1.mat is shown in Fig. 4

V. Attitude Estimation Using Madgwick Filter

The Madgwick filter operates by fusing data from the accelerometer and gyroscope to produce accurate and real-time attitude estimates. It combines sensor measurements with a quaternion representation $q = \begin{bmatrix}q1, q2, q3, q4\end{bmatrix}$ in w,x,y,z format which deals with the gimbal lock problem caused because of Euler angles. The major steps involved in implementing a Madgwick Filter are shown below and a detailed mathematical explanation for the same can be found in [1]. Fig. 5 shows the overview of the Madgwick Filter.

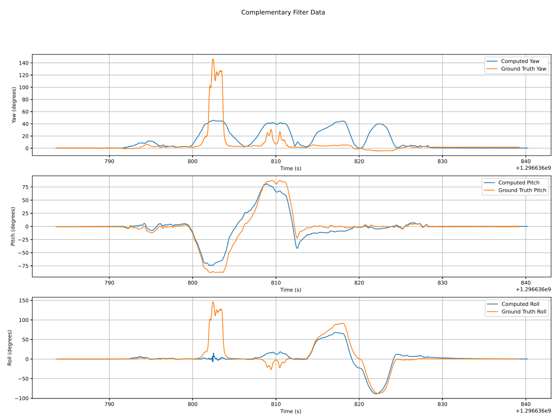

Fig. 4. Complementary Filter Attitude Estimation for imuRaw1

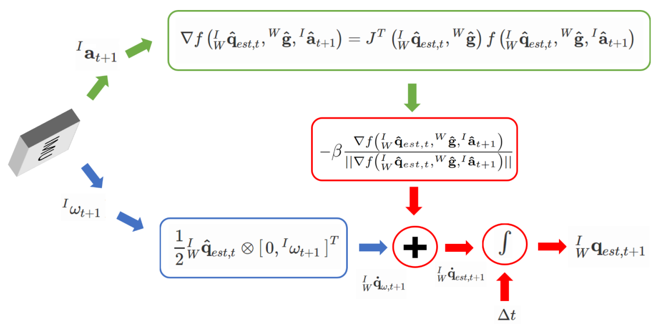

Fig. 5. Madgwick Filter

A. Calculating Orientation Increment from Accelerometer:

We model the attitude estimation of the accelerometer by modeling it as an optimization problem. We basically minimize the function (f).

$$\min_{{}^{I}_{W}\hat{\mathbf{q}} \in R^{4\times 1}} f\left({}^{I}_{W}\hat{\mathbf{q}}, {}^{W}\hat{\mathbf{g}}, {}^{I}\hat{\mathbf{a}}\right) \tag{9}$$

$$f\left({}^{I}_{W}\hat{\mathbf{q}}, {}^{W}\hat{\mathbf{g}}, {}^{I}\hat{\mathbf{a}}\right) = {}^{I}_{W}\hat{\mathbf{q}}^{*} \otimes {}^{W}\hat{\mathbf{g}} \otimes {}^{I}_{W}\hat{\mathbf{q}} - {}^{I}\hat{\mathbf{a}} \tag{10}$$

where, $\hat{\mathbf{q}}$ is the normalized quaternion in w,x,y,z format, ${}^{W}\hat{\mathbf{g}}$ is the gravity vector in the world frame given by ${}^{W}\hat{\mathbf{g}} = \begin{bmatrix}0 & 0 & 0 & 1\end{bmatrix}^{T}$ and ${}^{I}\hat{\mathbf{a}}$ is the normalized accelerometer measurements in the Inertial frame. To minimize this function, we use a gradient descent algorithm:

$$\nabla f\left({}^{I}_{W}\hat{\mathbf{q}}_{est,t}, {}^{W}\hat{\mathbf{g}}, {}^{I}\hat{\mathbf{a}}_{t+1}\right) = \tag{11}$$

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

$$f\left({}^{I}_{W}\hat{\mathbf{q}}_{est,t}, {}^{W}\hat{\mathbf{g}}, {}^{I}\hat{\mathbf{a}}_{t+1}\right) = \begin{bmatrix} 2\left(q_2 q_4 - q_1 q_3\right) - a_x \\ 2\left(q_1 q_2 + q_3 q_4\right) - a_y \\ 2\left(\frac{1}{2} - q_2^2 - q_3^2\right) - a_z \end{bmatrix} \tag{12}$$

J is the Jacobian of the function given by,

$$J\left({}^{I}_{W}\hat{\mathbf{q}}_{est,t}, {}^{W}\hat{\mathbf{g}}\right) = \begin{bmatrix} -2q_3 & 2q_4 & -2q_1 & 2q_2 \\ 2q_2 & 2q_1 & 2q_4 & 2q_3 \\ 0 & -4q_2 & -4q_3 & 0 \end{bmatrix} \tag{13}$$

The gradient update is given as:

$${}^{I}_{W}\mathbf{q}_{\nabla,t+1} = -\beta \frac{\nabla f\left({}^{I}_{W}\hat{\mathbf{q}}_{est,t}, {}^{W}\hat{\mathbf{g}}, {}^{I}\hat{\mathbf{a}}_{t+1}\right)}{\|f\left({}^{I}_{W}\hat{\mathbf{q}}_{est,t}, {}^{W}\hat{\mathbf{g}}, {}^{I}\hat{\mathbf{a}}_{t+1}\right)\|} \tag{14}$$

where β is a tunable parameter that models the magnitude of gyroscope error in the direction of the accelerometer measurements. After making some assumptions we are able to get this simplified equation which will help us converge in one step. These assumptions are given in detail in [1].

B. Calculating Orientation Increment from Gyroscope:

When it comes to gyroscopic data, the filter performs an incremental update to refine the orientation estimation. Suppose gyroscope measurements, represented as $\boldsymbol{\omega}$, provide information about the angular velocity of the object. To perform an incremental update, we apply a mathematical operation called quaternion multiplication, denoted as $\otimes$, which combines our current orientation quaternion q with a new quaternion derived from the gyroscope data. This is defined by equation (15):

$$\mathbf{q}_1 \otimes \mathbf{q}_2 = \begin{bmatrix} w_1 w_2 - x_1 x_2 - y_1 y_2 - z_1 z_2 \\ w_1 x_2 + x_1 w_2 + y_1 z_2 - z_1 y_2 \\ w_1 y_2 - x_1 z_2 + y_1 w_2 + z_1 x_2 \\ w_1 z_2 + x_1 y_2 - y_1 x_2 + z_1 w_2 \end{bmatrix} \tag{15}$$

$${}^{I}_{W}\dot{\mathbf{q}}_{\omega,t+1} = \frac{1}{2}{}^{I}_{W}\hat{\mathbf{q}}_{est,t} \otimes \left[0, {}^{I}\omega_{t+1}\right]^{T} \tag{16}$$

In the equation (16): - ${}^{I}_{W}\hat{\mathbf{q}}_{est,t}$ represents our current orientation estimate in the inertial frame. - ${}^{I}_{W}\dot{\mathbf{q}}_{\omega,t+1}$ represents the orientation increment from gyroscope measurements. - $\left[0, {}^{I}\omega_{t+1}\right]$ represents the angular velocity as a quaternion.

The result of this computation represents the incremental change in orientation caused by the gyroscope measurements over the given time interval. This incremental update is then applied to the current orientation estimate, to refine the estimate of the object's orientation.

C. Fuse Measurements:

The orientation estimates from both the gyroscope and accelerometer are fused to obtain the estimated attitude ${}^{I}_{W}\mathbf{q}_{est,t+1}$:

$${}^{I}_{W}\dot{\mathbf{q}}_{est,t+1} = {}^{I}_{W}\dot{\mathbf{q}}_{\omega,t+1} + {}^{I}_{W}\mathbf{q}_{\nabla,t+1} \tag{17}$$

$${}^{I}_{W}\mathbf{q}_{est,t+1} = {}^{I}_{W}\hat{\mathbf{q}}_{est,t} + {}^{I}_{W}\dot{\mathbf{q}}_{est,t+1}\Delta t \tag{18}$$

This is repeated for each time step to calculate the orientation over time.

VI. RESULTS

- The estimated orientations from gyroscope, accelerometer and complementary filter plotted against the Vicon ground truths for the six training data sets are shown in Fig. 6, Fig. 7, Fig. 8, Fig. 9, Fig. 10 and Fig. 11 respectively.
- The estimated orientations from Madgwick filter plotted against the Vicon ground truths for the six training data sets are shown in Fig. 12, Fig. 13, Fig. 14, Fig. 15, Fig. 16 and Fig. 17 respectively.
- The estimated orientations from Madgwick filter for the test data sets are shown in Fig. 18, Fig. 19, Fig. 20, Fig. 21 respectively.
- Rotplot Output Link

VII. CONCLUSION

- From the figures, we can observe that the gyroscope data is pretty accurate but suffers from drift over time while the accelerometer data has noise.
- The complementary filter is able to reduce the drift and noise after fusing the outputs of the accelerometer and the gyroscope data. However, the complementary filter still drifts over time, although not as bad as the gyroscope. Also, if we have violent motions, the complementary filter does not work well as it depends on the individual values of the gyroscope and the accelerometer.
- It was also observed that removing the bias, in the beginning, will give a better output in the complementary filter.
- From the plots, we can observe that the Madgwick Filter performs better than the Complementary filter for attitude estimation.
- Avoids Gimbal Lock: The use of quaternions eliminates the problem of gimbal lock, a limitation of Euler angles, ensuring robust and stable attitude estimation.
- Computationally Efficient: The Madgwick filter is computationally efficient as we use the gradient descent algorithm to optimize the function in a single step.

REFERENCES

[1] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of imu and marg orientation using a gradient descent algorithm," in *2011 IEEE International Conference on Rehabilitation Robotics*, 2011, pp. 1–7.

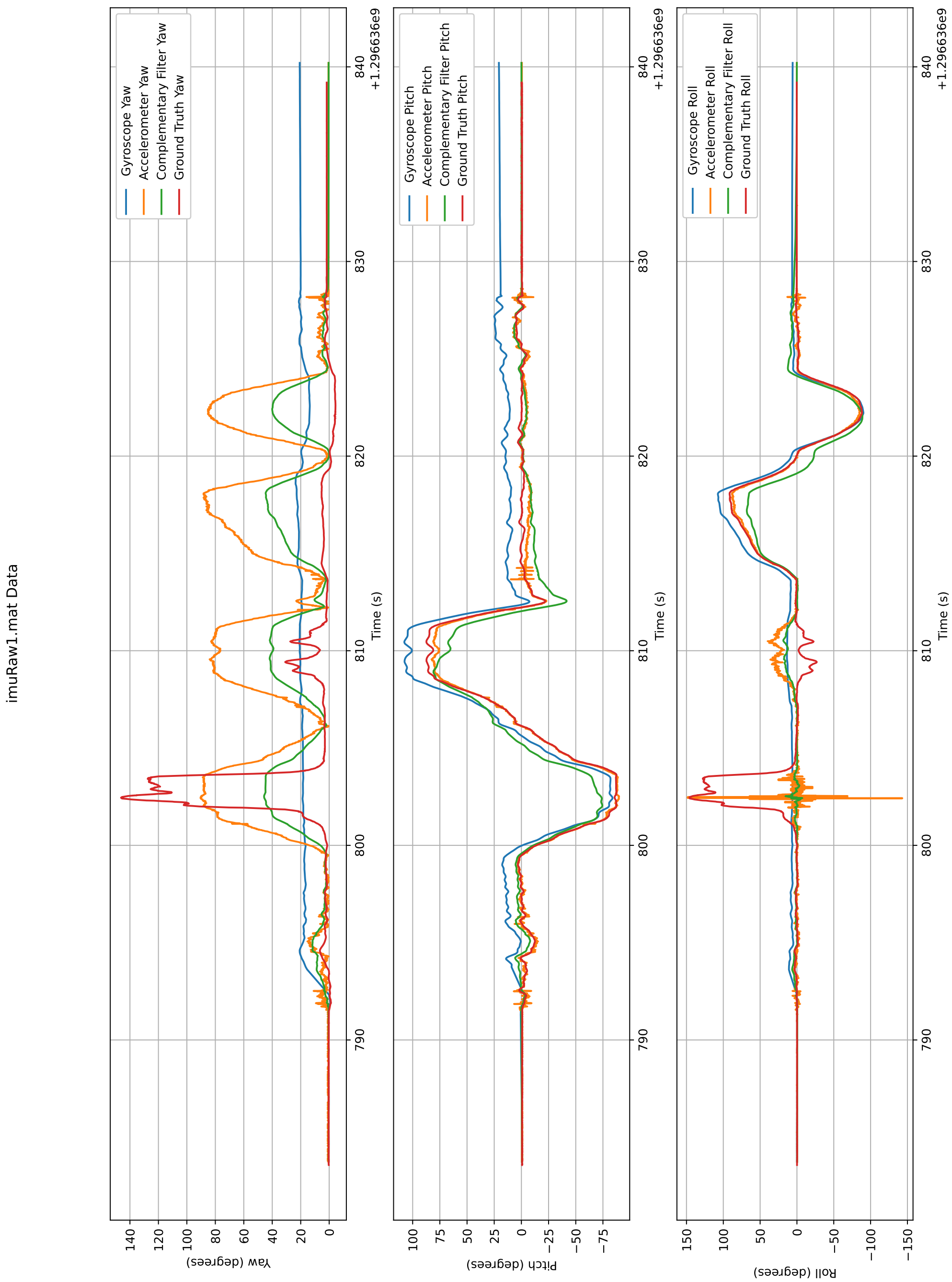

Fig. 6. Orientations for Dataset 1

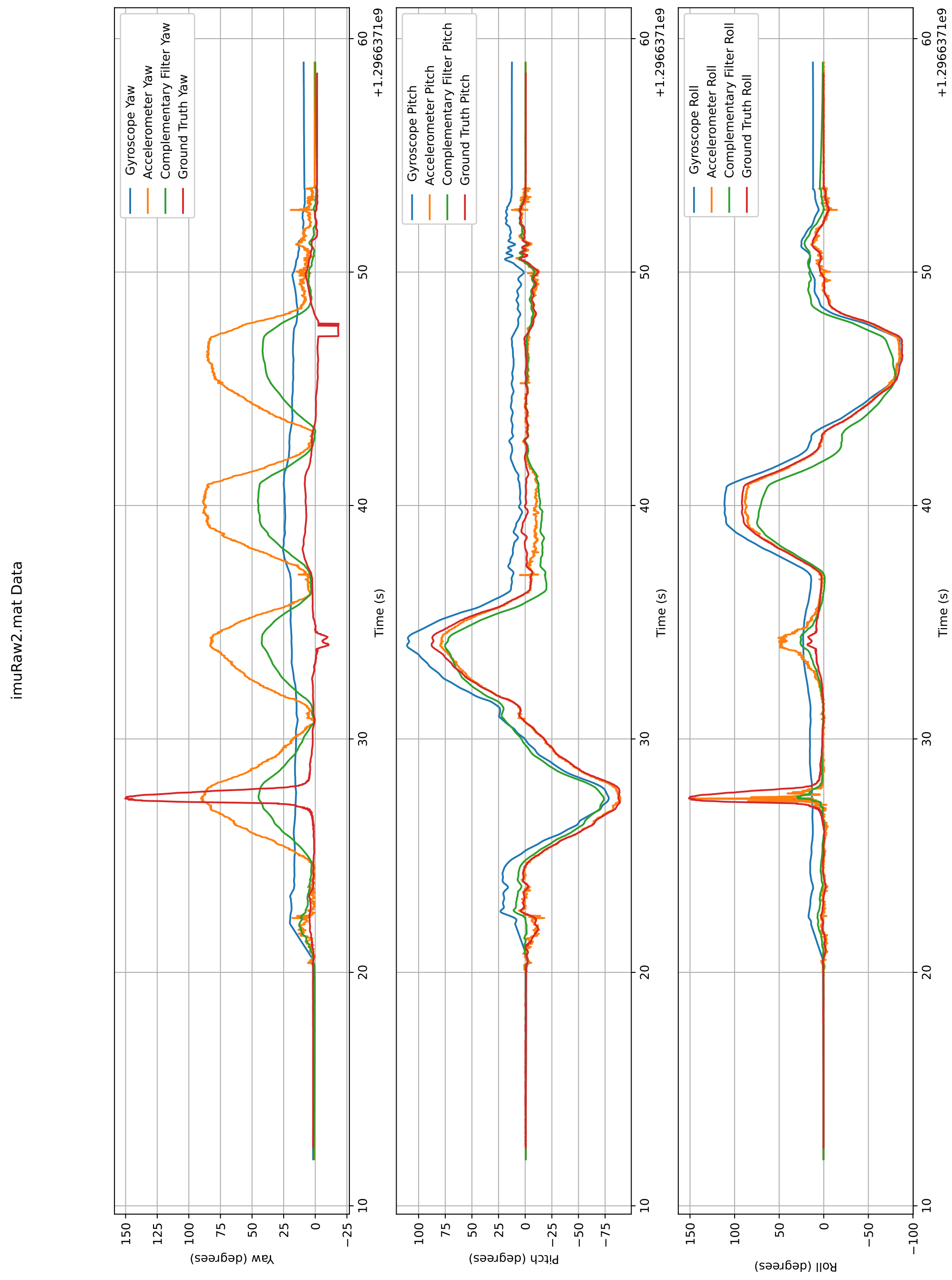

Fig. 7. Orientations for Dataset 2

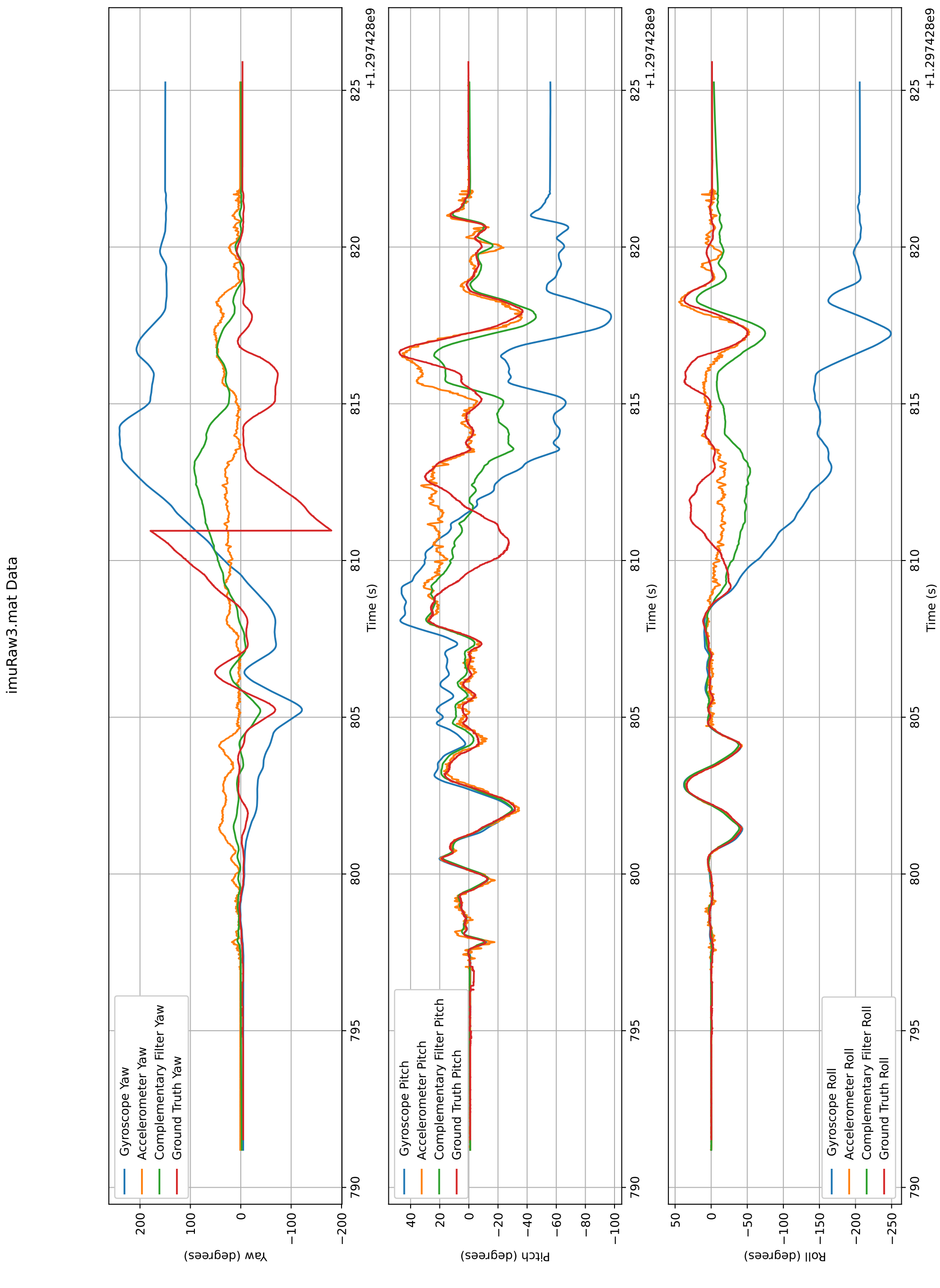

Fig. 8. Orientations for Dataset 3

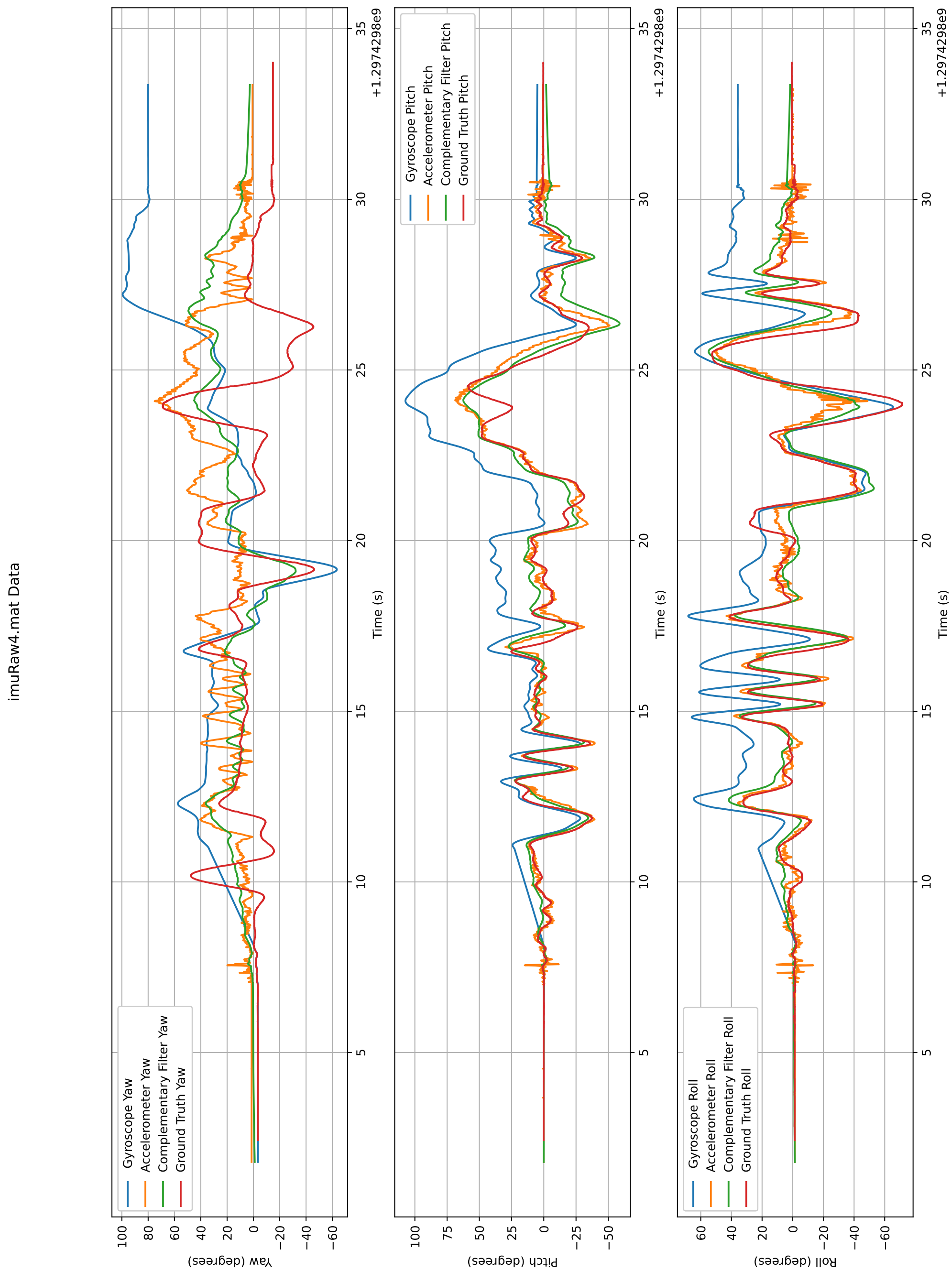

Fig. 9. Orientations for Dataset 4

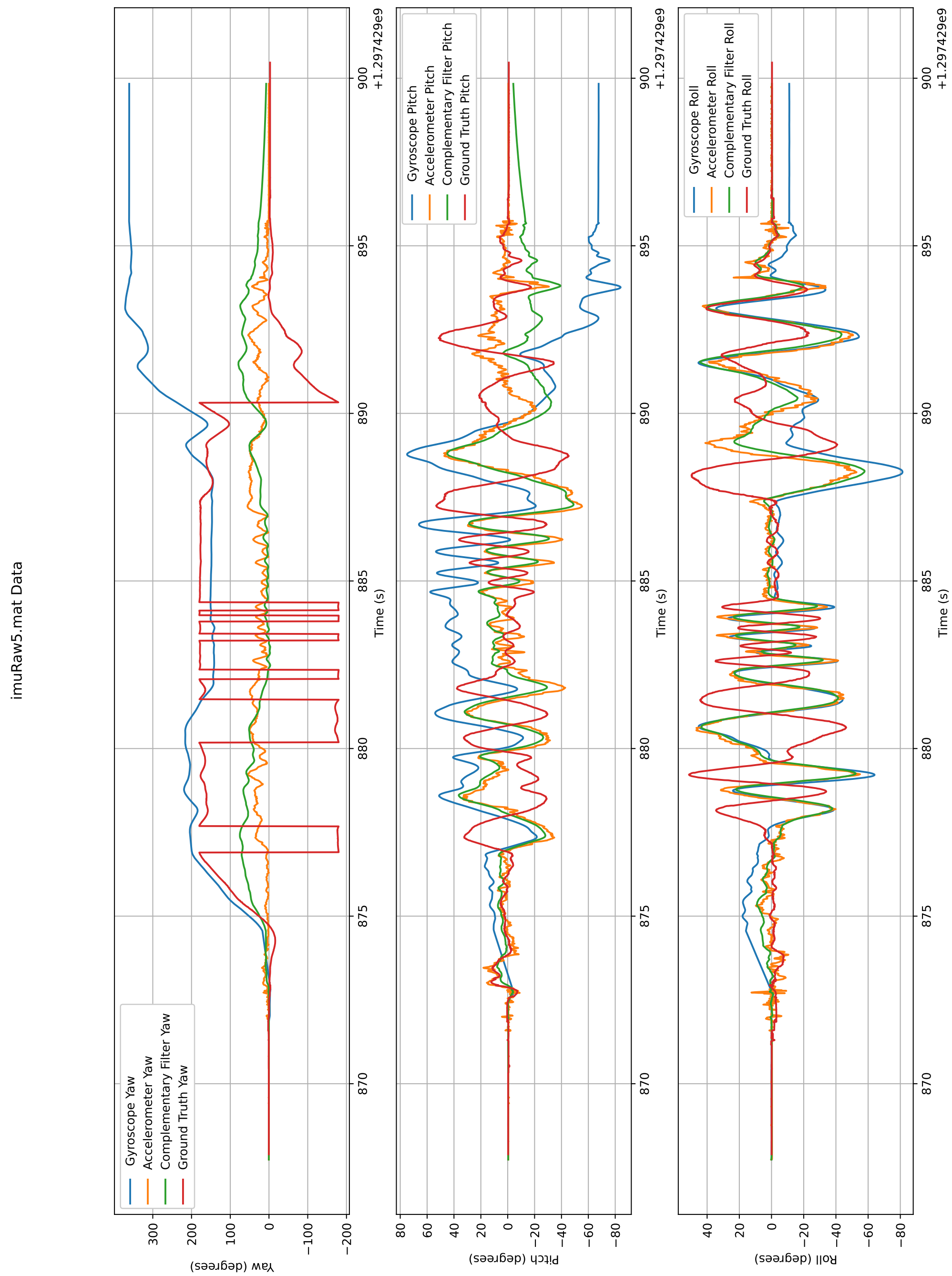

Fig. 10. Orientations for Dataset 5

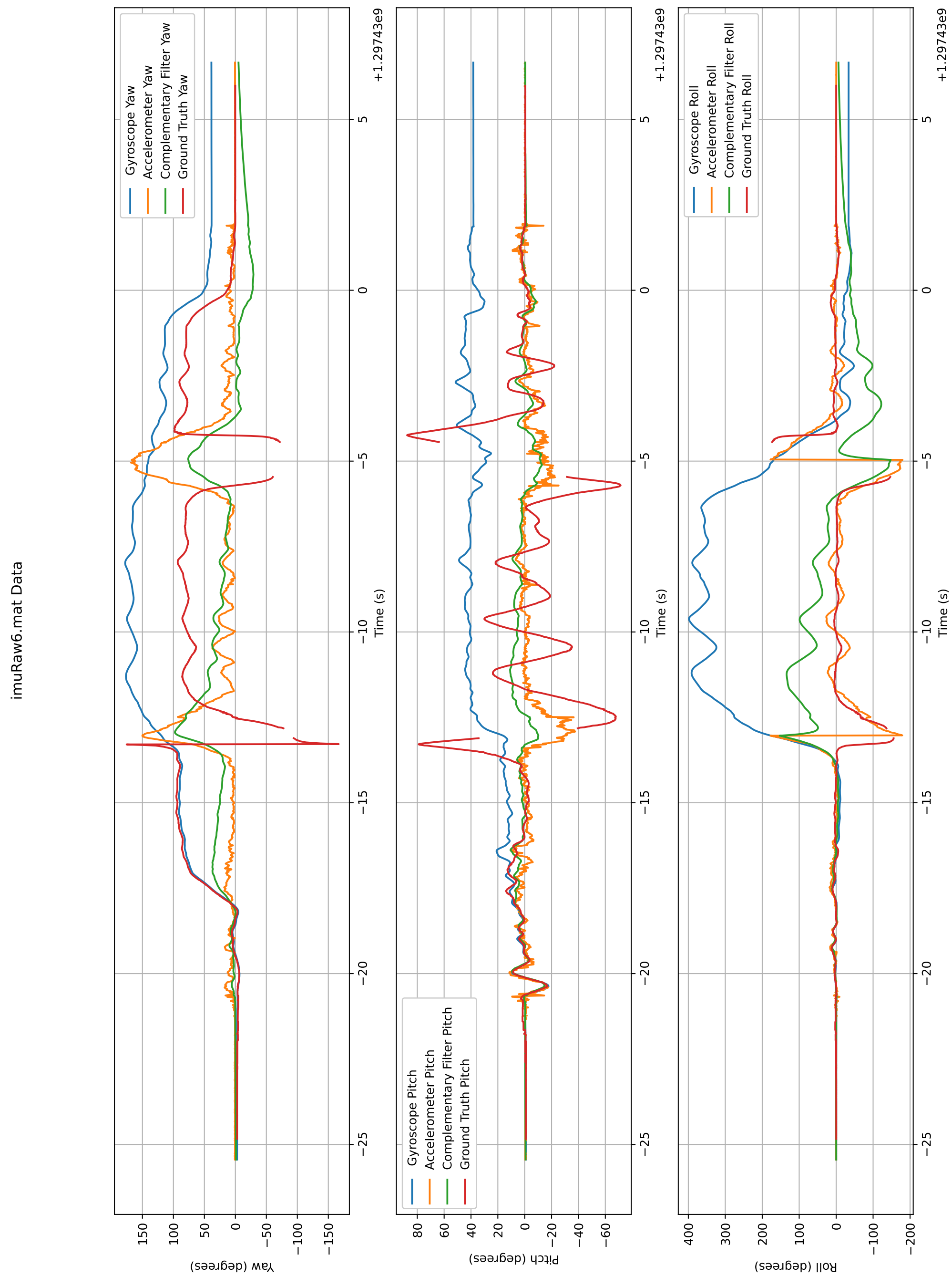

Fig. 11. Orientations for Dataset 6

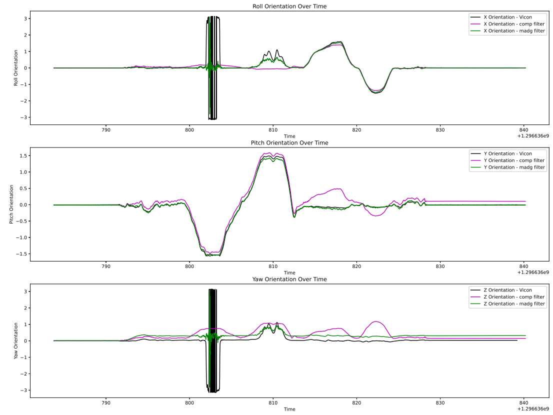

Fig. 12. Madgwick Filter Attitude Estimation for imuRaw1

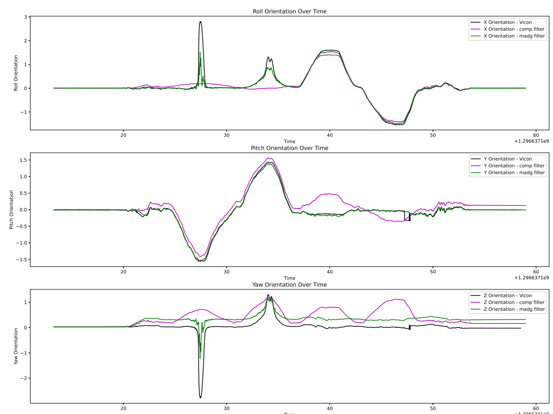

Fig. 13. Madgwick Filter Attitude Estimation for imuRaw1

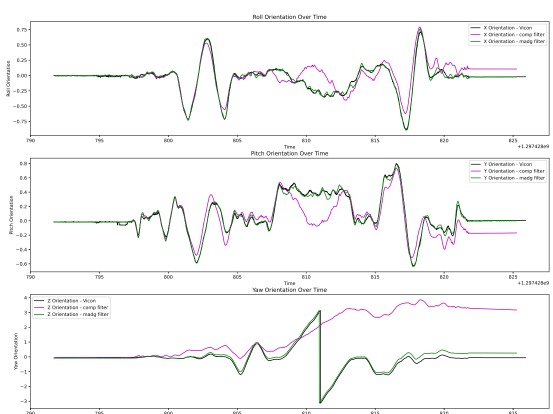

Fig. 14. Madgwick Filter Attitude Estimation for imuRaw1

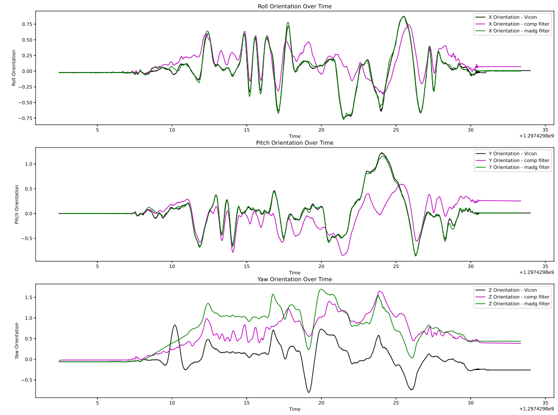

Fig. 15. Madgwick Filter Attitude Estimation for imuRaw1

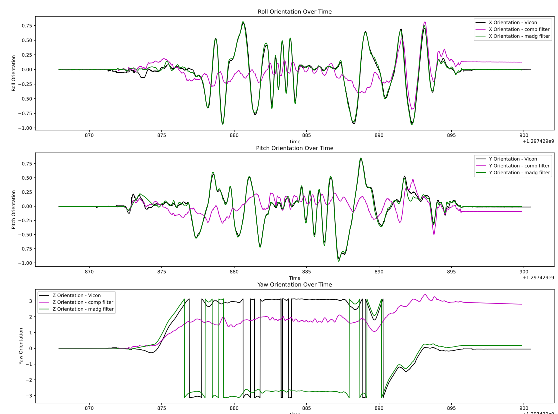

Fig. 16. Madgwick Filter Attitude Estimation for imuRaw1

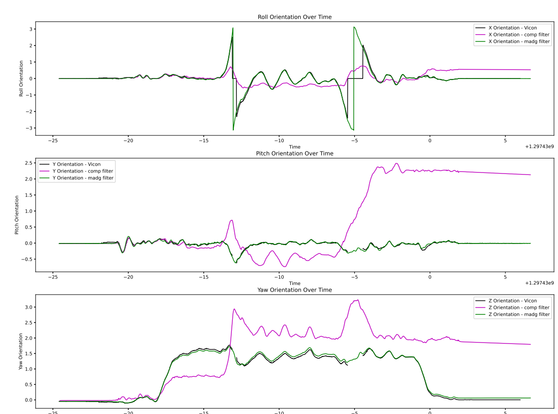

Fig. 17. Madgwick Filter Attitude Estimation for imuRaw1

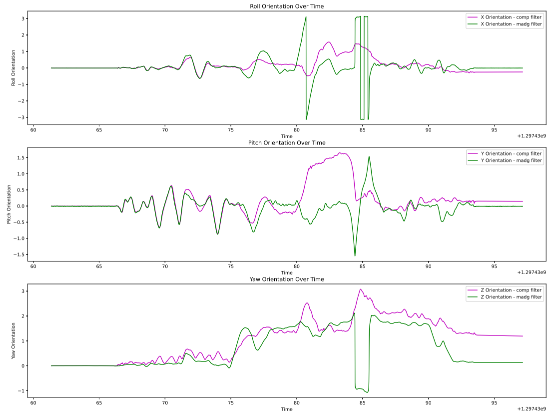

Fig. 18. Madgwick Filter Attitude Estimation for imuRaw7

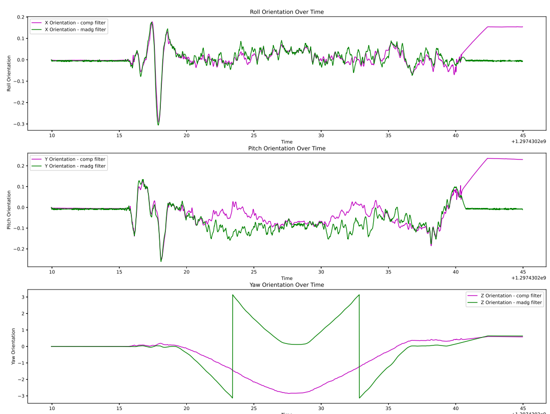

Fig. 19. Madgwick Filter Attitude Estimation for imuRaw8

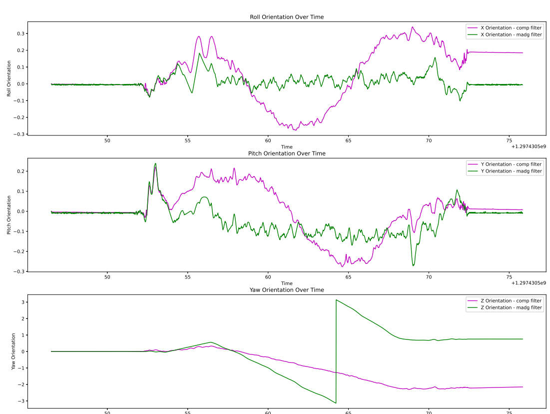

Fig. 20. Madgwick Filter Attitude Estimation for imuRaw9

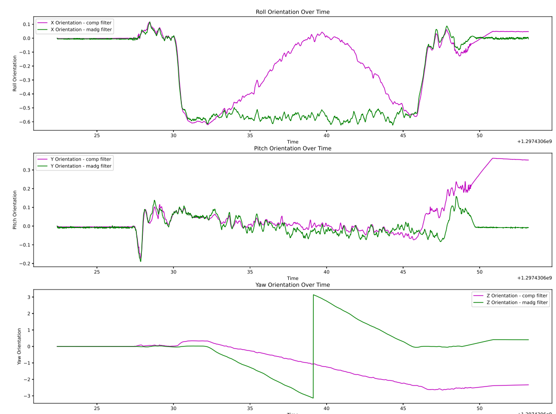

Fig. 21. Madgwick Filter Attitude Estimation for imuRaw10