EE 267 Homework 5

by Jasmin Palmer
partnered with myself

1) Theoretical Part

1.1) Rotation Quarternions

Given: θ , $\vec{v} = [v_x, v_y, v_z]^T$, $\|\vec{v}\| = 1$

The axis-angle to quaternion with normalized axis $\vec{v}$ can be described as:

$$q(\theta, \vec{v}) = \underbrace{\cos(\theta/2)}_{q_w} + i\underbrace{v_x \sin(\theta/2)}_{q_x} + j\underbrace{v_y \sin(\theta/2)}_{q_y} + k\underbrace{v_z \sin(\theta/2)}_{q_z}$$

where i, j, k are different imaginary #s or fundamental quaternion units such that:

$$i \neq j \neq k \quad , \quad i^2 = j^2 = k^2 = ijk = -1$$

$$\therefore \; \|q\| = \sqrt{q_w^2 + q_x^2 + q_y^2 + q_z^2} = 1$$

$\therefore$ The quaternion is unit length

1.2) 3-D Gyro Integration

$f = 1\,Hz$

$$\omega^{(1)} = \begin{bmatrix} \pi/2 \\ 0 \\ 0 \end{bmatrix} \quad \omega^{(2)} = \begin{bmatrix} 0 \\ 0 \\ -\pi/2 \end{bmatrix} \quad \omega^{(3)} = \begin{bmatrix} 0 \\ -\pi/2 \\ 0 \end{bmatrix} \quad \omega^{(4)} = \begin{bmatrix} 0 \\ 0 \\ \pi/2 \end{bmatrix} \quad \text{in } {}^{rad}/_{s}$$

i) $\Delta t = 1/f = 1 \qquad \theta^{(t+\Delta t)} = \theta^{(t)} + \omega^{(t+1)} \Delta t$

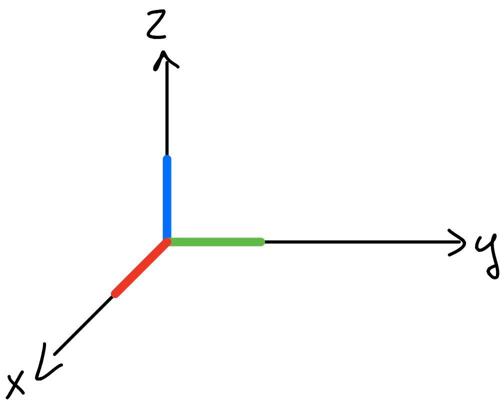
=> MatLab =>

$$\begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix}^{(1)} = \begin{bmatrix} 90^\circ \\ 0^\circ \\ 0^\circ \end{bmatrix} \qquad \begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix}^{(2)} = \begin{bmatrix} 90^\circ \\ 0^\circ \\ -90^\circ \end{bmatrix}$$

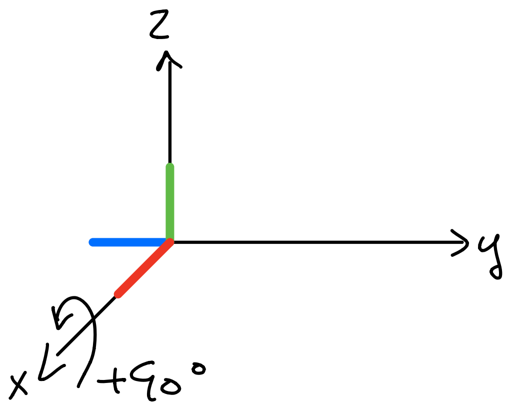
$$\begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix}^{(3)} = \begin{bmatrix} 90^\circ \\ -90^\circ \\ -90^\circ \end{bmatrix} \qquad \begin{bmatrix} \theta_x \\ \theta_y \\ \theta_z \end{bmatrix}^{(4)} = \begin{bmatrix} 90^\circ \\ -90^\circ \\ 0^\circ \end{bmatrix}$$

Step 1: Rotation around x-axis by $+90^\circ$
Step 2: Rotation around z-axis by -90°
Step 3: Rotation around y-axis by -90°
Step 4: Rotation around z-axis by $+90^\circ$

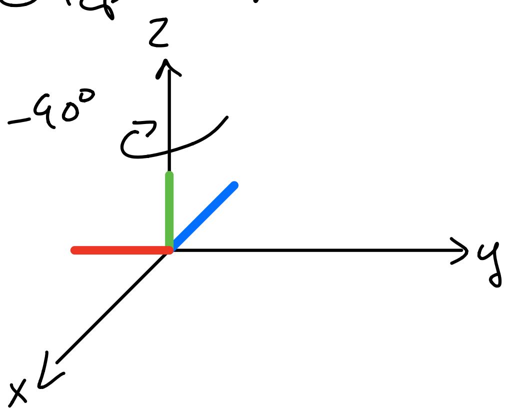
ii) Step 0:



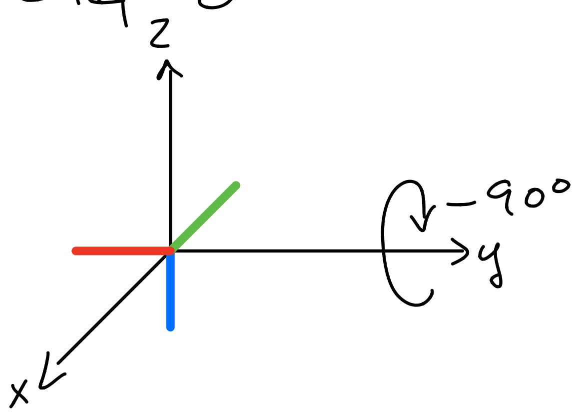
Step 1:



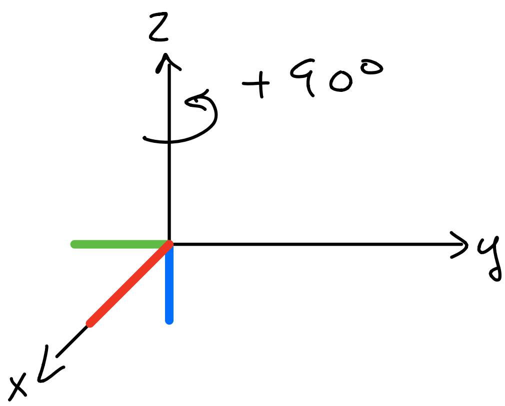
Step 2:



Step 3:



Step 4: Final Orientation



iii) $$\begin{bmatrix}\theta_x\\ \theta_y\\ \theta_z\end{bmatrix}^{(t+\Delta t)} \approx \begin{bmatrix}\theta_x\\ \theta_y\\ \theta_z\end{bmatrix}^{(t)} + \Delta t \begin{bmatrix}\omega_x\\ \omega_y\\ \omega_z\end{bmatrix}^{(t+\Delta t)}$$

$$\theta^{(0)} = [0 \quad 0 \quad 0]^T$$

$\Rightarrow$ MatLab $\Rightarrow$

$$\begin{bmatrix}\theta_x\\ \theta_y\\ \theta_z\end{bmatrix}^{(1)} = \begin{bmatrix}90^\circ\\ 0^\circ\\ 0^\circ\end{bmatrix} \qquad \begin{bmatrix}\theta_x\\ \theta_y\\ \theta_z\end{bmatrix}^{(2)} = \begin{bmatrix}90^\circ\\ 0^\circ\\ -90^\circ\end{bmatrix}$$

$$\begin{bmatrix}\theta_x\\ \theta_y\\ \theta_z\end{bmatrix}^{(3)} = \begin{bmatrix}90^\circ\\ -90^\circ\\ -90^\circ\end{bmatrix} \qquad \begin{bmatrix}\theta_x\\ \theta_y\\ \theta_z\end{bmatrix}^{(4)} = \begin{bmatrix}90^\circ\\ -90^\circ\\ 0^\circ\end{bmatrix}$$

$$\mathbb{R}^{(4)} = \mathbb{R}_z(-\theta_z)\,\mathbb{R}_x(-\theta_x)\,\mathbb{R}_y(-\theta_y)$$

$\Rightarrow$ MatLab $\Rightarrow$

$$\mathbb{R}^4 = \begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & -1 & 0\end{bmatrix}$$

Which does not match my expectation from ii

iv) The order of operations is different and 3D rotations are not commutative $\therefore$ they will not match.

```
% EE 267 HW 5 Problem 1.2 Calculations
clear; close all; clc;


w0 = [0; 0; 0];
w1 = [pi/2; 0; 0]; w2 = [0; 0; -pi/2];
w3 = [0; -pi/2; 0]; w4 = [0; 0; pi/2];

dT = 1;

theta0 = [0; 0; 0];

% Step 1:
theta1 = theta0 +dT*w1;
theta1_deg = rad2deg(theta1)

% Step 2:
theta2 = theta1 +dT*w2;
theta2_deg = rad2deg(theta2)

% Step 3:
theta3 = theta2 +dT*w3;
theta3_deg = rad2deg(theta3)

% Step 4:
theta4 = theta3 +dT*w4;
theta4_deg = rad2deg(theta4)

% iii)
theta4x = theta4(1);
theta4y = theta4(2);
theta4z = theta4(3);
Rx = [1 0 0; 0 cos(-theta4x) sin(-theta4x); 0 sin(-theta4x) cos(-
theta4x) ]
Ry = [cos(-theta4y) 0 sin(-theta4y); 0 1 0; -sin(-theta4y) 0 cos(-
theta4y)]
Rz = [cos(-theta4z) -sin(-theta4z) 0; sin(-theta4z) cos(-theta4z) 0; 0
 0 1]

R4 = Rz*Rx*Ry
```

```
theta1_deg =

    90
     0
     0


theta2_deg =

    90
```

```
     0
   -90


theta3_deg =

    90
   -90
   -90


theta4_deg =

    90
   -90
     0


Rx =

    1.0000         0         0
         0    0.0000   -1.0000
         0   -1.0000    0.0000


Ry =

    0.0000         0    1.0000
         0    1.0000         0
   -1.0000         0    0.0000


Rz =

     1     0     0
     0     1     0
     0     0     1


R4 =

    0.0000         0    1.0000
    1.0000    0.0000   -0.0000
   -0.0000   -1.0000    0.0000
```

Published with MATLAB® R2020a

2.1.1) Bias Estimation

Calibrated Bias Values:

GyrX: -0.47761
GyrY: 2.36250
GyrZ: 1.66289
AccX: 0.517
AccY: 0.141
AccZ: 9.698

If these sensors were perfect we would expect

GyrX: 0
GyrY: 0
GyrZ: 0
AccX: 0
AccY: 0
AccZ: 9.81 (1 g)

2.1.2) Noise Variance Estimation

Estimated Variance Values:

GyrX: 0.01028

GyrY: 0.00978

GyrZ: 0.00960

AccX: 0.000

AccY: 0.000

AccZ: 0.001

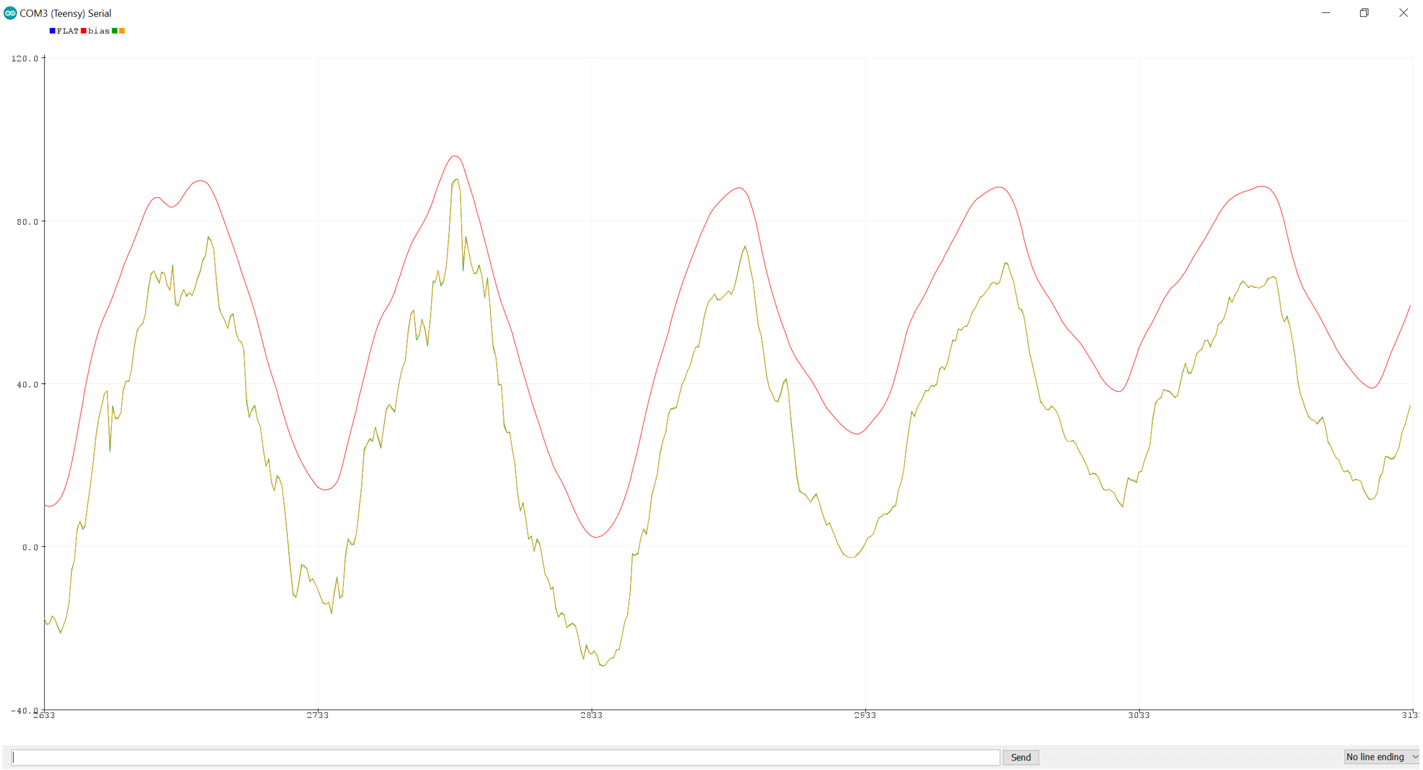
2.2.4) Flatland Orientation Tracking Comparison

Key:

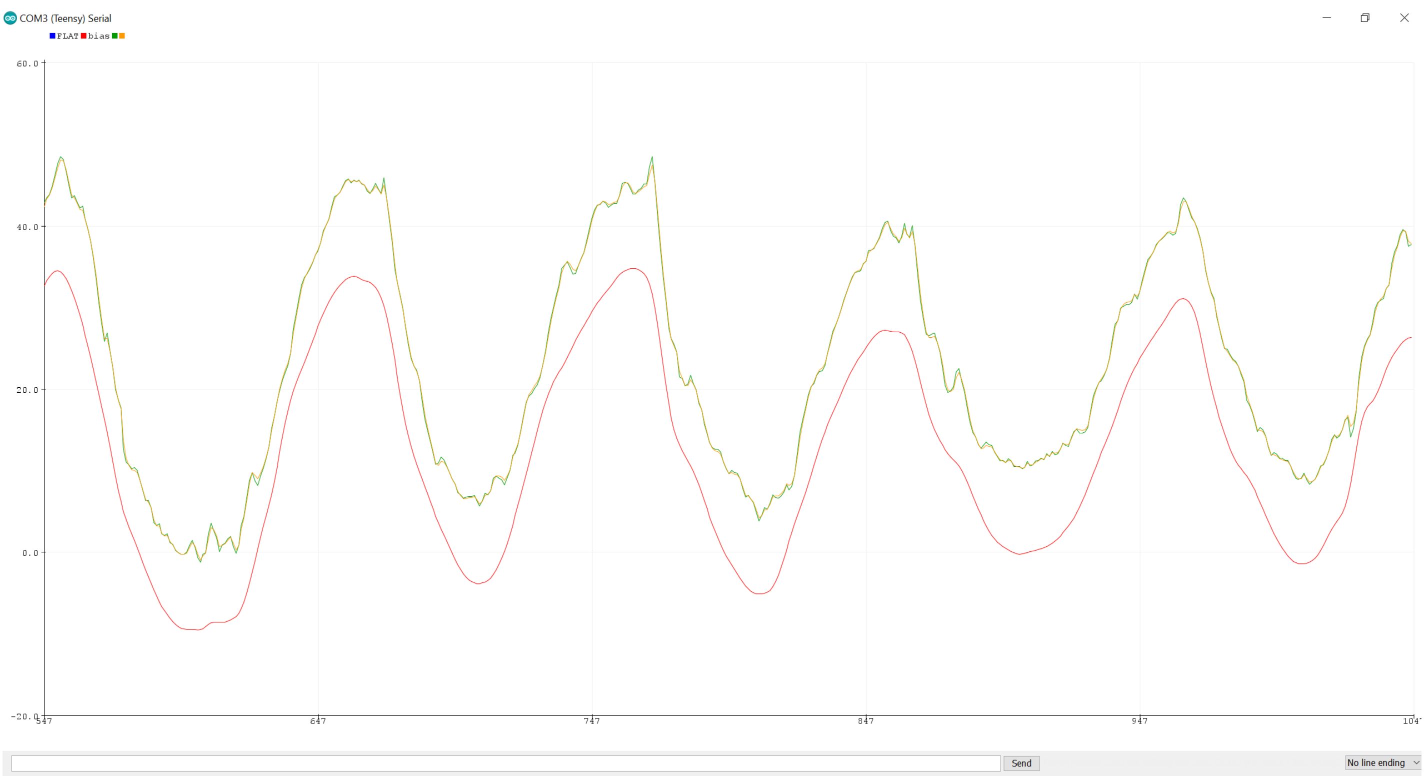
Gyro Only

Acc Only

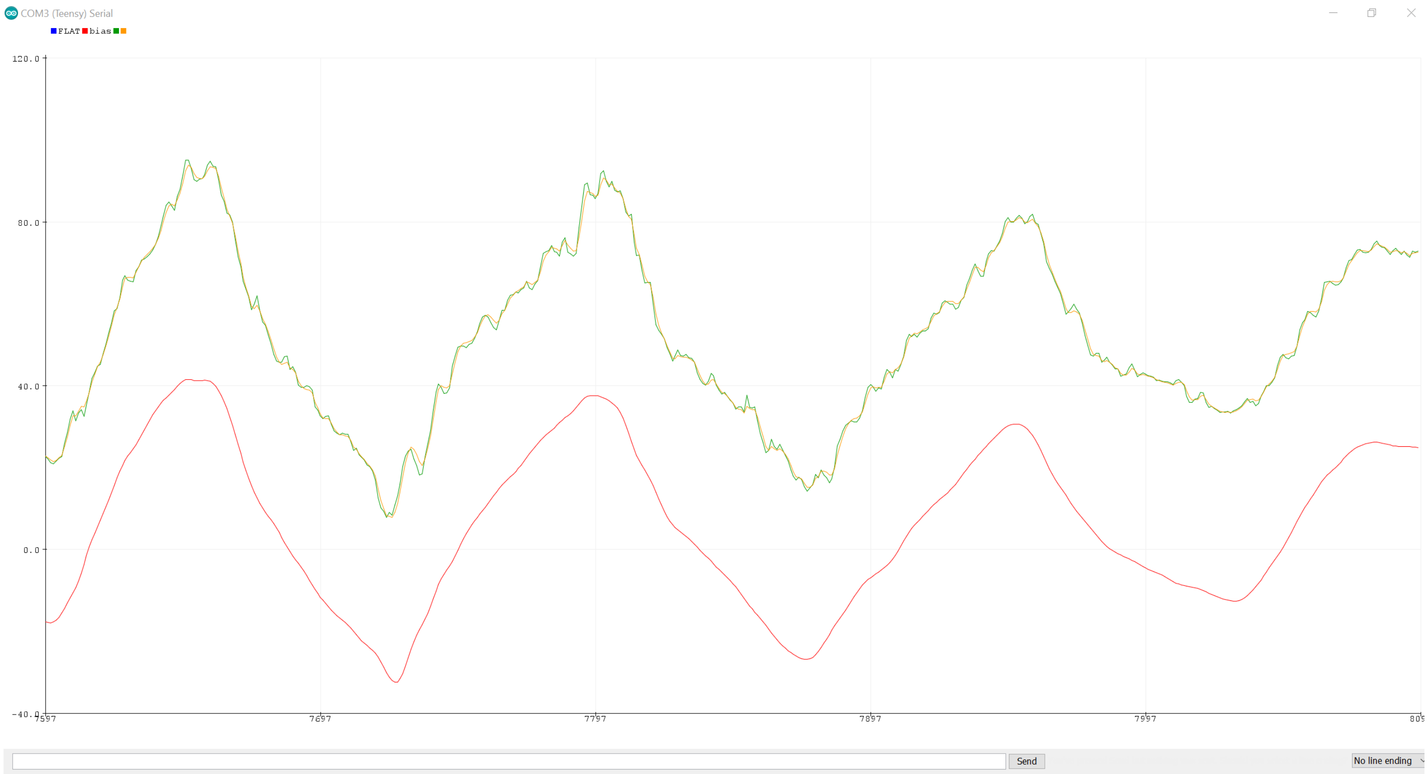
Comp Filter



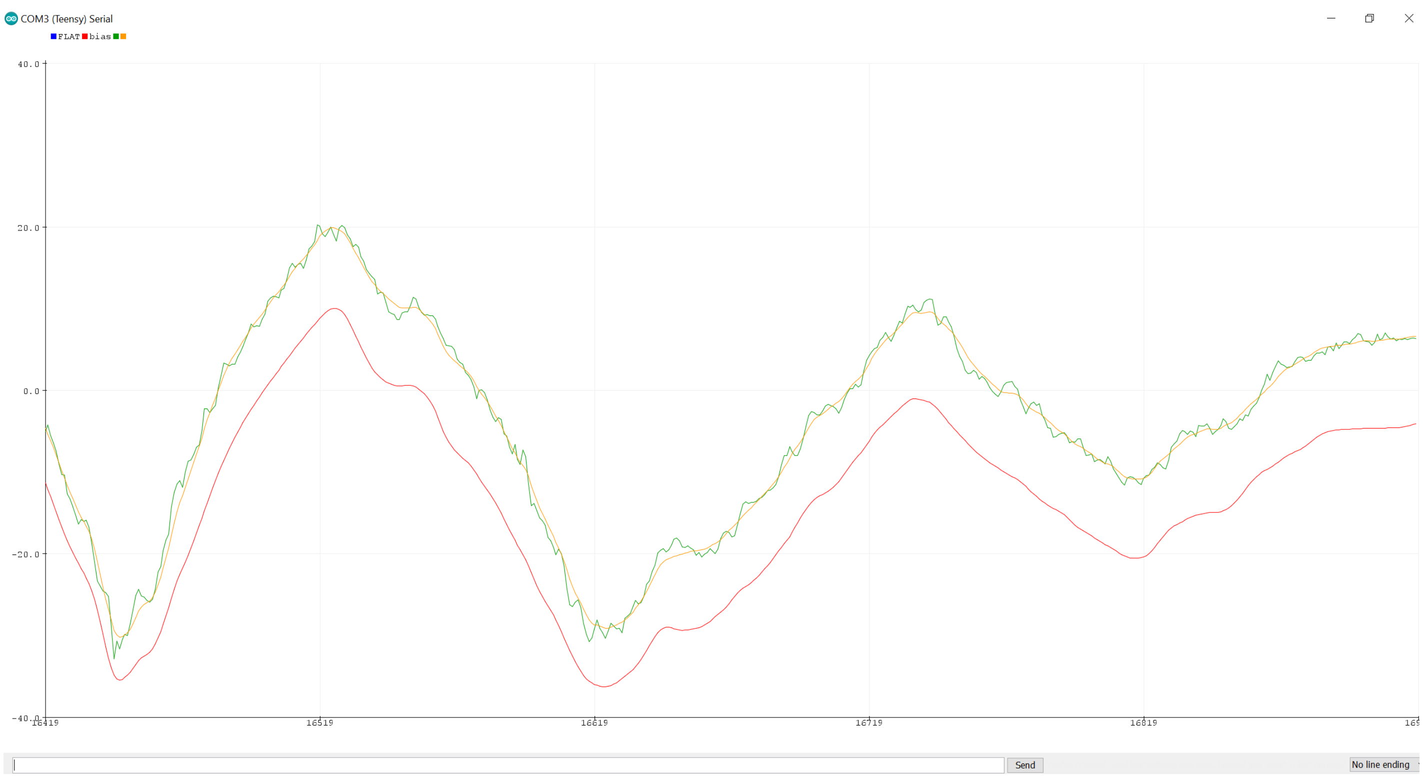
alpha = 0.1



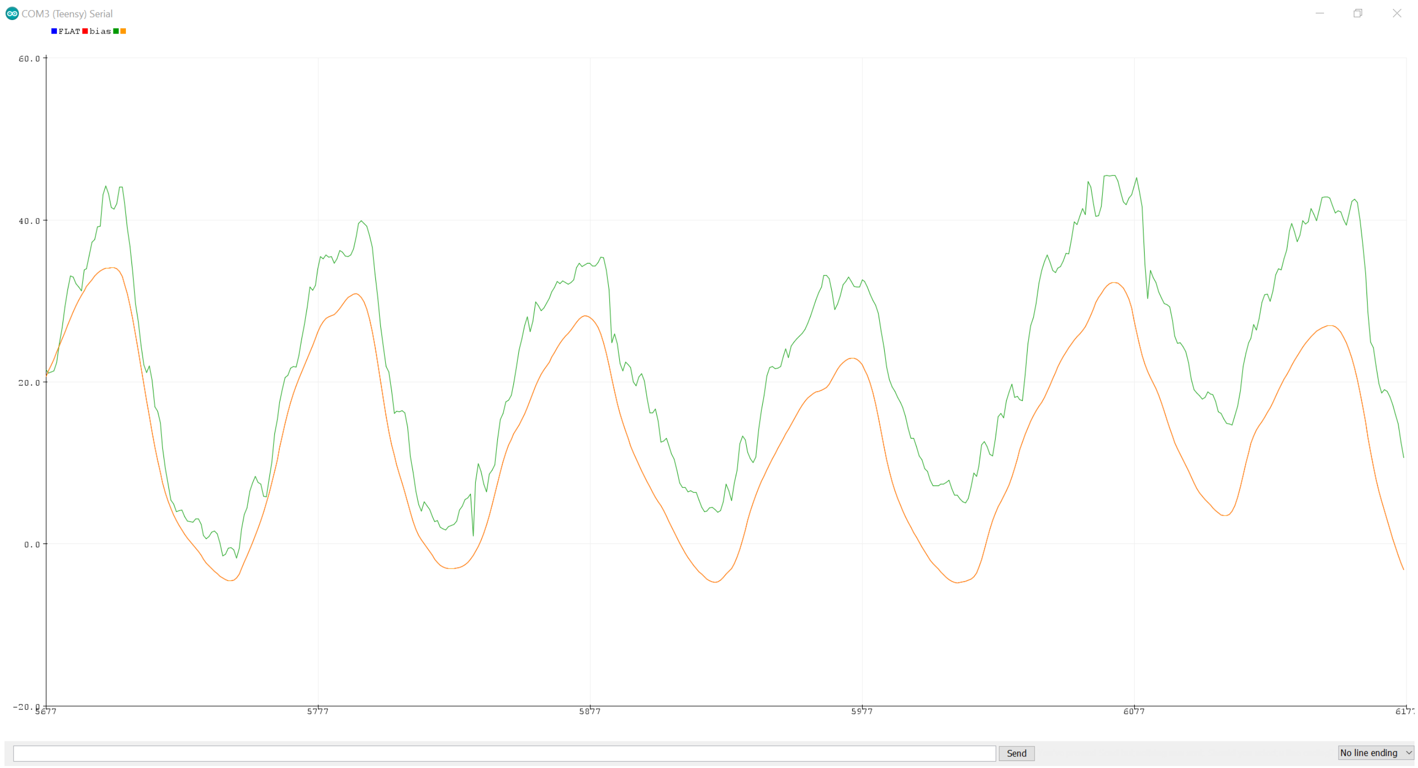
alpha = 0.3



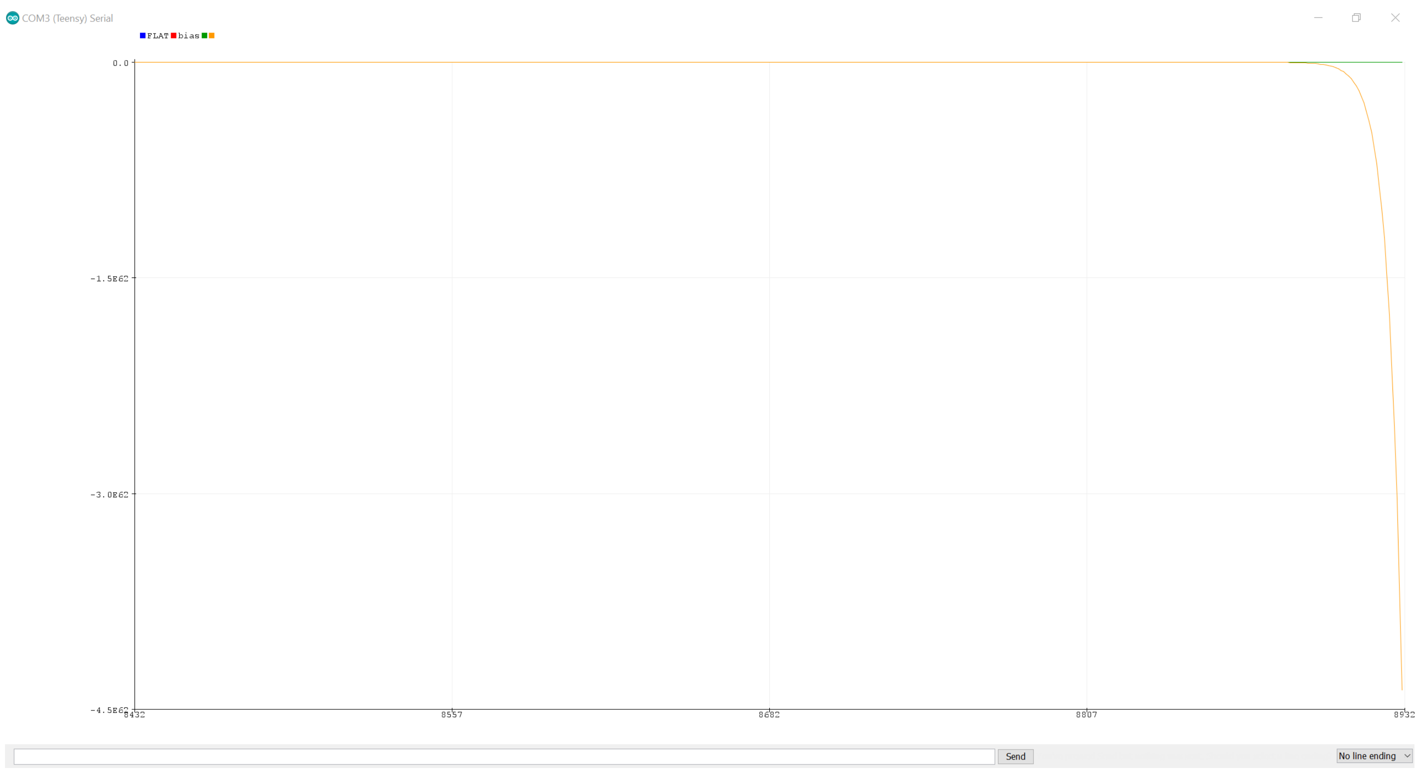
alpha = 0.5



Default: alpha = 0.9



alpha = 1.0



alpha = 1.1

Green and orange overlap more for lower alpha.

Red and orange overlap more for higher alpha until it becomes unstable at alpha = 1.1

2.4.4) Quaternion-Based Orientation Tracking Algorithm Comparison

Case 1: Quaternions w/ gyro only
vs Quaternions w/ complementary filtering
$\alpha = 0.9$

- More reactive and truer to observed orientation in the real world

Case 2: Euler angles w/ accelerometer only
vs Quaternions w/ complimentary filtering
$\alpha = 0$

- A lot more lag and less true to real world orientation.

2.5.3) Head & Neck Model Discussion

The addition of the head and neck kinematic constraints helped the image appear more stable and "grounded", as if I was actually looking into a room. Without the constraints, the scene was unstable and nauseating to look at.