

# 1 Theoretical Part

## 1.1 Rotation Quaternions

The rotation quaternion with the rotation angle as  $\theta$  and rotation axis  $\mathbf{v} = (v_x, v_y, v_z)^T$  is given as:

$$q(\theta, v) = \cos(\theta/2) + iv_x \sin(\theta/2) + jv_y \sin(\theta/2) + kv_z \sin(\theta/2). \quad (1)$$

The length of  $q$  is

$$|q| = \sqrt{\cos^2(\theta/2) + (v_x^2 + v_y^2 + v_z^2) \sin^2(\theta/2)} = \cos^2(\theta/2) + \sin^2(\theta/2) = 1.$$

## 1.2 3-D Gyro Integration

(i)

The refresh rate is 1 Hz so the time step is given as 1 s. The angle of rotation is given as  $\|\omega\|\Delta t$  and the rotation axis is  $\frac{\omega}{\|\omega\|}$ . Therefore, the rotation axis and amount of rotation for each 1, 2, 3, 4 is:

$$\begin{aligned} \|\theta^{(1)}\| &= 90^\circ, \text{ axis}^{(1)} = (1, 0, 0)^T \\ \|\theta^{(2)}\| &= -90^\circ, \text{ axis}^{(2)} = (0, 0, 1)^T \\ \|\theta^{(3)}\| &= -90^\circ, \text{ axis}^{(3)} = (0, 1, 0)^T \\ \|\theta^{(4)}\| &= 90^\circ, \text{ axis}^{(4)} = (0, 0, 1)^T \end{aligned}$$

Or,

1. Rotate around x-axis by  $90^\circ$ ;
2. Rotate around z-axis by  $-90^\circ$ ;
3. Rotate around y-axis by  $-90^\circ$ ;
4. Rotate around z-axis by  $90^\circ$ ;

(ii)

Following the procedure summarized in (i), we get the full rotation as:

$$R_{tot} = R_z(\pi/2)R_y(-\pi/2)R_z(-\pi/2)R_x(\pi/2) \quad (2)$$

$$= \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \quad (3)$$

$$= \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \quad (4)$$

So the sequence of operation is equivalent of getting the inverse of the y- and z- coordinates.

(iii)

Following the first-order Taylor series, the angle at each time point is:

$$\theta^{(0)} = (0, 0, 0)^T \rightarrow \theta^{(1)} = (\frac{\pi}{2}, 0, 0)^T \rightarrow \theta^{(2)} = (\frac{\pi}{2}, 0, -\frac{\pi}{2})^T \rightarrow \theta^{(3)} = (\frac{\pi}{2}, -\frac{\pi}{2}, -\frac{\pi}{2})^T \rightarrow \theta^{(4)} = (\frac{\pi}{2}, -\frac{\pi}{2}, 0)^T$$

To verify, the total rotation for Euler angle is given as:

$$R = R_z(\theta_z)R_x(-\theta_x)R_y(-\theta_y). \quad (5)$$

Therefore the rotation matrix for the Euler angle for each step is:

$$R^{(1)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} \quad (6)$$

$$R^{(2)} = \begin{pmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix} \quad (7)$$

$$R^{(3)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} \quad (8)$$

$$R^{(4)} = \begin{pmatrix} 0 & 0 & -1 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \quad (9)$$

The resulting R4 is not the same as the rotation matrix in (ii).

(iv)

Due to the non-commute nature of rotation operation, the actual procedure matters for rotation. The effect of a sequence of rotation cannot be equivalently represented as a full rotation given as the final outcome angle (Euler angle). From the operational perspective, the Euler angle is ill-defined when the vector is along certain axis around which the rotation happens, introducing ambiguity.

## 2 Programming Part

### 2.1 Noise and Bias Estimation

#### 2.1.1 Bias Estimation

The reported bias values:

	$x-$	$y-$	$z-$
gyr_bias	-3.57494	2.23188	-0.10047
acc_bias	1.059	0.719	8.899

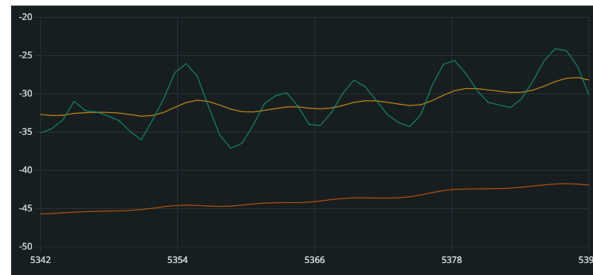
#### 2.1.2 Noise Variance Estimation

The reported variance values:

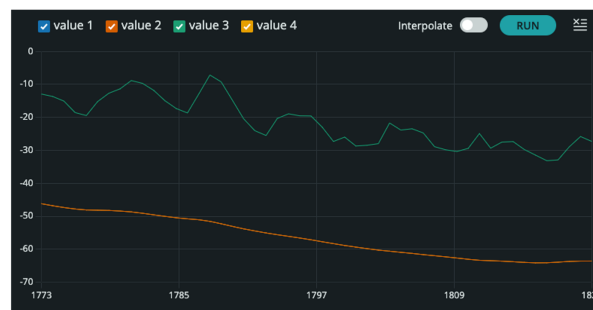
	$x-$	$y-$	$z-$
gyr_var	0.01021	0.01014	0.01017
acc_var	0.000	0.000	0.001

### 2.2.4 Flatland Orientation Tracking Comparison

(a)  $\alpha=0.9$



(b)  $\alpha=1.0$



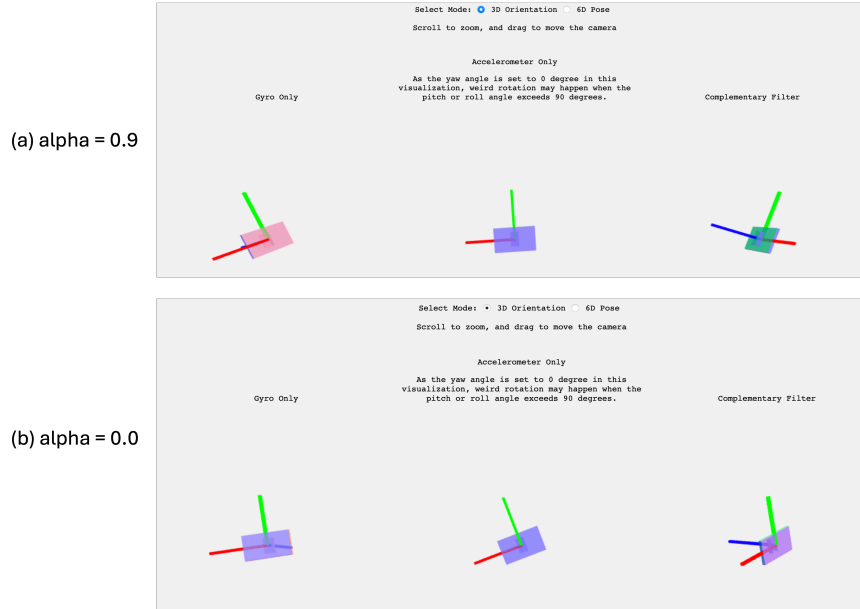
(c)  $\alpha=0.5$



Description: The gyro-only curve is not noisy but has an offset from the groundtruth. The acceleration-only curve is noisy. The complementary filter resulting in low-noise result without

global bias. Comparing the cases with different alpha values, for very small alpha, the complementary curve converges to the acceleration curve with large noise, while for large alpha, the complementary curve converges to the gyro-only curve with a bias. An intermediate level of alpha mixing factor can give a good result.

#### 2.4.4 Quaternion-Based Orientation Tracking Algorithm Comparison



Description: For the first case with  $\alpha = 0.9$ , the comparison between the gyro-only motion and the motion given by the complementary filter shows that they follow similar rotational motion. However there is a bias (offset) in between. For the second case (b) with  $\alpha = 0$ , comparing acc-only motion with the motion given by the complementary filter shows that the z-axis follows the same orientation. However, due to the undefinable y-rotation in Euler angle, the accelerometer-only case cannot respond to y-rotation. On the other hand the complementary filter enables full rotation. Besides both cases show large noise.

#### 2.5.3 Head & Neck Model Discussion

The actual motion parallax is actually quite obvious. For the case with head and neck model, when rotating head, the perspective changes a little bit as well, while it is not shown in the case without head and neck modelling. This helps to give a more realistic rendering scene!