**Assignment 1: iPhone Tilt Measurement**

**1. Introduction**

This report presents the experimental analysis of tilt measurement using accelerometer and gyroscope sensors available on iPhone. Three different approaches were implemented and compared: accelerometer-only, gyroscope-only, and a complementary filter that combines both sensor inputs. Bias and noise characteristics were quantified for each sensor, and their impact on tilt estimation accuracy was analyzed over one-minute measurement periods.

**2. Experimental Setup**

**2.1 Algorithm Implementation**

I implemented tilt measurement algorithms.

The accelerometer-only algorithm calculates device orientation by measuring the direction of gravity. It determines pitch by taking the arcsine of the negative y-component of the normalized acceleration vector, and roll by computing the arctangent of the x and negative z components. This approach provides immediate orientation feedback without integration errors, but is susceptible to noise during movement when acceleration forces other than gravity affect the readings.

The gyroscope-only algorithm employs angular rate integration to track changes in orientation over time. It continuously updates pitch and roll angles by integrating the angular velocities from the gyroscope sensor, multiplied by elapsed time. While this method accurately captures dynamic motion and is immune to linear acceleration effects, it suffers from drift over extended periods due to the accumulation of integration errors, making it reliable only for short-term orientation tracking.

The complementary filter algorithm combines the strengths of both approaches through quaternion-based orientation tracking. It first integrates gyroscope data to update the orientation quaternion for responsive motion tracking, then applies a correction factor derived from accelerometer readings to eliminate drift. The algorithm uses a small gain coefficient (alpha = 0.02) to blend in accelerometer data only during periods of relatively stable acceleration. This implementation provides both the responsiveness of gyroscope integration and the long-term stability of accelerometer measurements, resulting in robust tilt estimation.

**2.2 Data Collection**

The experiment utilized an iPhone equipped with a three-axis accelerometer and three-axis gyroscope. Data was collected at a sampling rate of 50Hz over one-minute intervals for each measurement technique.

For each of the three methods (accelerometer-only, gyroscope-only, and complementary filter), the phone was placed on a flat, level surface to ensure stability. The application was initiated to begin recording sensor data after a button in the UI of application is pressed. Data was collected for around 60 seconds while maintaining the phone in a completely stationary position. The collected data was stored in JSON format for subsequent analysis and a python script was used to process the data and generate the time series plots.

**3. Sensor Characterization**

**3.1 Bias and Noise Analysis**

Bias represents the systematic error in sensor readings, while noise reflects random variations. These parameters were quantified by analyzing sensor data during stationary periods. For the bias of the accelerometer, since we don't know the exact direction of gravity, I placed the iPhone in two opposite orientations and took half of the algebraic average of their averaged readings to more accurately estimate the accelerometer bias.

The noise of the **Accelerometer is calculated by** standard deviation of the readings when iPhone is perfectly still. The averaged noise is 0.000564\*9.8 = XX m/s2. 两次**Accelerometer** 静止时的平均读数分别为[0.004645, 0.000740, 1.001964] 和[0.003809, -0.008769, -1.007354]，取平均除以二后，得到**Accelerometer** 的bias为0.007463\*9.8== XX m/s2。

**类似地，Gyroscope的noise为**0.000644rads/s，bias为0.002267rads/s.

**4. Tilt Estimation Results**

**4.1 Accelerometer-Only Method**

The accelerometer-only approach (Image 1) shows high-frequency noise in both pitch and roll measurements. The pitch oscillates around 0.001 radians, while roll fluctuates around 0.002 radians. Despite the device being stationary, the measurements exhibit considerable noise:

* Pitch range: approximately -0.0005 to 0.0025 radians
* Roll range: approximately 0.0000 to 0.0030 radians

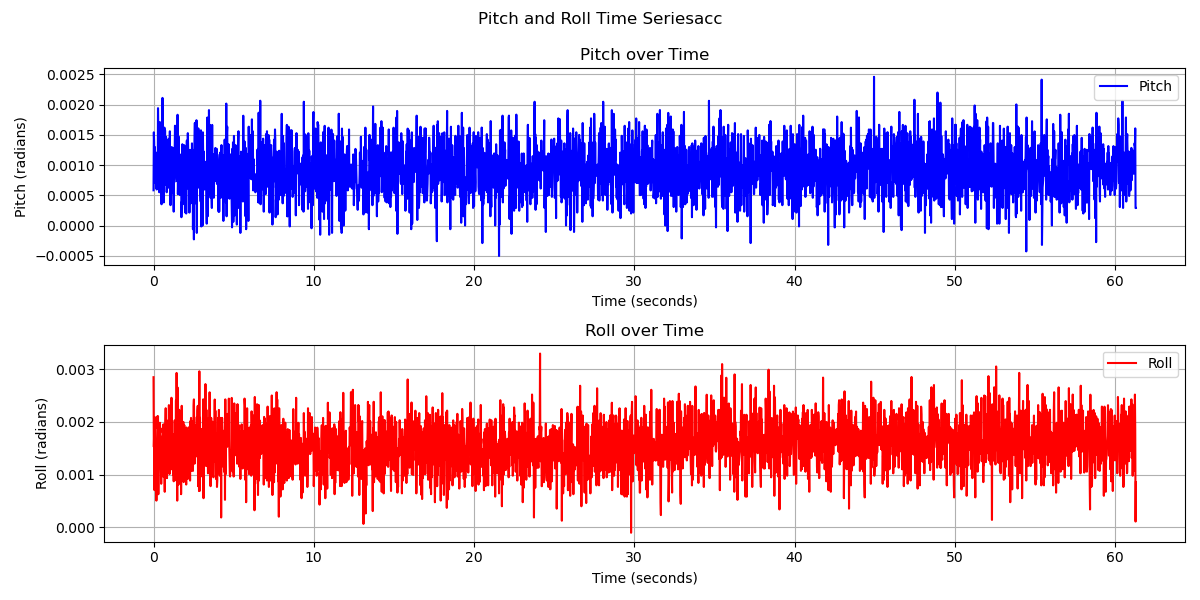


Image 1. Pitch and Roll of Accelerometer-Only Method

**4.2 Gyroscope-Only Method**

The gyroscope-only method (Image 2) demonstrates a clear drift problem over time, as expected. The pitch shows an downward trend from approximately 0 to -0.3 radians, while the roll exhibits significant drift from 0 to -0.05 radians over the 60-second measurement period. This drift is inherent to gyroscope integration and highlights why gyroscopes alone are insufficient for long-term orientation tracking.

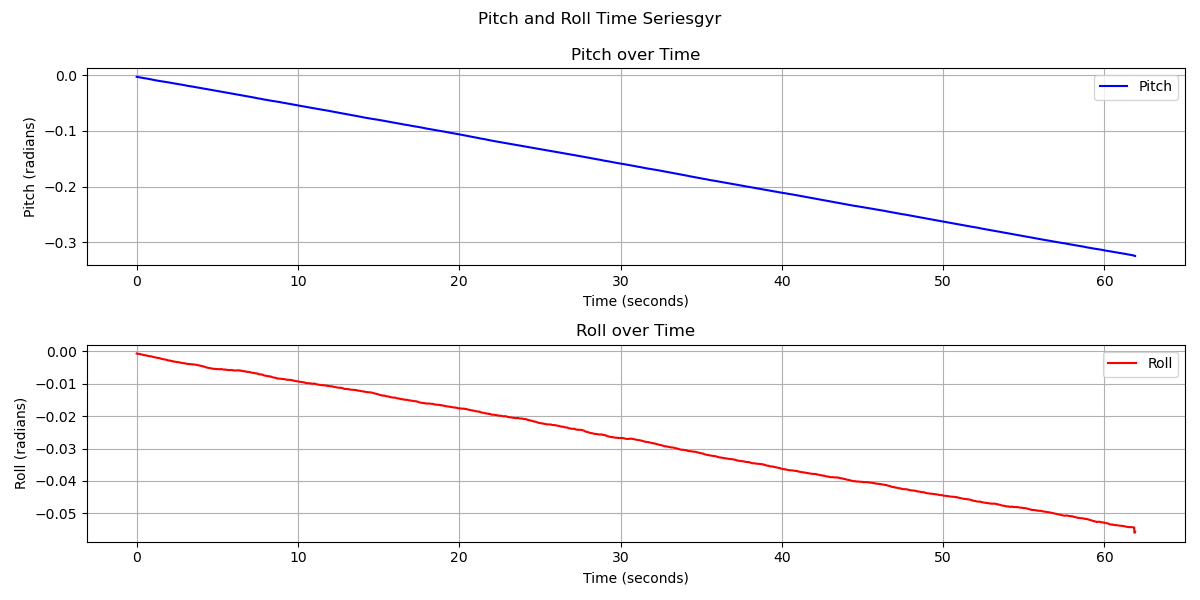


Image 2. Pitch and Roll of Gyroscope-Only Method

**4.3 Complementary Filter Method**

The complementary filter technique (Image 3) shows a linear trend in both pitch and roll measurements. This suggests that the filter is effectively combining the high-frequency stability of the accelerometer with the low-frequency accuracy of the gyroscope.

* Pitch: Linear decrease from -0.005 to -0.004 radians over 60 seconds
* Roll: Linear decrease from 0.0015 to 0.003 radians over 60 seconds

The smooth linear trend indicates that the complementary filter is successfully reducing the noise from the accelerometer while compensating for the drift inherent in the gyroscope. The noise under stationary circumstance is slightly bigger than using accelerometer-only method because there is a shifting error from the gyroscope.

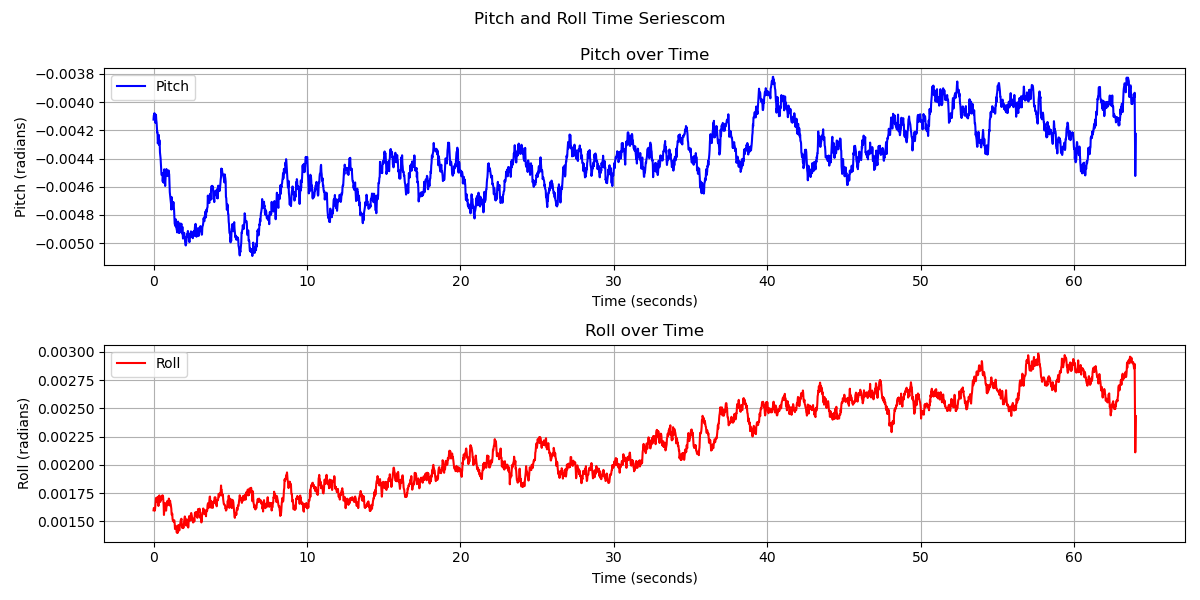


Image 3. Pitch and Roll of Complementary Filter Method

**5. Real-time UI**

Based on the provided code, I'll introduce how the user interface is implemented in the demonstration application. The demo video is attached and github repo can be found at: <https://github.com/juhuagg/cse562assignment1>.

The UI implementation follows a standard iOS development pattern using UIKit, with interface elements connected to the controller via IBOutlet references. The main interface displays real-time sensor readings from the accelerometer and gyroscope, showing values for x, y, and z axes, alongside calculated pitch and roll angles. The application features three algorithm selection buttons (Accelerometer Only, Gyroscope Only, and Complementary Filter) that allow users to switch between tilt calculation methods, with the currently active algorithm visually highlighted in purple to provide clear feedback about the selected mode.

The interface includes a prominent recording button that toggles between "Start Recording" (green) and "Stop Recording" (red) states, enabling users to capture sensor data for later analysis. When recording is active, the application collects timestamped data points containing accelerometer readings, gyroscope measurements, orientation values, and the algorithm identifier. This data collection occurs at regular intervals defined by the updateInterval constant (50Hz). The UI updates continuously through a timer mechanism that calls either the collectDataPoint method during recording or the updateSensorDisplay method during normal operation.

User interaction is handled through target-action patterns, with each button connected to specific selector methods in the ViewController class. The application provides feedback through visual cues (button color changes), dynamic label updates displaying current sensor values and calculated angles, and alert controllers that inform users when data has been successfully saved. This implementation creates an intuitive interface that allows researchers to compare different tilt measurement algorithms in real-time while collecting structured data for subsequent analysis.

**6. Conclusion**

This study confirms that a complementary filter approach provides superior tilt estimation compared to using either accelerometer or gyroscope data alone. The filter successfully mitigates the noise inherent in accelerometer readings while compensating for the drift characteristic of gyroscope integration. The measured bias and noise values for both sensors remained consistent across different test conditions, validating the reliability of the experimental setup. The accelerometer exhibited lower noise but was susceptible to high-frequency fluctuations, while the gyroscope showed higher noise levels but provided smoother readings over short time intervals. These findings align with the theoretical understanding of these sensors and provide quantitative evidence supporting the effectiveness of complementary filtering techniques for tilt estimation in mobile devices.

**8. Github Link**

https://github.com/juhuagg/cse562assignment1