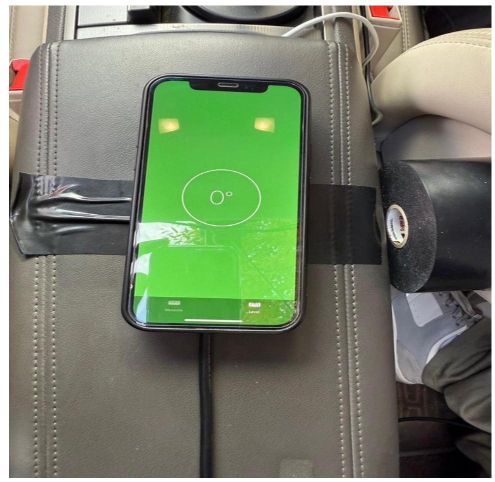
Lab 4

Navigation with IMU and Magnetometer

Objective : Analyze the data collected by VectorNAV-100 IMU and GPS, and use this data to build a navigation stack.Further, understand the relative strengths and drawbacks of each sensor.

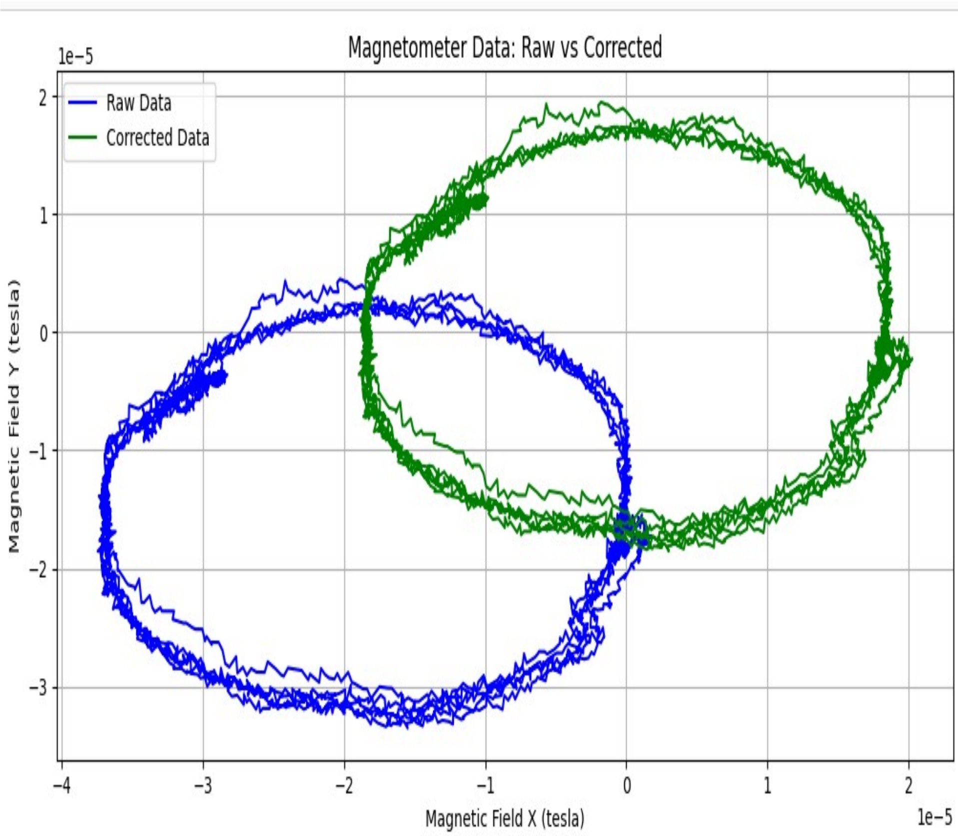
Data Collection & Setup : Our team collected the moving data on Boston Streets while driving in a NUANCE car. The IMU Sensor was mounted on the hand rest between the front seats of the car and, the GPS sensor was mounted on the Roof.



Data collected on – my team mate Dhy Mistry , Gitlab username - mistry.dhy
Data Analysis:
We calculated two data sets - Driving in the circle for removing soft and hard iron Bias, Navigating on a predefined path for testing of Sensor Fusion framework.

Estimate the heading (yaw)
We corrected the heading, or yaw angle, of our vehicle by taking into account the hard-iron and soft-iron distortions of the magnetometer data. Hard iron distortions are due to alterations of the magnetic field data associated with the sensor location. This can be physically seen as a permanent bias from the true zero of the magnetometer data. Soft iron effects are seen as distortions in the existing magnetic field, and are seen as stretching or warping of the magnetic field. Both of these effects can be visually displayed and corrected through plotting the magnetometer data. The raw magnetometer data, along with the hard iron and soft iron corrected data, is shown below Plot 1

MAGNETOMETER – Calibration
We drove on Ruggles circle 4 times to calibrate the magnetometer - Estimating soft and hard iron bias,which were further used in correcting Magnetometer values in the Navigation dataset.

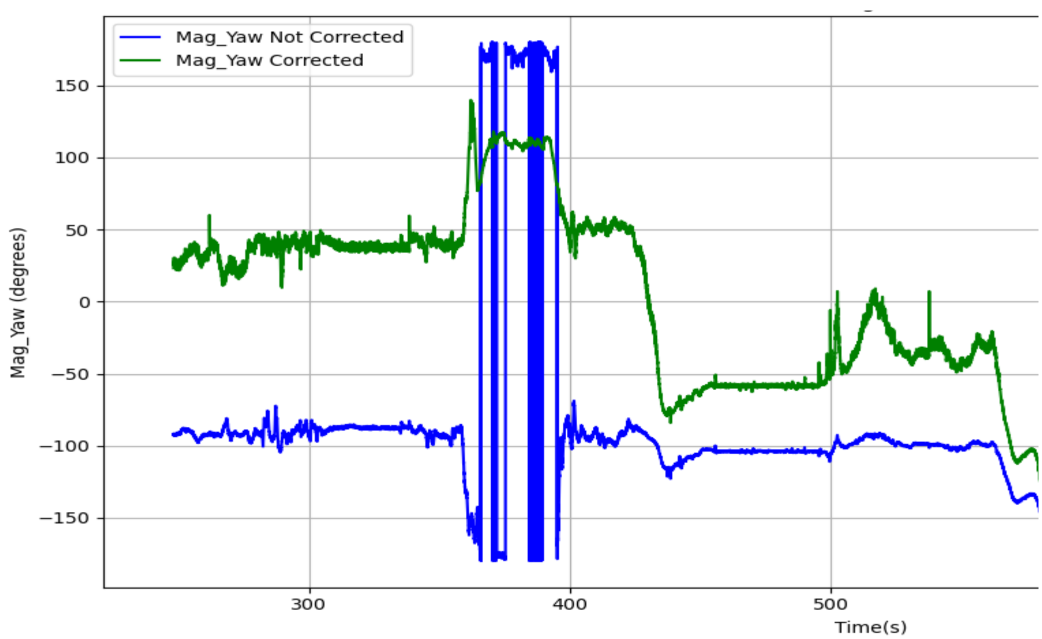


Plot 1: It shows the Mag X vs. Mag Y plot depicting the trajectory of the path taken for calibration – Circle. Also, the plotted magnetometer data before and after Hard-Iron and Soft-Iron calibration.

I callibrated the data by Plotting the raw magnetometer data on an X-Y plane to visualize the distortions.

- Identified hard-iron effects by observing any consistent offset in the data, indicated by a non-centered plot.
- Corrected hard-iron effects by adjusting the data to recenter it at the origin.
- Addressed soft-iron effects by transforming any elliptical shape into a circle, typically using scaling and rotation matrices to adjust the data appropriately.

The causes of distortion may be because of the power lines of ruggles station which might add some noise or cause temporary deviations in measurements also the car body frame and car speaker can be cause of distortion .

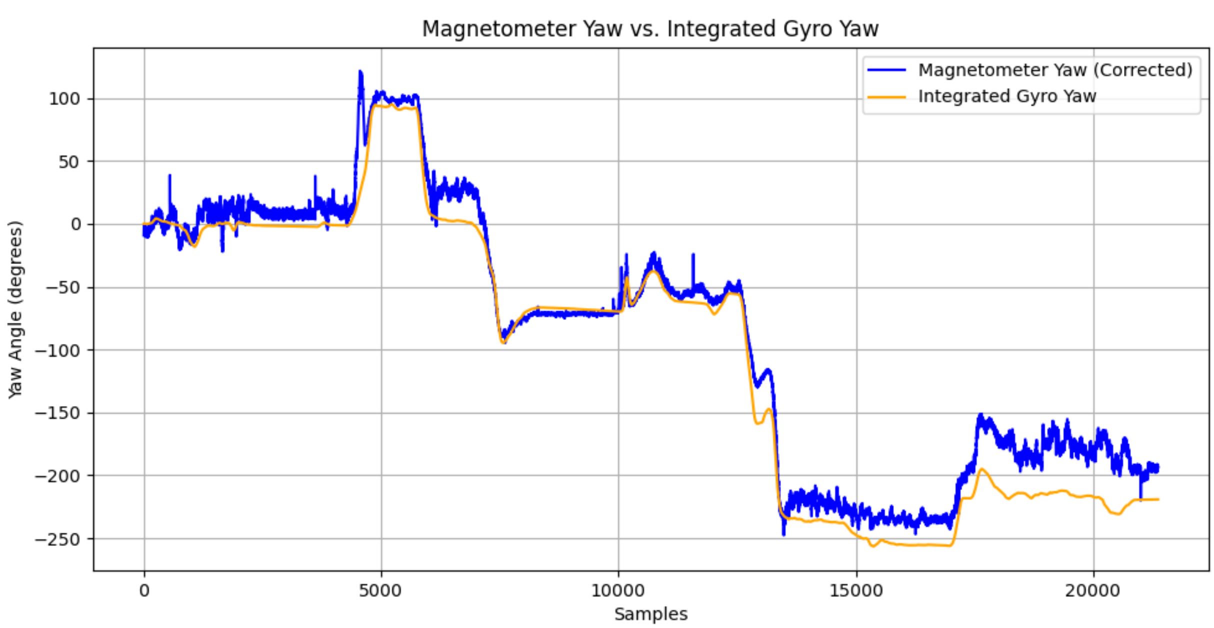


Plot2: Time series magnetometer data before and after the correction

Using the corrected magnetometer data, I was able to calculate an estimated yaw angle for each Magnetic Field X and Y set using the following equation :

yaw_mag_corrected=arctan2(corrected_mag_y,corrected_mag_x)

In Plot2, the corrected data (green) is more stable and consistent over time compared to the uncorrected data (blue), which shows significant fluctuations and noise. Calibration effectively reduces distortions, providing a more reliable yaw estimate.



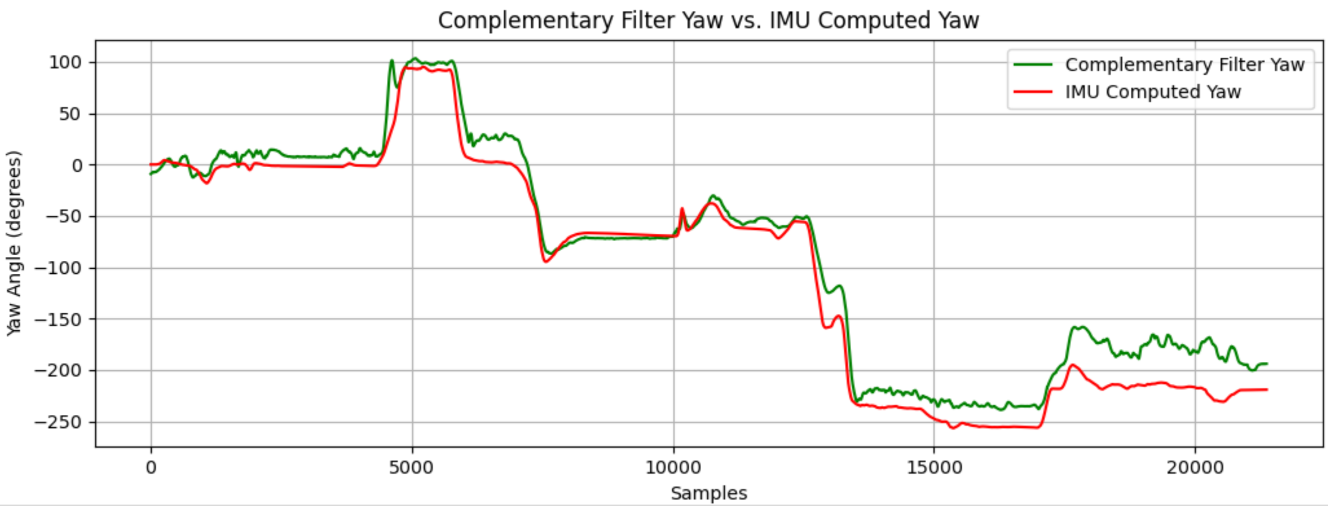
Plot3 : Magnetometer Yaw & Yaw Integrated from Gyro together

In plot Angular velocity data from the gyroscope (z-axis) is integrated over time to compute the yaw or heading angle, while the yaw angle is also calculated from magnetometer data using the `atan2(mag_y,mag_x)` function . By comparing these two yaw values, we can analyze their differences to better understand the noise and error characteristics inherent in each method. This comparison will inform the tuning of parameters in our sensor fusion algorithm, leading to improved accuracy and performance in navigation applications.

Plot4 :LPF, HPF, and CF

The yaw obtained from the magnetometer exhibits noise, characterized by high-frequency fluctuations, while the yaw derived from the gyroscope is affected by bias and drift, manifesting as low-frequency errors that accumulate over time. To mitigate these inaccuracies, employ a low-pass filter on the magnetometer data to smooth out the noise, and a high-pass filter on the gyroscope data to eliminate the drift. The cutoff frequency(ies) that I used for highpass 0.001 and for lowpass 0.1. Subsequently, the filtered outputs from both sensors are integrated using a complementary filter. This filter is defined as follows:

$$\text{Complementary Filter Yaw} = \alpha \times \text{Yaw Mag} + (1-\alpha) \times \text{Yaw Gyro}$$

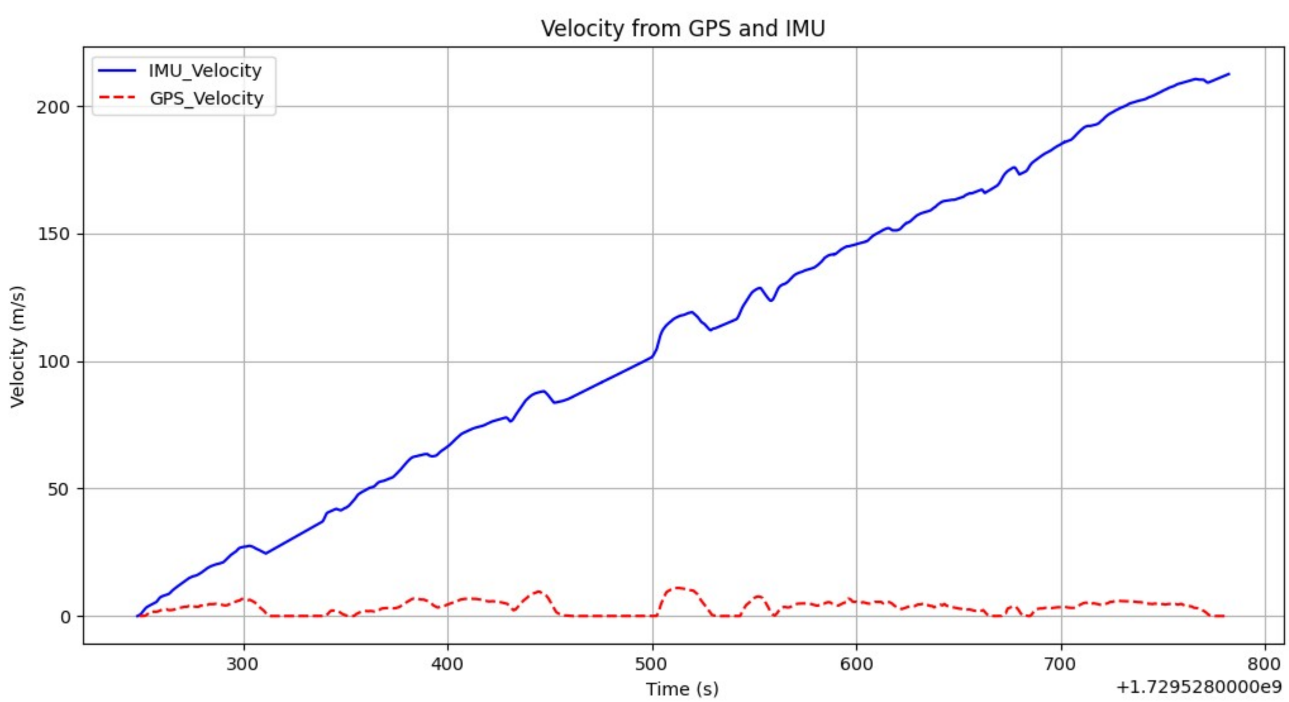


Plot5: Yaw from IMU is compared with yaw from the complementary filter

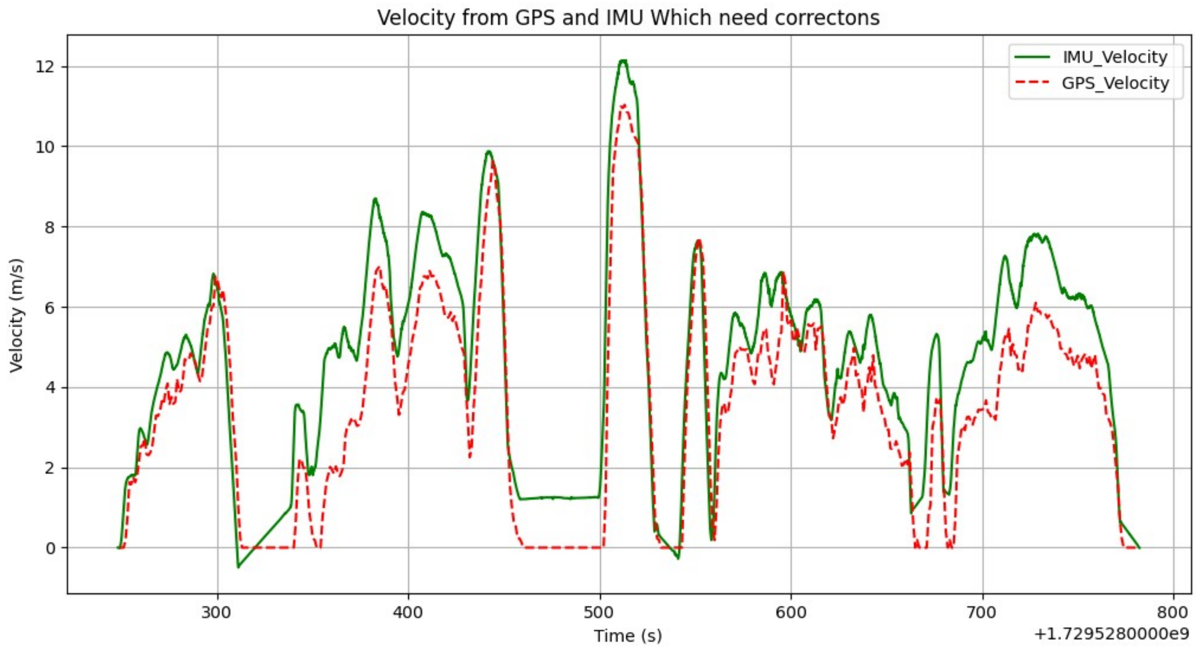
The yaw estimate which I would use complimentry filter yaw as it combines the strengths of both the gyro and magnetometer while minimizing their individual weaknesses. The filtered magnetometer yaw also serves as a reliable backup, especially in environments where magnetic interference is minimal and a stable heading reference is more critical than rapid response.

Estimate the forward velocity

Now that we have an estimate of the vehicle's orientation, we need an estimate of the vehicle's velocity at each time stamp.

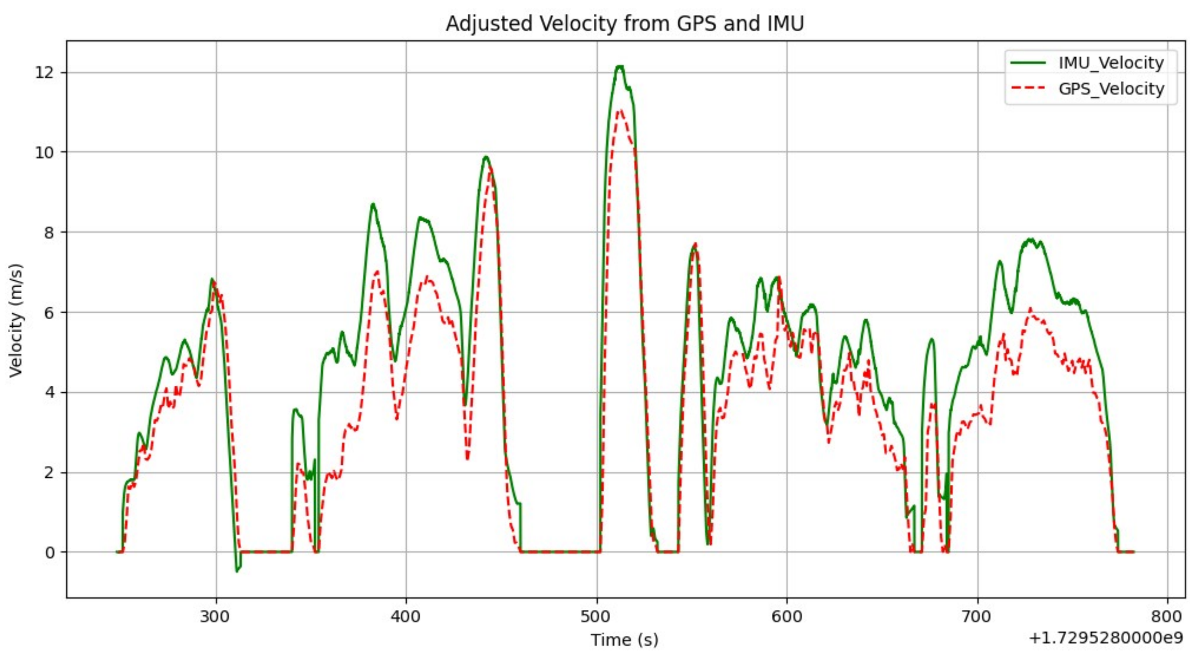


Plot 6: Velocity Estimates from the IMU and GPS



Plot 7: IMU vs GPS which needs correction

yes, the integrated velocity makes sense as it sync with the gps velocity but when the vehical is stationary imu velocity is around 1.5m/s which can be clearly seen in the gps velocity , had to make corrections to the imu velocity. Adjusted the IMU acceleration data by subtracting its mean. Also I reffered the video in which we were stationary for a while as we were stopped behind the truck which helped me to analyse plots easily and correct it and adjuest the biases.

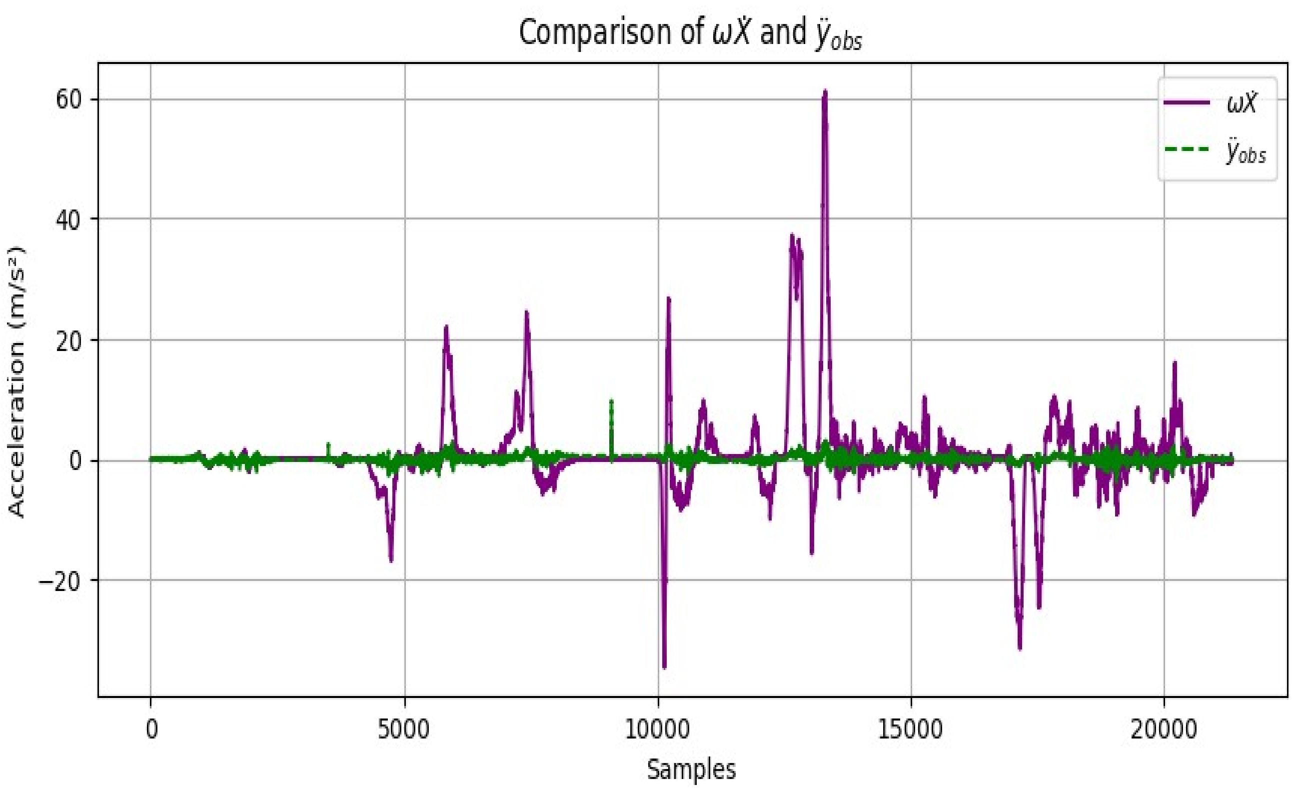


Plot 8: IMU vs GPS   (Adjusted velocity)

The velocity estimated from the IMU green line shows significant fluctuations and drift compared to the GPS velocity red dashed line. This drift is typical due to integration errors in accelerometer data. IMU velocity displays more noise and variability, with sharper peaks and troughs, which can be attributed to sensor noise and bias in the accelerometer readings.There are instances where the IMU velocity lags or leads the GPS velocity, indicating a delay or advance in capturing acceleration changes accurately. So these were drift in IMU velocity,noise and variability & lag in response.

Estimating Vehicle Trajectory

Now that we have accurate estimates of both the vehicle's orientation and velocity throughout our drive, we can estimate its trajectory. We first want to compare theoretically correct values to visualize where some errors may be and what they might be due to. First we compare the connection between the linear acceleration in the Y direction (Y double dot), yaw rate about the Z axis (Wz), and the velocity in the X direction (X dot) all experienced by the IMU. The connection between these parameters can be described by the equation: $\ddot{y} = W\text{z} * \dot{X}$



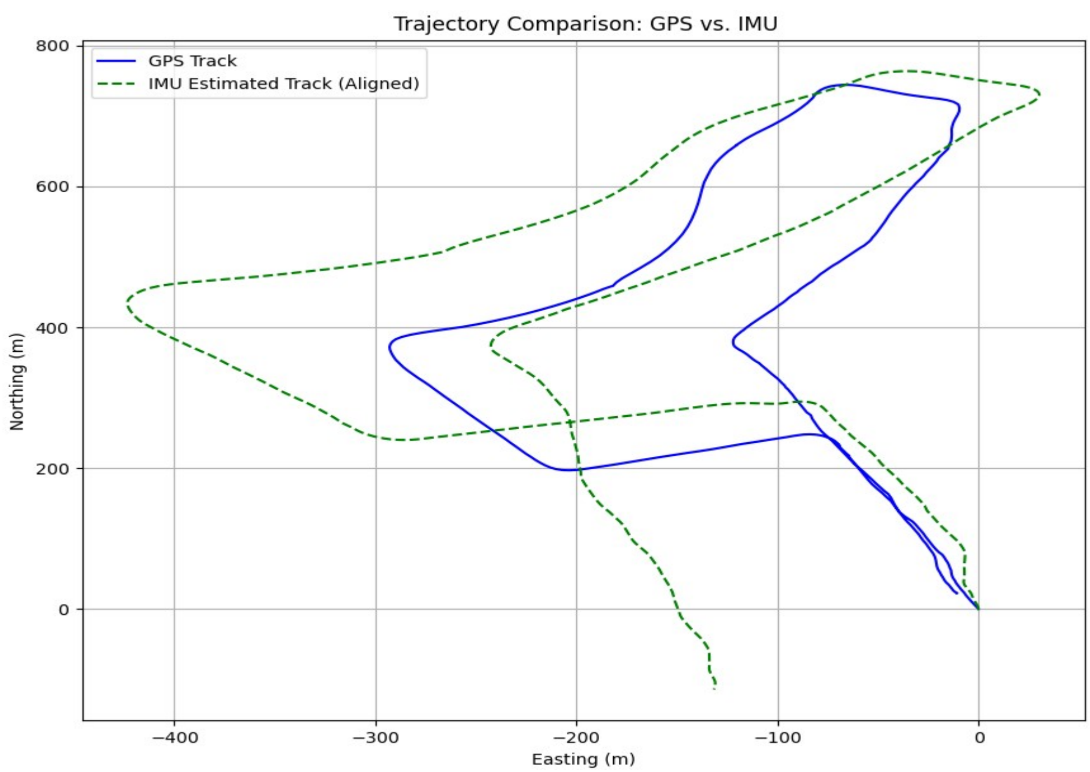
Plot 9 : for linear acceleration(y) observed by IMU vs. linear acceleration(y) calculated by ($\omega * dx/dt$)

After Computing $\dot{X}$ and $\ddot{y}obs$ The plot shows some level of agreement between ωX˙purple line and y¨obs green dashed line, particularly in the general trends and peaks. There are noticeable discrepancies, especially in the magnitude of peaks and troughs. The ωX˙ values tend to be more variable and exhibit larger spikes compared to y¨obs. Differences in sensor calibration or inherent noise in the accelerometer readings can lead to variations between the two estimates. Errors in integrating acceleration data to obtain velocity can accumulate, affecting the accuracy of ωX˙. Simplifications in the model, such as assuming no lateral movement or perfectly aligned sensors, can introduce errors if real-world conditions deviate from these assumptions. External influences like road conditions or vehicle dynamics might affect the accelerometer differently.

In order to obtain the estimated trajectory of the vehicle, I split each X-direction velocity reading in to 2components, an East and a North, similar to that read by the GPS device, using the following equations:

$$Vel\ (E) = |V| * cos(\ yaw)$$

$$Vel\ (N) = |V| * sin(\ yaw)$$

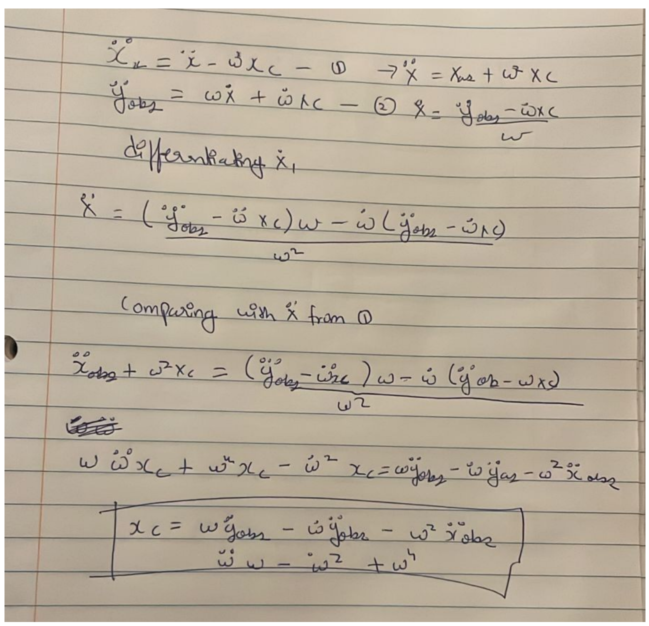


Plot 10 : Trajectory comparison GPS vs IMU

The IMU trajectory follows the GPS track, but there are discrepancies in shape and scale. Differences arise from sensor noise, biases in the IMU data. The imu trajectory matches in starting then the the noise starts to accumulate causes misalignment .

Given the specifications of the VectorNav sensor, it can navigate accurately for only a short period typically less than a minute before accumulating biases that cause position drift. Small sensor biases accumulate over time, and additional factors, such as external vibrations and varying vehicle dynamics, introduce noise, accelerating the drift. Initially, we observe that dead reckoning works effectively however, with each passing turn, errors in the IMU measurements increase. The primary cause of this drift is gyro bias instability, which results in an accumulating error in the estimated trajectory when compared to GPS data. Another significant source of error is the effect of unaccounted slopes during motion, which introduce gravity components along the IMU's moving axis, compounding the position error.The best performance is observed in the early stages, particularly when the car moves in a straight line on a level surface, where environmental and sensor-induced errors are minimized.

The above calculation for dead-reckoning with inertial sensors assumes that the sensor is placed at the center of mass - about which the car rotates. In cases where the offset between the COM of the car and the IMU position is less, the calculation error is minimal and can be ignored. However, this offset needs to be considered in critical applications like space vehicles. Therefor we need to calculate the offset from centre of mass of car using the below eqn.



$$\ddot{x}_{obs} = \ddot{x} - \dot{\omega}^2 x_c \quad (1) \quad \rightarrow \ddot{x} = \ddot{x}_{obs} + \omega^2 x_c$$

$$\ddot{y}_{obs} = \omega \dot{x} + \dot{\omega} x_c \quad (2) \quad \dot{x} = \frac{\ddot{y}_{obs} - \dot{\omega} x_c}{\omega}$$

differentiating $\dot{x}$,

$$\ddot{x} = \frac{(\dddot{y}_{obs} - \ddot{\omega} x_c)\omega - \dot{\omega}(\ddot{y}_{obs} - \dot{\omega} x_c)}{\omega^2}$$

Comparing with $\ddot{x}$ from (1)

$$\ddot{x}_{obs} + \omega^2 x_c = \frac{(\dddot{y}_{obs} - \ddot{\omega} x_c)\omega - \dot{\omega}(\ddot{y}_{obs} - \dot{\omega} x_c)}{\omega^2}$$

$$\omega \ddot{\omega} x_c + \omega^4 x_c - \dot{\omega}^2 x_c = \omega \dddot{y}_{obs} - \dot{\omega} \ddot{y}_{obs} - \omega^2 \ddot{x}_{obs}$$

$$\boxed{x_c = \frac{\omega \dddot{y}_{obs} - \dot{\omega} \ddot{y}_{obs} - \omega^2 \ddot{x}_{obs}}{\ddot{\omega}\omega - \dot{\omega}^2 + \omega^4}}$$

CONCLUSION

IMU and GPS devices are both exceptionally powerful and useful for positioning, each with its own strengths and limitations for different applications. This lab clearly shows that GPS is superior for maintaining accurate positioning over extended distances and time frames. However, IMUs excel in capturing subtle changes in vehicle orientation with high accuracy, even over longer periods. To use these technologies effectively, it's essential to understand their unique advantages and the impact of factors such as noise, gravity, and other external influences on raw data readings. When combined effectively, IMUs and GPS provide a comprehensive solution for accurately describing the position and orientation of robotic systems and other applications.