Autonomous Navigation Assignment-1 Report

DOLA RAM, 15197, DESE

August 21, 2018

1 Problem-4

With an android app like HyperIMU, log sensor stream. Using this data, plot the dead reckoning position as you move along a fixed closed path. Interpolate the GPS waypoints so obtained to plot a closed path along your trajectory (using SciPy's interpolation function) and measure the deviation of the dead reckoning measurement from this trajectory.

2 Code and Data

The code and the data have been uploaded on git lab with a read me file containing the instructions. The link is below: -
`https://gitlab.com/dolaram/HyperIMU_Sensors_Trajectory_Plot.git`.

3 Sensors Used

Three sensors of HyperIMU app in smart phone are used in to obtain the trajectory.

3.1 GPS

GPS is used to obtain the ground truth of the closed path traveled by the smart phone. GPS provides latitude, longitude an altitude of the current location. Since GPS give spherical coordinate, it has to converted to Cartesian coordinates to get trajectory of closed path.

3.2 Linear Acceleration Sensor

Linear Acceleration Sensor in HyperIMU app provide the acceleration in three directions in m/s^2. Sensor does not provide zero acceleration when device was at rest. So we need preprocessing to get final acceleration to be used.

3.3 Rotational Vector Sensor

Sensor provide the directional rotation of all three axis. Since device is kept in plane along the trajectory so sensor will provide the rotation of device about axis perpendicular to the plane of the device. This will help use to determine the exact trajectory using distance calculated from a linear acceleration.

4 Approach

Smart phone was taken along the path around the main building. GPS, Linear acceleration and Rotation angle data was collected at a sampling rate of 10ms. Steps done to get a approximate trajectory of the path followed using three sensors.

4.1 HyperIMU app Settings

Linear Accelerometer (along X, Y, and Z direction), Rotation vector (along X, Y, and Z direction) and GPS data (latitude, longitude, altitude) on the HyperIMU app are used to determining trajectory of device. Data obtained from the sensor is saved in .csv file and imported in python for analysis. Since path around Main building is close to rectangle so device have main acceleration in X-direction. Since device is kept plane so only Z-axis rotation is taken for analysis.

4.2 Velocity

Since accelerometer was not showing zero acceleration when device was at rest we have to apply error compensation. To correct this error we subtract mean of the acceleration when device was at rest from the observations.

Velocity is determined first by integrating acceleration by considering a piece-wise model and replacing sum in place of integrating for small duration, the sampling time between observations has been taken as 10ms. Plot is showing variation in speed. Device started from zero velocity and finally comes to zero velocity in the end of the trajectory.

Velocity calculation (along Y direction):

$$InitialVelocity = 0m/s, sampletime = 10ms$$

$$Velocity_{t+1} = Acceleration_t * sampletime + Velocity_t$$

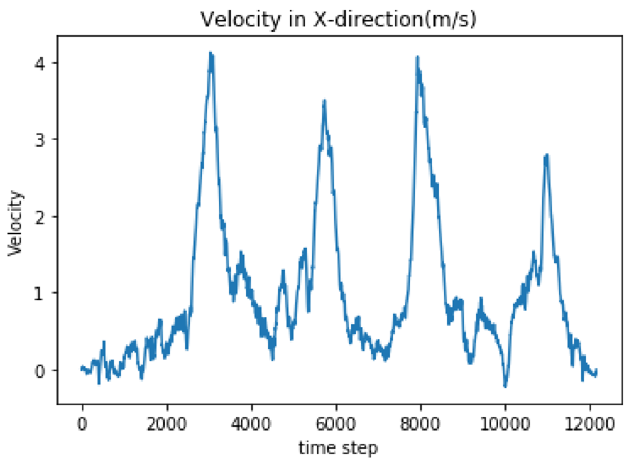

Figure 1: Velocity Plot

4.3 Distance

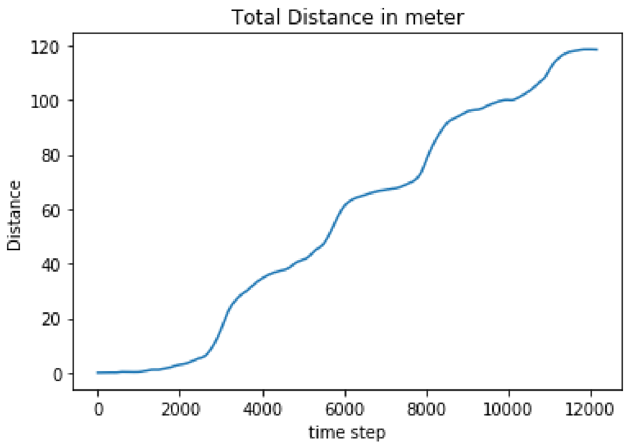

Figure 2: Distance Plot

$$Displacement_{t+1} = 0.5 * Acceleration_{t+1} * sampletime^2 + Displacement_t$$

The distance is calculated using velocity integral. Distance is the summation of the instantaneous velocity with sampling time. To obtain the approximate trajectory of device distance measurement is used with Z-axis rotation. Error between the actual distance covered and the distance shown by the sensor in within 26% of the actual value.

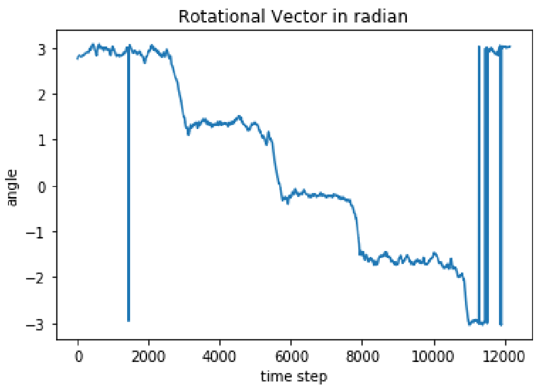

Figure 3: Rotational Vector plot

4.4 Final Trajectory

Final trajectory is plotted in Cartesian coordinate system. x and y coordinated are determined by incrementing differentially using angular and distance information at that instant.

$$X_{t+1} = X_t + dx * cos(angle)$$

$$Y_{t+1} = Y_t + dy * sin(angle)$$

where angle is obtained using rotation vector. The rotational vector gives the direction of the device in the form of sin(angle/2) so angle can be calculated by

$$angle = 2 * sin^{-1}(RotationalVector)$$

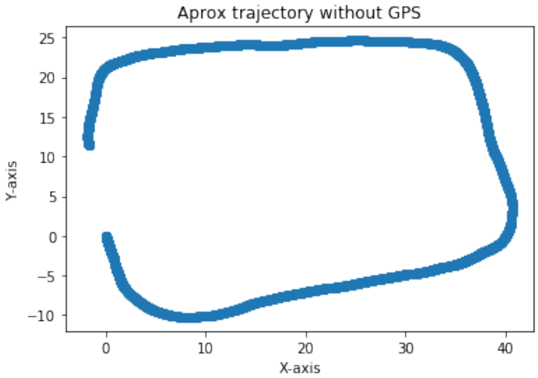

Figure 4: Final Trajectory plot

5 Ground Truth Plotting

The ground truth of the path followed is plotted using the GPS observations longitude, latitude and altitude. Since data from GPS is in spherical coordinate so points are transformed to Cartesian coordinates using:-

$$x = r * cos(latitude) * cos(longitude)$$

$$y = r * cos(latitude) * sin(longitude)$$

$$z = r * sin(altitude)$$

where r = 6,371,000m radius of earth.

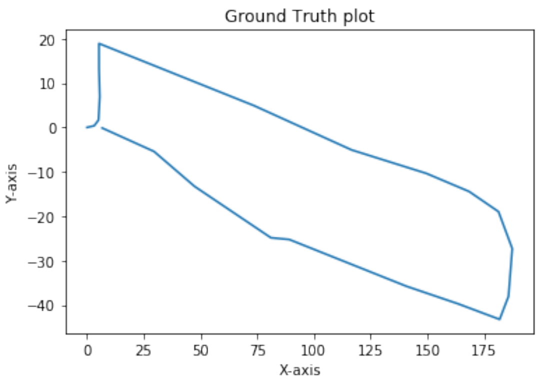

Figure 5: Ground Truth Plot