Visual Odometry:

Completed Tasks:

We have gotten the PixHawk to connect with the raspberry pi via USB and the pins. We also setup OPENCV and AruCo with python 2 and have downloaded Dronekit.

Current Tasks:

We are trying to make our code tow work using the PixHawk. Right now we are having problems with the code and it could be not setting up Dronekit correctly. Mekky said to first use a test code to see if it works.

Dead reckoning Team:

Current Tasks:

Ryan: Currently, the IMU subgroup is working on troubleshooting the IMU. The wiring from the raspberry pi to the IMU has been checked and altered slightly. Dr. Mekky and we believe that there may be a problem with the connection. On Tuesday, we went ahead and ran the data we got from last week through the bias calibration code and the code outputted a bias vector and scale factor matrix. We don't think the data we used was accurate enough so we are trying our best to find the source of error in the IMU readings.

Future Tasks:

Ryan: If we are able to get the IMU to give us accurate data, we will use the bias vector and scale factor matric from the bias calibration code and insert it into the existing Kalman filter code.

Pi Integration/GPS:

Completed Tasks:

Vivian: Nov.18th: Set up script to pull desired NMEA data from Screen text that Rushal made. We chose to pull out "GPGLL" (latitude and longitude), "GPGGA" (global attribute), and "GPGSA" (number of satellites). Nov.21st: Researched how to read GPS data through PixHawk.

https://stackoverflow.com/questions/16017419/python-read-log-files-and-get-lines-containing-specific-words

Josh: A script to save text from the screen to a file was written by Rushal and Vivian. Several errors in accessing and modifying the text file were corrected and a readable line of GPS location text was saved.

There was an error calling Dronekit on the Raspberry Pi while using MissionPlanner to verify positional readings. The libraries were reinstalled due to a missing directory error, but the functions were still unusable. More research in the use of Dronekit with the PixHawk is required.

Rushal: Successfully re-installed Dronekit on Pi. Used examples to verify that Dronekit functions properly. Dronekit functions. Successfully connected PixHawk to Pi using Mavlink. We are able to read the state of PixHawk on Pi terminal. Started looking into sending data from Pi to PixHawk using telemetry port or sending data from PixHawk to Pi: which can be reverse engineered to go the other way.