Project Proposal- 9 DOF IMU Sensor Fusion

Kaicheng Wang (kwang2@stanford.edu)

We would like to extend what we did in HW5, i.e. 6 DOF IMU sensor fusion, to 9 DOF. Specifically, we want to include magnetometer to correct yaw error. The method comes from the Oculus' paper mentioned in lecture 9 and lecture 10.

For now, our initial goal is simply to repeat section Ⅵ of the paper "Head Tracking for the Oculus Rift" published in 2014. We believe it is still the latest work in terms of head orientation tracking with low-cost MEMS sensors. The main challenge of this project would be calibrating magnetometer.

We want to obtain enough data to fit an ellipsoid of the raw magnetic field value. Then we perform an affine linear transformation to get rid of the offset from local field in circuit. At this point, out magnetometer is calibrated.

As for complementary filtering, we want to find a reference orientation first. The corresponding quaternion and magnetic field are $\hat{q}_{ref}$ and $\widetilde{m}_{ref}$. Later we would observe another pair of quaternion and magnetic field as $\hat{q}$ and $\widetilde{m}$. The difference between $\widetilde{m}' = \hat{q}^{-1} * \widetilde{m} * \hat{q}$ and $\widetilde{m}'_{ref} = \hat{q}_{ref}^{-1} * \widetilde{m}_{ref} * \hat{q}_{ref}$ is a clue for error of rotation about Y-axis. We will save more details in the proposal and leave them to our future codes.

If everything goes well, we might want to spend more time in calibration. We could explore the difference of calibrating in various environments. Furthermore, we could try different mathematical methods in terms of fitting data and getting rid of local measurement.

Here comes our timeline:
Milestone 1 (by June 1): magnetometer calibration (gather raw data in various places; fit data and perform affine linear transformation)

Milestone 2 (by June 4): complementary filtering (define appropriate reference orientation; perform yaw correction with optimal gain constant)

Milestone 3 (by June 8): hypermeter tuning and creative stuff (find the number of raw data points needed in practice; compare difference in different environment; try more mathematical methods)