ESE-650: LEARNING IN ROBOTICS

State Estimation using Unscented Kalman Filter

Ankit Vora

Aim:

To estimate the orientation of a body (roll , pitch and yaw) with respect to the world frame using an IMU. This has to be achieved by filtering the signals from the sensor using a Kalman Filter. Finally a panorama of a set of images has to be stitched based on the state estimation.

Approach

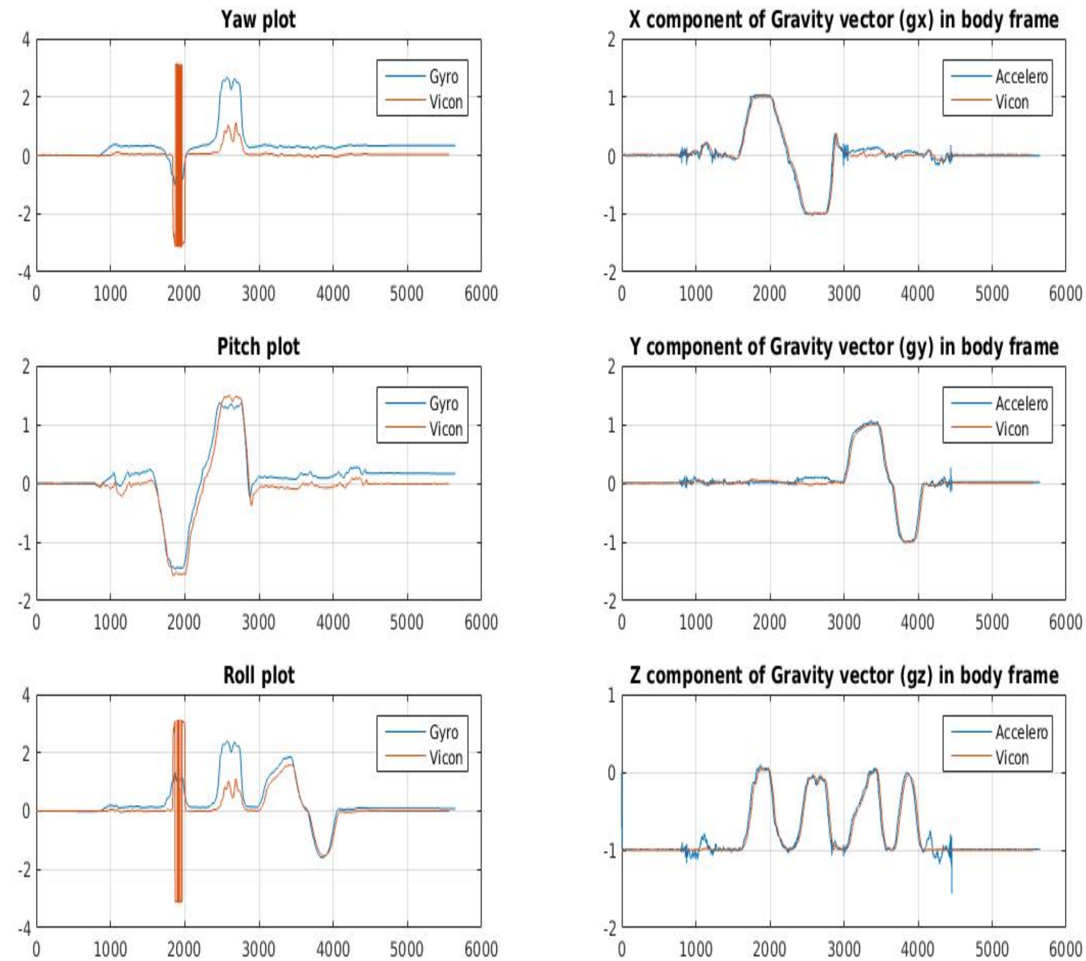
State Estimation

1. **Bias and sensitivity determination**

- The first step toward designing a filter was to determine the bias and sensitivity values of the accelerometer and gyroscope. Using the camera images, I could estimate that first 500 frames were times at which the body was at rest. This implies that the accelerometer should give values close to $[0\ 0\ -1]^T$ and the gyro should return $[0\ 0\ 0]^T$
- In addition, I followed this tutorial to get more details about the bias and sensitivity: **http://www.instructables.com/id/Accelerometer-Gyro-Tutorial/**. The datasheets provided sensitivity values which needed to be tuned. The bias was chosen as the mean of the first 500 values. The final bias and sensitivity values used are:

```
imuParams.sensGyr = 3.33*180/pi; % mv/(rad/s)
imuParams.sensAcc = 330; % mv/g
imuParams.biasAcc = [-510.1975 -501.7850 -504.535]';
imuParams.biasGyr = [373.6460 375.3760 369.6480]';
```

- Post adjusting these values, I visulaized my sensor data from the accelerometer and gyroscope. I found out the euler angles by integrating the gyro over time and also the gravity vector from the accelerometer. I plotted these with respect to the ground truth to see the extent of drift and mismatch in my true data and estimated data. The plot below shows comparison:



- Based on the above graph I tuned bias a little further to get better tracking.

2. **Designing filter**

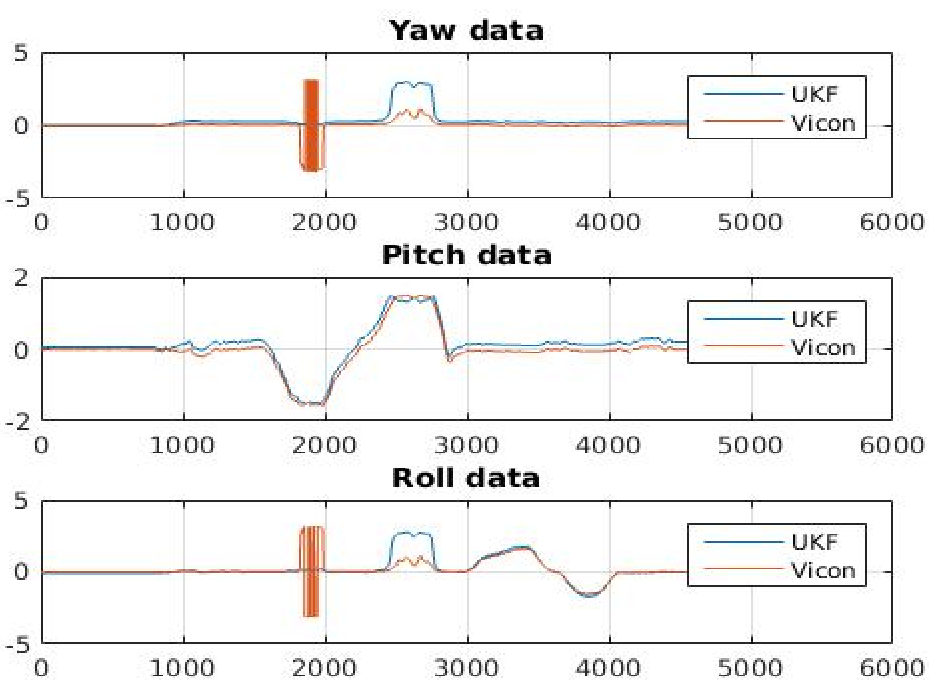
- Filter design was done based on the Kraft paper[1] titled 'A Quaternion based Unscented Kalman Filter for orientation tracking'. The approach followed was very much the standard paper approach. The main parameters of this filter were the initializations of the state and co-variance matrix along with the values of the covariance matrices Q and R of the process and measurement model.
- While tuning the filter, I realized that the P (covariance matrix of the state) was governed by the sensor covariance matrix that I have. The initial state was again [1 0 0 0 0 0 0]T since we started every time from rest. So the other two parameters, Q and R were the main tuning parameters then.
- I played around with a lot of values for Q and R. I tried various orders to check which one was perfect for my model. I got the best result when the orders were very low (10^{-6}).

Also I trusted my gyro values more than the accelerometer. Hence the accelerometer had a larger covariance than the gyro. The case was same for the process covariance matrix Q. A similar magnitude of values worked for me. Shown below are the Q and R matrix initializations that I had:

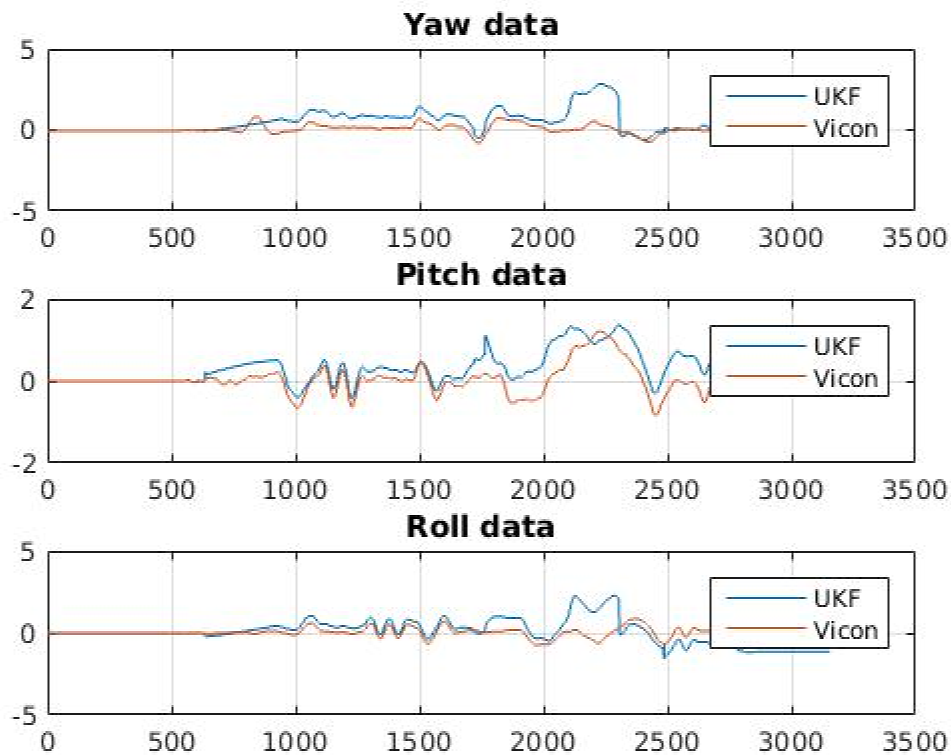
```
Ra = 10^1.1; Rb = 10^-1;
R = [Ra 0 0 0 0 0;...
     0 Ra 0 0 0 0;...
     0 0 Ra 0 0 0;...
     0 0 0 Rb 0 0;...
     0 0 0 0 Rb 0;...
     0 0 0 0 0 Rb];

Qa = 10^4; Qb = 10^-1;
Q = [Qa 0 0 0 0 0;...
     0 Qa 0 0 0 0;...
     0 0 Qa 0 0 0;...
     0 0 0 Qb 0 0;...
     0 0 0 0 Qb 0;...
     0 0 0 0 0 Qb];
```

- The output of my filter on various datasets are shown below:

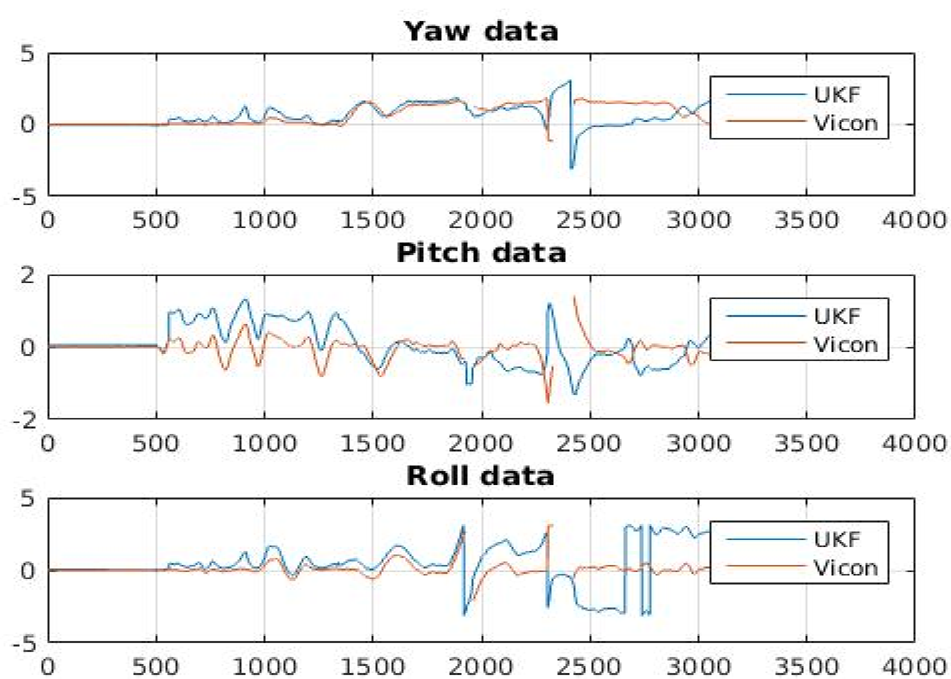
<u>Dataset:1</u>



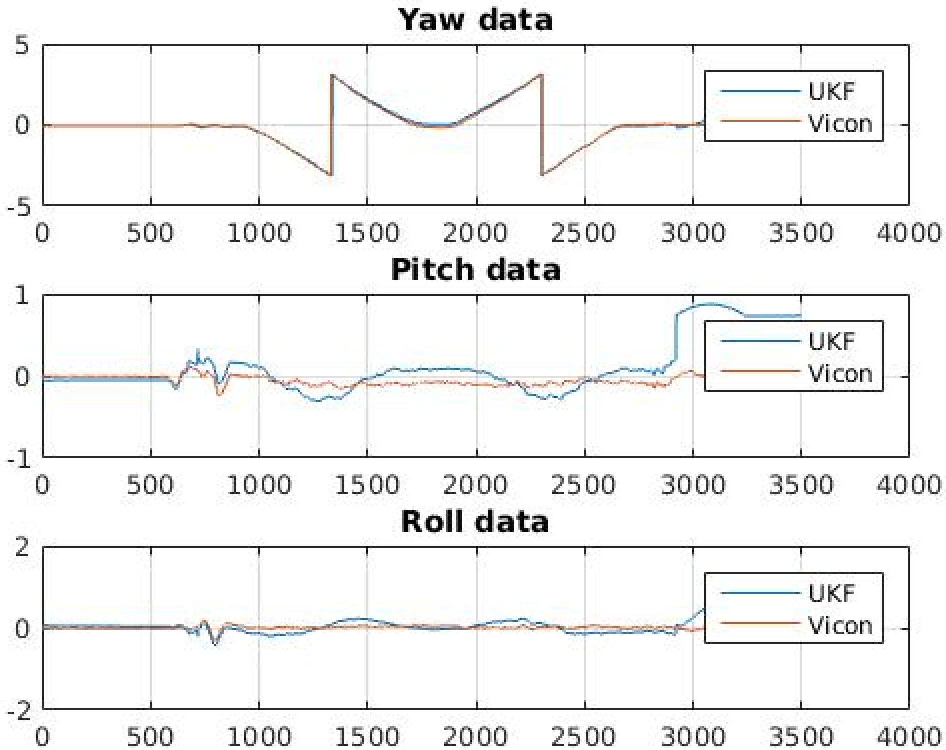
Dataset:4



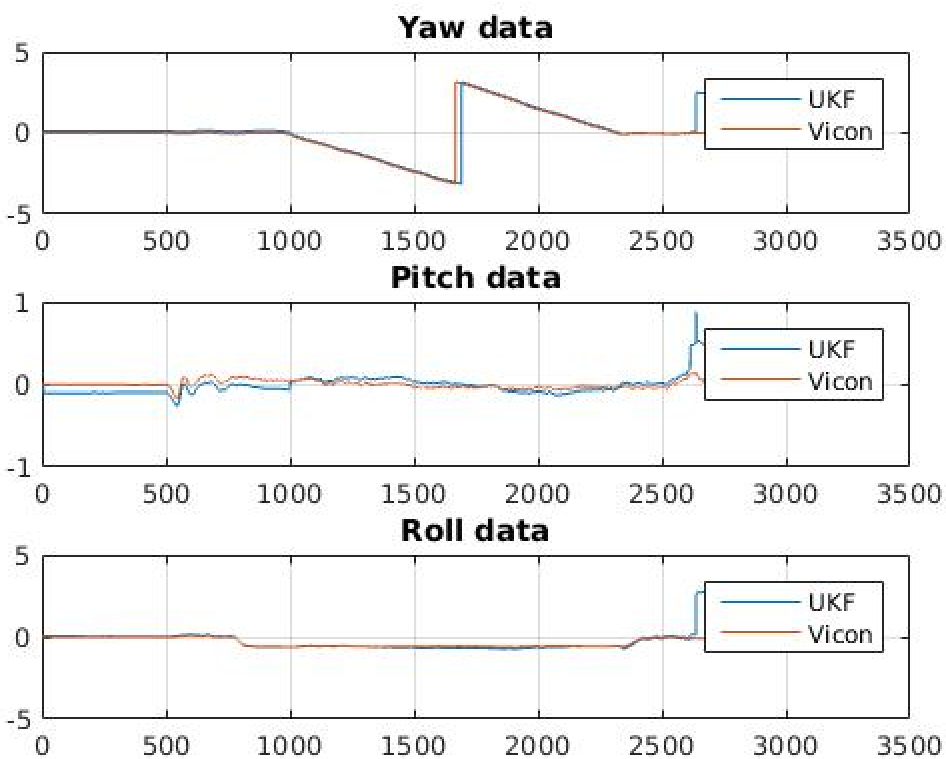
Dataset:7



Dataset:8



Dataset:TESTSET



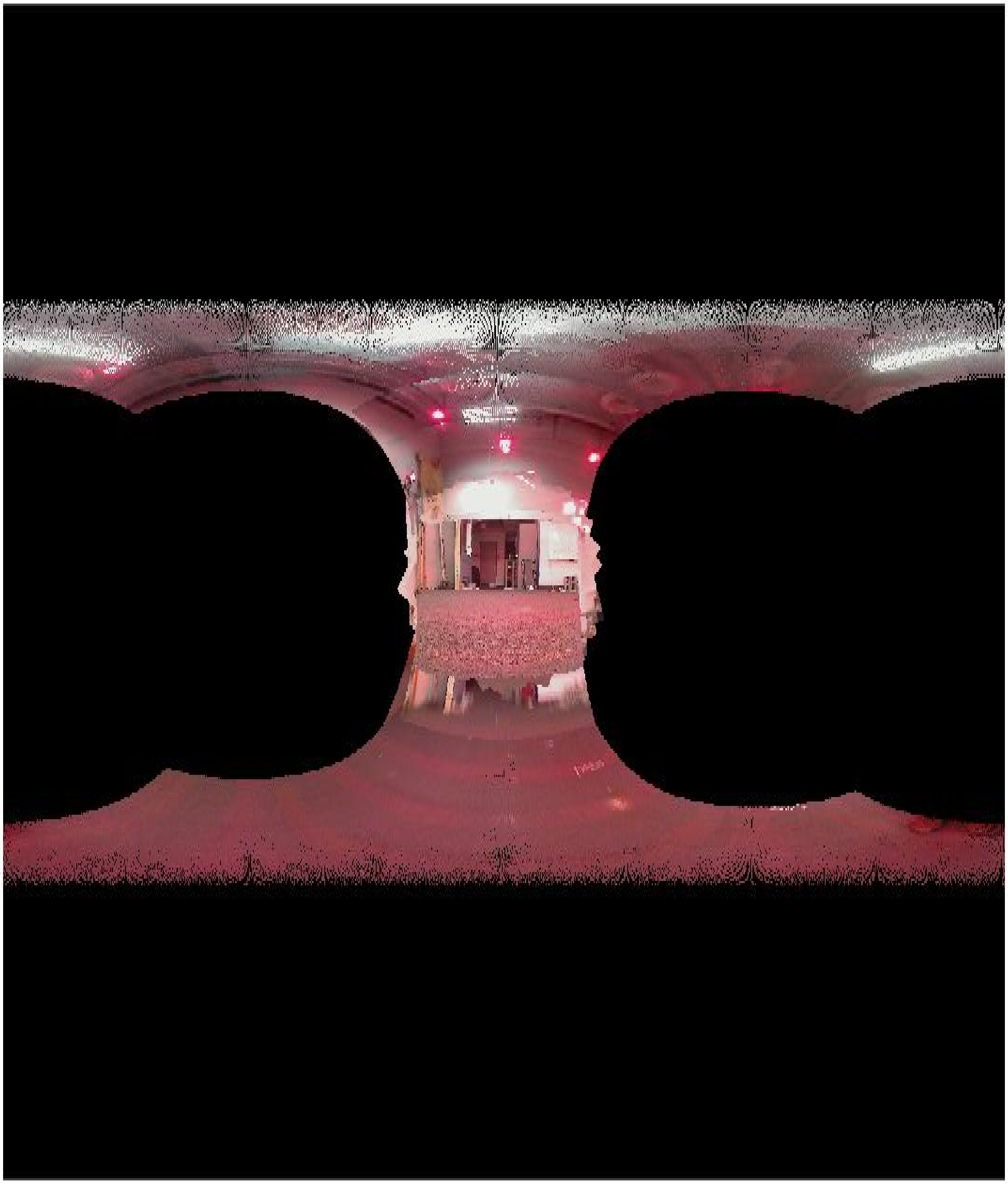
3. **Observations and Improvements**

- In this submission, I've tuned the values further. As one can say, the output is much better than the previous submission. But, there's scope for further improvement. I'll tune the Q and R values further.

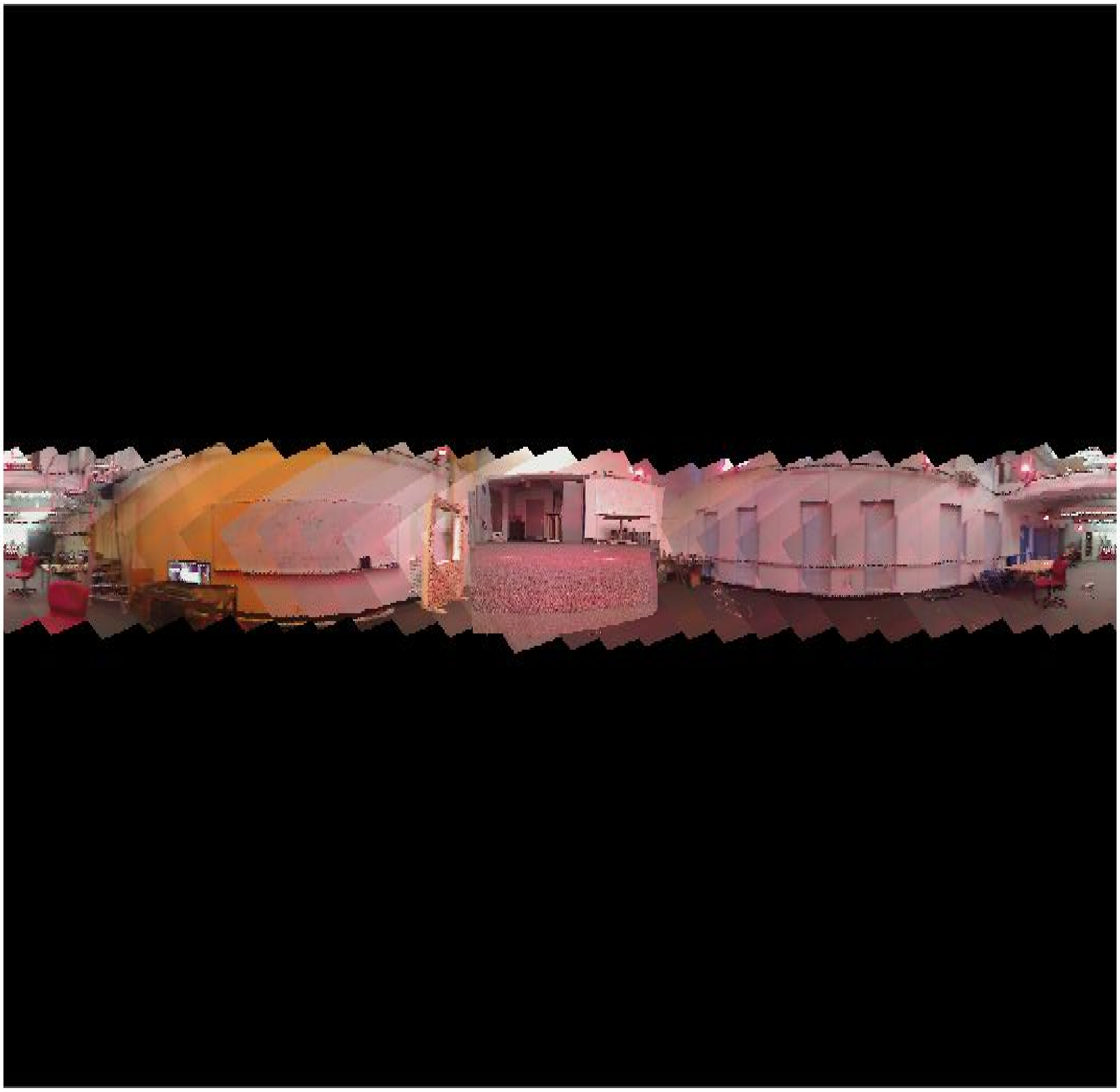
Image Stitching

- I followed a pretty naïve approach for the stitching. I synchronized my time stamps and then transformed image coordinates to spherical coordinates and projected them on a 2D canvas of 1600X800 resolution. In order to reduce the computation time, I'm sub sampling every 5 images.
- The output of few datasets is shown below:

Dataset:1



Dataset:Test



Running the code

- There's a 'main' file in the folder that needs to be executed. In the file, various '.mat' files like viconTest.mat, camtest.mat are loaded. These files need to be within their respective folders. I.e imuTest.mat should be within 'imu' folder. Else you can change the 'load' command as per your requirement. Once executed, the code will output the graphs and the stitched image.

References

1. 'A Quaternion based Unscented Kalman Filter for orientation tracking' – Edgar Kraft