ELEC 5660: Introduction to Aerial Robotics
Project 3: Phase 1

Assigned: 04-09-2024   Due: 11:59 PM 04-26-2024

1 Project Work

In Project 3 Phase 1, you need to implement the extender Kalman filter on your lecture. This is an individual project, which means you must complete it by yourself.

You will be provided with a ROS package named `ekf`, which is the skeleton code for the filter. You need to use it to implement this project fusing the odometries. You are supposed to follow the instructions below step by step.

2 Fuse the odometry with the measurement from IMU

For this part, please use the given launch file and the given bag in project 3 phase 1.
The things that you need to do in this part are described as follows.

1. Fuse the odometry obtained from your `tag_detector` with the measurement from IMU.

2. Publishing pose information in the form of `nav_msgs/Odometry`.

3. Plotting position and orientation after filter using **rqt_plot**.

4. Comparing your result with the odometry before filtering.

Some hints are listed as follows

1. The measurement from IMU is in the **body frame**(i.e. IMU's frame), but the pose estimation from your PnP solver is the pose of **world** in the **camera** frame. The important thing is that, this **world** frame's Z axis is pointing down. All frames here are right-hand frame.

2. If you can't understand the above sentences, you only need to know that finally you need calculate R_{wi} and T_{wi}. In the tag_detector package, what you get is R_{cw} and T_{cw}, and we provide R_{ic} and T_{ic}(relative transformation between camera and IMU) in the code. Here c means camera, i means IMU and w means world. You don't need to know the absolute pose of each frame, all you need to care about is only **relative transformations between frames**.

3. Be careful about the discontinuous nature in Euler angles when you obtain the Roll, Pitch and Yaw angles from rotation matrix.

4. IMU data is in high frequency(400 Hz) and image data is in low frequency (20 Hz). The timestamp is already synchronized and you can read them from the repsecting message header.

5. The filter is not very sensitive to your initial value of covariance matrix. But please remember to set a right initial value for your ekf state. And you can try to set a proper value for the gravity(around 9.8 m/s^2 but may not be exact).

In the launch file, we half the play rate of the provided dataset in case your hard disk may stuck when play a big volume of data, you can change the rate in the launch file, the number behind "-r" is the dataset play rate. Figure 1 are some results plotted in rqt_plot, the filter is not fine tuned, but they are correct results.

3   Submission

When you complete the tasks you could submit your code and documents to canvas before **Apr 26th, 2024 23:59:00**. The title of your submission should be "`proj3phase1_YOUR-NAMES`".

Please cite the paper, GitHub repo, or code url if you use or reference the code online. Please keep academic integrity. **Plagiarism** is not tolerated in this course.

Your submission should contain:

1. A **maximum 2-page** document including:

   (a) Figures plotted by **rqt_plot** and **rviz**.
   (b) Descriptions about your implementation.
   (c) Any other things we should be aware of.
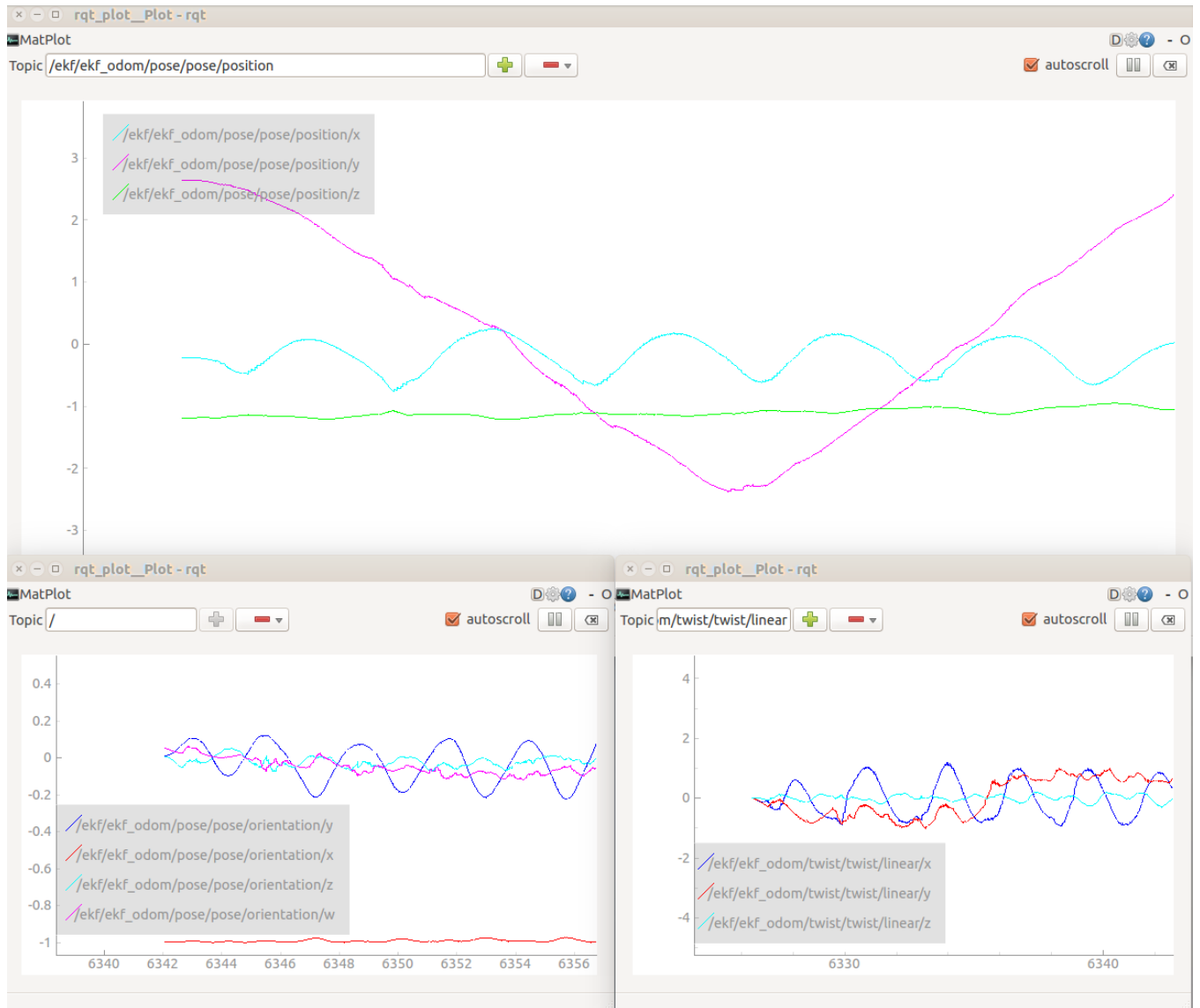
2. All files you need to run your code.

Figure 1: Position(above), orientation(left, down) and velocity(right, down)