Assignment2: Kalman Filter

February 28, 2023

1 Description

1.1 Part A

There is a nonholonomic moving robot in the 2d plane whose state vector (pose) is represented as $[x, y, \theta]^T \in \mathbb{R}^3$ (rotation, position). The moving robot will estimate its relative transformation between each step $T_{t,t+1}$. The moving robot will also get transformation with respect to the world coordinate $T^{t,w}$ from registration with observable landmarks L.

Note that the relative pose information is from odometry:

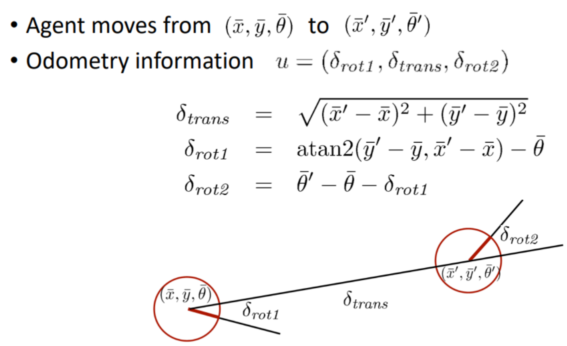

Figure 1: Motion model

1.2 Part B

There is a holonomic moving robot in the 2d plane whose state vector (pose) is represented as $[x, y, \theta, v_x, v_y, w]^T \in \mathbb{R}^6$ (rotation, position, speed). The moving robot will only get transformation with respect to the world coordinate $T^{t,w}$ from registration with observable landmarks L. Besides, the robot is moving at a constant velocity while the actual speed has Gaussian noise, you need to take it into account.

2 Your goal

2.1 Part A

- Suppose that both odometry information and $T_{t,w}$ is noisy, please try to implement a Kalman Filter-based sensor fusion method (use odometry and landmark observations to minimize the uncertainty of the trajectory).
- You need to 1) visualize the trajectory of landmark observation. 2) visualize the trajectory concatenated by odometry measurements. 3) visualize the trajectory after you apply the Kalman Filter.
- Your trajectory visualization should include the heading angles of each pose.

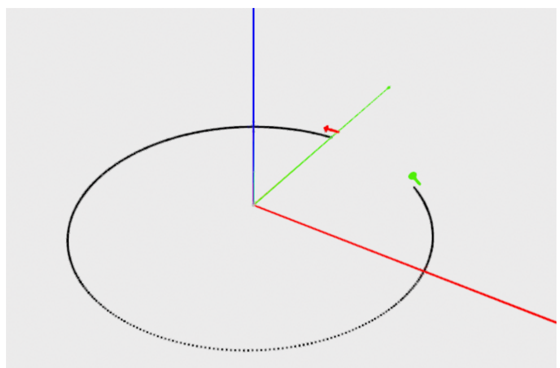

Figure 2: visualization of ideal trajectory, the red arrow is the starting point

2.2 Part B

- Suppose that $T_{t,w}$ is noisy, please try to implement a Kalman Filter for localization (use landmark observations and the constant velocity model to minimize the uncertainty of the trajectory).
- You need to 1) visualize the trajectory of landmark observation. 2) visualize the trajectory after you apply the Kalman Filter.
- Your trajectory visualization should include the heading angles of each pose.
- Please plot the robot's velocity over time (both linear and angular).

3 Data

Download All the data is in a zip file named "hw2_dataset.zip". Please download it from Piazza.

3.1 Part A

Format The zipped file contains files named by "landmark_A.txt" (transformation from t_i frame to world frame) and "odometry_A.txt" (transformation from t_i frame to t_{i+1}). In the "landmark_A.txt" file, each line has 3 numbers representing the xy coordinate and angle θ (the unit is meter and radian). And in "odometry.txt.", each line contains $(\delta_{rot1}, \delta_{trans}, \delta_{rot2})$

Noise level Assume all the noise subjects to 0 bias Gaussian distribution. For landmark observations, the standard deviation of the rotation part is 0.03, and the standard deviation of the translation part is 0.05. For odometry measurements, the standard deviation of the rotation part is 0.02, and the standard deviation of the translation part is 0.05.

3.2 Part B

Format The zipped file contains files named by "landmark_B.txt" (transformation from t_i frame to world frame). In "landmark_B.txt" file, each line has 3 numbers representing the xy coordinate and angle θ (the unit is meter and radian).

Noise level Assume all the noise subjects to 0 bias Gaussian distribution. For landmark observation, the standard deviation of the rotation part is 0.05, and the standard deviation of the translation part is 0.05. For speed, the standard deviation of the rotation part is 0.02, and the standard deviation of the translation part is 0.05.

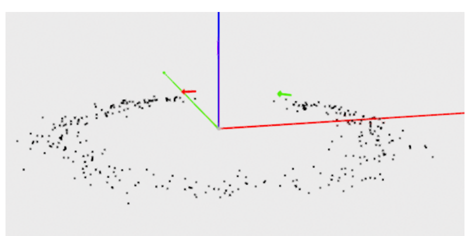

Figure 3: visualization of landmark observations

4 Implementation requirements

- Both C++ and Python are acceptable.
- You can only use third-party libraries for matrix computation (i.e. numpy and Eigen) and visualization(i.e. Open3D or OpenGL).

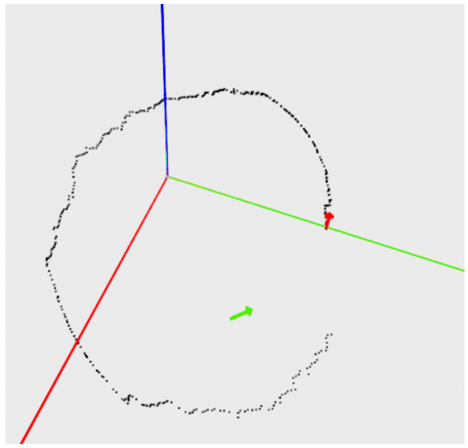

Figure 4: visualization of accumulated relative poses

5 Submission requirements

- Deadline: 19th of March 23:59.
- Submit your solution code with a one-page summary of your code (structure and usage) and results as a zip to the TA.
- Make sure the email has the header "CS284: HW2: Your Name"
- We would then arrange a meeting after the deadline in which we would ask each one of you to come in for 10 minutes to demonstrate your solution on your own computer.
- If submitted after the deadline but still within 24hrs, a 50% penalty is applied. If submitted more than 24hrs after the deadline, a zero score will be given.