UDACITY

PROJECT

Kidnapped Vehicle

A part of the Self-Driving Car Engineer Program

PROJECT REVIEW

CODE REVIEW

NOTES

SHARE YOUR ACCOMPLISHMENT!

Meets Specifications

Enthusiastic Learner,

This work is an outstanding one. Fabulous! I enjoyed reviewing this work 💯 %. It is straight, concise, well documented and a well-articulated material 👏 Bravo. A lot of knowledge in this area has been showcased in this project. Keep it up. Best wishes in your career ahead, stay excellent and good luck.

Further Improvement Suggestions

- To expand ones knowledge in this area, it might be good in ones extra time to look at the following materials:
  - See here http://users.isy.liu.se/rt/fredrik/reports/09TAESpftutorial.pdf where theory meets practice in particle filters
  - http://www.irisa.fr/aspi/legland/ref/arulampalam02a.pdf
  - Applying particle filters in robots http://robots.stanford.edu/papers/thrun.pf-in-robotics-uai02.pdf
  - A gentle introduction to particle filtering http://www.lancaster.ac.uk/pg/turnerl/PartileFiltering.pdf
  - Particle filters and its applications http://ocw.alfaisal.edu/NR/rdonlyres/Aeronautics-and-Astronautics/16-412JSpring-2005/F9652688-E118-442E-98CE-3013CBEB8F11/0/a5_hso_plnvl_mlr.pdf
  - Get to understand the difference particle filter has with other filters here http://www.dsi.unifi.it/users/chisci/idfric/Nonlinear_filtering_Chen.pdf
  - Robot mapping with particle filters http://duch.mimuw.edu.pl/~kowaluk/GOBR/slam11-particle-filter.pdf
  - https://en.wikipedia.org/wiki/Particle_filter

- Particle/Kalman Filter for Efficient Robot Localization
- Great content for Kalman and Particle filters
- See also Marginalized Particle Filters for Mixed Linear/Nonlinear State-space Models
- Particle Filters for Positioning, Navigation and Tracking

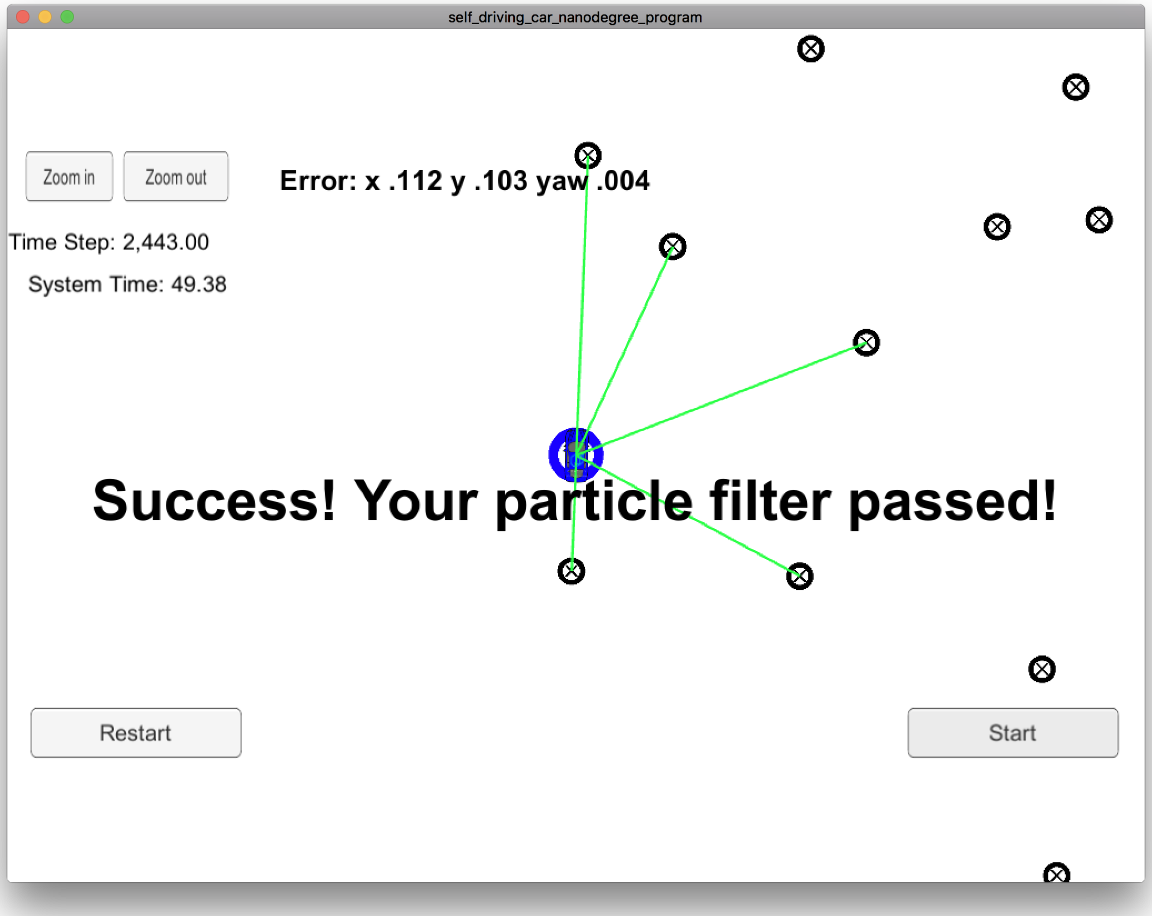
Accuracy

This criteria is checked automatically when you do ./run.sh in the terminal. If the output says "Success! Your particle filter passed!" then it means you've met this criteria.

Great job in this section. The success is a big one. This particle filter passed excitingly. keep it up! 👍

Further Improvement Suggestion

- Variance estimation in the particle filter



Error: x .112 y .103 yaw .004

Time Step: 2,443.00

System Time: 49.38

Success! Your particle filter passed!

Performance

This criteria is checked automatically when you do ./run.sh in the terminal. If the output says "Success! Your particle filter passed!" then it means you've met this criteria.

Outstanding performance in this section in using the algorithm to produce a perfect particle filter 👏

Further improvement suggestions

- Performance evaluation of particle filters here http://fusion.isif.org/proceedings/Fusion_2011/data/papers/247.pdf? could help broaden one's knowledge further.
- Get to know the good, bad and ugly of particle filters here http://www.cs.cmu.edu/~16831-f12/notes/F14/16831_lecture05_gseyfarth_zbatts.pdf
- Improving particle filters http://www.dis.uniroma1.it/grisetti/pdf/grisetti06tro.pdf
- Comparing particle filters performance http://asa.scitation.org/doi/abs/10.1121/1.3292596
- Some things to know in order to improve performance of particle filters here http://adsabs.harvard.edu/abs/2014AGUFMNG23C..05S
- [Performance Bounds for Particle Filters Using the Optimal Proposal]http://journals.ametsoc.org/doi/abs/10.1175/MWR-D-15-0144.1

Some code tuning ideas

- Some C++ debugging tips to consider as a developer, see
  - setting CMAKE_BUILD_TYPE to release
  - setting CMAKE_BUILD_TYPE to debug
  - In case of any failure or difficulty, most commonly Understanding why CMAKE_BUILD_TYPE cannot be set
- When cmake includes symbol tables in the generated file, GDB now can come in to assist in debugging critical faults like the famous segmentation fault, why some variables are behaving weird, etc, see
  - How to Debug Using GDB
  - GNU GDB Debugger Command Cheat Sheet
  - Debugging with GDB By Alexandra Hoffer

General

There may be ways to "beat" the automatic grader without actually implementing the full particle filter. You will meet this criteria if the methods you write in particle_filter.cpp behave as expected.

Correct steps were taken in this section as prescribed. What has been done in `particle_filter` in calculating correct particle filter pipeline looks great. Impressive 🎉

Suggestion and comments

- See excellent tutorial on particle filters here http://www.stats.ox.ac.uk/~doucet/doucet_johansen_tutorialPF2011.pdf

DOWNLOAD PROJECT

RETURN TO PATH

Student FAQ