

[◀ Return to "Self-Driving Car Engineer" in the classroom](#)[DISCUSS ON STUDENT HUB](#)

Kidnapped Vehicle

REVIEW

CODE REVIEW 2

HISTORY

Meets Specifications

Interestingly, I found the noise to be very important. Without adding noise in the prediction step, the main particle will steer off the trajectory of the car!

Yes, this is true, great observations! If we think a bit more about this, we will see this is reasonable. If we don't add noises in the prediction step, we will end up with fewer and fewer particles (the particles' state will be same, though the number of particles can be large), thus the weight update and resampling steps no longer serve any purposes. On the other side, our prediction model does not really perfectly match reality, so noise are needed.

Nice work in implementing a 2 dimensional particle filter in C++, and applying it to locate a kidnapped vehicle, the vehicle's state is successfully recovered from uncertain control and measurement environment, given known map data.

Histogram Filter, Kalman Filter and Particle Filter are all Bayes filters and can be used to solve localization problem, they differ in the fact that different mathematical representation of system state distribution is assumed in each filter. Please read [here](#) for more information.

If you wish to dig deeper into localization, the [Robust and Precise Vehicle Localization based on Multi-sensor Fusion in Diverse City Scenes](#) paper published by Baidu Apollo team could be an interesting read. It fuses information from complementary sensors such as GNSS, Lidar, and IMU to achieve centimeter level accuracy in various challenging scenes including urban downtown, highways, and tunnels.

Keep up the great work and Happy learning!

Accuracy

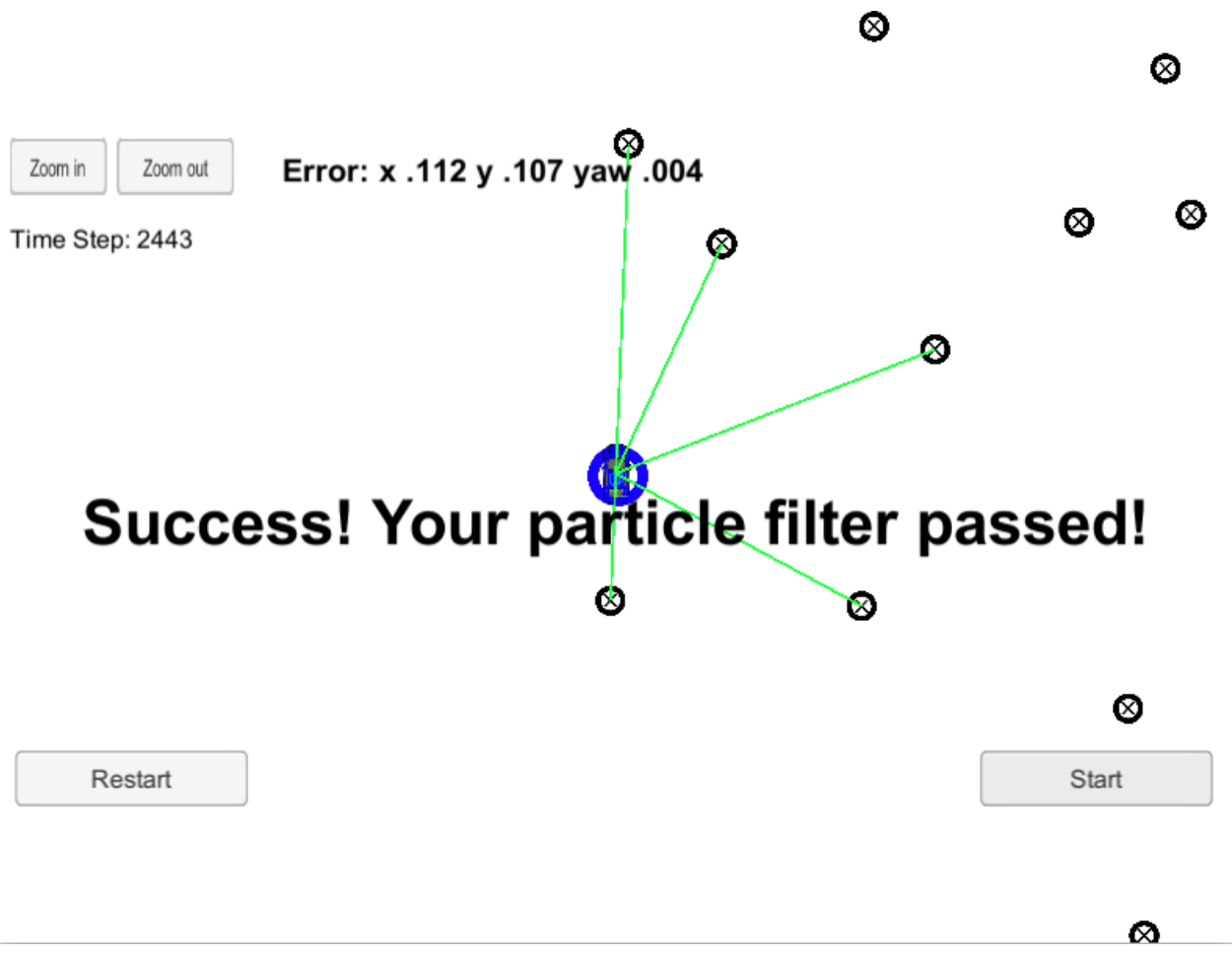
This criteria is checked automatically when you do `./run.sh` in the terminal. If the output says "Success! Your particle filter passed!" then it means you've met this criteria.

After running `./run.sh` in the terminal, the output says "Success! Your particle filter passed!". Well done!

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.

After running `./run.sh` in the terminal, the output says "Success! Your particle filter passed!". Your implementation met performance criteria when running on my laptop.



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.

Particle filter is properly implemented. Good job!

 [DOWNLOAD PROJECT](#)

2

[CODE REVIEW COMMENTS](#)



[RETURN TO PATH](#)