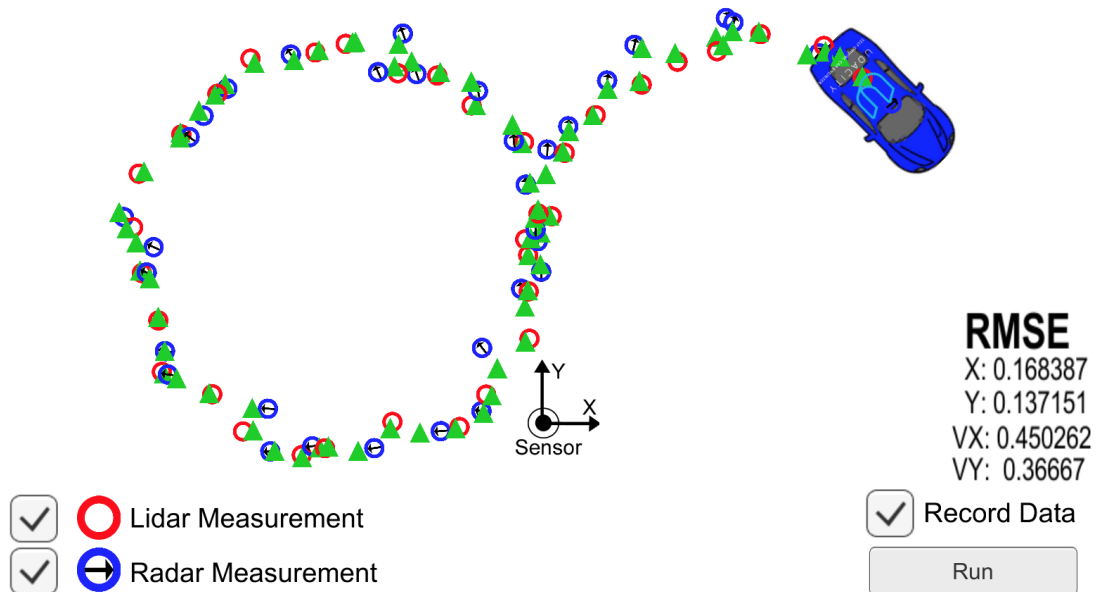


Like in the EKF project, you can use the Kalman filter tracker visualization tool. There is no requirement to use this tool, but the visualizer is useful for getting more intuition about what the Kalman filter is doing.



kalman filter visualizer

## Kalman Filter Tracker Visualization Tool

The Kalman Filter Tracker program is a visualization tool that plots noisy lidar and radar measurements as a simulated car drives in a figure eight pattern. The program can also record measurement data directly to a text file and then use a python script to call a compiled c++ Kalman Filter file to track estimated markers and RMSE values visually in real time.

## Download Links for Kalman Filter Tracker

- [Windows](#)
- [Mac](#)
- [Linux](#)

Copy and Save the [kalman\\_tracker.py](#) script code,

which is used to communicate between the simulator and your compiled c++ kalman filter program.

The script requires 1 argument which is the path to your compiled c++ file. Here is an example for how to call the script.



---

### Running the Program

1. first open the simulator program (Preferred settings are windowed 1600 x 900 with Simple)
2. Click Record Data in the simulator, a browser will then open to where you want to save your text file output. Make sure the path is the same as where kalman\_tracker.py file is.
3. Run kalman\_tracker.py with path to your compiled c++ file.
4. Click Run in the simulator to see estimation markers and RMSE values displayed in real time as measurement markers are collected.
5. Experiment with turning on/off lidar and radar measurements on the lower left of the simulator to see how RMSE is effected.

Note about RMSE: to display RMSE values in the simulator the compiled c++ file needs it's first cout row to contain "RMSE", then cout each RMSE value for every subsequent row. Here is an example of what that might look like. The four RMSE values are x, y, vx, vy errors.

```
cout <<
```

```
RMSE
```

```
1.0
```

```
1.0
```

```
1.5
```

```
2.0
```

NEXT