A Pseudo-Derivative Method for Sliding Window Path Mapping

Lane Finder

Contact:

- Joe McInnes - jmacinnes19@wooster.edu

Cite: *[A Pseudo-Derivative Method for Sliding Window Path Mapping in Robotics-Based Image Processing]* Landon Bentley, Joe MacInnes, Rahul Bhadani and Tamal Bose, 2018

The `lane_finder` class operates on an already inverse-perspective-mapped image featuring minimal noise (i.e. the lanes are the only thing in the image - mostly).

It implements the pseudo-derivative sliding window algorithm described in to find points along the road for an autonomous vehicle to follow. These points are relative to the car's location, but are not adjusted for any global coordinate space (eg GPS).

The algorithm relies on a variety of hyperparameters that are documented in the class's constructor.

Usage

Using the class in a ROS pipeline

The `lane_finder` class must be constructed with a single image and defined hyperparameters. Thus, a new `lane_finder` object is created for each frame of input. Using default hyperparameter values, this might look as simple as

```
lf = lane_finder(cv2_image_obj)
```

Calling the class's `pathGen` method returns a ROS path message defined in `ros_classes.py` that contains lane points to follow relative to the car's location in meters.

The program is compatible with Python 2.

Using lane_finder.py as an executable

`lane_finder` can be executed at the command line and passed a path to a single image to show visualize the algorithm's performance on a single image.

```
./lane_finder.py /path/to/image.jpg
```

Interpreting Visualizations

`lane_finder` has a `visualize` method that will return the image with information overlayed on top in the following format:

Green Circles → these points are lane points from the expected right and left lanes. These points are used to calculate the points to follow.

Red Circles → these circles represent the seek forward behavior of the algorithm. Each red circle is an iteration of seeking forward, where the algorithm is trying to rediscover the lane.

Pink Circles → these circles are the points along the road to follow and represent the algorithm's best guess at the mlane midpoints.

Copyright notice