UDACITY

‹ Return to "Self-Driving Car Engineer" in the classroom

DISCUSS ON STUDENT HUB

Finding Lane Lines on the Road

REVIEW

CODE REVIEW

HISTORY

Meets Specifications

Hello,

Good work overall, you did a good job with your pipeline, Keep this up!
Congratulations, on meeting all of the specifications and good luck with your Self Driving Car Nanodegree. 😄

Required Files

The project submission includes all required files:

- **Ipython notebook with code**
- **A writeup report (either pdf or markdown)**

Lane Finding Pipeline

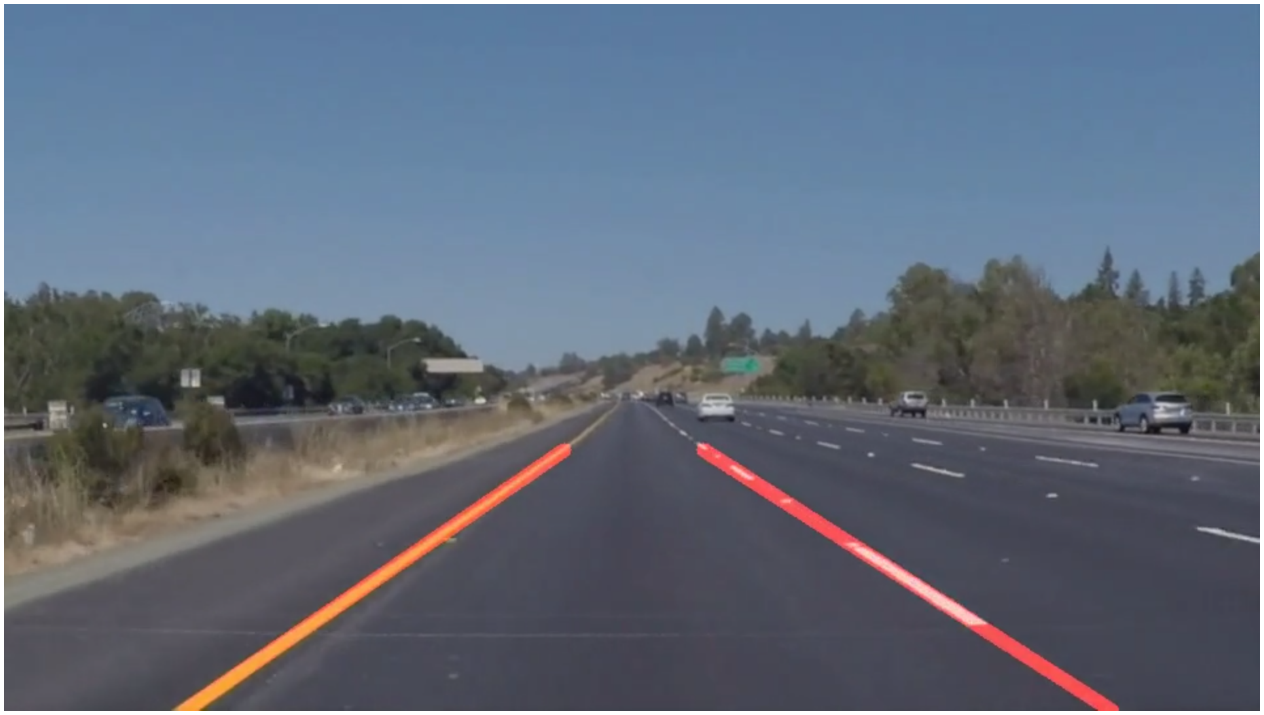
The output video is an annotated version of the input video.

In a rough sense, the left and right lane lines are accurately annotated throughout almost all of the video. Annotations can be segmented or solid lines

Your pipeline is pretty good, with resulting lines right on target. 👏

Visually, the left and right lane lines are accurately annotated by solid lines throughout most of the video.

Single solid line ✅
On target ✅'



The left and right lane lines are accurately annotated throughout most of the video. Well done!!

Your parameters looks good, yet you may try one optimal possible set of parameters which might not create much difference your current output but will be helpful in certain conditions like curved lanes, shadows etc.

- Increasing `min_line_len` (~100 and above) for Hough Transform will make your lines longer and will have less number of breaks.(this will make the solid annotated line longer in the output).
- `threshold` increasing(~ 50-60) will rule out the spurious lines.(defines the minimum number of intersections in a given grid cell that are required to choose a line.)
- Try to use `rho` as 2.

Reflection

Reflection describes the current pipeline, identifies its potential shortcomings and suggests possible improvements. There is no minimum length. Writing in English is preferred but you may use any language.

Pipeline description ✅
shortcomings ✅
Improvements ✅

DOWNLOAD PROJECT

RETURN TO PATH

Rate this project