UDACITY

PROJECT

Finding Lane Lines on the Road

A part of the Self Driving Car Engineer Nanodegree Program

PROJECT REVIEW

NOTES

SHARE YOUR ACCOMPLISHMENT!

Meets Specifications

Hi there!

Congrats on taking the first step in the Self Driving Nano degree.
I hope you enjoyed working through the project and developed a basic intuition on how to go about lane detection. I have added few suggestions and resources which will improve your understanding and make the project more robust.
All the best for the future projects 😄

Finding Lane Lines with Aaron Brown

Required Files

The project submission includes all required files

Lane Finding Pipeline

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

The output video is an annotated version of the input video. Well done 👍

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

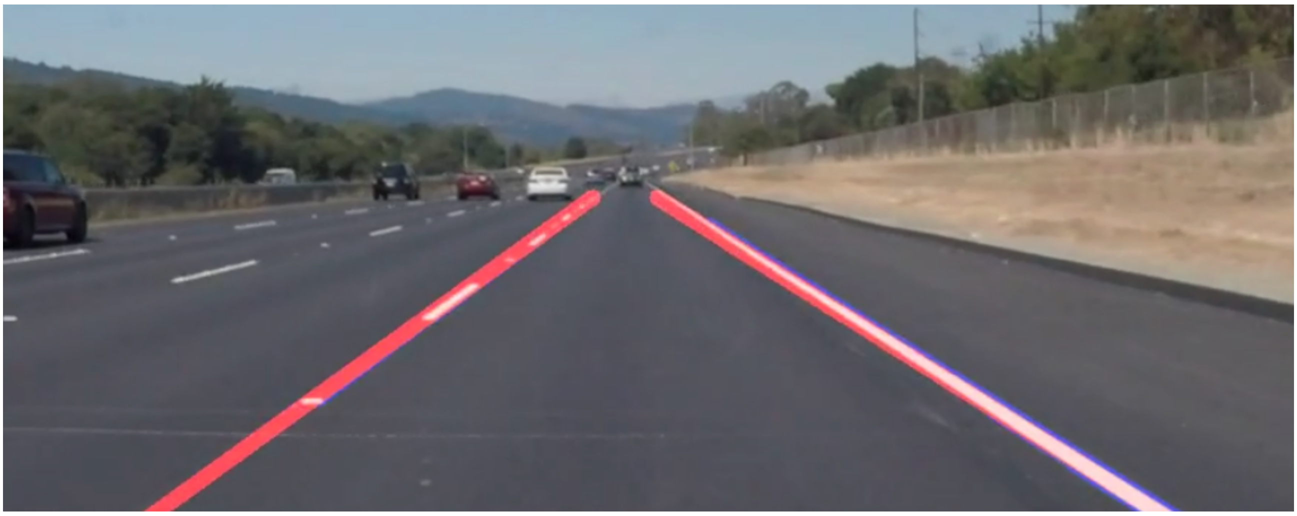
Well done. The left and right lane lines are accurately annotated throughout almost all of the video.

- `max_line_gap` maximum distance(in pixels)between segments that will be allowed to connect into a single line.
- `min_line_len` minimum length of a line (in pixels) that we will accept in the output
- Increasing `min_line_len` and `max_line_gap` (~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)
- Increasing max_line_gap will allow points that are farther away from each other to be connected with a single line.
- `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.)
- `kernel-size` in the Gaussian Filter will remove the noise making the image less blurry. Must be an odd number (3, 5, 7...)
- Use `rho` value of 2 or 1 which gives distance resolution in pixels of the Hough grid

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

The left and right lane lines were annotated throughout almost all of the video.

Style Guide for Python Code



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.

Great intuitions and understanding. You have correctly explained the shortcomings and potential improvements. 👍

Few points to think upon

- Can a different colour space be used to make the pipeline more robust.
- I encourage you to try and tune the parameters. For starters you can refer to the algorithm discussed here

DOWNLOAD PROJECT

Have a question about your review? Email us at review-support@udacity.com and include the link to this review.

RETURN TO PATH

Rate this review

Student FAQ