UDACITY

Back to Self-Driving Car Engineer

Finding Lane Lines on the Road

REVIEW

HISTORY

Meets Specifications

I had a lot of joy reviewing your work :)

Hope to review some of your future works friend,

Farewell,

Sedar,
@sedarolmez

Lane Finding Pipeline

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

Really excited when reading your code, I quite enjoyed how you tested various parameters on the images, the reduced region of interest, and also the "Hough Lines" test! You've managed to successfully test various values for the parameters and chosen the correct values for optimum output, your video "white.mp4" gives me the assumption that you haven't instantiated the kernel_size variable for the gaussian blur method, for this particular domain to blur the image to the right amount for the Gaussian Filter to be optimum you must instantiated kernel_size value=3.

Anyway, you may read more about the OpenCV API at:
http://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=gaussianblur#gaussianblur

Not many students are able to ensure that the "left-broken lane lines" for the video "white.mp4" or "yellow.mp4" produce a connected Hough Line, but you have, which is quite intriguing :) even though a few minor changes for your parameter values like min_line_length = 100, max_line_gap = 150 has a inconsistency, i.e. how can you assume that the lane line will be a small threshold of 100? in length, a threshold would require you to test a set of values within a min and max threshold, so a more suitable value for these variables would be, say: min_line_len = 100, max_line_gap = 160.

Overall, a really good solution and you have met the requirements for this specification!

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

Congratulations, for your "white.mp4" and "yellow.mp4" you have conveyed a firm understanding of the OpenCV API, you have used various method presented in the API.

Apologies for writing so much and boring you :D but, I'm quite excited with your solution, overall both your videos have Hough Lines which are unbroken but with the change of some parameter values above, would increase the distance of the line to consider a further trajectory. Nonetheless you have passed this specification with flying colours!

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

I would definitely recommend reading the python programming conventions: https://www.python.org/dev/peps/pep-0008/ in particular to try reduce the amount of code you write and rather create extra methods with various parameters instantiated different objects of the same class rather than repeating code countless of times in your draw_lines function :)

Regarding parameter values and in the future, try to test various values within thresholds for the domain you're coding for to never miss out a particular optimal output.

Not much to say about this requirement, you have passed it with flying colours and also I have elaborated a little more on the comments above.

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.

Amazing theoretical analysis, it's like reading Shakespeare but you explaining every inch of your code, I read all of it but you've blown me away, I would like to focus on a particular point you've made: "This initial solution is only applicable for images with similar characteristics of lighting" and you're right, we could use another approach, I

would suggest taking a look at colour maps from the Open CV api which detects a change in colour within the region_of_interest and performs depending on the colour, for animals it can be heat, for snow also: http://docs.opencv.org/2.4/modules/contrib/doc/facerec/colormaps.html

Not much to say about this requirement, you have passed it with flying colours and also I have elaborated a little more on the comments above :)

DOWNLOAD PROJECT

RETURN TO PATH