

Finding Lane Lines on the Road

Finding Lane Lines on the Road

The goals / steps of this project are the following: * Make a pipeline that finds lane lines on the road * Reflect on your work in a written report

Reflection

1. Describe your pipeline. As part of the description, explain how you modified the `draw_lines()` function.

My pipeline consisted of 5 steps. First, I converted the images to grayscale, then I did a gaussian blur with kernel size of 5. After this I did canny edge detection with a low threshold of 100 and high of 200. After this I selected just the quadrialteral in the screen where I expected relevant road lines to be. After this I used hough transform to detect lines in the image I set the hyper parameters as `rho = 2` `theta = np.pi/180` `threshold = 15` `min_line_length = 5` `max_line_gap = 20` After all of this then I drew the lines on top of the image.

In order to draw a single line on the left and right lanes, I modified the `draw_lines()` function by filtering the lines into left lines and right lines in seperate lists based on the slope which I calculated based on the two points for each line. If the slope was positive it went in the right if it was negative it went in the left. I filtered out lines with slopes with magnitude less than .5 or greater than .8 to not take into account implausible lines. Finally I averaged the center point for the left and right lines respectively and the slope for the left and right lines respectively. I then had one line for the left and one for the right. I then caluated both lines intercepts with y equal to two-thirds of the screen and the bottom of the screen. I then drew the left and right lines between those two intercepts for each line.

Below you can find the images I generated of the pipeline running.



Figure 1: whiteCarLaneSwitch Result

2. Identify potential shortcomings with your current pipeline

One potential shortcoming would be what would happen when running the challenge video. Due to the sharp bend and the changing in the lighting road markings and lines are not detected properly.

Another shortcoming could be if the car was disorientated and not directly within the lane it may not see any road markings when in reality there are some. So in this case an autonomous car would not be able to move back into the lane.



Figure 2: solidWhiteCurve Result



Figure 3: solidYellowLeft Result



Figure 4: solidWhiteRight Result

3. Suggest possible improvements to your pipeline

A possible improvement would be to average over the last few frames to mitigate against times when the lines are more difficult to detect.

Another potential improvement could be to use a more complicated model for the road marking than a line. A spline curve may be more appropriate.