Finding lane lines on the road

Renyuan Zhang

The goals of this project:

Make a pipeline that finds lane lines on the road using python

Description of the pipeline:

My pipeline consisted of 7 steps:

1. I converted the images to grayscale

2. Then I applied the gaussian blur of kernel size of 3.

3. Using Canny transform to get the edge of the vision.

4. Apply the region of interest to get the polygon region in front of the car.

5. Using hough transform to get the hough lines.

6(a). Use draw_lines() function to get a raw hough lines drawn map.

6(b). In order to draw a single line on the left and right lanes, I modified the draw_lines() function by adding a function to obtain the slope and major segments of the hough lines. And draw the extrapolate lines from top to bottom of the image.

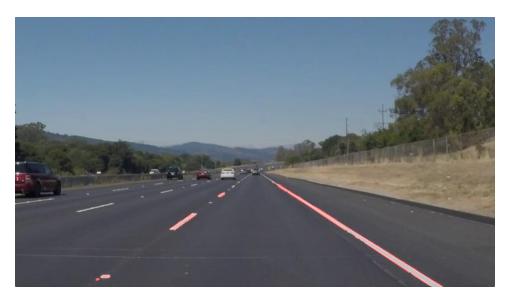7. Apply the weighted_img() funtion to draw the hough lines on the image.
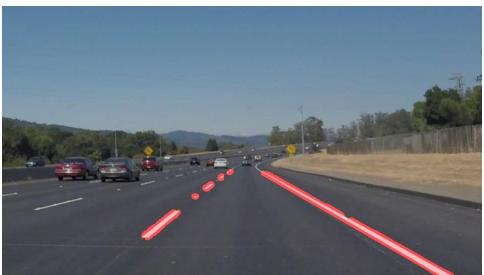
Here are some sample images (raw):

The video is done by using every process image function onto the video frame. The sample video can be viewed as solidWhiteRight.mp4.

Two single solid pipelines are implemented in the video. The example video can be viewed as solidYellowLeft.mp4:

I tried the challenge video and get a convincing result. The video can be seen as challenge.mp4.

Possible shortcomings and improvements:

The shortcomings come that there are multiple segments of lanes. The pipeline detection relies on the road conditions, road directions, sunshade and surrounding cars. So the draw_lines() function in my program should be modified to average of the points along the lane. Additionally, the threshold should be modified to control the image adaptively. This means we need to adjust the threshold based on the road conditions and lighting conditions.

Overall, it is a very useful project in detecting lane lines.