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

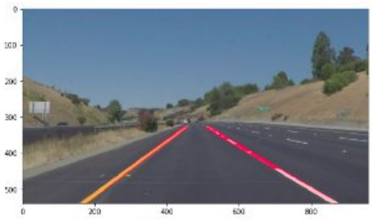
1. Description

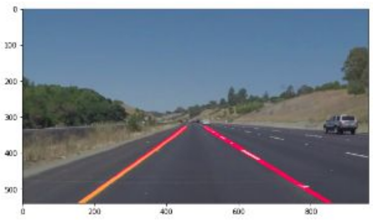
My pipeline consisted of 5 steps.

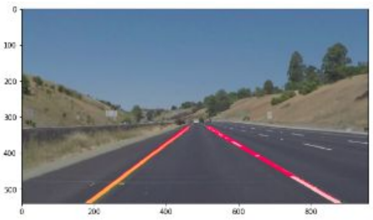
1) I convert the images to grayscale;
2) I apply gaussian filter with 5 x 5 kernel size to smooth the image;
3) I apply canny edge detection algorithm to identity the edges with lower threshold 50 and higher threshold 150;
4) I defines a rectangular to mask the interest area where the left lane and the right lane will be contained within the rectangular;
5) I apply the hough line transform to be able to identify lanes

In order to draw a single line on the left and right lanes, I modified the draw_lines() function by following steps:

1) Divide the lines array obtained from HoughLinesP() into two sublist by whether the gradient of two points is bigger than zero or smaller than zero, let the sublist with gradient bigger than zero be left_lines and the sublist with gradient less than zero be right_lines;
2) Sort the two sublist by the value of their gradient, we sort the leftline list in increasing order and sort the rightline in decreasing order;
3) We append all points with nearby gradient into seperate lists, contour_left for leftline and contour_right for rightline, respectively;
4) We use contour left and contour right to fit a linear line by provoking fitLine() with the least squares method;
5) Finally, we compute the two end point position, one is on the last row of the rectangular and another is the first row of the rectangular which have been defined previously, then we draw a line connecting those two end points as left lane and right lane







2. Identify potential shortcomings

One potential shortcoming is that if we have multiple edges nearby or within our predefined region of interest, it would be difficult to identify the accurate lane just by comparing their gradient;
If the vechicle is driving in a turning point, the line may not be enough to draw the lane, we need to consider using polyline instead;
If the gradient becomes smaller, it might be harder to identify the edge by hardcoding threshold. For instance, under the sunlight, the color of lane and the color of road becomes similar.

3. Suggest possible improvements to your pipeline

One possible improvement is to use polyline instead of line to make the lane more robust to different road environment;
Another improvement is to be able to handle the gradient threshold automatically adapt to the change of environment.