

Finding Lane Lines on the Road

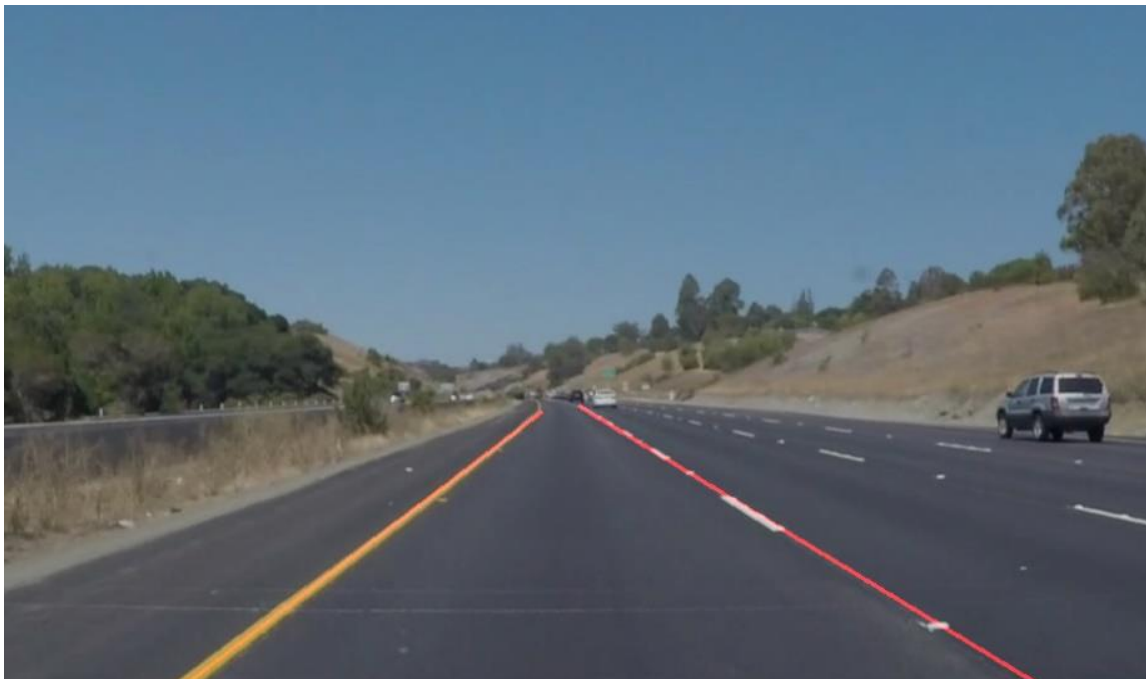
The goals / steps of this project are the following:

- Make a pipeline that finds lane lines on the road
- Reflect on your work in a written report

Reflection:

My pipeline consists of below steps:

1. Converting the images read from different color channel to grayscale.
2. Then applying gaussian blur for smoothing with kernel size 5
3. Then canny edge detection technique is used for edge detection
4. Region of interest is extracted with a quadrilateral mask
5. Then Hough transform is applied on masked edges with below parameters:
 - a. $\rho = 3$ # distance resolution in pixels of the Hough grid
 - b. $\theta = \text{np.pi}/180$ # angular resolution in radians of the Hough grid
 - c. $\text{threshold} = 70$ # minimum number of votes (intersections in Hough grid cell)
 - d. $\text{min_line_length} = 70$ # minimum number of pixels making up a line
 - e. $\text{max_line_gap} = 250$ # maximum gap in pixels between connectable line segments
6. Then I extended the draw_lines function to draw left lines and right lines by their slope. First, I collected all x and y values from left part of the image and then right. Then average slope and constant "b" is calculated from all the points on left and again right.
7. Finally, detected lane lines are drawn on the original image as below:



Potential short comings:

I see that when the lane lines are curvy my algorithm showing some slant lines over the image and not very well detecting the lane line.

Possible improvement:

We need to figure out to extend our pipeline to detect curvy lane lines.

Applying Perspective transform to get a bird view to detect curvature is one possibility.