Write Up - 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

* Tune the parameters in the pipeline to make it work on a live video

<u>**Reflection**</u>

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

My pipeline consisted of 5 steps as follows:

1. Convert the original image to gray image.

2. Apply Gaussian blurring to the gray image to make it more smooth for canny algorithm to detect the edges.

3. Apply Canny function to convert gray image to edges detected based on derivative of intensity at each pixel.

4. Passing the image edges through region masking function to get only the region of interest (area between the two lanes).

5. Apply HoughLinesP function to detect lines (on left and right side) from the masked image by tuning the input parameters and the draw_lines() function.

6. Finally imposed the output from HoughLinesP function onto the the original image to detect the lane lines.

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

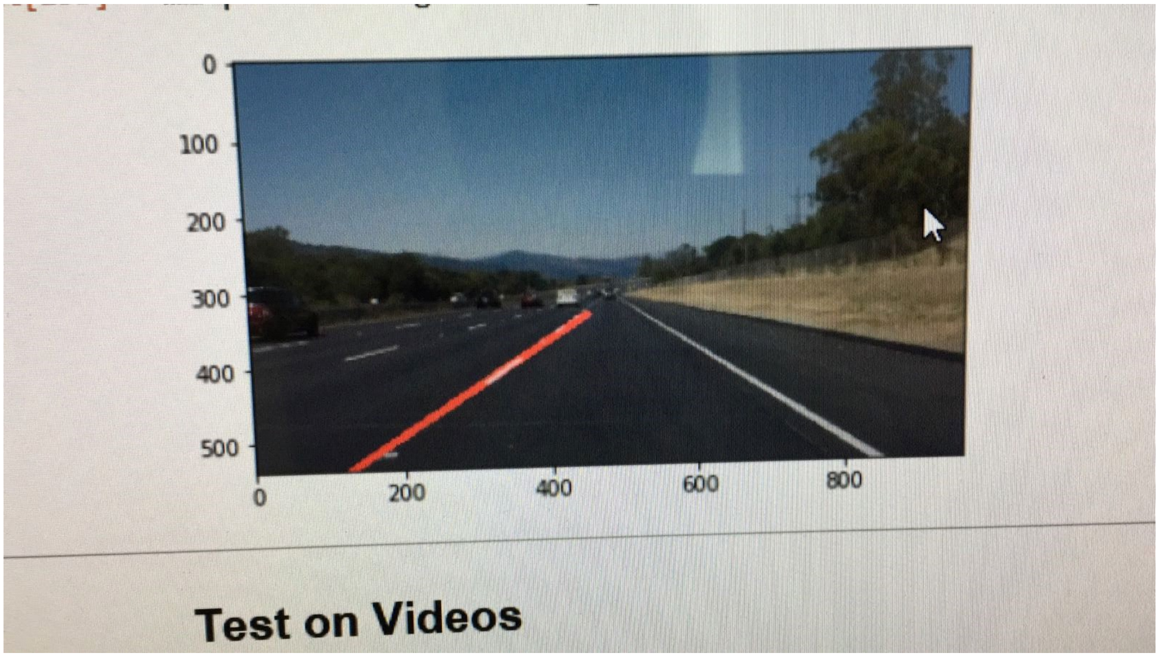
1. After looking at the first run, it felt like there are multiple lines (detected using the Hough algorithm) superimposed to be as lane lines. So something should be done to smoothen thatione lane line (also taking reference from the notes section).

2. Looked at the udacity forums and did some research, finally thought I have to somehow average out the effect of all small lines as one line.

3. Calculated average slope for all the lines and one centroid (average sum of x and y corrdinates of the lines) to form a line equation.

4. Finally using slope and center coordinate, calculated x-coordinate of the two endpoints of the final lane line (passed y-coordinate from the vertices of masking region for both the lanes).

Image below after modifying the draw_lines() function:



Test on Videos

Further tuning of parameters:

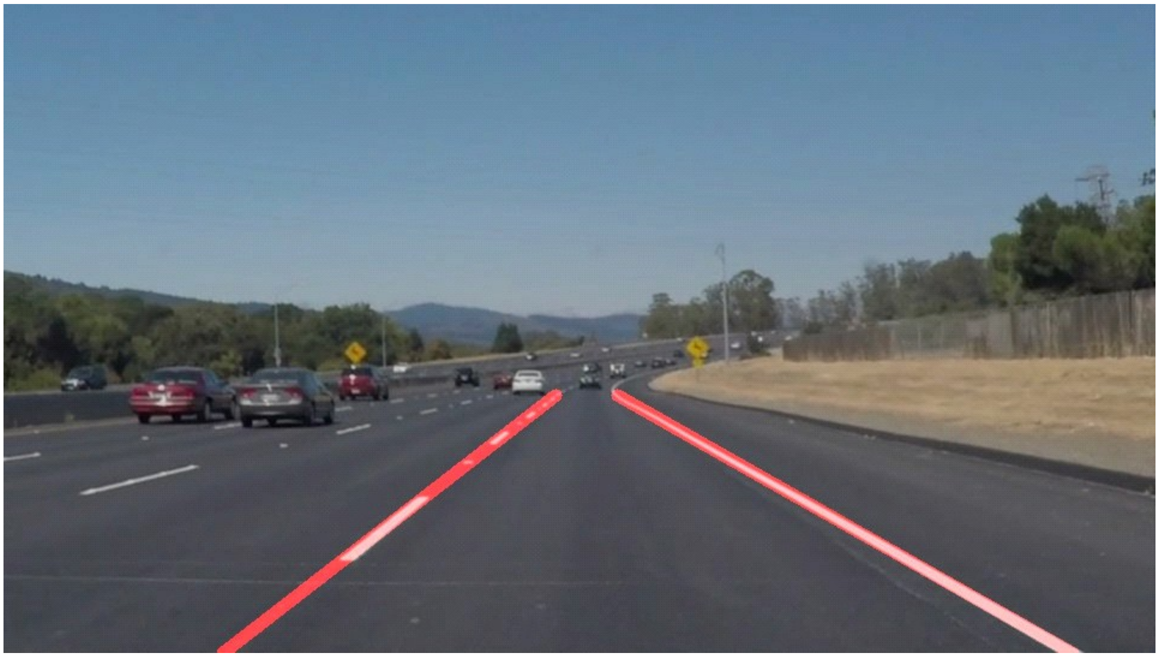
1. After modifying the draw_lines() function, I got nice 2 single lines (left and right) but they were not aligned with the actual lanes. So I had to tune the hough parameters and ranges of slopes to fix it, but it didnt help much as there were lot of combinations to test.

2. I then thought of running a error calc function to calculate slope deviation of my lines from the actual lane lines for different combinations of hyperparamters (like canny, hough parameters).

3. Exported the results into a file and looked at the values where the deviation was minimum (almost close to zero).

4. Then used that combination as inputs to my function to get the lane detection to working.

Image below after tuning the parameters:



*** Identify potential shortcomings with your current pipeline**

One potential shortcoming would be what would happen when the road is too much curved, the lines are going a little off and not able to detect lane properly for that particular frame, as the curve for the masking region no more satisfies a line equation.

Another shortcoming could be that my code is not considering the parameter tuning based on each frame in the video, its jsut using the same parameters and applying it to every frame. Also it behaves the same way for white and yellow color lanes.

*** Suggest possible improvements to your pipeline**

A possible improvement would be to improve on dynamic tuning of the function parameters as in the frame is read in a video.

Another potential improvement could be to have an identifier to better detect a yellow and a white lane line.