

Project 1 - Finding Lane Lines on the Road

Goals of this project:

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

Reflection

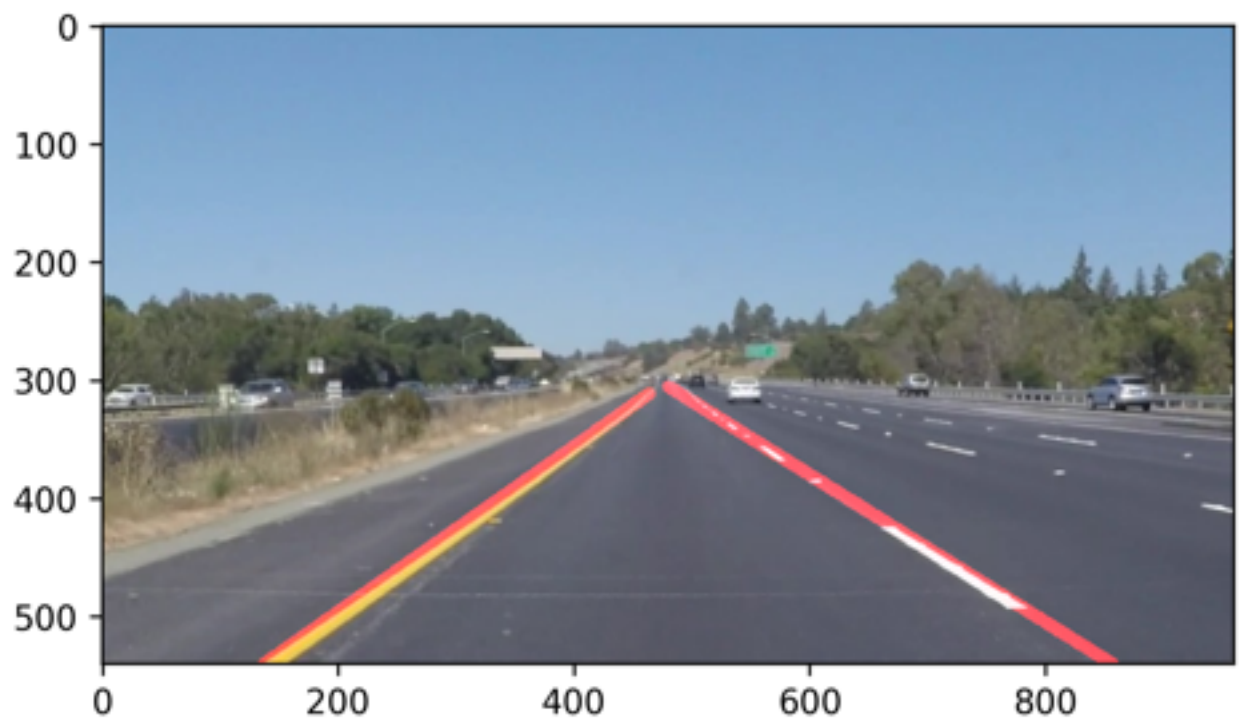
1. Describe your pipeline. As part of the description, explain how you modified the `draw_lines()` function.

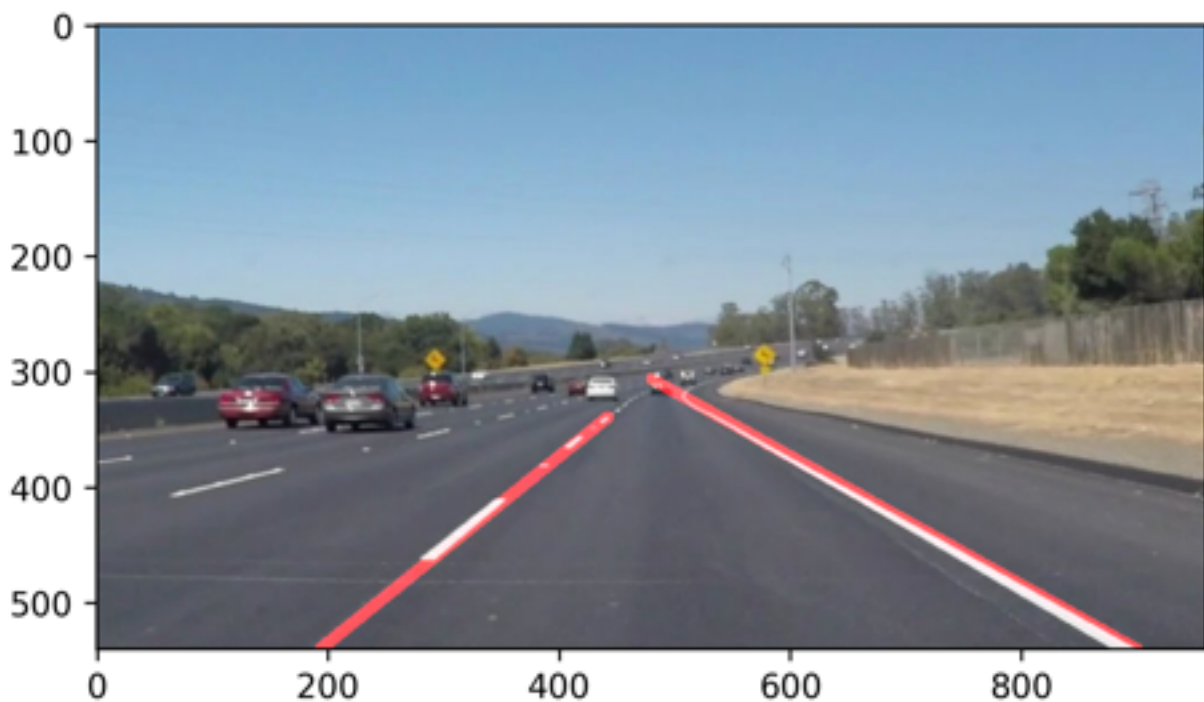
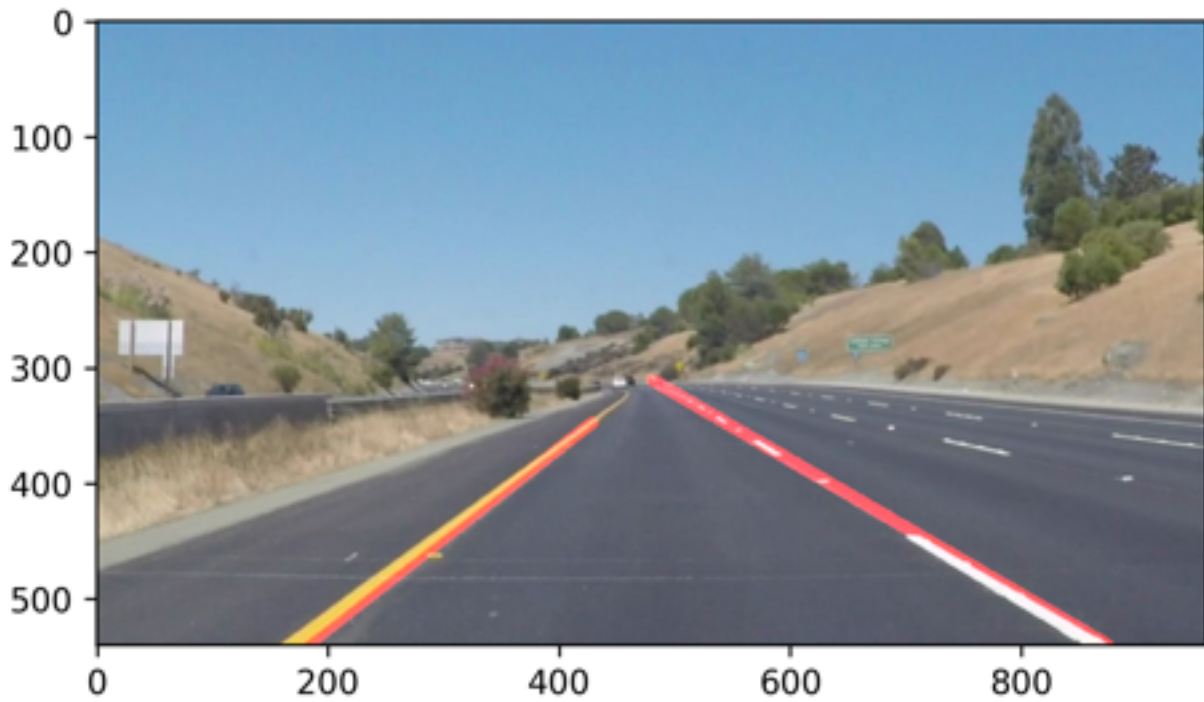
My pipeline consisted of 10 steps, as below:

- Convert color image to gray scale.
- Convert color image to HSV scale. Purpose of this transformation is mainly for yellow lines.
- Create white color mask and yellow color mask. These masks provide the positions of white color and yellow color on the image.
- Apply white and yellow masks to boost white and yellow colors on the gray image.
- Apply gaussian filter (5 X 5 kernel)
- Apply canny detection (threshold [50, 180])
- Apply `cv2.fillPoly()` to isolate the region of the interest. Here I have to emphasize that I apply the `cv2.fillPoly()` twice to isolate two ROIs: region one is for the left lane and region II is for the right lane. I basically split the image two parts based on the mid line of the color image. Vertices of the two ROIs are defined by `vertices_left` and `vertices_right` in the code. Please refer to these two parameters for details.
- Apply Hough transformation on two ROIs, respectively. (rho : 2; theta: 1; threshold: 40; min_line_length: 20; max_line_gap: 200). We have two outputs: `lines_left` based on the left ROI and `lines_right` base on the right ROI.
- Apply `draw_lines` functions to highlight the detected left and right lanes based on `lines_left` and `lines_right`, respectively. The rule is to find out the edge which has the maximum length. Then the slope and pivot of future fitted straight line are determine by the two points from the selected edge with the maximum length from each ROI, respectively.
- Combine the two images with the highlighted lines into one final image.

In step C, I borrowed the concept from Jeremy Shannon (<https://medium.com/udacity/udacity-self-driving-car-nanodegree-project-1-finding-lane-lines-719ac1adbed9>), which creates a yellow mask to enhance the yellow color lanes.

Here are the processed images.





2. Identify potential shortcoming with your current pipeline

I notice that the highlighted lines of current pipeline can be shorter than the actual lanes, especially with curved lines. The possible reason could be the inaccurate determination of either starting point or end point of the fitted line.

3. Suggested possible improvements to your pipeline

A possible improvement would be to figure out a more robust and reliable logic to determine the starting or end point for the fitted line.