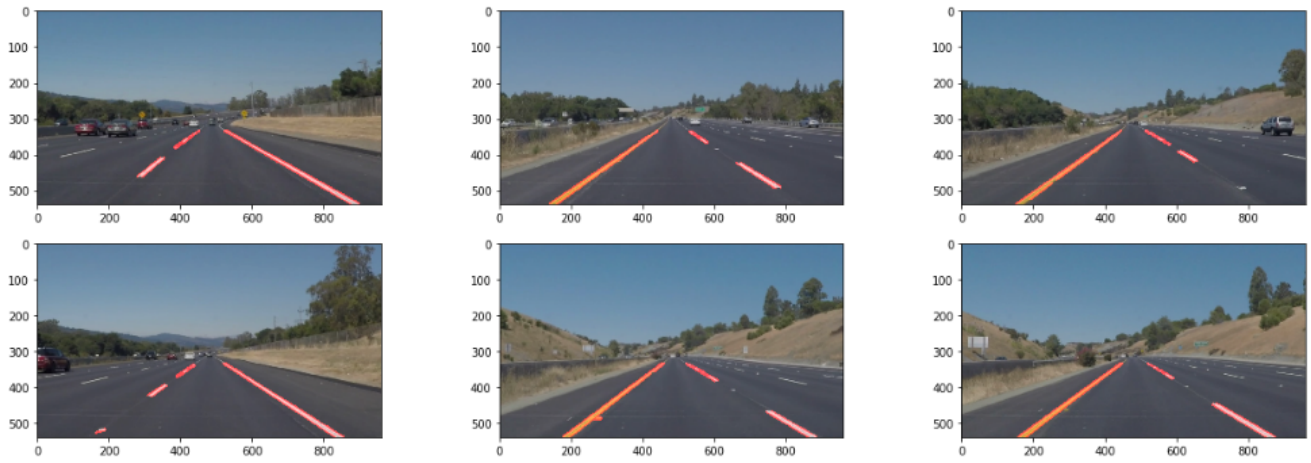
Finding Lane Lines on the Road

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

My pipeline consisted of 5 steps.
The image for each of the steps is included in the P1.ipynb
a) First, the images were converted to grayscale
b) Gaussian Blur is applied to smoothen the gray image.
c) Canny transform was applied to the blur image with low and high threshold as hyper parameters to the image to edges
d) Masked region is then determined by the vertices of a polygon determined through region of interest function. **The vertices defined in my program are independent of the image size**. It works with any image size.
e) Hough transform is applied to get raw Hough lines; again the hyper parameters are chosen such that Hough lines are drawn on lane markers

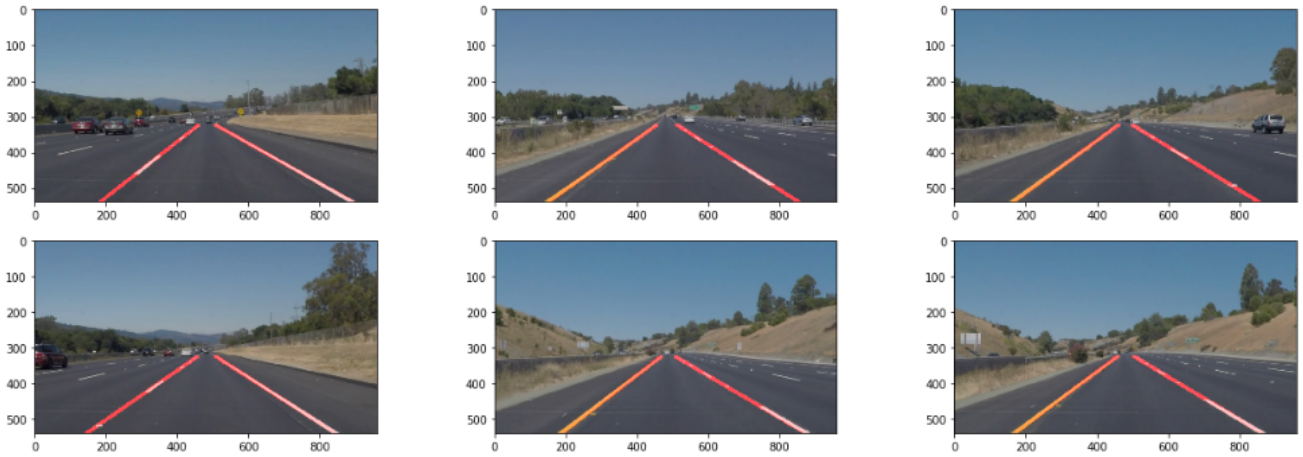
The initial image and the 6 test-images were used to fine-tune the hyper-parameters
Resultant line segments are shown below (also in the folder **test_images_output**)



In order to draw lines on the left and right lanes, a new function **draw_extrapolated_lines** is defined which evolved with different test cases

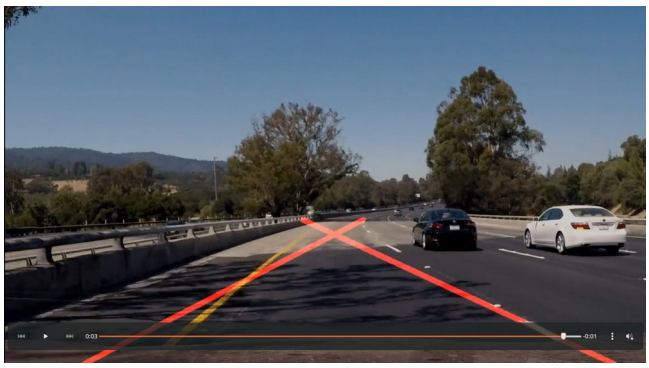
a) Slopes are first calculated for each of the Hough lines. Initially the slopes were used to assign Hough lines to left and right lanes (positive to right lane and negative to left lanes). A linear fit is used for the coordinates of left and right lanes separately to determine slopes and intercepts respectively. The extreme coordinates of the extrapolated lines are bounded by the masked region. Y coordinates of the masked region along with slopes and intercepts are used to determine corresponding X coordinates for left and right lanes separately. Right and left lanes are drawn with these end coordinates.

b) Later the assigning of Hough lines to left and right lane came with additional conditions of slope greater than a certain threshold (another hyper-parameter) to avoid spurious lines (close to horizontal). The resultant extrapolated lines help to identify lane lines as below (also in the folder **test_images_extrapolated**). The idea of the slope threshold was obtained by looking at other approaches for the same problem.



2. Identify potential shortcomings with your current pipeline

The shortcoming of the above pipeline is highlighted in the challenge video where:
a) when the road is very winding: assigning Hough lines to left and right lane could be challenging
b) when the road has sections with high contrast (like old and newly re-paved road as well as dark shadows from the trees)



3. Suggest possible improvements to your pipeline

a) Assigning raw line segments to left and right lanes on winding roads could be better done by using spline fits of the mid-points of the lines ensuring that the distance between the spline fits is constant.
b) Converting to a gray scale image is not good enough as the canny edge and hough transform would provide spurious raw lines as below.

Gradients for each channel could be calculated, and canny edge transform can be used to find gradients for equivalent white and yellow colors (as the lanes are yellow or white)