Write-up for Finding Lane Lines on the Road

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

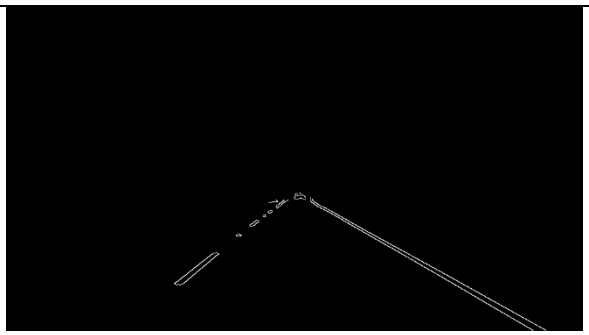
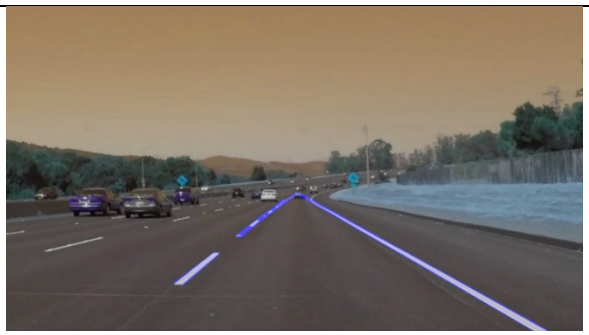
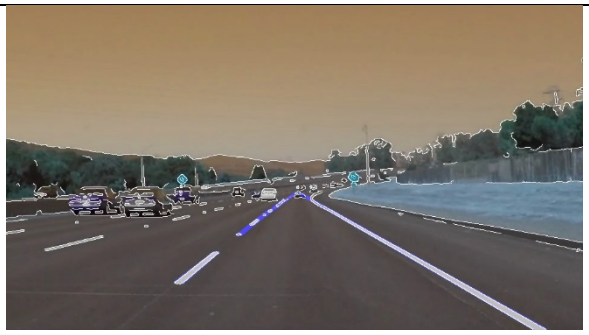
Reflection

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

My pipeline consisted of 7 steps

1. Read Image
2. Obtain grayscale image
3. Define a kernel size and apply Gaussian smoothing
4. Define Edge Detection Parameters and run Canny Edge Detection
    a. Low Threshold = 60
    b. High Threshold = 120
5. Defining a four sided polygon to mask
    a. Vertices are (0,960), (470, 310), (490, 310), (960, 560)
6. Define the Hough transform parameters and ran Hough Transformation on edge detected image
    a. Rho = 2
    b. Theta = 1 Radian
    c. Threshold = 40
    d. Max Line Length = 50
    e. Min Line Length = 25
7. Create a "color" binary image to combine with line image

1. solidWhiteCurve	
2. gray_solidWhiteCurve	
3. blur_gray_solidWhiteCurve	
4. edges_solidWhiteCurve	

5. masked_edges_solidWhiteCurve	
6. line_image_solidWhiteCurve	
7. lines_edges_solidWhiteCurve	

2. Identify potential shortcomings with your current pipeline

I see at least 2 potential shortcomings in the current pipeline

1. Hardcoded Parameters – for Canny Edge Detection, Masking and for Hough Transformation
2. The pipeline may not handle acute curves and wide black & white crossings (a.k.a. Zebra crossing)

3. Suggest possible improvements to your pipeline

In order to overcome the above shortcomings, in future I'd like to apply one or more of the following approaches:

1. Use ML techniques such as gradient descent to improve the parameters for various transformations.
    a. Although to use gradient descent, we would need to fine tune the hyper parameters such as learning rate.
2. Hough Transformation for detecting non-linear surfaces (circles, spheres, etc.)