**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 5 steps.

1. I converted the images to grayscale, then I

2. Use a gaussian kernel to filter the image.

This helps in getting rid of noisy parts of the image which makes the next steps more reliable

3. Perform canny edge detection.

This step basically detects the edges in the image with the help of the image gradient and hysteresis

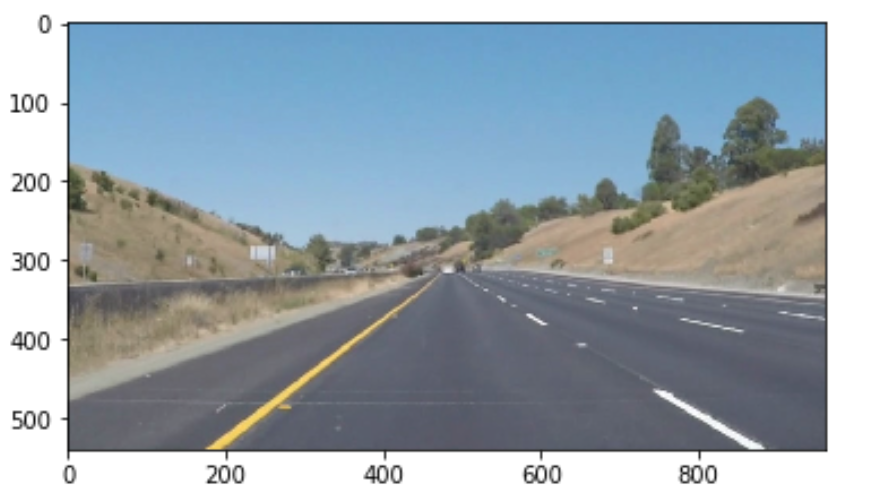
4. Mask the region of interest

5. Use hough transformation to find lines from the edges.

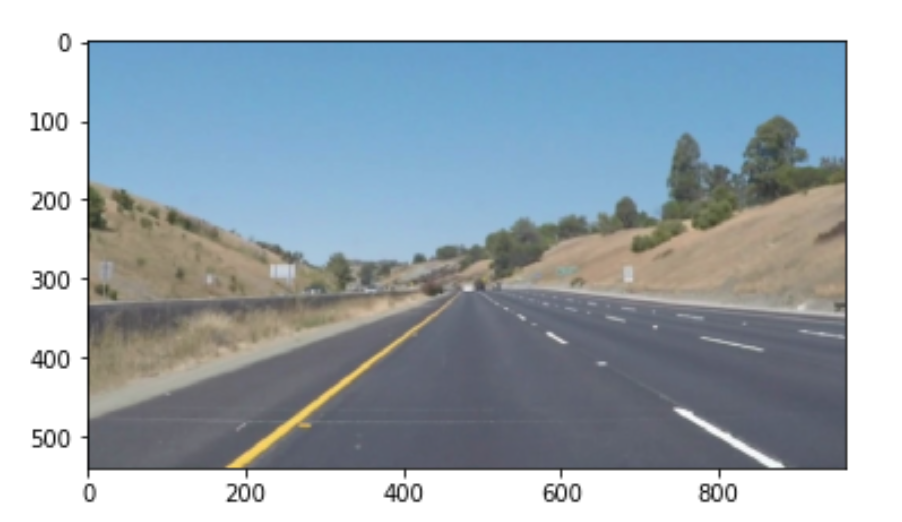
Transforms each point to a line in hough space where the intersection of these lines shows the presence of a line in image space.

Finally Draw straight white lines on the lanes in the image

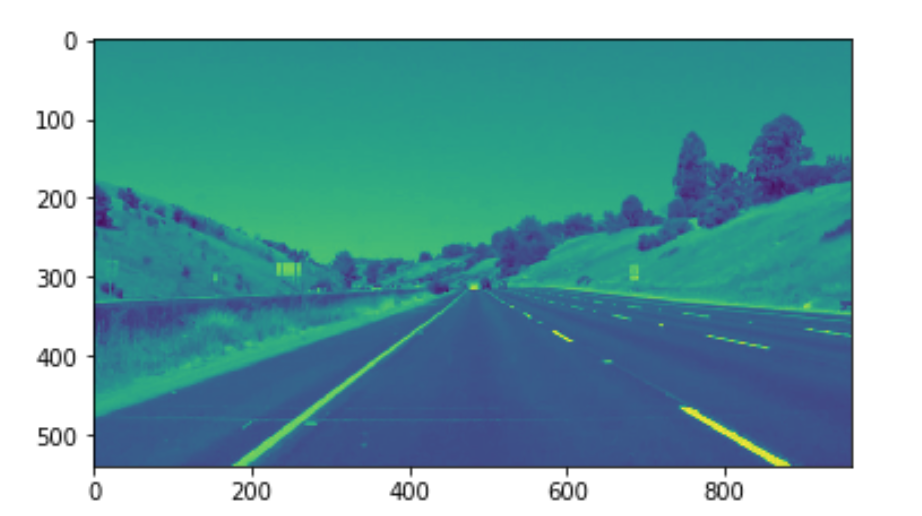
Original Image



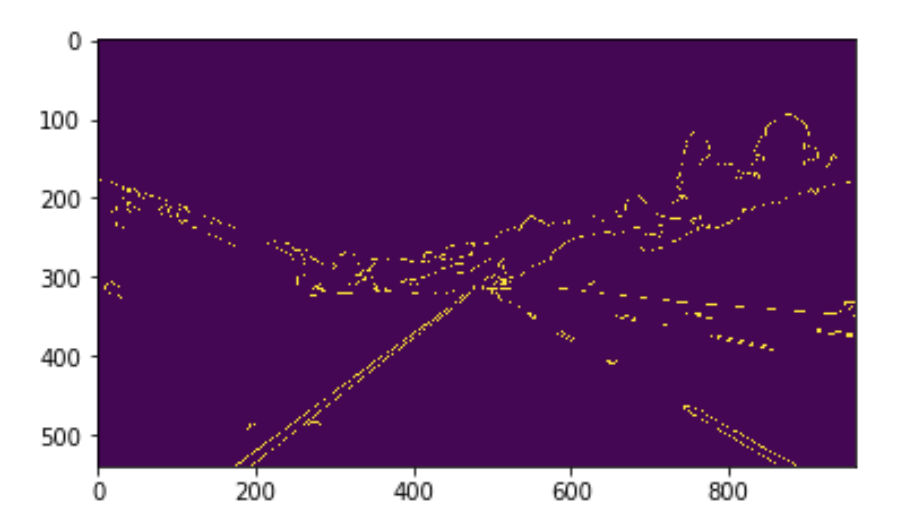
Blur\_Image



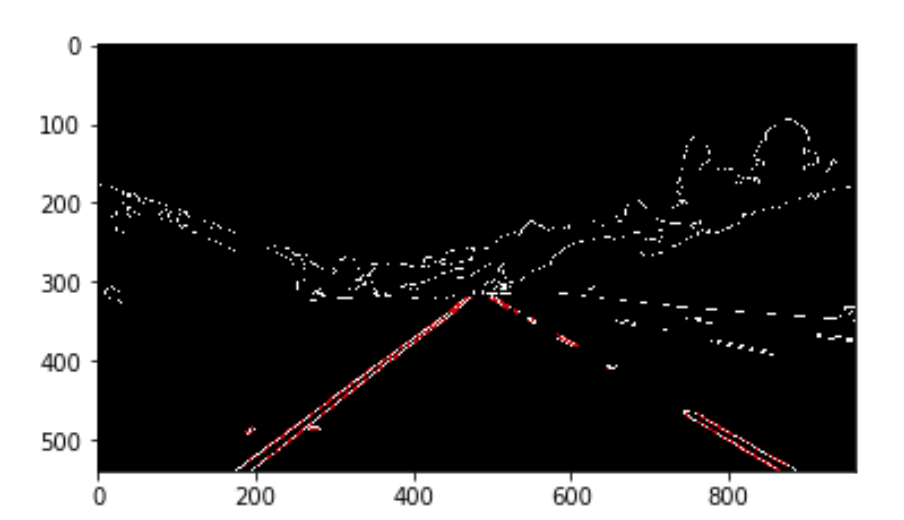
Gray scaled Image



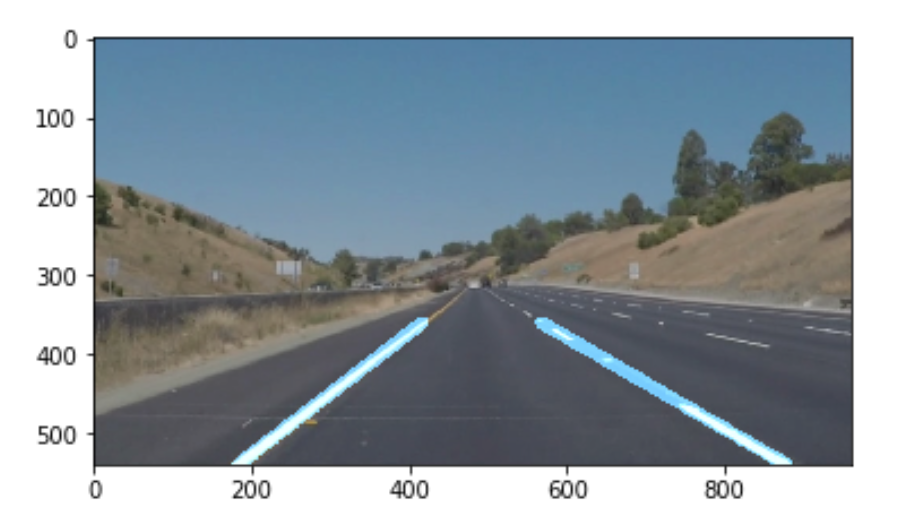
Canny Edge on gray scaled blur image



Hough Lines



Connected lines Image



### 2. Identify potential shortcomings with your current pipeline

- Add color thresholding

- Not good correlating HSL/HSC

- Might not work effectively in different kinds of lighting conditions i.e. real life road conditions

- Not good if there is a car ahead/ actual road conditions

- Won't be that effective on turns/curves/bumps

### 3. Suggest possible improvements to your pipeline

1. perfectly tweaked parameters for the Hough transformation

2. a non-linear line fitting for detecting curves more reliable

3. my smoothing window was set to 10, which is around 0.4s. This can eventually lead to problems in fast transitions.

4. neural network/a deep learning model