

****Finding Lane Lines on the Road****

[//]: # (Image References)

[image1]: ./examples/grayscale.jpg "Grayscale"

Reflection

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

My pipeline consisted of below steps:

I converted the image to grayscale
Applied Gaussian smoothing
Applied Canny for edge detection
Chose my region of interest
Using hough transform, all the lines were detected in the region of interest

Modifications to the draw_line was to get slopes closer to the right and left line. After getting all the left lines and right lines, I extrapolated all the line to the boundary edges and the center. I ignored the lines which ends are out of boundary. I took averages of all the left line and right lines separately so that I have only one extrapolated left line and one extrapolated right line. After I drew the line in red with thickness 2 and imposed it on original picture.

![alt text][image1]

2. Identify potential shortcomings with your current pipeline

Noise removal wasn't completely effective. In the challenge video, there were lines going little haphazard too.

3. Suggest possible improvements to your pipeline

A possible improvement would be to predict better lane lines and utilize past information