Project 1 : Lane Detection:

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

My pipeline consisted of these key features :

Grayscale: This converts the colored image to gray scale format

Gaussian Blur: I added the Gaussian filter to remove the noise from the image.

Canny: Used canny function to detect the edges in the image

Region of Interest: Masked the image with a selected region of interest using fillPoly function.

Hough Lines: Used the Hough transform function to create Hough lines on the image. This created line segments on the lanes in the images.

Line Extrapolation: To extrapolate the Hough line segments, I used the average of slope and intercept for each line segments. This resulted into lot of noise as there were many small line segments. To overcome this issue, I gave more weight to the longer line segment by multiplying it with the length of the line segment. Also, left and right lanes were detected by using the slope. If the slope is negative then the line segment represents the left lane.

2. Identify potential shortcomings with your current pipeline

My pipeline doesn’t perform that well on the images which has curves. The lanes starts to intersect with each other.

3. Suggest possible improvements to your pipeline

Curved lanes are affecting the line extrapolation function. If more controlled weights are applied to each segment then it might resolve the current issue.