PROJECT 1 WRITEUP: LANE LINES DETECTION

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

My pipeline consisted of 5 steps. First, I converted the images to grayscale, then I applied canny edge detection, then I applied a region of interest, then I applied Hough Lines transformation, and finally I draw the detected lane lines.

In order to draw a single line on the left and right lanes, I modified the draw_lines() function by first splitting the lines into lines belonging to the left lane and lines belonging to the right lane, and that is by relying on the slope values. Then for each group of lines, I fitted a line using polyfit() and I computed the average slope and intercept values. Finally, I found the start and end points for each line and I draw the lines on the video.

2. Identify potential shortcomings with your current pipeline

One potential shortcoming would be what would happen when the lane lines are curved since I don't think the pipeline does well with very curved lanes.

Another shortcoming could be when there is shadows and noise in the image, the pipeline won't detect the edges very well.

3. Suggest possible improvements to your pipeline

A possible improvement would be to create a polynomial curve instead of a line to fit perfectly with curved lanes.

Another potential improvement could be to speed up the image_process() function performance to fit well with real world situations.