

Pipeline Description (With input image solidWhiteRight.jpg)

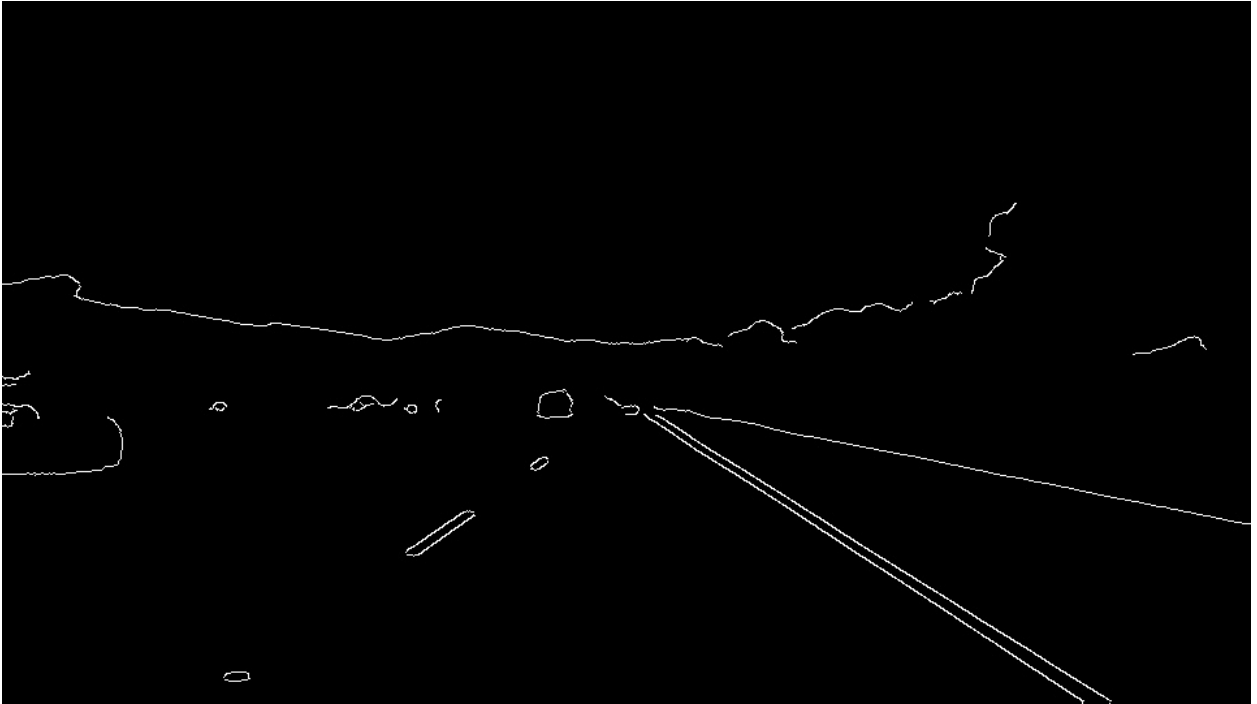
1. Grayscale



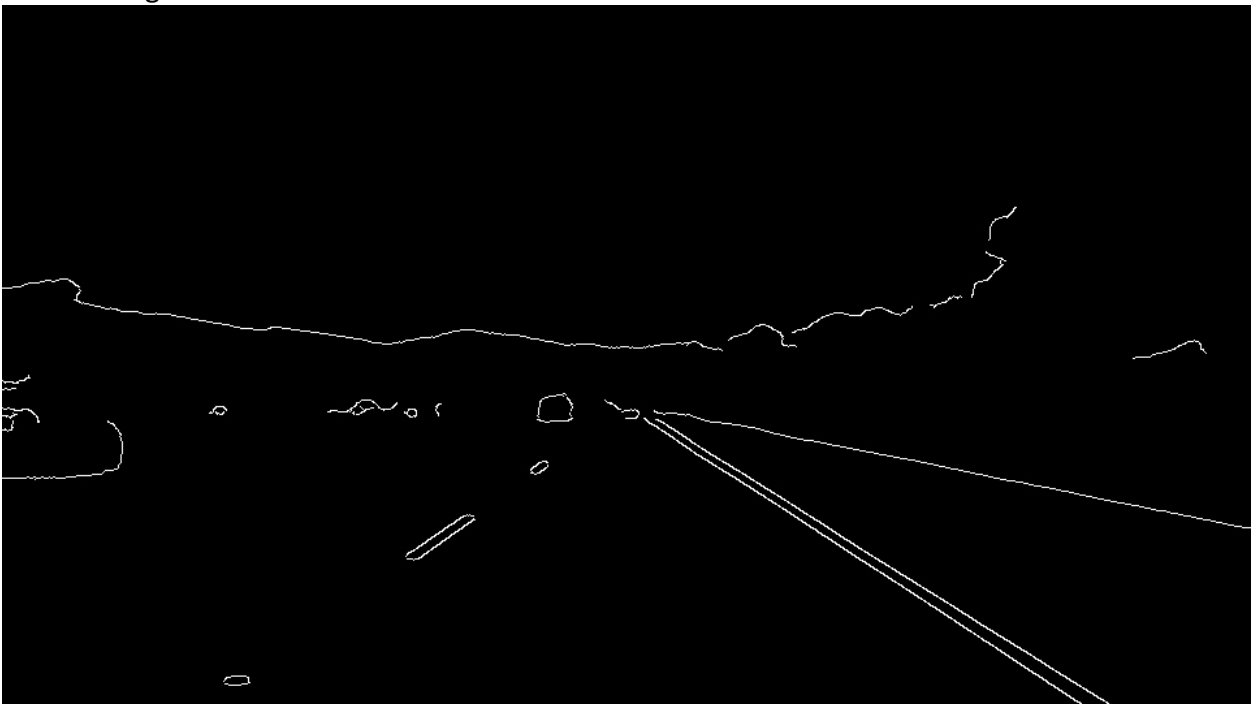
2. Gaussian blur (kernel size=15)



3. Canny edge detection (low threshold = 28, high threshold=115)



4. Masked edge



5. Hough Transform (rho = 1, theta = pi/180, threshold = 30, minimum line length = 30, maximum line gap = 250)

To Improve draw line draw_lines function, I separate line segments by their slope $((y_2 - y_1) / (x_2 - x_1))$ to decide which segments are part of the left line vs. the right line. Then for each side, I call draw_single_line function by passing the x-coordinates(x_array) and y-coordinates(y_array) of each line, and call np.polyfit(x_array, y_array, 1) to get m and b values for $y = mx + b$ for the line to be drawn in the end



Potential Shortcomings with my Pipeline

I found out from the solidYellowLeft.pm4 video, that some pieces of horizontal lines can be falsely identified as lanes. This makes me aware that current pipeline can identify lines with wrong directions within the mask range as lanes

Possible Improvements to my Pipeline

I want to find out a way to detect directions of lines, and only lines within certain range of directions can be recognized as lanes. One way I think of is to check slopes of lines from Hough Transform, and only pass/draw lines that with comparatively larger absolute slope values