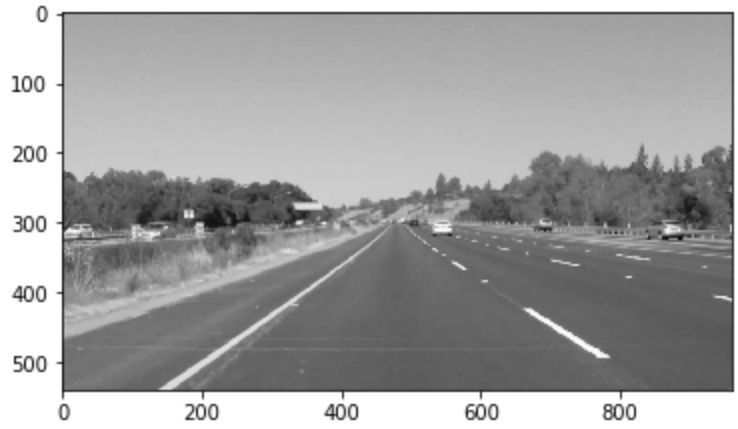
Finding Lane Lines on the Road

Goals/Steps:

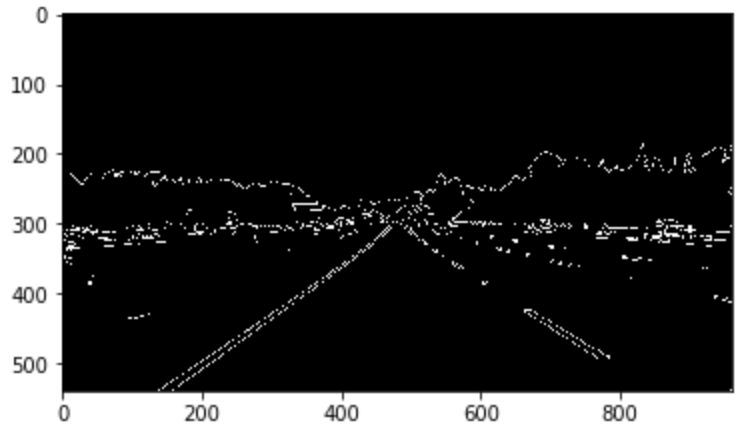
- Make a pipeline that finds lane lines on the road
- Reflect on your work as a written report

Reflection:

My pipeline consisted of 5 steps:
Getting the black and white version of an image by Grayscaling:

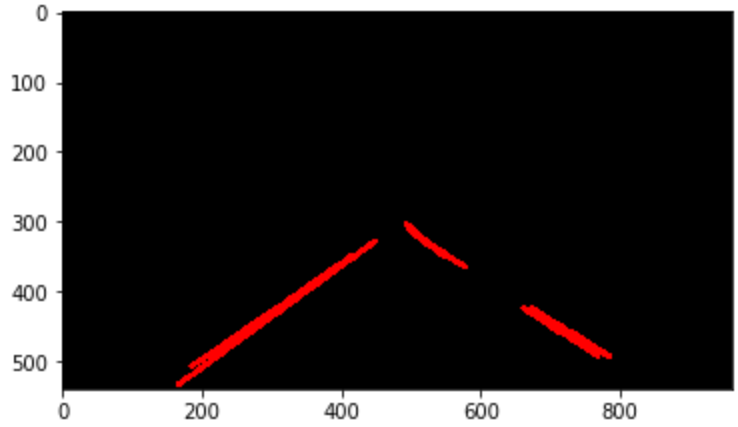


Applying Gaussian Blur and Canny Edge Detection to identify the edges in the image:

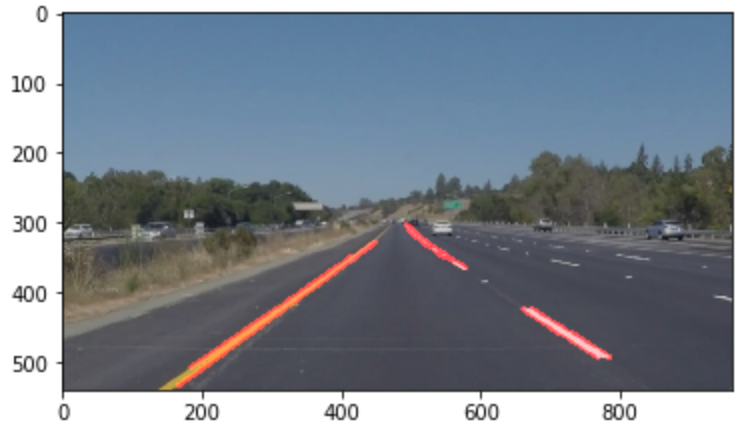


Defining vertices on the image to block out everything except the area of interest:

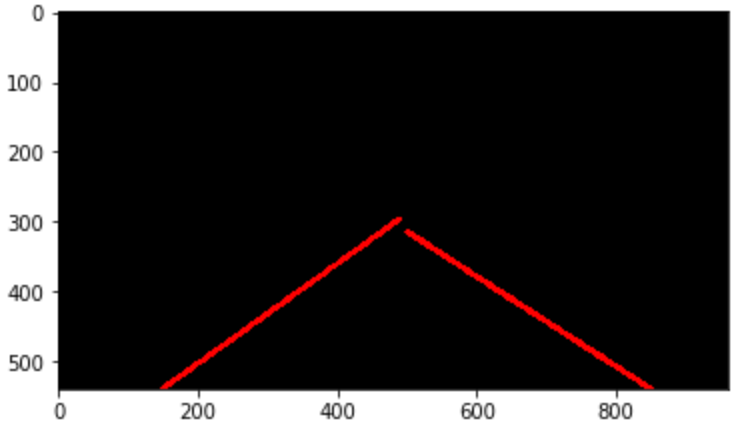
Using Hough Transformation to draw lines on the detected line segments on the image:

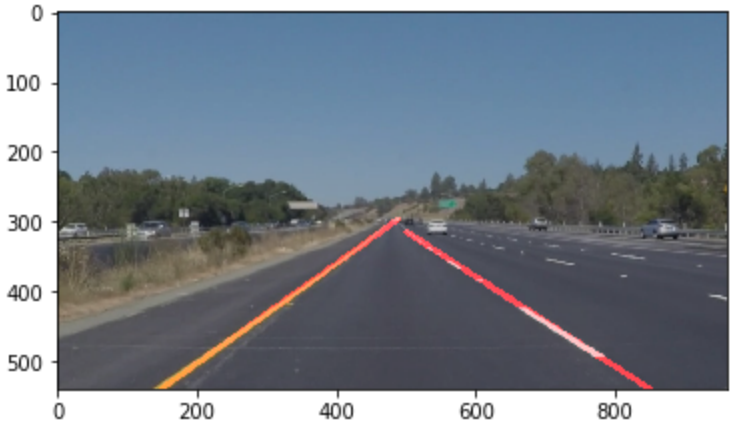


And finally, applying these lines to the original image:



However, in a few more steps, I was able to draw single lines on the right and left line by using the slope to identify which was the right and left lane. I used the numpy functions polyfit and poly1d to define the parameters x1, x2, y1, and y2, and drew a line through those points using the slope. This was the final result after changing the draw_lines function:





Applying this to video is easy, since a video is just a bunch of images!

Potential Shortcomings:

A potential shortcoming would be when the lines are significantly curvy or not straight enough to a point where the line drawn would not suffice. Another would be where if there was something in the way of a lane that would prevent the computer from detecting it.

Possible Improvements:

One possible improvement would be to modify the Draw_lines function even more so that it would also curve lines to better fit the lanes. Another would be maybe to modify the Hough transformation function so that the lines would be better predicted onto the image.