Finding Lane Lines on the Road - Reflection

Modification of the draw_lines() function

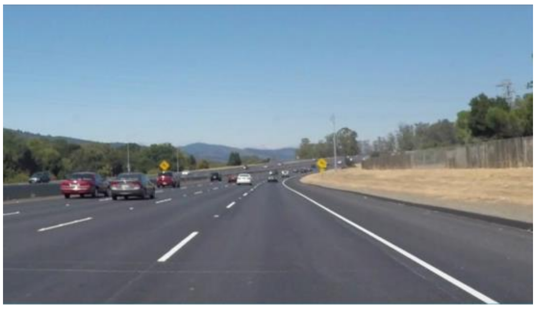
My Lane finding pipeline consists of 6 steps, as follows:

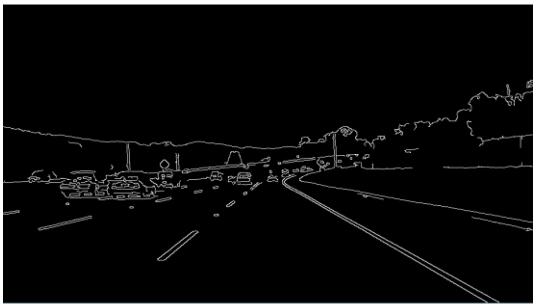
1. Convert image to grayscale
2. Blur the image using a Gaussian Kernel of size 5 and standard deviation of 1
3. Obtain edge-map using Canny Edge Detection
4. Retain edges that fall inside the region of interest
5. Apply Hough Transform to edge-map
6. Draw the lines on the image

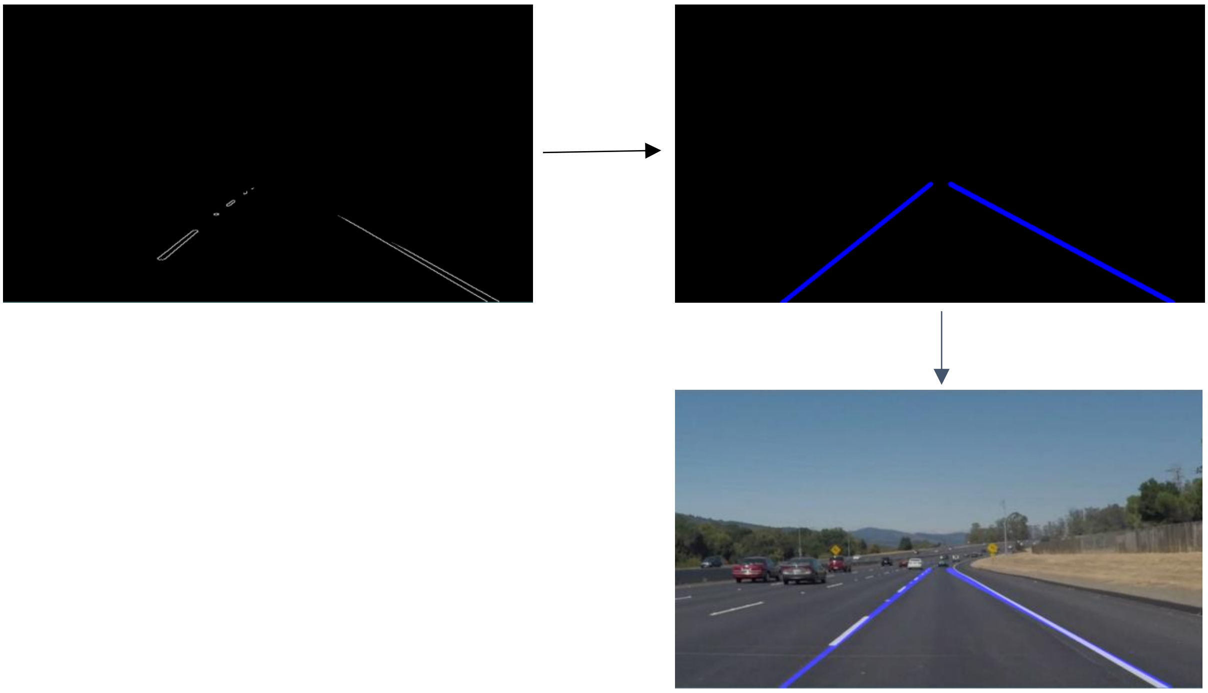
To draw the lines on the lane-lines I first separated the lines that belong to the left and the right sides. I did this based on the slopes and their location on either half of the image i.e. if a line was located on the left half of the frame, it was classified as being a line on the left and so on.

The top and bottom coordinates for these lines were extracted and the slope of this line was computed. This slope was then used to extrapolate the line from top to bottom of the ROI (region of interest) using the extreme y-coordinates (row indices) of the ROI.

This line was then drawn on the image to mask the lane-lines.







The above diagram is a visual example of the pipeline with the last step of overlaying the lines on the original image.

Potential Shortcomings

One shortcoming in my pipeline is that the lane lines that are drawn are not very stable and wobble a little.

Other possible shortcoming is that if any object falls inside the ROI can cause major distortions and problems in accurately detecting the actual lane-lines. This is because currently the lines are being detected using an edge-map and applying Hough Transform to obtain lines without any set of rules that make sure the lines do belong to the lanes.

Possible Improvements

To reduce wobble I'll have to have some consistence between consecutive frames such as averaging the lane-lines between them.

In order to make sure the lines belong to the lane lines I can either apply some hard constraints on the slopes of the lines being considered, their location in the frame and/or make the ROI more complex such that it allows only the area with lane-lines to analysed. This is an impractical solution as the real world situations are more chaotic. I would rather fit in an object detection and classification framework which assists the lane detection and makes sure that the lines being considered do belong to the lane lines.