Finding Lane Lines on the Road

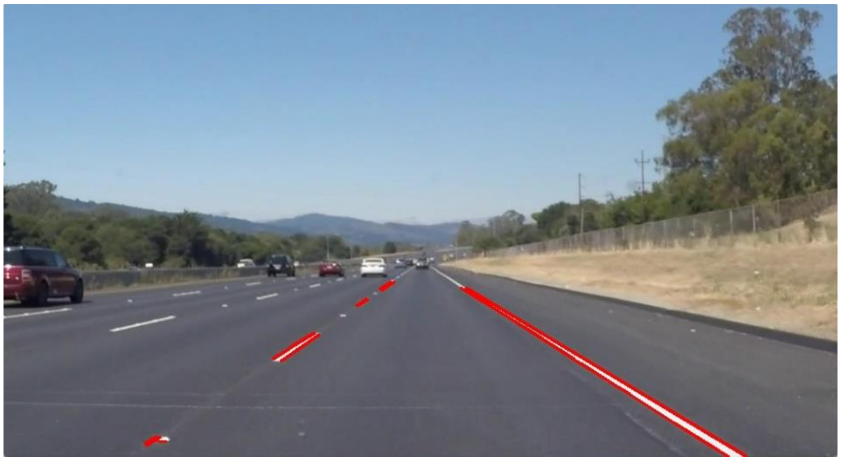
The goals / steps of this project are the following:

* Make a pipeline that finds lane lines on the road

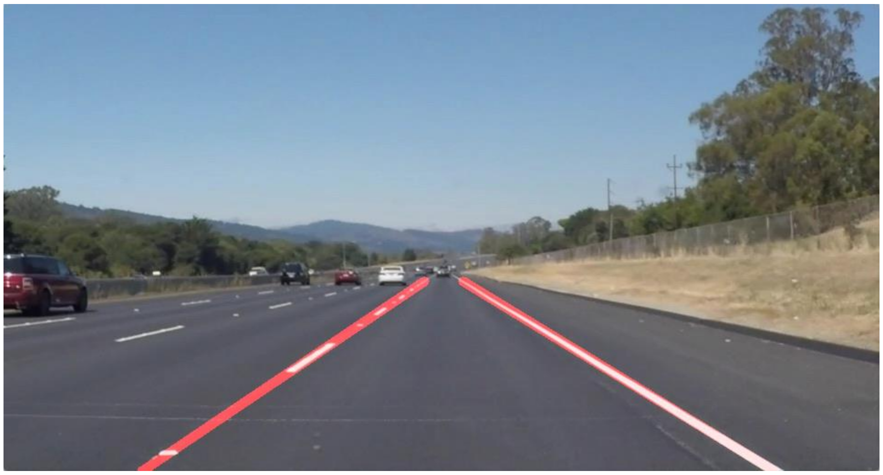
* Reflect on your work in a written report

A Road image is provided:

As part of the project, the Lane lines are to be identified by drawing lines on the lanes as follows:



Next, the disconnected lane lines are to be joined to form a straight lane line

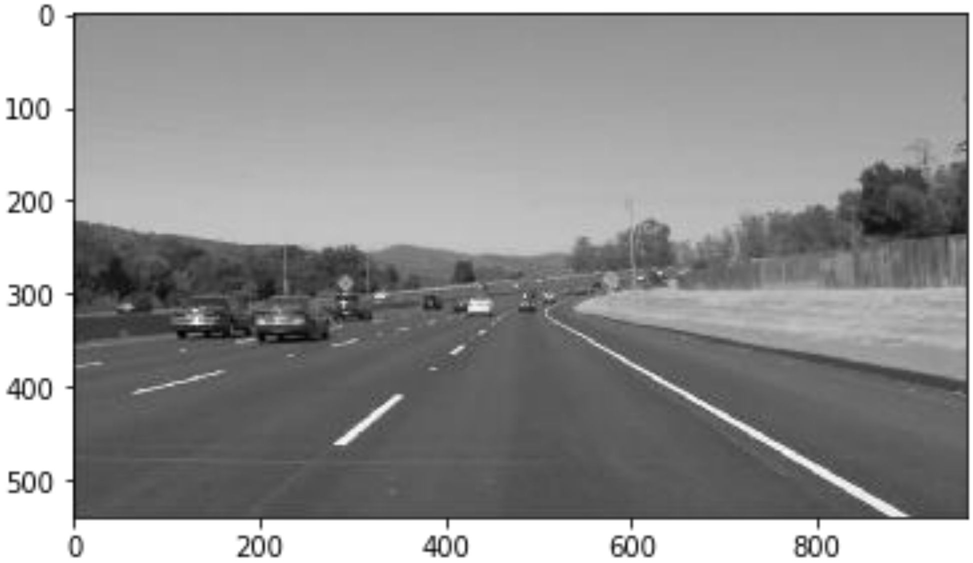


Reflections:

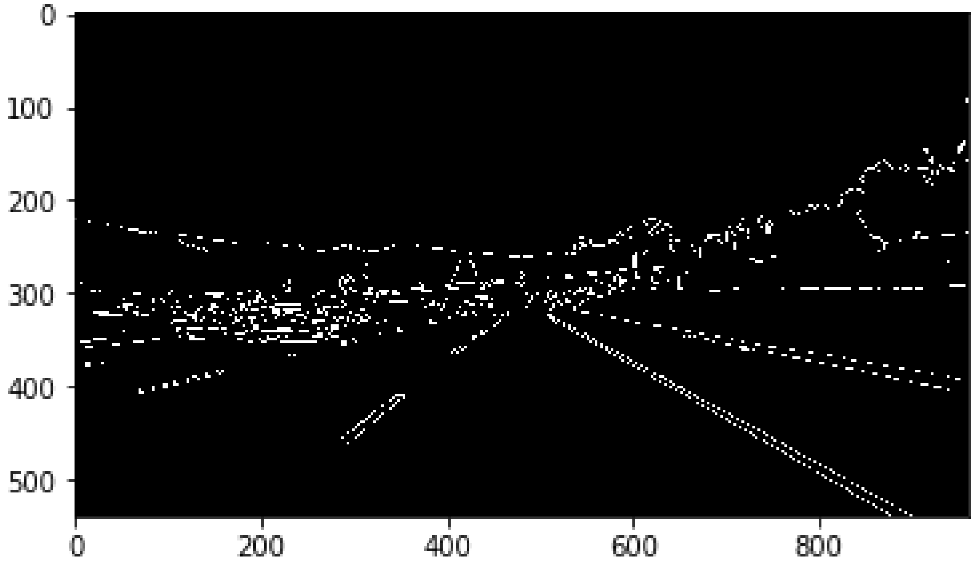
Description

The pipeline consists of six steps :

- Gray Scale Image Conversion using **cv2.cvtColor** method.

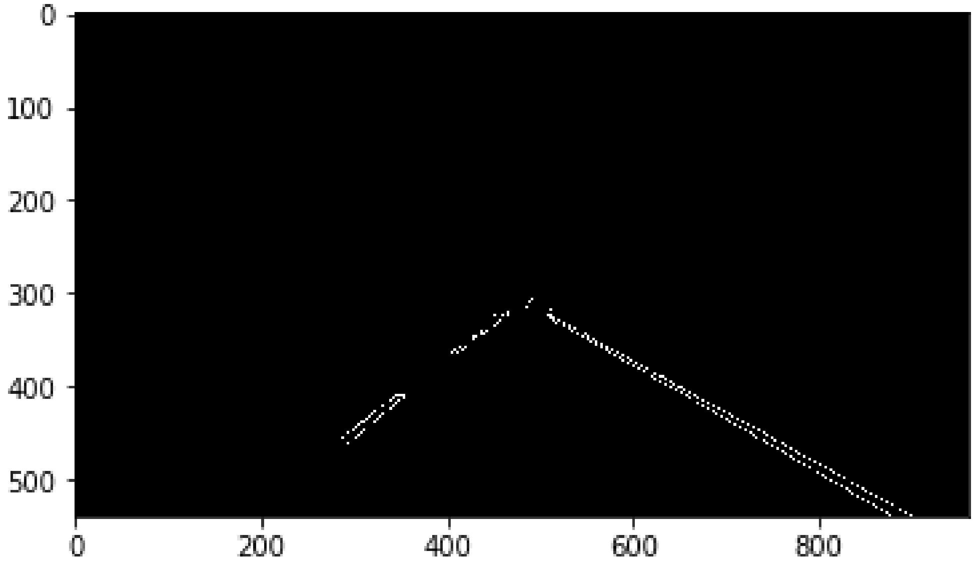


- Apply a Gaussian Blur to the given image using **cv2.GaussianBlu**r method. Kernel Size value 3 is used.
- Then Canny Edge Transformation is used to detect edges using **cv2.canny** function. The pixel values chosen for detecting edges have a low threshold of 50 and a high threshold of 150.



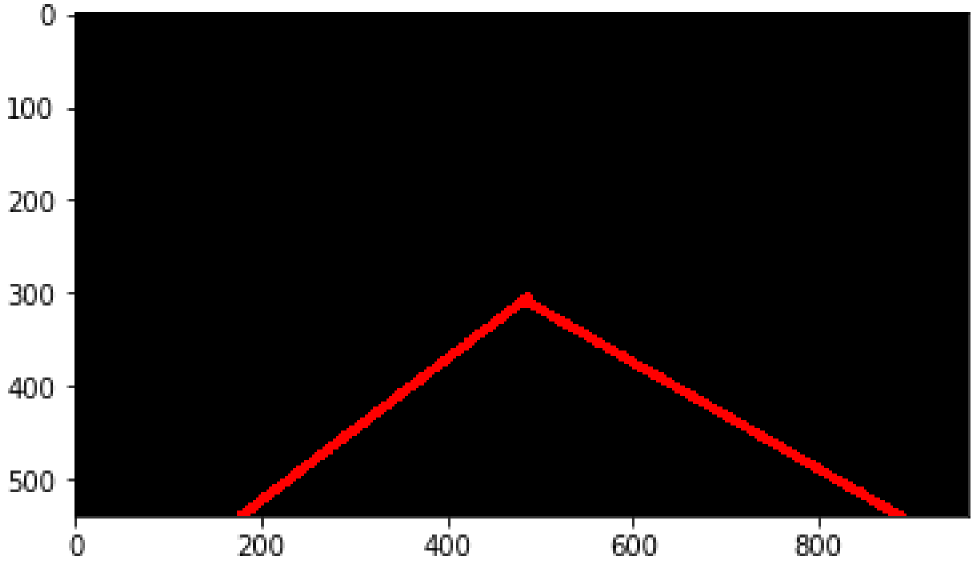
Canny Edge Transformed Image

- Then we eliminate the parts which we are not interested in. Region Masking is used to cover only the road part of the image using **region_of_interest** function. A triangular mask is taken for defining the region of interest.

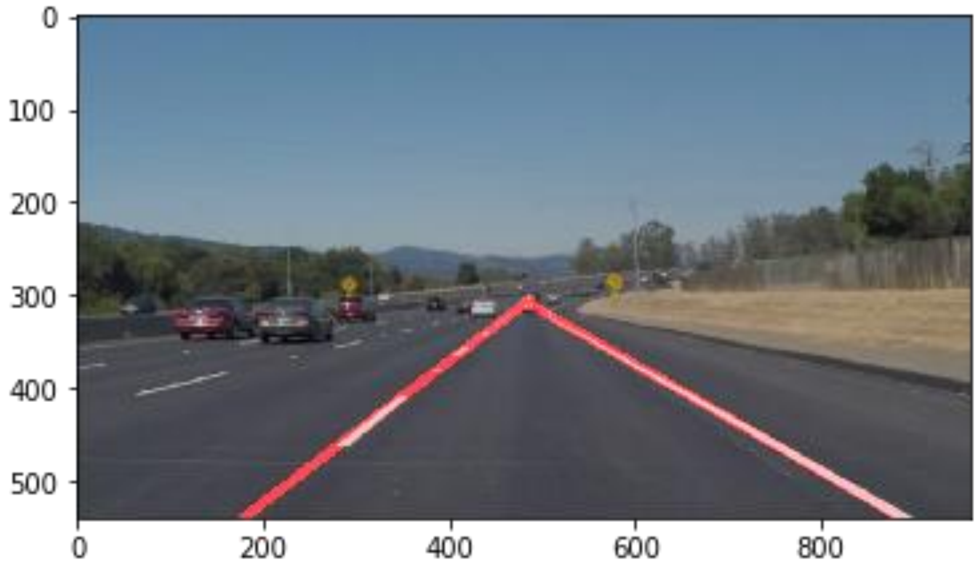


- Next, Hough Transformation is used on the masked image to draw lines on the edges detected using **cv2.HoughLinesP** function. The discontinuous edges are joined with the help of extrapolation technique used in draw_lines function.

- The **draw_lines** function implements extrapolation(guess data points beyond the range of the data set).For this, we segregate the discontinuous lines as part of the left lane or right lane based on the slope of the lines. Then using **numpy.polyfit** function we find the average value of slope and intercept of left lane and right lane. Then **numpy.poly1d** function is used to find the value of y=mx+c. Then cv2.line function is used to draw the continuous lines as follows.



- Then the output of hough transformation is merged with original image using **weighted_img** function to represent the lines on it.



Potential shortcomings and Suggestions:

- Lines shake a lot on the video and therefore, better way to average the lines should be done.
- Line size needs to be improved.
- Lines seem to move away for the curvy edges and does not give a proper lane detection.
- Hard coded values used for region masking could be improved.