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

Pipeline

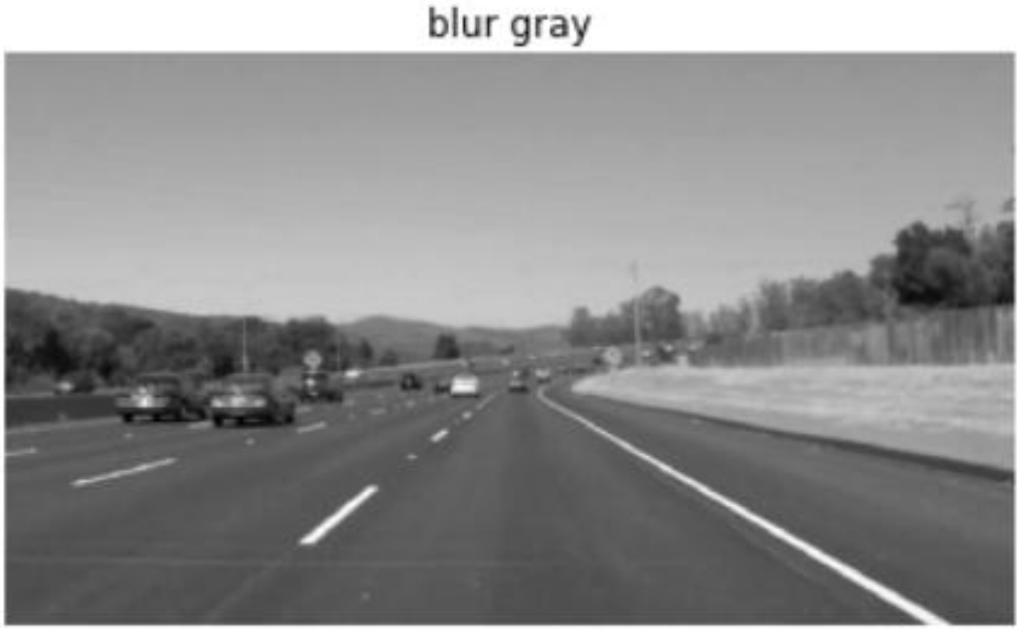
I created a pipeline of 6 steps.

#1

Convert the image to grayscale using the provided function “grayscale”.

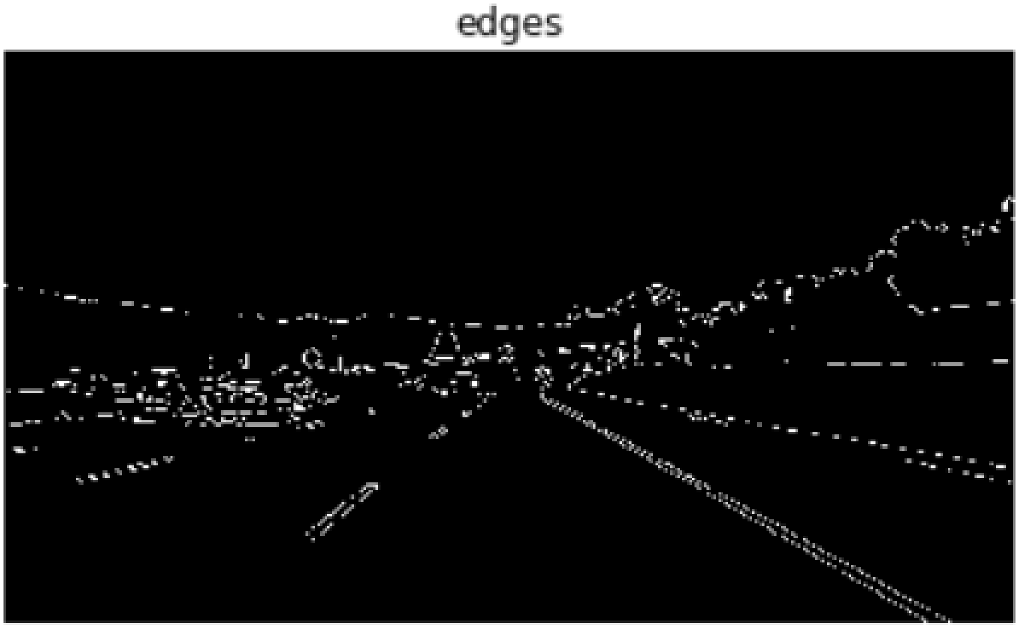
#2

Blur the image using a kernel_size of 5 with the provided Gaussian filter “gaussian_blur” function.



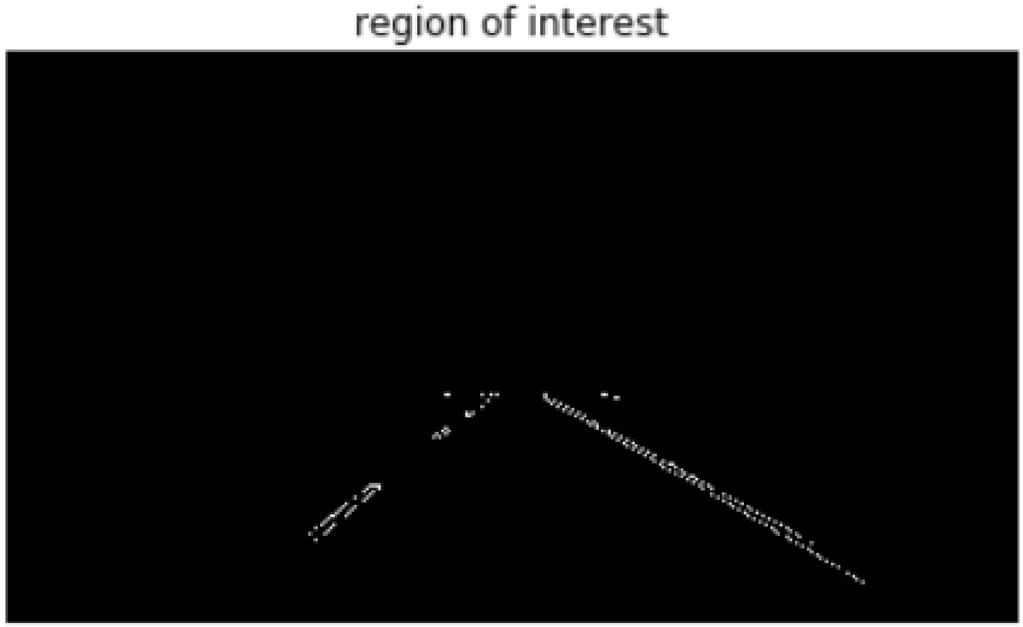
#3

Appy Canny edge detection



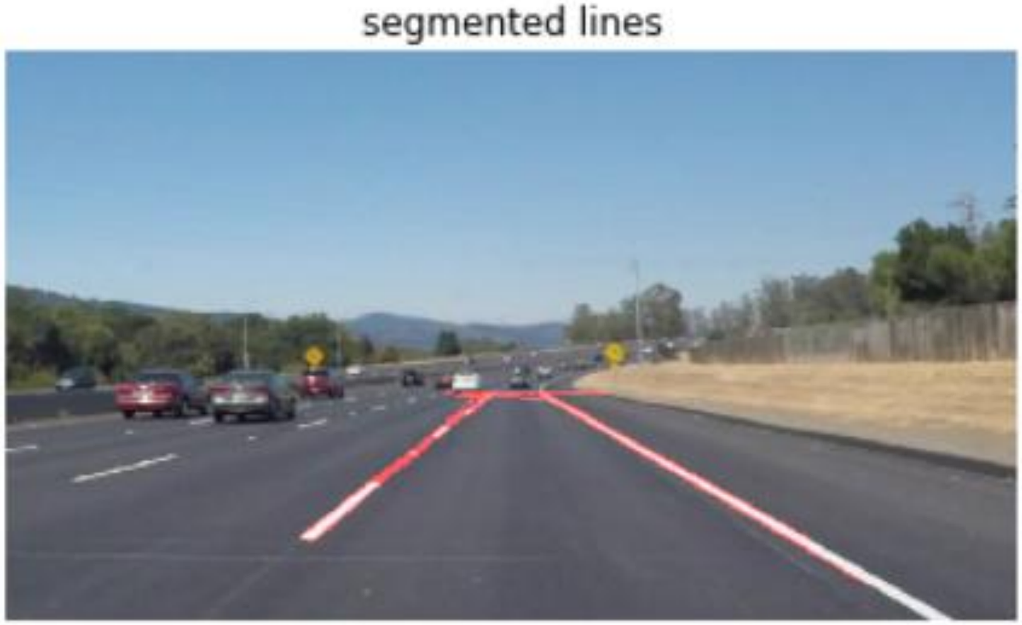
#4

Use region of interest function to filter the desired location



#5

Apply Hough transform for detecting straight lines and apply it to the original image to get the initial segmented.



#6

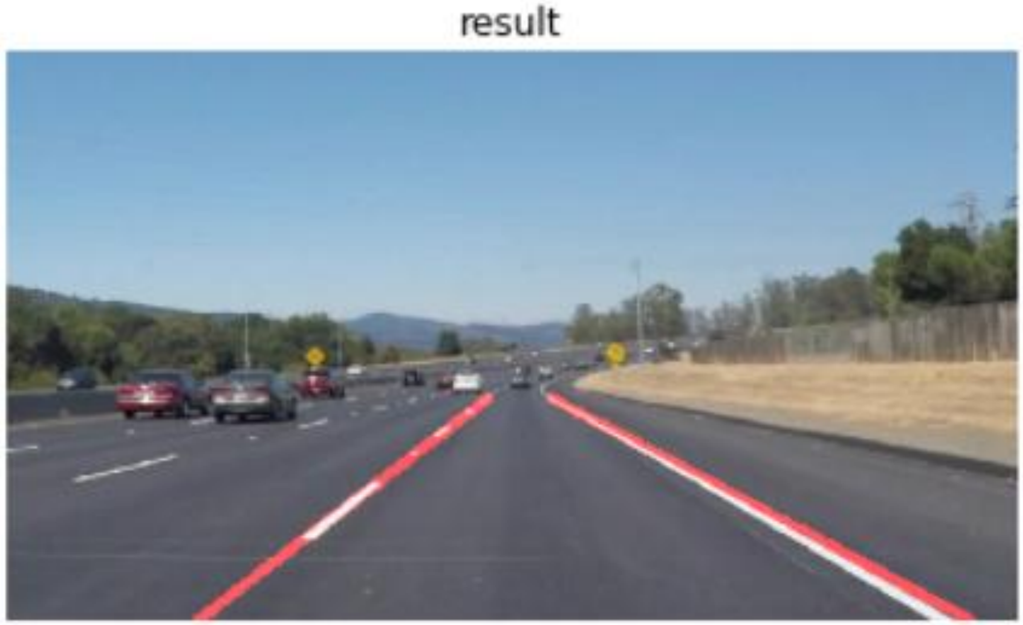
The original draw_lines function (previously detailed) was copied as draw_lane_lines in order to draw a single line on the left and right lanes.

The first step was to separate right and left segments and remove possible horizontal or vertical lines. To do so use the associated slope ((y2-y1)/(x2-x1)) and accumulate lines to the right and left in arrays. This process is done in the get_main_lines function.

Then, using the median I get the middle point of all slopes and then generate a new line estimation for the right and left lanes (get_points function).

Finally use the draw_lane_lines function to draw the line on the original image.

The final result is:



Potential shortcomings with your current pipeline

The pipeline applied to the images worked fine. The original videos solidWhiteRight.mp4 and solidYellowLeft.mp4 showed some problems from frame to frame processing.

In some cases the line was not on top of the detected segments and in some frames the lines were not there, they were not detected.

The pipeline applied to challenge.mp4 video is very bad.

Suggest possible improvements to your pipeline

The challenge video details changes in textures, colors, brightness and orientation which make the pipeline almost useless.

To solve the issue that some frames are not detected, it would be nice to use an average of the last few image frames to calculate the lines, all fluctuations would be improved.

To solve issues related to color and light changes the pipeline could be modified to be adaptative in that regard, changing parameters based on the image brightness.

In some cases the orientation changed a little so the region of interest should change accordingly.

When the lines started to curve, the pipeline didn’t work either, it only handled straight lines. A possible improvement is to handle rectified images and add logic to handle curved lanes.