Finding Lane Lines on the Road

The goals was:

- find lane lanes on image
- find lane lanes on video stream

Finding lane lanes on image

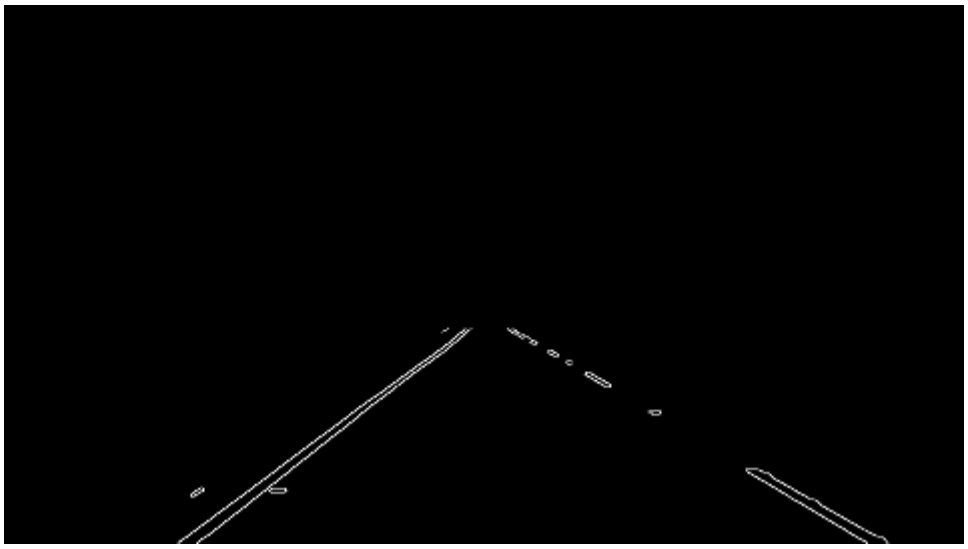
My pipeline consisted of 5 steps:

1. Convert RGB image to grayscale
2. Reduce noise by using Gaussian blur with 5x5 kernel
3. Detect edges with Canny algorithm
4. Define Region of Interest
5. Apply Hough transform within Region of Interest in order to detect lane lines

Converted image to grayscale

Image after Gaussian blur

Detected edges



Defined Region of Interest

Detected lines without modified draw_lines function

Detected lines with modified draw_lines function

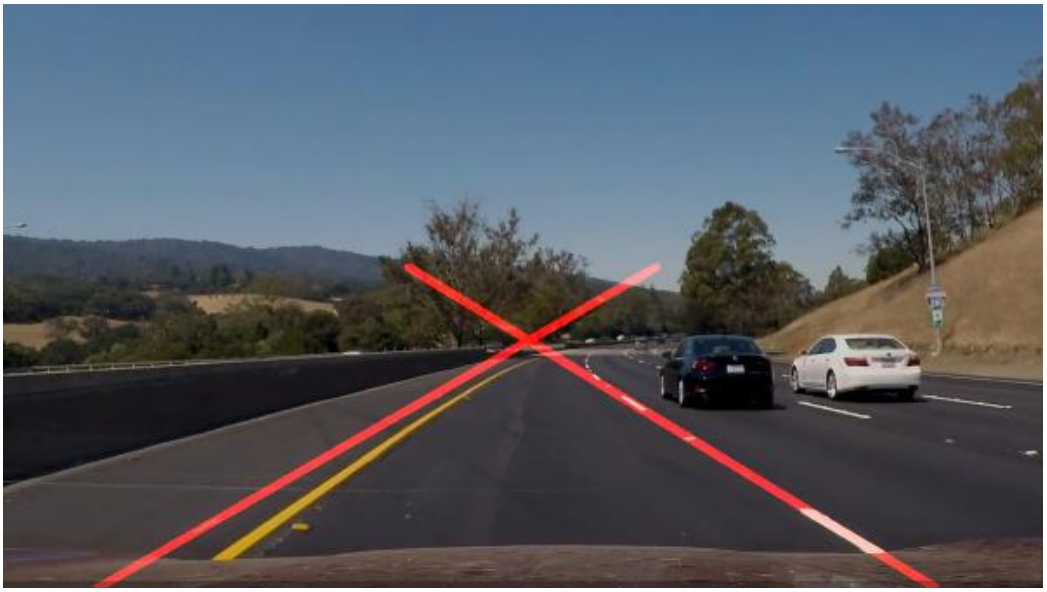
Modifications:

To get final output with single line on the left and right lanes I modified draw_lines() function by: detecting lines with the same slope in order to decide which one is left or right lane and then find the top and bottom intersect with the region of interest. With x,y values I could draw single line.

Potential shortcomings

My algorithm fails if there will by snow on road ,that is the lane lines will be invisible or very difficult to see.

Another problem is that algorithm detect only straight lines. It is very well seen during lane lines detecting in “challenge.mp4” clip. Most of the time the vehicle goes by turn so instead of straight lines there are curves.



Failed detecting curved lane lines

Possible improvements

First of all I would create a GUI utility to test all of the algorithms features (canny, Hough transform, Gaussian blur) and get a result in real time, so that I can decide on the optimal overall settings.

Another idea is to implement algorithms to detect boundaries between lane lines both straight and curved. Implement perspective transformation and use same kind of histogram analyzation.