Project 1: Finding Lane Lines on the Road

Project Summary

This project aims to detect lane lines on the road. A pipeline which finds the lines from the image or video will be presented below.

Reflection

Pipeline explanation

The pipeline consists of 6 steps which are:

1. Gray Scale
2. Gaussian Smoothing
3. Canny Transform
4. Masking
5. Hough Transform
6. Only one line for each lane and extrapolation

1. **Gray Scale**

First of all, in order to use Canny edge detection process, image should be transform from RGB to Gray Scale. Grayscale has left us with an 8-bit image, so each pixel can take between 0 and 255 possible values. By this we can look value changes in order to find edges easily. The original image and gray scaled image can be seen below.

Figure 1 - Original image and gray scale image

2. Gaussian Smoothing

After gray scale transform, in order to get smooth edges via filter the noises before applying Canny edge detection, we can apply gaussian smoothing (Blurring). Gaussian Smoothing image can be seen in below.

Figure 2 - Gaussian Smoothing

3. Canny Edge Detection

After getting blurred gray scale image, Canny edge detection algorithm is applied. This algorithm is using pixel values changes and extracting to edges regarding thresholds.

Figure 3 - Canny Edge Detection

As it seen from Figure 3, processed image has too many edges due to different features of the picture. However now we have road lines also.

4. Region of interest

By Canny edge detection we have road lines and other edges. Due to our aim is finding lane lines, we can focus on lane line area and get rid of from other features. In order to do that image is masked with four sided polygon. Polygon corner points are selected as road lane lines area. Masked image can be seen in Figure 4.

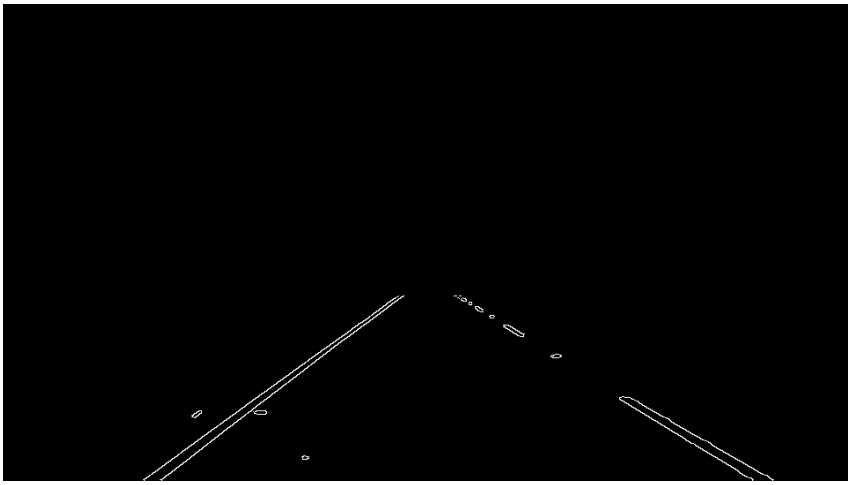

Figure 4 - Masked image

5. Hough transform

Now we have lane lines and some other objects. In order to find lines inside the image, Hough transform method is used. These method is searching algorithm which finds possible lines in figure and gives line points as outputs. By calibration of Hough transform parameters, unnecessary lines are eliminated like in Figure 5.

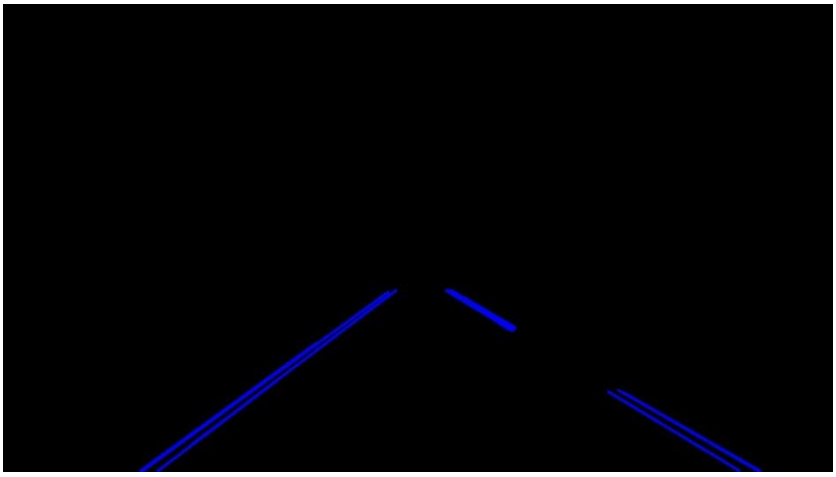

Figure 5 - Hough Transform

However, these parameters are not enough to eliminate lines which are not road lane lines.

6. **Draw Lines for lane lines**

After Hough transform we have lines points which are in x1,y1,x2,y2 format. In this step we want to get only one line for each road lane line as left lane line and right lane line.

Draw Lines function has three step

1. Filter lines regarding slope thresholds
2. Find average line equation
3. Draw extrapolated average line

Filter lines regarding slope thresholds

First of all, all slopes are calculated and slope values are examined. Left road lane lines were around -0.70 thus left road lane lines are separated regarding threshold which are between -0.5 and -3. These slope threshold values are applied right lane lines as between 0.5 and 3.

Find average line equation

Then average points of lines are calculated by using of filtered values for each lane lines. This average calculation is done in order to calculate average line formula which is

$$y = mx + b$$

From these averaged points, average slope is calculated as

$$m_{avarage} = \frac{y_{2_{avarage}} - y_{1_{avarage}}}{x_{2_{avarage}} - x_{1_{avarage}}}$$

Draw extrapolated average line

At the last step, we have all information about finding of extrapolated line points. In order to draw line, we need to get start and end points of extrapolated line. If we select as bottom of the image as initial ($y_{initial} = 540$). And also if we select as end of the line y point as end of the region of interest ($y_{end} = 330$). X points can be calculated as

$$x_{end} = \frac{y_{end} - y_{2_{avarage}}}{m_{avarage}} + x_{2_{avarage}}$$

$$x_{initial} = \frac{y_{end} - y_{1_{avarage}}}{m_{avarage}} + x_{1_{avarage}}$$

After calculation of line points for each road lane line, result can be seen below.

Figure 6 - Draw Lines function output with the original image

2. Potential Shortcomings

If there are some lines different from road lane lines found from hough transform and also if they are in slope thresholds, extrapolated lines will not be road lane lines.

If Canny edge detection or hough transform can not find road lane line in a frame, this frame will not have any lane line equation, it will be empty.

If there is curvy lane lines on road, hough transform will find too many lines with different slopes. Current pipeline is not enough for curve road

Road lane line color difference is highly important for this pipeline. Grayscale is not enough to find edges when lane line color is similar to road pavement etc.

3. Possible Improvement Points

If there is not any detection of road lane line, prediction of line can be used regarding before line values in order to avoid empty detection.

Instead of hough transform, curve fitting methods should be use

Instead of grayscale, there should be advanced color filtering method in order to distinct road lane lines from road pavement when their color values are so close to each other.