

Project 1

Finding Lane Lines on the Road

Goals

- Make a pipeline that finds lane lines on the road
- Reflect on your work in a written report

Reflections

My pipeline consisted of 9 steps:

1. Convert the input image into **grayscale**
2. **Gaussian smoothing** to reduce noise in the image
3. **Canny edge detection** to find out the highly contrasting edges in the image
4. **Create masked images** by defining a left region of interest for detecting the left lane and right region of interest for the right lane
5. **Hough transform** to find out an array of endpoints of detected lines segment
6. **Linear regression** to the left and right endpoints of detected lines segment
7. **Draw extrapolated linear regression lines** on a blank image which are representing the left and right lanes
8. **Mask** the previous image to include the lane lines only in the region of interest
9. **Draw** the image with the lanes

Potential shortcoming and possible improvements:

One shortcoming is when the road is not straight; it will not be able to reflect the true shape of the lanes, especially on strongly curved lanes. Because this method is assuming that the lanes are straight only. And it will draw the lane according to the best fit line using the regression model that has been trained on the end points of the detected lines. A better approach could be to use a high order `numpy.polyfit` function that returns the least square polynomial fit to the points. That would be a better approach to represent real life lanes, which are not only straight.

Another shortcoming is when the lighting condition is changed. Especially at night there will be difficulty in detecting the lanes. This can be improved using color space filtering before doing the Canny edge detection.

Also, if there are nearby cars in front of our car, there will be definitely difficulties in detecting the lane. My method is assuming that the canny detection on the region of interest will yield only the edges of lanes. If there are other objects in this region of interest, then the detected edges of these objects will be included too in the calculation of the regression line that represents the lanes of the road. Color space filtering can reduce this error but cannot eliminate it totally. Maybe using deep learning to detect objects and vehicles and eliminate them from the image can solve the issue.