

Finding Lane Lines on the Road

Udacity SDCND

6/1/2017

Daniel Tobias

Pipeline

The image pipeline to detect lane markers can be described as the following sequence. Convert to grayscale, gaussian blur, canny edge, extract region of interest, hough line transform then transpose hough lines to the original image.

The gaussian blur was used to smooth out the images for the canny edge detection so that actual lines would show up instead of noise in the image. Since the lines were showing up in the images and video at roughly the same position, a region of interest mask was used to focus in on where the lanes would show up. This ROI would have to change if the camera position or angle changed. I modified the line points inside the hough line function to overwrite the y value of the lowest point of the line to be the on the lowest pixel row, while modifying the paired x value to adjust to the lines slope relative to the other point's y value.

Shortcomings

Using a static region of interest mask to focus in on where the lanes would be would not scale or handle different environments well. If the camera position or angle changed slightly the entire ROI mask would need to be adjusted. In the solidYellowLeft video there are some frames where the pipeline picks up horizontal lines that cross the middle of the lane the car is driving in. Different lighting and weather might also not work well with this pipeline. Glare from other headlines in a wet road environment might get detected as lanes and cause the car to recognize them as lanes which would be fatal. Averaging the points to draw a single averaged line may have adverse effects, if the points are not filtered out.

Improvements

To handle stray horizontal lines you could add additional code that checked for lanes being drawn that exceed a certain slope value, IE horizontal lines/lines. Having a flexible ROI that took into consideration where previous lanes were found might help the pipeline handle larger turn radiuses. Using color augmentations to the image to help the canny edge detection find the lane markings might the accuracy of the pipeline. Using the distance between the lanes could be used to double check if future lines detected are indeed lane lines