Self-Driving Car Project #1

Finding Lane Lines on the Road

Complete on Dec 23rd, 2017

Haowei Zhang

Part 1

Task:
Describe your pipeline. As part of the description, explain how you modified the *draw_lines*() function.

Response:
My pipeline is made up of a series operations on raw 2D 3-channel images to extract lane lines on the road. Following are the procedures. Note that the image to deal with are all shapes of (540, 960).

(a) Convert the raw 3-channel image to grayscale image.

(b) Blur the grayscaled image with a Gaussian blur to eliminate high-frequency noise. The kernel size I picked was 5.

(c) Apply Canny Edge Detector to extract edge features out of the former processed image and store it as an edge feature image. I define a low threshold of 50 and a high threshold of 150.

(d) Because the camera on the car doesn't move, I generate a static polygon shape mask that the lane lines lie within. The coordinates that I wrap the polygon region are (100, 540), (480, 300), (500, 300), (940, 540). I would only be dealing with edges that fall within the mask to extract lane lines.

(e) Next step I apply Hough Transform to extract coordinates of line segments within the masked region of the image.

(f) The line segments extracted in previous step not only includes true line segments that form the lane, but also includes false line segments that constitute any structured elements in the masked region, e.g car. To get rid of those false positives, for every line segment I check if its slope does match the one of the lane, i.e 1.0 ± 0.5 for left lane and -1.0 ± 0.5 for right lane. Doing this I successfully eliminate all the false positives and left with real line segments that form the lane.

(g) Then I use cv.fitLine() function to fit a linear model of left and right lanes given coordinates of corresponding line segments. This works well and in the last step I use this linear model to find the corresponding endpoints of those lanes.

Part 2

Task:
Identify potential shortcomings with your current pipeline.

Response:
Following are some shortcomings I thought of:

(a) This lane detection algorithm only works on straight lanes on road. If the road has a turning angle, this lane detection would not be able to detect properly.

(b) If the camera turns some angles, this lane detection algorithm won't be able to capture all the line segments of the lane because the masked region is now defined hard-code.

(c) The way it detects endpoints of the lanes is ambiguous. It would miss the endpoints if edge detector accidentally captures similar elements that are falsely detected as part of line segments that form the lane.

Part 3

Task:
Suggest possible improvements to your pipeline.

Response:
Following are some possible improvements I thought of:

(a) Would be ideal to detect circular lanes when the car slightly turns left or right.

(b) Would be ideal to have a better algorithm to detect where the lane ends.