Finding Lane Lines on the Road

Finding Lane Lines on the Road

The goals / steps of this project are the following:

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

My Pipeline

The algorithm is very straightforward and similar to the once from the last quiz. First, turn the image into grayscale, since color is not an useful indicator for the lanes. We have both yellow and white lanes. Then use the canny filter to find the edges. After that, apply a mask on the image, so that we only care about edges in a given region. Otherwise, there will be too mush noise. To make the model more robust and works better for the challenge, I set the mask to be proportional to the size of the image. In the challenge the resolution of the video is different.
Hard-coded mask will not make sense. Like the image below.

In this case, the hard-coded mask is too big, which includes the trees on the sides.
Then, hough algorithm is applied. By converting the image to the hough space, it's easier to find lines in the original space. To draw only one single line, the slopes of each small piece of line are found and the line is extended to the baseline and center of the image. Eventually, the average of the start and end point is drawn.

Shortcomings

In the yellow lane video, the lane on the left is very disconnected, so that in some frames, there will be no line drawn at all. It's due to the fact that no edge is found. Also, my pipeline doesn't work perfectly in the challenge video. The road is turning and there is a curve, which makes the lane even more disconnected. So that in some frames, the program will generate wrong outcomes. Like the one I showed above.

Possible future improvements

Try to make the decision of the current frame dependent on the knowledge before this frame, because most of the time the lane wouldn't just disappear or flying around.