Finding Lane Lines on the Road

Finding Lane Lines on the Road

The goals / steps of this project are the following:

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

Reflection

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

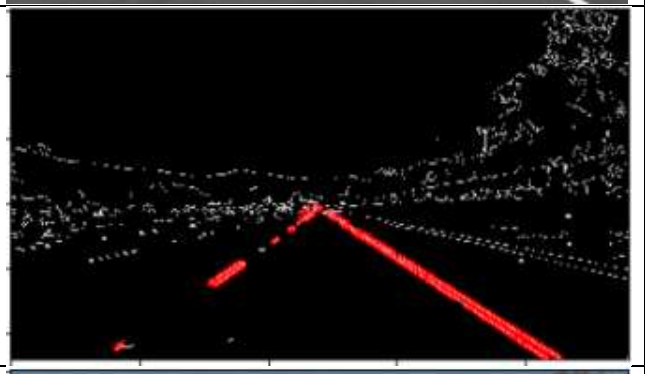
My pipeline consisted of 6 steps:

1) I converted the images to grayscale
2) Applied a Canny filtering with threasholds of 40 and 150
3) Masked the Image with a 4-sided polygon to focus only on a specific part of the video
4) Performed a Hough transform
5) Iterate and Draw Output Lines on the Image
6) Extrapolated segmented values to create full lines.

In order to draw a single line on the left and right lanes, I modified the draw_lines() function in the following way:

- Each line gradient is calculated
- Lines are sorted by gradient to left or right side.
- The min and max coordinates are calculated
- Coordinates are then modified to extrapolate the lines to bottom and middle parts of the video.
- Each of the right and left lines are drawn on the result image with the cv2.line() function

Images of how the pipeline works:

Pipeline Description	Pipeline Images
Grayscale transformation	
Canny filtering + Hough transform lines	
Hough transform lines + image	
Result image - Extrapolated lines	

2. Identify potential shortcomings with your current pipeline

Potential shortcomings are:

1) Modifying vertices coordinates can cause error. Ideally this shouldn't happen.

3. Suggest possible improvements to your pipeline

Possible improvements are:

1) Solve vertices coordinates issue
2) Use more of the helper functions to make pipeline shorter
3) Create a function to call each time extrapolating is performed for left/right top/bottom coordinates. This would shorten the pipeline and make it less complex.
4) Although code works it can be cleaned up to make more efficient