

Finding Lane Lines on the Road

Mochamad Vicky Ghani Aziz (gani@misagani.com)



Summary

In this project, I will be writing code to identify lane lines on the road, first in an image, and later in a video stream. And created Reflection that describes the current pipeline, identifies its potential shortcomings and suggests possible improvements.

Instructions

My Goal is to write code including a series of steps (pipeline) that identify and draw the lane lines on a few test images. Once I can successfully identify the lines in an image, I will use that code into the block provided to run on a video stream.

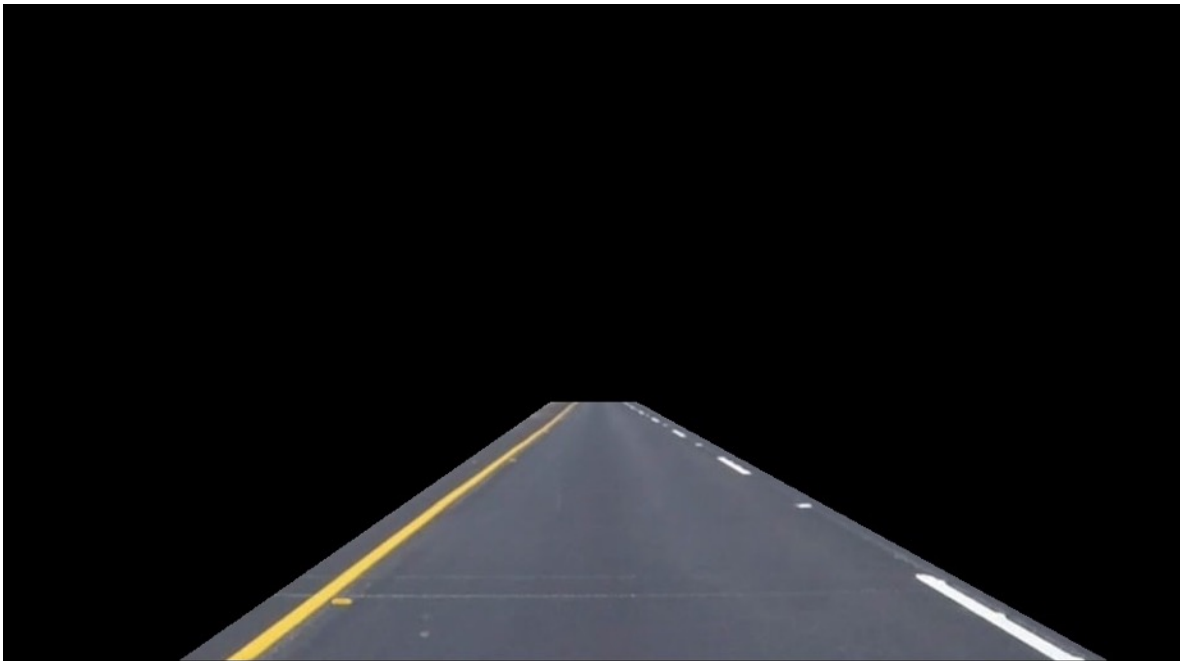
Pipeline

My pipeline consists of these steps.

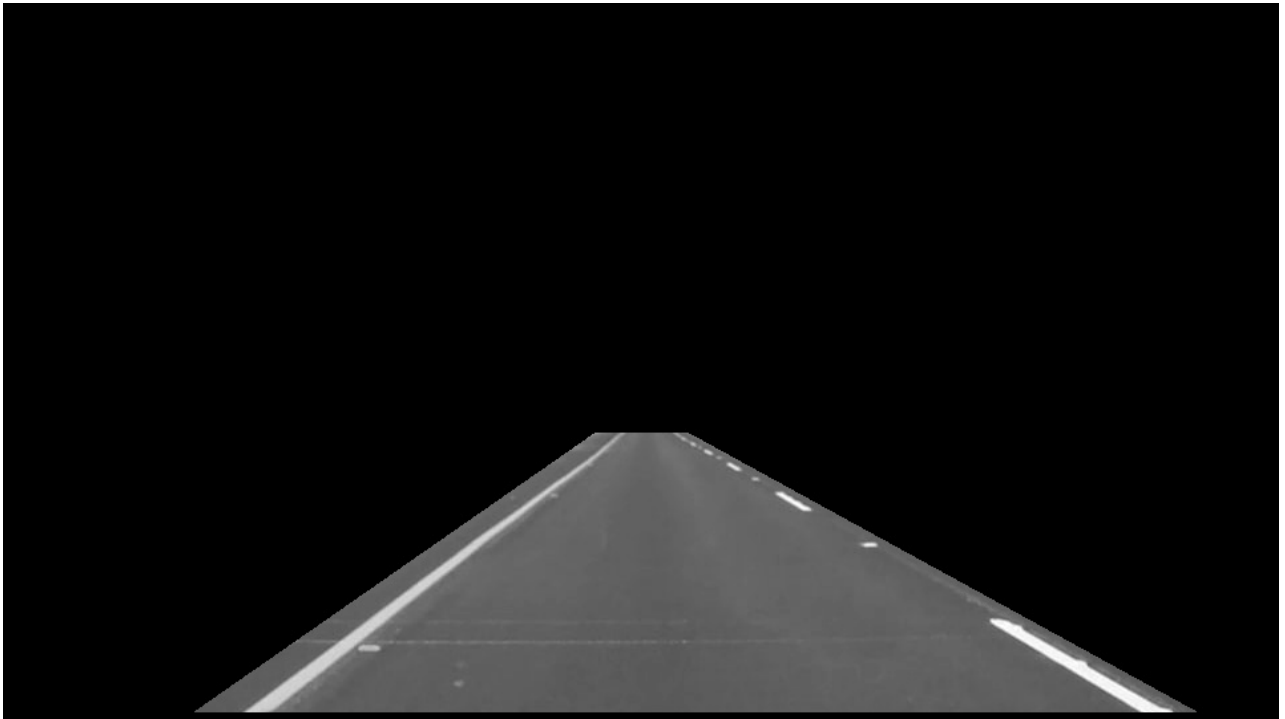
- **Getting Image Source.** This will return original image.



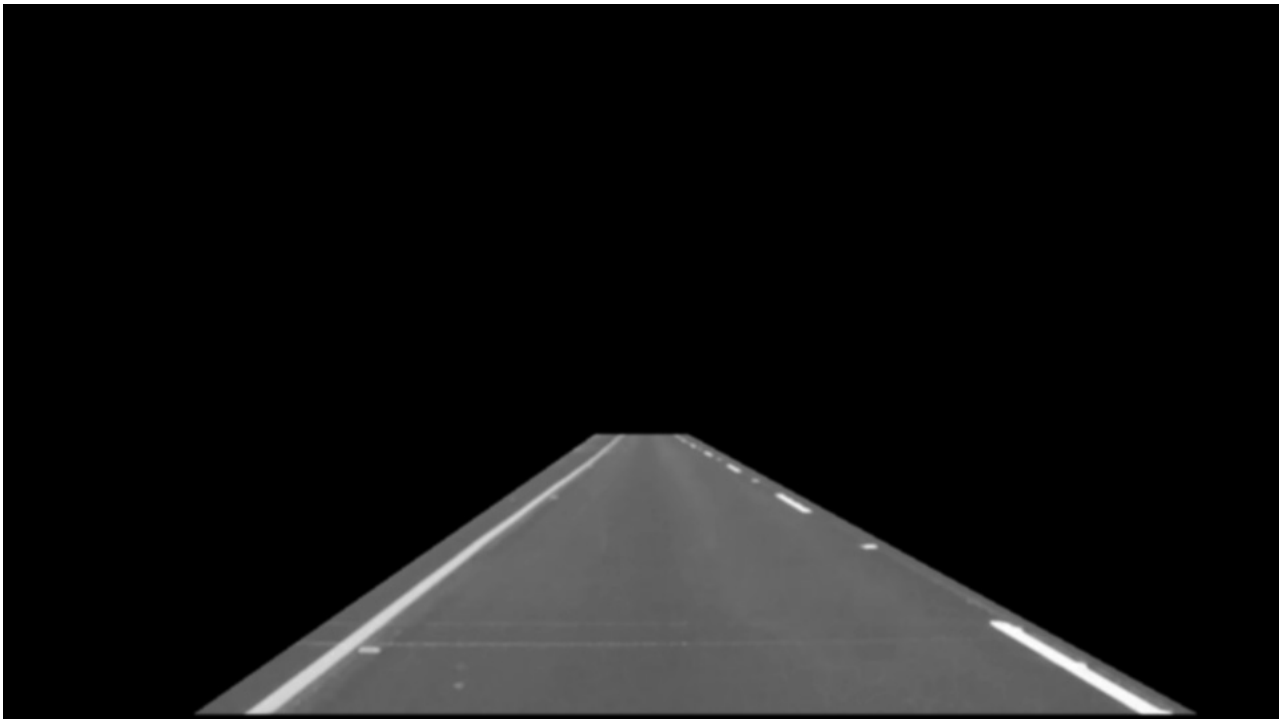
- **Region Masking.** Applies an Image Masking. Only keeps the region of the image defined by the polygon formed from vertices. The rest of the image is set to black.



- **Grayscale.** This will return an image with only one-color channel.



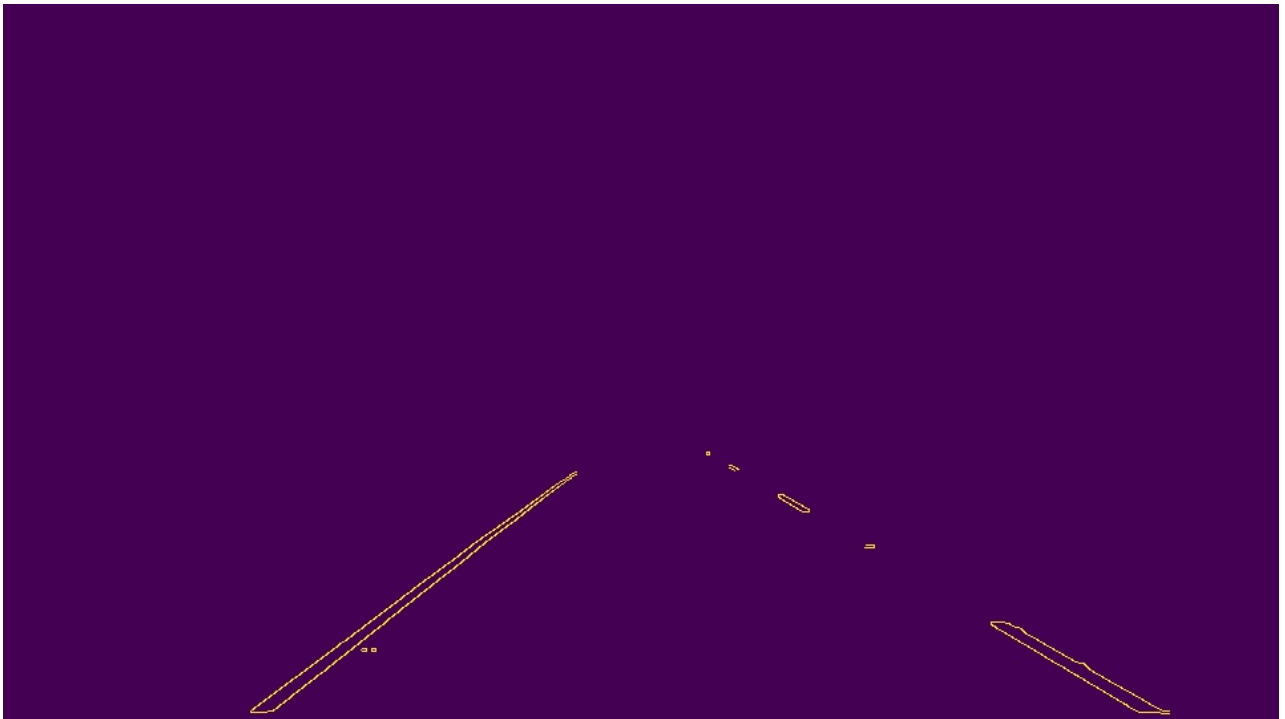
- **Gaussian Blur.** Adding the Gaussian Noise Kernel.



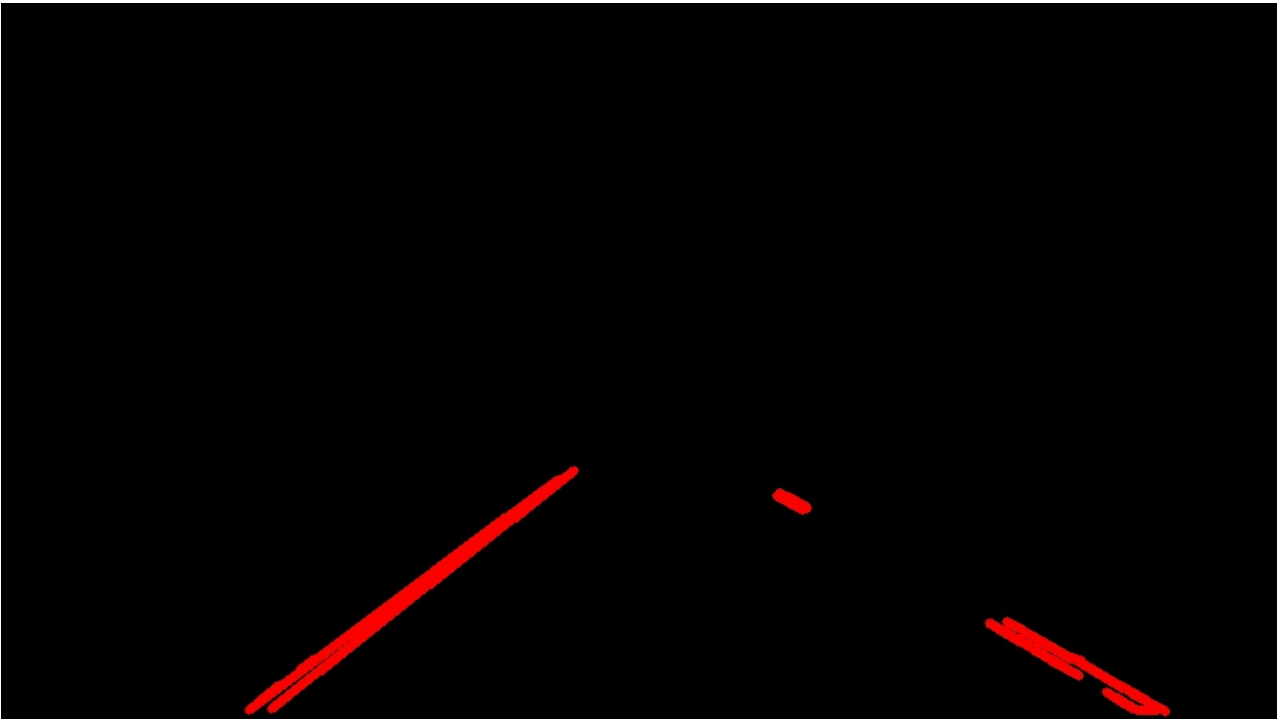
- **Color Selection.** Applies the Color Selection.



- **Canny Edge.** Applies the Canny Edges Transform.

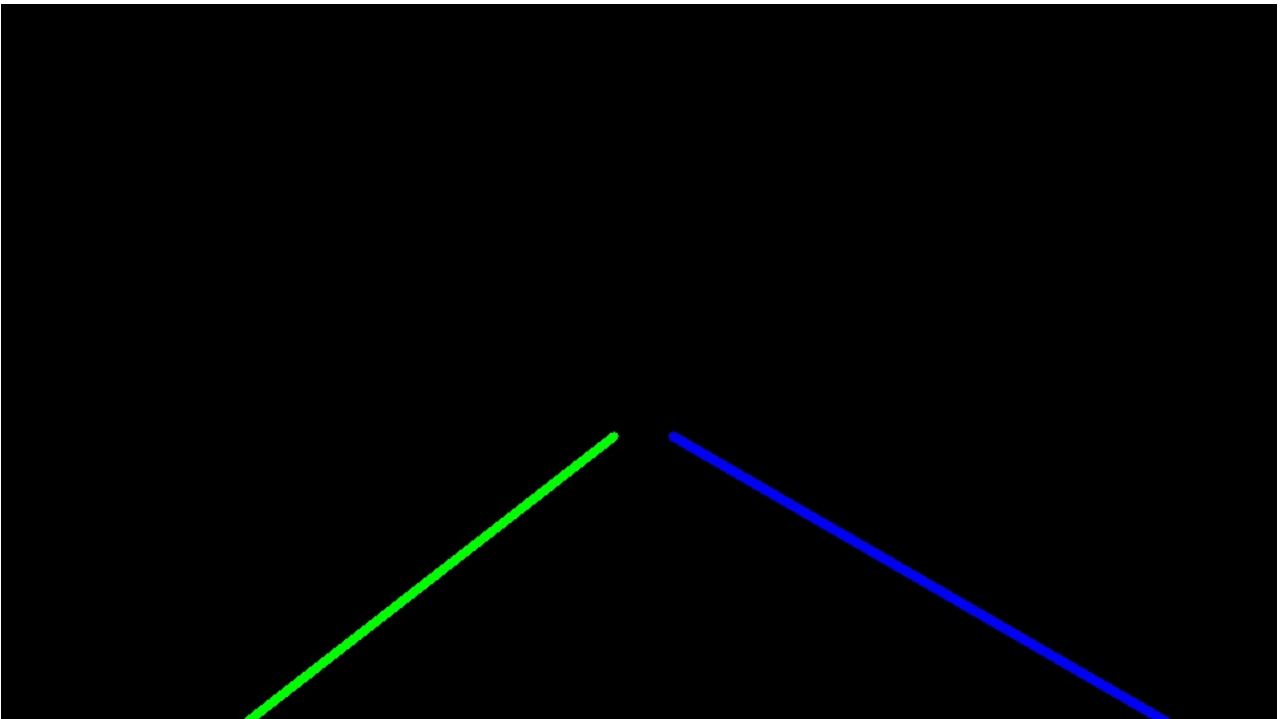


- **Hough Transform.** Applies the Hough Transform.



After that, I draw solid lines combine with Real Images.

- **Draw Lines.** Drawing Solid Lines based on slope from Hough Transform result.



- **Weighted Image.** Combine Lines and Real Image.



Potential Shortcomings

There are several potential shortcomings.

- Potential shortcoming would happen when lanes meet sharp-turn. Need to be fixing the slope algorithm.
- Another shortcoming would happen if the line not solid, and the color not consistent.
- Another shortcoming would happen if the line not visible, maybe there are object covering it.
- Another shortcoming would happen if the line covered with shadow on one side. For example, if left line not covered, and right line covered, then only left line that detected by this algorithm.
- And one more shortcoming would happen if the resolution of Images/Videos different (not standardized). Because the lines that created, using fixed coordinates.

Possible Improvements

There are several possible improvements.

- Possible improvement would be to creating a historical data of slope. So we can predict next slope depend on before data. And then we can using statistics for searching the slope, not simple just like this, only counting the average value.
- Another possible improvement would be to create dynamic algorithm for drawing a lines. Possibly by using shape of the image, and then proportionally using that for drawing a lines.
- Or maybe using more Advanced Image Processing techniques.

Challenge

I try the Challenge video and adding one my own video. Very Excited!