

Project 1 - Lane Detection Project Using OpenCV

Project Requirement:

The input is a folder containing images. The program displays each image from the folder, and marks the detected lane line with a distinct color. Broken or 'dotted' white lines (the center line between lanes) should be shown as one long lane line. Lane lines can be continuous or dotted lines. They can be white, yellow or red (fire lane). The lane can be straight or curve. The program should also compute and print the total number of detected lane lines. Please note that we are counting the number of lane lines and not the number of lanes. For example, double white lines will be counted as two lane lines. Dotted lines which form a single lane line will be counted as one. The same lane line should not be counted twice. The images could be of any size. There could be images without lane line.

Steps followed in Lane Detection:

- Convert the image from BGR to RGB
- Resize all the images to one standard size
- Convert original image to grayscale.
- Blur the grayscale image using gaussian blur
- Convert original image to HLS color space.
- Isolate the gray color, get the contour, blur it, and dilate it to get the gray mask
- Isolate yellow from HLS to get yellow mask.
- Isolate white from HLS to get white mask
- Bit-wise OR yellow and white masks to get common mask.
- Bit-wise AND mask with darkened image .
- Apply slight Gaussian Blur.
- Apply canny Edge Detector (adjust the thresholds — trial and error) to get edges.
- Reduce the noise from the canny image
- Get the contour again for a maximum area
- Again apply canny to get the masked canny image
- Then dilate the mask
- Define Region of Interest and crop the dilated mask according to the region
- Apply Hough Transformation to get the Hough lines
- Merge multiple similar and small Hough Lines into one same line
- Consolidate and extrapolate the Hough lines and draw them on original image

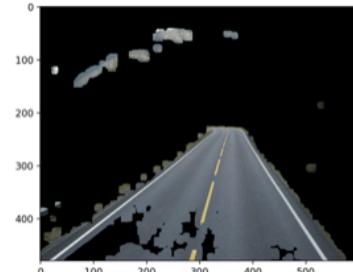
Step-by-step result of one particular test image:



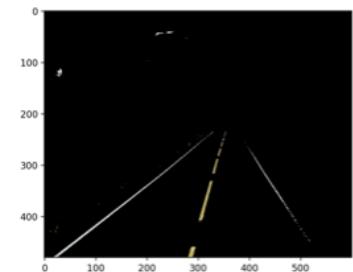
Following are the different transformations on it to detect lanes:



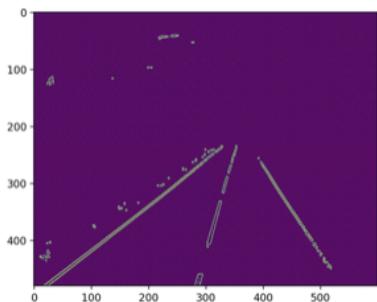
a. Resized image to a standard size



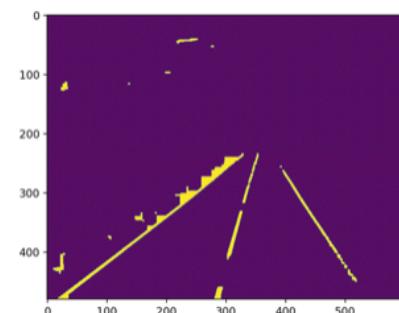
b. Gray filter applied to extract gray regions



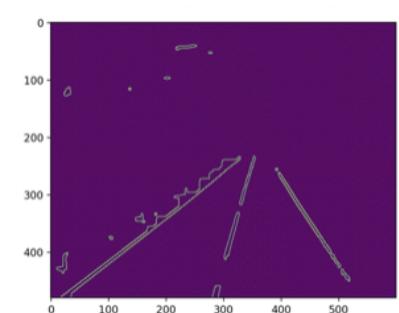
c. Filtered Yellow and White colors from the gray Image



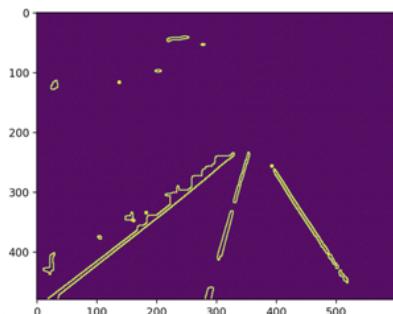
d. Applied Canny edge Detection to the last image



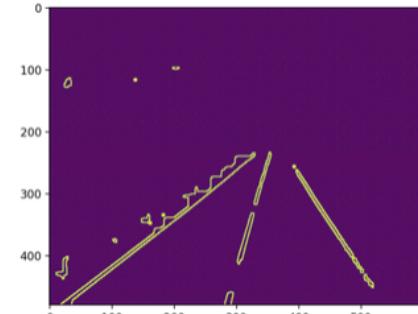
e. Retrieved the contours of the last image



f. Again applied Canny edge Detection to the last image



g. Dilated the canny image



h. Applied region of interest to the last image to focus more on road



i. Applied Hough Transformation to get lane lines

Results and Observations of the given Test Images:

