Project 1: Lane Detection.

ENPM 673, Robotics Perception

Gunjan K Khut

UID – 115813083

The following steps were involved in the lane detection project:

1. **Denoising the image using Median filter** – I started the project by applying a median filter to smoothen the edges in the image. This allowed a clear line detection in the lanes which were having sharper ridges in the lanes. The output is shown in the image below:

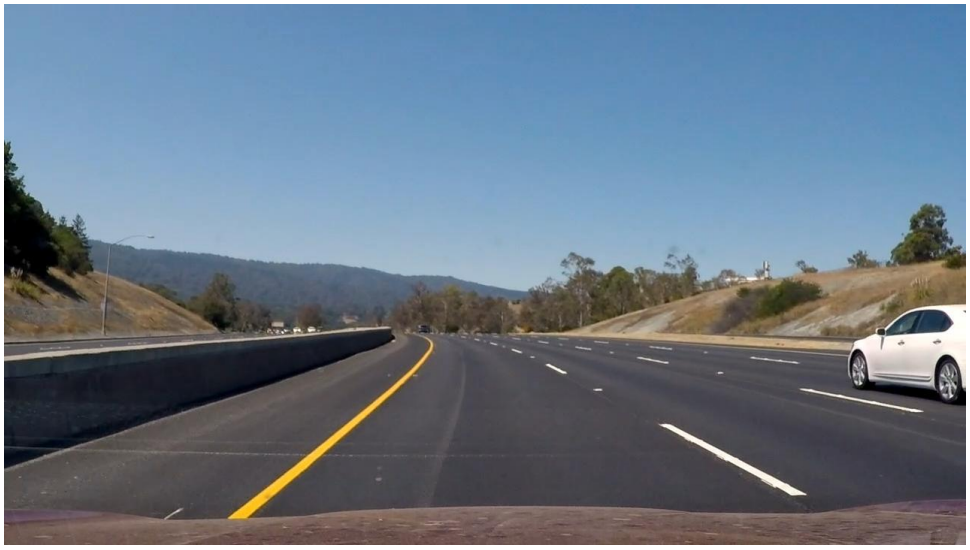

Fig (1a) – Input Image (Noisy)

Fig (1b) – Denoised Image

2. Binarizing the image - I converted the grayscale image into binary (BW) image. The threshold value for binarizing the image was selected on the values obtained from histogram and trial and error. The binarized image should clearly show the lanes of the road.

Fig (2) – Binary Image

3. Edge Detection - We used canny edge detection method which helps detect the edges present in the frames which we obtained after applying the median filter.

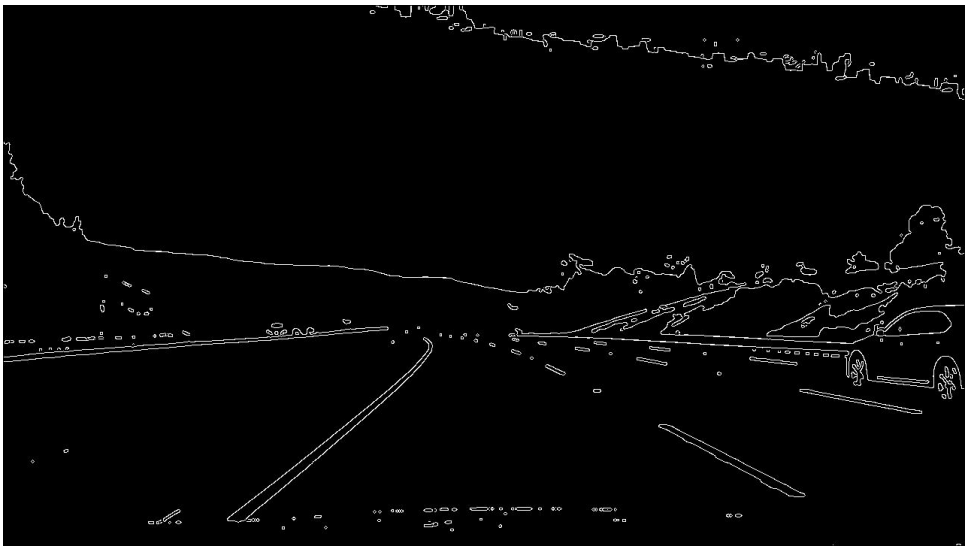

Fig (3) – Image with edge detected

4. Masking the image - The next step was to mask the right size of the image such that the top half of the image and the sides were ignored as our region of interest always lies in the bottom half of the frame. The shape of the mask is a trapezoid and the result of applying the mask can be seen below.

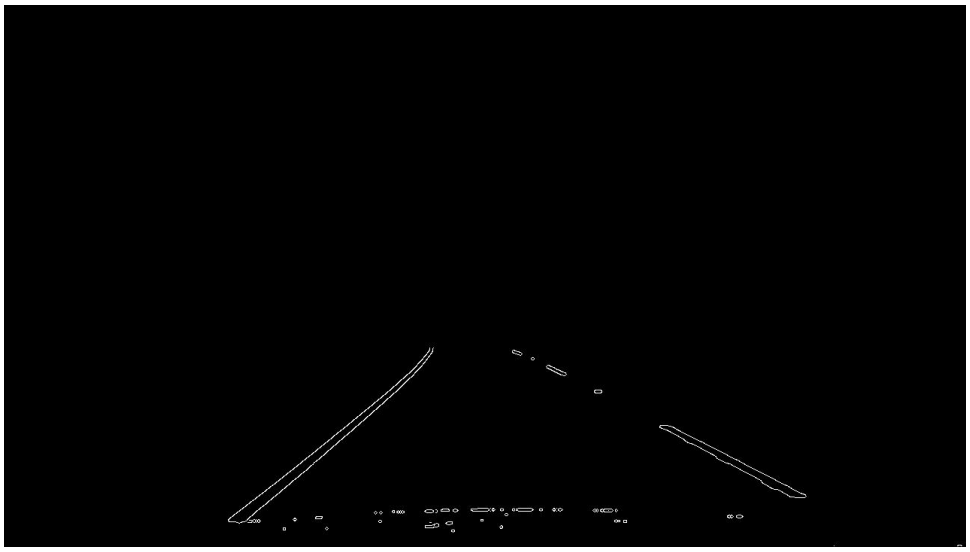

Fig (4)- Image with unwanted part masked

5. Hough Transform – Then we performed the Hough transform for the image which we obtained after masking. We first found the theta and rho values required to find the Hough peaks then used the values to find the Hough lines. There was a lot of trial and error performed to identify the best set of required parameters for the frame.

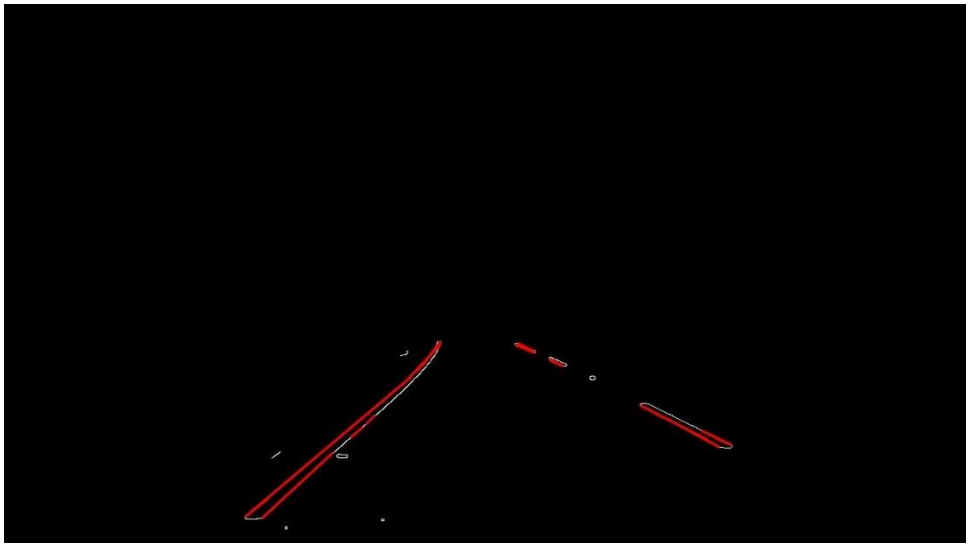

Fig (5)- Hough Transform

6. Separating the lines – I calculated the slope of all detected Hough lines, which helped me separate the lines present at the left lane, at the right lane and which were not a part of either lane.

7. Regression – The points (both starting and ending points) belonging to left and right lanes are stored. Then we performed the regression operation to fit these points as one line. Then fit polynomial for both the left and right sides of the lane. This gives the lane which is required by the car to follow the road. We have fitted with three known points as the lane curves sometimes. The known points which are used for fitting these curves are the starting points and ending points.

Fig (6)- Regression

A blue mask is then applied to the area between the left and the right lane along with a black border which is provided on both the lanes. This completes the project of lane detection.

8. Identifying the turn- To predict the turn, I used the slope of the left line. The turn is predicted by comparing the slope of the left line with certain threshold values, the threshold values are obtained experimentally. The method is collaboratively developed by me and my classmates. The names of the classmates are following.
    - Utsav Patel
    - Dipam Patel

EXTRA CREDIT

The same technique was followed for the extra credit challenge video.

- There were a few difficulties observed due to the variation in the road colors in the images. So, a few added thresholding parameters were added to obtain the yellow and white lines before filtering the image to determine the edges.

- NOTE: Out of the 480 images in the challenge video provided, the detection was performed for 260 apart from a few skipped frames where detecting the yellow lines was very difficult no matter the thresholding due to the signs in the floor.

Fig (7)- Going Straight