

# Visual Recognition Assignment 1 (Part 1)

Mohd. Rizwan Shaikh  
IMT2019513

February 9, 2022

## **Abstract**

The aim of this assignment is to implement Hough Transform from scratch and use line fitting to detect road lane in images.

## **Contents**

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Image Data</b>	<b>2</b>
<b>3</b>	<b>Method</b>	<b>2</b>
<b>4</b>	<b>Results</b>	<b>5</b>
<b>5</b>	<b>Other Experiments</b>	<b>6</b>

# 1 Introduction

In this assignment, we implement Hough Transform from scratch and use it for line fitting to detect road lane in images. For verifying the implementation, we test it on four separate images. The images used for this assignment are from IIIT Bangalore campus except one which is taken from the internet.

## 2 Image Data



Figure 1: Input Images

## 3 Method

First of all, we apply Canny Edge detector on all the images. Since OpenCV's *canny()* implementation can handle multiple channels, we input the images directly into it without converting them to gray scale. The output of Canny Edge detector is shown below:

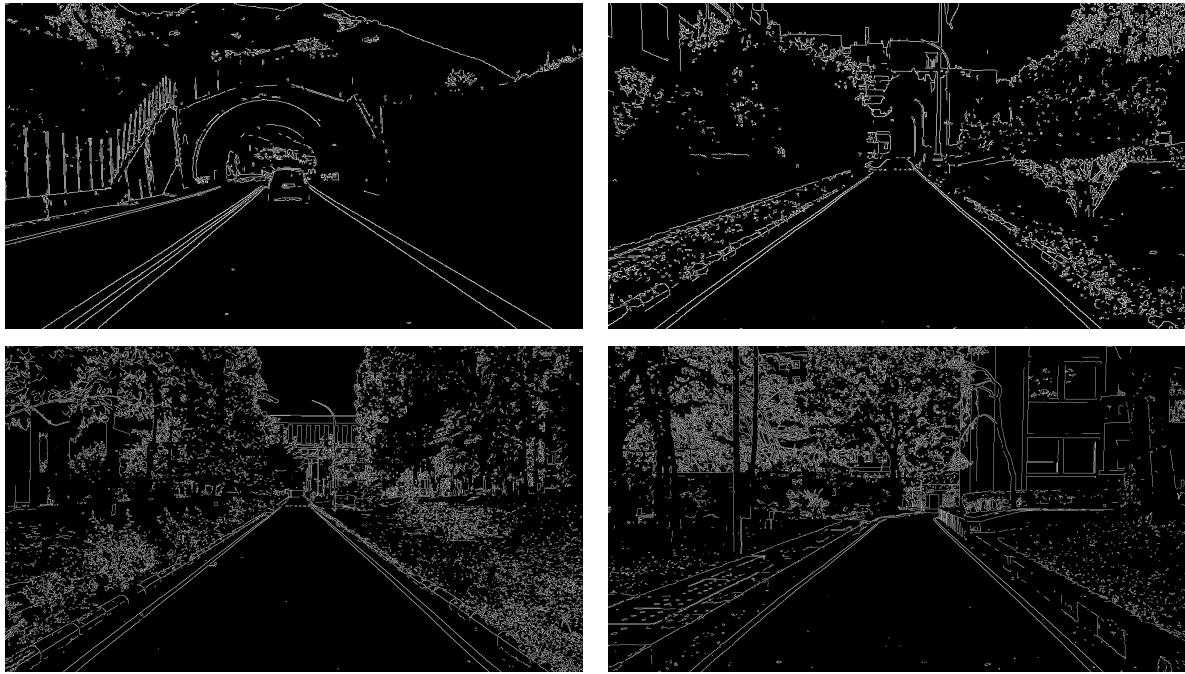


Figure 2: Edge detection in images using Canny Edge Detector

In all the above images, the upper half of the image is irrelevant for detecting the lanes. Hence, we try to clear the upper half portion by applying a triangular mask as shown below:

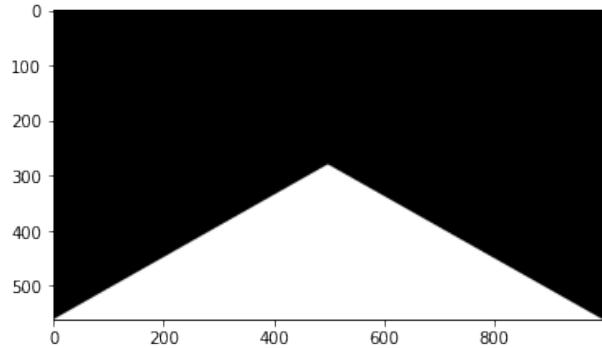


Figure 3: Mask used for clearing the excess noise (Upper half portion of images)

For masking the images, we do pixel wise AND of the mask with the images. The masked images are shown below:

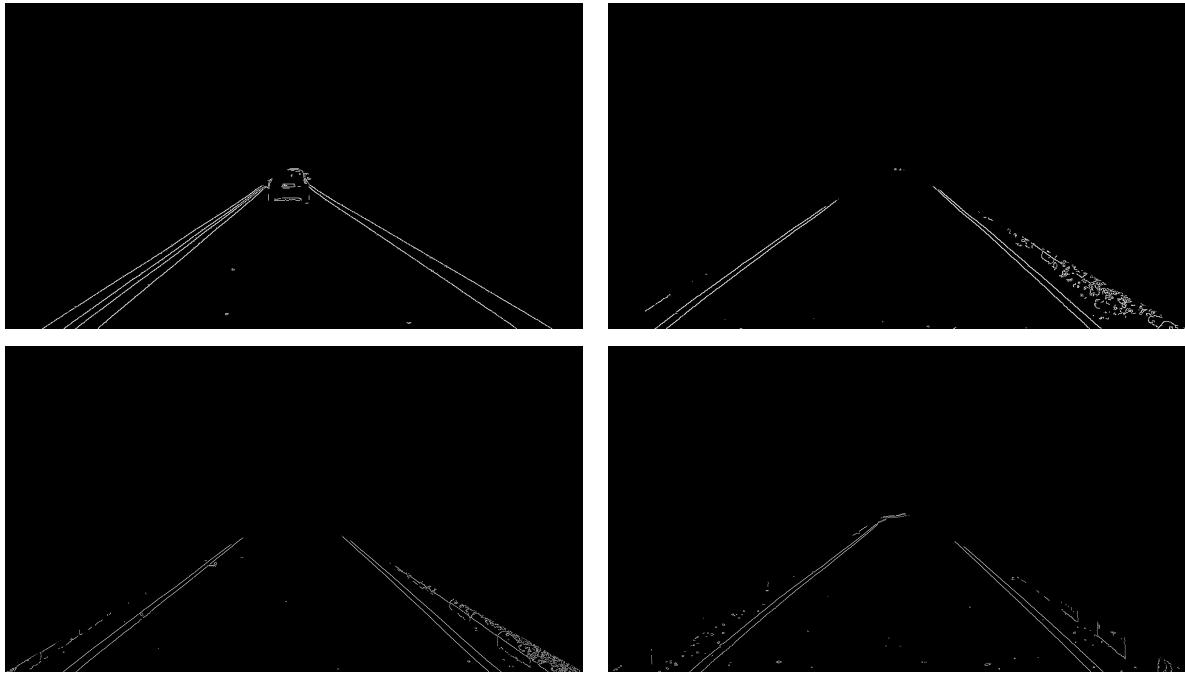


Figure 4: Masked images after removing excess noise

We now implement Hough Transform to convert the image space to parameter space. The equation of line that passes through a point is shown below in both image space as well as parameter space.

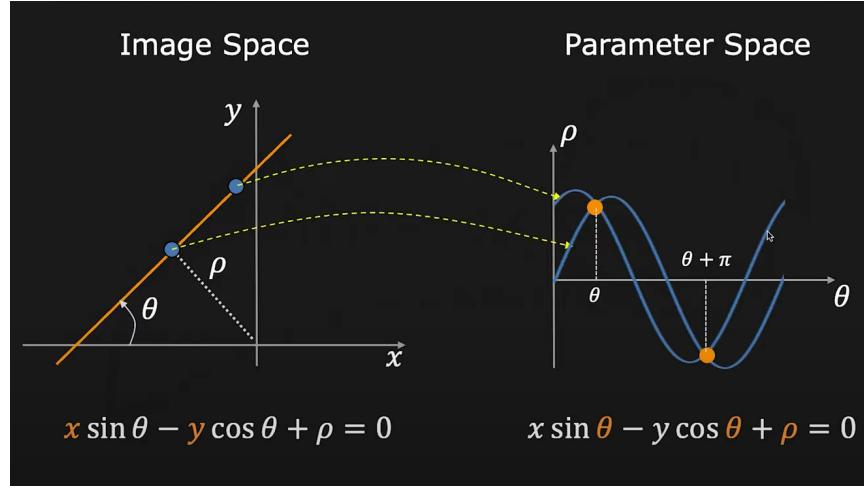


Figure 5: Equation of a line that passes through a point in image space and parameter space

We implement this transformation on the masked images to get their parameter space as shown below:

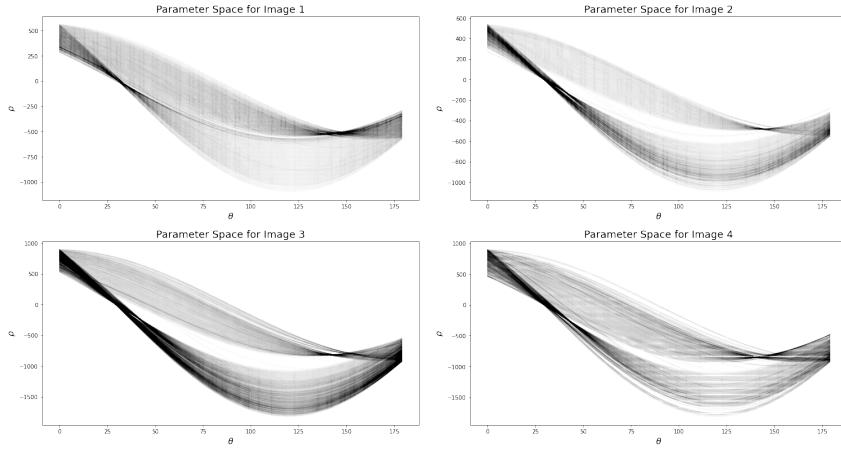


Figure 6: Parameter Space for the images

With the help of accumulator array, we get four pairs of  $\rho$  and  $\theta$ . These four pairs corresponds to the four highest counts in the accumulator array. With the help of these four pairs, we plot four lines (two on each side of the lane) to detect the lane.

## 4 Results

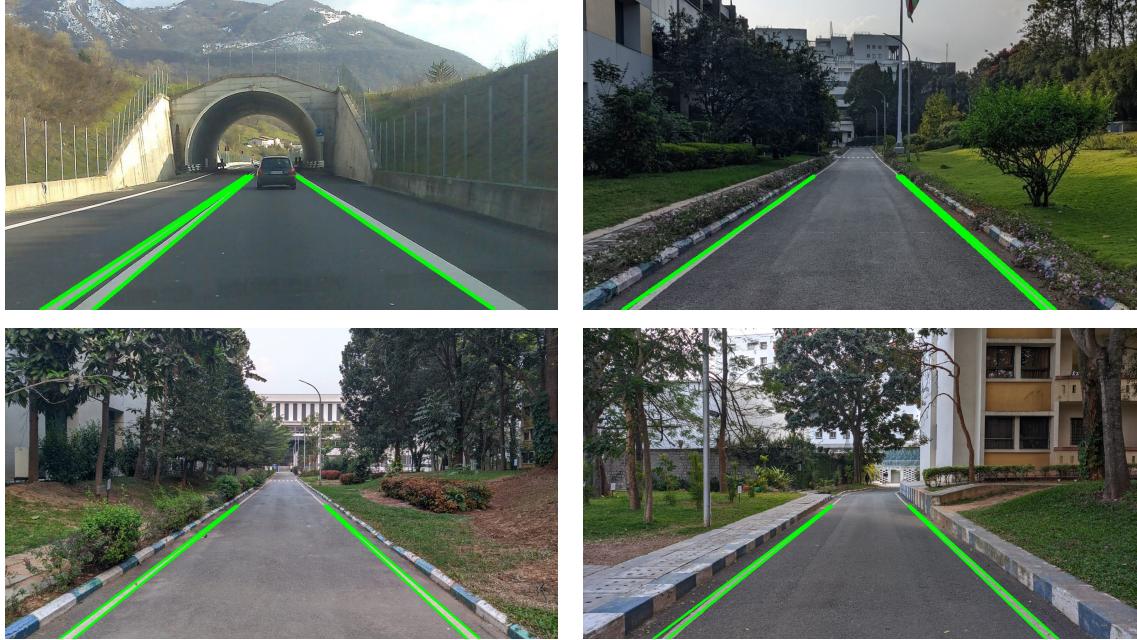


Figure 7: Detected Lanes in the images

In these images, we see that we have successfully detected the lane in front of the car using line fitting through Hough Transform. Since we use line fitting to detect the lanes,

it is absolutely necessary that the major portion of the road in front is straight for the method to work correctly.

## 5 Other Experiments

My very first attempt involved converting the image to gray scale first and then applying canny edge detection. After this, Hough transformation was implemented and line fitting was done on the images without masking them. The output obtained is given below:

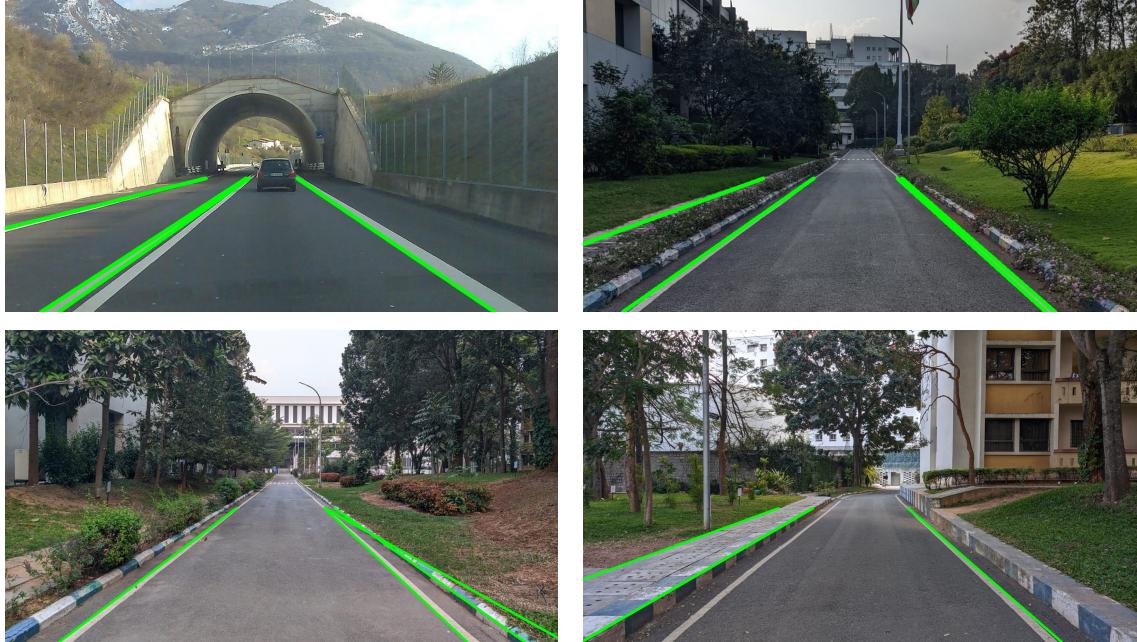


Figure 8: Lane detection in images without masking

Here we see that in all the four images, the Hough transform picked up noise and plotted lines on other portion of the road and not on the lane in front of the car.

Similar output was obtained when the mask used for removing noise covered exactly the upper half portion of the image.