

Edge-Linking-using-Hough-Transform

Aim:

To write a Python program to detect the lines using Hough Transform.

Software Required:

Anaconda - Python 3.7

Algorithm:

Step1:

Import all the necessary modules for the program.

Step2:

Load a image using imread() from cv2 module.

Step3:

Convert the image to grayscale.

Step4:

Using Canny operator from cv2,detect the edges of the image.

Step5:

Using the HoughLinesP(),detect line co-ordinates for every points in the images.Using For loop,draw the lines on the found co-ordinates.Display the image.

code :

```
# Read image and convert it to grayscale image
import cv2
import numpy as np
import matplotlib.pyplot as plt
from cv2 import cvtColor
image=cv2.imread("OIP.jpg")
cv2.imshow("ORIGINAL",image)

gray=cv2.cvtColor(image,cv2.COLOR_RGB2GRAY)
```

```

plt.figure(1)
plt.subplot(1,2,1)
plt.imshow(gray)
plt.title('Original')
plt.axis('off')

plt.subplot(1,2,2)
plt.imshow(image)
plt.title('gray')
plt.axis('off')

# Find the edges in the image using canny detector and display

edges = cv2.Canny(image, 120, 150)
plt.imshow(edges)
plt.title('EDGES')
plt.axis('off')

# Detect points that form a line using HoughLinesP

lines=cv2.HoughLinesP(edges,1,np.pi/180,threshold=80,minLineLength=50,maxLineGap=250)

# Draw lines on the image

for line in lines:
    x1,y1,x2,y2=line[0]
    cv2.line(image,(x1,y1),(x2,y2),(0,0,205),2)

# Display the result

plt.imshow(image)
plt.title('HOUGH')
plt.axis('off')

```

Output

Input image and grayscale image

Original



gray



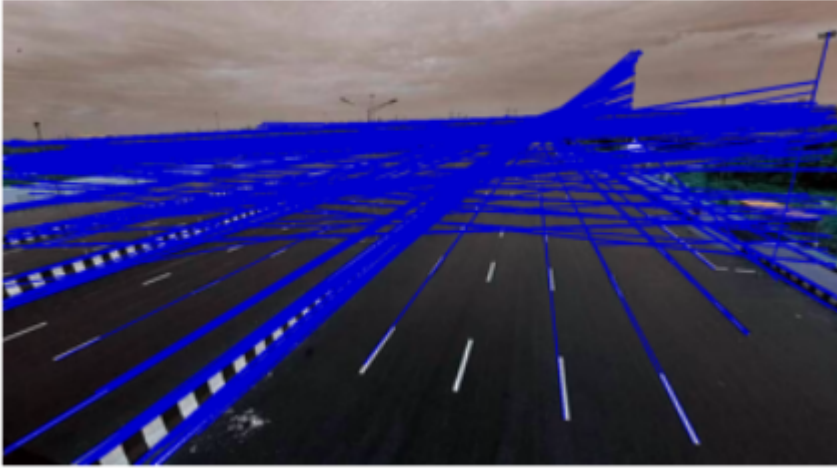
• Canny Edge detector output

EDGES



• Display the result of Hough transform

HOUGH



Result:

Thus the program is written with python and OpenCV to detect lines using Hough transform.