Road Lane Detection Code

```
import matplotlib.pyplot as plt
import cv2
import numpy as np

def region_of_interest(img, vertices):
  mask = np.zeros_like(img)
  match_mask_color = 235
  cv2.fillPoly(mask, vertices, match_mask_color)
  masked_image = cv2.bitwise_and(img, mask)
  return masked_image


def drow_the_lines(img, lines):
  img = np.copy(img)
  text_to_show = "The Vehicle is moving in correct diretion"
  blank_image = np.zeros((img.shape[0], img.shape[1],3), dtype=np.uint8)
  font = cv2.FONT_HERSHEY_SIMPLEX
  cv2.putText(blank_image,
              "Lane_Detect: " + text_to_show,
              (20,40),
              font,
              fontScale = 1,
              color = (255,255,255))
  for line in lines:
    for x1, y1, x2, y2 in line:
      cv2.line(blank_image, (x1,y1), (x2,y2),(0,235,0), thickness=10)

  img = cv2.addWeighted(img, 0.8, blank_image, 1, 0.0)
  return img

#image = cv2.imread('test.mp4')
#image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
def process(image):
  print(image.shape)
  height = image.shape[0]
  width = image.shape[1]
  region_of_interest_vertices = [
```

```
                                    (0, height),
                                    (width/2, height/2),
                                    (width, height)
    ]
    grey_image = cv2.cvtColor(image, cv2.COLOR_RGB2GRAY)
    canny_image = cv2.Canny(grey_image, 100, 200)
    cropped_image = region_of_interest(canny_image,

np.array([region_of_interest_vertices],np.int32),)
    lines = cv2.HoughLinesP(cropped_image,
                            rho=6,
                            theta = np.pi/180,
                            threshold=160,
                            lines=np.array([]),
                            minLineLength=40,
                            maxLineGap=25)
    image_with_lines = drow_the_lines(image, lines)
    return image_with_lines

#cap = cv2.VideoCapture("Sample_Video.mp4")
cap = cv2.VideoCapture(0)
while(cap.isOpened()):
    ret, frame = cap.read()
    frame = process(frame)
    cv2.imshow('frame', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```