

```

import cv2
import numpy as np

VIDEO_PATH = "./assets/identify_lines_video.mp4"
OUTPUT_VIDEO_PATH = "./output_video.avi"

cap = cv2.VideoCapture(VIDEO_PATH)

# Obtener las dimensiones del video
frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
fps = int(cap.get(cv2.CAP_PROP_FPS))

out = cv2.VideoWriter(OUTPUT_VIDEO_PATH,
cv2.VideoWriter_fourcc(*'XVID'), fps, (frame_width, frame_height))

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    frame_bw = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    blurred_frame = cv2.GaussianBlur(frame_bw, (5, 5), 0)
    canny_frame = cv2.Canny(blurred_frame, 225, 255)

    # Detectar líneas con la Transformada de Hough
    lines = cv2.HoughLinesP(canny_frame, 1, np.pi / 180, 80,
minLineLength=50, maxLineGap=10)
    if lines is not None:
        for line in lines:
            x1, y1, x2, y2 = line[0]
            # Dibujar las líneas detectadas en el cuadro original
            cv2.line(frame, (x1, y1), (x2, y2), (0, 255, 0), 5)

    # Mostrar el cuadro con las líneas detectadas
    cv2.imshow('Líneas Detectadas', frame)
    out.write(frame)

    # Salir si se presiona la tecla 'q'
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
out.release()
cv2.destroyAllWindows()

```