In [48]:

```
import cv2
import numpy as np
```

In [52]:

```
# Video input path
video_path = "roads.mp4"
video = cv2.VideoCapture(video_path)

# Video output path
output_path = "output_video.mp4"

# Get video properties (width, height, frames per second)
width = int(video.get(3))
height = int(video.get(4))
fps = int(video.get(5))

# Define codec and create VideoWriter object
fourcc = cv2.VideoWriter_fourcc(*'XVID')
output_video = cv2.VideoWriter(output_path, fourcc, fps, (width, height))

_, prev_frame = video.read()

while True:
    ret, curr_frame = video.read()
    if not ret:
        break
    frame_diff = cv2.absdiff(curr_frame, prev_frame)
    frame_diff_gray = cv2.cvtColor(frame_diff, cv2.COLOR_BGR2GRAY)

    # Erosion followed by dilation để lọc gọn ảnh
    # kernel = np.ones((2,2),np.uint8)
    # opening = cv2.morphologyEx(frame_diff_gray, cv2.MORPH_OPEN, kernel)

    # Blur frames
    opening = cv2.blur(frame_diff_gray, (10,10))

    _, threshold = cv2.threshold(opening, 15, 255, cv2.THRESH_BINARY)

    number_contours, hierachy = cv2.findContours(threshold.copy(), cv2.RETR_CCOMP,
                                                 cv2.CHAIN_APPROX_SIMPLE)

    test = curr_frame.copy()
    for i in range(len(number_contours)):
        if hierachy[0][i][3] == -1:
            x1,y1,w,h= cv2.boundingRect(number_contours[i])
            a = cv2.rectangle(test, (x1,y1), (x1+w, y1+h), color =[0,255,0], thickness=
1)

    # Your processing code here

    # Write the processed frame to the output video
    output_video.write(a)  # You can change this to write the processed frame

    prev_frame = curr_frame

    cv2.imshow("frame", a)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Release the VideoCapture and VideoWriter objects
video.release()
output_video.release()
cv2.destroyAllWindows()
```