import cv2

import numpy as np

# Load video or webcam

cap = cv2.VideoCapture(""C:\Users\axlin\Downloads\155546-810170148\_tiny.mp4"") # or use 0 for webcam

# Create background subtractor

fgbg = cv2.createBackgroundSubtractorMOG2(history=500, varThreshold=50, detectShadows=True)

while True:

ret, frame = cap.read()

if not ret:

break

# Resize for faster processing

frame = cv2.resize(frame, (640, 360))

# Apply background subtraction

fgmask = fgbg.apply(frame)

# Threshold to clean the mask

\_, thresh = cv2.threshold(fgmask, 200, 255, cv2.THRESH\_BINARY)

# Morphological operations to remove noise

kernel = np.ones((5,5), np.uint8)

clean\_mask = cv2.morphologyEx(thresh, cv2.MORPH\_OPEN, kernel)

# Find contours

contours, \_ = cv2.findContours(clean\_mask, cv2.RETR\_EXTERNAL, cv2.CHAIN\_APPROX\_SIMPLE)

# Draw bounding boxes

for cnt in contours:

area = cv2.contourArea(cnt)

if area > 500: # Filter small noise

x, y, w, h = cv2.boundingRect(cnt)

cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

cv2.putText(frame, 'Marine Object', (x, y - 10), cv2.FONT\_HERSHEY\_SIMPLEX, 0.6, (0, 255, 0), 2)

# Display

cv2.imshow('Marine Object Detection', frame)

if cv2.waitKey(30) & 0xFF == ord('q'):

break

cap.release()

cv2.destroyAllWindows()