

Nhóm 5 -PBL4-Lý Huỳnh Hữu Trí – Nguyễn Ngọc Hiếu – Tô Đông Trung

Code python :

Bài toán bài toán phát hiện chuyển động , theo dõi và đếm xe :

Object:

```
from random import randint
class Object:
    tracks = []
    def __init__(self, i, xi, yi, max_age):
        self.i = i
        self.x = xi
        self.y = yi
        self.tracks = []
        self.R = randint(0, 255)
        self.G = randint(0, 255)
        self.B = randint(0, 255)
        self.done = False
        self.state = '0'
        self.age = 0
        self.max_age = max_age
        self.dir = None

    def getRGB(self):
        return self.R, self.G, self.B

    def getTracks(self):
        return self.tracks

    def getId(self):
        return self.i

    def getState(self):
        return self.state

    def getDir(self):
        return self.dir

    def getX(self):
        return self.x
```

```

def getY(self):
    return self.y

def updateCoords(self, xn, yn):
    self.age = 0
    self.tracks.append([self.x, self.y])
    self.x = xn
    self.y = yn

def setDone(self):
    self.done = True

def timedOut(self):
    return self.done

def going_DOWN(self, mid_start):
    if len(self.tracks) >= 2:
        if self.tracks[-1][1] > mid_start and self.tracks[-2][1] <=
mid_start: # nếu vượt qua ranh giới
            self.dir = 'down'
            return True
        else:
            return False
    else:
        return False

def age_one(self):
    self.age += 1
    if self.age > self.max_age:
        self.done = True
    return True

```

Hàm (Function) phát hiện chuyển động , theo dõi và đếm xe:

```

import time

import cv2
import numpy as np

import Object

# argument parsing
def Detect_Tracking(vd):
    cap = cv2.VideoCapture(vd)
    # Print the capture properties to console, height, width and FPS

```

```

print('Height: ', cap.get(4))
print('Width: ', cap.get(3))
print('Frame per Seconds: ', cap.get(5))

cnt_up = 0
cnt_down = 0
w = cap.get(3)
h = cap.get(4)

# Entry / exit lines
line_down = int((h / 2))

pt1 = [0, line_down]
pt2 = [w, line_down]
pts_L1 = np.array([pt1, pt2], np.int32)
pts_L1 = pts_L1.reshape((-1, 1, 2))

up_limit = int(1 * (h / 5))
down_limit = int(4 * (h / 5))

line_down_color = (255, 0, 255)

# background subtraction
fgbg = cv2.createBackgroundSubtractorMOG2(200, 16, True)

# Structuring elements for morphographic filters
kernelOp = np.ones((6, 6), np.uint8)
kernelOp2 = np.ones((5, 5), np.uint8)
kernel_cl = np.ones((22, 22), np.uint8)

# Variables
font = cv2.FONT_HERSHEY_SIMPLEX
object = []
pid = 1

# lấy fps
prev_frame_time = 0

while cap.isOpened():
    # đọc hình ảnh từ video
    ret, frame = cap.read()
    new_frame_time = time.time()
    fps = 1 / (new_frame_time - prev_frame_time)
    prev_frame_time = new_frame_time
    FPS = fps

```

```

# FPS = cv2.VideoCapture.get(5)
# top_left, bottom_right = (0, 100), (460, 0)
for i in object:
    i.age_one()

# áp dụng background subtraction

fgmask2 = fgbg.apply(frame)
try:
    ret, imBin2 = cv2.threshold(fgmask2, 254, 255,
cv2.THRESH_BINARY)
    mask2 = cv2.morphologyEx(imBin2, cv2.MORPH_OPEN, kernelOp)
    mask2 = cv2.morphologyEx(mask2, cv2.MORPH_CLOSE,
kernel_cl)
    mask2 = cv2.erode(mask2, kernelOp2)
except:
    print("tổng xe trong video là : " + str(cnt_down +
cnt_up))
    print('END of File')
    break

# Contours
contours0, _ = cv2.findContours(mask2, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)
for cnt in contours0:
    area = cv2.contourArea(cnt)
    if area > 900:
        M = cv2.moments(cnt)
        cx = int(M['m10'] / M['m00'])
        cy = int(M['m01'] / M['m00'])
        x, y, w, h = cv2.boundingRect(cnt)
        new = True
        if cy in range(up_limit, down_limit):
            for i in object:
                if abs(cx - i.getX()) <= w and abs(cy -
i.getY()) <= h:
                    # đối tượng gần với đối tượng đã được
phát hiện trước đó

                    new = False
                    i.updateCoords(cx, cy) # cập nhật tọa độ
trong đối tượng

                    if i.going_DOWN(line_down):
                        cnt_down += 1
                        a = str(cnt_down)
                        b = time.strftime("%c")

```

```

        with open("Data.txt", 'a') as f:
            f.write("ID: " + a + "--DOWN-->" + str(b)
+ "\n")

            break
        if i.getState() == '1':
            if i.getDir() == 'down' and i.getY() >
down_limit:
                i.setDone()
            elif i.getDir() == 'up' and i.getY() <
up_limit:
                i.setDone()
        if i.timedOut():
            # xóa người khỏi danh sách
            index = object.index(i)
            object.pop(index)
            del i # giải phóng bộ nhớ
        if new:
            p = Object.Object(pid, cx, cy, 5)
            object.append(p)
            pid += 1
        # vẽ contour
        for i in object:
            color_contour = i.getRGB()
            cv2.circle(frame, (cx, cy), 5, color_contour,
-2)

            img = cv2.rectangle(frame, (x, y), (x + w, y +
h), color_contour, 2)
            cv2.putText(img, "Moving Object", (x, y - 40),
cv2.FONT_HERSHEY_COMPLEX, 0.5,
                        (0, 255, 0),
                        2, cv2.LINE_AA)
            # LINE_AA là giao diện như : màu sắc, độ dày,
loại đường

        # tracking code
        for i in object:
            if len(i.getTracks()) >= 2:
                pts = np.array(i.getTracks(),
np.int32)

                pts = pts.reshape((-1, 1, 2))
                frame = cv2.polylines(frame, [pts],
False, color_contour, 2)
            # display info
            str_down = ' DOI TUONG DI XUONG: ' + str(cnt_down)
            cv2.line(frame, (10, 35), (10, 55), (0, 0, 255), 2)

```

```

cv2.line(frame, (10, 55), (5, 45), (0, 0, 255), 2)
cv2.line(frame, (10, 55), (15, 45), (0, 0, 255), 2)
FPS = "FPS : " + str(FPS)

# ranh giới để phân biệt đối tượng đi đang đi lên hay đi xuống
ng

frame = cv2.polylines(frame, [pts_L1], False, line_down_color,
thickness=3)

cv2.putText(frame, FPS, (20, 20), font, 0.5, (255, 255, 255),
1, cv2.LINE_AA)
cv2.putText(frame, str_down, (20, 40), font, 0.5, (255, 255,
255), 1, cv2.LINE_AA)

cv2.imshow('Original Video', frame) # display original video
cv2.imshow('Masked Video', mask2) # display B & W video
# cv2.imshow('roi' , roi)

# press ESC to exit
k = cv2.waitKey(30) & 0xff
if k == 27:
    break

cap.release()
cv2.destroyAllWindows()

```

- Bài toán bài toán phát hiện trộm chuyển động ,phát chuông cảnh báo:

```

import time

import cv2
import numpy as np
import pygame
from pygame import mixer
def phat_hien_trom():

```

```

mixer.init()
mixer.music.load('Tieng-coi-xe-can-h-sat-www_tiangdong_com.mp3')
backSub = cv2.createBackgroundSubtractorMOG2()

top_left, bottom_right = (200, 100), (700, 680)

cap = cv2.VideoCapture('9014172086922407300.mp4')
kernelOp = np.ones((6, 6), np.uint8)
kernelOp2 = np.ones((5, 5), np.uint8)
kernel_cl = np.ones((22, 22), np.uint8)

while True:
    ret, frame = cap.read()
    fgMask = backSub.apply(frame)
    # Khu nhieu
    ret, imBin2 = cv2.threshold(fgMask, 254, 255,
cv2.THRESH_BINARY)
    fgMask = cv2.morphologyEx(imBin2, cv2.MORPH_OPEN, kernelOp)
    fgMask = cv2.morphologyEx(fgMask, cv2.MORPH_CLOSE, kernel_cl)
    fgMask = cv2.erode(fgMask, kernelOp2)

    contours, ret = cv2.findContours(fgMask, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

    # khu vực cấm
    cv2.rectangle(frame, top_left, bottom_right, (255, 255, 0), 2)

    for i in range(len(contours)):
        # x là điểm cuối cùng bên trái
        # y là điểm cuối cùng bên phải
        # x + với rộng , y + với cao

        (x, y, w, h) = cv2.boundingRect(contours[i])
        cx = x + w / 2
        cy = y + h / 2

        trong_vong_canh_bao = top_left[0] < cx < bottom_right[0]
and top_left[1] < cy < bottom_right[1]
        area = cv2.contourArea(contours[i])
        if area < 600:
            continue
        if trong_vong_canh_bao:
            img = cv2.rectangle(frame, (x, y), (x + w, y + h), (0,
255, 0), 5)

            mixer.music.play()
            object.append([(x, y), (x + w, y + h)])
            cv2.putText(frame, "WARNING !!!", (200, 100),
cv2.FONT_HERSHEY_SIMPLEX, 2, (0, 0, 255), 4)
            cv2.putText(img, "TROM DI CHUYEN", (x, y - 20),
cv2.FONT_HERSHEY_COMPLEX, .5,
                        (0, 255, 0),
                        2, cv2.LINE_AA)

```

```

        with open("Data_trom.txt", 'a') as f:
            b = time.strftime("%c")
            f.write("DOI_TUONG_VAO_NHA_LUC : " + b + "\n")
            print(object)
    else:
        pygame.mixer.stop()
        break
    cv2.imshow('Camera', frame)
    cv2.imshow('Masked Video', fgMask)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cv2.destroyAllWindows()

```

File main điều khiển luồng cho hai bài toán trên

```

from Human_Detection_and_Tracking import Detect_Tracking
from night_deetection_tracking import phat_hien_trom
if __name__ == '__main__':
    try:
        while 1:
            print("|-----|")
            print("| --- Mời bạn chọn video để thực hiện demo --- | ")
            print("| 1. video đối tượng chuyển động trên đường | ")
            print("| 2. video chuyển động vào trời tối | ")
            print("| 0. thoát lựa chọn | ")
            print("|-----|")
            case = int(input(" Nhập vào lựa chọn của bạn : "))
            if case == 1:
                Detect_Tracking("giaothongtrenduong.mp4")
                break
            elif case == 2:
                phat_hien_trom()
                break
            else:
                print("Nhập không đúng lựa chọn")
    except:
        print("END")

```