import cv2 as cv

from cv2 import FONT\_HERSHEY\_COMPLEX

from cv2 import COLOR\_RGB2BGR

from cv2 import COLOR\_BGR2RGB

import mediapipe as mp

#create a variable for hands in mp solutions module

# False means that the input is not static(webcam)

hands = mp.solutions.hands.Hands(False,2,1,0.5,0.5)

#create a mp draw to draw

mpDraw = mp.solutions.drawing\_utils

#to start the webcam

capture = cv.VideoCapture(0)

capture.set(3,1440)

while True:

isTrue, frame = capture.read()

frame = cv.flip(frame,1)

#height = 480,width = 640

frameRGB = cv.cvtColor(frame, cv.COLOR\_BGR2RGB)

#show results

results = hands.process(frameRGB)

# if any hand is detected

if results.multi\_hand\_landmarks != None :

count = {"RIGHT":0, "LEFT":0}

for hand\_index,hand\_info in enumerate(results.multi\_handedness):

hand\_label= hand\_info.classification[0].label

handlandmarks = results.multi\_hand\_landmarks[hand\_index]

#multi handedness contains only label

myHandx = []

myHand = []

myHandy = []

mpDraw.draw\_landmarks(frame,handlandmarks,mp.solutions.hands.HAND\_CONNECTIONS)

for Landmark in handlandmarks.landmark:

myHand.append((Landmark.x,Landmark.y))

myHandx.append(Landmark.x)

myHandy.append(Landmark.y)

fingertip\_ids = [myHandy[8],myHandy[12],myHandy[16],myHandy[20]]

thumb\_tip = myHandx[4]

thumb\_mcp = myHandx[2]

for ids in fingertip\_ids:

mcp = myHandy.index(ids) - 2

if ids < myHandy[mcp]:

count[hand\_label.upper()] += 1

if (hand\_label == "Right" and thumb\_tip<thumb\_mcp):

count["RIGHT"] += 1

if (hand\_label == "Left" and thumb\_tip>thumb\_mcp):

count["LEFT"] += 1

cv.putText(frame,str(sum(count.values())),(100,300),cv.FONT\_HERSHEY\_SIMPLEX,5,(255,0,0),2)

cv.imshow("Video",frame)

if cv.waitKey(20) & 0xFF ==ord('d'):

break

capture.release()

cv.destroyAllWindows()