Day 13: Object Tracking Basics

Outcomes:

- Track a colored object across video frames
- Use contours to find object position
- Identify the **largest object** of a given color
- Draw tracking visuals (center point, path)

Object Tracking

What is Object Tracking

Object tracking means:

- Detecting the same object
- In every frame
- And following its movement over time

You already have all the building blocks.

For color-based tracking:

1. Detect color (HSV)
2. Clean mask (morphology)
3. Find contours
4. Pick the largest contour
5. Track its position

Finding the largest contour

```
largest = None
max_area = 0

for cnt in contours:
    area = cv.contourArea(cnt)
    if area > max_area:
        max_area = area
        largest = cnt
```

This assumes: The object you want to track is the biggest detected color region.

Getting Object Position

```
x, y, w, h = cv.boundingRect(largest)
cx = x + w // 2
cy = y + h // 2
```

This gives the object’s center.

Drawing Tracking Visuals

```
cv.rectangle(frame, (x,y), (x+w,y+h), (0,255,0), 2)
cv.circle(frame, (cx,cy), 5, (0,0,255), -1)
```

Rectangle: Object Position

Circle: Center Point

21-objectDetection.py

```python
import cv2 as cv
import numpy as np

cap = cv.VideoCapture(0)

if not cap.isOpened():
    print("Camera not accessible.")
    exit()

cv.namedWindow("Trackbars")

def nothing(x):
    pass

cv.createTrackbar("LH", "Trackbars", 0, 179, nothing)
cv.createTrackbar("LS", "Trackbars", 30, 255, nothing)
cv.createTrackbar("LV", "Trackbars", 50, 255, nothing)
cv.createTrackbar("UH", "Trackbars", 179, 179, nothing)
cv.createTrackbar("US", "Trackbars", 255, 255, nothing)
cv.createTrackbar("UV", "Trackbars", 255, 255, nothing)

kernel = np.ones((7, 7), np.uint8)

while True:
    ret, frame = cap.read()
    if not ret:
        print("An error occurred.")
        break

    hsv = cv.cvtColor(frame, cv.COLOR_BGR2HSV)

    lh = cv.getTrackbarPos("LH", "Trackbars")
    ls = cv.getTrackbarPos("LS", "Trackbars")
    lv = cv.getTrackbarPos("LV", "Trackbars")
    uh = cv.getTrackbarPos("UH", "Trackbars")
    us = cv.getTrackbarPos("US", "Trackbars")
    uv = cv.getTrackbarPos("UV", "Trackbars")

    lower_bound = np.array([lh, ls, lv])
    upper_bound = np.array([uh, us, uv])

    mask = cv.inRange(hsv, lower_bound, upper_bound)
    opened = cv.morphologyEx(mask, cv.MORPH_OPEN, kernel)

    contours, hierarchy = cv.findContours(opened,
cv.RETR_EXTERNAL, cv.CHAIN_APPROX_SIMPLE)

    largest = None
    max_area = 0

    for cnt in contours:
        area = cv.contourArea(cnt)
        if area > max_area:
            max_area = area
            largest = cnt
    if largest is not None:
        x, y, w, h = cv.boundingRect(largest)

        cx = x + w // 2
        cy = y + h // 2

        cv.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0),
1)
        cv.circle(frame, (cx, cy), 3, (0, 0, 255), -1)

    cv.imshow("Frame", frame)

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

cap.release()
cv.destroyAllWindows()
```