# 5. MULTIPLE OBJECT TRACKING USING OPENCV

|  |  |
| --- | --- |
| **EX.N0 : 5** | **DESIGN AND IMPLEMENT MULTIPLE OBJECT TRACKING USING OPENCV** |
| **DATE : 25/02/2025** |

**AIM:**

To design and implement Multiple Object Tracking (MOT) using OpenCV to track multiple objects in a video stream in real-time.

# ALGORITHM:

Step 1: Import Libraries: Import OpenCV and NumPy. Step 2: Load Video: Capture video using cv2.VideoCapture.

Step 3: Select Objects: Manually select objects to track in the first frame using selectROI().

Step 4: Initialize Trackers: Create a separate tracker for each object and initialize them with the selected bounding boxes.

Step 5: Track Objects: Update each tracker in each frame to get the new position of the objects.

Step 6: Display and Exit: Draw bounding boxes around tracked objects and display the frame; exit on pressing q.

# PROGRAM:

import cv2

import numpy as np

cap = cv2.VideoCapture('input\_video.mp4') # Replace with 0 for webcam or video file path tracker\_type = 'CSRT' # Other options: 'KCF', 'MOSSE'

tracker = cv2.TrackerCSRT\_create() if tracker\_type == 'CSRT' else cv2.TrackerKCF\_create() ret, frame = cap.read()

if not ret:

print("Failed to read video") cap.release()

exit()

trackers = [] while True:

bbox = cv2.selectROI("Select Object to Track", frame, fromCenter=False, showCrosshair=True) trackers.append(cv2.TrackerCSRT\_create()) # Create a new tracker for each object

trackers[-1].init(frame, bbox) # Initialize tracker with the selected bounding box cv2.destroyWindow("Select Object to Track")

cv2.imshow("Tracking", frame) cv2.waitKey(1)

print("Press 'q' to start tracking after selecting all objects.") if len(trackers) > 0: # If at least one object is selected

key = cv2.waitKey(0)

if key == ord('q'): # Press 'q' to continue break

while cap.isOpened():

ret, frame = cap.read() if not ret:

break

for tracker in trackers:

ret, bbox = tracker.update(frame) # Get updated position of the object if ret:

x, y, w, h = [int(v) for v in bbox]

cv2.rectangle(frame, (x, y), (x + w, y + h), (255, 0, 0), 2) # Draw rectangle for the object else:

cv2.putText(frame, "Tracking failed", (10, 30), cv2.FONT\_HERSHEY\_SIMPLEX, 1, (0, 0,

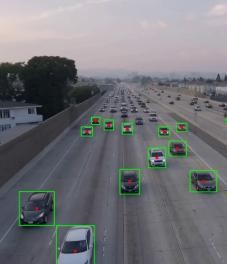
255), 2)

cv2.imshow("Multiple Object Tracking", frame) if cv2.waitKey(1) & 0xFF == ord('q'):

break cap.release()

cv2.destroyAllWindows()

# OUTPUT:

****

**RESULT:**

Thus the Program has been executed successfully and verified.