

Họ Tên : Nguyễn Sỹ Hà

MSSV : 2151013018

There are 2 assignments

1. Object Tracking (0.5)
2. Object Speed Estimation (0.5)

Input: Video clip (MP4 or camera). Choosing the object by mouse (drag rectangle on screen)

Output: ONE PDF (source codes and result screen shots)

Hint: using `cv2.calcOpticalFlowPyrLK` for 2 adjacent frames in clip.

Turn in: ONE PDF (source codes and result screen shots)

Result:







Source

```
import cv2
import numpy as np

# Initialize the tracker
tracker = cv2.TrackerMIL_create()

# Read the video
cap = cv2.VideoCapture('pedestrians.mp4')

# Read the first frame
ret, frame = cap.read()

# Select the bounding box
bbox = cv2.selectROI(frame, False)

# Initialize tracker with first frame and bounding box
ret = tracker.init(frame, bbox)

prev_bbox = None
```

```

while True:
    # Read a new frame
    ret, frame = cap.read()

    # Update tracker
    ret, bbox = tracker.update(frame)

    # Calculate speed if tracking succeeds
    if ret:
        # Calculate center positions
        center_x = (bbox[0] + bbox[2]) // 2
        center_y = (bbox[1] + bbox[3]) // 2

        # Calculate distance moved
        if prev_bbox is not None:
            prev_center_x = (prev_bbox[0] + prev_bbox[2]) // 2
            prev_center_y = (prev_bbox[1] + prev_bbox[3]) // 2
            distance = np.sqrt((center_x - prev_center_x)**2 + (center_y -
prev_center_y)**2)
        else:
            distance = 0

        # Calculate time delta (assuming FPS is known)
        fps = 30 # Replace with actual FPS
        time_delta = 1 / fps

        # Calculate speed
        if time_delta != 0:
            speed = distance / time_delta
        else:
            speed = 0

        # Draw bounding box and display speed
        p1 = (int(bbox[0]), int(bbox[1]))
        p2 = (int(bbox[0] + bbox[2]), int(bbox[1] + bbox[3]))
        cv2.rectangle(frame, p1, p2, (255,0,0), 2, 1)
        cv2.putText(frame, f"Speed: {speed:.2f} px/s", (100, 50),
cv2.FONT_HERSHEY_SIMPLEX, 0.75, (0, 255, 0), 2)

        # Update previous bounding box
        prev_bbox = bbox

    # Display result
    cv2.imshow("Tracking", frame)

    # Exit if ESC key is pressed
    if cv2.waitKey(1) & 0xFF == 27:

```

```
        break

# Release the VideoCapture object
cap.release()

# Close all OpenCV windows
cv2.destroyAllWindows()
```