Nguyễn Hoàng Thanh – 2151013087

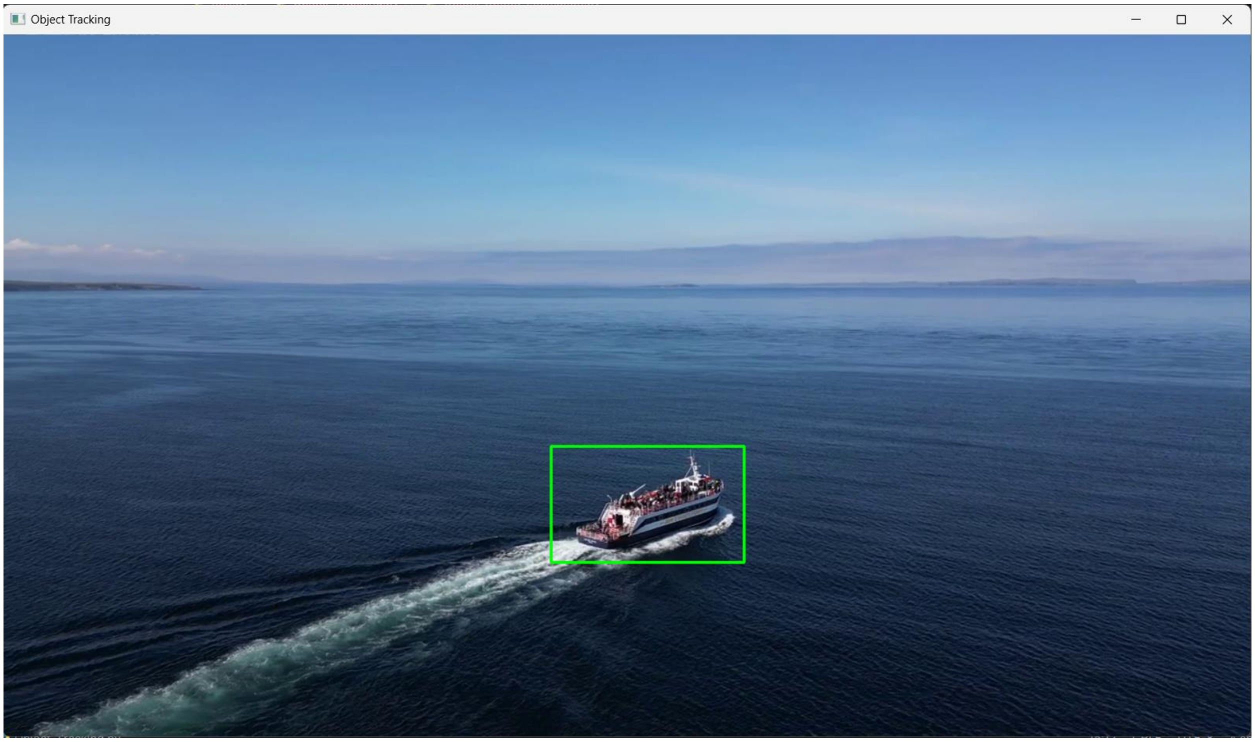
LAB 6: Lucas Kanade

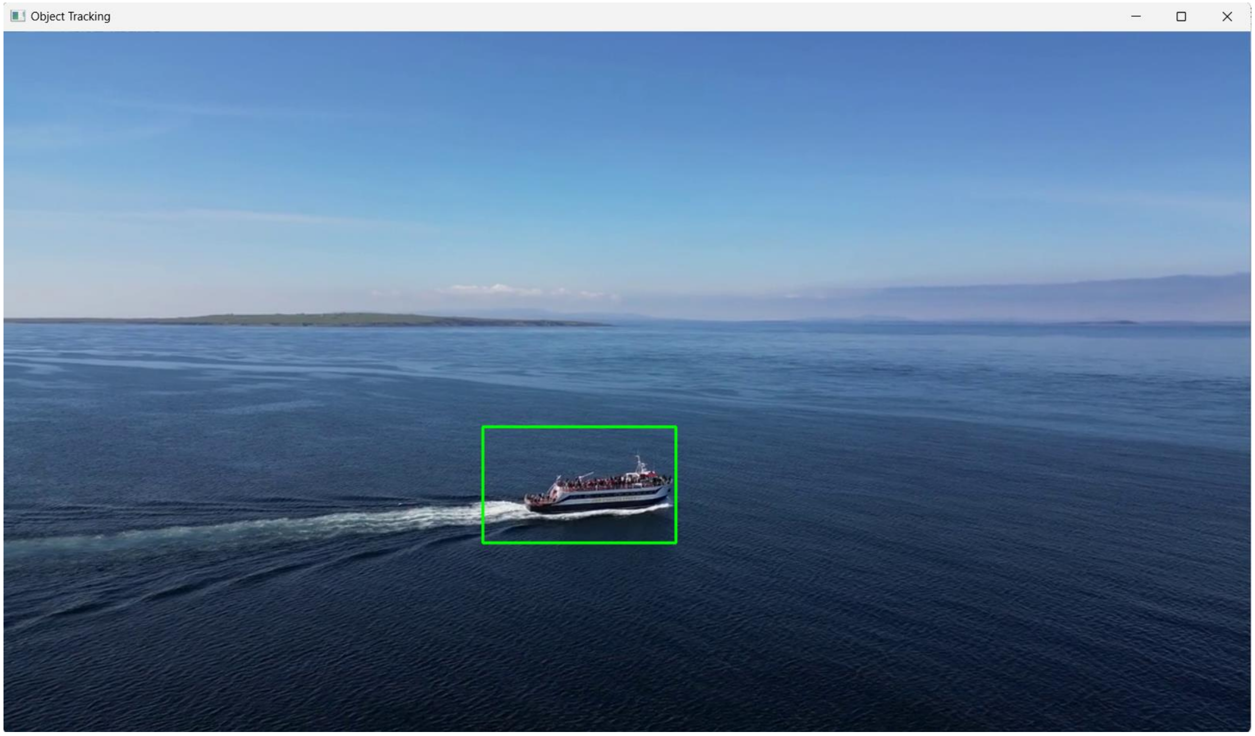
Assignment 1: Object Tracking

Source code:

```
import cv2
import numpy as np
def trackObject(I1, I2, roi):
    lk_params = dict(winSize=(15, 15), maxLevel=2,
criteria=(cv2.TERM_CRITERIA_EPS | cv2.TERM_CRITERIA_COUNT, 10, 0.03))
    grayI1 = cv2.cvtColor(I1, cv2.COLOR_BGR2GRAY)
    grayI2 = cv2.cvtColor(I2, cv2.COLOR_BGR2GRAY)

    optical_flow, status, err = cv2.calcOpticalFlowPyrLK(grayI1, grayI2, roi,
None, **lk_params)
    good_points = optical_flow[status == 1]
    roi = good_points.reshape(-1, 1, 2)
    return roi

cap = cv2.VideoCapture('Datasets/video.mp4')
ret, prev = cap.read()
if not ret:
    print("Failed to read video")
    cap.release()
    exit()

x, y, w, h = cv2.selectROI('Select ROI', prev, False)
roi = np.array([[[x + w / 2, y + h / 2]]], dtype=np.float32)
cv2.destroyWindow('Select ROI')

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

    roi = trackObject(prev, frame, roi)
    x, y = roi[0][0]

    cv2.rectangle(frame, (int(x - w / 2), int(y - h / 2)), (int(x + w / 2),
int(y + h / 2)), (0, 255, 0), 2)

    cv2.imshow('Object Tracking', frame)

    prev = frame.copy()
    if cv2.waitKey(30) & 0xFF == 27:
        break

cap.release()
cv2.destroyAllWindows()
```

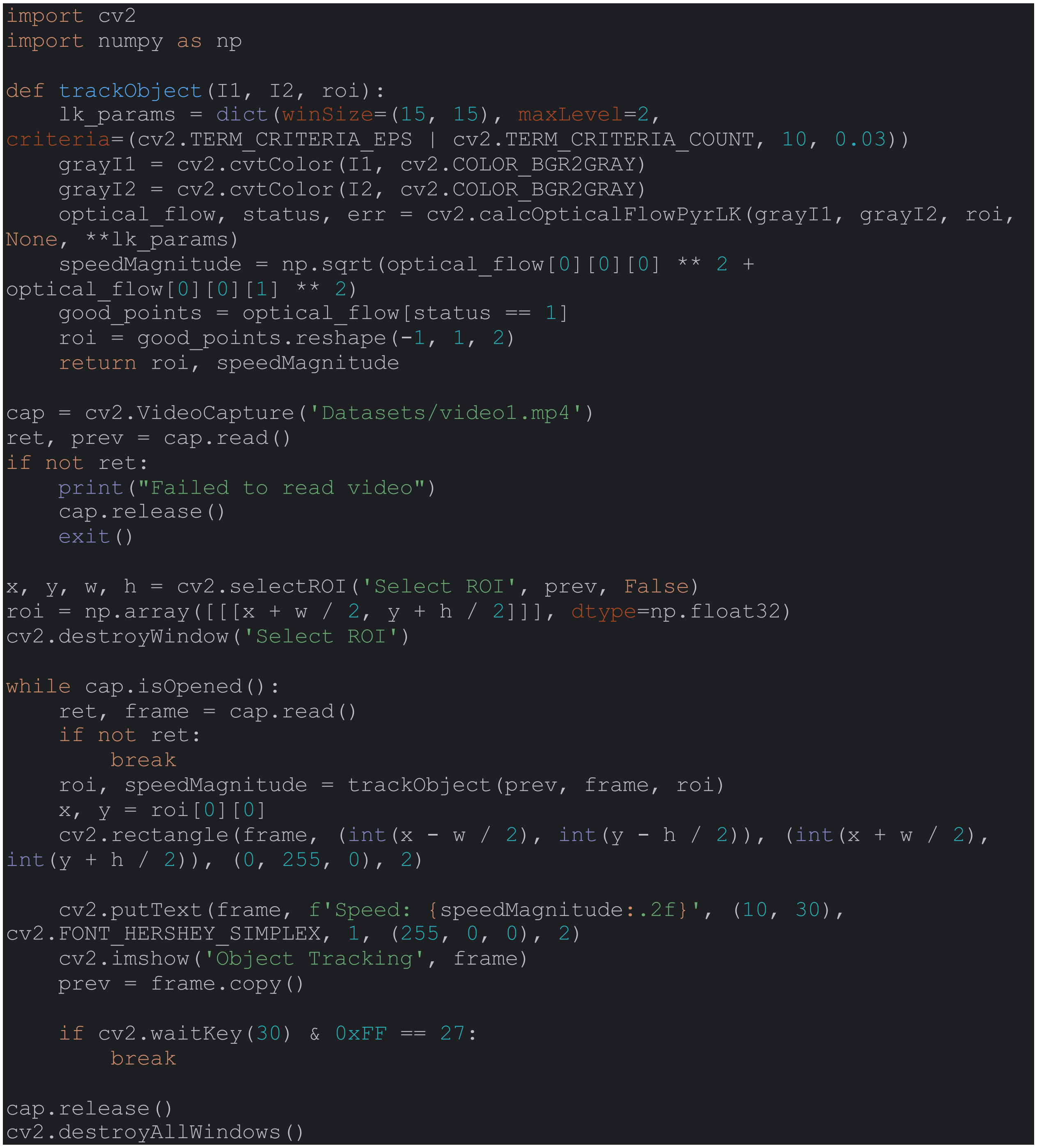
Result:





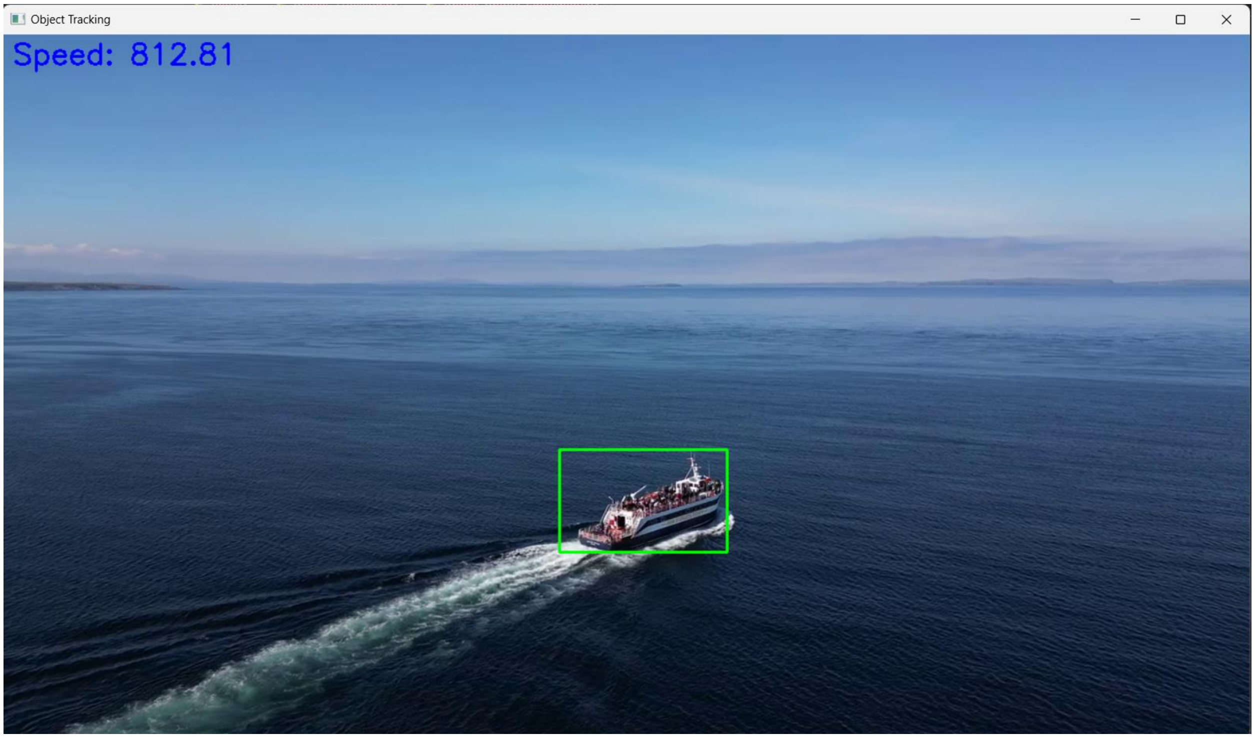
Assignment 2: Object Speed Estimation

Source code:

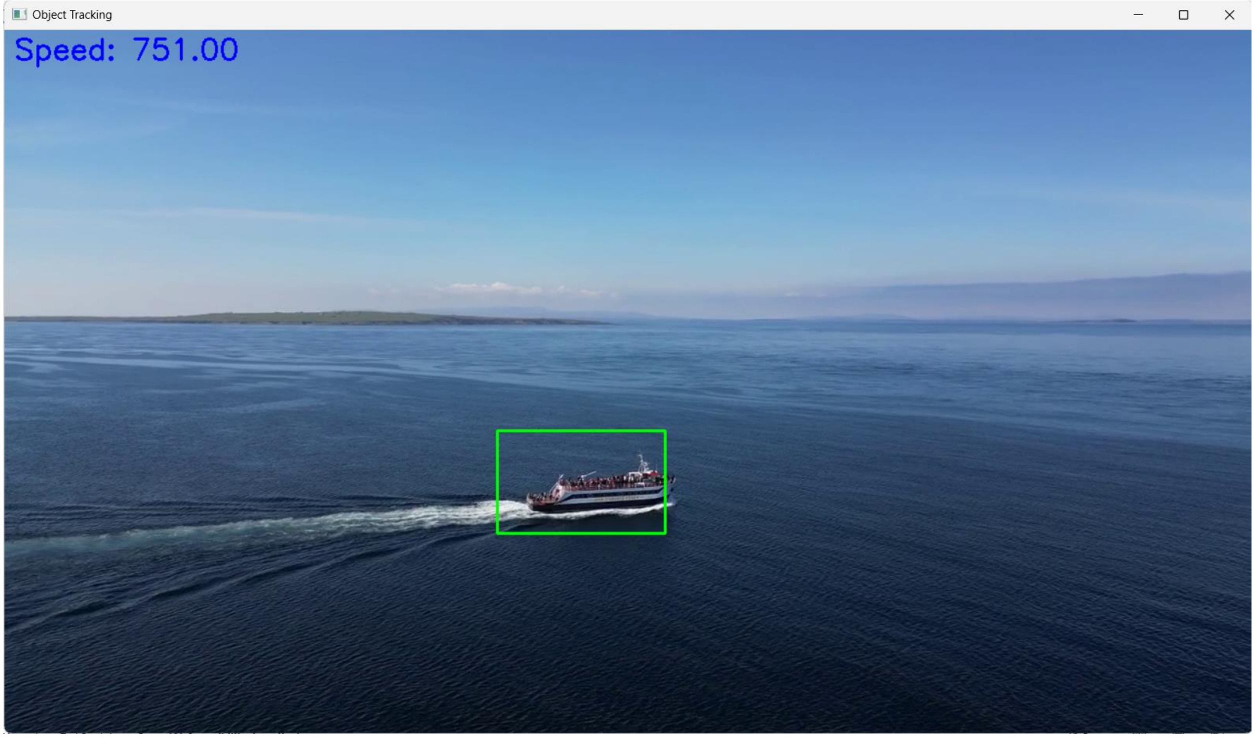


```
import cv2
import numpy as np

def trackObject(I1, I2, roi):
    lk_params = dict(winSize=(15, 15), maxLevel=2,
criteria=(cv2.TERM_CRITERIA_EPS | cv2.TERM_CRITERIA_COUNT, 10, 0.03))
    grayI1 = cv2.cvtColor(I1, cv2.COLOR_BGR2GRAY)
    grayI2 = cv2.cvtColor(I2, cv2.COLOR_BGR2GRAY)
    optical_flow, status, err = cv2.calcOpticalFlowPyrLK(grayI1, grayI2, roi,
None, **lk_params)
    speedMagnitude = np.sqrt(optical_flow[0][0][0] ** 2 +
optical_flow[0][0][1] ** 2)
    good_points = optical_flow[status == 1]
    roi = good_points.reshape(-1, 1, 2)
    return roi, speedMagnitude

cap = cv2.VideoCapture('Datasets/video1.mp4')
ret, prev = cap.read()
if not ret:
    print("Failed to read video")
    cap.release()
    exit()

x, y, w, h = cv2.selectROI('Select ROI', prev, False)
roi = np.array([[[x + w / 2, y + h / 2]]], dtype=np.float32)
cv2.destroyWindow('Select ROI')

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break
    roi, speedMagnitude = trackObject(prev, frame, roi)
    x, y = roi[0][0]
    cv2.rectangle(frame, (int(x - w / 2), int(y - h / 2)), (int(x + w / 2),
int(y + h / 2)), (0, 255, 0), 2)

    cv2.putText(frame, f'Speed: {speedMagnitude:.2f}', (10, 30),
cv2.FONT_HERSHEY_SIMPLEX, 1, (255, 0, 0), 2)
    cv2.imshow('Object Tracking', frame)
    prev = frame.copy()

    if cv2.waitKey(30) & 0xFF == 27:
        break

cap.release()
cv2.destroyAllWindows()
```

Result:



```
Speed: 812.81
```



```
Speed: 751.00
```