

```
In [1]: ▶ import cv2
import sys

# Load video, 0 for webcam, str for path to video
video = cv2.VideoCapture(0)
# video = cv2.VideoCapture('test_clip.mp4')

# Exit if video not opened.
if not video.isOpened():
    print('Could not open video!')
    sys.exit()

# Tracker Variables
tracker = None
roi = (0, 0, 0, 0)
# -1 for not tracking, 0 for init tracking, 1 for update tracking
tracking_flag = -1

# Loop simulate Camera Preview Callback
while True:

    # Capture user Key Press to simulate App Control
    key = cv2.waitKey(1) & 0xff
    # User Press Enter
    if key == 13:
        # Not tracking
        if tracking_flag == -1:
            # Pause and Let user select ROI
            roi = cv2.selectROI(frame, False)
            # Init tracking
            tracking_flag = 0
        # Is tracking
        if tracking_flag == 1:
            # Reset ROI
            roi = (0, 0, 0, 0)
            # Clear Tracker
            tracker.clear()
            # Stop tracking
            tracking_flag = -1
    # User Press ESC
    elif key == 27:
        break

    # Start timer
```

```

start = cv2.getTickCount()

# Read Next frame.
read_success, frame = video.read()
if not read_success:
    print('Cannot read video file!')
    sys.exit()

if tracking_flag == -1:

    # Display Text
    cv2.putText(frame, "Press ENTER to select ROI!", (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.75, (0, 0, 255), 2)

elif tracking_flag == 0:

    # Initialize KCF Tracker and Start Tracking
    # 1. Create a KCF Tracker
    # 2. Initialize KCF Tracker with grayscale image and ROI
    # 3. Modify tracking flag to start tracking
    # Your code starts here
    KCF_tracker = cv2.TrackerKCF_create()
    ready = KCF_tracker.init(frame, roi)
    tracking_flag = 1

else:

    # Update tracking result is succeed
    # If failed, print text "Tracking failure occurred!" at top left corner of the frame
    # Calculate and display "FPS@fps_value" at top right corner of the frame
    # Your code starts here
    ready, roi = KCF_tracker.update(frame)
    if(ready == False):
        cv2.putText(frame, "Tracking failure occurred!", (10, 40), cv2.FONT_HERSHEY_SIMPLEX, 0.75, (0, 0, 255), 2)
    FPS = cv2.getTickFrequency()/(cv2.getTickCount() - start)
    cv2.putText(frame, "FPS@" + str(FPS), (400, 100), cv2.FONT_HERSHEY_SIMPLEX, 0.75, (0, 0, 255), 2)

# Draw ROI Rectangle
p1 = (int(roi[0]), int(roi[1]))
p2 = (int(roi[0] + roi[2]), int(roi[1] + roi[3]))
cv2.rectangle(frame, p1, p2, (255, 0, 0), 2, 1)
# Display result
cv2.imshow("ECE420 Lab7", frame)

```