```
import cv2

#Motion Detect From Camera
cap=cv2.VideoCapture(0) #Live Capture from Camera
ret,frame1=cap.read() #Reading First frame
ret,frame2=cap.read() #Reading second frame


#Motion Detect From Video
cap2 = cv2.VideoCapture("test1.mp4")#Loading Video
ret,f1=cap2.read() #Reading First frame
ret,f2=cap2.read() #Reading second frame

def findMotion(frm1,frm2): #Function to Find Contour
    diff=cv2.absdiff(frm1,frm2)  #Finding the difference
    gray = cv2.cvtColor(diff, cv2.COLOR_BGR2GRAY) #Converting to gray color
    blur= cv2.GaussianBlur(gray,(5,5),0) #Bluring the gray
    _,thresh=cv2.threshold(blur,20,255,cv2.THRESH_BINARY) #Finding threshold
    dilated=(cv2.dilate(thresh,None,iterations=3)) #dilated the threshould
    contours,_=cv2.findContours(dilated,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)
#Finding contours
    #cv2.drawContours(frame1,contours,-1,(0,255,0),2) #contors also shows
    for contour in contours: #Fetching contour from contoues
        (x, y, w, h) = cv2.boundingRect(contour) #Finding contour area
        if cv2.contourArea(contour) < 2000: #Avoiding small motion areas by
adjusting the value < 2000
            continue
        cv2.rectangle(frm1, (x, y), (x + w, y + h), (0, 255, 0), 2) #Show rectangle
over the motion area
        cv2.putText(frm1, "Status:{}".format('Movement Occur'), (10, 20),
cv2.FONT_HERSHEY_SIMPLEX, .75, (255, 0, 0), 2) #Tracking the motion and shows the
status while a motion detects
        frm1[y:y + h, x:x + w] = cv2.GaussianBlur(frm1[y:y + h, x:x + w], (15, 15),
cv2.BORDER_DEFAULT) #Bluring the motion area (For Extra credit)

while cap.isOpened:  #Continues while capturing goes on
    findMotion(frame1, frame2) #function Call to find Motion From Camera
    findMotion(f1, f2) #function Call to find Motion from Video Clip named
test1.mp4
    cv2.imshow("Web Cam",frame1) #Shows the Camframe
    cv2.imshow("Video",f1) #Shows the Videoframe

    #cv2.imshow("Feed2", thresh) #Shows Threshold Frame
    frame1=frame2  #Camera Frame one becomes frame two
    ret,frame2=cap.read() #Reading the new frame to frame two

    f1=f2 #Video Frame one becomes frame two
    ret,f2=cap2.read() #Reading the new frame to frame two

    key=cv2.waitKey(1) #geting a key press and waits
    if key==ord('q'): #If pressing 'q' It breaks from the while loop
        break

cv2.destroyAllWindows() #Distroy's the opend window
cap.release() #Stops the camera
cap2.release() #Stops the video
```