# MOMENT TRACTING IN WEBCAM.  
  
*import* cv2  
*import* numpy *as* np  
  
cap = cv2.VideoCapture(0)  
  
ret, frame1 = cap.read()  
ret, frame2 = cap.read()  
  
*while*(cap.isOpened()):  
 diff = cv2.absdiff(frame1, frame2) # (absolute difference)function is use to find the diff. in 1st nd 2nd frame in video.  
 gray = cv2.cvtColor(diff, cv2.COLOR\_BGR2GRAY) # converting diff in grayscale mode because we find contours in latest stages of vedio.  
 # we know that gray scale is bst mode to find contours in image.  
 blur = cv2.GaussianBlur(gray, (5,5), 0) # bluring the diff image.  
  
 \_, thresh = cv2.threshold(blur, 20, 255, cv2.THRESH\_BINARY) # appliying threshhold technique in blur image.  
 # then we dilate our image to fill all holes in our thresh image to find out better contours in image.  
 dilated = cv2.dilate(thresh, *None*, iterations=3) # iterations mean layer which we want to cover holes (we now previous vedios)  
 contours, \_ = cv2.findContours(dilated, cv2.RETR\_TREE, cv2.CHAIN\_APPROX\_SIMPLE)  
  
 # we want rectangles in motion peoples not contours, so we itterate over all the contours.  
 *for* contour *in* contours:  
 (x, y, w, h) = cv2.boundingRect(contour) # it gives us x, y, width and height of frame.  
  
 *if* cv2.contourArea(contour) < 5: # means if area is very small than we don't want to do anything.  
 *continue* cv2.rectangle(frame1, (x, y), (x+w, y+h), (0, 255, 0), 2)  
 cv2.putText(frame1, 'STATUS: {}'.format('Movement'), (10, 20), cv2.FONT\_HERSHEY\_SIMPLEX, 1, (0, 0, 255), 3)  
 #cv2.drawContours(frame1, contours, -1, (0,255,0), 2 ) # we going to draw contour in only in our frame 1.  
  
 cv2.imshow('vedio', frame1)  
 frame1 = frame2  
 ret, frame2 = cap.read() # this means 3rd frame is assigned into frame2 variable and so on.  
  
  
 *if* cv2.waitKey(40) == 27: #the argument of waitkey is (40milisecond), the function wait for 40ms for next frame.  
 *break*cap.release()  
cv2.destroyAllWindows()