**Before Generating Graph**

1.To make more smooth:- change iterations

It run more time to smooth image

2. In thresh frame white color(255) is given if difference in intensity is more than 60. Increase difference more for more clear white part detection

import cv2,time

first\_frame=None        #for black part,Motion = difference in current frame and first frame

video=cv2.VideoCapture(0)

while True:

    check,frame=video.read()

    gray=cv2.cvtColor(frame,cv2.COLOR\_BGR2GRAY)

    gray=cv2.GaussianBlur(gray,(21,21),0)       #blur remove noise, (21,21)->width and height of gussian curve ,0->std deviation

    if first\_frame is None:     #for first frame

        first\_frame=gray

        continue        #so not run down code

    delta\_frame=cv2.absdiff(first\_frame,gray)

    thresh\_frame=cv2.threshold(delta\_frame,60,255,cv2.THRESH\_BINARY)[1]        #if difference is 30 or more,255 put white

    thresh\_frame-cv2.dilate(thresh\_frame,None,iterations=5)  #to smooth the image, more iteration more smooth white area

    # find conters

    (cnts,\_)=cv2.findContours(thresh\_frame.copy(),cv2.RETR\_EXTERNAL,cv2.CHAIN\_APPROX\_SIMPLE)   #retr external gives external counters.(boundary points)

    #white area with area less than 1000 are not human or animal so ignore them

    for contour in cnts:

        if cv2.contourArea(contour)<1000:

            continue

        (x,y,w,h)=cv2.boundingRect(contour)

        cv2.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),3)  #for countor of area>1000 show green boundary

    cv2.imshow("Capturing",gray)

    cv2.imshow("Threshold",delta\_frame)

    cv2.imshow("Threshold Frame",thresh\_frame)

    cv2.imshow("Me",frame)

    key=cv2.waitKey(1)

    if(key==ord('q')):

        break

video.release()

cv2.destroyAllWindows()