CHAPTER 1
INTRODUCTION

1.1 GENERAL

Real Time Human Motion Detection and Tracking.This document describes a real-time system for human detection,tracking and motion analysis from a saved video and the various process on that video.Generally the motion detection is the process of detecting a change in the position of an object relative to its surroundings or a changes in the surroundings relative to the object eg (**fig 1.1)**. Motion detection can be Achieved by either mechanical or electronic methods.

Fig 1.1 MOTION DETECTION

1.2 OUTLINE OF THE PROJECT

My project is about to detect human motion and tracking it by contours and give the location of the region in a frame in x and y axis by taking a particular scene of a video image.Human motion analysis is receiving increasing attention from computer vision researchers.Motion and analysis of human body parts involves the low level segmentation of the human body into segments connected by joints and recovers the 3D structure of the human body using its 2D projection over a sequences of images.

CHAPTER 2

AIM AND SCOPE

2.1 AIM

To detect the motion of a human and tracking by drawing contour on it. Saving the required scene as a image and finding the location of the region in a frame.

2.2 PROBLEM STATEMENT

Our job is to detect the motion of a human and track it by drawing contour on it e.g**(fig 2.1)** on a playing video which is extracted from a saved file.To find a even a very small movement which is not visible by our naked eye and saving scene as a image and find region of a particular location on a video.

Fig 2.1 TRACKING MOTION

2.3 LITERATURE REVIEW

My project is build by using a python as a flatform and opencv as a main source and some packages of opencv like cv2, numpy and matplotlib.

PYTHON is an interpreted, high level, general purpose programming language. we can use python for developing desktop GUI application, websites

and web application by taking care of common programming tasks. python is one of the best languages to implement computer vision.

OPENCV is a python wrapper for the original opencv c++ implementation. The library is cross platform and free for use under the opensource BSD license. Opencv is capable of image analysis and processing but have more like capture from cam. This means it is great at taking frames out of video or taking in two frames from a stereoscopic camera and run algorithim to extract information.

IMPORTED PACKAGES:

CV2 opencv is implemented using cv2 and numpy. To load, display, save an image. it is easier for user to understand opencv python than cv2 and makes it easier to find the packages with search engines. cv2 (old interface in old opencv version was namwd as cv) is the name that opencv developers chose when they created the binding generators.

NUMPY numpy is a library for the python programming languages,adding support for large,multidimensional arrays and matrices,along with a large collection of high level mathematical function to operate on these arrays.All the opencv arrays structures are converted into and from numpy arrays.so whatever operation you can do in numpy,you can combine it with opencv,which increase number of weapons in your arsenal.

MATPLOTLIB is a plotting library for the python programming language and its numerical mathematics extension numpy.it provides an object-oriented API for embedding plots into applications using general purpose GUI toolkits like Tkinter,wxpython,Qt or GTK+.In opencv we can display with matplotlib along with x and y axis range e.g**(fig 2.2)**. we can zoom images,save it etc using matplotlib.

Fig 2.2 *PLOTTED IMAGE*

INBUILD METHODS

VIDEOCAPTURE() this method graps the next frame from video file or camera and the return true(non-zero) in the case of success..that is,we call videocapture ::grap() for each camera and after that call the videocapture::retrieve() to decode and get frame from each camera.

IMREAD() used to avoid data structure and function name conflicts with others libraries,**opencv** has its own namespace: cv..Then create a mat object that will store the data of the loaded image.Mat image;now we call the **imread** function which loads the image name specified by the first argument(argv[1]).use the function **cv2.imread()** to read an image.the image should be in the working directory or a full path or image should be given.

IMSHOW(I) displays the grayscale image I in a figure.**imshow** uses the default display range for the image data type and optimize figure,axes and image object properties for image display..**imshow(**RGB) displays the truecolor image RGB in a figure. **imshow(**BW) displays the binary image BW in a figure.use the function **cv2.imshow()** to display an image in a window.The window automatically fits to the image size.

FINDCONTOURS( ) contours are defined as the line joining all the points along the boundary of an image that are having the same intensity.contours

come handy in shape analysis, finding the size of the object of interest and object detection. opencv has **findcontours()** function that helps in extracting the contours from the image .It works best on binary image, .so we should first apply thresholding techniques e.g**(fig 2.3)**, sobel edges,etc. The contours are a useful tool for shape analysis and object detection and recognition.

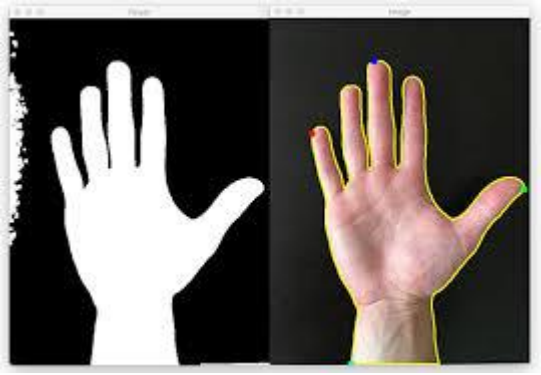

FIG 2.3 THRESHOLD IMAGE

In opencv finding contours is like finding white object from black background e.g **(fig 2.4).**

FIG 2.4 CONTOUR IMAGE

A contour is a closed curve of points or the line segments that represents the boundaries of an object in the image. contours are essentially the shapes of object in image.

WAITKEY( ) is a keyboard binding function in opencv it is used to introduce a delay of n milliseconds while rendering images to windows. Its argument is the

time in milliseconds for any keyboard event .if you press any key in that time,the program continues. if 0 is passed, it waits indefinitely for a key stroke.It can also be set to detect specific key strokes like, if key a is pressed etc.

DESTROYALLWINDOWS( ) simply destroy all the windows we created.If you want to destroy any specific windows, use the function cv2.destroyAllwindows() where you pass the exact window name as the argument.

2.4 SCOPE

The initial goal of computer vision was to enable machines to see the the visual world and interpret it the way a human would, but AII has advanced computer vision beyond human vision and now machines can be see things humans can't, like air quality and temperature.**fig 2.5** gives the brief details about computer vision in which computervision are used.

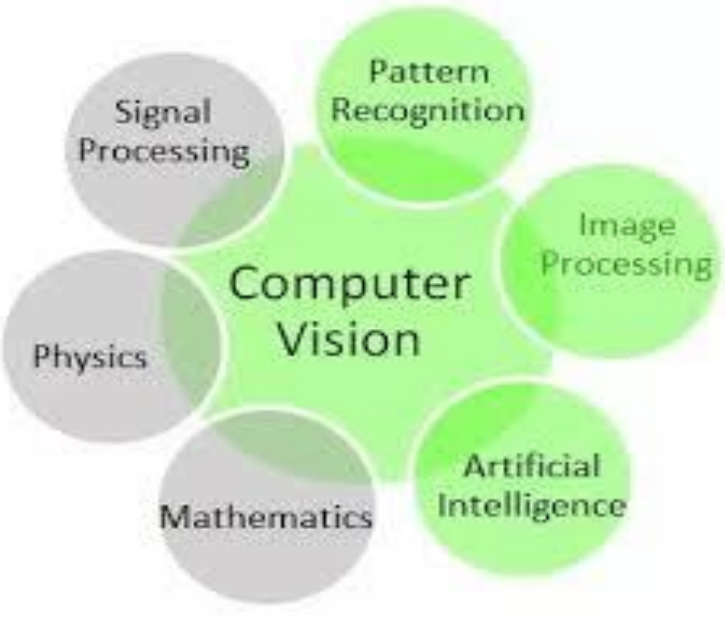

FIG 2.5 OVERVIEW OF COMPUTERVISION

CHAPTER 3
AIM AND ALGORITHIM

3.1 GENERAL

The first video will extract from file then read continuously two consecutive sequence of frame then find the absolute difference between the two frames to detect motion by using the method **cv2.absdiff()** and then track the motion by drawing contour on it. Before drawing the contours convert the frame into gray scale by using the method **cv2.cvtcolor()** then apply the blur to remove noises from the images by using **cv2.Gaussianblur()** by paasing gray scale image as a argument**.** the contour can be easily drawn in binary image rather than coloured image so threshold will apply on the image using this method **cv2.threshold().**Dilated of the image takes place because dilation add pixel to the boundaries of object in image. Then contours will drawn on the detected object based on our range given in contours area we can change the range in contour area to detect a very small movement. contours will be drawn by using **cv2.drawContours() method** by passing processed threshold image as a argument**.** Even we can draw rectangle bounding on the object using **cv2.boundingrect** method by passing x and y coordinates and contour width as a argument. Then we want to find the region of a particular location we can find by converting a scene in a video as a 2D plotted image by plt.imshow() and **plt.show ()** method this is possible by packages of **matplotlib**. Then the scene can saved as image from 2D plotting, we can find location by moving the curser on it then the result will be given as x and y coordinates. we can control keyboard events by using keyboard binding function **cv2.waitkey( )** then the all the created windows are cleared by the method **cv2.destroyAllwindows().**

3.2 OVERVIEW

First extract the video from file on which detection and tracking has to be made my passing video name or a path in which the video file has been located as argument else if you want to detect motion live video then pass camera number as a argument instead of filename. Then if you run the video will play then if you want to find the location of the region or you want save a particular scene as image then press **'esc'** then the image will display in a 2D plotted pattern then the required actions can take place. After completing our work we can press '**q'** to exit and close all the opened file.

3.3 PROJECT DESCRIPTION

computer vision tasks include methods for acquiring, processing, analysing and understanding digital images, and extraction of high dimensional data from the real world in order to produce numerical or symbolic information ,e.g in the forms of decision. The project is build using python and opencv this python program will allow you to detect motion and also store the time interval of the motion.

After playing the video the rectangle contour on the moving object will found like **fig(4.1)** then if user want to find the region of particular location then the particular scene should be converted to a plotted image **fig(4.2)** by pressing esc then location can be find by moving a curser on that image **fig(4.3)** then the result will get in the form of x and y axis then that image can be stored if required like as shown in the **fig(4.4)**. The motion can be detected either from a saved file or in live video.

3.4 WORK FLOW

The below flowchart gives the pictorial representation of the motion detection from a video.

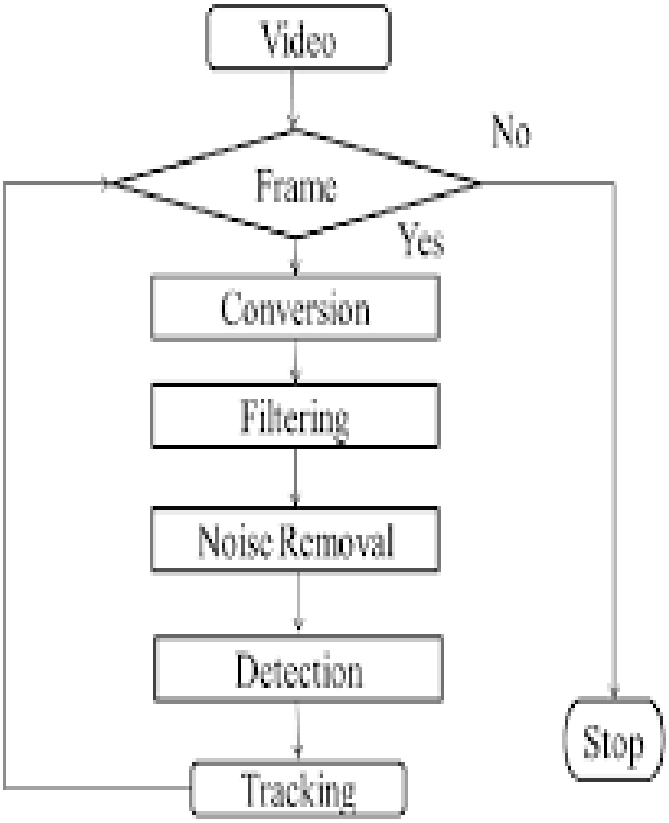

FIG 3.1 FLOWCHART FOR TRACKING

CHAPTER 4

SYSTEM IMPLEMENTATION

4.1 REQUIREMENTS

I. HARDWARE IMPLEMENTATION

1. Any operating system with minimum of 2GB RAM.
2. Keyboard
3. Mouse
4. Monitor

II. SOFTWARE IMPLENTATION

1. Any one browser installed in os
2. Python
3. Opencv

4.2 ROLE OF PYTHON

This python program will allow you to detect motion and also store the time interval of the motion. Requirement .Video can be treated as stack of pictures called frames. Analysis of all windows. Time record of movements. Python wrapper for the original opencv .It make use of numpy, which is a highly optimized library for numerical operation with a MATLAB-style syntax.

4.3 RESULT AND DISCUSSION

The result of the project is the motion of a human is detected and tracked by contour and rectangle bounding is drawn from a extracted video.

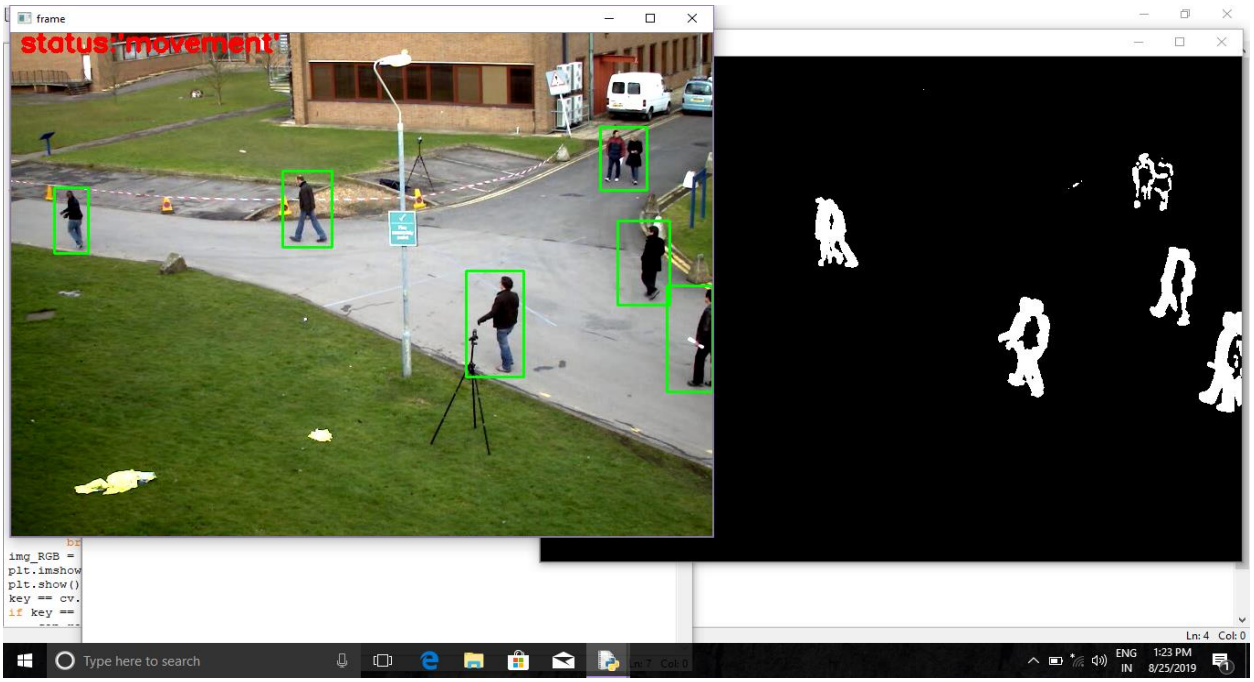

FIG 4.1 MOTION DETECTED AND TRACKED

The required scene are plotted as a image from a video using matplotlib.

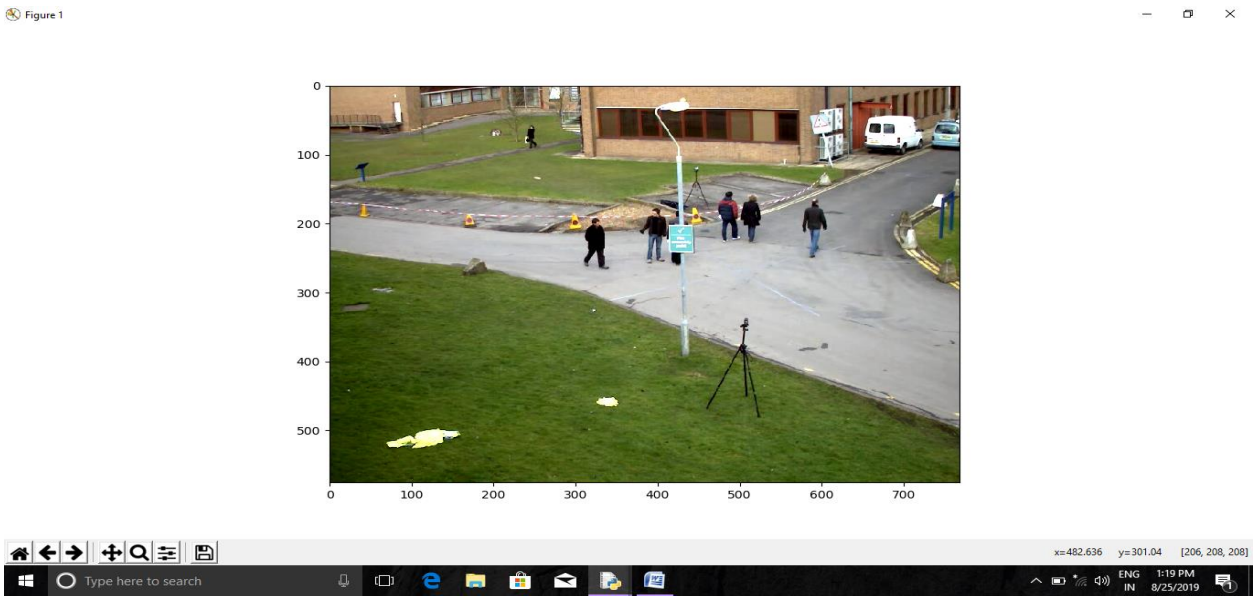

FIG 4.2 PLOTTED SCENE

The location of a particular region can be find the result will be in x and y axis

FIG 4.3 LOCATION IS FOUND

The image can be saved if required in our drive.

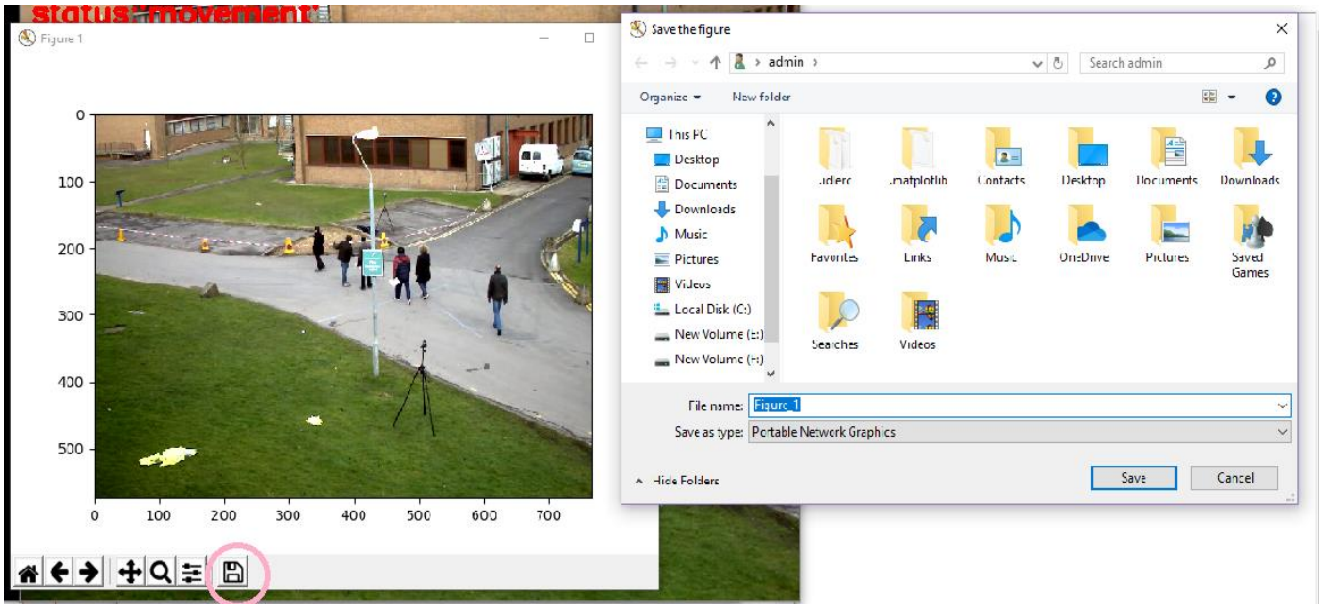

FIG 4.4 SAVING IMAGE

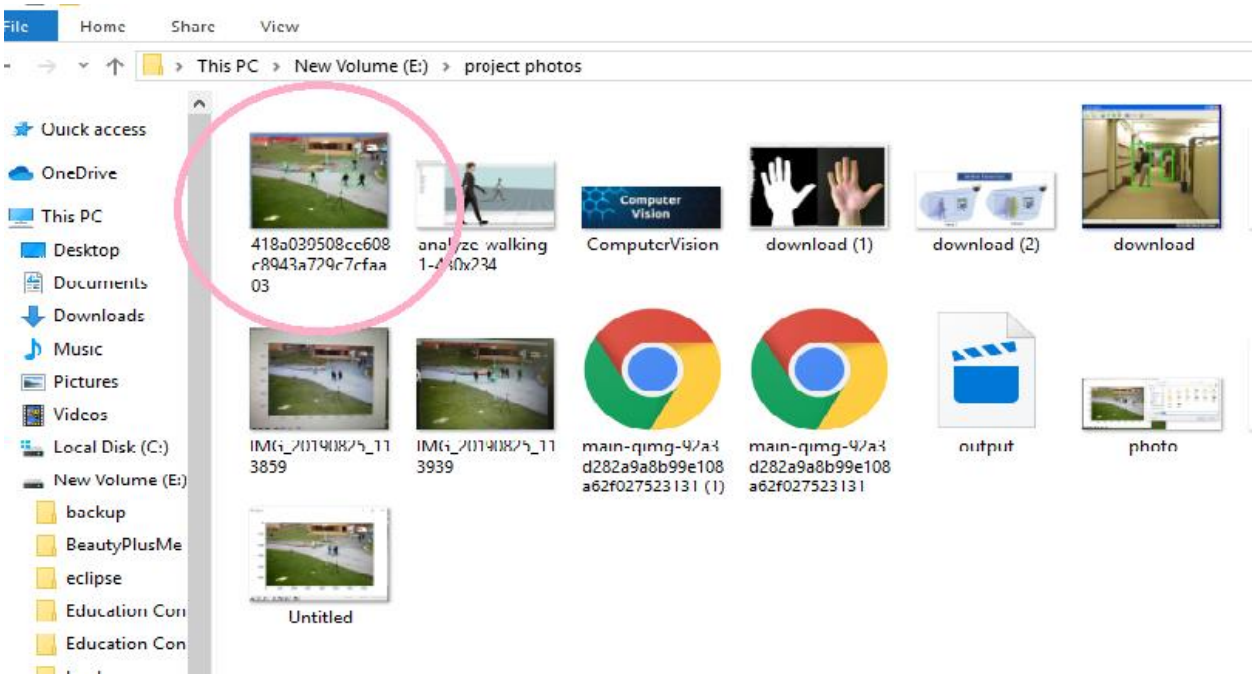

FIG 4.5 IMAGE SAVED

CHAPTER 5
CONCLUSION AND FUTURE WORK

5.1 CONCLUSION

The technique is simple. It is easily understood by the users who has not need much more knowledge about technical field. This project will be useful for any investigation purpose or any administrative purpose.

5.2 FUTURE WORK

This project can be further implemented by using DEEP LEARNING and MEACHINE LEARNING and text scanner for a better look and a better presentation. The latest technology computer vision used along with python in this project further to obtain appearance of detection.

REFERENCES

[1] PYTHON : http://www.imarticus.org

[2] COMPUTER VISION http://www.realpython.com

[3] MOTION DETECTION http://www.pyimagesearch.com

[4] COMPUTER VISION A modern approach:Forsyth and ponce,pearson

APPENDIX

A. SOURCE CODE

```
 import cv2 as cv
import  numpy as np
from matplotlib import pyplot as plt

 extracting video from file
cap = cv.VideoCapture('vtest.avi')
ret, frame1 = cap.read()
ret, frame2 = cap.read()

 finding difference in frames
while (cap.isOpened()):
diff = cv.absdiff(frame1,frame2)

   processing frames
  gray = cv.cvtColor(diff,cv.COLOR_BGR2GRAY)
  blur = cv.GaussianBlur(gray,(5,5),0)
```

```
_,thresh = cv.threshold(blur,20,255,cv.THRESH_BINARY)
cv.imshow('output',thresh)
dilated = cv.dilate(thresh,None,iterations=3)
```

Drawing contours

```
contours,_=cv.findContours(dilated,cv.RETR_TREE,cv.CHAIN_APPROX_SIMPLE)
cv.drawContours(frame1,contours,-1,(0,255,0),2)
for contour in contours:
 (x,y,w,h) = cv.boundingRect(contour)
  if cv.contourArea(contour)<900:
       continue
   cv.rectangle(frame1,(x,y),(x+w,y+h),(0,255,0),2)

cv.putText(frame1,"status:'movement'",(10,20),cv.FONT_HERSHEY_SIMPLEX,1,(0,0,255),3)
cv.imshow('frame',frame2)
cv.imshow('frame',frame1)
frame1 = frame2
ret,frame2 = cap.read()
key = cv.waitKey(100)
if key ==27:
    break
img_RGB = cv.cvtColor(frame1,cv.COLOR_BGR2RGB)
plt.imshow(img_RGB)
plt.show()
key == cv.waitKey(100)
if key == ord('q'):
cap.release()
cv.destroyAllWindows()
```