Indoor Surveillance System in Dynamic Environment

Akash Kumbhare1, Rohit Nayak1, Akshay Phapale1, Rupali Deshmukh2, Dr. Shashi Dugad3.
1 Student 2 Asst. Professor 3Professor, Department of Computer Engineering
Fr. Conceicao Rodrigues Institute of Technology1,2 Vashi, Navi Mumbai. 400708
TIFR (Tata Institute Of Fundamental Research)3, India

Abstract: In the past few years, it has wide applications like video surveillance in highly secure areas. This paper aims to detect any intrusion with a high accuracy and reduce false alarms. This paper proposes a method that uses moving object detection and extraction algorithm to detect the anomaly. It has always been difficult to monitor an indoor area like a bank locker or a room containing confidential documents which requires 24-hr surveillance. The most difficult operation is to monitor these areas even under dynamic environment to prevent any intruder to steal information. The background adjustment includes intensity, brightness and motion due to fan etc. to get a clear view of subject. The paper is currently proposed to integrate with single external camera only. The resolution of the images that will be processed are limited with respect to the camera capability. Efficient video surveillance system which is cost-effective as well as highly accurate, with a user-friendly GUI.

Keywords: Human Detection, Object Detection, Static camera, Reference Background, video surveillance, Blob.

I. INTRODUCTION

Insecurity and crimes constitute some major problems facing the immediate society today. People live with a fear of being attacked by burglars, vandals and thieves. Today in the society security is one of the major issues and having a 24*7 human eye is just impossible. In order to be secured of safety, it has become a necessity to realize and manage smart surveillance system. Despite all the efforts, resources and time that has been devoted to the development of tools that will reduce crime rates and make the world safer place to live, these problems are still increasing substantially. These give rise to the need for an increasing development in the technology of motion sensors. Even with the introduction of the alarm system which has greatly reduced the level of insecurity, there is still a problem of false alarm which need to be minimized. Also, changes in illumination, noise and compression artifacts make motion detection a challenging problem. In order to achieve robust detection, there is a need of a new technique i.e. Using various image processing algorithms which will detect motion. This system will provide security and ensure alarms are activated only when an unauthorized person try to gain access to the protected area.

II. LITERATURE SURVEY

A. Existing Systems

Surveillance in dynamic environment has been an important component in video surveillance system. For example, like [4] too uses background subtraction method and also saliency detection with no supervision as such. The only disadvantage is that these system are perfect for static background whereas for dynamic background results are poor. [3] Uses the concept of multiple agent system where static cameras & mobile robots are used which learn the environment first and learn and start surveillance based on the learnt environment, detects abnormalities and alerts. This concept is still to be implemented and is purely theoretical. [2] Uses modelling of background and various filters. This system was successful in implementing for dynamic environment but the number of false alarms was too high. The novel idea of this paper was to overcome these shortcomings and to propose such a system is purely based on image processing that does not use any expensive materials but only one external camera and otherwise pure algorithm. The algorithm consists of usage of two frames one which is a reference frame & the other being a current frame. The frames are adjusted w.r.t brightness and then compared. The output gives the value in the form of a Histogram which indicates if an intruder is detected or not. The system comes with a G.U.I with an authentication feature to check for genuine users only.
B. Loopholes In Existing Systems

The existing system [1] was successful in implementing a system for dynamic environment but had some flaws such as high number of false alarms which resulted in low accuracy. The system [2] was still a theoretical concept and was yet to be implemented which means results were not fixed, as well the cost of building the system was way too high since it uses 3 separate robots which needed learning process at first and then supervision and a human presence to monitor it. Although [3] requires no supervision and no training period but the results are still inconclusive and may lead into false alarms or may even fail to detect an intruder.

C. Need For Proposed Systems

This paper proposes such an idea which will overcome all the above loopholes by improvising existing system [4], by modifying its algorithm in such a way that it will reduce false alarm, being a low cost idea which may require human supervision to avoid unnecessary errors leading to disfunctioning of the system.

III. PROPOSED SYSTEM

This paper mainly deals with the constraint of dynamic environment. Dynamic Environment means an environment which is not fixed i.e. it is always in motion due to some or the other external factors. These external factors may include motion produced by fan which sets curtains in motion OR light i.e. variations in brightness or contrast. Currently, the two constraints/parameters decided for this paper is Intensity of light & motion produced by fan. Let us take an example of these two parameters :- 1) Consider a room for surveillance. The room may contain windows let’s assume. There are seasonal changes (weather conditions) i.e. change in light intensity. Let’s assume there is a tree outside the window whose image developed under light may be of a human. If that image is seen inside our room the surveillance may pick it up and may view it as an intruder. This will lead to a false alarm. 2) Consider the same room with curtains near the windows. There may be an electric fan being switched on which will produce some wind which will set the curtains on motion. Surveillance may pick it up as a motion of an intruder and may raise a false alarm. This paper aims to reduce these irregularities by using an existing algorithm which will be able to differentiate between an actual intruder and a pseudo intruder. Upon detecting an intruder it will raise an alarm which will notify the person in-charge. As soon as notifying is done video is saved for evidence purpose thus fulfilling the purpose of surveillance system.

IV. IMPLEMENTATION

A. Background Subtraction And Object Detection

The surveillance is done for monitoring an indoor area like a bank locker or a room containing confidential documents which requires 24-hr surveillance. The first step is to detect whether there is any motion in the area. Initially, the software switches on the webcam in video mode. The first step is to extract the frames from the video. Motion Detection is done in the following way:

1) MOG: It is a Gaussian Mixture-based Background/Foreground Segmentation Algorithm. It uses a method to model each background pixel by a mixture of K Gaussian distributions (K = 3 to 5). The weights of the mixture represent the time proportions that those colors stay in the scene. The probable background colors are the ones which stay longer and more static. While coding, there is a need to create a background object using the function, cv2.createBackgroundSubtractorMOG(). It has some optional parameters like length of history, number of gaussian mixtures, threshold etc. It is all set to some default values. Then inside the video loop, use backgroundsubtractor.apply() method to get the foreground mask.

B. Morphological Operation

Morphological transformations are some simple operations based on the image shape. It is normally performed on binary images. It needs two inputs, one is our original image, second one is called structuring element or kernel which decides the nature of operation. Two basic morphological operators are Erosion and Dilation.

1) Erosion: The basic idea of erosion is just like soil erosion only, it erodes away the boundaries of foreground object (Always try to keep foreground in white). So what it does? The kernel slides through the image (as in 2D convolution). A pixel in the original image (either 1 or 0) will be considered 1 only if all the pixels under the kernel is 1, otherwise it is eroded (made to zero).
So what happens is that, all the pixels near boundary will be discarded depending upon the size of kernel. So the thickness or size of the foreground object decreases or simply white region decreases in the image. It is useful for removing small white noises.

2) Dilation: It is just opposite of erosion. Here, a pixel element is '1' if at least one pixel under the kernel is '1'. So it increases the white region in the image or size of foreground object increases. Normally, in cases like noise removal, erosion is followed by dilation. Because, erosion removes white noises, but it also shrinks the object. So it is dilated. Since noise is gone, they won't come back, but the object area increases. It is also useful in joining broken parts of an object.

C. Contours

1) Finding Contours: Contours is found using void findContours(InputOutputArray image, OutputArrayOfArrays contours, int mode, int method)

where thresholding :- Image after background subtraction.
contours :- OutputArrayOfArrays
hierarchy :- OutputArray
CV_RETR_TREE :- Its an contour retrieval mode. Retrieves all of the contours and reconstructs a full hierarchy of nested contours.
CV_CHAIN_APPROX_SIMPLE:- Its an contour approximation method. Compresses horizontal, vertical, and diagonal segments and leaves only their end points.
The function retrieves contours from the binary image. The contours are a useful tool for shape analysis and object detection and recognition.
2) Draw Contours: Contours is drawn using void drawContours(InputOutputArray image, InputArrayOfArrays contours, int contourIdx, const Scalar& color, int thickness=1, int lineType=8, InputArray hierarchy=noArray(), int maxLevel=INT_MAX, Point offset=Point() )

where contourIdx – Parameter indicating a contour to draw. If it is negative, all the contours are drawn.
colour – Colour of the contours.
thickness – Thickness of lines the contours are drawn with. If it is negative (for example, thickness=CV_FILLED ), the contour interiors are drawn.
lineType – Line connectivity.
hierarchy – Optional information about hierarchy. It is only needed if you want to draw only some of the contours .
maxLevel – Maximal level for drawn contours. If it is 0, only the specified contour is drawn. If it is 1, the function draws the contour(s) and all the nested contours. If it is 2, the function draws the contours, all the nested contours, all the nested-to-nested contours, and so on.

3) Calculations: Step 1 -> Taking the original image as input.
Step 2 -> calculating the rows of the frame using original. Rows
Step 3 -> calculating the columns of the frame using origin. Columns
Step 4 -> calculating the area of the frame.
Step 5 -> Taking the only 5% area of the original frame using area = rows*columns*0.05
step 6 -> Calculating the contours area using contourArea(contours[i]) for number of iteration i.e. equal to contours.size() stored it in area1.
Step 7 -> if(area1 > area) then intruder is detected and intruder flag is set to true.

D. Algorithm

Step 1. Start the surveillance using webcam.
Step 2. Read frames one by one.
Step 3. Apply BackgroundSubtractorMOG method on this colored frame.
Step 4. Apply morphological operation like dilation or erosion to remove noise.
Step 5. Apply contours to get idea of area through which intrusion is detected.
Step 6. We take 5% of area of original frame. Let us take area1.
Step 7. If in background, area covered by white pixel say area is greater than above area1 then we set the flag.
Step 8. Ring the alarm.
Step 9. Stop.
E. Output

1) Surveillance ON:

2) Intrusion:

3) BackgroundMOG:

V. DESIGN:

A. FlowDiagram

Fig 1: Flow diagram of Intrusion Detection in varying intensity

Figure 1 shows that how the intrusion is detected in varying light intensity. Here after authentication, surveillance starts and Background Subtraction MOG method is applied and then morphological...
VI. CONCLUSION

The existing systems have some or the other cons such as inaccurate or high cost or have algorithmic errors which needs to be overcome so as to get an efficient surveillance system. The main aim of this paper is to design such a system which will work mainly in dynamic environment in indoor areas such as bank lockers or a room containing fans curtains where false alarms needs to be reduced which may be induced due to motion of curtains by motion of fans or change in light intensity due to external factors like seasonal changes.

REFERENCES