

Autonomous Real Time Surveillance System

**Nilesh Kadam, Sneha Madesh, Rishikesh Walawalkar, Shreemant Gawand,
Nandakishor Karlekar**

Padmabhushan Vasantdada Patil pratisthan's College Of Engineering
Eastern Express Highway , Near Everard Nagar , Sion –Chunabhatti ,Mumbai – 400022

Abstract: Video surveillance[1] is becoming increasingly important for the security-sensitive areas such as airports, banks, casinos, and parking lots. Many efforts have been made in this field, to detect and track the human and human activities in scene and recognize simple motion such as walking and running. In this paper we present a autonomous real-time surveillance[3] sys-tem framework for recognizing complex activities in. In our system we use javacv and opencv to detect moving objects in the scene, and use tracking algorithm to record activities for further analysis. Compared with the traditional surveillance system, an autonomous system surveillance offers much better flexibility in video content processing and transmission[5]. At the same time, it, also, can easily implement advanced features such as real time transmission of video and audio if connected through known ip address

I. INTRODUCTION

The main contribution of this work is the integration of surveillance system with an alert system that not only provides security but also alerts the user in real times using face recognition[2]. If any person crosses in front of your camera or , the software will alert you. Immediately, an SMS is sent to the owner of the setup regarding the intrusion such that the owner can open the application in his cell and view the snapshot of the video happenings. Also, an alarm is activated at the PC's end to alert the people around regarding intrusion. This application allows the user to track the activities happening at a particular location. Take the snapshots of the video recorded through IP Camera in a PC.

An Web camera, is a type of digital video camera commonly employed for surveillance, and which unlike analog Closed circuit Television[5] (CCTV) cameras can send and receive data via a computer network and the Internet. Store these snapshots as images in the Database.

In General Issues of Existing System:

- Existing surveillance systems are often comprised of black and white, poor quality analogue videos with little or no signal processing, recorded on the same cassette.
- Most of the recorded images are of insufficient quality to hold as evidence in a law court.
- The effectiveness and response of the operator is largely dependant on his/her vigilance rather than the technological capabilities of the surveillance system.
- It is expensive to have human operators monitoring real-time camera footage 24/7.

II. PROPOSED SYSTEM

The system presented in this Project is an autonomous and real-time surveillance system[3], capable of analyzing video streams. These streams are continuously monitored in specific situations for several days , learning to characterize the actions taking place there. This system also infers whether events present a threat that should be signaled to user. The concept of this system is to detect security breaches and alert the user about it through SMS and mail. The system is basically implemented at small level using a web camera for capturing video and then motion detection is performed .Security breach when identified, immediately alerts in the form of SMS and mail are sent to the user.

III. PROPOSED ALGO FOR MOTION DETECTION

Motion detection[6] is carried out as follows

Step 1) Capture frame

Step 2) Set minimum threshold.

Step 3) Construct three Image objects used during motion detection say prevImg, currImg and diffImg .

prevImg, which holds the 'previous' frame.

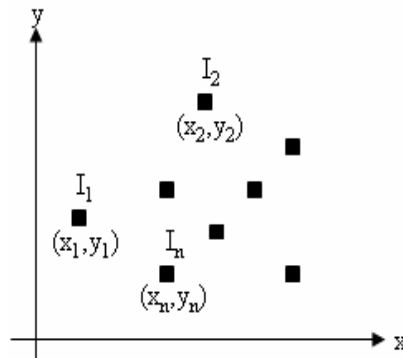
diffImg, which holds the difference image, is set to be an empty grayscale.

currImg which holds the current image.

Step 4) The 'current' frame (the current webcam image) is compared with the previous frame, differences are detected, and used to calculate a center-of-gravity point.

Step 5) Calculates difference between images by comparing their intensity which as follows:

Let's assume that each pixel has an intensity (I_1, I_2, \dots, I_n) and a (x, y) coordinate ($(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$), as shown



The sum of all the pixels' moments around the y-axis can be written as:

$$M_y = I_1 * x_1 + I_2 * x_2 + \dots + I_n * x_n$$

The pixels' moments around the x-axis is:

$$M_x = I_1 * y_1 + I_2 * y_2 + \dots + I_n * y_n$$

The total intensity of the system (shape) is the sum of the intensities of its pixels:

$$I_{sys} = I_1 + I_2 + \dots + I_n$$

Knowing I_{sys} and M_y allows us to obtain the distance of the shape from the y-axis:

$$x = \frac{M_y}{I_{sys}}$$

In a similar way, the distance of the shape from the x-axis is:

$$y = \frac{M_x}{I_{sys}}$$

Step 6) The ones needed for the center-of-gravity calculation ($m(0, 0), m(1, 0), m(0, 1)$) are retrieved with the necessary p and q values.

$$m(p, q) = \sum_{i=1}^n I(x, y) x^p y^q$$

The $m()$ moments function takes two arguments, p and q , which are used as powers for x and y .

$$M_y = m(1, 0)$$

$$M_x = m(0, 1)$$

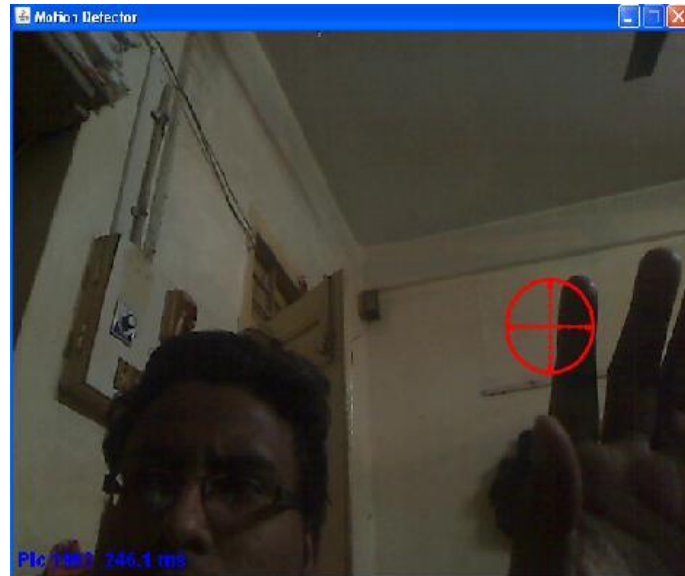
$$I_{sys} = m(0, 0)$$

$$x = \frac{m(1, 0)}{m(0, 0)}$$

$$y = \frac{m(0, 1)}{m(0, 0)}$$

$m()$ function calculate the center-of-gravity point is \bar{x}, \bar{y} , which it returns as a Point object.

Step 7) If the returned Point object is more than the decided threshold then is detected as motion.



IV. PROPOSED ALGO FOR FACE RECOGNIZATION

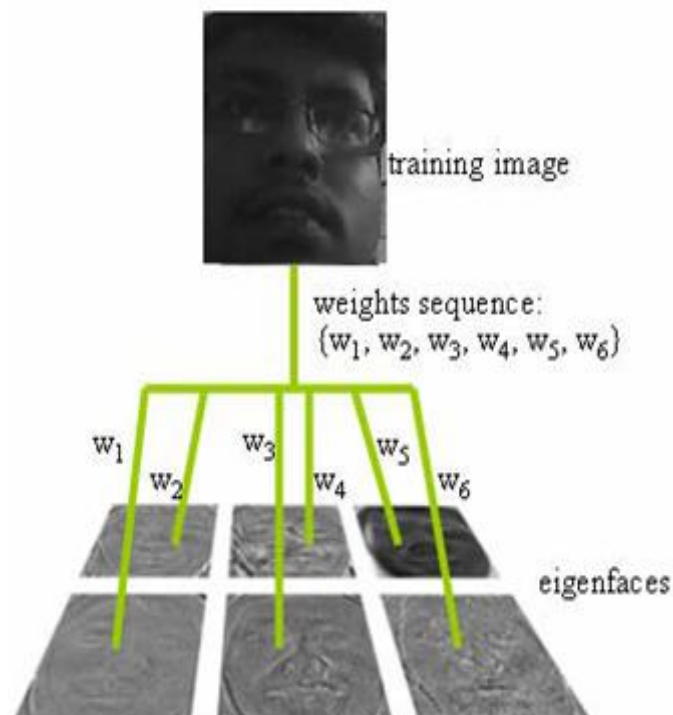
Face Recognition[2] is carried out as follows

Step 1) Create Training Images which are to be stored along with designated name of the person, with different facial expression.

Step 2) It's important that the training images should be cropped and orientated in a similar way, so that the variations between the images are caused by facial differences rather than differences in the background or facial position.

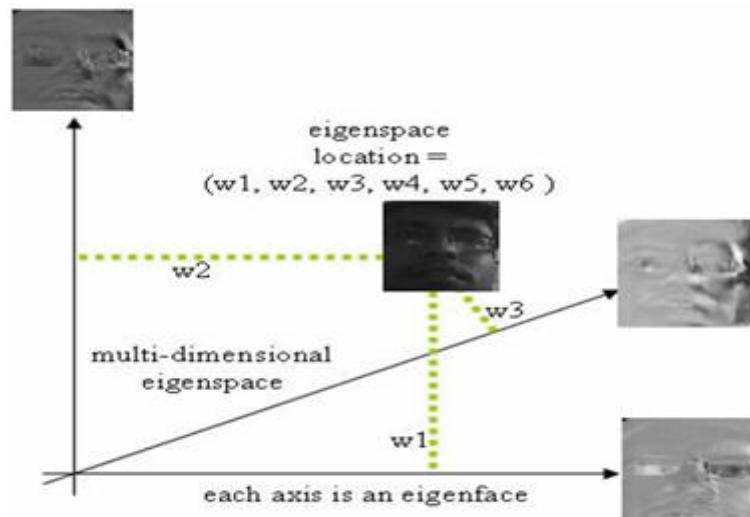
Step 3) The training process creates *eigenfaces* (also called as *ghost faces*) which are composites of the training images which highlight elements that distinguish between faces.

Step 4) The idea is that a training image can be decomposed into the weighted sum of multiple eigenfaces, and all the weights are stored as a sequence.

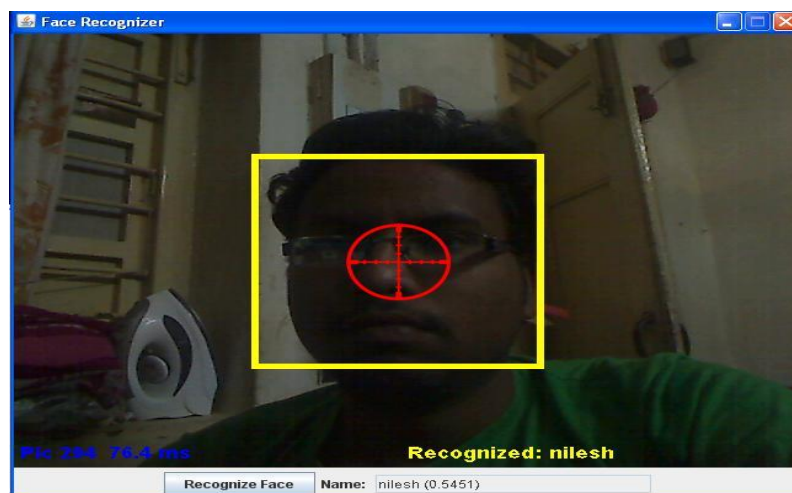


Step 5) The weights can now be viewed as the image's coordinates in the eigenspace.

Step 6) New picture is captured and new face is decomposed into eigenfaces, with a weight assigned to each one denoting its importance.



Step 7) The resulting weights sequence is compared with each of the weights sequences for the training images, and the name associated with the 'closest' matching training image is used to identify the new face.



V. CONCLUSION

Autonomous surveillance systems significantly contribute to situation awareness. Such systems transform video surveillance from a data acquisition tool to information and intelligence acquisitions Systems. Real-time video analysis provides smart surveillance systems with the ability to react to an activity in real-time, thus acquiring relevant information at much higher resolution. The long-term operation of such systems provides the ability to analyze information in a spatiotemporal context. As such systems evolve, they will be integrated both with inputs from other types of sensing devices and also with information about the space in which the system is operating, thus providing a very rich mechanism for maintaining situation awareness.

REFERENCES

- [1] Remagnino, Jones, Paragios, and Regazzoni, *Video Based Surveillance Systems Computer Vision and Distributed Processing*. Norwell, MA: Kluwer, 2002.
- [2] Blanz and Vetter, "Face recognition based on fitting morphable model," *IEEE PAMI*, vol. 25, no. 9, pp. 1063–1074, Sept. 2003.
- [3] R. Collins et al. "A system for video surveillance and monitoring." VSAM Final Report, Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep. CMU-RI-TR-00-12, May 2000.
- [4] R. T. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, and L. Wixson. *A system for video surveillance and monitoring*, Carnegie Mellon Univ., Pittsburgh, PA, Tech. Rep., CMU-RI-TR-00-12, 2000
- [5] G. D. Hager, and P. N. Belhumeur. *Efficient region tracking with parametric models of geometry and illumination*, *IEEE Trans. PAMI*, vol. 20, pp. 1025-1039, Oct. 1998.
- [6] J.-G. Kim, H.S. Chang, J. Kim, and H.-M. Kim. Threshold-based camera motion characterization of MPEG video. *ETRI Journal*, 26(3):269{272, 2004.