Volume : 1, Issue : 1
January - June 2011

ISSN : 2229 - 3515

international journal of ADVANCES IN SOFT COMPUTING TECHNOLOGY

Editor-in-Chief
Dr.Vaka Murali Mohan

Published by
BHAVANA RESEARCH CENTER

Real Time Event Detection in Video-Based Surveillance Systems

Prasada Rao, T, P[1]., Harinath Reddy B [1]and John Paul, P [2]
1.TRR Engineering College, Inole(V), Patancheru (M), Medak, AP, INDIA
2.The Engineering Academy, Eurekaport, Ameerpet, Hyderabad, AP, INDIA

KeyWords:
Suspicious, Event Detection, Video-Based Surveillance, Segmentation.

Abstract: *In this paper, we present a surveillance system that supports a human operator by automatically detecting abandoned objects and drawing the operator's attention to such events. It consists of three major parts: foreground segmentation based on Gaussian Mixture Models, a tracker based on blob association and a blob-based object classification system to identify abandoned objects. For foreground segmentation, we assume that video sequences of the background shot under diferent natural settings are available a priori. The tracker uses a single-camera view and it does not difer- entiate between people and luggage. The classification is done using the shape of detected objects and temporal tracking results, to successfully categorize objects into bag and non-bag(human). If a potentially abandoned object is detected, the operator is notified and the system provides the appropriate key frames for interpreting the incident.*

1. Introduction:

Rapid developments in the digital video technologies, large-scale multi-camera networks are now more common. There is an increasing demand for automated multi-camera/sensor event-modeling technologies that can efficiently and effectively extract events and activities occurring in the surveillance network. Automated video surveillance has emerged as a trendy application domain in recent years, and accessing the semantic content of surveillance video has become a challenging research area. Surveillance for safety and security is a major requirement of public transport and other public places to address the modern demands of mobility in major urban areas and to effect improvements in quality of life and environment protection. The surveillance task is a complex one involving technology, management procedures and people. Visual surveillance based on Television system is an important part of such systems, but visual processing is not sufficient and the geographical distribution of devices and management has to be taken into account. Different semantic levels of interpretation are required according to the complexity of the corresponding applications.

***Prof. T. P. PRASADA RAO**
Head, Dept. of CSE
TRR Engineering College (NBA Accredited)
Inole (V), Patancheru (M), Hyderabad, AP
Ph. No.: 91-9490321360
E-mail: tpprasadarao@gmail.com

The effectiveness and real-time response of our system are demonstrated by extensive experimentation on indoor and outdoor video shots in the presence of multi-object occlusion, different noise levels, and coding. The results of the considerable amount of research dealing with automated access to video surveillance have appeared in the literature; however, significant semantic gaps in event models and content based access to surveillance video remain such as Zhong Zhang et al [1] described a real-time system that fuses tracking information from multiple cameras, thus vastly expanding the capabilities of IVS by allowing the user to define rules on the map of the whole area, independent of individual cameras. T. D'Orazio and M. Leo [2] presented a survey of soccer video analysis systems for different applications: video summarization, provision of augmented information, high-level analysis. Computer vision techniques have been adapted to be applicable in the challenging soccer context. Ediz Şaykol et al [3] proposed a scenario-based query-processing system for video surveillance archives. Jayavardhana Gubbi et al [4] observed that the smoke is visible well before flames can be sighted. A novel method was proposed for smoke characterization using wavelets and support vector machines. Uğur Töreyin, B et al [5] proposed a method to detect fire and/or flames in real-time by processing the video data generated by an ordinary camera monitoring a scene. Ioannis Tziakos et al [6] introduced the use of dimensionality reduction for video event detection without explicitly using motion estimation or object tracking. Jacob M. Gryn et al [7] presented methods for recovering direction maps from video, constructing direction map templates (defining target motion patterns

of interest) and comparing templates to newly acquired video (for pattern detection and localization). Sergio A. Velastin et al [8] presented a surveillance architecture that reflects the distributed nature of the monitoring task and allows for distributed detection processes, not only dealing with visual processing but also with devices such as acoustic signature detection and mobile smart cards, actuators and a range of other possible sensors. Yun Zhai et al [9] presented a composite event detection system for multi-camera networks. The proposed framework is capable of handling relationships between primitive events generated from a single camera view; multiple camera views and nonvideo sensors with spatial and temporal variations. Tao Xiang and Shaogang Gong [10] reported the problem of surveillance video content modeling. Aishy Amer et al [11] investigated a real-time system to detect context-independent events in video shots. Claudio Sacchi et al [12] presented a real-time post-processing error-recovery algorithm explicitly devoted at enhancing the performances of outdoor video-surveillance systems working in remote modality. The aim of the proposed algorithm is to distinguish between changed blocks due to variations in the observed scene and noise-altered blocks that contain errors caused by channel noise. The proposed video-based surveillance system is shown in figure 1 as video surveillance system block diagram, which works in real time environment is able to distinguish transitory and stopped foreground objects from static background objects in dynamic scenes, detect and distinguish left objects, track objects and generate object information, classify detected objects (into bag and non-bag) in video imagery. Our system's use is limited only to stationary cameras and video inputs from PTZ (Pan/Tilt/Zoom) cameras, where the view frustum may change arbitrarily is not supported.

2. Object Detection

The system is initialized by feeding video imagery from a static camera monitoring a site. To distinguish foreground objects from stationary background, we use a combination of adaptive Gaussian mixture models and low-level image post-processing methods to create a foreground pixel map at every frame. The group the connected regions in the foreground map to extract individual object features such as bounding box, area, and perimeter and color histogram. The object tracking algorithm utilizes extracted features together with a correspondence matching scheme to track objects from frame to frame. The output of the tracking step is analyzed further for detecting suspicious events which in our case is 'a bag being abandoned by a person '.

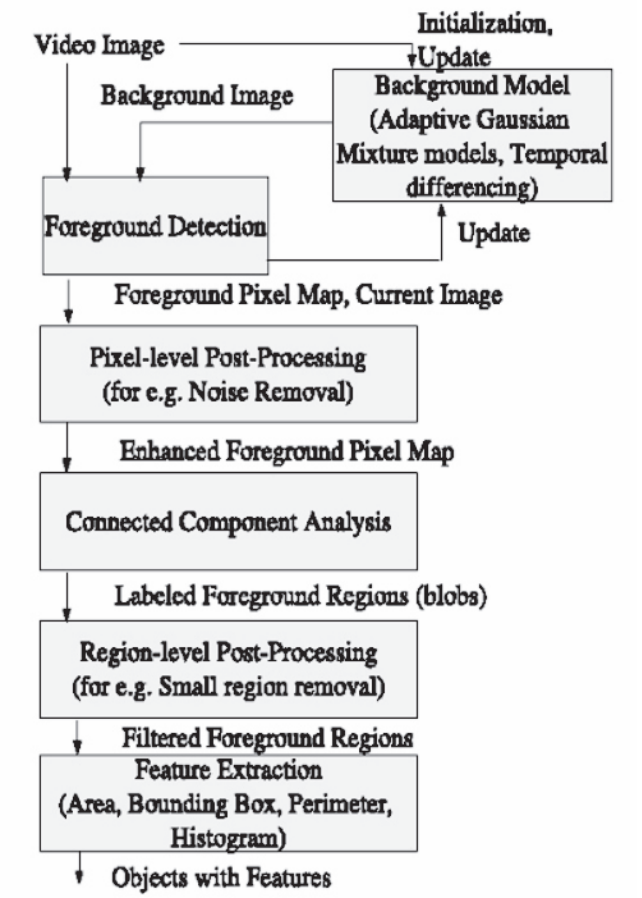

Fig.1 Video Surveillance System Block Diagram

Foreground Detection

The background scene related parts of the system is isolated and its coupling with other modules is kept minimum to let the whole detection system to work flexibly with any one of the background models. Next step in the detection method is detecting the foreground pixels by using the background model and the current image from video. This pixel level detection process is dependent on the background model in use and it is used to update the background model to adapt to dynamic scene changes. Also, due to camera noise or environmental effects the detected foreground pixel map contains noise. Pixel-level post-processing operations are performed to remove noise in the foreground pixels. Once we get the filtered foreground pixels, in the next step, connected regions are found by using a connected component labeling algorithm and objects' bounding rectangles are calculated. The labeled regions may contain near but disjoint regions due to defects in foreground segmentation process. Hence, some relatively small regions caused by environmental noise are eliminated in the region-level post-processing step. In the final step of the detection process, a number of object features (like area,

bounding box, perimeter and color histogram of the regions corresponding to objects) are extracted from current image by using the foreground pixel map. A sample fore- ground region detection is shown in fig.(3.3). We use a combination of a background model and low-level image post-processing methods to create a foreground pixel map and extract object features at every video frame. Background models generally have two distinct stages in their process: initialization and update. Following sections describe the initialization and update mechanisms together with foreground region detection methods are used.

Temporal Differencing

Temporal differencing makes use of the pixel-wise difference between two or three consecutive frames in video imagery to extract moving regions. A two-frame temporal differencing method is implemented in this system. Let In(x) represent the gray-level intensity value at pixel position (x) and at time instance n of video image sequence I, which is in the range [0, 255]. The two-frame temporal differencing scheme suggests that a pixel is moving if it satisfies the following:

$$\left|I_n(x) - I_{n-1}(x)\right| \geq T_n(x) \qquad (1)$$

Hence, if an object has uniform colored regions, the Equation (1) fails to detect some of the pixels inside these regions even if the object moves. The per-pixel threshold, T, is initially set to a pre-determined value and later updated as follows:

$$T_{n+1}(x) = \alpha T_n(x) + (1-\alpha)(y.\left|I_n(x) - I_{n-1}(x)\right|, x \in BG \qquad (2)$$

$$T_m(x)\ x \in FG$$

3. Adaptive Gaussian Mixture Model

The values of an individual pixel (e.g. scalars for gray values and vectors for color images) over time is considered as a "pixel process" and the recent history of each pixel, X1, . . . , Xt, is modeled by a mixture of K Gaussian distributions. The probability of observing current pixel value then becomes: where wit is an estimate of the weight (what portion of the data is accounted for this (Gaussian) of the ith Gaussian (Git) in the mixture at time t, μit is the mean value of Git and it is the covariance matrix of Git and is a Gaussian probability density function:. Decision on K depends on the available memory and computational power. The covariance matrix assumed to be of the following form which assumes that red, green and blue color components are independent and have the same variance. The procedure for detecting foreground pixels is as follows. At the beginning of the system, the K Gaussian distributions for a pixel are initialized with predefined mean, high variance and low prior weight. When a new pixel is observed in the image sequence, to determine its type, its RGB vector is checked against K Gaussians, until a match is found. A match is defined as a pixel value within (=2.5) standard deviations of a distribution. Next, the prior weights of the K distributions at time t(wkt), are updated as follows

$$w_{kt} = (1-\alpha)\, w_{k\tau} + \alpha\, M_{k\tau} \qquad (3)$$

where is the learning rate and M(k, t) is 1 for the matching Gaussian distribution and 0 for the remaining distributions. In order to detect the type (foreground or background) of the new pixel, the K Gaussian distributions are sorted by the value w/. This ordered list of distributions reflect the most probable background from top to bottom since by Equation (3.6) background pixel processes make the corresponding Gaussian distribution have larger prior weight and less variance. Then the first B distributions are chosen as the background model, where and T is the minimum portion of the pixel data that should be accounted for by the background. If a small value is chosen for T, the background is generally uni-modal.

4. Detecting Connected Regions

After detecting foreground regions and applying post-processing operations to remove noise and shadow regions, the filtered foreground pixels are grouped into connected regions (blobs) and labeled by using connected component labeling algorithm [12]. After finding individual blobs that correspond to objects, the bounding boxes of these regions are calculated.

Region Level Post-Processing

Even after removing pixel-level noise, some artificial small regions remain due to inaccurate object segmentation. In order to eliminate this type of regions, regions that have smaller sizes than a pre-defined threshold are deleted from the foreground pixel map.

Extracting Object Features

Once we have segmented regions we extract features of the corresponding objects from the current image. These features are size (S), center-of-mass or just centroid (Cm) and color histogram (Hc). Calculating the size of the object is trivial and we just count the number of foreground pixels that are contained in the bounding box of the object. In order to calculate the center-of-mass point, Cm = (xC, yC), of an object O. The color histogram, Hc, is calculated over RGB intensity values of object pixels in current image. In order to reduce computational complexity of operations that use Hc, the color values are quantized. Let N be the number of bins in the histogram, then every bin covers 255 Color values. The color histogram is calculated by iterating over pixels of O and

incrementing the stored value of the corresponding color bin in the histogram, Hc. So for an object O the color histogram is updated as follows: $Hc(ci, cj, ck) = Hc(ci, cj, ck) + 1$ $c(=(ci, cj, ck)), = O$

where c represents the color value of (i, j, k)th pixel. In the above equation i, j and k are the variables indexing into three color channels. In the next step the color histogram is normalized to enable appropriate comparison with other histograms in later steps. Finally, the histogram obtained would be a 3D matrix (N N N) where N, as defined before, is the number of bins. This feature represents how "stretched out" a shape is. The perimeter is the number of pixels that are part of an object and have atleast one 4-connected neighbor that is not in the object. For example, circle has the minimum perimeter for a given area, hence exhibits highest compactness.

5. Object Tracking:

The aim of object tracking is to establish a correspondence between objects or object parts in consecutive frames and to extract temporal information. Our approach makes use of the object features such as size, center-of-mass, bounding box, color histogram and compactness, which are extracted in previous steps to establish a matching between objects in consecutive frames. Further our tracking algorithm detects object occlusion and distinguishes object identities after the split of occluded objects. We first associate objects between the previous frame and current frame using centroid matching technique, where Euclidean distance between the centroids of objects is considered. To handle simple object occlusions, we incorporated histogram-based correspondence matching approach in object association. Using this approach, the identity of objects entered into an occlusion could be recognized after a split. . The same approach comes in handy to recover from segmentation errors like "momentary splits", where an object is split into two in a frame and then merge in the next frame. The tracking information, track labels, obtained from the tracking phase is then used by the classification and abnormality detection phase for further processing.

Object Classification:

An efficient method to online categorize moving objects into the pre-defined classes using the eigen-features and the support vector machines. The classification method which uses view dependent visual features of detected objects to train a neural network classifier to recognize four classes: human, human group, vehicle and clutter. The inputs to the neural network are the dispersed ness, area and aspect ratio of the object region and the camera zoom magnification. Like the previous method, classification is performed at each frame

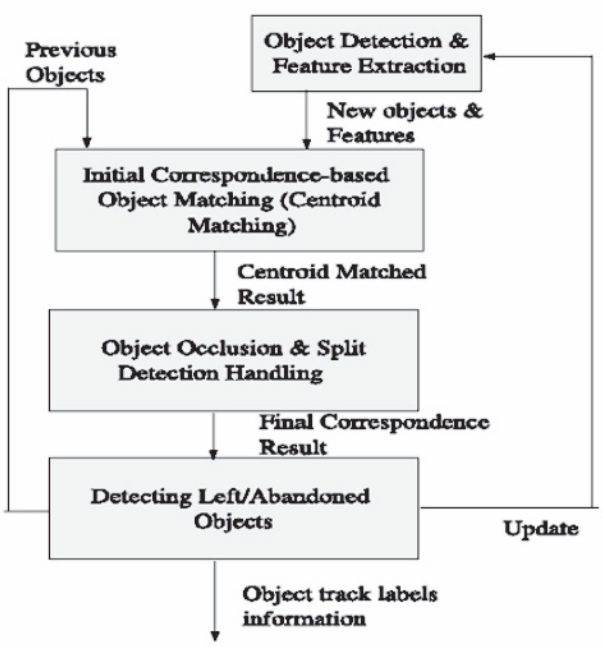

Fig 2. The Object Tracking System Block Diagram

Event Detection:

Our objective is to detect the event 'abandoning of a bag by a person' in video which is a very critical surveillance application. Since we are dealing with bag abandoning event, we just need to focus on object splits. Three kinds of splits are possible as: Bag-Person (possibly an abnormal event), Person-Person (normal event) and Momentary split. The first two cases don't need any explanation. They are the probable cases of bag abandoning event for which classification is performed for the next 25 frames. If the split is of "bag-person" type, then that person is decided as the owner of the bag. The third case is a kind of split which arises due to segmentation error and would be discarded eventually. Such cases arise due to bad segmentation, where an object would split into two objects and in the next frame merge again. Left luggage detection process relies on three critical assumptions about the properties of a bag as: Bags probably don't move, Bags probably appear smaller than people and Bags must have an owner. In detection of abandoned package is done by considering characteristics like lack of motion, minimum distance from nearest person, etc.. An object is classified abandoned, depending on factors like minimum extent, sufficiently high probability of being foreground, no of humans nearby a still object etc…

Alarm Criteria:

constraints depending on which an alarm should be set notifying the operator of a possible abnormal event. We impose two constraints before deciding that a bag has been abandoned.

Space constraint:

we enclose the bag, once it is detected for 25 frames from the time of split, with a circle and the person with an ellipse. We have defined two regions, namely "unattended" region and "abandoned region". Each region is defined by a radius and the bag is enclosed with the circle corresponding to that radius. The decision of whether the person is within a particular region is taken by checking if the ellipse (which encloses the person) and the corresponding circle (which encloses the bag) intersect. One more way of imposing space constraint, which we have not explored, is done by 3D modeling. In [12] the authors have used revised ray-tracing method which converts a given pixel location on screen into a three-dimensional location in the real-world. If ground-truth data is available, then using homography the pixel distance on screen can be converted to the actual distance

Time Constraint:

Once the bag has been detected, we wait for t1 seconds and by that time if the owner is beyond the unattended region then we say that the bag has been unattended. Then we wait for t2 seconds more, and by that time if the person is beyond the abandoned region then we say that the bag has been abandoned. Bag being unattended is like a caution to the operator, of a possible abnormal event. Alarm would be set if the person is beyond the unattended region for a considerable amount of time, which is a pre-defined setting. Classification results using quadratic discriminated analysis were found to be quite accurate. We used a very small training set of 60 cases, where 30 of them belong to one class - bag, and the rest belong to the second class - non-bag(people). Using a test set of 50 cases, gave us an accuracy of 95%. Few examples of the extracted blobs were shown in figs.(5.1,5.2). The classification rule which we used, though simple, gave good results because of the right selection of features as well as the second level confirmation done using centroid variances. Once the objects were classified, time constraint and distance constraint were imposed to detect the abnormal event.

6. Conclusions:

This work presents a surveillance system that supports a human operator by automatically detecting abandoned objects and drawing the operator's attention to such events. It consists of three major parts: foreground segmentation based on Gaussian Mixture Models, a tracker based on blob association and a blob-based object classification system to identify abandoned objects. For foreground segmentation, we assume that video sequences of the background shot under different natural settings are available a priori. The tracker uses a single-camera view and it does not differentiate between people and luggage. The classification is done using the shape of detected objects and temporal tracking results, to successfully categorize objects into bag and non-bag (human). If a potentially abandoned object is detected, the operator is notified and the system provides the appropriate key frames for interpreting the incident.

7. References

1. Zhong Zhang., Andrew S., W Yin., LiYu and Péter L.V"Video Surveillance Using a Multi-Camera Tracking & Fusion System" Multi-Camera Networks, Principles & Applications, 2009, pp 435-456
2. T. D'Orazio and M. Leo "A review of vision-based systems for soccer video analysis" Pattern Recognition, Vol.43/8, Aug 2010, pp2911-2926.
3. Ediz Şaykol., Uğur Güdükbay and Özgür Ulusoy "Scenario-based query processing for video-surveillance archives" Engg. Applications of Artificial Intelligence, Vol. 23, Iss 3, 2010, pp 331-345.
4. Jayavardhana Gubbi., Slaven Marusic and Marimuthu Palaniswami "Smoke detection in video using wavelets and support vector machines" Fire Safety Journal, Vol. 44, Issue 8, Nov 2009, pp 1110-1115
5. B. Uğur Töreyin., Yiğithan Dedeoğlu., Uğur Güdükbay and A. Enis Çetin "Computer vision based method for real-time fire and flame detection" Pattern Recognition Letters, Vol. 27, Issue 1, 2006, pp 49-58.
6. Ioannis Tziakos., Andrea Cavallaro and Li-Qun Xu "Video event segmentation and visualisation in non-linear subspace" Pattern Recognition Letters, Vol. 30, Issue 2, January 2009, pp 123-131.
7. Jacob M. Gryn., Richard P. Wildes and John K. Tsotsos "Detecting motion patterns via direction maps with application to surveillance" Computer Vision and Image Understanding, Vol. 113/ 2, 2009, pp 291-307
8. Sergio A. Velastin., Benny Lo and Jie Sun "A flexible communications protocol for a distributed surveillance system" J of Network & Computer Applications, Vol. 27/ 4, 2004, pp 221-253
9. Yun Zhai., Rogerio Feris., Arun H., Sharath P. "Composite Event Detection in Multi-Camera & Multi-Sensor Surveillance Networks" Multi-Camera Networks, Principles & App 2009, pp 457-480
10. Tao Xiang and Shaogang Gong "Activity based surveillance video content modeling" Pattern Recognition, Vol. 41, Issue 7, July 2008, pp 2309-2326.
11. Aishy Amer., Eric D & Amar M "Rule- based real-time detection of context-independent events in video shots" Real-Time Imaging, Vol. 11/3, 2005, pp 244-256.
12. "A real-time algorithm for error recovery in remote video-based surveillance applications" Signal Processing: Image Comm, Vol.17/2, 2002, pp 165-186.