A
DISSERTATION
ON

"Detection of Suspicious Activity in Video Surveillance"

Submitted under the partial fulfillment of the requirement for the award of the degree of

MASTER OF TECHNOLOGY
in
Software Engineering

Submitted by:
ANURAG AGARWAL
Roll no.: 03/SE/09
Registration no.: 05/MT/SE/FT

Under the guidance of:
MR. SHAILENDER KUMAR
Asst. Professor,
Deptt. of Computer Engineering
Delhi Technological University
Bawana Road, Delhi – 110042

DEPARTMENT OF COMPUTER ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
DELHI, 2011

CERTIFICATE

DELHI TECHNOLOGICAL UNIVERSITY
BAWANA ROAD, DELHI – 110042

This is to certify that the work contained in this dissertation entitled **"Detection of Suspicious Activity in Video Surveillance"** submitted by **Anurag Agarwal,** Roll no. 03/SE/09 in partial fulfillment of the requirement for the award of the degree of Master of Technology in software Engineering at Delhi Technological, Delhi is a record of the candidate's own work carried out by him under my guidance and supervision in the academic year 2010-2011.

Mr. Shailender Kumar
Asst. Professor
Project Guide
Delhi Technological University
Delhi

ACKNOWLEDGEMENT

It is a great pleasure to have the opportunity to extend my heartiest felt gratitude to everybody who helped me throughout the course of this project.

I take this opportunity to express my deep sense of gratitude and indebtedness to my learned supervisor **Mr. Shailender Kumar, Asst. Professor, Dept. of COE, DTU,** for his invaluable guidance, encouragement and patient reviews. With his continuous inspiration only, it has become possible to complete this dissertation. He helped in pointing out places in several drafts of the thesis where clarity could be improved and claims made more precise.

I also extend my gratitude towards **Dr. Daya Gupta, Head, Dept. of COE, DTU** who has always been cooperative throughout the whole coursework and gave us valuable inputs. I am also thankful to all other faculty members and staff of the Department of Computer Engineering at Delhi Technological University for sharing their knowledge and experiences with me as well as for their kind support.

I also like to thank my batch mates at Delhi Technological University for sharing their ideas and opinions on several topics that were important for my work. I also owe gratitude towards my parents for their patience and support. They have been always around to cheer me up in the odd times of this work.

Anurag Agarwal

ABSTRACT

The video surveillance systems have gained popularity since last few decades because of their use in the detection of unusual activities, surveillance, patrolling, and other scientific and engineering problems. Activity detection is an important component of video surveillance and involves tasks like recognition of humans, their activities with respect to their surroundings and the further analysis for any abnormality or suspicious behavior.

This recognition can be done either manually or automatically with the help of computers. Though it is very easy for a human to analyze the video for suspicious activities, and this is the way which is in widespread use, the other way can be to do it automatically. Autonomous video surveillance requires automatic processing of video sequences.

This work, therefore, proposes the approach to do the surveillance automatically. The detailed approach along with its advantages over other approaches has been discussed at length. The various constraints that have been taken into account are also elaborated. The design of the system takes input from the video frames taken at the place where we provide surveillance. The system does both the low-level processing, like motion detection and tracking, and also performs high level decision making jobs like unusual activity detection. This work, therefore, aims to translate the low-level input into a high-level semantically meaningful activity description. The three major components of the work include moving object detection, tracking and unusual activity detection.

The approach in this dissertation is substantiated by taking two unusual activities, first is abandoning of bag by a person and the second is carrying of bag by a person. Only a single person is involved and outdoor background and static background is taken. The analysis is made on offline videos and no real-time detection or analysis is made. The development is done in C++ using OpenCV library on Linux platform.

List of Figures

List of Tables

Table of Contents

Chapter 1

Introduction and Problem Statement

1.1 Introduction

The job of video surveillance system is to analyze video sequences to detect unusual or abnormal activities. Activity detection a very crucial component of video surveillance systems for activity based analysis of surveillance videos. Detection of human activities uses computer vision techniques on video sequences to detect what a human is doing in his surrounding environment. It is difficult to obtain activity information both quickly and accurately. Activity detection has great importance in many applications, particularly in the surveillance industry. Human activity detection is one of the complex tasks that human brain does effortlessly, but many difficulties arise when a computer system attempts to process the activity. The vast amounts of data in the video sequences often make it difficult to make decisions for a computer system [7].

Recognition of human activities in video surveillance can be manual or automatic [11]. In manual video surveillance system, a human analyses the video content. Such types of systems are currently in widespread use. Autonomous video surveillance requires automatic processing of video sequences. The systems that perform simple motion detection are typical examples for such types. The system takes input from the video frames taken at the place where we provide surveillance. The system does both the low-level processing, like motion detection and tracking, and also performs high level decision making jobs like unusual activity detection [6].

Humans perceive the video events as the high-level semantic concepts, when he observes the video sequence. But this is not the case with computer surveillance. The major challenge in video surveillance with computers is to translate the low-level input into a high-level semantically meaningful activity description [1]. Video event recognition attempts to fix the problem of reconciling this perception of video events with a computer surveillance system.

The present video surveillance systems mostly in use depend on human operators to analyze the content of video for any unusual activity. This method is not beneficial when the amount of data to analyze is large. Generally, analysis is done in this case only when some mishappening occurs. But the automatic approach to analyze and detect suspicious behavior will help to quickly and efficiently detect any such abnormal activity and may even provide warning before the occurrence of any big casualty.

Such video surveillance systems require reliable, fast and robust algorithms for detection of moving object, tracking and analysis of unusual activity [11]. This will be a lot more help to human operators whose job will be very simplified and they will just need to press the panic button in case of any emergency. The human operators will not need to go through every video frame for analysis. Furthermore, the reaction time is reduced significantly.

The basic approach to automatic video surveillance involves three steps, detecting moving object, tracking and identifying of unusual activity. The first step of detecting moving object deals with segmentation of moving objects from stationary background. Temporal differencing, background subtraction, statistical methods, and optical flow are the commonly used techniques for object detection. Segmentation of object is difficult and involves significant problem because of dynamic environmental conditions such as illumination changes, waving tree branches in the wind and shadowing. So it needs to be a well robust and fast video surveillance system [13, 14].

Tracking is the next step in the video analysis, which can be simply defined as the temporal correspondence conception among detected moving objects from frame to frame. This procedure identifies temporal recognition of the segmented objects and generates cohesive information about the segmented objects in the surveillance area. The tracking step output is generally used to enhance and support object motion segmentation, features extraction of object and higher level analysis of unusual activity [16]. The final step is to recognize the unusual activity in a video. These algorithms output can be used for assisting the human operator with high level semantic data and this output can help him to make the more accurate decisions.

The final and the most prominent step in this system is to understand unusual activities in a video scene. It is a domain with scope for extensive research and has many promising applications. Thus, it attracts the attention of several researchers, commercial companies and institutions.

The role of visual surveillance systems is very crucial in the circumstances where continuous patrolling is not possible by human guards like in nuclear reactors, international border patrolling, etc [7]. Requirement for video surveillance systems in public has application areas like shopping complexes, monitoring of parking lots, and banking or financial establishments. This brings arise the requirement of understanding the human activities and to make computer vision system able to construct a higher level semantic knowledge of the consequence appearing in a video scene [8]. Some scenarios are given below that might be handled by video surveillance systems [17].

Public and Commercial Security:

1. Monitoring of banks, airports, museums, departmental stores, stations, parking lots and private properties for crime prevention and detection.
2. Patrolling of highways for accident detection.
3. Access control.
4. Surveillance of forests and properties for fire detection.

Smart Video Data Mining:

1. Extracting statistics from sport activities.
2. Compiling consumer demographics in amusement parks and shopping centers.
3. Logging routine maintenance tasks at industrial and nuclear facilities.
4. Counting endangered species.

Military Security:

1. Patrolling of national borders.
2. Monitoring peace treaties.

The utilization of object detection, object tracking and activity detection algorithms are not restricted to video surveillance systems only. Other application domains also get benefits from the advanced research on these algorithms. Some areas are virtual

reality, human machine interface, video compression, video editing and multimedia databases and augmented reality [9, 11].

Thus, we can visualize how important and useful this automated surveillance system can be at personal, commercial and business level. The benefits can be far and wide and can have major implications on how we manage our security and surveillance systems.

1.2 Problem Definition and Scope

Understanding unusual activities in a video scene is a challenging scientific problem. Some unusual activities are specified in previous section. In this dissertation we define two unusual activities, first is abandoned the bag by a person and second one is carried the bag by a person. Only one person is involved in the unusual activity in our scenario and we take static background in the outdoor videos. We use OpenCV library in C++ language on the Linux platform.

This dissertation presents a video surveillance system in which analysis is done on offline videos. The approach used in this work uses three components, viz., detecting moving objects, tracking those objects, and then finally to detect the unusual activities. We should note that this is not real-time analysis and only offline videos can be analyzed using this system. In the system that we are going to present, adaptive background subtraction models are used for the detection of moving objects [13]. In background subtraction, each pixel comprises a Gaussian mixture and an online approximation is used for further updating the model. Based on the variance and mean of each of the Gaussian in the mixture, the Gaussians which correspond to background are determined [12]. On observing, the pixels whose values do not match the background distributions are considered as foreground until there is a Gaussian that concludes them with consistent and sufficient evidence supporting them as comprising background pixels [14]. This background method consists of two significant parameters α, which is the learning constant and ρ, which denotes the proportion of data that should be maintained for the background [15]. A foreground weight parameter is added for each single Gaussian model, and this parameter is used with background weight parameter to construct the energy function. A relatively stationary background

is assumed and adaptive threshold for each pixel is used assuming that noise at each pixel is time varying [1].

After segmentation pixels of the moving object from the static background scene, the tracking algorithm is used to track the detected moving objects in successive video frames with the correspondence based matching scheme. It also handles the occlusion cases in which some object might be occluded by some other object [16]. It uses 2-Dimensional object features such as centroid of the object, its size and position to match corresponding objects frame by frame. The tracking algorithm of the object does not distinguish between objects that mean the algorithm deals equally with both person and nonperson, like a human or bag [1]. The stationary object detection is performed by recognizing the trajectory of each blob and analyzing it.

The final stage of this system is the detection of the unusual activity which is the abandoning of bag or carrying of it. The process first searches for the abandoned or carried bag objects, measuring the likelihood to find the bag [5, 40]. Finally, once the bag and person has been detected, it checks for the unusual activity. The activity recognition algorithm incorporated here is Bayesian framework analysis [6].

1.3 Motivation

Our motivation is to present a surveillance system with detection of moving object, tracking, and activity detection capabilities. This surveillance system will be helpful in surveillance as well as other areas as has been mentioned in the introduction section. So it will be a great help for the people who now have to analyze all the video frames for surveillance. They will now just have to report the suspicious events as and when they are identified through this system. The only thing we should notice is that this system works on pre-stored offline videos and is not able to analyze video frames feeded to it in real-time. So removing this constraint can be one of the future step of this work.

1.4 Organization of the Thesis

The organization of the rest of this thesis is as follows. In chapter 2, we present a brief literature survey in background subtraction, tracking and detection of unusual activity for video surveillance applications. We explore our methods for moving object detection with background subtraction, object tracking and the unusual activity detection in chapter 3. Experimental results and summary are supplemented in chapter 4. Conclusion and future work are discussed in chapter 5 and references are given in chapter 6.

1.5 Summary

In this introductory chapter, we have discussed what video surveillance system is and why we need it. We also discussed why the automated approach of surveillance is at par with the manual analysis of videos for the same purpose. We then discussed a broad outline of our approach and the various components that are part of this approach. Then we elaborated on the scope of this work and the various constraints on this system. We also mention the motivation for pursuing this dissertation and also the various application areas where this work finds abundant use. The organization of this dissertation is discussed in the end.

Chapter 2

Literature Review

A number of literature surveys have been done about object detection, classification, tracking and activity analysis in the video surveillance. We present the survey, which deals only those works that are related to the same context as our thesis study. However, for comprehensive study about computer vision, we also present brief information about some techniques and approaches which are used in similar tasks that are not used in our study.

A generic video surveillance framework is shown in Fig. 2.1 [8, 19, 21]. Although, some process steps demand exchange of information with some other steps, this framework causes a good structure for the discussion.

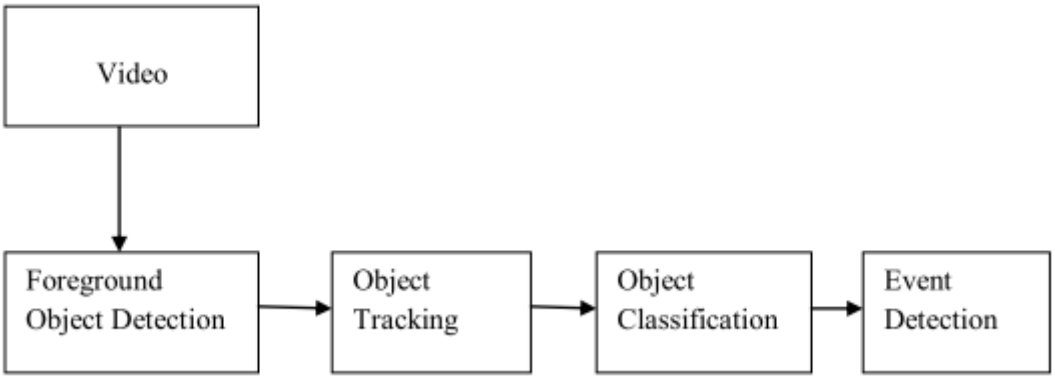


Figure 2.1: Generic Video Surveillance Framework

2.1 Background Subtraction

An application has different needs that are related to video processing, thus requiring different approaches. However moving objects are present in every application. Thus, detecting the moving regions, such as vehicles and people, is the first basic step of every computer vision system. Moving objects are the focus of attention and their analysis is required in subsequent steps [11]. The reliable process for detecting moving objects is difficult due to sudden illumination, repetitive motions, cluttering

and occlusion [14]. Temporal differencing, optical flow, and statistical methods are the frequently used techniques for foreground background segmentation [23]. We elaborate on these techniques below.

2.1.1 Temporal Differencing

Temporal differencing uses the difference of pixel to pixel in consecutive frames in order to detect the moving object region [23]. This method is highly sensitive to dynamic scene changes. It generally fails to detect all the pixels relevant to an object in dynamic conditions [1]. This method also fails in detecting the stopped objects in a scene. A two frame differencing approach is discussed in [15] where pixels at a location are marked as foreground pixels if equation (2.1) is satisfied.

$$|I_n(x,y) - I_{n-1}(x,y)| > \zeta \tag{2.1}$$

Where ζ is the predefined threshold, I_t the current image and I_{t-1} is the previous image. If the difference of pixels is above the threshold value then those pixels are classified as foreground. Three frame differencing techniques can be used to overcome the shortcoming of two frame differencing [12].

2.1.2 Optical Flow

Optical flow methods use the flow vectors of movement of objects over time to detect moving object in a frame [7]. In optical flow most methods assume that color or intensity of a pixel is invariant along the displacement from one frame to another [11]. Optical flow provides a description of both the moving regions and the velocity of moving object. Computation of optical flow is complex due to noise and illumination changes and cannot be used without using specialized hardware for real time system [23].

2.1.3 Statistical Methods

Statistical methods are more advanced methods that use the statistical characteristics of individual pixels to overcome the shortcomings of basic methods of background subtraction [7]. These statistical methods keep and dynamically update the statistics of pixels, which belong to background scene. The statistics of each pixel is compared to

statistics of the background model and the pixels that are different to background model are subsequently identified. In case of illumination changes, shadows and in scenes that contain noise, this approach is more reliable [23].

In statistical background method, each pixel is modeled with its minimum (M_m) and maximum (M_n) intensity values and observe maximum intensity difference (D) between any consecutive frames during initial training time where frame contains no moving objects [22]. In the current frame a pixel is identified as foreground if it satisfies the equation (2.2).

$$|M_m(x,y) - I_k(x,y)| > D(x,y)$$
$$\text{Or} \qquad (2.2)$$
$$|M_m(x,y) - I_k(x,y)| > D(x,y)$$

After detecting the foreground pixels, some post processing morphological operations such as dilation, erosion, and closing are used to enhance the detected regions and reduce the effects of noise. Also, connected component labeling is applied to eliminate the small-sized regions [12]. The statistics of the background pixels are updated with new image data which do not belong to the moving regions of current image.

An adaptive background mixture model for background subtraction is another example of statistical methods that was discussed in [13]. In their description, each pixel is separately modeled by a Gaussian mixture which are updated online by incoming pixel data. In order to detect a foreground or background pixel, evaluate the Gaussian distributions of mixture model for that pixel [13]. We describe this model in detail as we used it in our system.

2.1.4 Shadow and Light Change Detection

The above algorithms for background subtraction have been used for video surveillance and perform well in indoor and outdoor environments. However, most of these algorithms are suspicious to both global illumination changes like sun being covered by clouds and local like shadows and highlights [15]. Motion detection methods fail as they consider shadows as foreground in foreground segmentation causing higher levels such as object tracking to perform inaccurately [14]. In the

literature, the proposed methods use either chromaticity or stereo information to remove shadows and handle with sudden light changes [13].

A motion detection and shadow detection method is discussed in [15]. In this paper every pixel is classified by the color model that discriminates brightness from the chromaticity component. Distortion of brightness and chromatic information between the background and current image pixels is used to classify the pixel into different categories (e.g. background, shadow, moving foreground object and highlighted background). T. Horprasert et al. in [37] described the approach that uses gradient information and chromatic information to handle shadows. They used the observation that an area that comes into shadow gives results that have significant changes in intensity rather than much change in chromaticity.

Two heuristics are used in literature for shadow detection scheme:

(a) Change in reduction rate of intensity reduces smoothly between neighboring pixels,

(b) Intensity values of shadow region pixels are as compared to the background pixels [12].

2.2 Object Tracking

Tracking is a difficult and significant problem that comes into interest among researchers of computer vision. The objective of tracking is to establish the correspondence of objects in the consecutive frames of video [24]. Tracking is the significant task for most of the video surveillance systems since it provides cohesive temporal information about moving regions which are used to enhance lower level processing results such as motion segmentation and also used to enable higher level data processing such as activity recognition [16].

In congested situations tracking has been a difficult process to apply due to inaccurate objects segmentation. Occlusion of objects, shadows, and stationary items in the scene are the common problems for erroneous segmentation. Thus, coping with occlusions and dealing with shadows at motion detection is important for robust tracking at segmentation level and tracking level [18].

Object tracking in video scene can be categorized according to the applications requirements. There are two common approaches for tracking object as a whole [17], one is based on position prediction or motion estimation and the other one uses correspondence matching [21]. The methods that track the human body parts employ model based approach to locate and track the body parts.

Number of views is also considered in tracking; there are single-view and multiple-view tracking [24]. Tracking can also be grouped according to criteria such as dimension of tracking space, tracking environment, the sensor's multiplicity (monocular vs. stereo), and the camera's state (moving vs. stationary), etc. Different tracking methods are summarized as follows.

2.2.1 Model-Based Tracking

The geometric structure of human can be detected as stick figure, 2-Dimention contour or volumetric model [11]. We describe each of these in detail below.

2.2.1.1 Stick Figure

Human motion is represented by the movements of the limbs, torso, and head, so the stick-figure representation uses the human body as the combination of line segments that are linked by joints [23]. The stick figure is analyzed in various ways, e.g., distance transforms or by means of median axis transform.

The motion of joints provides the way to estimate and recognize the whole figure. Meghna Singh et al. [36] represented structure of human body in the silhouette through a stick figure, which articulates ten sticks with six joints. In addition, angle constraints and prediction of each joint were added to reduce the complexity of matching process. This kind of representation of the human body is also used by Feng Niu et al. [9] to build a hierarchical model of human motions using Hidden Markov Models which recognizes view-independent tracking in monocular video sequences.

2.2.1.2 2-D Contour

This type of human body representation is directly relevant to the projection of human body in the image plane. In such representation, human body segments are correspondent to 2-D ribbons [21].

A cardboard people model has been proposed in [17]. In this method, a set of connected planar patches limbs of human body. The parameterized object motion of these patches was used for the analysis of articulated motion of the human limbs.

In Nizar Zarka et al. [18] work, the subject's outline was figured as the edge regions represented by 2-D ribbons and these were U-shaped edge segments. It is easy to extract a silhouette or contour from the image. Robert Bodor et.al [19] used the spatial-temporal pattern based upon 2-D contour representation in XYT space to track and analyze the walking figures. They first recognized the characteristic pattern represented by the lower limbs of a human while walking, and then located the projection of head movements in the spatio-temporal domain, followed by the other joint trajectories [19].

2.2.1.3 Volumetric Models

2-D models have some disadvantages because of its restriction to the angle of camera. So many researchers are trying to find the geometric structure of human in more details with the help of 3-D models such as spheres, elliptical cylinders, and cones, etc [16]. 3-D volumetric models are more complex, they require more parameters in order to expect the better results and during the matching process 3-D models lead to more expensive computation [18].

Tao Gao et al. [21] used the correspondence between 3-D body model of elliptical cones and real image sequence. Based on iterative Kalman filtering, the information of both edge and region is used to determine the orientations to the camera and degrees of freedom of joints [11]. Kalman filter is a state estimation model based on Gaussian distribution. It is restricted to conditions where probability distribution of state parameters is unimodal. It is inadequate in dealing with multi-modal distributions in the presence of cluttered background, occlusion, resembling the tracked objects, etc.

2.2.2 Region-Based Tracking

In region-based tracking the approach is to identify a connected object moving region. Today it is being used widely. This approach explains the use of blob features to track the human [4]. In this approach, a human body is considered as a combination of

blobs describing various body parts such as limbs, torso, and head. Then both human body and background image are modeled with Gaussian distributions. Javier Varona et al. [3] proposed a background subtraction method that combined gradient information and color to effectively handle shadows in segmentation of a moving object. Then tracking process is performed at various abstraction levels: regions, and people, etc. Each region has a bounding box that can merge and split.

The region-based tracking method runs reasonably well. However, in some situations difficulties arise. In the case of shadows, it may result in merging with blobs associated with people [4]. Shadows may be removed with the help of the fact that pixels of shadow regions tend to have a lesser extent of texture. Congested situations is the another problem for video tracking [18, 19]. People, under these conditions, partially occlude each another instead of being separated from each other. So task of segmenting an individual human becomes difficult. The solution to this problem requires multiple camera tracking system.

2.2.3 Active Contour Based Tracking

Active contour models based tracking directly extracts the shape of objects. The idea is to represent the bounding contour of the objects and dynamically update it over time. Liang Wang et al. [22] discussed a variation framework for detecting and tracking moving objects in a video. In a statistical framework, the observed frame difference density function was estimated using a mixture model and it was used to produce the initial motion detection boundary. Then detection and tracking problems were recognized in a common framework that applied an active contour objective function [21]. Complex curves could be detected and tracked using the level set formulation scheme, while topological changes for evolving curves were naturally dealt.

The advantage in the region-based tracking approach is to have an active contour based representation to reduce the computational complexity but it needs a good initial fit [7, 22]. In the presence of partial occlusion one could keep tracking if somehow one could initialize an individual contour for each moving object region. But it is quite difficult to initialize, especially in the case of complex objects.

2.2.4 Feature-Based Tracking

In feature-based tracking, distinguishable points or lines on the objects are used as sub-features to realize the tracking task. The advantage is that some of the sub-features of tracked object remain visible even in the case of partial occlusion. Feature extraction and feature matching are included in Feature-based tracking approach [22]. It is easier to extract the low-level features such as points but higher-level features such as blobs and lines are relatively difficult to track. So, there is usually a trade-off between tracking efficiency and feature complexity.

Jiang Dan and Yu Yuan used the point-feature tracking in their work [38]. They selected the center of mass as feature point of a person for tracking, who was bounded by a rectangular box. Even if occlusion happened between two objects during tracking, as long as velocity of center of mass could be estimated effectively, tracking was successful.

The use of multiple cameras is one of the tracking aspects and has been actively researched. Multi camera tracking is very useful for improving results by reducing handling occlusions, ambiguity, etc. A multivariate Gaussian model uses to match the human objects in consecutive frames taken by cameras at various locations, and also discusses the automatic switching between neighboring cameras [6]. For multiple cameras based tracking systems, one need to decide which camera is being used at which time instant. For a successful multi-cameras tracking system, it is a crucial problem how to handle the selection and data fusion between cameras [22].

2.3 Activity Recognition

After successfully tracking the moving objects from one frame to another in a video, the problem of recognizing an event from image sequences follows naturally. Activity recognition involves action recognition and description [7]. Activity recognition can guide the development of many human motion analysis systems. It is the most important area of future research in motion analysis.

Activity recognition is to analyze the human motion patterns, and give high level description of actions. It may be viewed as classification problem of time varying

data, i.e., matching an unknown sequence with labeled reference sequences to represent an event [32, 41]. The basic problem of activity detection is how to learn the reference action sequences, and how to effectively interpret events. The activity recognition algorithm assumes that the shape of each type of object is known [5]. The basic types of objects include human, vehicle and carried objects. This information is either provided by the detection or tracking methods or specified by system users [1]. All these are the hard problems and have received attention from researchers.

2.3.1 Dynamic Time Warping

In Dynamic time warping technique, a non-linear warping function is computed that aligns two variable length time sequences [22]. To finds the similarity between two time series the warping function can be used. DTW is the template based dynamic programming matching technique, used widely for speech recognition. It has the benefit of conceptual simplicity, and used in the patterns matching of human movement. But this approach uses the techniques that are specific to a certain application domain. So applying these techniques to other areas raises difficulties [11].

2.3.2 Finite State Machines

Finite State Machines (FSM) [1], or Finite State Automata, is formalism useful for representing the temporal aspects of video events. A state transition diagram is used with start and accepts states for recognition of processes. Finite State Machines are deterministic models and produce computationally efficient solution to analyze the occurrences of an event. The FSM model analyzes the sequential aspects of video events, and it is a simple model that learns from training data [24].

In FSMs model, the single-thread events are formed by a sequence of states. FSM event models are utilized in event domains and include aerial surveillance, hand gestures [11], and single actor behavior. The inherent ability of FSM formalism is to capture sequences that allow it to be related with different abstractions including object-based abstraction and pixel-based [24]. FSMs are an important tool in event understanding because of their easiness, pedagogy and ability to model temporal sequence. Extensions of FSM have been proposed for capturing the hierarchical

properties of video event [23]. The probabilities into the FSM framework have been introduced to address the uncertainty in video events. It should be noted that in the area of the event understanding the terms “probabilistic FSMs” and “HMMs” are interchangeably used. The main distinction is that HMMs assume a hidden state variable while FSMs assume a fully observable state [9].

2.3.3 Bayesian Networks

In order to deal with uncertainty of observations and recognition of video events, Bayesian Networks event models have been proposed to utilize the probability as a mechanism for dealing with uncertainty [5, 36]. Bayesian Networks (BN) (also known as independence diagrams, Bayesian Belief networks, or probabilistic networks) are a class of directed acyclic graph models. Nodes in Bayesian Networks indicate the random variables which may be continuous (described by parametric distribution) or discrete (finite set of states) [20]. Structure of the graph is used to represent the conditional independence between these variables. The structure of Bayesian Networks allows use of the joint probability over all variables with few parameters, and using the notion of conditional independence [3].

The joint probability causes known values to be used by any node in the Bayesian Network. Often Bayesian network event models represent the event as a hidden or unknown variable and the observations as known variables [5]. The parameters (conditional and prior probabilities) and structure (nodes and arcs) of Bayesian Network are used to represent the distribution of unknown variables given the values of known variables [36].

Bayesian Networks do not have an inherent capacity for classifying temporal composition of the video events. Choosing abstraction schemes and single frame classification are the solutions to this problem. Bayesian Networks have been used to recognize events such as indoor surveillance and aerial surveillance. More complex Bayesian Networks have been used to recognize events such as American football plays and parking lot surveillance [33].

Bayesian network is a graphical model that handles complex conditional dependencies on the set of random variables that are modeled as conditional

probability densities [11]. Bayesian network has also been used to recognize activities using the contextual information of the involved objects. Bayesian networks are more general than HMMs by considering conditional dependencies between random variables; the temporal model is used as Markovian in the case of HMMs [30].

2.3.4 Hidden Markov Model

Hidden Markov Models are a class of directed graph models extended to the temporal evolution of the state. One variable represents the hidden state and other variable represents the observation state with a single time slice [10]. Evolution of the process is estimated by time slices described by the model over time. This structure represents a model where observations are dependant only on current state and current state is only dependent on the state at the previous time slice e.g. Markov assumption [30].

HMMs have two stages, one is training and the other is classification. In the training stage, number of states of HMM must be specified and corresponding states transformation and outcome probabilities should be optimized in order to generate the symbols that correspond to the observed image features. In the classification stage, the probability to generate the test symbol sequence by a particular HMM is computed, that is, corresponding to the observed image features [9]. HMMs are better than DTW in processing unsegmented data, and therefore, extensively applied to the matching of motion patterns.

A number of works in literature using this approach are in the event domains of single person actions (e.g. “jumping ", " walking ", etc) [22], sign language and gesture recognition, and tennis stroke recognition. The events recognized by this approach are generally of few seconds in length. These approaches are generally dependant on acceptable segmentation of the video sequences into event clips. That is, before classifying the event into a given video sequence, a clip is given that is known to contain an event (only one event).

2.3.5 Conditional Random Fields

One drawback of HMMs in particular, is their dependence on availability of a prior observation and this prior observation is not always known so it is frequently estimated using assumptions that will need efficient computation, such as dependence

or independence between the observations, given the state [11]. This is often an invalid assumption in the domain of video events. The conditional distribution is modeled effectively in a discriminative statistical framework and there is no requirement for such restrictive assumptions.

Conditional Random Fields are undirected graphical models, which generalize the HMM by putting the feature functions corresponding to the global observation in place of the transition probabilities [23]. These functions may be arbitrarily set in any number. Existing known problems for HMMs of observation and evaluation can be extended to CRFs. CRF parameters can be learned using convex optimization methods (e.g. conjugate gradient descent).

In event modeling, for similar event recognition tasks CRFs have been shown better performance than HMMs. This has the ability to choose arbitrarily dependent abstraction scheme. Furthermore, in CRFs, unlike in HMMs, abstraction features based selections are not only limited to the current observation but also consider other combinations of past and future observations [11]. CRF models have a major disadvantage of their parameter learning time in comparison to HMMs. CRFs are recently popular event models that can straightforward apply to those cases where HMMs had been applied before and CRFs achieve better event recognition results. The tradeoff in this approach is a significantly longer training time [12].

2.4 Summary

In this chapter, the literature work in background subtraction, object tracking, and activity recognition has been discussed. Regarding background subtraction, we discussed about the frequently used techniques - temporal differencing, optical flow, and statistical methods. In tracking, we discussed model based tracking, region based tracking, active contour based tracking and features based tracking. Finally we end the chapter with activity recognition, where we discussed dynamic time wrapping, finite state machines, Bayesian networks, Hidden Markov Models and conditional random fields.

Chapter 3

Proposed Approach

The overview of object detection, object features extraction, tracking and activity detection system is shown in figure 3.1. The proposed approach of whole system makes use of the observation discussed in [23, 41]. This system is able to distinguish moving and stopped foreground objects from static background scene, track the objects and detect the unusual activity. In this chapter we describe the computational models applied in this system to achieve the goals specified above.

The computational complexity and the constant factors of the algorithms are important for video surveillance system. The selected algorithms for various problems in computer vision are affected by computational run time performance and their quality. Furthermore, our system uses the stationary camera. We initialize the system by giving the video imagery from a static camera where surveillance is provided. Methods are able to work on color video imagery.

The first step is to separate foreground objects from stationary background. We use an adaptive background subtraction method and post-processing methods to make a foreground pixel representation at every frame. We then do the grouping of connected regions in the foreground pixel map and object features such as bounding box and center of mass are calculated [35].

Tracking is next step after background subtraction. An object level tracking algorithm is used in our video surveillance system. We don't track the object parts such as limbs of human, but track the object as a whole from frame to frame [24].

Final step is the unusual activity (abandoned/carried object such as bag) detection. This system uses a single camera view and unusual activity is detected using the background subtraction and object tracking result.

3.1 Background Subtraction

Detecting foreground objects from stationary background scene is both a difficult and significant research problem. The first step is to detect the foreground objects for almost all the visual surveillance systems. It creates a focus of attention for later processing steps such as tracking and activity detection and reduces computational time since only pixels need to be dealt that belong to foreground objects [26].

Dynamic scene changes such as light reflectance, shadows, sudden illumination variations, and camera noise make reliable object detection difficult. Hence, object detection step need necessary attention to make robust, fast, and reliable visual surveillance system.

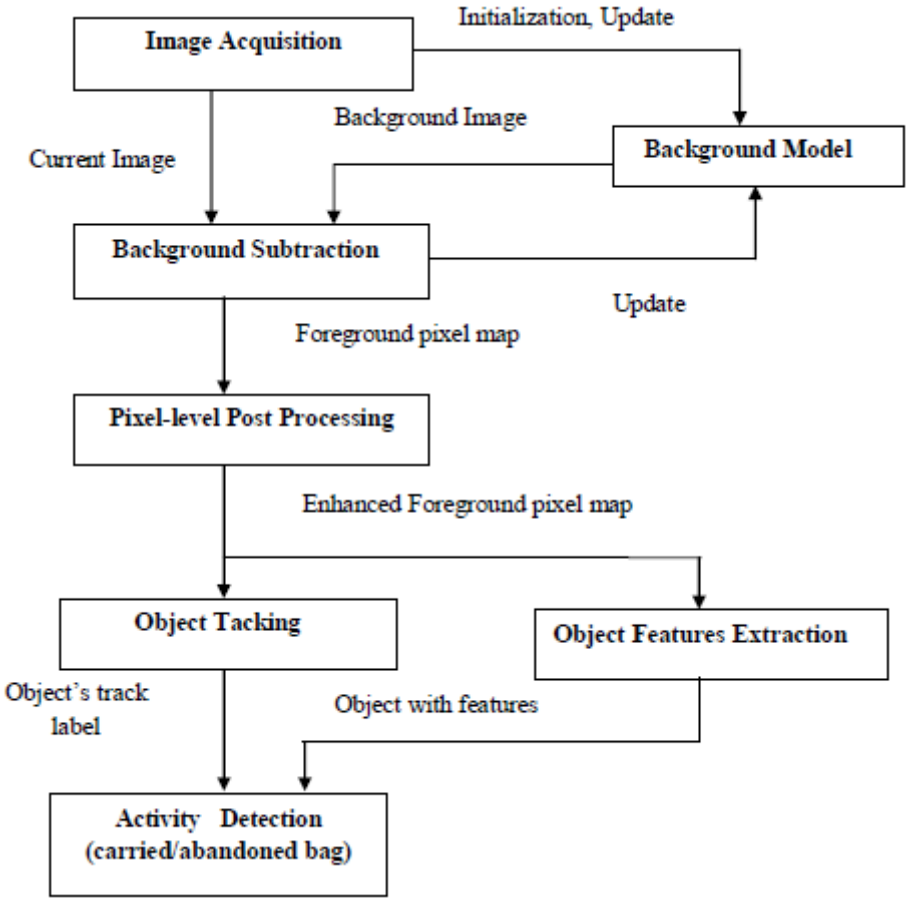


Figure 3.1: System Block Diagram [23, 41]

Our method depends on a three stage process to extract foreground objects from the video imagery [29]. The first step is to initialize the background scene. There are

various techniques in the literature that are used to model the background scene. In order to evaluate the quality and compare run-time performance of different background scene models for object detection, we compared temporal differencing, OpenCV Gaussian mixture model and our Gaussian mixture model. The foreground detection related parts of the system is compared and our Gaussian mixture model is combined with other modules to let the whole detection system work flexibly.

Next step in the Background subtraction method is to update the foreground object pixels by using the background model and current image from the video [27, 30]. This process depends on the background model in use and used it to update the background model to obviate to dynamic scene changes. The detected foreground pixel contains noise due to environmental effects or camera noise. To remove the noise in the foreground pixels perform the pixel-level post-processing operations.

Once we get the foreground pixels, connected component algorithm is used to find the connected regions and objects bounding rectangles are calculated in the next step. Due to defects in foreground segmentation process the labeled regions may be disjoint [33]. Hence, it is experimentally required to be effective to merge those isolated regions. Also, due to environmental noise some relatively small regions are eliminated in the pixel-level post-processing step. Area and the center of mass of the regions corresponding to objects are the extracted object features from current video image by using the foreground pixel map.

We use a combination of background subtraction model and pixel level post-processing methods to create a foreground pixel distribution map and extract object features in every video frame. Initialization and update are the two distinct stages of background models process [28]. In following sections, the initialization and update mechanisms of foreground region detection methods are described which are tested on our system. The comparison of computational run-time and qualities of these models for detecting foreground objects are given in section.

3.1.1 Adaptive Gaussian Mixture Model

Gaussian mixture model can robustly deal with slowly moving objects, lighting changes, clutter, and removing or introducing objects from the scene. The previous

model was unimodal background model that could not handle light change, image acquisition noise, and multiple surfaces noise at the same time. But Gaussian mixture model uses the mixture of probability distribution to represent each pixel in the model. GMM has these promising features, so we implemented and integrated GMM model in our visual surveillance system [26].

The basic idea is to define a detected region and segment the pixels of interest. It is necessary that color attribute of each pixel is modeled through an adaptive mixture of Gaussian distributions of an image sequence [32]. For each new captured observation, the mixture of Gaussian distribution model is updated and reduces the influence of past observations and allowing the model adaptation corresponding to a gradual variation of illumination. The Gaussian distributions model represents both foreground and background. It is necessary to describe the distribution of pixels subset to represent the background model. At each observation, the subset definition updates according to the associated mean and weight of every distribution representing the frequency that distribution better modeled the pixel.

In GMM the values on observing each pixel (*e.g.* vectors for color values and scalars for gray values) over time is modeled as a pixel procedure and the recent history of individual pixel $\{X_1, \ldots, X_t\}$ comprises the mixture of K Gaussian distributions [31]. The probability of finding the current background pixel value is computed using equation (3.1) [12].

$$P\left(\vec{X}_{j,t}\middle|\vec{X}_{j,1}, \ldots, \vec{X}_{j,t-1}\right) = \sum_{j=1}^{K} \omega_{j,t} \times \eta(\vec{X}_{j,t}, \vec{\mu}_{j,t}, \Sigma_{j,t}) \tag{3.1}$$

Where K is the number of Gaussian distribution, $\omega_{j,t}$ is an estimation of the weight of the j_{th} Gaussian of the mixture at time, $\mu_{j,t}$ is the mean value and $\Sigma_{j,t}$ is the corresponding covariance matrix and η is a Gaussian probability density function that is computed in equation (3.2) given in [13].

$$\eta\left(\vec{X}_{j,t}, \vec{\mu}_{j,t}, \Sigma_{j,t}\right) = \frac{1}{2\pi^{n/2}\left|\Sigma_{j,t}\right|^{1/2}} e^{-\frac{1}{2}(\vec{X}_{j,t}-\vec{\mu}_{j,t})^T \Sigma_{j,t}^{-1} (\vec{X}_{j,t}-\vec{\mu}_{j,t})} \tag{3.2}$$

Where n is the n-dimensional from vector $\vec{X}_{j,t}$. In this case, n = 3 because we adopt RGB color space and K depends on computational power and available memory, normally range is 3-5 [13].

Color is an important factor to describe objects. In order to find the probability distribution of color characteristics, we assume different color channels are independent from each other [11], so variation matrix is defined as in equation (3.3) [13, 15].

$$\Sigma_{j,t} = \begin{matrix} (\sigma_{j,t}^2)^R & 0 & 0 \\ 0 & (\sigma_{j,t}^2)^G & 0 \\ 0 & 0 & (\sigma_{j,t}^2)^B \end{matrix} \quad (3.3)$$

Where $(\sigma_{j,t}^2)^{\mathrm{R}}$, $(\sigma_{j,t}^2)^{\mathrm{G}}$ and $(\sigma_{j,t}^2)^{\mathrm{B}}$ are the RGB channel variances.

Every time when a new pixel $X_{j,t}$ is observed it is checked against the already existing K distributions. A match is defined as in equation 3.4 [15].

$$\left|X_{j,t}^x - \mu_{j,t}^x\right| \leq 2.5 * \sigma_{j,t}^x \quad (3.4)$$

Where x denotes R and B, respectively. If a match is found for some distribution, then eq. 6 is updated. If no distribution is matched among the existing *K* distributions then replace the least probability distribution with the new distribution using the mean, weight and variance of the current pixel $X_{j,t}$, the initial high variance and low weight, respectively [42]. The least probable distribution is finding out by the lowest ω/σ value. The prior weights of *K* distributions at time *t*, $\omega_{k,t}$ are updated as the equation 3.5 given in [28].

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1} + \alpha(S_{k,t}) \quad (3.5)$$

Where α is the learning rate having the values between 0 to 1 and speed at which distribution parameters change depends on time constant $1/\alpha$.

$S_{k,t}$ is 1 if match is found and 0 for the remaining values. $\vec{\mu}_{j,t-1}$ and $\alpha_{j,t-1}$ are parameters for unmatched distributions that contain the same value and the parameters that match the new distribution are updated using equation 3.6 – 3.10 given in [26].

$$\vec{\mu}_{j,t} = (1 - \rho)\vec{\mu}_{j,t-1} + \rho\, X_{j,t} \quad (3.6)$$

$$\sigma_{j,t}^{2\,R} = (1 - \rho)\, \sigma_{j,t-1}^{2\,R} + \rho(R_{j,t} - \mu_{j,t-1}^R)^2 \quad (3.7)$$

$$\sigma_{j,t}^{2\,G} = (1 - \rho)\,\sigma_{j,t-1}^{2\,G} + \rho(G_{j,t} - \mu_{j,t-1}^{G})^2 \tag{3.8}$$

$$\sigma_{j,t}^{2\,R} = (1 - \rho)\,\sigma_{j,t-1}^{2\,R} + \rho(R_{j,t} - \mu_{j,t-1}^{R})^2 \tag{3.9}$$

$$\rho = \alpha * \eta(\vec{X}_{j,t}, \vec{\mu}_{j,t-1}, \sigma_{j,t-1}) \tag{3.10}$$

Where, parameter ρ is the second learning rate.

The Gaussian parameters must be adjusted when a match is found within the existent *K* Gaussian distributions [36]. The weights (ω) of all Gaussian distributions must be adjusted and the standard deviation (σ) and the mean (μ) are updated for the matched Gaussians, while unmatched Gaussians remain same [15]. Update the weights, deviations and means using the equations (3.6) to (3.9) and ρ is calculated using equation (3.10).

After every updating operation, the *K* distribution are ordered by the value of ω/σ, and the most likely background distribution is always on the top of the *K* distribution then chose the first *R* distribution as the real background using equation (3.11).

$$R = arg_r \min\,(\Sigma_{k=1}^{r}\omega_k > T) \tag{3.11}$$

Where threshold *T* is the minimum fraction of background model or it is defined as the minimum prior probability of background to be in the image scene [3].

In order to get the faster adaptation of mean and the variance value, we just cut off the η component from the ρ definition. The purpose of updating the parameters in time is that σ will have a larger value than the proposed values in many literatures [26]. If any object moves suddenly than it will be detected using former learning rate while with the larger σ value the true background will get the dominant place. To make the background subtraction more efficient cut off the η value that save the time and space [13]. Then there is no requirement to store the value $\eta(\vec{X}_{j,t}, \vec{\mu}_{j,t}, \Sigma_{j,t})$. Record the *K* distributions by the value $\omega_{k,t}$ instead of ω/σ, thus the computational load will be less. After this reduction the parameters that must be computed and stored are mean value vector $\mu = (\mu^R, \mu^G, \mu^B)$ and variance vector $\sigma = (\sigma^R, \sigma^G, \sigma^B)$ and weight $\omega_{k,t}$ of each model [12]. But three additional parameters ρ, η, ω/σ must be calculated and stored in original GMM. Therefore, computational load will be higher in original

GMM. Thus performance of our method is more efficient than original GMM. Figure 3.2(a) shows the video image and background subtraction result is shown in figure 3.2(b).

(a) (b)

Figure 3.2: Results of Background Subtraction

(a) Video Image (b) Image after Background Subtraction

3.1.2 Temporal Differencing

Temporal differencing makes use of the difference of the pixel to pixel between two or three consecutive frames in a video to extract moving regions. It is an extremely sensitive method to dynamic scene changes. It fails to extract all the relevant pixels of the foreground objects especially when the object moves slowly or has the uniform texture [14]. When any foreground object stops moving in video scene, temporal differencing method fails to detect the change between consecutive frames and loose the stopped object. Then it required special supportive algorithms to detect stopped objects.

We preset a two consecutive frame temporal differencing method. Let l_n(k) represents the intensity value of gray level at pixel position (k)and at time instance n of video frame sequence l which is in the range [0, 255]. In a two frame temporal differencing method, a moving pixel satisfies the equation (3.12) given in [29].

$$|l_n(k) - l_{n-1}(k)| > \tau_n(k) \tag{3.12}$$

Where, τ_n is the pre-defined threshold. Hence, if any object has uniform colored regions then equation 3.14 fails to detect some pixels inside the region even if the

object moves in the video [41]. The per pixel threshold, is initially set to a pre defined value and later updated as equation (3.13).

$$\tau_{n+1}(k) = \begin{cases} \alpha\tau_n + (1 - \alpha)(\delta \times |l_n(k) - l_{n-1}(k)|), & k \in BG \\ \tau_n(k), & k \in FG \end{cases} \quad (3.13)$$

Where $\alpha, \delta \in [0.0, 1.0]$ are the learning constants which determine the amount of information that is put to the background and threshold from the incoming image. If background pixels are considered as time series then background image is a weighted temporal average of incoming image sequences and threshold image is considered as a weighted temporal average of δ times the difference of incoming image sequences and the background [6].

3.1.3 Pixel Level Post Processing

The output of foreground detection algorithms we explained in background subtraction techniques generally contains noise and therefore it is not appropriate for further processing without post processing operations. In foreground detection there are several factors that cause the noise such as [20]:

Camera noise: This noise is caused by the image acquisition components of camera. The intensity of an edge pixel between two different colored objects may be corresponding to one object's color in one frame of video and in the next frame to the other's color [37].

Reflectance noise: When some parts in the background scene reflect the light then foreground detection algorithm detect reflectance as foreground regions and it fails to detect the actual foreground object [15].

Shadow noise: Most of the foreground detection algorithms detect shadow as foreground that cast on objects. It makes the algorithm fails to detect actual foreground object accurately [20].

3.1.3.1 Shadow Elimination and Noise Removal

Shadow detection as a foreground object creates confusion for next analysis phase. It is necessary to distinguish between objects and their shadows. The RGB colors

vectors of the pixel in shadow region have the same direction with a little deviation to the same color vector of corresponding background pixels and the brightness value of the shadow pixel is less than the brightness of the corresponding background pixels [5]. In order to define this, let I_k represent the RGB colors of current image pixel at position, and represent the colors of corresponding background pixel. Furthermore, let $\hat{I}_k$ represent the vector that starts at origin $O(0,0,0)$ and end at point I_k, let $\hat{B}_k$ is the vector for the corresponding background pixel B_k and let d_k represent the dot product (.) between $\hat{I}_k$ and $\hat{B}_k$. Figure 2 shows these points and vectors in RGB colors space. This approach of shadow removal makes use of the observation discussed in [13, 37]. This shadow detection scheme classifies a pixel as shadow that is the part of the detected foreground if it satisfies the conditions specified in equations (3.14) and (3.15) given in [37].

$$\left(d_k = \frac{\hat{I}_k}{\|\hat{I}_k\|} \cdot \frac{\hat{B}_k}{\|\hat{B}_k\|}\right) < \tau \tag{3.14}$$

$$\|\hat{I}_k\| < \|\hat{B}_k\| \tag{3.15}$$

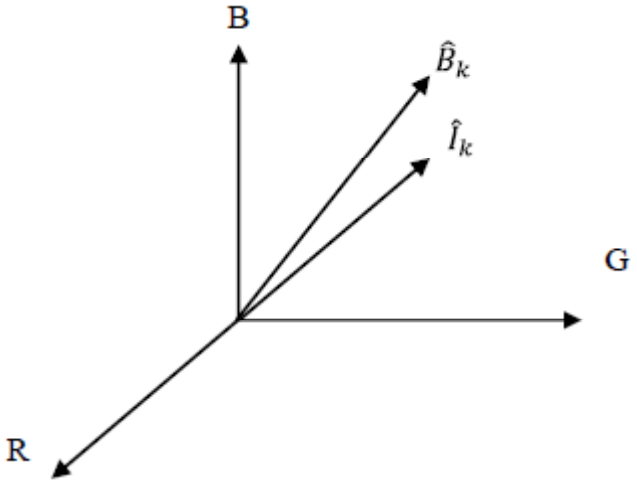


Figure 3.3: RGB vectors of current image pixel, $\hat{I}_k$ and corresponding background pixel, $\hat{B}_k$.

Where τ is a predefined threshold that is close to 1. Dot product is used to check whether $\hat{I}_k$ and $\hat{B}_k$. have same direction or not and if the dot product (d_k) of normalized $\hat{I}_k$ and $\hat{B}_k$ is close to 1, this implies that both vectors are in same direction with little amount of deviation [28].

In order to remove noise, morphological operations dilation and erosion are applied to the foreground pixel map. Our aim is to apply these operations to remove noisy foreground pixels, which do not correspond to actual foreground region and to remove the noisy background pixels inside or near the actual foreground region. Erosion removes one unit thick boundary pixels from foreground regions and dilation is the reverse of erosion that expands the boundaries of foreground region with one unit thick pixels [9]. The difficulty in applying these morphological operations is to decide the amount and order of these operations. The amount these operations affects the quality and the computational complexity and the order affects the quality of noise removal.

3.1.3.2 Detecting Connected Regions

After detecting foreground objects and applying the post processing operations to remove shadow and noise, the filtered foreground pixels are grouped into the connected components (blobs) by using a two level connected component algorithm [4]. After finding the individual blobs, which correspond to objects, calculate the bounding box of these regions.

3.1.3.3 Region Level Post-Processing

Some small regions remain as noise due to inaccurate object segmentation after removing the pixel level noise. To eliminate this type of noise, the average region size is calculated in terms of pixels for each frame. Regions that have smaller size than the fraction of average region size are eliminated from the foreground pixels map [41]. Some objects parts are found as disjointed from the primary body due to segmentation errors. In order to correct this shortcoming, bounding boxes of regions are merged together that are close to each other. Figure 3.4 shows the result of shadow elimination and morphological operation.

Figure3.4: Result of pixel level post processing

3.1.3.4 Extracting Object Features

After segmentation of foreground regions, we extract the features of corresponding objects from the current image scene. These features are the size (S_i) and center of mass (C_e) of the object. In order to estimate the size of the object we just calculate the number of pixels of foreground contained in the bounding box and to calculate the center of mass $C_e = (xC_e, yC_e)$ of an object O, use the equation (3.16) given in [6].

$$xC_i = \frac{\sum_i^k x_i}{k}, \qquad yC_i = \frac{\sum_i^k y_i}{k} \tag{3.16}$$

Where k is the number of pixels in object O.

3.2 Object Tracking

The objective of object tracking is to construct a correspondence between objects in consecutive frames. Detection of objects for tracking in frame by frame is a significant and difficult problem. It is a crucial part for video surveillance system since without tracking the object, the system could not extract the cohesive temporal information about objects and further higher level event analysis steps would be difficult [17, 38]. On the other hand, inaccurate segmentation of foreground objects due to occlusions, shadow, and reflectance makes tracking a difficult and active research problem.

An object level tracking algorithm is used in our video surveillance system. We don't track the object parts such as limbs of human, but track the object as a whole from frame to frame. In tracking step, the extracted information is adequate for most of the video surveillance applications. Our approach uses the object features such as center of mass, size and bounding box that are extracted to establish a matching between objects in frame to frame [19, 39]. The tracking algorithm detects the object occlusion and distinguishes object identities and tracking algorithm is able to detect abandoned and carried objects as well. The first step in object tracking algorithm is to match the objects in previous frame to new objects in the current frame [24]. We now explain the correspondence based object matching in detail.

The matching of objects is stored in bi-partite graph $G(l,m)$. In this graph, vertices show the objects (one vertex partition depicts the previous objects O_i's, and the other partition depicts the new objects, O_j's) and edges depict a match between the two objects. In $G(l,m)$, l is the size of partition for previous objects, and m is the partition size for new objects. A sample matching graph is shown in figure 3.5 make use of the observation given in [16]. For each previous object O_i, iterate over a new object and first check whether a new object O_j in new objects list is close to O_i or not [38]. Two objects are close to each other, which have center of mass C_i and C_j if following condition specified in equation (3.17) is satisfied.

$$Dist(C_i, C_j) < \tau \tag{3.17}$$

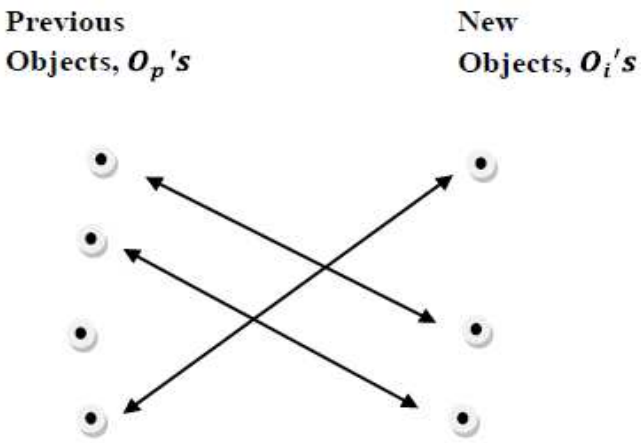


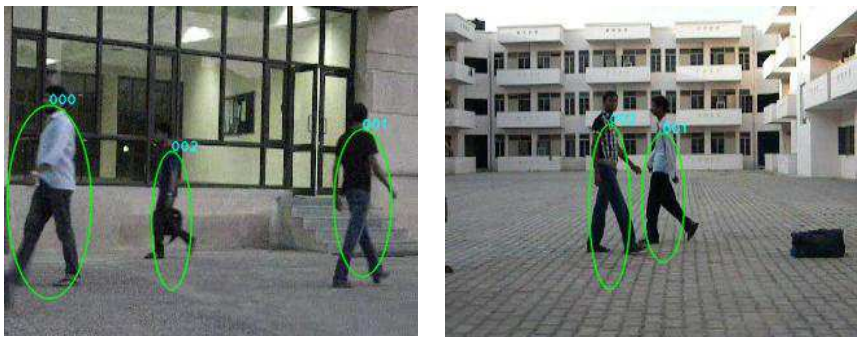
Figure 3.5: Sample object matching graph [16]

It is not necessary for every two objects that they a successful match if they are close to each other within a threshold. So in next step to improve correct matching, we check similarity of two objects. We take size ratio of the objects as the criterion for similarity comparison [21]. This check is used the fact that objects do not shrink or grow too much in the consecutive frames. The two objects could be alike if they satisfy the equation (3.18).

$$\frac{S_i}{S_j} < u \;\; or \;\; \frac{S_j}{S_i} < u \tag{3.18}$$

Where μ is a pre-defined threshold and S_j is size of object O_j. If above two steps are executed then it would occur to the condition where a previous object could match to more than one object. After second step further check that the object O_i has already a match or not [24]. Connect the corresponding vertices in bi-partite graph $G(l,m)$ if object O_i does not have already a match and continue with next object O_j, but if O_i has already a match O_t, then additional steps are required to resolve the correspondence conflict.

In order to resolve a matching conflict, the correspondences of objects O_j and O_t are compared to O_i. In other words, by comparing the correspondence of O_j and O_i with the correspondence of O_t and O_i, we try to find which one of O_j or O_t is the correct match with the object O_i. The correspondences of objects are compared by using distance between center of mass point of O_i and O_j or O_t . Let d_{sj} be the distance between center of mass of O_i and O_j, and let d_{st} be the distance between center of mass of O_i and O_t. The correspondence is concluded in favor of if $d_{st} < d_{sj}$, otherwise resolution is in favor of O_j [38]. Figure 3.6 shows the result of object tracking with 3 persons in figure 3.6(a) and with 2 persons in figure 3.6(b).



(a) (b)

Figure 3.6: Results of Tracking

3.3 Activity Detection

Terms appearing throughout the literature, such as "activity", "behavior", "action", "scenario", "gesture", and "event" are used to describe the same concepts. These concepts have an ambiguous definition in literature. In this section the aim is to clarify these terms and propose a specific terminology which we will use to describe specific work. Activity detection could either be an application in sports, where the activity could be defined as a goal. It could also be an application in the surveillance area like abandoning or carrying of bag or accident detection through traffic surveillance video [1, 5]. It could be a general application like detection of fire. The abandoned or carried object detection process uses the output of the tracking model and features extraction as the input for each frame, and determines if they are abandoned or carried [36,20]. The output of tracking step contains the number of objects, their identities and features extraction step contains the parameters of bounding box. To tackle the abandoned or carried bag problem, the detection process has the following steps:

Step 1: Identify the object bag item.
Step 2: Identify the person.
Step 3: Test for unusual activity detection.

Step 1: To identify the bag, we use the result of the tracker bounding box (X_l) and the foreground segmentation F_s to form object blob by taking the intersection of area in each frame $\overline{X}_l^i \cap F_s$. The likelihoods defined in equation (3.19) and in equation (3.20) are such that small and stationary blob is more likely to be item of bag [5].

$$p_h(B^i = 1|\overline{X}_{1:l}^i) \propto N(S_l^i, \mu_h, \sigma_h) \tag{3.19}$$

$$p_k(B^i = 1|\overline{X}_{1:l}^i) \propto \exp(-\lambda v_l^i) \tag{3.20}$$

Where p_h is the size likelihood, p_k is the velocity likelihood, $B^i = 1$ points that blob i is a bag, S_l^i is the size of blob i at time l, μ_h is the mean bag blob size, σ_h is the bag blob variance, v_l^i is the blob velocity and λ is a hyper-parameter. The long living blob is more likely to be the abandoned bag or carried bag location, frame wise likelihoods are summed without normalizing by blob lifetime [37]. The overall likelihood in

equation (3.21) that a blob is a carried or abandoned bag item combines p_h and p_k [25].

$$p(B^i = 1|\overline{X}_{1:l}^i) \propto \sum_{t=i:T} N(S_l^i, \mu_h, \sigma_h)\, exp\,(-\lambda v_l^i) \tag{3.21}$$

The bag likelihood term $p(B^i = 1|\overline{X}_{1:l}^i)$ gives preference to long lasting and small objects. The person is selected by thresholding the likelihood using equation (3.22) given in [25].

$$p\left(B^i = 1\middle|\overline{X}_{1:l}^i\right) > T_B \tag{3.22}$$

A shape template $\mathcal{T}^i$ is constructed from the longest frame segment below a low threshold v_t, to model what bag looks like when it is stationary [34]. Bag existence likelihood is determined for blob person by extracting image features from the binary image at stationary bag $\mathcal{L}_l$ and performing an element wise multiplication using equation (3.23) [5, 25].

$$p(E_l = 1/B^i) \propto \sum_c \sum_d \mathcal{T}^i\,(c,d) \times \mathcal{L}_l(c,d) \tag{3.23}$$

Where $E_l = 1$ indicates that bag exists at time *l*, and *c* and *d* are pixel indices [5].

Step 2: Separate bounding boxes only result when the person goes away from the bag or person comes to take the bag, then one of the two cases can occur:

(1) the original bounding box follows the person and a new box is formed to track the abandoned bag or carried bag location, or

(2) the original bounding box stays with the abandoned bag or carried bag location, and a new bounding box is formed and follows the person [43].

Thus, to identify the person, check the history of tracker when bag first appeared as determined by existence likelihood of bag. If that tracker goes away and dies while bag remains stationary, it must be one identifying the owner. If tracker remains with the bag, we begin search for nearby births of new trackers. The first nearby birth is deemed the person. If no nearby birth is found, then bag has no owner, and no need to go to further step [36].

Step 3: With the bag and the person detected, and having the knowledge of their location, the last job is straightforward: determining if the bag is abandoned or carried

[5]. If the distance between the location of the center of mass of the abandoned object or carried object and the person is greater than fixed value and increases continuously then unusual activity is detected.

3.4 Summary

In this chapter we have discussed how objects can be segmented from a video. Two methods Gaussian mixture models and temporal differencing have been seen in this regard. Temporal differencing fails to extract all the relevant pixels of the foreground objects especially when the object stops moving or has the uniform texture. The Gaussian mixture model method was dealt in detail [13]. Then the resultant foreground pixel map was subjected to post processing operations pixel-level post-processing and connected component labeling. These operations resulted in the real final foreground pixel map, from which the features like center of mass, perimeter, and bounding box were extracted [20].

Then a tracking phase which successfully tracks the whole body object in consecutive frames is discussed. This approach makes use of features like center of mass, size and bounding box. We first associate the objects between previous frame and current frame using center of mass matching method, where we considered the distance between center of mass of objects. To handle the object occlusions, histogram based correspondence matching approach is incorporated in object association. In this approach, if objects entered into an occlusion the identity of objects could be recognized after a split. The tracking information obtained from the tracking module is then used for further processing by the unusual activity detection phase [24].

In the activity detection phase, we use Bayesian inference as the activity modeling. The abandoned or carried object detection process uses the output of the tracking model and foreground segmentation as the input for each frame, and determines if object is abandoned or carried. The output of tracking step contains the number of objects, their identities and parameters of bounding box are calculated in features extraction step. Separate bounding boxes only result when the person goes away from the bag or person comes to take the object. With the bag and the person identified, and

knowledge of their location in a given frame, then we determine that the bag is abandoned or carried [5].

Chapter 4

Experimental Results and Evaluation

The application is implemented in C++ using OpenCV library [43] in Linux environment. OpenCV is an open source (see *http://opensource.org*) computer vision library available from *http://SourceForge.net/projects/opencvlibrary*. The library is written in C and C++ and runs under Linux, Windows and Mac OS X.

The architecture of the application is made flexible in order to load different types of video clips. All of the tests in the next sections are performed on Ubuntu 8.10 Linux operating system on a computer with an Intel Pentium IV 3.06GHz CPU and 1 GB of RAM.

If a static object introduced or moving object stops into the scene then it will be merged into the background. But we don't want it to be merged into the background because this would cause to lose the object. So a time control method is required for this application so that foreground doesn't merge. We can see that the reason for merging the object is due to updating mechanism of GMM. The merging time of objects can be controlled through the updating algorithm. An existing time control factor is added for every single Gaussian Model [13]. If we create five single Gaussians for one pixel and a moving object is stopped into the scene, then the last Gaussian will be replaced with new Gaussian model, because none of five will be matched. Then existing time control factor of this model set to $ד = 0$. Because the object is still in the scene so next time this model will be matched again and background models are not updated. This time set $ד = ד + 1$, and check if $ד \leq T$, then don't perform the update operation unless $ד > T$. Where $ד$ is measured in number of frames and T is user defined threshold [27].

Our activity detection results are more robust and reliable in case of occlusion and shadow elimination. If the distance between center of mass of abandoned object or carried object location and the person is greater than the fixed value and increases continuously then unusual activity is detected.

4.1 Comparison of Background Subtraction Results

4.1.1 Results of GMM given in OpenCV:

In figure 4.1(a) a person is walking with bag and its background subtraction result is shown in figure 4.1(b). In figure 4.1(c) Person abandons the bag and goes away from the bag and in figure 4.1(d) only person is detected and stationary bag is not detected.

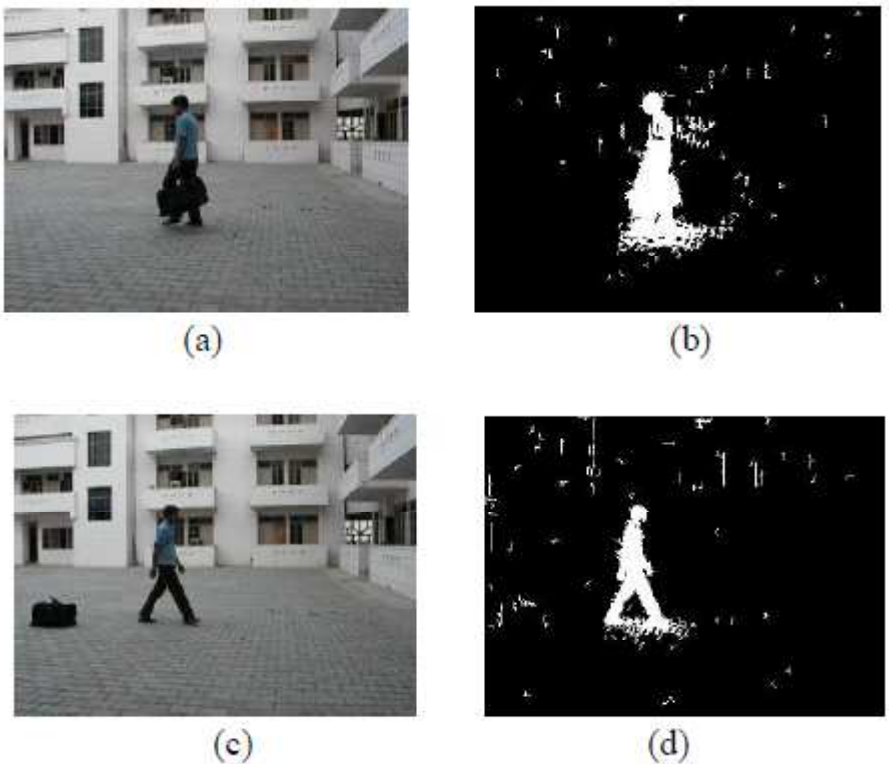


Figure 4.1: Results of GMM given in OpenCV

4.1.2 Results of Background Subtraction and Tracking:

In figure 4.2(a) a person is walking with bag and its background subtraction result is shown in figure 4.2(b). In figure 4.2(c) Person abandoned the bag and goes away from the bag and in figure 4.2(d) both person and bag are detected.

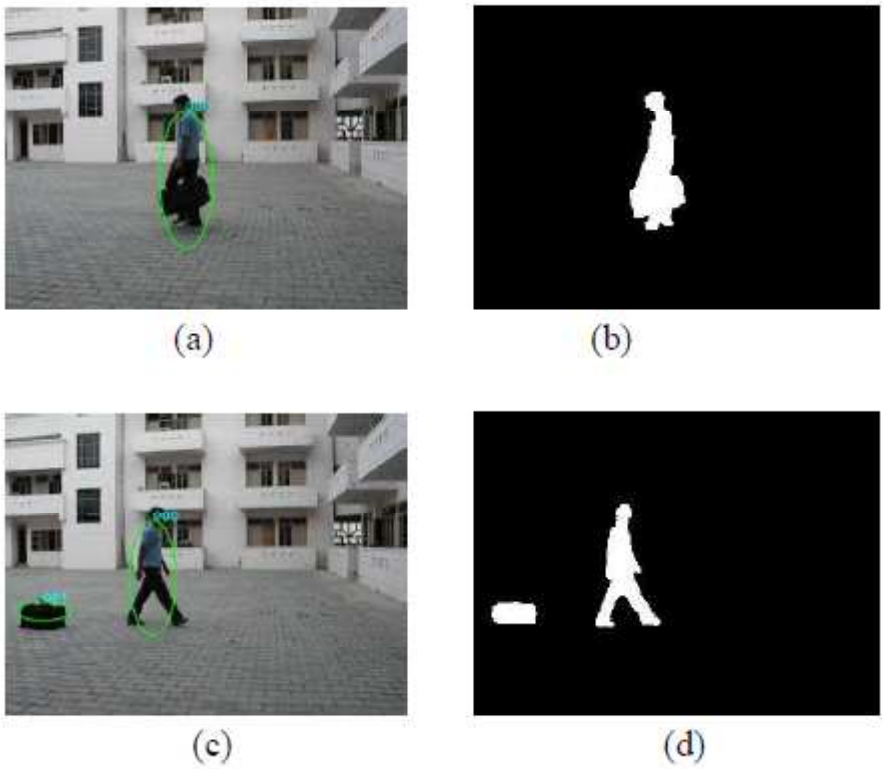


Figure 4.2: Our results of background subtraction and tracking

4.2 Activity Recognition Results

The proposed method is about 1.5 times faster than original Gaussian mixture model and our activity detection result are more robust and reliable in case of occlusion and shadow elimination. If the distance between center of mass of abandoned object or carried object location and the person is greater than fixed value and increases continuously and position of abandoned object does not change than unusual activity is detected.

4.2.1 Results of Abandoned Bag Detection:

Video-1 Results:

In figure 4.3(a) a person is walking and its corresponding background subtraction result is shown in figure 4.3(d). In figure 4.3(b) person abandoned the bag and its corresponding background subtraction result is shown in figure 4.3(e). In figure 4.3(c) person goes away from the abandoned bag, so unusual activity is detected and its

background subtraction result (person and bag both are detected) is shown in figure 4.3(f).

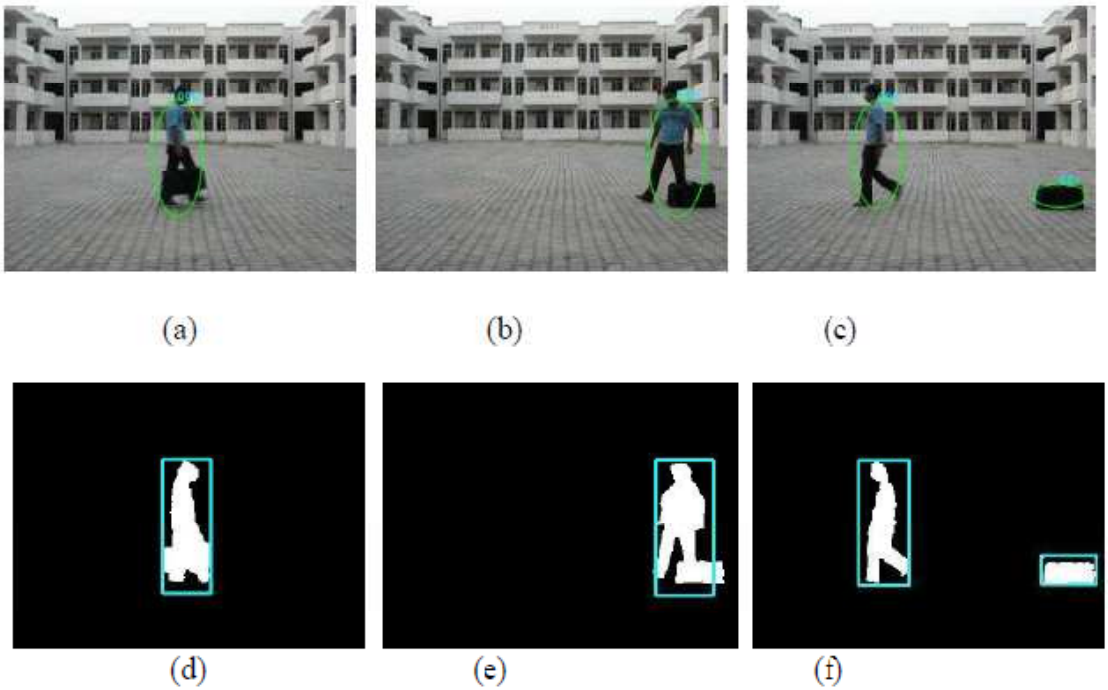


Figure 4.3: Video 1 results of abandoned bag detection

Video-2 Results:

In figure 4.4(a) a person is walking and its corresponding background subtraction result is shown in figure 4.4(d). In figure 4.4(b) person abandoned the bag and its corresponding background subtraction result is shown in figure 4.4(e). In figure 4.4(c) person goes away from the abandoned bag, so unusual activity is detected and its background subtraction result (person and bag both are detected) is shown in figure 4.4(f).

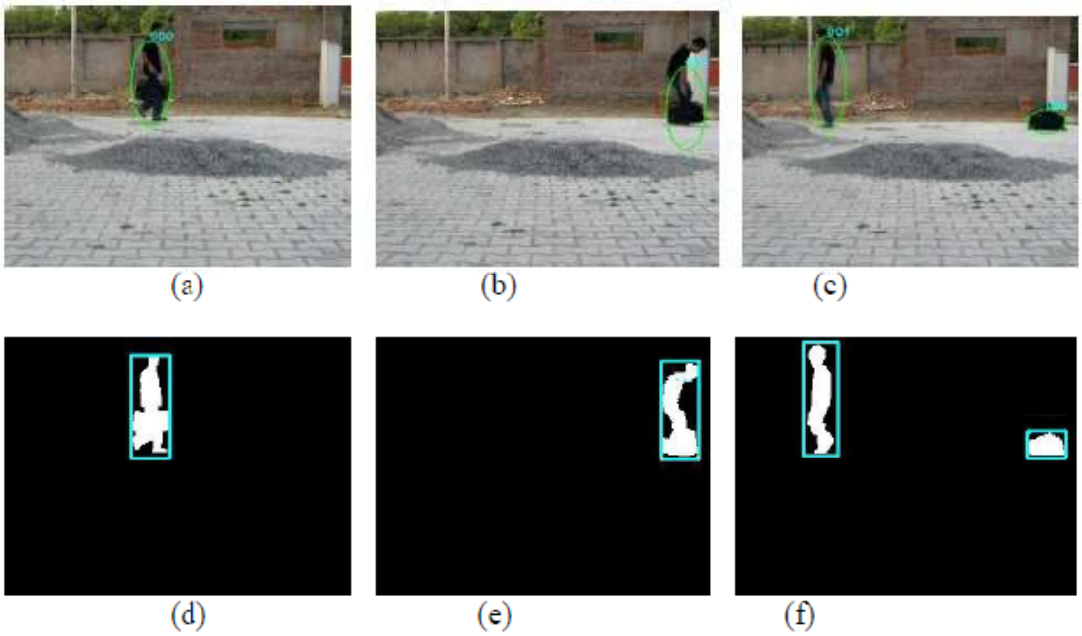


Figure 4.4: Video 2 results of abandoned bag detection

Video-3 Results:

In figure 4.5(a) a person is walking and its corresponding background subtraction result is shown in figure 4.5(d). In figure 4.5(b) person abandons the bag and its corresponding background subtraction result is shown in figure 4.5(e). In figure 4.5(c) person goes away from the abandoned bag, so unusual activity is detected and its background subtraction result (person and bag both are detected) is shown in figure 4.5(f).

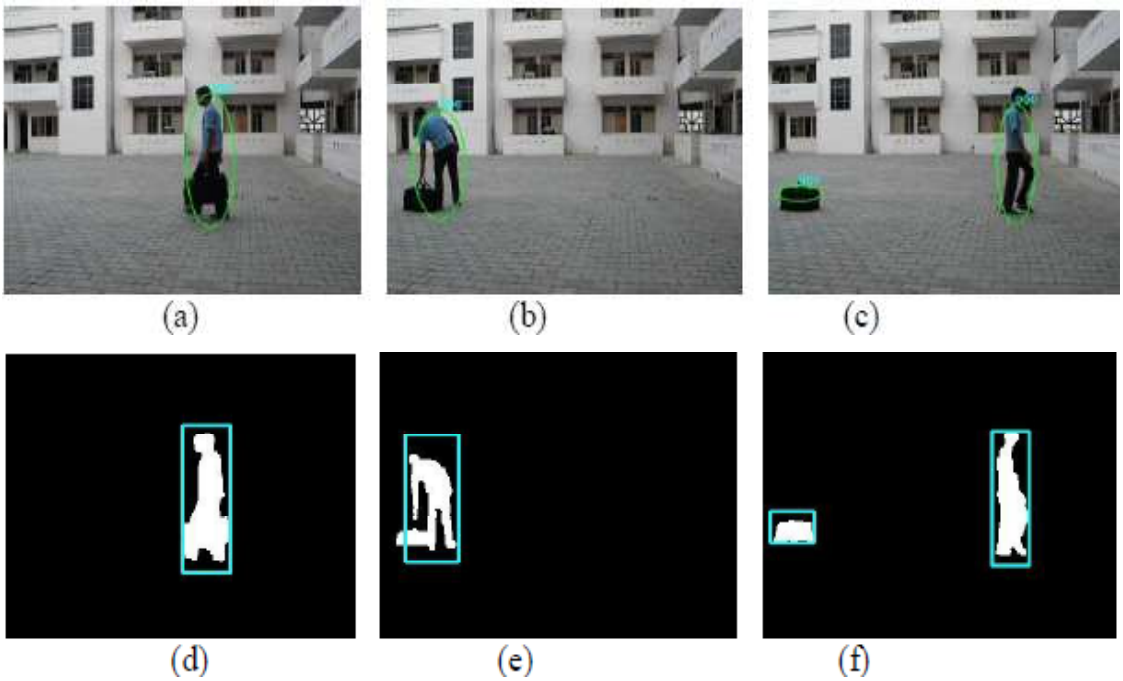


Figure 4.5: Video 3 results of abandoned bag detection

4.2.2: Results of Carried object detection:

Video-4 Results:

In figure 4.6(a) a person is walking and its corresponding background subtraction result is shown in figure 4.6(d).). In Figure 4.6(a), bag is the part of background so it does not detect in figure 4.6(d) as a foreground object. In figure 4.6(b) person carried the bag and its corresponding background subtraction result is shown in figure 4.6(e). In figure 4.6(c) person goes away from the bag location, so unusual activity is detected and its background subtraction result (person with bag and bag location both are detected) is shown in figure 4.6(f). In figure 4.6(f), bag location is detected as foreground object which shows that bag has been removed from its initial location.

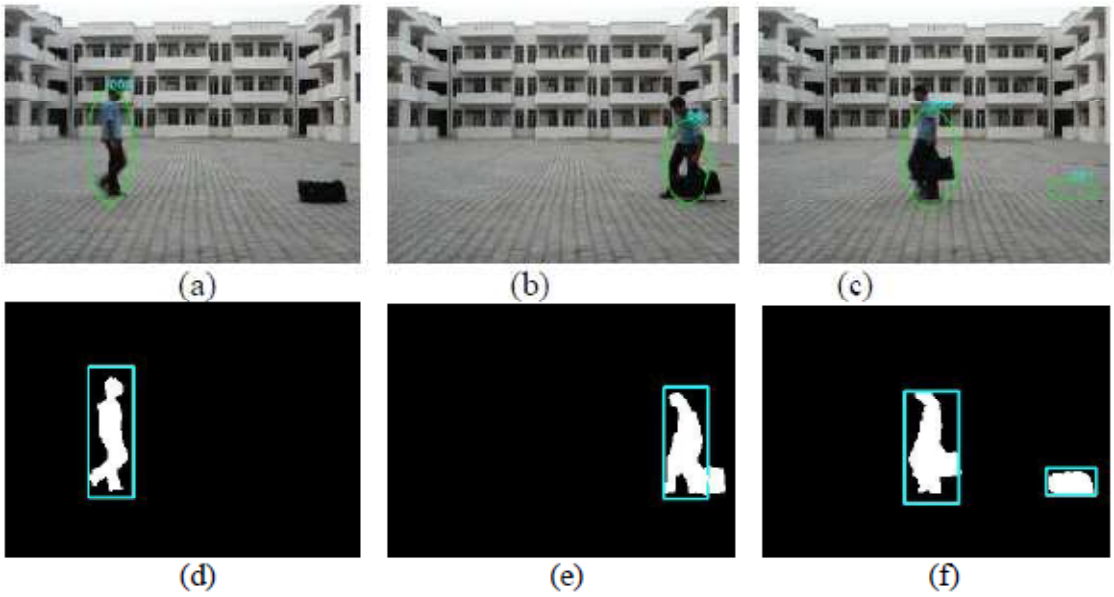


Figure 4.6: Video 4 results of carried bag detection

Video-5 Results:

In figure 4.7(a) a person is walking and its corresponding background subtraction result is shown in figure 4.7(d). In Figure 4.7(a), bag is the part of background so it does not detect in figure 4.7(d) as a foreground object. In figure 4.7(b) person is near to bag and its corresponding background subtraction result is shown in figure 4.7(e). In figure 4.7(c) person carried the bag and goes away from the bag location, so unusual activity is detected and its background subtraction result (person with bag and bag location both are detected) is shown in figure 4.7(f). In figure 4.7(f), bag location is detected as foreground object which shows that bag has been removed from its initial location.

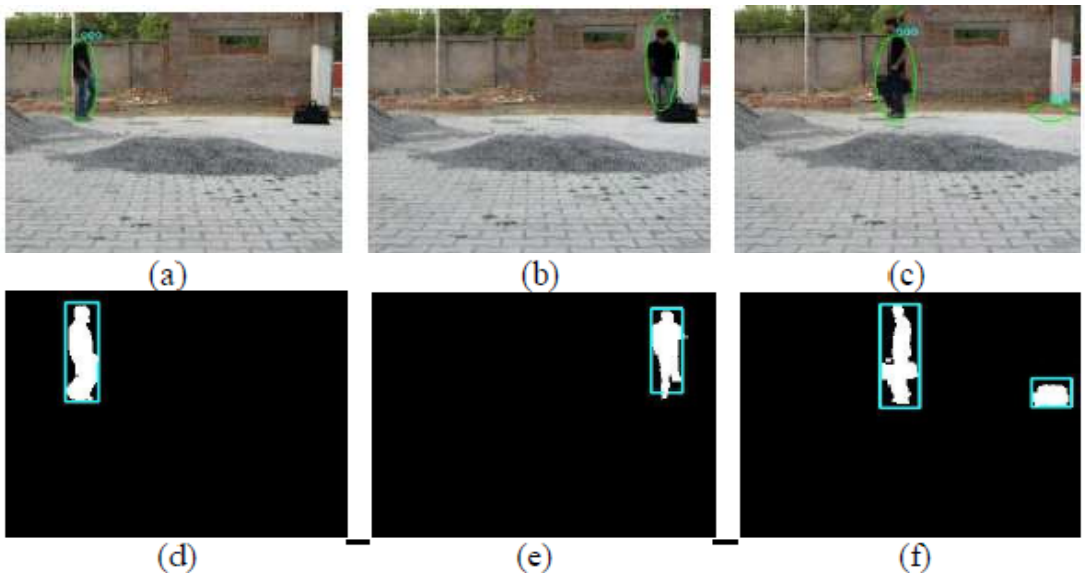


Figure 4.7: Video 5 results of carried bag detection

Video-6 Results:

In figure 4.8(a) a person is walking and its corresponding background subtraction result is shown in figure 4.8(d). In Figure 4.8(a), bag is the part of background so it does not detect in figure 4.8(d) as a foreground object. In figure 4.8(b) person is near to bag and its corresponding background subtraction result is shown in figure 4.8(e). In figure 4.8(c) person carried the bag and goes away from the bag location, so unusual activity is detected and its background subtraction result (person with bag and bag location both are detected) is shown in figure 4.8(f). In figure 4.8(f), bag location is detected as foreground object which shows that bag has been removed from its initial location.

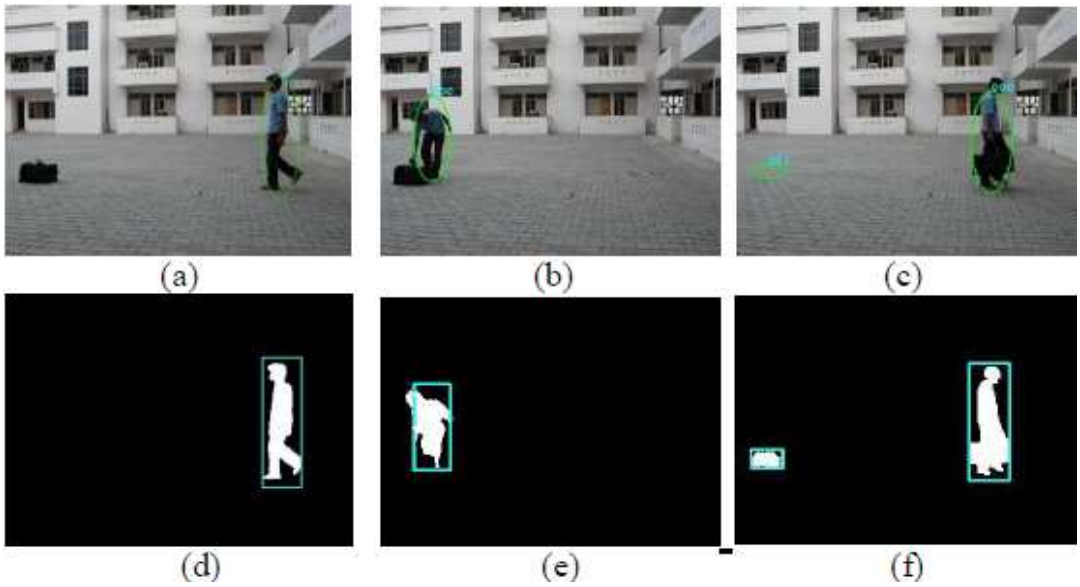


Figure 4.8: Video 6 results of carried bag detection

Table-4.1 explicitly demonstrates the advantage of proposed method over surveillance system using GMM given in OpenCV for activity detection.

Table-4.1. Comparison of Proposed Method with GMM

Video	Duration of Video (sec)	No. of Frames in Video	Time for proposed method using Improved GMM (sec)	Time for system using GMM in OpenCV (sec)
Video 1	13	320	31	35
Video 2	14	348	35	40
Video 3	10	252	20	24
Video 4	10	250	19	24
Video 5	10	249	19	24
Video 6	12	248	26	30

The proposed method has been tested on a number of videos in different situation. This choice of different contexts was made to emphasize the reliability and robustness of the propose method In order to have a quantitative estimation of error, we characterized the detection rate (DR) and the false alarm rate (FAR) [26].

$$DR = TP_{+VE}/(TP_{+VE} + FN_{-VE}) \quad (4.1)$$

$$FAR = FP_{+VE}/(TP + FP_{+VE}) \quad (4.2)$$

Where TP_{+VE} (true positives) are the actual detected foreground regions; FP_{+VE} (false positives) are the detected regions that do not correspond to actual foreground region; and FN_{-VE} (false negatives) are moving objects that do not detected.

In Table 4.2, the results are obtained on the different image sequences are shown compared with two traditional methods. The DR parameter is always over 93%, and the FAR parameter is under 3.6%, which demonstrating that the proposed method is reliable, and robust in the different environmental context.

Table: 4.2 Rates to Measure the Confidence for Sequence

Algorithm	DR%	FAR%
Temporal Differencing	41.35	65.85
Gaussian Mixture Method	65.27	45.72
Proposed Method	95.78	3.74

For the proposed method, a sample of size 100 was used to represent the background; the update is performed using the detected results directly as the update decision. For the proposed method, the maximum number of distributions allowed at each pixel was 10.

In figure 4.9, the graph shows that the proposed method is more robust than traditional GMM and Temporal Differencing method. As the background complexity increases, false negative increases in traditional GMM and temporal differencing. The detection of regions that are not actual foreground objects also increases in both existing methods than the proposed method.

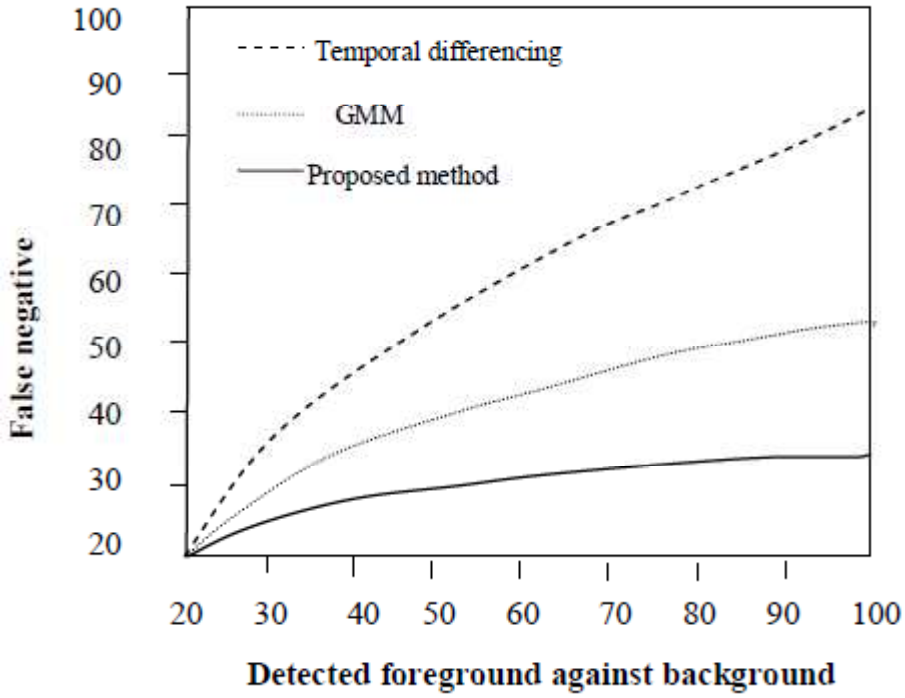


Figure 4.9: False negative with detected foreground against background

Some reasons are given below for detecting FP_{+VE} and FN_{-VE} in video sequence.

- Bag with little or no protrusion
- Protruding parts of clothing
- Due some camera noise
- Carried/abandoned object not segmented from background
- Swinging small objects

Chapter 5

Conclusion and Future Work

5.1 Conclusion

In this thesis, we presented a set of methods for a video surveillance system. We implemented different object detection algorithms and compared them by results. The adaptive GMM background subtraction technique gives most promising results in terms of quality of object detection and computational complexity.

The proposed object tracking algorithm successfully tracks the whole body objects in consecutive frames. In sample applications, our tests show that correspondence based matching approach yields promising results and no complicated techniques are required for tracking of whole body of objects. In handling simple object occlusions, histogram based matching approach distinguishes the objects identities entered into an occlusion after a split. But in crowded scenes such approach is not feasible to handle the object occlusions, thus a pixel based approach, like optical flow is a requisite to identify accurate object segments [22].

Our system is designed for unusual activity detection task for one person in the offline videos and the two unusual activities are abandoned or carried bag detection. The implementation of this approach runs at 10-12 frames per second on Pentium IV 3.06 GHz for 320×240 color video frames. The application is implemented in C++ using OpenCV library in Linux environment with a single camera view. The methods we presented for video surveillance system show promising results for abandoned object and carried object detection.

5.2 Future Work

We present a surveillance system that works on offline videos so it is required to convert it into real time. No background subtraction algorithm is perfect for true object detection, so our method needs improvements in handling partially object occlusions, sudden illumination changes, and darker shadows. To enhance object detection results and eliminate inaccurate object segmentation, higher level semantic analysis extraction steps would be used. Other possible avenues for future work include using multiple cameras views that can reduce the object occlusion problem and investigating methods for maintaining object identities in the tracker better [12]. Usually real world scenarios are more complicated than the scenarios we presented here, in terms of number of persons involved in the activities and variation in execution style. So more sophisticated algorithms are needed to consider to handle such complexities. This system can be used as an initial base system for advanced research in the field of video surveillance system.

Chapter 6

References

1. Chi-Hung Chuang, Jun-Wei Hsieh, Luo-Wei Tsai, Sin-Yu Chen, and Kuo-Chin Fan, "Carried Object Detection Using Ratio Histogram and its Application to Suspicious Event Analysis," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 911 – 916, June 2009.

2. Gian Luca Foresti, Lucio Marcenaro, and Carlo S. Regazzoni, "Automatic Detection and Indexing of Video-Event Shots for Surveillance Applications," *IEEE Transactions on Multimedia*, vol. 4, no. 4, pp. 459 – 471, 2002.

3. Javier Varona, Jordi Gonzàlez, Ignasi Rius, and Juan José Villanueva, "Importance of Detection for Video Surveillance Applications," *Optical Engineering*, vol. 47, no. 8, August 2008.

4. Jie Yang, Jian Cheng, and Hanqing Lu, "Human Activity Recognition Based on the Blob Features," in *Proc. IEEE International Conference on Multimedia and Expo*, pp. 358 – 361, June 2009.

5. Fengjun Lv, Xuefeng Song, Bo Wu, Vivek Kumar Singh, and Ramakant Nevatia, "Left-Luggage Detection using Bayesian Inference," in *Proc. IEEE 9th International Workshop on Performance Evaluation of Tracking and Surveillance*, New York, USA, pages 83 – 90, June 2006.

6. Ahmed Fawzi Otoom, Hatice Gunes, and Massimo Piccardi, "Automatic Classification of Abandoned Objects for Surveillance of Public Premise," in *Proc. IEEE Congress on Image and Signal Processing*, vol. 4, pp. 542 – 549, May 2008.

7. Pavan Turaga, Rama Chellappa, V. S. Subrahmanian, and Octavian Udrea, "Machine Recognition of Human Activities: A Survey, "*IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 11, pp. 1473 – 1488, Nov. 2008.

8. Fadhlan Hafiz, A.A. Shafie, M.H. Ali, and Othman Khalifa, "Event-Handling Based Smart Video Surveillance System," *International Journal of Image Processing (IJIP),* vol. 4, no. 1, 2008.

9. Feng Niu and M. Abdul-Mottaleb, "View-Invariant Human Activity Recognition Based on Shape and Motion Features," in *Proc. IEEE Sixth International Symposium on Multimedia Software Engineering*, pp. 546 – 556, 2004.

10. Md. Zia Uddin, J. J. Lee, and T. S. Kim, "Independent Component Feature-based Human Activity Recognition via Linear Discriminant Analysis and Hidden Markov Model," *in Proc. IEEE 30th Annual International Conference on Engineering in Medicine and Biology Society*, pp. 5168 – 5171, Aug. 20 – 25, 2008.

11. G. Lavee, E. Rivlin and M. Rudzsky, "Understanding Video Events: A Survey of Methods for Automatic Interpretation of Semantic Occurrences in Video," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 39, no. 5, pp. 489 – 504, Sept. 2009.

12. T. Bouwmans, F. El Baf, and B. Vachon, "Background Modeling using Mixture of Gaussians for Foreground Detection - A Survey," *Recent Patents on Computer Science*, vol. 1, no. 3, pp. 219-237, Nov. 2008.

13. Zhen Tang and Zhenjiang Miao, "Fast Background Subtraction Using Improved GMM and Graph Cut," in *Proc. IEEE International Conference on Image and Signal Processing*, vol. 4, pp. 181 – 185, 2008.

14. J. Mike McHugh, Janusz Konrad, Venkatesh Saligrama, and Pierre-Marc Jodoin, "Foreground-Adaptive Background Subtraction," *IEEE Signal Processing Letters*, vol. 16, no. 5, pp. 390 – 393, May 2009.

15. Li Ying-hong, Xiong Chang-zhen, Yin Yi-xin, Liu Ya-li, "Moving Object Detection Based on Edged Mixture Gaussian Models," in *Proc. IEEE International Conference on Intelligent Systems and Applications*, pp. 1-5, 2009.

16. Aishy Amer, "Voting-Based Simultaneous Tracking of Multiple Video Objects," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 11, pp. 1448- 1462, 2005.

17. Junzo Watada, Zalili Binti, and Musaand Graduate, "Tracking Human Motions for Security System," in *Proc. IEEE SICE Annual Conference,* pp. 3344 – 3349, August 2008.

18. Nizar Zarka, Ziad Alhalah, and Rada Deeb, "Real-Time Human Motion Detection and Tracking," in *Proc. IEEE International Conference on Information and Communication Technologies: from Theory to Applications,* pp. 1-6, 2008.

19. Robert Bodor, Bennett Jackson, and Nikolaos Papanikolopoulos, "Vision-Based Human Tracking and Activity Recognition," in *Proc. of 11th Mediterranean Conference on Control and Automation,* vol. 1, pp. 18-22, 2003.

20. Rita Cucchiara, Costantino Grana, Massimo Piccardi, and Andrea Prati, "Detecting Moving Objects, Ghosts and Shadows in Video Streams," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 25, no. 10, pp. 1337 – 1342, 2003.

21. Tao Gao, Zheng-guang Liu, Wen-chun Gao, and Jun Zhang "Robust Tracking and Object Classification towards Automated Video Surveillance," in *Proc. International Conference on Image Analysis and Recognition,* pp. 463 – 470, 2004.

22. Liang Wang, Weiming Hu, and Tieniu Tan, "Recent Developments in Human Motion Analysis," *Pattern Recognition*, vol. 36, no. 3, pp. 585 – 601, 2003.

23. Teddy Ko "A Survey on Behavior Analysis in Video Surveillance for Homeland Security Applications," in *Proc. IEEE 37th International Conferences on Applied Imagery Pattern Recognition,* pp. 1 – 8, 2008.

24. Mayssaa Al Najjar, Soumik Ghosh, and Magdy Bayoumi, "Robust Object Tracking Using Corresponding Voting for Smart Surveillance Visual Sensing Nodes," in *Proc. IEEE Sixteenth International Conference on Image Processing*, pp. 1133 – 1136, 2009.

25. Kevin Smith, Pedro Quelhas, and Daniel Gatica-Perez, "Detecting Abandoned Luggage Items in a Public Space," in *Proc. 9th IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, New York, USA, pages 75 – 82, June 2006.

26. H. L. Ribeiro and A. Gonzaga, "Hand Image Segmentation in Video Sequence by GMM: A Comparative Analysis," in *Proc. IEEE 19th Brazilian Symposium on Computer Graphics and Image Processing*, pp. 357 – 364, 2006.

27. Qin Wan and Yaonan Wang, "Background Subtraction Based on Adaptive Non-parametric Model," in *Proc IEEE 7th World Congress on Intelligent Control and Automation*, pp. 5960 – 5965, June 2008.

28. Shenz Zun-bing and Cui Xian-yu, "An Adaptive Learning Rate GMM for Background Extraction," *Optoelectronics Letters,* vol. 4, no. 6, pp. 460-463, Nov. 2008.

29. Julien Pilet, Christoph Strecha, and Pascal Fua, "Making Background Subtraction Robust to Sudden Illumination Changes," in *Proc. European Conference on Computer Vision, Marseille, France*, pp. 567-580, Oct. 2008.

30. Neil Robertson, Ian Reid, and Michael Brady, "Automatic Human Behavior Recognition and Explanation for CCTV Video Surveillance," *Security Journal*, vol. 21, pp. 173-188, July 2008.

31. By Sven Fleck and Wolfgang Straqer, "Smart Camera Based Monitoring System and its Application to Assisted Living," *Proceedings of IEEE*, vol. 96, no. 10, pp. 1698 – 1714, 2008.

32. Wanqing Li, Igor Kharitonenko, Serge Lichman, and Chaminda Weerasinghe, "A Prototype of Autonomous Intelligent Surveillance Cameras," in *Proc. IEEE International Conference on Video and Signal Based Surveillance*, pp. 101-107, 2006.

33. LI Yingjie and YIN Yixin, "Towards Suspicious Behavior Discovery in Video Surveillance System," in *Proc. IEEE 2nd International Workshop on Knowledge Discovery and Data Mining*, pp. 539 – 541, 2009.

34. Dima Damen and David Hogg, "Detecting Carried Objects in Short Video Sequences," in *Proc. of the 10th European Conference on Computer Vision: Part III*, Marseille, France, pp. 154 – 167, 2008.

35. Weiyao Lin, Ming-Ting Sun, Radha Poovandran, and Zhengyou Zhang, "Human Activity Recognition for Video Surveillance," in *Proc. IEEE International Symposium on Circuits and Systems*, pp. 2737 – 2740, May 2008.

36. Meghna Singh, Anup Basu, and Mrinal Kr. Mandal, "Human Activity Recognition Based on Silhouette Directionality," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 18, No. 9, pp.1280 – 1292, Sept. 2008.

37. Thanarat Horprasert, David Harwood, and Larry S. Davis, "A Robust Background Subtraction and Shadow Detection," in *Proc. Asian Conference on Computer Vision*, Taipei, Taiwan, pp. 983–988, 2000.

38. Jiang Dan and Yu Yuan, "A Multi-object Motion Tracking Method for Video Surveillance," in *Proc. IEEE 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, vol. 1, pp. 402 – 405, 2007.

39. Ying Fang, Huiyuan Wang, Shuang Mao, and Xiaojuan Wu, "Multi-Object Tracking Based on Region Corresponding and Improved Color-Histogram Matching," in *Proc. IEEE International Symposium on Signal Processing and Information Technology*, pp. 1 – 4, 2007.

40. Chiraz BenAbdelkader and Larry Davis, "Detection of People Carrying Objects: A Motion-Based Recognition Approach," in *Proc. IEEE 5th International Conference on Automatic Face and Gesture Recognition*, pp. 378 – 383, 2002.

41. Kenneth Ellingsen, "Salient Event Detection in Video Surveillance Scenarios," in *Proc. ACM First workshop on Analysis and Retrieval of Events/Actions and Workflows in Video Frames*, pp. 57-64 , 2008.

42. Hua Zhong, Jianbo Shi, and Mirk´o Visontai, "Detecting Unusual Activity in Video," in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 819 -826, 2004.

43. Gary Bradski and Adrian Kaehler, *Learning OpenCV*, First Edition, Sebastopol: O'Reilly Media, 2008.