Niraj Gadhe, Prof:Dr. B.K.Lande / IOSR Journal of Engineering (IOSRJEN)
www.iosrjen.org ISSN : 2250-3021
Vol. 2 Issue 2, Feb.2012, pp. 300-309

"A SURVEY OF CORE TECHNOLOGIES TO BUILD AN INTELLIGENT SURVEILLANCE SYSTEM"

Niraj Gadhe, Mtech-NIMS
VJTI-mumbai

Prof:Dr. B.K.Lande
Shah And Anchor Engineering College

Abstract: Public Security is a great concern past few years in INDIA as well as the World due to terrorist activities. It has been found that to enhance security a network of cameras is being deployed across the prominent places in the world. Such a traditional approach is costly as well as totally depends on physical and mental status of the observer looking at the camera data. In the light of such weak security infrastructure we propose a method to develop an intelligent system which not only will capture and feedback the video at real time but also model human behavior. To develop an Intelligent System for detecting , Modeling , Classification of human behavior using image processing and machine vision to detect abnormal malicious intents. We have identified the necessary steps and modules to achieve our proposed system. In the light of the study we have anticipated various algorithms and provide a survey of work done in the fields of image processing and video processing. We have identified important and core technologies as Image segmentation, Background removal, Object classification, motion tracking and behavior analysis and have presented each technology with its recent developments.

Index Terms : modeling , image processing, machine vision, image segmentation, background removal, tracking.

1 INTRODUCTION

Public Security is a great concern past few years in INDIA as well as the World due to terrorist activities. It has been found that to enhance security a network of cameras is being deployed across the prominent places in the world. The traditional system collects a large hours of video data and gives a live feed on multiple observation stations where a large amount of human resource is required to keep the vigilance on activities captured by each camera. Such a traditional approach is costly as well as totally depends on physical and mental status of the observer looking at the camera data.

In the light of such weak security infrastructure we propose a method to develop an intelligent system which not only will capture and feedback the video at real time but also model human behavior. We think of detecting , modeling and classifying human behavior using image processing to detect malicious behavior. Further we may maintain a database of suspects found over time and use face detection and recognition to identify the probable matches for the suspects.

To illustrate the scope and scale of large surveillance transit systems, consider the following examples. [19][20] Transit systems are spread through hundreds of kilometers and already require several tens of thousands of employees for daily operations. As the volume of video data increases, most existing digital video-surveillance systems provide the infrastructure only to capture, store, and distribute video while exclusively leaving the task of threat detection to human operators. Detecting specific activities in a live feed or searching in video archives (i.e., video analytics) almost completely relies on costly and scarce human resources.

The focus of this paper is to study and review existing algorithms and methods in field of image processing and machine vision to develop an intelligent surveillance system. The rest of the paper is organized as follows: Section 2 discusses important terms and core technologies in machine vision. Section 3 presents a survey of existing techniques to build an intelligent surveillance system. . A detailed review of present image processing algorithms and their advantages are discussed in Section 3. Section 4 we . Finally we conclude by anticipating some challenges in our work and its future scope in Section 6.propose an architechture of an intelligent surveillance system and enlist necessary modeules. Lastly we conclude with the future work.

2 Core technologies in machine vision:

Core technologies involved in analyzing human behavior are :[21] Motion detection typically refers to movement of objects .This pre-processing can be done via Background Subtraction and Temporal Differencing. A popular object segmentation strategy is **background subtraction**. Background subtraction compares an image with an estimate of the image as if it contained no objects of interest. It extracts foreground objects from regions where there is a significant difference between the observed and the estimated image. Common algorithm include

Stauffer and Grimson (adaptive Gaussian mixture model or GMM). In temporal differencing, video frames are separated by a constant time and compared to find regions that have changed. Unlike background subtraction, temporal differencing is based on local events with respect to time and does not use a model of the background to separate motion. Typically, two or three frames are used as separation time intervals, depending on the approach. A small time interval provides robustness to lighting conditions and complex backgrounds, since illumination changes and objects in the scene are more likely to be similar over short periods of time.

The next step in the behavior-recognition process is **object classification**. In general, for object classification in surveillance video, there are shape-based, motion-based, and feature-based classification methods.

Tracking which is next process is defined as the problem of estimating the trajectory of a pedestrian in the image plane while he is in the transit station or vehicle.. Selecting good features that can be used for future tracking or identification is a necessity, since the object's appearance in a later frame may vary due to orientation, scale, or other natural changes..Feature should be unique too. Some common features used in image-processing are color, edges, motion, and texture

3. Techniques to build an intelligent surveillance system

Literature survey will be mainly centered around acknowledging important works in the field of image and video processing related to our system. We will mainly study methods for ***Background subtraction, Segmentation, Object classification, motion Tracking of objects and classification of human behavior*** .

3.1 Background Subtraction

It is a problem on how to isolate objects or parts of objects from the rest of the image. In order to perform background subtraction, we first must "learn" a model of the background. Once learned, this *background model* is compared against the current image and then the known background parts are subtracted away. The objects left after subtraction are presumably new foreground objects..[21][22]

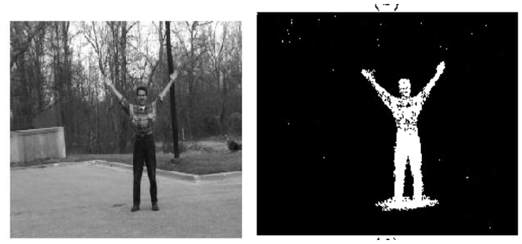

Fig 1. Background Subtraction

A) **Types of background subtraction**

Frame Differencing-Basic idea here is to subtract one frame from another subsequent frames and label any difference that is big enough (threshold) as Foreground . It catches mainly motion, simplest model and not suitable for complex scenes. **Average Background Method:**Here we learn the average and standard deviation of each pixel as its model of the background .First we accumulate images over time from a video. Secondly we collect frame to frame image differences over time.(compare means and deviations of each pixel values) Thirdly once the background has been learned we segment the image into foreground and background regions Lastly we compile segmentations from different color channels into a single mask image. **Advanced background method**: It is to compare a new value observed for a pixel with prior observed values. If the value is close to a prior value, then it is modeled as a perturbation on that color. If it is not close, then it can seed a new group of colors to be associated with that pixel. The result could be envisioned as a bunch of blobs floating in RGB space, each blob representing a separate volume considered likely to be background

B) **Advanced Background Subtraction Methods:**

1. **Gaussian models:**

Running Gaussian average was first proposed by Wren , Azarbayejani, Darell, pentland in 1997 .The main idea is to of actually fitting one Gaussian distribution (μ, σ) over the histogram . This was supposed to give the Background PDF. This Background PDF was to be updated from time to time for suppressing background changes. This update was given by running average as follows[3]

$$\mu_{t+1} = \alpha F_t + (1-\alpha)\mu_t$$
$$\sigma^2_{t+1} = \alpha(F_t - \mu_t)^2 + (1-\alpha)\sigma^2_t$$

Generalized Gaussian Family model is most representative background model. GGF model was proposed by H.Kim, R. Sakamoto et all to cope up with the background changes and shadows using this model. Background was modeled as pixel variation in a static scene over time with GGF distribution defined as follows[1].

$$p(x:\rho)=\frac{\rho\gamma}{2\Gamma(1/\rho)}\exp(-\gamma^{\rho}|x-\mu|^{\rho}) \text{ with } \gamma=\frac{1}{\sigma}\left(\frac{\Gamma(3/\rho)}{\Gamma(1/\rho)}\right)^{1/2} \quad (1)$$

where s a gamma function and is a variance of the distribution. In (1), represents a Gaussian distribution while represents a Laplace distribution.. The models for each pixel in the background are decided by calculating excess kurtosis g2 of the first m frames. Excess kurtosis measures whether the data are peaked or flat relative to a normal distribution and calculated using (2), where n is the number of samples and m is the mean. The excess kurtosis of Gaussian and Laplace distributions is 0 and 3, respectively.

$$g_2=\frac{n\sum_{i=1}^{n}(x_i-\mu)^4}{\left(\sum_{i=1}^{n}(x_i=\mu)^2\right)^2}-3 \quad (2)$$

Background subtraction is done as per follows. First, the initial region classification is performed by subtracting the intensity components of the current frame from the background model Classify the initial object region into three categories using two thresholds based on background subtraction BD, as in (3). LI and LB indicate the luminance components of pixel p in the current frame and the background model, respectively, and b is a scale parameter of the background model:

Then refine the suspicious regions from the initial classification by using a hue component because the shadow or lighting changes the colour property of the background much less than the luminance. Apply (3) to the hue component in a similar manner with a single parameter, K3, and classify the suspicious regions into the background and foreground regions. Thresholds K1–K3 are determined by training data with the following condition.

$$BD(p)=|L_I(p)-L_B(p)|$$

$$\begin{cases} BD(p)<K_1b(p) & \Rightarrow \text{background} \\ K_1b(p)\le BD(p)\le K_2b(p) & \Rightarrow \text{suspicious} \\ K_2b(p)\le BD(p) & \Rightarrow \text{foreground} \end{cases} \quad (3)$$

$$(K_1,K_2,K_3)=\underset{K_1,K_2,K_3}{\arg\min}\begin{pmatrix}\beta\times\text{false negative error}\\ +\text{false positive error}\end{pmatrix} \quad (4)$$

MOG model- Mixture of Gaussian :

In Mixture of Gaussian (MOG) background is termed as parametric frame of values where each pixel is represented with number of Gaussian Functions as Probability Distribution function as in equation 1 below.[2][18]

$$F(i_t=\mu)=\sum_{i=1}^{k}\omega_{i,t}\cdot\eta(\mu,\sigma) \qquad 1$$

where,

η = the *i-th* Gaussian component

μ = intensity mean

σ = standard deviation

$\omega_{i,t}$ = portion of the data accounted by *i-th* component

In addition, for MoG approach, only pixel that is within a scaling factor of background standard deviation is considered as part of background. This can be determined by comparing the pixel value with Gaussian component tracking .Comparitive study was carried out by Shahrizat Mohammad et all [2] by implementing a project on comparison of performances shown by MOG model on varying different parameters of the Gaussian model. The main parameters considered were as follows.

i. T_s = Background component weight threshold
ii. D = Standard deviation scaling factor
ii. ρ = Learning rate
iv. K = Total number of Gaussian components
v. M = Maximum number of components M in the background model

The parametric values of one component were changed gradually and others were kept constant. Output and performance for all cases were observed and compared. Different combinations showed different refinements. Ultimately the conclusion was given as per following table in the work done .[2][18]

2. Kalman Filters

The basic idea is that, under a strong but reasonable set of assumptions, it will be possible—given a history of measurements of a system—to build a model for the state of the system that maximizes the a posteriori† probability of those previous measurements. [22][21]. The internal state of the system is described by the background intensity Bt and its temporal derivative B't, which are recursively updated as follows: [8]

$$\begin{bmatrix}B_t\\B'_t\end{bmatrix}=\mathbf{A}\cdot\begin{bmatrix}B_{t-1}\\B'_{t-1}\end{bmatrix}+\mathbf{K_t}\cdot\left(I_t-\mathbf{H}\cdot\mathbf{A}\cdot\begin{bmatrix}B_{t-1}\\B'_{t-1}\end{bmatrix}\right)$$

Matrix A describes the background dynamics and H is the measurement matrix. Their particular values are as follows:

$$\mathbf{A} = \begin{bmatrix} 1 & 0.7 \\ 0 & 0.7 \end{bmatrix}, \quad \mathbf{H} = [1 \;\; 0]$$

The Kalman gain matrix Kt switches between a slow adaptation rate α1 and a fast adaptation rate α2 > α1 based on whether It-1 is a foreground pixel:

$$\mathrm{K_t} = \begin{bmatrix} \alpha_1 \\ \alpha_1 \end{bmatrix} \text{if } I_{t-1} \text{ is foreground, and} \begin{bmatrix} \alpha_2 \\ \alpha_2 \end{bmatrix} \text{otherwise.}$$

A work done by Jesse scott, Michael A. Pusaten, et all in Video background estimation used Kalman filter with one dimension.[6] The main algorithm was divided into two steps - Mean Intensity update and Standard deviation update. In these equations the following variables need defined: **A** matrix relates the state at the previous time step to the state at the current step. **B** matrix relates the optional control input to the state. **Q** represents the process noise covariance matrix. **H** matrix relates the state to the measurement. **R** represents the measurement noise covariance matrix. **x** is the state variable with k being the current and k-1 being the prior. **u** is the control variable with k being the current. **z** is the measurement with k being the current

$$\hat{x}_k^- = A\hat{x}_{k-1} + Bu_k \quad (1)$$

$$P_k^- = AP_{k-1}A^T + Q \quad (2)$$

$$K_k = P_k^- H^T (HP_k^- H^T + R)^{-1} \quad (3)$$

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (4)$$

$$P_k = (I - K_k H)P_k^- \quad (5)$$

- **Algorithms is defined as follows:**

1. Initialize variables and background estimates.
2. Aquire current frame.
3. Execute Kalman filter update equations.
4. Subtract current frame from background.
5. Generate binary mask from difference image .
6. Group neighbouring foreground pixels.
7. Draw box around each group in current frame.

The update equations (6) and (7) are derived from the fundamental versions presented in (1) through (5). The mean update (6) and standard deviation update (7) equations utilize variables that control certain aspects of the original equations. The following is a list of relations between variable matrices in the fundamental equations to variables in the derived update equations. The **Q** matrix is related to the d variable that is the process noise gain. The R matrix is related to the s variable that is the measurement noise gain. The variable g is the ratio of **(s2/d2)** used as a ratiometric noise gain.

$$\mu_{ij}[k] = \frac{(g_{ij} * \mu_{ij}[k-1]) + \left(\frac{z_{ij}[k]}{d_{ij}} * \sigma_{ij}[k-1]\right)}{(g_{ij} + \sigma_{ij}[k-1])} \quad (6)$$

$$\sigma_{ij}[k] = \frac{(g_{ij} * \sigma_{ij}[k-1])}{(g_{ij} + \sigma_{ij}[k-1])} \quad (7)$$

3. **Bayesian Background Modeling**

In probability theory and applications, **Bayes' theorem** (alternatively **Bayes' law** or **Bayes' rule**) relates the conditional probabilities *P*(*A* | *B*) and *P*(*B* | *A*). It is commonly used in science and engineering. The theorem is named for Thomas Bayes [22]

- **Simple form:**

For events *A* and *B*, provided that

$$P(B) \neq 0$$

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)}.$$

In a Bayesian inference step, the probability of evidence *B* is constant for all models A_n. The posterior may then be expressed as proportional to the numerator:

$$P(A_n|B) \propto P(B|A_n)P(A_n).$$

- **Steps for Bayesian Learning:**

The following steps are performed for each observation made at a particular spatial pixel Location.[9]

Draw N samples each from all the prior distributions of the K Means. Let us call the obtained samples as

$$\{\mu_{11}, \mu_{12}, \ldots, \mu_{1N}\}, \{\mu_{21}, \mu_{22}, \ldots, \mu_{2N}\}, \ldots, \{\mu_{K1}, \mu_{K2}, \ldots, \mu_{KN}\}.$$

When a pixel value x is observed at the particular location, compute the sum of likelihoods for each Mean distribution, given that observation

$$L_r = \sum_{i=1}^{N} l(\mu_{ri}; \mathbf{x}), \qquad r = 1, 2, \ldots, K.$$

The likelihood of each Mean sample mri is calculated as the probability of observing x in a Gaussian distribution centered at µri, with covariance matrix ΣM.

$$l(\mu_{ri}; \mathbf{X}) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma_M|^{\frac{1}{2}}} \cdot e^{-\frac{1}{2}(\mathbf{X}-\mu_{ri})' \Sigma_M^{-1} (\mathbf{X}-\mu_{ri})}$$

The model variance, ΣM. can be thought of as a parameter to control the sensitivity of the system. Its effect on clustering is described later. The next step is to determine which cluster the pixel observation belongs to. The observation would belong to the cluster having the highest sum of likelihoods value Lr. The prior distribution of the Mean of this cluster is updated to obtain a posterior distribution using step

4. The distributions of the Means of the other clusters are left unchanged The steps to update a prior distribution to a posterior one are:

(a) If the rth distribution is to be updated, compute weights qi for each sample µri of the prior distribtion as follows:

$$q_i = \frac{l(\mu_{ri};\mathbf{x})}{L_r}, \quad i = 1,2,\ldots,N$$

(b){ µr1, µr2……. µrN } are then resampled using the weighted bootstrap method with weights {q1;q2; :::;qN} to obtain samples from the posterior distribution of µr which are

$$\{\mu_{r1}^*, \mu_{r2}^*, \ldots, \mu_{rN}^*\}$$

Then the next pixel observation is made, the posterior samples become the prior samples of the rth cluster Mean. Steps 1 through 5 are repeated for every observation of the pixel process. It is important to note that this entire process is done just for one pixel process.

- **Classification of frames using above method:**

The first few initial frames in a video are termed as learning frames.. They are used to build the distributions of cluster means using above process.Classification is done for subsequent frames as below. Typically, in a video sequence involving moving objects, at a particular spatial pixel position a majority of the pixel observations would correspond to the background. Therefore, background clusters would typically account for much more observations than the foreground clusters. This means that the prior weight (ω) of any background cluster would be higher than that of a foreground cluster. The clusters are ordered based on their prior weight. [18][5]

Based on a certain threshold Th, the first B clusters are chosen as background clusters, where

$$B = argmin_b \left(\sum_{k=1}^{b} \omega_k > Th \right)$$

Th is a measure of the minimum portion of the data that should be accounted for by the background.The sum of likelihoods **(Lr)** is used to determine the cluster to which the observed pixel belongs. If this cluster is not one of the first B clusters as described above, the pixel would be a foreground pixel. The classification process is only performed on the pixel locations inside the regions of motion in the current frame. All pixel locations in the current frame outside the regions of motion are classified as background.

3.2 Segmentation Algorithms

Image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics. [22][21][4]. **Thresholding** is a method is based on a threshold value to turn a gray-scale image into a binary image. Several popular methods are used in industry including the maximum entropy method, Otsu's method (maximum variance), and et al. k-means clustering can also be used. **Clustering** methods is another important method.The K-means algorithm is an iterative technique that is used to partition an image into *K* clusters. The basic k means algorithm is: (1956)

(1) Pick *K* cluster centers, either randomly or based on some heuristic
(2) .Assign each pixel in the image to the cluster that minimizes the distance between the pixel and the cluster center
(3) .Re-compute the cluster centers by averaging all of the pixels in the cluster
(4) Repeat steps 2 and 3 until convergence is attained (e.g. no pixels change clusters)

Distance is the squared or absolute difference between a pixel and a cluster center. The difference is typically based on pixel color, intensity, texture, and location, or a weighted combination of these factors. *K* can be selected manually, randomly, or by a heuristic. The k-means clustering was invented in 1956. The most common form of the algorithm uses an iterative refinement heuristic known as Lloyd's algorithm.[22]. **Split-and-merge methods** is based on a quadtree partition of an image. This method starts at the root of the tree that represents the whole image. If it is found non-uniform then it is split into four son-squares (the splitting process), and so on so forth. Conversely, if four son-squares are homogeneous, they can be merged as several connected components (the merging process). The node in the tree is a segmented node. This process continues recursively until no further splits or merges are possible.

A) Advanced Segmentation Algorithms

a) Watershed algorithm:

This algorithm converts lines in an image into "mountains" and uniform regions into "valleys" that can be used to help segment objects. The watershed algorithm first takes the gradient of the intensity image; this has the effect of forming valleys or *basins* (the low points) where there is no texture and of forming mountains or *ranges* (high ridges corresponding to edges) where there are dominant lines in the image. It then successively floods basins starting from userspecified (or algorithm-specifi ed) points until these regions meet. Regions that merge across the marks so generated are segmented as belonging together as the image "fills up". In this way, the basins connected to the marker point become "owned" by that marker. We then segment the image into the corresponding marked regions **[21][19]**

- **Meyer's watershed flooding algorithm**

(1) .A set of markers, pixels where the flooding shall start, are chosen. Each is given a different label.
(2) The neighboring pixels of each marked area are inserted into a priority queue with a priority level corresponding to the gray level of the pixel.
(3) The pixel with the highest priority level is extracted from the priority queue. If the neighbors of the extracted pixel that have already been labeled all have the same label, then the pixel is labeled with their label. All non-marked neighbors that are not yet in the priority queue are put into the priority queue.
(4) Redo step 3 until the priority queue is empty.

The non-labeled pixels are the watershed lines.[21][22]

b) Mean shift segmentation

Mean shift finds the peak of a color-spatial (or other feature) distribution over time or over space. Given a set of multidimensional data points whose dimensions are (*x*, *y*, blue, green, red), mean shift can find the highest density "clumps" of data in this space by scanning a *window* over the space. However, that the spatial variables (*x*, *y*) can have very different ranges from the color magnitude ranges (blue, green, red). [21][7].Therefore, mean shift needs to allow for different window radii in different dimensions. In this case we should have one radius for the spatial variables (spatialRadius) and one radius for the color magnitudes (colorRadius). As mean-shift windows move, all the points traversed by the windows that converge at a peak in the data become connected or "owned" by that peak. This ownership, radiating out from the densest peaks, forms the segmentation of the image.

- **Algorithm runs as follows.**

(1) Choose a search window:
its initial location;
its type (uniform, polynomial, exponential, or Gaussian);
its shape (symmetric or skewed, possibly rotated, rounded or rectangular);
its size (extent at which it rolls off or is cut off).
(2) Compute the window's (possibly weighted) center of mass.
(3) Center the window at the center of mass.
(4) Return to step 2 until the window stops moving.

3.3 Classification of objects and Motion Tracking

A) Classification based on shape using shape filters

- **Distance set shape filters**

(1) Let S1 be set of feature points extracted from an image of reference object .
(2) Let S2 be set of feature points extracted from an image called the test image.
(3) Typically Set S2 will be large and complex as it represent a complex scene bigger than the reference object.
|S2|> |S1|
(4) If the test image contains reference object the set S2 will have a subset S2' which is isomorphic (having one to one mapping) to set S1.
(5) Let pϵS1 and qϵS2' be two counterpart points. The distance set DSS1,N1(p) of p to its neighbors in S1 is identical with the distance set DSS2, N1 (q) of q to its N1 nearest neighbors in S2'.
(6) This property can be used to identify for objects in a test image similar to our reference object.
(7) Similar to a band-pass filter, which will retain only signal components within a certain frequency band, the proposed filter will allow to pass only those points of which, regarding their distance sets, are similar to points of , and will filter out all other points. In analogy to a band-pass filter, we will call this filter the ***distance set shape filter***.[15]

B) Object Recognition and Reconstruction based on Shape knowledge and knowledge repository

Target objects in the scene are countable and their shapes could be predefined as a repository of shape knowledge. When an object of interest in the scene is selected, an object recognition process could be carried out with the assistance of the knowledge repository. [16]

3.4 Human Behavior recognition Algorithms

Case study of Video and Image Retrieval Analysis Tool (VIRAT, 2008) is considered to study these algorithms. . VIRAT's purpose is to develop and demonstrate a system for UAV video data exploitation. For organizational purposes, transit surveillance operationally relevant behaviors are divided into four general groups:[20][21][22]

A) Single Person or no interaction consist of following subcategories:

Loitering is defined as the presence of an individual in an area for a period of time longer than a given time threshold. The technique proposed in uses a refined GMM subtraction algorithm to detect motion blobs in a calibrated scene. Blobs are classified as humans using size and shape descriptors, and a short-term biometric based on the color of clothing is used for tracking purposes. A suspicious activity is defined as individuals loitering, using a time threshold longer than the maximum time that it would typically take to catch a bus.(Fig6)

B) Multiple-Person Interactions

This behavior detection process consists of foreground segmentation, blob detection, and tracking. Semantic descriptions of suspicious human behaviors are defined through groups of low-level blob based events. For example, fights are defined as many blobs' centroid moving together, merging and splitting, and overall fast changes in the blobs' characteristics.

Fig 6: Sample single-person or no interaction behavior. Suspicious person (marked with an ellipse) loitering for a long period of time without leaving in a bus.

Attacks are defined as one blob getting too close to another blob, with one blob perhaps being initially static, and one blob erratically moving apart. Algorithms include the use of a nearest neighbor classifier based on trajectory information to detect human interactions such as walking together, approaching, ignoring, meeting, splitting, and fighting.Bayesian networks and moment-invariant feature descriptions to detect events, including sitting down, standing up, bending over, getting up, walking, hugging, bending sideways, squatting, rising from a squatting position, falling down, jumping, punching, and kicking.

C) Person -Vehicle Interaction

Most existing automated vehicle surveillance systems are based on trajectory analysis. Detected events are U-turns, sudden brake, and pedestrians trespassing the street or a small group of predefined events, such as accidents , illegal parking , congestion status , illegal turns, or lane driving .Common approaches to trajectory analysis are based on Kalman filter. (Fig8)

D) Person–Facility/Location Interactions

Intrusion or trespassing is defined as the presence of people in a forbidden area. A forbidden area can also be defined in terms of time or spatial relationships A large number of intrusion detection algorithms rely on the use of a digital "trip wire." A trip wire is typically a line drawn over the image, which separates regions into "allow" and "do not allow" areas. whenever a bottom corner of the bounding rectangle of an object intersects this line (rails in a subway), an intrusion is detected, and a warning is given. **Object Stationarity (Object Removal and Object Left Behind):** here Most algorithms presented a simple background subtraction method to find stationary objects that were not present before. Many other methods have been proposed to deal with objects left behind or removed. In an edge matching algorithm is used, which compares the current frame to the background model to detect objects removed or left behind. (Fig 9)

1. Proposed Syetem architechture.

We propose a system To develop an Intelligent System for detecting , Modeling , Classification of human behavior using image processing and machine vision to detect abnormal malicious intents. We are aiming to develop a system that would not only do the work done by traditional surveillance system but also help observer in detecting behavior patterns . So we are trying to develop a software based on image and video processing algorithms that we help us detect and classify human behavior. As detecting a human behavior from a scene we need to follow some sequential steps in our processing of the video. Fig 10

Fig8 : Person vehicle interaction

Fig 9. Sample person–facility/location interaction. Object left behind in a train station

Technology used will be OpenCV [OpenCV] is an open source (see *http://opensource.org*) computer vision library available from *http://SourceForge.net/projects/opencvlibrary*. Th e library is written in C and C++ and runs under Linux, Windows and Mac OS X. Th ere is active development on interfaces for Python, Ruby, Matlab, and other languages. OpenCV was designed for computational efficiency and with a strong focus on realtime applications. OpenCV is written in optimized C and can take advantage of multicore processors. If you desire further automatic optimization on Intel architectures. These sequential steps can be generalized and explained in brief as follows .We need to understand the behavior of system first. The real time module consists of all routines that run during real time along with capturing the video. Such modules will be time intensive and fast in nature and simple in nature.

a) Frame detection module.

This routine will detect each frame at real time from a video sequence and present its status in form of matrix, its frame no, pixels, color content and sample rate, etc. As we get features of the each frame we can apply various processing algorithms to these frames using image processing algorithms. Mainly due to durability and simplicity of its use we will be opting for Gaussian models for background subtraction.

b) Background Removal and segmentation module:

This will detect and remove background features to detect the object in the scene . Algorithms like GMM based will be a tool for such removal. We are interested in analyzing human behavior. So we need to detect an object in the scene which corresponds to a human figure. For this we need to remove unwanted background and separate foreground data containing moving objects. Then we apply segmentation algorithms to separate moving objects from background.

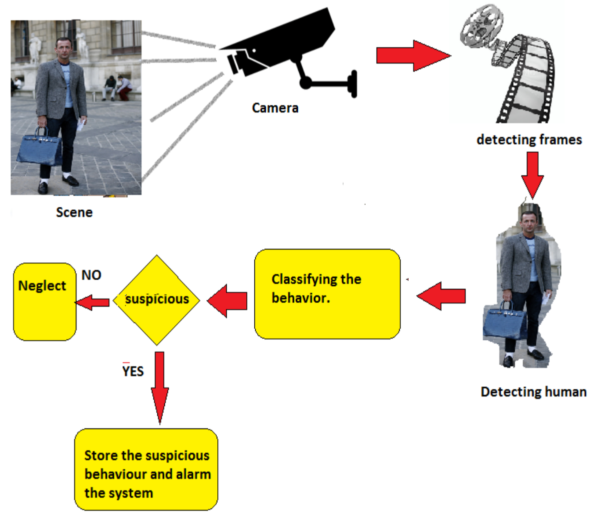

Fig:10.Generalised architechture of our system.

c) Detecting human and their behavior module:

As our earlier module will give us a frame containing moving objects now we need to detect what type of object it is. We need to classify it as human or non human. Non human objects are to be deleted and human objects are to be maintained in frames. Once we get a human object in frame we can track the motion of such object in subsequent frames. We will have a training given to system to detect some human behaviors of malicious intent. We will take a simple real world problem like, *Loitering:* Standing or roaming at a place more than required and for more time than a preset threshold.*Trespassing or intrusion:* Detect events of trespassing as well attempts to do so depending on region boundaries and crossing of such boundaries in the scene.*Leaving an object at a place and flee*: To detect a human leaving an object at a Place and not taking its possession for a preset threshold. We can use shape based repository for this purpose.

d) Capturing frames containing suspicious activities module:

Capturing frames containing probable suspicious behavior , classifying them and maintaining it for further use and alarming the system .

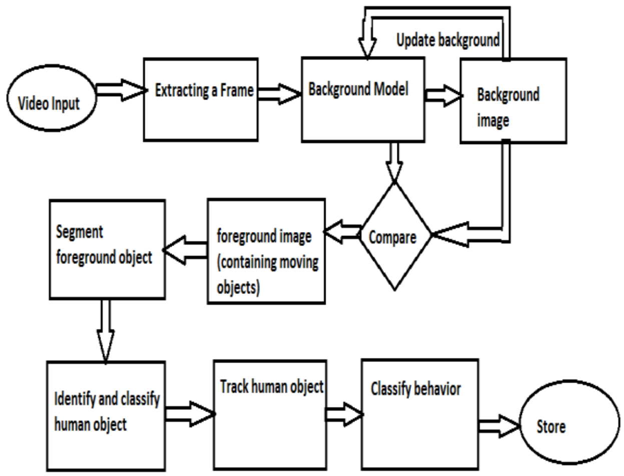

Fig 10: Architecture of proposed system.

CONCLUSION

Thus we have successfully defined our problem statement to develop an Intelligent System for detecting , Modeling , Classification of human behavior using image processing and machine vision to detect abnormal malicious intents. We have identified the necessary steps and modules to achieve our proposed system. In the light of the study we have anticipated various algorithms and work done in the fields of image processing and video processing. We have identified important and core technologies as Image segmentation, Background removal, Object classification, motion tracking and behavior analysis and have studied each technology with its recent developments. This algorithms and their study will further benefit us in developing our system as a whole.

ACKNOWLEDGMENTS

I would like to thank all those people whose support and cooperation has been an invaluable asset during the course of this paper. We would also like to thank our Guide **Prof. Dr. B.K.Lande** for guiding me throughout this paper and giving it the present shape. It would have been impossible to complete the paper without their support, valuable suggestions, criticism, encouragement and guidance.I am also grateful for all other teaching and non-teaching staff members of the Computer Technology for directly or indirectly helping us for the completion of this seminar and the resources provided.

Future work

Future work will mainly be focused on developing and implementing an intelligent surveillance system using the methods we have learned. Use of proper methods , with technology considering domain of application should be done to implement intelligence. The fusion of the various machine vision techniques to build a real time , fast and stable intelligent system is yet a dream and should be achieved.

REFERENCES

[1] H. Kim, R. Sakamoto, I. Kitahara, T. Toriyama and K. Kogure “Background subtraction using generalized Gaussian family model” in ELECTRONICS LETTERS 31st January 2008 Vol. 44 No. 3

[2] Shahrizat Shaik Mohamed, Nooritawati Md Tahir, Ramli Adnan “Background Modelling and Background Subtraction Performance for Object Detection” in 2010 6th International Colloquium on Signal Processing & Its Applications (CSPA)

[3] Dar-Shyang Lee, Member, IEEE “Effective Gaussian Mixture Learning for Video Background Subtraction “ in IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 27, NO. 5, MAY 2005

[4] Michael E. Farmer, Member, IEEE, and Anil K. Jain, Fellow, IEEE “A Wrapper-Based Approach to Image Segmentation and Classification” in IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 14, NO. 12, DECEMBER 2005

[5] A. Colmenarejo, M. Escudero-Vin˜olo and J. Besco´s “Class-driven Bayesian background modelling for video object segmentation” in ELECTRONICS LETTERS 1st September 2011 Vol. 47 No. 18

[6] Jesse Scott et all “Kalman Filter Based Video Background Estimation” in 2007IEEE

[7] Madhurima , et all “Object tracking in a video sequence using Mean-Shift Based Approach” in IJCEM International Journal of Computational Engineering & Management, Vol. 11, January 2011

[8] Sen-Ching S. Cheung and Chandrika Kamath “Robust techniques for background subtraction in urban traffic video”

[9] Padmini Jaikumar, Abhishek Singh and Suman K Mitra “Background Subtraction in Videos usingBayesian Learning with Motion Information” in 2008.

[10] Scott Helmer and David G. Lowe "Object Class Recognition with Many Local Features" in Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops.

[11] Lun Zhang, Stan Z. Li, Xiaotong Yuan and Shiming Xiang "Real-time Object Classification in Video Surveillance Based on Appearance Learning" in 2007 IEEE

[12] Tie Liu, Zejian Yuan, Jian Sun, Jingdong Wang, Nanning Zheng, Fellow, IEEE,et all "Learning to Detect a Salient Object" in IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 33, NO. 2, FEBRUARY 2011

[13] Zhiyong Wang, Zheru Chi and David Feng "STRUCTURAL REPRESENTATION AND BPTS LEARNING FOR SHAPE CLASSIFICATION" in Proceedings of the 9th International Conference on Neural Information Processing (ICONIP'O2), Vol. 1

[14] Cosmin Grigorescu, Student Member, IEEE, and Nicolai Petkov "Distance Sets for Shape Filters and Shape Recognition" in IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 12, NO. 10, OCTOBER 2003

[16] Chensheng Wang, Fei Wang "A Knowledge-based Strategy for Object Recognition and Reconstruction" in 2009 International Conference on Information Technology and Computer Science

[17] Junqiu Wang, Member, IEEE, and Yasushi Yagi, Member, IEEE "Integrating Color and Shape-Texture Features for Adaptive Real-Time Object Tracking" in IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 17, NO. 2, FEBRUARY 2008

[18] Zoran Zivkovic "Improved Adaptive Gaussian Mixture Model for Background Subtraction" in Proc. ICPR, 2004

[19] Margarita Osadchy and Daniel Keren "A Rejection-Based Method for Event Detection in Video" in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 14, NO. 4, APRIL 2004

[20] Joshua Candamo, Matthew Shreve, Member, IEEE, Dmitry B. Goldgof, Fellow, IEEE, Deborah B. Sapper, and Rangachar Kasturi, Fellow, IEEE "Understanding Transit Scenes: A Survey on Human Behavior-Recognition Algorithms" in IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 11, NO. 1, MARCH 2010

[21] Book on " Learning Open Cv – Computer vision with Open CV library" by Gary Bradski and Adrian Kaebler – o'reilly

[22] www.wikipedia.org