A Multiscale Parametric Background Model for Stationary Foreground Object Detection

Steven Cheng[1], Xingzhi Luo[2] and Suchendra M. Bhandarkar[1,2]
[1]Artificial Intelligence Center, [2]Department of Computer Science
The University of Georgia, Athens, Georgia 30602, USA
canada@uga.edu, xingzhi@cs.uga.edu, suchi@cs.uga.edu

Abstract

Detection of stationary foreground *objects within a dynamic scene is one of the goals of a video surveillance system. A parametric background maintenance and updating scheme, based on a multiple Gaussian mixture model that operates on multiple time scales, is proposed. Each color cluster in the proposed model is assigned a weight which measures the time duration and temporal recurrence frequency of the cluster. Sudden illumination changes are handled by using an adaptive histogram template whereas gradual illumination changes are automatically resolved with the adaptive background model. Stationary foreground objects are detected by maintaining their temporal history in the dynamic scene at multiple time scales. Experimental results show that the proposed scheme performs well in three distinct real-world settings.*

1. Introduction

The detection of stationary *foreground* objects is one of the primary objectives of many security-based video surveillance systems installed in train stations, airports, and other highly-populated locations. Immediate detection of suspicious packages or objects is vital to the safety of innocent citizens in the current age of terrorists who often use primitive home-made explosive devices. In a more mundane setting, the detection of a stationary foreground object could be used to signal incidents of forgotten luggage at transportation hubs or, illegally parked vehicles or disabled vehicles on roadsides. This paper proposes a multiscale parametric background model for the detection of stationary foreground objects in a dynamic scene.

Background modeling is the process of extracting the background from an image in order to analyze the actions and behaviors of foreground objects. Two common obstacles encountered during background extraction are changes in illumination and the *sleeping person problem* [12]. Gradual illumination changes are generally well handled by adaptive background modeling schemes, whereas sudden and transient illumination changes need special consideration since they can drastically alter the color distribution associated with the background image. The other obstacle, commonly known as the *sleeping person problem*, is encountered when a foreground object remains stationary in the scene for an extended period of time. An adaptive background model will eventually cause the stationary foreground object to merge into the background image. The time-to-merge is dependent on the temporal scale of adaptation of the background model. Once a stationary foreground object is merged with the background image, it is no longer detectable in the foreground via background subtraction. In this paper, stationary foreground objects in the dynamic scene are detected by having the background model adapt to scene changes at multiple temporal scales. Sudden illumination changes are handled by using an adaptive histogram template whereas gradual illumination changes are automatically resolved with the adaptive background model.

2. Background Modeling

There are two common approaches to background modeling of real-time dynamic scenes: parametric approaches and nonparametric approaches. Parametric approaches assume that the pixel intensity or pixel color can be modeled using a probability distribution with a known parametric form. The Gaussian distribution, with the mean and variance as its defining parameters, is a popular choice. Background models that use a single Gaussian distribution to model the pixel color/intensity typically estimate the model parameters via temporal averaging [6] or temporal nonlinear filtering [4], [9], [8]. Single Gaussian distribution background models are not viable for complex dynamic scenes where the background pixels often exhibit a multi-

IEEE Workshop on Motion and Video Computing (WMVC'07)
0-7695-2793-0/07 $20.00 © 2007 IEEE

modal color distribution [11]. The Multiple Gaussian Mixture (MGM) background model [11] addresses this problem by modeling the color distribution at each pixel by a set of Gaussian distributions. The MGM background model and its computationally efficient variants [1], [2], [5], [7] have been observed to perform well under varying illumination conditions and in both, indoor and outdoor environments.

In a nonparametric background model, the background pixel samples are directly used to represent the background color distribution [3]. The nonparametric model can be made more robust by integrating temporal and spatial information [10]. The incorporation of a foreground model and a Markov Random Field (MRF) to impose spatial constraints on the background and foreground, have been observed to reduce the effects of random noise [13]. Although nonparametric models have been observed to perform better than their parametric counterparts, especially when dealing with dynamic backgrounds, their high computational complexity renders them difficult to implement in situations where real-time performance is needed. Also, nonparametric models do not easily generalize to multiple time scales and consequently do not provide an easy solution to the sleeping person problem. The parametric MGM background model, on the other hand, is shown to have a natural multiscale extension which makes possible the detection of stationary foreground objects by solving the sleeping person problem.

2.1. Background Image Assumptions

In order to distinguish the background from moving foreground objects, two assumptions about the dynamic scene are made. The first assumption is that background pixel values tend to persist for longer time durations than pixel values of foreground objects. This assumption is naturally true for scenes involving a few fast-moving objects. The second assumption states that background pixel values recur more frequently than foreground pixel values. This situation occurs when foreground objects periodically occlude the background image pixels in a busy scene. Since the proposed background model incorporates both assumptions within its pixel weighting scheme, it permits the modeling of both, busy and non-busy scenes.

2.2. The MGM Background Model

The version of the MGM background model used in this paper is the one proposed in [1], [7]. The MGM background model incorporates k Gaussian distributions for each pixel of the image $I_{x,y}$. Since all terms in the MGM background model are assumed to be associated with a particular pixel, the x and y subscripts are omitted. Each distribution or color cluster, χ_i is characterized by the following attributes:

1. μ_i: Cluster mean.
2. σ_i^2: Cluster variance.
3. N_i: Cluster weight.
4. tl_i: Time that χ_i was last updated.
5. n_i: Number of image intensity values that have matched cluster χ_i in the current time interval/slice.

The background updating process employs a two-stage approach. The initial stage tallies the statistical properties of the pixel value for each of the k distributions. The image intensity I_t in frame t is compared to each of the k color clusters. I_t is deemed to match cluster χ_i if $\mu_i - 2.5\sigma_i \leq I_t \leq \mu_i + 2.5\sigma_i$. A match results in the updating of μ_i and σ_i^2 using equations (1) and (2), respectively.

$$\mu_i = \mu_i + \frac{1}{L}(I_t - \mu_i) \tag{1}$$

$$\sigma_i^2 = \sigma_i^2 + \frac{1}{L}[(I_t - \mu_i)^2 - \sigma_i^2] \tag{2}$$

where L is an integer representing the inverse of the learning rate. If I_t does not match any of the clusters, then the cluster with the lowest weight N is replaced with a new cluster χ_j where $\mu_j = I_t$, $\sigma_j^2 = \sigma_0^2$, $N_j = 1$, $n_j = 1$, and $tl_j = t$. The variance σ_0^2 is initialized to a high value since it is assigned to a newly created cluster.

The second phase of the background updating model occurs once every F frames and involves updating the cluster weights. Each color cluster is assigned a weight N_i representing the likelihood that χ_i corresponds to the actual background pixel value. N_i takes into account both, the temporal persistence of a certain color and its temporal recurrence frequency at a pixel location. The temporal history of the pixel color is maintained by dividing the time axis into discrete intervals or slices. At the end of each time slice F, the cluster weights are updated according to the time duration and temporal recurrence frequency weight update rules described by equations (3) and (4), respectively. The temporal recurrence frequency affects the cluster weight only if the cluster has been sufficiently represented (i.e. $n_i > \delta$, where δ is the recurrence frequency threshold) during the current time slice.

$$N_i = N_i + \begin{cases} F, & \text{if } n_i > F/2 \\ n_i, & \text{otherwise} \end{cases} \tag{3}$$

$$N_i = \begin{cases} N_i + F/2, & \text{if } n_i > \delta \text{ and } t - tl_i > 2F \\ N_i, & \text{otherwise} \end{cases} \tag{4}$$

All clusters satisfying the condition $N_i > N_{max}/3$ where $N_{max} = \max(N_i)$, are deemed to represent a background pixel value. Once the background pixels have been determined for the current frame, if $N_{max} > 1.25\Delta$ (where

Algorithm 1 Background Model Updating

1. If I_t matches χ_i, then update μ_i and σ_i^2 using equations (1) and (2). Set $n_i = n_i + 1$.
2. If I_t does not match χ_i, then replace the cluster with the lowest N value with a new cluster initialized with $\mu = I_t$, $\sigma^2 = \sigma_0^2$, $N = 1$, $n = 1$, and $tl = t$.
3. If $t = mF$ for all integers m:
   (a) Check duration and update N_i:
      i. If $n_i > F/2$, then $N_i = N_i + F$.
      ii. Otherwise, $N_i = N_i + n_i$.
   (b) Check recurrence frequency and update N_i:
      i. If $n_i > \delta$ and $t - tl_i > 2F$, then $N_i = N_i + F/2$ and $tl_i = t$.
      ii. Otherwise, $N_i = N_i$.
   (c) If $N_i > N_{max}/3$ then χ_i is considered to belong to the background.
   (d) Set $n_i = 0$ for all χ_i.
   (e) If $N_{max} > 1.25\Delta$, then $N_i = N_i * 4/5$ for all χ_i.
   (f) Delete any χ_i where $N_i = 0$ or $t - tl_i > \Delta$

Δ is a prespecified upper bound on the cluster size), then each N_i is scaled by a factor of $4/5$. This is necessary because if N is unbounded, then a cluster with a relatively large value of N would dominate the updating scheme, making it difficult for other clusters to be considered as part of the background. Finally, any cluster with $N = 0$ or $t - tl > \Delta$ is deleted. The full background model updating procedure is described in Algorithm 1.

2.3. Multiscale MGM Background Model for Detecting Static Foreground Objects

In the proposed MGM background model, a cluster with $N_i > N_{max}/3$ is considered as a background pixel value. Consequently, the parameter Δ influences the amount of time required for a static object to merge into the background due to its relationship with N_{max}. For the purpose of measuring the duration that an object remains stationary, N_{max} can be approximated as Δ. This assumption is valid in the ideal case where the background model has been sufficiently trained with the actual background image.

Let B_t be the background image where $\Delta = 3 \times t \times fr$ and fr is the frame rate. The parameter t represents the approximate time (in seconds) at which a static foreground object will merge into the background image. Thus the parameter Δ represents a temporal scale for adaptation of the model to stationary foreground objects in the dynamic scene. Background models with a larger value of Δ (i.e., at a coarser scale) will allow static foreground objects to persist in the foreground for a longer period of time whereas background models with a smaller value of Δ (i.e., at a finer scale) will permit the static foreground objects to merge with the background earlier. Thus, the background image B_t includes all objects that have been motionless for a period $T_s > t$ where $T_s \approx N_i/fr$. It can be shown that if $i \leq j$, then $B_j \subseteq B_i$. In other words, objects in the coarser-scale background image will also be present in the finer-scale background image. This is consistent with general scale space theory where image features detected at a coarser scale are expected to persist at a finer scale.

A system that maintains an MGM-based background model at two different scales, resulting in two background images B_i and B_j where $i < j$, is able to detect foreground objects that have been static for greater than i seconds but not more than j seconds (i.e. $i < T_s \leq j$) because after i seconds, the static foreground object will merge into the background image B_i, but persist in the foreground of B_j. The location of the static foreground object can be determined by generating a difference background image $DB_{i,j} = B_i - B_j$.

3. Experimental Results

The multiscale background updating model was tested on three unique video sequences. Two scenarios are situated in an indoor environment: a living room and a train terminal[1] while the last example takes place in an outdoor no-parking zone. In all experiments, the sampling rate fr is set at $30fps$ and the four model parameters are set at $k = 4$, $L = 1024$, $F = 60$, and $\delta = 20$. In other words, four Gaussian distributions are used for each pixel, the learning rate is set to $1/1024$, weight updates occur every $2secs$ (60 frames), and the recurrence frequency threshold is set to 20 frames.

For the living room scenario, two background models, B_{30} and B_{60}, are trained for $\approx$ 5400 frames ($\approx 180secs$). This ensures that both models are sufficiently trained since $\Delta_{B_{30}} = 2700$ and $\Delta_{B_{60}} = 5400$. Immediately after the training period, a person enters the scene, places a laptop bag on the ground, and leaves. The laptop bag remains in the scene for $\approx$ 3750 frames ($\approx 125secs$), at which time a person retrieves the bag returning the scene to its original state.

[1] The train terminal video is provided by the PETS2006 workshop with the support and collaboration of the British Transport Police and Network Rail.
http://www.cvg.rdg.ac.uk/PETS2006/index.html

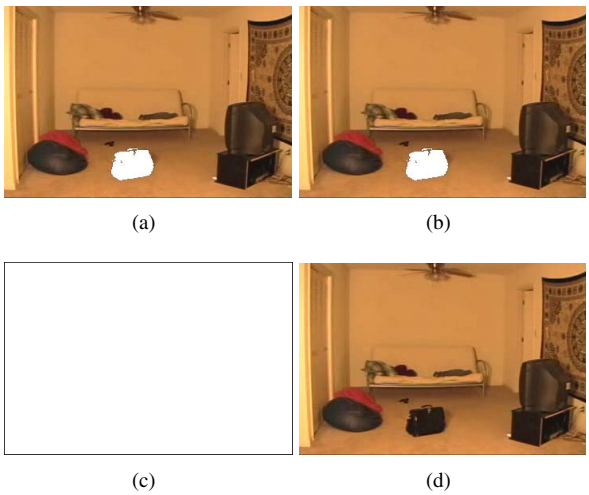

Figure 1. Frame 5753 (a) B_{30} (b) B_{60} (c) $DB_{30,60}$ (d) Monitor view

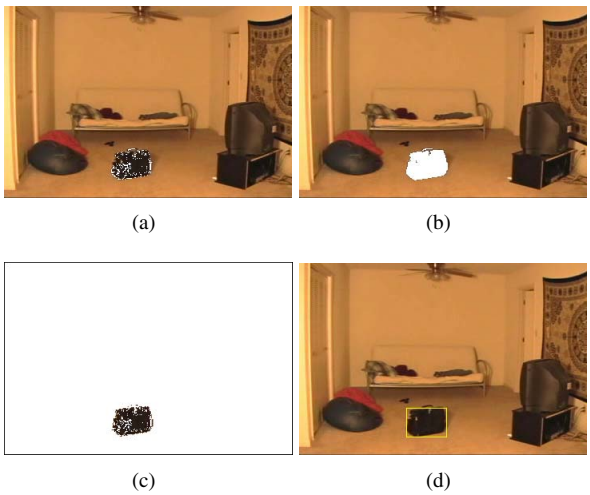

Figure 2. Frame 6729 (a) B_{30} (b) B_{60} (c) $DB_{30,60}$ (d) Monitor view with warning indicator (yellow bounding box)

Figure 1 shows the state of the overall system at frame 5753 ($\approx 192secs$). Figures 1(a) and 1(b) show the background images B_{30} and B_{60}, respectively. It can be seen that the laptop bag is still present in the foreground of both background models. Figure 1(c) is the difference background image $DB_{30,60}$ and Figure 1(d) shows the image displayed on a monitor screen. Since $DB_{30,60}$ shows no image, it can be concluded that the object has been static for less than $30secs$ and hence, no warning is issued.

Approximately 976 frames ($\approx 32.5secs$) after the bag is placed in the scene, the unattended foreground object begins to merge into the background of B_{30}. Figure 2 is a snapshot of the system at frame 6729. Figures 2(a) and 2(b) show the states of B_{30} and B_{60}, respectively. The object is present in B_{30}, but does not appear in B_{60} implying that the object has been static for more than $30secs$, but less than $60secs$. This is confirmed in $DB_{30,60}$ shown in Figure 2(c). The 30-second warning indicator (low-degree alert) is shown in Figure 2(d) as a yellow bounding box around the object.

Since the train terminal video is a more complicated scenario, three background models, B_5, B_{15}, and B_{20}, are instantiated to supply the system with both a warning period ($5secs < T_s \leq 15secs$) and a higher priority alarm signal ($T_s > 15secs$). The training period for this experiment is set to around 1800 frames ($\approx 60secs$), which is equivalent to the largest Δ of the three background models. Approximately 1700 frames ($\approx 57secs$) into the video, a man enters the scene with a long carrying case, lingers in the target area for $\approx 30secs$, leans the case up against the railing and then leaves the scene. For the remainder of the video, the carrying case remains in the same location while commuters continue navigating through the camera's field of view, occasionally obscuring portions of the carrying case. Figures 3 through 5 show key frames of the video at three separate time instances. For each of the figures, subfigures (a) through (c) show the three background models B_5, B_{15}, and B_{20}, respectively. Also, subfigures (d) and (e) are the difference background images $DB_{5,15}$ and $DB_{15,20}$, and subfigure (f) is the image shown on the monitor display.

Figure 3 is a snapshot of frame 2792 where the carrying case has just been dropped off against the railing. Since the unattended object (i.e., carrying case) has been static for less than $5secs$, $DB_{5,15}$ does not identify any connected component large enough to be classified as an object. Figure 4 shows the state of the system at frame 2946 which is $\approx 5secs$ after the object was left unattended. The static object appears in $DB_{5,15}$ indicating that the object has been stationary for $\geq 5secs$. The system notifies the monitoring party by displaying a yellow bounding box (low-degree alert) around the unattended object on the system's monitor display. Once the unattended object persists in the scene for $\geq 15secs$, an alarm is triggered. Figure 5 shows video frame 3219, which is $\approx 15secs$ after the initial time the object was abandoned. The alarm is displayed as a red bounding box (high-degree alert) around the unattended object on the system's monitor display.

The third example is an outdoor, no-parking zone for which three background models B_{30}, B_{60}, and B_{90}, are used to partition the time into a warning period ($30secs < T_s \leq 60secs$) and an alarm period ($t_s > 60secs$). Although the time intervals described above are contrived, the video does represent a realistic traffic scenario. The training period lasts ≈ 10000 frames at which point a vehicle enters

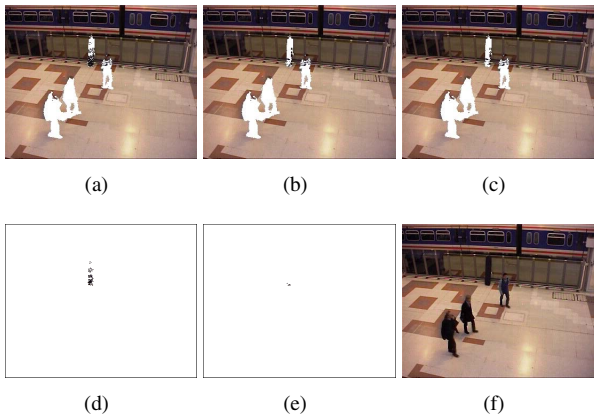

Figure 3. Frame 2792 (a) B_5 (b) B_{15}. (c) B_{20} (d) $DB_{5,15}$ (e) $DB_{15,20}$ (f) Monitor view

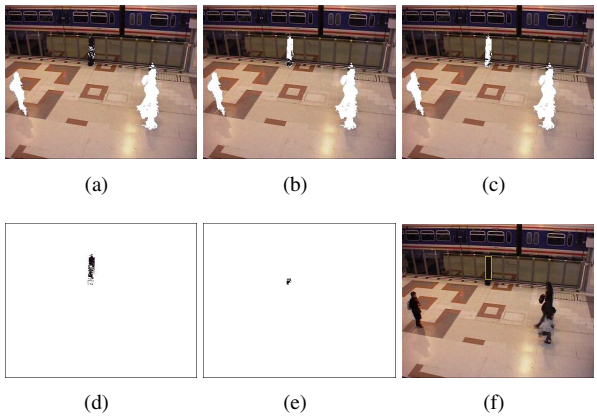

Figure 4. Frame 2946 (a) B_5 (b) B_{15} (c) B_{20} (d) $DB_{5,15}$ (e) $DB_{15,20}$ (f) Monitor view with warning indicator (yellow bounding box)

the field of view and halts near the center of the frame. The driver exits the vehicle and the scene. After $60secs$, the driver returns to the vehicle and drives the car out of the scene. The output from three significant time instances are shown in Figures 6 through 8. Subfigures (a) through (c) display the three background models B_{30}, B_{60}, and B_{90}, respectively. The subfigures (d) and (e) show the difference background images $DB_{30,60}$ and $DB_{60,90}$. Subfigure (f) is the output viewed on the monitor display.

Figure 6 shows frame 10501 in which the vehicle has just arrived in the scene. The car has not been parked for more than $30secs$ because it does not show up in $DB_{30,60}$. In Figure 7, $\approx 33secs$ after frame 10501, the car begins to appear in B_{30} and consequently, $DB_{30,60}$. Hence, the car has been stationary for more than $30secs$ indicating an illegally parked vehicle (signified by a yellow bounding box). The

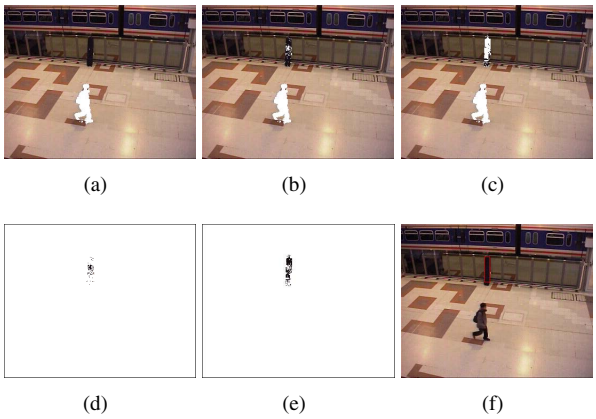

Figure 5. Frame 3219 (a) B_5 (b) B_{15} (c) B_{20} (d) $DB_{5,15}$ (e) $DB_{15,20}$ (f) Monitor view with alarm indicator (red bounding box)

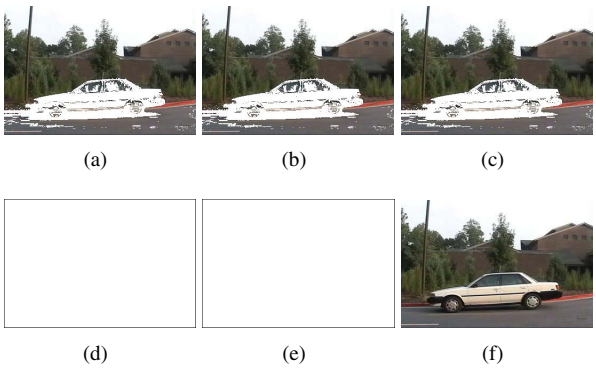

Figure 6. Frame 10501 (a) B_{30} (b) B_{60} (c) B_{90} (d) $DB_{30,60}$ (e) $DB_{60,90}$ (f) Monitor view

system allows the driver another $30secs$ following the initial warning to move the offending vehicle. At the $60secs$ mark, an alarm is triggered. Figure 8 shows frame 12385, which is $\approx 63secs$ after the car was parked. The car has merged into B_{60} and is revealed in $DB_{60,90}$ indicating that the car has been idle for greater than $60secs$. A red bounding box is displayed in the monitor view to indicate that an alarm has been issued.

4. Concluding Remarks and Future Work

The experimental results show that using a multiscale MGM background model is a valid approach to dynamic scene analysis. The system was able to signal a warning when an object remained stationary for a specified amount of time. Although the system performed reasonably well, a few weaknesses were revealed. The first problem arises

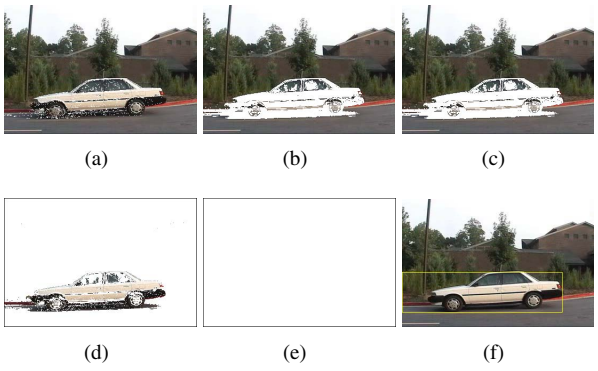

Figure 7. Frame 11483 (a) B_{30} (b) B_{60} (c) B_{90} (d) $DB_{30,60}$ (e) $DB_{60,90}$ (f) Monitor view with warning indicator (yellow bounding box)

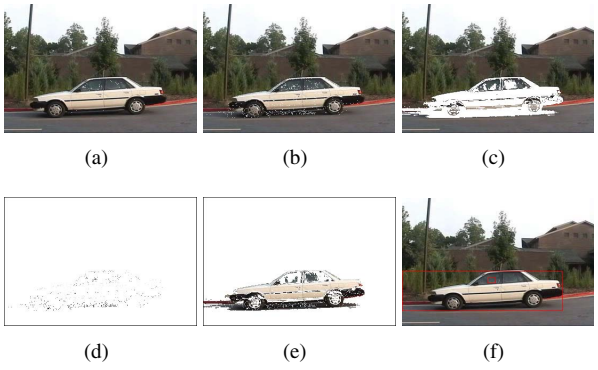

Figure 8. Frame 12385 (a) B_{30} (b) B_{60} (c) B_{90} (d) $DB_{30,60}$ (e) $DB_{60,90}$ (f) Monitor view with alarm indicator (red bounding box)

when the training period $F_t \ll \Delta$. When the background model is insufficiently trained, the foreground objects are observed to merge into the background sooner than expected. A related issue is the situation in which portions of the same object merge into the background at different rates; a problem shared by most pixel-based background modeling schemes. This affects the stationary object detection process and prevents the desired sharp transition between the warning and alarm stages. The multiscale MGM background model also presents the computational burden of having to maintain and update the background at multiple time scales; a crtical issue for a real-time video surveillance system. Fortunately, the multiscale approach is well-suited for a multiprocessor architecture in which a background model at a specified temporal scale is managed independently by one of the CPUs in the system.

The sleeping person problem has a counterpart known as the *waking person problem* which occurs when an object that originates in the background is moved. A missing object detection scheme could be implemented by exploiting the solution to the waking person problem. This scheme would be useful for detection of stolen and misplaced objects in a crowded room. The drawbacks, improvements and future directions discussed in this section will be investigated in our future work.

References

[1] S. Bhandarkar and X. Luo. Fast and robust background updating for real-time traffic surveillance and monitoring. *Proc. IEEE Wkshp. Machine Vision for Intelligent Vehicles (MVIV)*, pages 1 – 6, San Diego, CA, June 21, 2005.

[2] D. Butler, S. Sridharan, and J. V.M. Bove. Real-time and robust background updating for video surveillance and monitoring. *Proc. IEEE ICME*, Baltimore, MD, July 2003.

[3] A. Elgammal, R. Duraiswami, D. Harwood, and L. Davis. Background and foreground modeling using nonparametric kernel density estimation for visual surveillance. *Proc. IEEE*, 90(2), July 2002.

[4] D. Farin, P. deWith, and W. Effelsberg. Robust background estimation for complex video sequences. *Proc. IEEE ICIP*, 1:145–148, Barcelona, Spain, 2003.

[5] P. KaewTraKulPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection,. *Proc. Wkshp. Advances in Vision-based Surveillance Systems*, Kingston, UK, September 2001.

[6] S. Kamijo. Traffic monitoring and accident detection at intersections. *IEEE Trans. Intelligent Transportation Systems*, 1(2):108–118, 2000.

[7] X. Luo and S. Bhandarkar. Real-time and robust background updating for video surveillance and monitoring. *Springer Lecture Notes in Computer Science*, 3656:1226 – 1233, 2005.

[8] M. Massey and W. Bender. Salient stills: Process and practice. *IBM Systems Journal*, 35(3&4):557–573, 1996.

[9] S. McKenna, S. Jabri, Z. Duric, A. Rosenfield, and H. Wechsler. Tracking groups of people. *Computer Vision and Image Understanding*, 80:42–56, 2000.

[10] Y. Sheikh and M. Shah. Bayesian object detection in dynamic scenes. *IEEE Conf. Computer Vision and Pattern Recognition*, San Diego, CA, June 2005.

[11] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. *IEEE Conf. Computer Vision and Pattern Recognition*, 2:246–252, 1999.

[12] K. Tooyama, J. Krumm, B. Brumit, and B. Meyers. Wallflower: Principles and practice of background maintenance. *Proc. Intl. Conf. Computer Vision*, pages 255–261, Corfu, Greece, Sept. 1999.

[13] Y. Zhou, W. Xu, H. Tao, and Y. Gong. Background segmentation using spatio-temporal multi-resolution mrf. *Proc. IEEE MOTION05 Wkshp*, Breckenridge, CO, Jan. 2005.