Robust and Efficient Foreground Analysis for Real-time Video Surveillance

Ying-Li Tian, Max Lu, and Arun Hampapur
IBM T.J. Watson Research Center
PO BOX 704, Yorktown Heights, NY 10598
{yltian}@us.ibm.com

Abstract

We present a new method to robustly and efficiently analyze foreground when we detect background for a fixed camera view by using mixture of Gaussians models and multiple cues. The background is modeled by three Gaussian mixtures as in the work of Stauffer and Grimson [11]. Then the intensity and texture information are integrated to remove shadows and to enable the algorithm working for quick lighting changes. For foreground analysis, the same Gaussian mixture model is employed to detect the static foreground regions without using any tracking or motion information. Then the whole static regions are pushed back to the background model to avoid a common problem in background subtraction – fragmentation (one object becomes multiple parts). The method was tested on our real time video surveillance system. It is robust and run about 130 fps for color images and 150 fps for grayscale images at size 160x120 on a 2GB Pentium IV machine with MMX optimization.

1. Introduction

Robust detection of moving objects in video streams is a significant issue for video surveillance. Background subtraction (BGS) is a conventional and effective approach to detect moving objects in the stationary background. To detect moving objects in a dynamic scene, adaptive background subtraction techniques have been developed [1, 3-13]. Stauffer and Grimson [11] modeled each pixel as a mixture of Gaussians and used an on-line approximation to update the model. Their system can deal with lighting changes, slow-moving objects, and introducing or removing objects from the scene. Monnet *et al.* [10] proposed a prediction-based online method for the modeling of dynamic scenes. Their approach has been tested on a coast line with ocean waves and a scene with swaying trees. However, they need hundreds of images without moving objects to learn the background model, and the moving object cannot be detected if they move in the same direction as the ocean waves. Mittal and Paragios [9] presented a motion-based background subtraction by using adaptive kernel density estimation. In their method, optical flow is computed and utilized as a feature in a higher dimensional space. They successfully handled the complex background, but the computation cost is relatively high. Some hybrid change detectors have been developed which combine temporal difference imaging and adaptive background estimation to detect regions of change [1, 5]. Huwer et al. [5] proposed a method of combining a temporal difference method with an adaptive background model subtraction scheme to deal with lighting changes. However, none of these methods can adapt to quick image variations such as a light turning on or off. Recently, Li et al. [7] proposed a Bayesian framework that incorporates spectral, spatial, and temporal features to characterize the background appearance at each pixel. Their method can handle both the static and dynamic backgrounds and good performance was obtained on image sequences containing targets of interest in a variety of environments, e.g., offices, public buildings, subway stations, campuses, parking lots, airports, and sidewalks.

Although many researchers focus on the background subtractions, few papers can be found in the literature for foreground analysis [2, 13]. Cucchiara et al. [2] analyzed the foreground as moving object, shadow, and ghost by combining the motion information. The computation cost is relatively expensive for real-time video surveillance systems because of the computation of optical flow.

Recently, the mixture of Gaussians method is becoming popular because it can deal with slow lighting changes, periodical motions from clutter background, slow moving objects, long term scene changes, and camera noises. But it cannot adapt to the quick lighting changes and cannot handle shadows well. A number of techniques have been developed to improve the performance of the mixture of Gaussians method [3, 4, 8, 12]. In this paper, we employ a mixture of Gaussians method to analyze the foreground as *moving objects*, *abandoned objects*, or *removed objects (ghosts)* while detecting the background. The intensity and texture information are integrated to remove shadows and to make the algorithm working for quick lighting changes. Our method provides a solution for the following problems:

- Extending the mixture of Gaussians BGS method works for quick lighting changes by integrating texture information.
- Removing shadows for the mixture of Gaussians BGS method by using normalized cross-correlation of the intensities.
- Detecting the static foreground regions by using the same mixture of Gaussians of the background model.
- Avoiding the fragments of foreground objects by pushing the whole static foreground regions back to the background model.
- Classifying the static foreground regions as abandoned or removed objects (ghosts).

2. Adaptive Background Mixture Models

Stauffer and Grimson [11] introduced a mixture of *K* Gaussians (*K* is from 3 to 5) for BGS. For a pixel *X* at time *t*, the probability of the pixel can be written as [11]:

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} * \eta(X_t, \mu_{i,t}, \Sigma_{i,t}), \qquad (1)$$

where

$$\eta(X_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(X_t-\mu_t)^T \Sigma^{-1} (X_t-\mu_t)}, \qquad (2)$$

and

$$\omega_{i,t} = (1-\alpha)\omega_{i,t-1} + \alpha(M_{k,t}). \qquad (3)$$

where μ is the mean, α is the learning rate and $M_{k,t}$ is 1 for the model which matched and 0 for the remaining models. By assuming the red, green, and blue pixel values are independent and have the same variances, $\Sigma_{k,t} = \sigma_k^{\ 2} I$. After the Gaussians are ordered by the value of ω / α , the first *B* distributions are chosen as the background model, where

$$B = \arg\min_b \left(\sum_{k=1}^{b} \omega_k > T\right), \qquad (4)$$

where *T* is the minimum portion of the background model. In implementation, two significant parameters -- α and *T* are needed to be set. See more details in Stauffer and Grimson [11]. In our system, we set *K = 3* (three Gaussians), α =0.005, and *T* = 0.4. We implement the method on both grayscale and RGB video inputs.

The mixture of Gaussians method is robust to slow lighting changes, periodic motions from a cluttered background, slow moving objects, long term scene changes, and camera noises. But it cannot adapt to the quick lighting changes and cannot handle shadows. We describe some solutions for above problems in the next section.

3. Foreground Analysis

3.1 Texture integration for quick lighting changes

The mixture of Gaussians method generates large areas of false positive foreground when there are quick lighting changes (see Fig. 2a). To make the mixture of Gaussians method work for quick lighting changes, we integrate the texture information to the foreground mask for removing the false positive areas. The basic idea is that the texture in the false positive foreground areas which is caused by lighting changes should be similar to the texture in the background.

The gradient value is less sensitive to lighting changes and is able to derive an accurate local texture difference measure [6]. Here we define a texture similarity measure at pixel *X* between the current frame and the background image as

$$S(X) = \frac{\sum_{u \in W_x} 2 \,\| g(u) \| \cdot \| g_b(u) \| \cos\theta}{\sum_{u \in W_x} (\| g(u) \|^2 + \| g_b(u) \|^2)}, \qquad (5)$$

where W_x denotes the *M* x *N* neighborhood centered at pixel *X*, g and g_b is the gradient vector of the current frame and the background image respectively, and θ is the angle between the vectors. The gradient vector $g(X) = (g^x(X), g^y(X))$ and the partial derivatives $g^x(X)$ and $g^y(X)$ are obtained by the Sobel operator. In the false positive foreground areas caused by quick lighting changes, there are no texture changes between the current frame and the background. Hence, $S(X) \approx 1$. The foreground mask will be removed for the areas with $S(X) \geq T_s$. In our system, we set a similarity threshold T_s as 0.7.

3.2 Intensity integration for shadow removal

Color information is used for shadow removal by several investigators. To keep our system works for grayscale images, the intensity information is employed instead of color information. The normalized cross-correlation of the intensities is calculated at each pixel of the foreground region between the current frame and the background image. For pixel *X*, in the *M by N* neighborhood, the normalized cross-correlation is calculated as

$$NCC(X) = \frac{\sum_{u\in W_x} I_t(u)\cdot I_b(u) - \frac{1}{MN}\sum_{u\in W_x} I_t(u)\sum_{u\in W_x} I_b(u)}{\sqrt{(\sum_{u\in W_x} I_t^2(u) - \frac{1}{MN}[\sum_{u\in W_x} I_t(u)]^2)(\sum_{u\in W_x} I_b^2(u) - \frac{1}{MN}[\sum_{u\in W_x} I_b(u)]^2)}}, \qquad (6)$$

where W_x denotes the *M* x *N* neighborhood centered at pixel *X*, $I_t(u)$ and $I_b(u)$ is the intensity at pixel *u* of the current frame and the background respectively.

The pixel *X* is shadow if $NCC(X) \geq T_s$ and $I_t(X) \geq T_I$. Here we add the constrain of $I_t(X) \geq T_I$ to avoid detect shadows in very dark areas. Otherwise, the pixel *X* is real foreground.

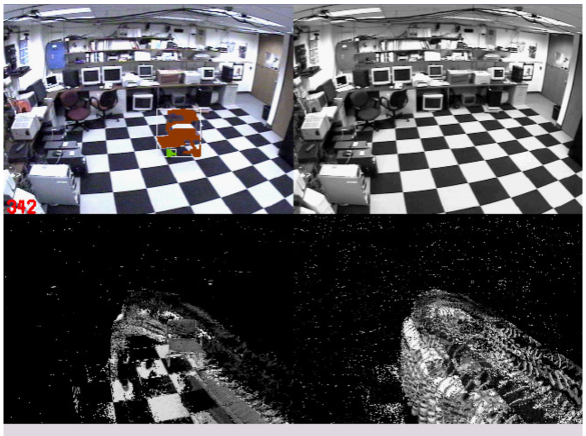
3.3 Static Object Detection and Foreground Fragment Reduction

Static Region Detection. Here, we discuss how to detect the static region (top left in Fig. 1a) by using the mixture of Gaussians of the background model. Fig. 1a shows an example of a detected static object and three Gaussian mixtures of the background model. Generally, the 1st mixture of Gaussians (top right in Fig. 1a) shows the persistence pixels and represents the background image. The repetitive variations and the relative static regions are updated to the 2nd mixture of Gaussians. The static chair can be seen in the bottom left of Fig. 1a. The 3rd mixture of Gaussians (bottom right in Fig. 1a) represents the pixels with quick changes. The *(B+1)th* mixture of Gaussians of the background model (see equation (4)) is used to detect if a foreground pixel belongs the static region:

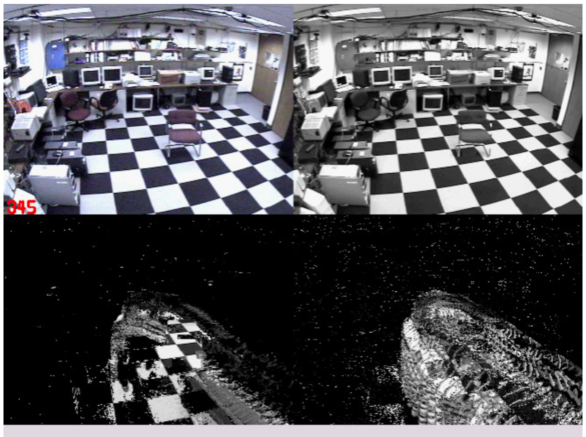
$$pixel \in static \quad region, \quad if\ \omega_{B+1} > T. \qquad (7)$$

Foreground Fragment Reduction. Foreground fragments are usual for many background subtraction methods. In the mixture of Gaussians BGS method, the different parts of a static region are often updated to the background model at different speeds based on the similarity of the pixel values between the static region and the background model. Hence many foreground fragments are caused by static regions (examples are shown in Fig. 5.)

By pushing back the static region to the background model when the static region is biggest, we can avoid the fragment of the foreground. To push the static region back to the background model, we reset the weight of the static region as the maximum weight which was defined in the program. The mean and variance of the *(B+1)th* Gaussian distribution is exchanged with the 1st Gaussian distribution for each pixel in the static region mask. Fig. 1b shows the static region detected in Fig. 1a has been pushed back to the background image (top right in Fig. 1b). Notice that the region corresponding to the static region in the 2nd mixture (bottom left in Fig. 1b) has been changed with the region in the background image (top right in Fig. 1b).



(a)

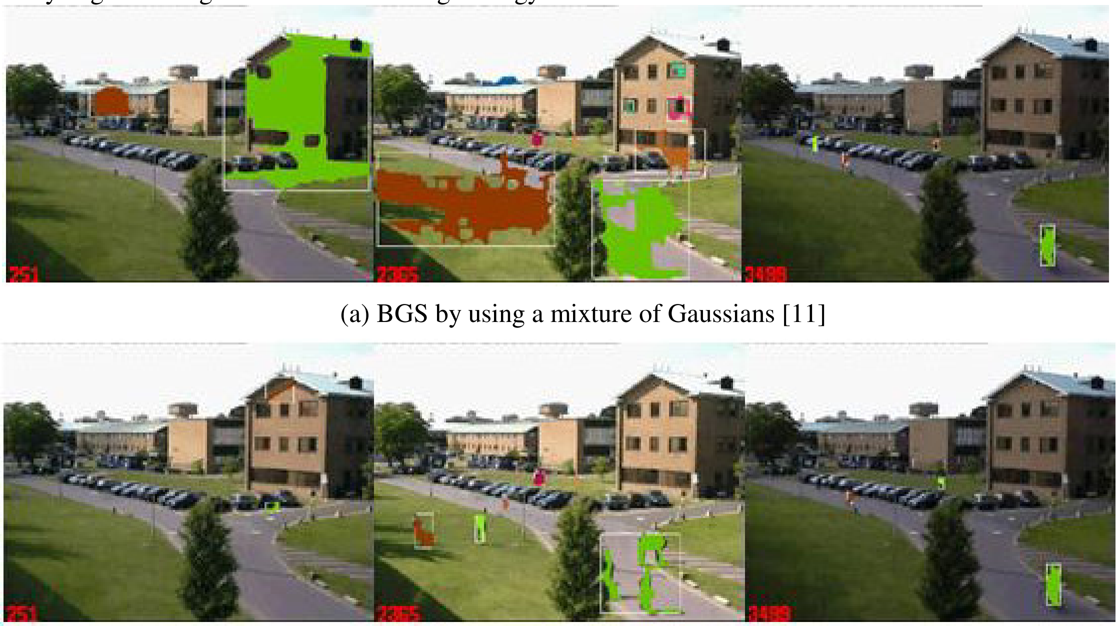


(b)

Fig. 1. Static region detection. (a) The static region mask is visualized on the original image (top left). The 1st mixture of Gaussians (top right), the 2nd mixture of Gaussians (bottom left), and the 3rd mixture of Gaussians (bottom right) of the background model are shown respectively. (b) Push the static region (the chair) to the background image (top right) from the 2nd mixture (bottom left) when the size of the static region is biggest.

3.4 Abandoned and Removed Objects Detection

Detecting abandoned and removed objects is very important for video surveillance and security. In our system, a gradient-based method is applied to the static foreground regions to detect the type of the static regions as abandoned or removed objects (ghosts) [13]. It does this by analyzing the change in the amount of edge energy associated with the boundaries of the static foreground region between the current frame and the background image. The static region is an abandoned object if there are significantly more edges. Conversely, the static region is a removed object if there are less edges. If the edge measure is similar, it typically means that there has been a state change (e.g. a door closing). More details can be found at paper [13].



(a) BGS by using a mixture of Gaussians [11]

(b) BGS by using the proposed method

Fig. 2. Examples of the background subtraction results on frame 251, 2365, and 3499 for the PETS01 sequence with quick lighting changes.

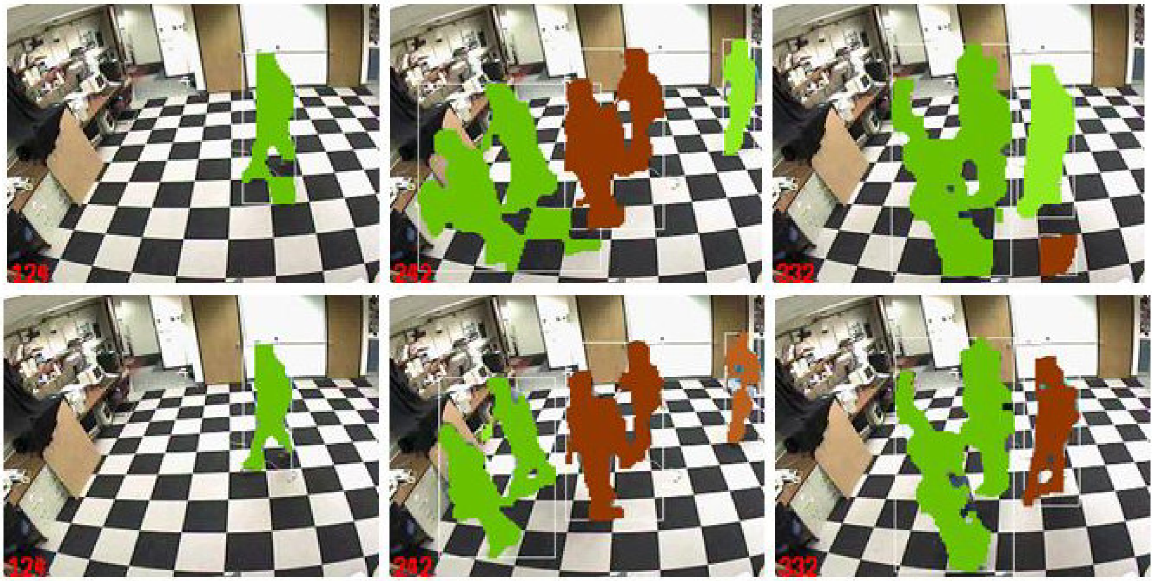
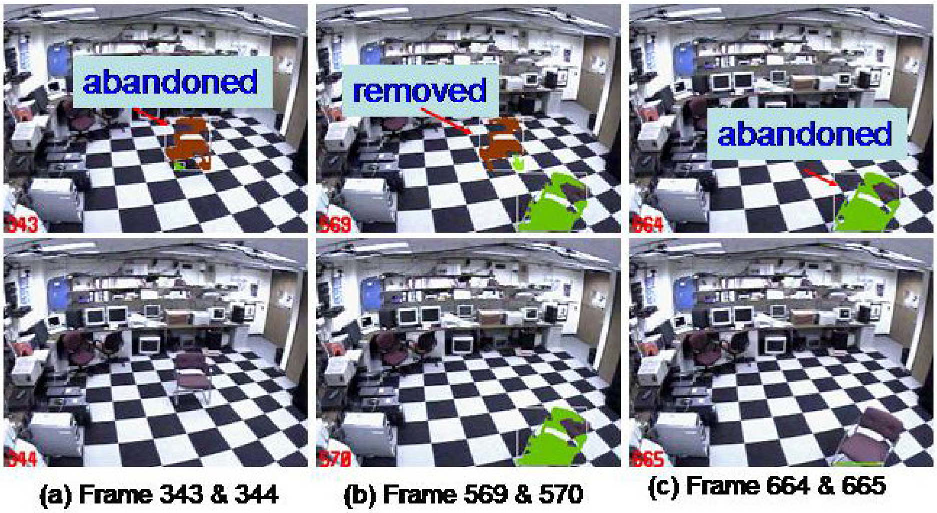
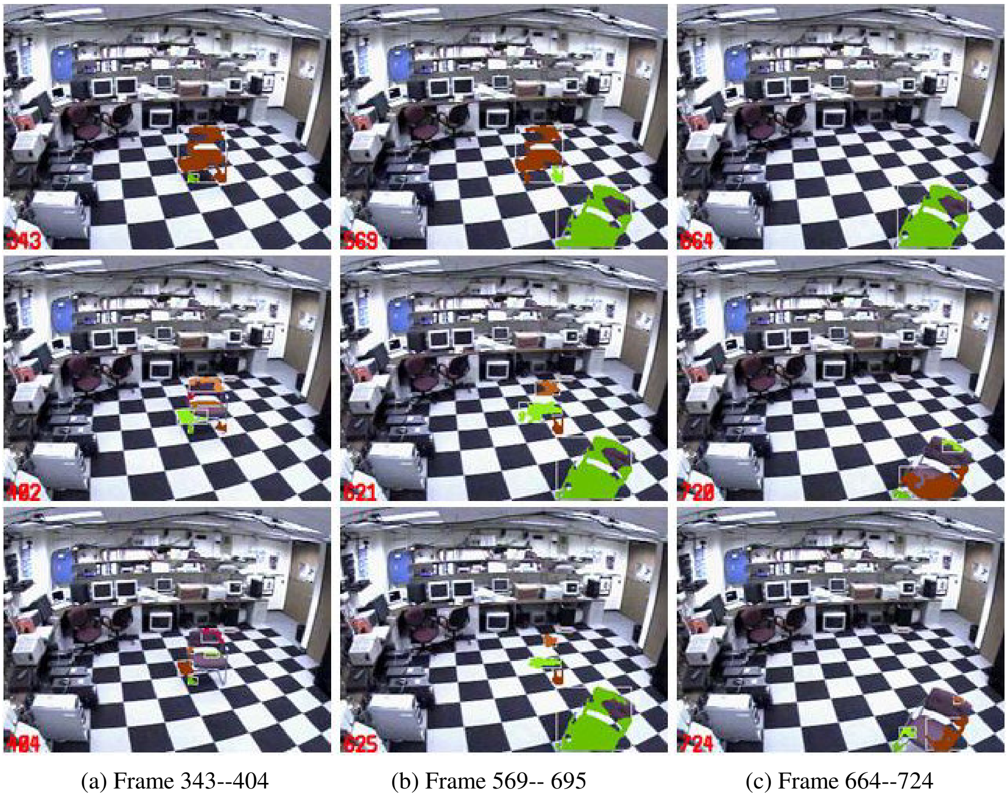


Fig. 3. Examples of the background subtraction results of the sequence with shadows. Upper row shows results of the mixture of Gaussians [11] and lower row shows results of our method.



(a) Frame 343 & 344 (b) Frame 569 & 570 (c) Frame 664 & 665

Fig. 4. Examples of static object detection, foreground fragment reduction, and abandoned and removed object discrimination. Static objects were detected at frame 343, 569, and 664 respectively and were pushed back to BG model to avoid fragment problem. The static object is abandoned object in (a) and (c), and removed object in (b).



(a) Frame 343--404 (b) Frame 569-- 695 (c) Frame 664--724

Fig. 5. Examples of the fragment problem without pushing the static region back to the background model.

4. Experimental Results

The proposed algorithm is being used in our real-time smart video surveillance system [14]. In this section, some examples demonstrate the effectiveness of our algorithm for background subtraction and foreground analysis in a variety of environments. Notice that **same parameters** were used for **all sequences**. The algorithm runs about 130 fps for color images and 150 fps for grayscale images at size 160x120 on a 2GB Pentium IV machine with MMX optimization. More quantitative results for the performance evaluation of our system can be found at paper [15].

4.1 BGS Results for Sequences with Quick Lighting Changes and Shadows

Fig. 2 shows an example result on one PETS 2001 sequence with quick lighting changes. PETS refers to the IEEE Performance Evaluation of Tracking and Surveillance Workshops. In Fig. 2a, large areas of false positive foreground were detected by the mixture of Gaussians method [11]. Fig. 2b shows that our method successfully handles the quick lighting changes by integrating texture information.

An example for shadow removal is shown in Fig. 3. The results from our method are compared to that from the mixture of Gaussian method [11]. By integrating intensity information, most of the shadows are removed, but it cannot remove strong shadows.

4.2 Foreground Analysis Results

Static Object Detection and Foreground Fragment Reduction. In the test sequence, a chair has been left at about frame 230. Then it was moved to another position at about frame 540 and was abandoned at the new position at about frame 560. Fig. 4 shows three moments (at frame 343, 569, and 664) that static objects were detected. The static regions were pushed back to the background model in the next frame (frame 344, 570, and 665) to avoid fragments. Fig. 5 shows that many foreground fragments caused by the static region detected at frame 343, 569, and 665 without pushing them back to the background model. The fragments had been adapted to the background model until frame 410, 633, and 731. The fragments lasted about 65 frames and made the tracking more difficult.

Abandoned and Removed Object Discrimination. In Fig. 4, the static object was discriminated as an abandoned object in frame 343, and 664, a removed object in frame 569 respectively.

5. Discussion and Conclusion

We presented a new method to robustly and efficiently analyze foreground and improved the mixture of Gaussians BGS method working for quick lighting changes and shadow removal by integrating texture and intensity information. Without using any tracking or motion information, static objects were detected by using the same Gaussian mixture model and were discriminated to abandoned or removed objects by analyzing the change in the amount of edge energy associated with the boundaries of the static foreground regions. The whole static regions are pushed back to the background model to avoid a fragment problem in background subtraction.

The algorithm works well in most situations with the following limitations: 1) the learning rate α affects how long is an object keeping static would be considered as a static object. 2) holes appeared on the foreground mask for large homogeneous objects because there is less texture.

6. References

[1] M. Cristani, M. Bicegi, and V. Murino, "Integrated Region- and Pixel-based Approach to Background Modeling", Proceedings of the MOTION, 2002.

[2] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati, "Detecting Moving Objects, Ghosts, and Shadows in Video Streams," IEEE Trans. on PAMI, 25: (10), October 2003.

[3] H. Eng, J. Wang, A. Kam, and W. Yau, "Novel Region-based Modeling for Human Detection within High Dynamic Aquatic Environment," Proceedings on CVPR, 2004.

[4] M. Harville, "A Framework for High-level Feedback to adaptive, per-pixel, Mixture-of-Gaussian Background Models", Proceedings on ECCV, 2002.

[5] S. Huwer and H. Niemann, "Adaptive Change Detection for Real-time Surveillance applications," Proc. of the 3rd IEEE Workshop on Visual Surveillance, pp.37-45, 2000.

[6] L. Li and M.K.H. Leung, "Integrating Intensity and Texture Differences for Robust Change Detection", IEEE Transactions on Image Processing, Vol. 11, No. 2, 2002.

[7] L. Li, W. Huang, I.Y.H. Gu, and Q. Tian, "Statistical Modeling of Complex Backgrounds for Foreground Object Detection", IEEE Transaction on Image Processing, Vol. 13, No. 11, 2004.

[8] O. Javed, K. Shafique, and M. Shah, "A Hierarchical Approach to Robust Background Subtraction using Color and Gradient Information," IEEE Workshop on Motion and Video Computing, 2002.

[9] A. Mittal and N. Paragios, "Motion-based Background Subtraction using Adaptive Kernel Density Estimation," Proceedings on CVPR, 2004.

[10] A. Monnet, A. Mittal, N. Paragios, and V. Ramesh, "Background Modeling and Subtraction of Dynamic Scenes", Proc. on ICCV, Pages 1305–1312, 2003.

[11] C. Stauffer and W.E.L. Grimson, "Adaptive Background mixture Models for Real-time Tracking", CVPR99, June, 1999.

[12] P. KaewTraKulPong and R. Bowden, "An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection," In Proc. 2nd European Workshop on Advanced Video Based Surveillance Systems, 2001.

[13] Connell, J., "Detection and Tracking in the IBM PeopleVision System," IEEE ICME, June 2004.

[14] A. Hampapur, L. Brown, J. Connell, Max Lu, Hans Merkl, S. Pankanti, A.W. Senior, Chiao-fe Shu, and Y.-L. Tian, "Multi-scale Tracking for Smart Video Surveillance," IEEE Transactions on Signal Processing, 22: (2), March 2005.

[15] L. Brown, A.W. Senior, Ying-Li Tian, J. Connell, A. Hampapur, Chiao-fe Shu, Hans Merkl, and Max Lu, "Performance Evaluation of Surveillance Systems Under Varying Conditions," IEEE Workshop on Performance Evaluation of Tracking and Surveillance, 2005.