

A Hybrid Adaptive Scheme Based on Selective Gaussian Modeling for Real-Time Object Detection

Mayssaa Al Najjar, Soumik Ghosh, Magdy Bayoumi

The Center for Advanced Computer Studies
University of Louisiana at Lafayette
Lafayette, USA
{mea5645, sxx5317, mab}@cacs.louisiana.edu

Abstract—Object detection is receiving a growing attention with the emergence of surveillance systems. This paper presents a hybrid adaptive scheme based on selective Gaussian modeling for detecting objects in complex outdoor scenes with gradual illumination changes and dense, moving background objects like swinging tree branches. The proposed technique combines simple frame difference (FD), simple adaptive background subtraction (BS), and accurate Gaussian modeling to benefit from the high detection accuracy of Mixture of Gaussian solution (MoG) in outdoor scenes while reducing the computations required, thus, making it faster and more suitable for real time surveillance applications. Moreover, by applying selective component matching and updating and hysteresis thresholding, the probability of detecting a background pixel as foreground decreases leading to better detection accuracy than MoG as demonstrated in the quantitative and qualitative comparison.

I. INTRODUCTION

As surveillance systems are becoming more popular, robust detection techniques are needed to determine moving objects. There are three main approaches for object detection: temporal difference, background subtraction, and optical flow [1]. Temporal difference is very adaptive and suitable for dynamic environments, but suffers from the aperture problem where interior object pixels are not detected. Optical flow techniques are usually employed with moving cameras, but are computationally complex and cannot be used in real time applications without specialized hardware. Background subtraction techniques are the most common schemes in surveillance systems with stationary cameras. A background model is kept and pixels in the current frame that vary significantly from this background are classified as foreground. The general procedure includes preprocessing to enhance the images, followed by background modeling, foreground detection, and data validation [2]. Different modeling techniques are related to the application, whether it is indoor with good illumination or outdoor with slightly more variations, or forest with large variations and swinging trees. This paper presents a hybrid background subtraction technique for monitoring outdoor scenes with gradual illumination changes and swinging tree branches. The technique combines FD and MoG to reduce the computations of MoG and improve its accuracy by focusing the attention on the most probable foreground pixels.

This paper is organized as follows. Section 2 reviews the major detection schemes used in surveillance systems. Section 3 presents

the proposed hybrid technique to reduce MoG's computations while providing better detection accuracy. Section 4 shows a qualitative and quantitative comparison of the major detection techniques and the proposed one, demonstrating the efficiency of the proposed scheme.

II. RELATED WORK

There are several background modeling schemes ranging from simple techniques keeping single background state estimates and providing acceptable accuracy for simple applications, all the way to complicated techniques with full density estimates; thus providing better accuracy at the expense of increased memory and complexity. FD is the simplest technique with the background being the previous frame. This method has low memory requirements, $O(1)$, high speed, and high adaptability; but suffers from the aperture problem. IVSS [3] uses FD to generate "hypotheses" about the objects, then verifies them by extracting the features using Gabor filter, and classifies objects using support vector machine. Median filter (MF), where the background is the median of all N frames in the buffer, is fast, simple, but has high memory requirements, $O(N)$ [4]. Approximated median filter (AMF) estimates the median without keeping a buffer [4]. This technique is good for indoor applications and was used for urban traffic monitoring, but suffers from slow adaptation when there is a large change in background. Any error in the background takes a long time to be corrected. Instead, the running average filter (RAF) uses exponential weighting and selective updating of background pixels, while keeping one frame i.e. low memory requirements, $O(1)$. VSAM [5] performs 3FD to determine the regions of legitimate motion, and RAF to fill the interior pixels. It is fast, effective with low memory requirements, $O(1)$ since four frames are required.

Unfortunately, none of these single estimate techniques works well for scenes with gradual illumination changes. To do so, each pixel is modeled as a probability density function (pdf), with two values: mean and standard deviation; hence preserving the speed and low memory requirements, $O(1)$. Several variations of this technique exist for intensity images, or multiple component color spaces; for instance, PFinder [6] uses statistical models based on color and shape features. Knight [7] uses statistical models of gradients and color to classify pixels, keeps six values per pixel for gradient/color information, and region mapping or $O(f)$ where f is the number of features. It groups foreground pixels in regions based on their color, and then checks their boundaries for foreground gradients, and keeps those whose boundary overlaps with detected foreground gradients.

Other statistical methods are needed for complex outdoor scenes. For instance, W^4 system uses a bimodal background model that keeps

three values per pixel: minimum intensity, maximum intensity, and maximum intensity difference between consecutive frames [8]. The initial background is constructed using MF and is then updated based on the change map. The memory requirement for computing the initial background is high, $O(N)$, since N frames are used. But background updating and foreground detection require keeping six values per pixel in the current frame (including dynamic maps) and two values for 3FD. Stauffer MoG [9] is very popular in surveillance systems for outdoor scenes because it is adaptive, online, and can handle multimodal backgrounds [10], [11]. MoG maintains a pdf for each pixel and thus has intermediate to high memory requirements, $O(K)$, where K is the number Gaussian distributions per pixel.

Prismatica [12] uses motion estimation, background estimation, and FD. Motion vectors are computed using Full Search Block Matching algorithm, but vectors that do not correspond to moving objects are suppressed. A background history array $H_{64 \times 64 \times 25}$ of l_{25} layers is kept with each element consisting of estimate background intensity and an occurrence counter. The values in the top layer are the most likely background. A similarity measure compares a candidate background block to all those in the history array and decides on the one to update. This is quite complicated as it computes similarity measures for every candidate background block. It also has high memory requirements, $O(bl)$, for keeping previous frame, $b=64 \times 64$ block status values, and $2bl$ values for H . Eigen Background (EB) is another technique that explores spatial correlation [4]. In the learning phase, N images are averaged and mean subtracted, the covariance matrix, and the best M eigenvectors are stored in an eigenvector matrix. In the classification phase, the image is projected onto the eigenspace and then onto the image space, and subtracted from the original image to detect foreground pixels. Even though the memory requirements of the training set are proportional to the number of samples, which is pretty large, the requirements in the classification phase are much less, $O(M)$.

III. PROPOSED TECHNIQUE

Even though MoG provides the best detection results in complex scenes with closely spaced moving background objects like waving trees, it requires considerable amounts of computations, which is a problem in resource constrained real-time smart camera nodes. If we can perform selective MoG, then we can greatly reduce the computations in the component matching, parameter updating, and sorting process as well as focus the attention on most probable foreground pixels, which speeds up the process and reduces the detection errors. Instead of performing the computations and comparisons for all the pixels in the frame, we update only the parameters of the portion of the scene where an object is suspected and sort the corresponding distributions. Since objects are usually very small compared to the whole frame, this leads to huge reductions in computations. In other words, the proposed technique will be beneficial in two folds:

- Less computation than MoG and faster response times. Since motion areas are much smaller than the whole image, pixel matching, parameter updating, and sorting are significantly reduced. The amount of reduction is proportional to the size of the motion area to the total frame size.
- Better detection accuracy due to selective MoG which focuses the attention on most probable foreground pixels and decreases the probability of misclassifying a background pixel as foreground, and hysteresis thresholding that improves the recall for grayscale images.

Table 1 shows a comparison among different detection techniques employed in current surveillance systems. Fig. 1 shows the block diagram for the proposed technique. The first step is to compute the non static region or the portion of the scene that contains motion followed by selective component matching and updating and then background segmentation and foreground detection.

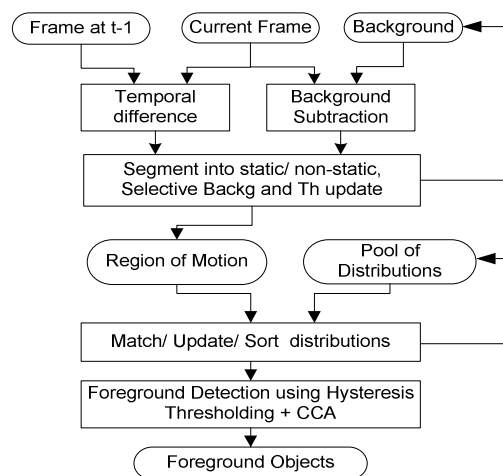


Figure 1 Proposed technique block diagram

A. Computing Region of Motion

The first step is to compute the region of motion. FD is applied to get the boundaries of the portion of the scene that is moving. A pixel $I_t(x, y)$ is classified as foreground if the difference between $I_t(x, y)$ and its predecessors at time $t-1$ is larger than a threshold I_{Th} :

$$|I_t(x, y) - I_{t-1}(x, y)| > I_{Th} \quad (1)$$

Even though simple FD is less immune to noise than 3FD, any noise introduced will be corrected in the later detection stage. FD is very adaptive to dynamic changes and requires keeping one previous frame only. However, FD cannot detect all interior object information and fails to detect any if the object stops moving at a certain frame. Thus an additional background subtraction technique is required to form the region of motion. A simple background is kept and selectively updated at each frame. A pixel in the current frame is considered part of the region of motion if the absolute difference between its current value and its background value is larger than I_{Th} :

$$|I_t(x, y) - B_t(x, y)| > I_{Th} \quad (2)$$

Initially, the background is the first frame, assuming there are no objects. I_{Th} may be initialized using Niblack's method [13]:

$$I_{th}(x, y) = m(x, y) + c \times s(x, y) \quad (3)$$

where $m(x, y)$ and $s(x, y)$ are the mean and standard deviation of a local area, which is supposed to be small enough to preserve local details but large enough to suppress noise, and c defines how much of

TABLE I. COMPARISON OF DETECTION SCHEMES USED IN SURVEILLANCE SYSTEMS

Technique	Speed	Memory	Accuracy
FD	Fastest	Low, $O(I)$	Low
MF	Fast	High, $O(N)$	Medium
AMF	Fastest	Low, $O(I)$	Low/Medium
VSAM	Fast	Low, $O(I)$	Medium
Gaussian	Fast	Low, $O(I)$	Medium
Min/Max	Medium	Medium, $O(N)$	High
MoG	Medium	Medium, $O(K)$	High
Prismatica	Slower	High, $O(lxb)$	High
EB	Medium	Medium, $O(M)$	High
Knight	Medium	Medium, $O(f)$	High
Proposed	Fast	Medium, $O(K)$	High

the total print object boundary is taken as a part of the given object. The whole image could also be taken as one area since the threshold will be corrected for each pixel in the training phase anyway. At each new frame, $B(x,y)$ and $I_{Th}(x,y)$ are then updated for non moving pixels:

$$B_{t+1}(x,y) = \alpha_1 B_t(x,y) + (1-\alpha_1) I_t(x,y) \quad (4)$$

$$I_{Th,t+1}(x,y) = \alpha_1 I_{Th,t}(x,y) + (1-\alpha_1) (5 \times |I_t(x,y) - B_t(x,y)|) \quad (5)$$

where α_1 is a time constant specifying the rate of adaptation. The final motion region contains interesting moving objects or uninteresting swinging of trees or even both. Eventually, only interesting moving objects should be left. So, the next step is to cluster/fill the objects in the motion area and then perform selective MoG to get the interesting objects out of the whole motion area.

B. Selective Match and Update

Each pixel is modeled as a mixture of K Gaussians [9]:

$$f(I_t = u) = \sum_{i=1}^K w_{i,t} \eta(u; \mu_{i,t}, \sigma_{i,t}) \quad (6)$$

Where $\eta(u; \mu_{i,t}, \sigma_{i,t})$ is the i^{th} Gaussian distribution also called component, $w_{i,t}$ is the weight or probability of each Gaussian, and K is the number of distributions, which ranges from 3 to 5. Usually, 3 Gaussian distributions are kept per pixel; at least one distribution is needed to represent foreground objects and two distributions to represent multimodal backgrounds. Increasing the number of distributions improves the performance to a certain extent at the expense of increasing memory requirements and computations. Even though K may be go up to 7, not much improvement is obtained above K = 5.

For each new frame, the pixels inside the motion area are checked and the corresponding parameters are updated. Given a pixel I_t in the motion area; we check the corresponding distributions for the distribution that I_t best fits or is most likely to belong to. The best match is defined as the distribution whose mean is not just the closest to I_t but also close enough to be considered alike. Let $d_{k,t}$ be the distance($I_t, \mu_{k,t}$) for $k=1:K$,

$$match = \left\{ \eta(\mu_{k,t}, \sigma_{k,t}) \mid d_{k,t} < \lambda \text{ and } d_{k,t} = \min[d_{1,t}, \dots, d_{K,t}] \right\} \quad (7)$$

For grayscale images, λ is usually 2.5, which accounts for almost 98.76% of the values from that distribution [14]. If there is a match, the corresponding parameters will be updated as:

$$\mu_{i,t} = (1-\rho)\mu_{i,t-1} + \rho I_t \quad (8)$$

$$\sigma_{i,t}^2 = (1-\rho)\sigma_{i,t-1}^2 + \rho(I_t - \mu_{i,t})^2 \quad (9)$$

$$\text{Where } \rho = \alpha / w_{i,t} \quad (10)$$

The approximation in (10) is faster and more logical than the one used in [9] for winner-takes-all scenarios [15]. Note that distance computations may be modified to avoid square root operations. All weights are then updated as:

$$w_{i,t} = (1-\alpha)w_{i,t-1} + \alpha M_{i,t} \quad (11)$$

$$M_{i,t} = \begin{cases} 1 & \text{if matched} \\ 0 & \text{if nonmatched} \end{cases} \quad (12)$$

Otherwise, the component with the least weight is replaced by a new component with mean I_t , large variance and small weight and maintain the means and variances of the other components, but lower their weights according to (11).

C. Foreground Detection using Hysteresis Thresholding

All the components are then sorted by their values of w_i/σ_i , with the higher ranked components being classified as background. This is because high values correspond to bigger weight values and smaller variances, which means more prominent components. The first B distributions that verify (13) are chosen as background distributions:

$$B = \arg \min_b \left(\sum_{k=1}^b w_k > T \right) \quad (13)$$

The threshold T is a value between 0 and 1. If T is very small, then most of the distributions will be classified as foreground and consequently one distribution at most will correspond to the background, which means that we will not be able to handle multimodal backgrounds. If T is very large, then most of the distributions will be classified as background; thus objects will quickly become part of the background. A trade off would be to choose T around 0.6. This value may vary depending on the type of the scene, whether it is a busy scene with lots of moving objects or just few objects.

The next step is to compare the current pixel to these background distributions and classify it as background if it matches any of the background distributions, otherwise as a foreground pixel. Instead of using simple matching, Power and Schoonees suggested using hysteresis thresholding when looking for a match [15]. The idea is to have two thresholds, T_{low} and T_{high} . If the difference between the current pixel value and the distribution mean is less than T_{low} , the pixel is strongly classified as background. If the difference between the current pixel value and the distribution mean is larger than T_{high} , the pixel is classified as foreground. Otherwise, the pixel is a foreground candidate or a weak candidate as shown in (14).

$$I_t = \begin{cases} foreground & \text{diff} > T_{high} \\ background & \text{diff} < T_{low} \\ candidate & \text{otherwise} \end{cases} \quad (14)$$

Additional connected component check procedure is needed to classify the candidate pixels as foreground or background. If a candidate pixel is found to be 8-connected to a foreground pixel, it becomes a foreground pixel. Afterwards, morphological operators are applied to form the final foreground mask. Any object that is small enough is discarded.

IV. SIMULATION RESULTS

The proposed technique was compared to some commonly used detection techniques with the video sequences taken from [16] so that obtained tracked objects may be compared to their manually computed ground truth objects. Figure 2 shows a 160x120 test frame, the ideal ground truth, and the detected objects using 3FD, MF, RAF, single Gaussian, Stauffer MoG, and the proposed technique.

The same preprocessing/morphological operations were applied to all algorithms. For the proposed technique, $I_{Th}=0.025$, $K=4$, $\alpha_1=0.9$, $v_{init} = 0.09$, $\alpha=0.005$, $T=0.7$ (non busy scene), $T_{low}=2$, $T_{high}=3$. The proposed technique outperforms the other techniques and is able to handle scenes with waving trees and light changes. The holes in the detected objects are due to processing on grayscale images where color information is discarded. The implemented algorithms are compared using the evaluation metrics from [2] and [17]: recall, precision, false alarm (FA), and detection failure (DF).

$$\text{Recall} = \frac{\text{Foreground Pixels correctly identified by algorithm}}{\text{Total Foreground Pixels in Ground Truth}} \quad (15)$$

$$\text{Precision} = \frac{\text{Foreground Pixels correctly identified by algorithm}}{\text{Total Foreground Pixels detected by algorithm}} \quad (16)$$

$$\text{False Alarm} = \frac{\text{Pixels misclassified as Foreground by algorithm}}{\text{Total Foreground Pixels detected by algorithm}} \quad (17)$$

$$\text{Detection Failure} = \frac{\text{Foreground Pixels not detected by algorithm}}{\text{Total Foreground Pixels in Ground Truth}} \quad (18)$$

Table 2 summarizes the comparison results. 3FD is chosen for its simplicity, speed, and low requirements. In many cases, system designers prefer a simple algorithm that can be easily implemented at the camera ends, that is very dynamic, and does not leave a trail behind the moving object, and detection errors may be recovered at a later stage. MF is simple with relatively good accuracy, but requires lots of memory, and thus exponential weighting and selective updating of background pixels is preferred even if the performance degrades a little. The problem with all these unimodal background techniques is that they cannot incorporate the small repetitive background motion inside the background model. Even single Gaussian is not able to handle these variations in the background appearance. So even though the recall values are high, especially when compared to those obtained with MoG, the precision values are unacceptable. Even though these algorithms detect most of the foreground objects, which leads to high recall value, they are also detecting lots of background objects as foreground, indicated by the small precision values. This means that they are detecting interesting and non-interesting objects as foreground and have no mean to distinguish moving background from moving objects. Gaussian models, on the other hand, provide statistical descriptions of background models and can thus distinguish swinging trees from moving objects. However, MoG requires lots of computations, which brings the proposed technique to the table as it reduces the amount of computations, and thus speeds up the detection process, while providing better detection results. The accuracy is improved by focusing the attention on most probable foreground pixels: Better accuracy because of selective MoG and hysteresis thresholding: probability of detecting a pixel belonging to the background as foreground decreases.

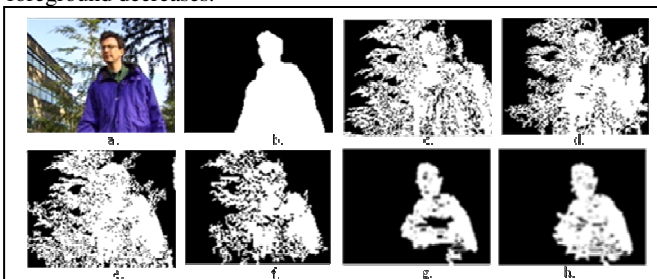


Figure 2 a. Frame showing man walking in outdoor scene with waving trees, b. Ground Truth, c. Result using FD, d. Result using MF, e. Result using RAF, f. Result using Single Gaussian, g. Result using Stauffer MoG, h. Result using proposed technique

V. CONCLUSION

We presented a hybrid detection technique for real-time monitoring of outdoor environments with gradual illumination changes and waving tree branches. The technique combines simple FD with adaptive background subtraction and selective MoG processing from matching to updating and sorting and foreground detection using dual thresholding. The proposed technique requires fewer computations than MoG and faster response times while offering comparable and even better detection accuracy. This is due to the selective processing and dual thresholding which focuses the attention on most probable foreground pixels and thus decreases the probability of misclassifying a background pixel as foreground, and reduces the holes for grayscale images.

TABLE II. QUANTITATIVE COMPARISON

	Recall	Precision	FA	DF
3 FD	0.8270	0.4481	0.1730	0.5519
Median	0.8856	0.5009	0.1144	0.4991
RAF	0.8808	0.4474	0.1192	0.5226
SG	0.7124	0.5750	0.2876	0.4250
MoG	0.6101	0.9658	0.3899	0.0342
Proposed	0.7414	0.9561	0.2586	0.0439

ACKNOWLEDGMENT

This work has been supported by the US Department of Energy and Louisiana Board of Regence, UCOMS project.

REFERENCES

- [1] M. Valera and S.A. Velastin, "Intelligent distributed surveillance systems: a review," *IEEE Proc. Vision, Image and Signal Processing*, vol. 152, no. 2, pp. 192 – 204, April 2005.
- [2] S.S. Cheung and C. Kamath, "Robust techniques for background subtraction in urban traffic video," Center for Applied Scientific Computing, Lawrence Livermore National Laboratory.
- [3] Y. Xiaoqing, S. Zehang, Y. Varol, G. Bebis, "A distributed visual surveillance system," *IEEE Conf. Advanced Video and Signal Based Surveillance*, pp. 199 - 204, July 2003.
- [4] M. Piccardi, "Background subtraction techniques: a review," *IEEE Conf. on Systems, Man and Cybernetics*, vol. 4, pp. 3099-3104, October 2004.
- [5] R.T. Collins, A.J. Lipton, and T. Kanade, "A system for video surveillance and monitoring," *Proceedings of the American Nuclear Society (ANS) Eighth International Topical Meeting on Robotics and Remote Systems*, April, 1999.
- [6] C. Wren, A. Azrbayejani, T. Darrell, A.P. Pentland, "Pfinder: real-time tracking of the human body," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, no.7, pp. 780-785, 1997.
- [7] M. Shah and O. Javed, K. Shafiq, "Automated visual surveillance in realistic scenarios". *IEEE Multimedia*, vol. 14, no. 1, pp. 30-39, 2007.
- [8] I. Haritaoglu, D. Harwood, L.S. Davis, "W4: real-time surveillance of people and their activities," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp.809-830, August 2000.
- [9] C. Stauffer and W.E. Grimson, "Adaptive background mixture models for real time tracking," *IEEE Proc. CVPR*, pp.246-252, June 1999.
- [10] P. Remagnino and G.A. Jones, "Classifying surveillance events from attributes and behaviour," *British Machine Vision Conference*, 2001.
- [11] Y.L.Tian, M. Lu and A. Hampapur, "Robust and efficient foreground analysis for real-time video surveillance," *IEEE Proc. CVPR*, vol. 1, pp. 1182-1187, June 2005.
- [12] S.A. Velastin, B.A. Boghossian, B.P. Lo, J. Sun, and M.A. Vicencio-Silva, "PRISMATICA: Toward ambient intelligence in public transport environments," *IEEE Trans. Systems, Man and Cybernetics, Part A*, vol. 35, no. 1, pp. 164-182, January 2005.
- [13] G. Leedham, C. Yan, K. Takru, J. Tan, L. Mian, "Comparison of some thresholding algorithms for text/background segmentation in difficult document images," *IEEE Conf. Document Analysis and Recognition*, vol. 2, pp.895-900, 2003.
- [14] J. Wood, "Statistical background models with shadow detection for video based tracking," 2007.
- [15] P.W. Power and J.A. Schoonees, "Understanding background mixture models for foreground segmentation," *Proc. Image and Vision Computing*, pp. 267-271, 2002.
- [16] K. Toyama, J. Krumm, B. Brumitt, B. Meyers, "Wallflower: principles and practice of background maintenance," *IEEE Conf. Computer Vision*, vol. 1, pp. 255-261, 1999.
- [17] J.C. Nascimento and J.S. Marques, "Performance evaluation of object detection algorithms for video surveillance," *IEEE Trans. Multimedia*, vol. 8, no. 4, pp. 761-774, 2006.