

# INTEGRATION OF BACKGROUND MODELING AND OBJECT TRACKING

Yu-Ting Chen<sup>1,2</sup>, Chu-Song Chen<sup>1,3</sup>, and Yi-Ping Hung<sup>1,2,3</sup>

<sup>1</sup> Institute of Information Science, Academia Sinica, Taipei, Taiwan.

<sup>2</sup> Dept. of Computer Science and Information Engineering, National Taiwan University, Taipei, Taiwan.

<sup>3</sup> Graduate Institute of Networking and Multimedia, National Taiwan University, Taipei, Taiwan.

## ABSTRACT

Background model and tracking became critical components for many vision-based applications. Typically, background modeling and object tracking are mutually independent in many approaches. In this paper, we adopt a probabilistic framework that uses particle filtering to integrate these two approaches, and the observation model is measured by Bhattacharyya distance. Experimental results and quantitative evaluations show that the proposed integration framework is effective for moving object detection.

## 1. INTRODUCTION

Background modeling/subtraction is a fundamentally important module for many applications, such as visual surveillance and human gesture analysis. By learning a background model from a training image sequence, the problem of moving object detection is transformed to that of classifying a static scene into foreground and background regions.

Methods of background modeling are mainly studied in pixel level and statistical distribution of each individual pixel is usually modeled by Gaussian distribution. Generally, a single Gaussian model used in [12] and [5] is not sufficient to represent the background since the backgrounds is often non-stationary. In [10], Stauffer and Grimson proposed a state-of-the-art framework, *Mixture of Gaussian* (MoG), to modeled each pixel with  $k$  Gaussians, where  $k$  lies in 3 to 5, and an on-line  $K$ -means approximation was used instead of using exact EM. Besides, the MoG approach is modified or extended in several researches. For example, [3] and [4] used YUV color plus depth and *Local Binary Pattern* (LBP) [9] histogram as features, respectively. In [8], Lee proposed an effective learning algorithm for MoG. Instead of using Gaussian mixtures, several other methods adopted different models. For example, Toyama et al. [11] proposed a Wallflower framework to address the background maintenance problem in three levels, pixel, region, and frame levels. In [2], Elgammal et al. proposed a non-parametric background subtraction method utilizing Parzen-window density estimation. In [7], Kim et al. presented a real-time algorithm called CodeBook that is efficient in either memory or speed.

After moving object is detected by background modeling, some tracking algorithms might be performed to track the object. Typically, there are two mechanisms, appearance model and search algorithm, for object tracking. For example, MeanShift [1] used color histogram as the appearance model to measure the similarity of the target object and candidates. On the other hand, the search algorithm finds the most likely state of the tracked object according to its similarity measurement. For example, Isard and Blake proposed CONDENSATION algorithm [6] to track the contour of object.

In previous researches, background modeling and object tracking are usually performed independently to each other. Actually, good detection result of background modeling can provide good prior information for tracking. On the contrary, good tracking results might be a better prior knowledge for adjusting the background models. In this paper we provide a framework to integrate and cooperate background modeling and object tracking approaches with the use of probabilistic framework.

To integrate these two approaches, recall that each incoming image is classified into foreground and background regions by learned background model. To do this, the feature of each pixel of incoming image is compared against existing background models until a match is found. A match is defined as the distance between feature and learned model is less than a threshold  $T$ . If a matched model is found and is a stable model (see Section 2), the pixel is detected as background; otherwise, the pixel is classified as foreground. Typically, the threshold  $T$  is usually kept as a static variable in previous researches. Based on the following two observations, the selection of  $T$  is not easy:

- When the color of moving object is similar to that of the background, a strict  $T$  is preferred to prevent foreground object from being classified as background.
- When the color of moving object is dissimilar to that of the background, a loose  $T$  is suitable to decrease false alarm (background regions are detected as foreground).

On the basis of these two observations, we address the problem of variable threshold selection for background modeling. In this paper, color histogram of object is used as ap-

pearance model for object tracking and the tracking result is used to select a discriminative  $T$  to separate input image into foreground and background regions with maximum separability. In addition, such adjusted background model can provide better detection result as a prior for tracking.

In this paper, our contribution is that a probabilistic framework uses particle filtering to integrate background modeling and object tracking, and the observation model is measured by Bhattacharyya distance as used in [1]. In addition, existing background approaches can be adopted with merely a slight modification and MoG [10] approach is used in this work. Experimental results and quantitative evaluations show that the proposed integration framework is effective for moving object detection.

## 2. GENERAL DESCRIPTION OF BACKGROUND MODELING

A pixel-based approach can be generally characterized as a quadruple  $\{F, M_{(t)}, \Phi, \Gamma\}$ . The first element  $F$  depicts the feature extracted for a pixel, which might be gray/color values [2, 3, 7, 10], depth [3], etc. The second element  $M_{(t)}$  consists of the background models maintained at time  $t$  for the pixel, e.g. each model in MoG is represented as a single Gaussian distribution in the mixture. Note that almost all the methods maintained  $M_{(t)} = \{M_{(t)}^S, M_{(t)}^P\}$ , where  $M_{(t)}^S$  and  $M_{(t)}^P$  are the sets of *stable* and *potential* background models, respectively. For example, in MoG [10], the first  $B$  Gaussian densities constitute  $M_{(t)}^S$  and the other constitute  $M_{(t)}^P$ . In CodeBook [7], background model and cache model stand for  $M_{(t)}^S$  and  $M_{(t)}^P$ , respectively. The third element  $\Phi$  is a function determining whether a given pixel  $q$  at time  $t$  is background based on pixel feature, stable background models, and threshold:

$$\{1, 0\} \leftarrow \Phi[F(q), M_{(t)}^S, T], \quad (1)$$

where  $F(q)$  is the feature of  $q$ ,  $T$  is the threshold for finding out the matched model in  $M_{(t)}^S$ , and 1 and 0 stand for background and foreground respectively. Note that only the stable model  $M_{(t)}^S$  is involved in the determination. To realize  $\Phi$  typically involves the search of the matched model in  $M_{(t)}^S$ . That is, the distance between matched model and  $F(q)$  is less than  $T$ . The fourth element  $\Gamma$  is another function that updates the model and generate a new model at time  $t + 1$  based on pixel feature  $F(q)$ , current model  $M_{(t)}$ , and threshold  $T$ :

$$M_{(t+1)} \leftarrow \Gamma[F(q), M_{(t)}, T], \quad (2)$$

and a new pair of models,  $M_{(t+1)} = \{M_{(t+1)}^S, M_{(t+1)}^P\}$ , is obtained. To realize  $\Gamma$  typically involves the search of the matched model to  $F(q)$  in  $M_{(t)}$ .

Note that in Eq. (1), if no matched model is found, the corresponding pixel is determined as foreground; otherwise the pixel is background. Therefore, more false positive and more false negative results are obtained with the use of strict

and loose  $T$ , respectively. In previous researches, the value of  $T$  is usually defined as a static variable. To our knowledge, no research has used variable  $T$ . In this framework, particle filtering is used to select a suitable  $T$  according to object tracking result. Besides, our approach does not restrict adopted background modeling approach, and MoG is used in this work.

## 3. VARIABLE THRESHOLD SELECTION

After background model is learned, an initial value is selected for  $T$ . In our experiment, we choose initial  $T$  as 3 in MoG model. Once a moving object is detected at time  $t$ , the tracking algorithm is started and the color histogram  $O_t$  of detected object in foreground region  $R$  is calculated. To compute  $O_t$ , let  $\{u_i^j\}_{i=1, \dots, n; j \in \{R, G, B\}}$  be the intensity value at color channel  $j$  of the pixel  $u$  located at  $i$  of incoming image  $I_t$ . We use 16 bins to calculate the intensity histogram for each color channel  $j$ . Therefore, the color histogram has  $K = 48(16 \times 3 = 48)$  bins. Besides, we define a function  $b : u_i^j \rightarrow \{1, \dots, K\}$  which maps  $u_i^j$  to the bin index  $b(u_i^j)$  of the histogram, and the color histogram  $O_t$  is calculated by

$$O_t(k) = C \sum_{u_i \in I_t; u_i \in R} \delta[b(u_i^j) - k], \quad (3)$$

where  $C$  is a normalization term to ensure  $\sum_{k=1}^K O_t(k) = 1$  and  $\delta$  is the Kronecker delta function. With the appearance information  $O_t$ , the object can be tracked by measuring the similarity of  $O_t$  and color histogram of candidates at time  $t + 1$ . In addition, particle filtering with prior information of  $O_t$  is used to choose discriminative threshold  $T$ . In the following, we briefly introduce particle filtering and adopted dynamic model and observation model.

### 3.1. Particle Filtering

Particle filtering is based on Bayesian Approach and Monte Carlo Sequential Method, and the main concept is captured by CONDENSATION [6]. For simplicity, we use formulation of CONDENSATION to briefly describe the particle filtering.

Let state parameter vector at time  $t$  be denoted as  $\mathbf{x}_t$ , and its observation as  $\mathbf{z}_t$ . The history of state parameters and observations from time 1 to  $t$  is denoted as  $\mathcal{X}_t = \{\mathbf{x}_1, \dots, \mathbf{x}_t\}$  and  $\mathcal{Z}_t = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$ , respectively. Particle filtering is used to approximate posterior distribution of state  $\mathbf{x}_{t+1}$  given observation  $\mathcal{Z}_{t+1}$ . From Bayesian rule and Markov chain with independent observations, the rule for propagation of posterior over time is:

$$p(\mathbf{x}_{t+1} | \mathcal{Z}_{t+1}) \propto p(\mathbf{z}_{t+1} | \mathbf{x}_{t+1}) \int_{\mathbf{x}_t} p(\mathbf{x}_{t+1} | \mathbf{x}_t) p(\mathbf{x}_t | \mathcal{Z}_t). \quad (4)$$

Note that the recursive form allows the posterior at time  $t$  be the prior at time  $t + 1$ . Particle filtering infer posterior

$p(\mathbf{x}_{t+1}|\mathcal{Z}_{t+1})$  by a finite set of  $N$  particles  $\mathcal{S}_t = \{\mathbf{s}_t^{(n)}, \pi_t^{(n)}\}$ , where  $\mathbf{s}_t$  is a value of state  $\mathbf{x}_t$  and  $\pi_t$  is a corresponding sampling probability. Besides, dynamic model,  $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$ , and observation model,  $p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1})$ , are needed and we will describe our choice of these two probabilities in the following subsection. More details and theoretical foundation can be found in [6]. One iteration steps are shown below:

1. Select samples  $\mathcal{S}'_t = \{\mathbf{s}'_t{}^{(n)}, \pi_t'^{(n)}\}$  from  $\mathcal{S}_t$ .
2. Predict by sampling from  $s_{t+1}^{(n)} = p(\mathbf{x}_{t+1}|\mathbf{x}_t = \mathbf{s}'_t{}^{(n)})$  and  $\pi_{t+1}^{(n)} = 1/N$ .
3. Measure and weight  $\pi_{t+1}^{(n)}$  in terms of the measured feature  $\mathbf{z}_{t+1}$  as:  $\pi_{t+1}^{(n)} = p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1} = \mathbf{s}_{t+1}^{(n)})$ .
4. Normalize  $\pi_{t+1}^{(n)}$  such that  $\sum \pi_{t+1}^{(n)} = 1$ .

### 3.2. Variable Threshold Selection

To select  $T$ , particle filtering is used and  $N$  particles are sampled with all  $\mathbf{s}_t$  and  $\pi_t$  are initialized as 3 and  $1/N$ , respectively. Recall that we need to define the dynamic model and observation model for particle filtering.

#### 3.2.1. Dynamic Model

An unconstrained *Brownian* motion is used as dynamic model:

$$\mathbf{s}_{t+1}^{(n)} = \mathbf{s}'_t{}^{(n)} + v_t, \quad (5)$$

where  $v_t \sim \mathcal{N}(0, \Sigma)$  is a normal distribution.

#### 3.2.2. Observation Model

To begin with, the reference background image  $Ref_t$  shall be calculated from background model  $M_{(t)}$ . In our experiments, we use the mean of most stable Gaussian model (with maximum  $\sigma/\omega$  value in MoG) to represent the pixel value of image  $Ref_t$ . In time step  $t+1$ , input frame image  $I_{t+1}$  can be classified into foreground region  $R_{FG}^{(n)}$  and background region  $R_{BG}^{(n)}$  by assigning  $T = \mathbf{s}_{t+1}^{(n)}$  for each particle.

Therefore, two color histograms,  $I_{t+1}^{FG}$  and  $I_{t+1}^{BG}$ , of foreground and background regions of image  $I_{t+1}$  and one color histogram,  $Ref_t^{BG}$ , of background region of image  $Ref_t$  can be calculated by:

$$I_{t+1}^{FG}(k) = C_1 \sum_{u_i \in I_{t+1}; u_i \in R_{FG}^{(n)}} \delta[b(u_i^j) - k], \quad (6)$$

$$I_{t+1}^{BG}(k) = C_2 \sum_{u_i \in I_{t+1}; u_i \in R_{BG}^{(n)}} \delta[b(u_i^j) - k], \quad (7)$$

and

$$Ref_t^{BG}(k) = C_3 \sum_{u_i \in Ref_t; u_i \in R_{BG}^{(n)}} \delta[b(u_i^j) - k], \quad (8)$$

where  $C_1$ ,  $C_2$ , and  $C_3$  are all normalization terms.

With the use of discriminative threshold  $T$ , the color histogram of tracked object is similar to that of the foreground region of image  $I_{t+1}$ , and the color histogram of background region of image  $Ref_t$  is similar to that of background region of image  $I_{t+1}$ . That is,  $O_t$  and  $Ref_t^{BG}$  are similar to  $I_{t+1}^{FG}$  and  $I_{t+1}^{BG}$  respectively, and Bhattacharyya distance is used to measure the similarity between two histograms  $h_1$  and  $h_2$ :

$$dist(h_1, h_2) = \sqrt{\sum_{i=1}^K \sqrt{h_1(i)h_2(i)}}, \quad (9)$$

where  $h_1(i)$  and  $h_2(i)$  are  $i^{th}$  bin value of  $h_1$  and  $h_2$ . Therefore, two distances,  $dist(O_t, I_{t+1}^{FG})$  and  $dist(Ref_t^{BG}, I_{t+1}^{BG})$  can be calculated. The observation model is defined as the linear combination of these two distances as:

$$\begin{aligned} \pi_{t+1}^{(n)} &= p(\mathbf{z}_{t+1}|\mathbf{x}_{t+1} = \mathbf{s}_{t+1}^{(n)}) \\ &= \alpha \times dist(O_t, I_{t+1}^{FG}) + (1 - \alpha) \times dist(Ref_t^{BG}, I_{t+1}^{BG}), \end{aligned} \quad (10)$$

where  $0 \leq \alpha \leq 1$  is a user defined parameter and we set  $\alpha = 0.5$  in our experiments.

Once all  $N$  particles are measured, the threshold  $T$  at time step  $t+1$  is selected as  $\mathbf{s}_{t+1}^{(n)}$  whose corresponding  $\pi_{t+1}^{(n)}$  has the maximum sampling probability over all  $N$  particles. Image  $I_{t+1}$  can then be classified into foreground and background according to  $T$ . Finally,  $I_{t+1}^{FG}$  is calculated and used for updating the color histogram of tracked object for robust tracking in time step  $t+2$  as:

$$O_{t+1}(i) = \beta O_t(i) + (1 - \beta) I_{t+1}^{FG}(i) \quad (i = 1, \dots, K), \quad (11)$$



where  $0 \leq \beta \leq 1$  is a user defined parameter and we set  $\beta = 0.8$  to  $0.95$  in our experiments.

## 4. EXPERIMENTAL RESULTS







To evaluate proposed method, one outdoor and one indoor video sequences of the ATON project (<http://cvrr.ucsd.edu/aton/shadow>) are adopted as the benchmarks as summarized in Table 1. These two sequences of ATON include outdoor *Campus* sequence with signal noises and static indoor *Intelligent Room* sequence. Detection results of our method with 10 particles and original MoG are shown in Table 2 and 3, respectively. From these results, our method with variable  $T$  has generally better results than original MoG method.

Besides, we use false positive (background pixels are classified as foreground), false negative (foreground pixels are classified as background), and the summation of false positive and false negative to quantitatively evaluate the effect of our method. The post processing and all parameter for our method and MoG are set the same and the evaluation results are shown in Table 4. Table 4 shows that proposed method can provide an averagely low error classified number of pixels of background modeling. In addition, the average speed of *Campus* and *Intelligent Room* sequences are 5.30 fps and 8.22 fps by using 3.4 GHz processor and 768 MB memory.

**Table 1.** Two benchmark sequences used in our experiments.

Sequence Name	Campus	Intelligent Room
Sequence Image		
Frame Number	400	170
Sequence Type	Outdoor	Indoor
Image Size	320×240	320×240
Frames for Training	20	20

**Table 2.** Detection results of *Campus* sequence.

Original Image	Our Method	MoG
		
		

## 5. CONCLUSION







A method for integrating background modeling and object tracking is presented in this paper. In this framework, color histogram of moving object is used as appearance model for object tracking. Besides, the tracking result is used to select a discriminative threshold  $T$  for background modeling via particle filtering. Experimental results show that the proposed framework can further improve the performances of the adopted background modeling approach.

**ACKNOWLEDGMENTS:** This work was supported in part under grants NSC 94-2752-E-002-007-PAE and 94-EC-17-A-02-S1-032.

## 6. REFERENCES

- [1] D. Comaniciu, V. Ramesh, and P. Meer, "Real-time Tracking of Non-rigid Objects using Mean Shift," *Proc. CVPR*, 2000.
- [2] A. Elgammal, D. Harwood, and L. S. Davis, "Non-parametric Model for Background Subtraction," *Proc. ECCV*, 2000.
- [3] M. Harville, "A Framework for High-level Feedback to Adaptive, Per-pixel, Mixture-of-Gaussian Background Models," *Proc. ECCV*, 2002.
- [4] M. Heikkilä and M. Pietikäinen, "A Texture-based Method for

**Table 3.** Detection results of *Intelligent Room* sequence.

Original Image	Our Method	MoG
		
		

**Table 4.** Quantitative evaluations by averaged false positive (FP), false negative (FN), and the summation of FP and FN.

Sequence Name	Algorithm	FP	FN	FP+FN
Campus	Our Method	133.37	124.68	258.05
	MoG	364.16	34.47	398.63
Intelligent Room	Our Method	426.33	114.44	540.77
	MoG	563.56	63.78	627.34

Modeling the Background and Detecting Moving Objects," *IEEE Trans. on PAMI*, 28(4), 2006.

- [5] T. Horprasert, D. Harwood, and L. S. Davis, "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection," *Proc. ICCV Frame-rate Workshop*, 1999.
- [6] M. Isard and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," *Proc. ECCV*, 1996.
- [7] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. S. Davis, "Real-time Foreground-Background Segmentation Using CodeBook Model," *Real-Time Imaging*, 11(3), 2005.
- [8] D. S. Lee, "Effective Gaussian Mixture Learning for Video Background Subtraction," *IEEE Trans. on PAMI*, 27(5), 2005.
- [9] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution Gray-scale and Rotation Invariant Texture Classification with Local Binary Patterns," *IEEE Trans. on PAMI*, 24(7), 2002.
- [10] C. Stauffer and W. E. L. Grimson, "Adaptive Background Mixture Models for Real-time Tracking," *Proc. CVPR*, 1999.
- [11] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wall-flower: Principles and Practice of Background Maintenance," *Proc. ICCV*, 1999.
- [12] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time Tracking of the Human Body," *IEEE Trans. on PAMI*, 19(7), 1997.