

# Optical Engineering

[SPIDigitalLibrary.org/oe](http://SPIDigitalLibrary.org/oe)

## **High-accuracy background model for real-time video foreground object detection**

Wen-Kai Tsai  
Chung-Chi Lin  
Ming-Hwa Sheu

# High-accuracy background model for real-time video foreground object detection

## Wen-Kai Tsai

National Yunlin University of  
Science & Technology  
Department of Electronics Engineering  
Graduate School of Engineering  
Science and Technology  
Doulou, Yunlin, Taiwan

## Chung-Chi Lin

Tunghai University  
Department of Computer Science  
Taichung, Taiwan

## Ming-Hwa Sheu

National Yunlin University of  
Science & Technology  
Department of Electronics Engineering  
Graduate School of Engineering  
Science and Technology  
Doulou, Yunlin, Taiwan  
E-mail: sheumh@yuntech.edu.tw

**Abstract.** Video foreground object detection faces the problems of moving backgrounds, illumination changes, chaotic motion in real word applications. This paper presents a hybrid pixel-based background (HPB) model, which is constructed by single stable record and multi-layer astable records after initial learning. This HPB model can be used for background subtraction to extract objects precisely in various complex scenes. Using the multi-layer astable records, we also propose a homogeneous background subtraction that can detect the foreground object with less memory load. Based on the benchmark videos, the experimental results show that single stable and 3-layer multi-layer astable records can be enough for background model construction and are updated quickly to overcome the background variation. The proposed approach can improve the average error rates of foreground object detection up to 86% when comparing with the latest works. Furthermore, our method can achieve real-time analysis for complex scenes on personal computers and embedded platforms.  
© 2012 Society of Photo-Optical Instrumentation Engineers (SPIE). [DOI: 10.1117/1.OE.51.2.027202]

Subject terms: foreground object detection; hybrid pixel-based background model; background subtraction.

Paper 110960 received Aug. 9, 2011; revised manuscript received Nov. 16, 2011; accepted for publication Dec. 12, 2011; published online Mar. 2, 2012.

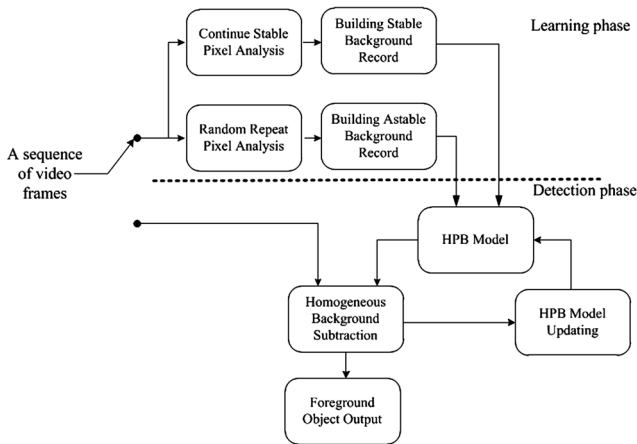
## 1 Introduction

Foreground object detection in a complex scene is an important step in many computer vision applications, such as visual surveillance,<sup>1-3</sup> intelligent behavior recognition<sup>4-7</sup> and vehicle motion tracking.<sup>8-12</sup> It is always desirable to accurately extract the foreground object(s). These applications are highly dependent on the results of foreground object detection. A common and efficient approach for extracting foreground objects from complex background is using background subtraction methods.<sup>13-16</sup> Background subtraction methods detect foreground objects by comparing each current frame with its background model. A difficulty with the background subtraction method is that complex scenes are usually dynamic. Complex scenes could be caused by waving trees, falling rain, illumination changes, and other background changes. To process these complex scenes, the technology of a robust background model is crucial.

Early background models<sup>17-20</sup> had the advantage of low memory consumption and high processing speed. These approaches work well with stationary scenes, but usually they cannot handle complex scenes properly. Therefore, a number of background modeling methods have been proposed, and the most common one is the Mixture of Gaussians Model (MoG).<sup>21</sup> By using more than one Gaussian distribution per pixel, it is able to handle complex scene, but MoG consumes more memory space and processing time. Therefore, some methods<sup>22-24</sup> proposed to improve this drawback. Instead of MoG, Kim<sup>13</sup> presented a real-time object detection algorithm based on codebook model that is efficient in either memory or processing time, but

it does not take into account dependence between adjacent pixels. Heikkila and Pietikainen<sup>25</sup> presented a novel approach to background subtraction, in which the background is modeled by texture features. It is capable of real-time processing under  $160 \times 120$  image size. Wang<sup>3</sup> presented a background modeling method called SACON that computes sample consensus and estimates a statistical model of the background scene. Chan and Chien<sup>26</sup> used a multi-background registration technique to calculate weight value for each pixel to update their background model. According to the weight value, the updating mechanism determines whether the pixel is replaced or not, thus it consumes less memory and computation time. Chiu<sup>27</sup> proposed a fast background model construct algorithm that improved the original weighting average method. It uses a probability-based algorithm to construct the background model and to detect the object. This approach works well with slight background changes, but it needs connect-component labeling methods to overcome the challenges of complex scenes.

In recent years, video compression technology and neural networks have been used to solve many problems in video surveillance. Wang<sup>28</sup> proposed a background modeling approach in the frequency domain, which constructs background model using DCT coefficients to achieve lower processing time. However, it has difficulty handling complex scenes. Maddalean and Petrosino<sup>29</sup> presented a new self-organizing method for modeling background by learning motion patterns that was employed to model complex scenes. This method records the weight vector at each pixel by using a large memory to store the neuronal map. Tsai<sup>30</sup> proposed a Fourier spectrum background model that can adapt to illumination changes, but it is only suitable for indoor work and grayscale images. Ref. 31 proposed a



**Fig. 1** The block diagram of the proposed foreground object detection.

background subtraction algorithm called Vibe that can be initialized with a single frame. This algorithm was embedded in a digital camera which has a low speed ARM processor. Ref. 32 presents a real-time approach for background subtraction which can overcome gradual and sudden illumination changes. This approach segments each pixel by using a probabilistic mixture-based and non-parametric model.

In this paper, a hybrid background modeling approach is based on stable and multi-layer astable records. It can effectively offer foreground object detection in complex scenes. Our approach is applicable to those background pixels which vary over time. In the detection phase, it takes into account the dependence of adjacent pixels in the astable background

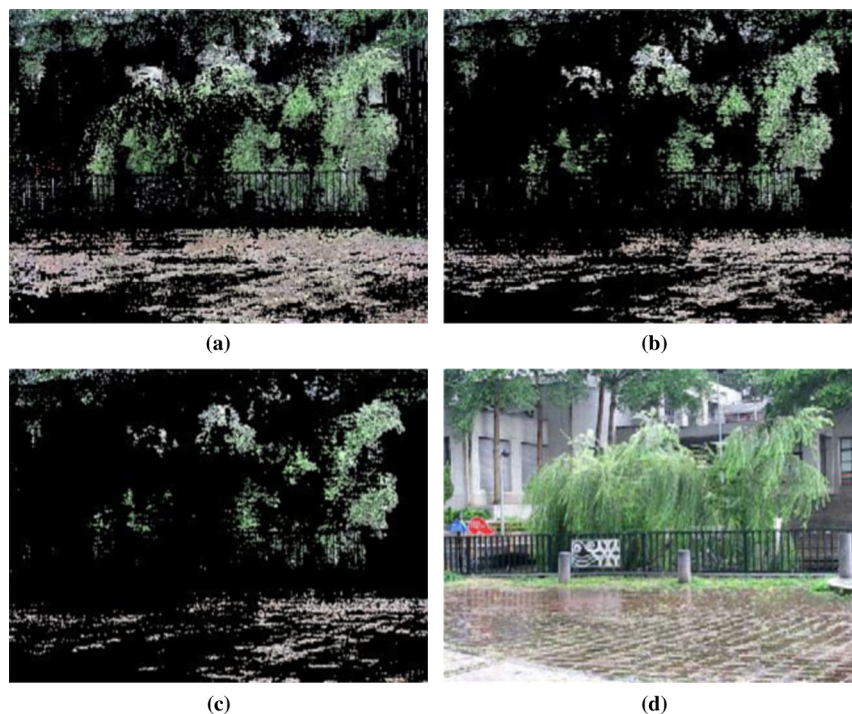
record by using homogeneous background subtraction; therefore, we can extract the foreground object with a low error rate. In this way, our hybrid pixel-based background (HPB) model and detection method are resistant against erroneous detection in the complex scene.

## 2 Proposed HPB Model and Homogenous Background Subtraction

In our proposed foreground object detection system, the block diagram is illustrated in Fig. 1, which includes learning and detection phases. In the beginning, the video input is switched to the learning phase. According to the pixel variation and analysis, the steady pixels are kept in a stable record and the varying pixels are saved by the multi-layer astable records. After using the above records to construct an HPB model completely, the system switches the input sequence into the detection phase and starts to perform the foreground object detection.

### 2.1 Creating a Stable Background Record

In the learning phase, stable pixel analysis is applied to make a stable background record (SBR). For a video sequence with  $n \times n$ -pixel frame size, let  $x_{i,j}(t)$  be denoted as the pixel at location  $(i, j)$  in RGB at time  $t$ , as shown in Eq. (1). Equation (5) is used to assess the similarity between two pixel  $x_{i,j}(t)$  and  $x_{i,j}(t-1)$ . These two pixels are regarded as similar if their difference is smaller than threshold value ( $Th = 30$ ). Stable Time,  $ST_{i,j}(t)$ , is used to store the duration of those pixels that are alike; a value of  $ST_{i,j}(t)$  stands for the pixel has unchanged for a period of time. If  $x_{i,j}(t)$  is not similar to  $x_{i,j}(t-1)$ , Stable Time will be reset to 0 and counting is resumed. Thus, the value of  $ST_{i,j}(t)$



**Fig. 2** HPB model at 300th frame: The stable background record (a), and the astable backgrounds  $ABR^1$  (b),  $ABR^2$  (c), and  $ABR^3$  (d) for the dynamic scene.

implies the stability of a pixel, and stable background record can be built according to Eqs. (1) and (2),

$$x_{i,j}(t) = [R_{i,j}(t), G_{i,j}(t), B_{i,j}(t)], \quad (1)$$

$$ST_{i,j}(t) = \begin{cases} ST_{i,j}(t-1) + 1 & \text{if } |x_{i,j}(t) - x_{i,j}(t-1)| \leq Th \\ 0, & \text{else} \end{cases}, \quad (2)$$

where  $Th$  is a threshold for evaluating the similarity. Thus, a stable background can be built when the value of  $ST_{i,j}(t)$  is large enough. The stable background record ( $SBR_{i,j}$ ) can be obtained according to Eq. (3).

$$SBR_{i,j} = \begin{cases} x_{i,j}(t), & \text{if } ST_{i,j}(t) \geq Th_S \\ \phi, & \text{else} \end{cases}. \quad (3)$$

The  $SBR_{i,j}$  is a empty set when the stable time  $ST_{i,j}(t)$  of a pixel is less than  $Th_S$ . The pixel is stable for a long period of time if  $ST_{i,j}(t)$  is larger than or equal to  $Th_S$ . Thus, the  $SBR_{i,j}$  can obtain the value of  $x_{i,j}(t)$ .

## 2.2 Constructing the Astable Background Record

Building the astable background record (ABR) is important since a stable background record is not sufficient to obtain a foreground object precisely. The astable background record consists of multi-layer 2-dimension buffers to store the complex scene. In the learning phase, the record  $ABR_{i,j}^u$  stands for one background pixel stored in  $(i,j)$  buffer of the  $u$ th layer buffer. Let  $MC_{i,j}^u$  represent the match counter to count how many  $x_{i,j}(t)$  is matched to  $ABR_{i,j}^u$ . The multi-layer astable background record is gradually established frame by frame.

Step 1: Initialization at  $t = 1$ ;

$$ABR_{i,j}^1 = x_{i,j}(1)$$

Step 2: Major learning for the pixel of frames from  $t = 2$  to  $N$ ;  
for  $i,j = 1, 2, \dots, n$

Step 2.1: Find  $x_{i,j}(t)$  matching to the  $ABR_{i,j}^u$ , while  $ABR_{i,j}^u \neq \Phi$   
if  $|x_{i,j}(t) - ABR_{i,j}^u| \leq Th$   
then,  $MC_{i,j}^u = MC_{i,j}^u + 1$ ;  
end

Step 2.2: If there is no match, save the input pixel into new layer  $ABR_{i,j}^u$ .  
 $ABR_{i,j}^u = x_{i,j}(t)$  end

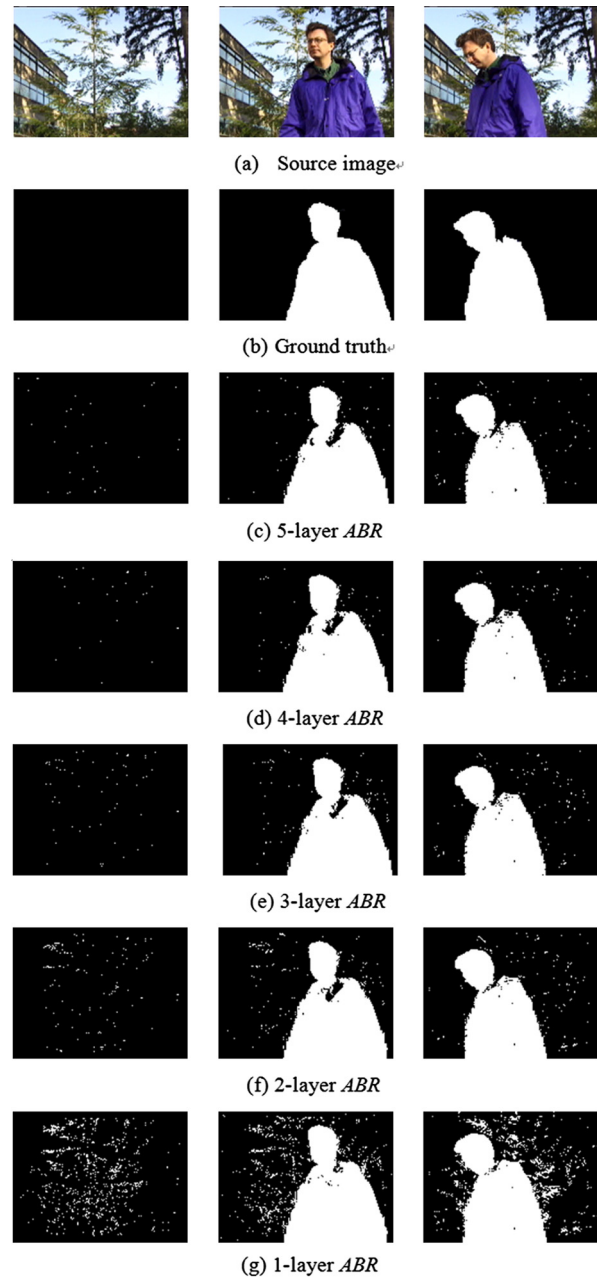
Step 3: Release the  $ABR_{i,j}^u$  based on the criteria (a) or (b)

- (a)  $MC_{i,j}^u < Th_f$
- (b)  $|ABR_{i,j}^u - SBR_{i,j}^u| < Th$

In the beginning frame, the input pixel  $x_{i,j}(1)$  is stored in the first layer of the  $ABR_{i,j}^1$ . For the second frame, the pixel  $x_{i,j}(2)$  is compared with the corresponding  $ABR_{i,j}^1$ . If they are similar, then  $MC_{i,j}^1$  will be increased. If there is no match, then  $x_{i,j}(2)$  will be stored in the second layer of  $ABR_{i,j}^2$ , and so on. At the end of learning phase, we have to delete the useless  $ABR_{i,j}^u$  which meet criteria (a) or (b) to reduce the

memory requirements. When  $MC_{i,j}^u$  is less than  $Th_f$  ( $Th_f = 15$ ), it means that this  $ABR_{i,j}^u$  could be a foreground element appearing temporarily or noise. If  $ABR_{i,j}^u$  is similar to  $SBR_{i,j}$ , the corresponding  $ABR_{i,j}^u$  should be deleted to save memory space.

Figure 2 shows an example of finding a stable background record (SBR) and 3-layer ABR. Figure 2(a) illustrates a static background record after 300 learning frames. Figure 2(b) shows the  $ABR^1$ , in which shaking leaves, falling rain, and water were recorded correctly as an astable background. Figures 2(c) and 2(d), which show the other shaking leaves, falling rain, are the second and the third layers of the astable background record, respectively.



**Fig. 3** Foreground detection results of 1-, 2-, 3-, 4-, and 5-layer ABR: (a) source image, (b) ground truth, (c) 5-layer ABR, (d) 4-layer ABR, (e) 3-layer ABR, (f) 2-layer ABR, (g) 1-layer ABR.



### 2.3 Foreground Object Detection with Homogenous Background Subtraction

After the construction of HPB model, foreground objects can be obtained by background subtraction. However, according to Figs. 2(b)–2(d), it can be observed that the astable background records have composition of homogeneous blob movements in the shaky area. In order to reduce detection error and save recording memory, the characteristic of homogeneity in an area has to be taken into account while performing the background subtraction. Thus, the input  $x_{ij}(t)$  is compared to the neighbors  $ABR_{i\pm s, j\pm p}^u$  in area  $((s, p)1, 2, \dots, r)$ . The neighboring area can be  $r \times r$  pixels and centered at  $x_{ij}(t)$ . A foreground object (FO) can be detected by homogenous background subtraction, as in Eq. (4):

$$FO_{ij}(t) = \begin{cases} 0, & \text{if } |x_{ij}(t) - SBR_{ij}| \leq Th \\ 0, & \text{if } |x_{ij}(t) - ABR_{i\pm s, j\pm p}^u| \leq Th \\ 1, & \text{else} \end{cases}, \quad (4)$$

where  $s, p = 1, 2, \dots, r$  and  $u = 1, 2, \dots, m$ . While FO is equal to 1, it stands for foreground object pixel; otherwise, it represents background.

### 3 Background Updating

In the detection phase, we must update the HPB model over time to prevent detection errors resulting from outdated background information. Since the SBR and ABR are established in different ways, methods for updating them are different as well. The SBR is composed of stable pixels, therefore only background information updates are needed. On the other hand, updating the background information for ABRs needs a more complex replacement mechanism.

In Eq. (5), when  $SBR_{ij}$  and  $x_{ij}(t)$  match, we use the running average to update the corresponding  $SBR_{ij}$ ,

$$SBR_{ij} = \alpha \times x_{ij}(t) + (1 - \alpha) \times SBR_{ij}, \quad (5)$$

$$\text{if } |SBR_{ij} - x_{ij}(t)| < Th,$$

where  $\alpha$  is a constant whose value is less than 1.

Similarly, as in Eq. (6), when  $x_{ij}(t)$  and  $ABR_{ij}^u$  match, we also use the running average to update the ABR. However, when  $x_{ij}(t)$  and  $ABR_{ij}^u$  do not match, we use an exponential

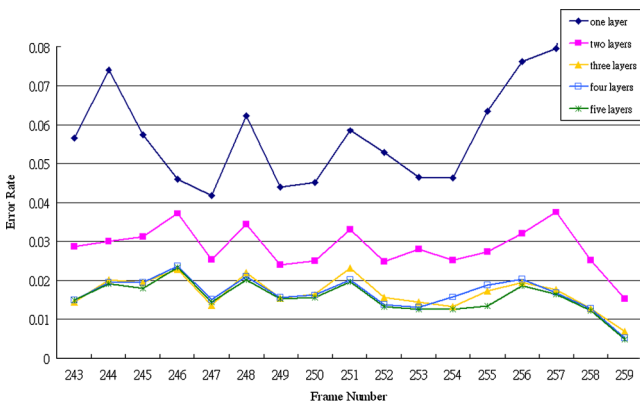


Fig. 4 Error rates of 1-, 2-, 3-, 4-, and 5-layer ABRs.

distribution probability density function to determine if  $ABR_{ij}^u$  should be replaced. Thus,  $pr_{ij}^u$  represents the probability of whether or not the replacement should occur. The ABR update and replacement procedure is as follows: Compare the input pixel  $x_{ij}(t)$  with  $ABR_{ij}^u$  for all  $ij, u$ .



Fig. 5 Object detection results in a moving background: (a) source image, (b) ground truth, (c) Codebook, (d) MoG (Wu), (e) SACON, (f) Chien, (g) ViBe, (h) Proposed.

Step 1: If  $x_{i,j}(t)$  matches to  $ABR_{i,j}^u$  ( $|ABR_{i,j}^u - x_{i,j}(t)| < Th$ ).

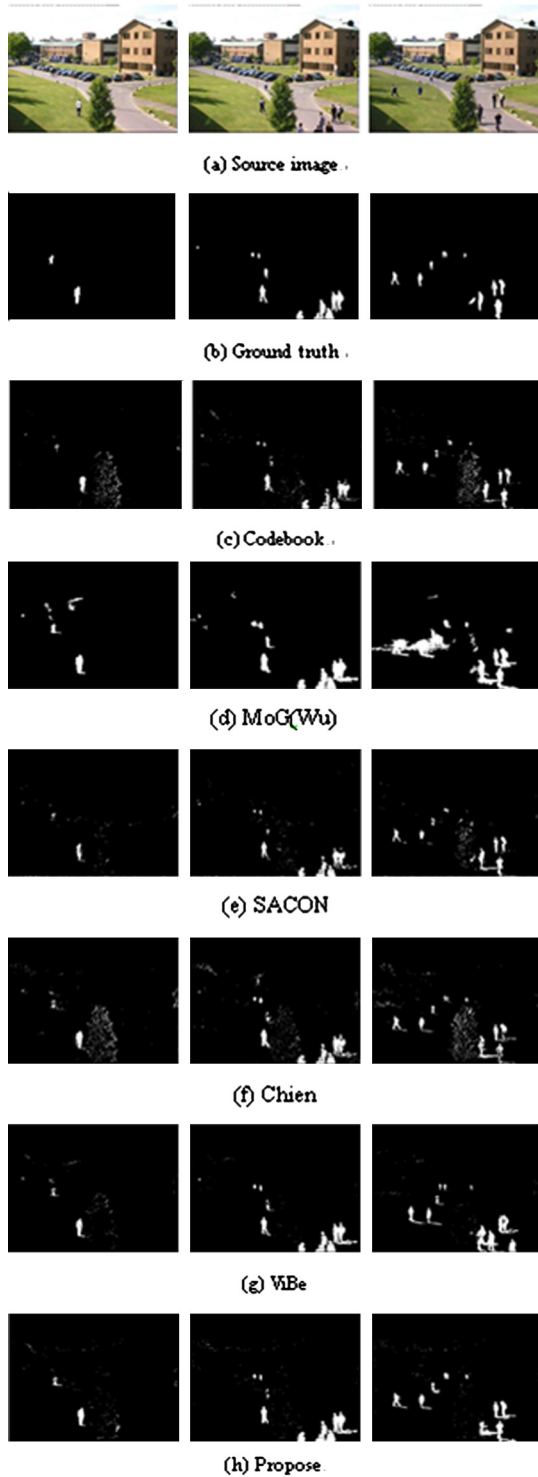
$$ABR_{i,j}^u = \alpha \times x_{i,j}(t) + (1 - \alpha) \times ABR_{i,j} \quad (6)$$

Step 2: If  $x_{i,j}(t)$  not match to  $ABR_{i,j}^u$ .

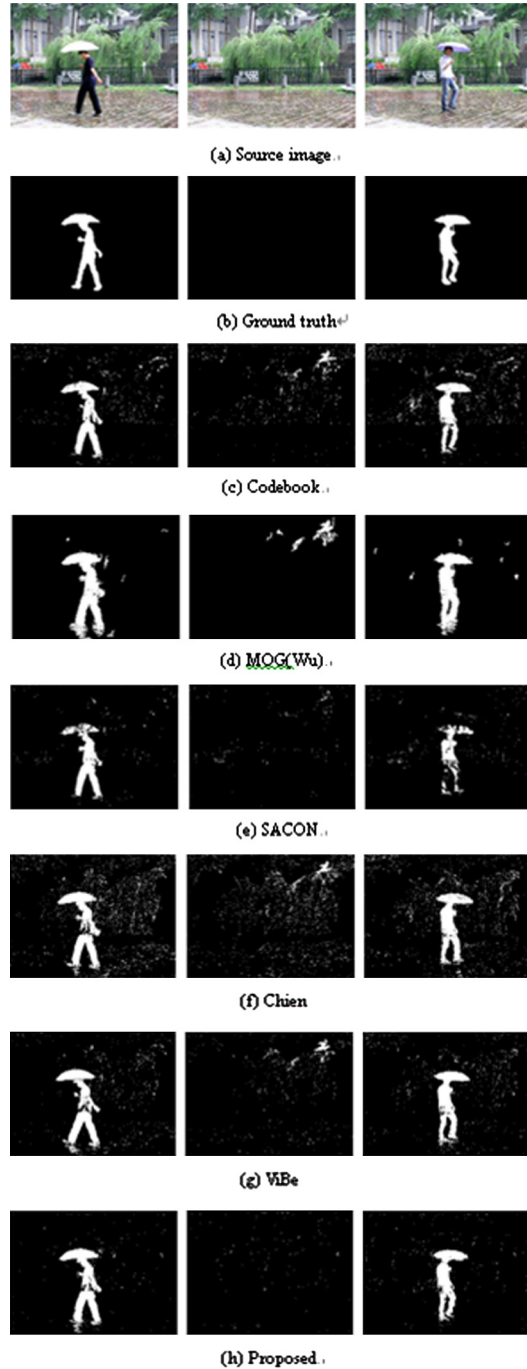
Step 2.1: Find the minimum probability value of  $ABR_{i,j}^u$ .

$$k = \operatorname{argmin}\{pr_{i,j}^u | u = 1, 2, 3, L, m\}$$

Step 2.2: Delete the minor probability value in  $ABR_{i,j}^u$ .  
 if  $pr_{i,j}^u < Th_p$   
 $ABR_{i,j}^k = x_{i,j}(t)$   
 end



**Fig. 6** Object detection results with illumination changes: (a) source image, (b) ground truth, (c) Codebook, (d) MoG (Wu), (e) SACON, (f) Chien, (g) ViBe, (h) Proposed.



**Fig. 7** Object detection in the raining sequence: (a) source image, (b) ground truth, (c) Codebook, (d) MoG (Wu), (e) SACON, (f) Chien, (g) ViBe, (h) Proposed.

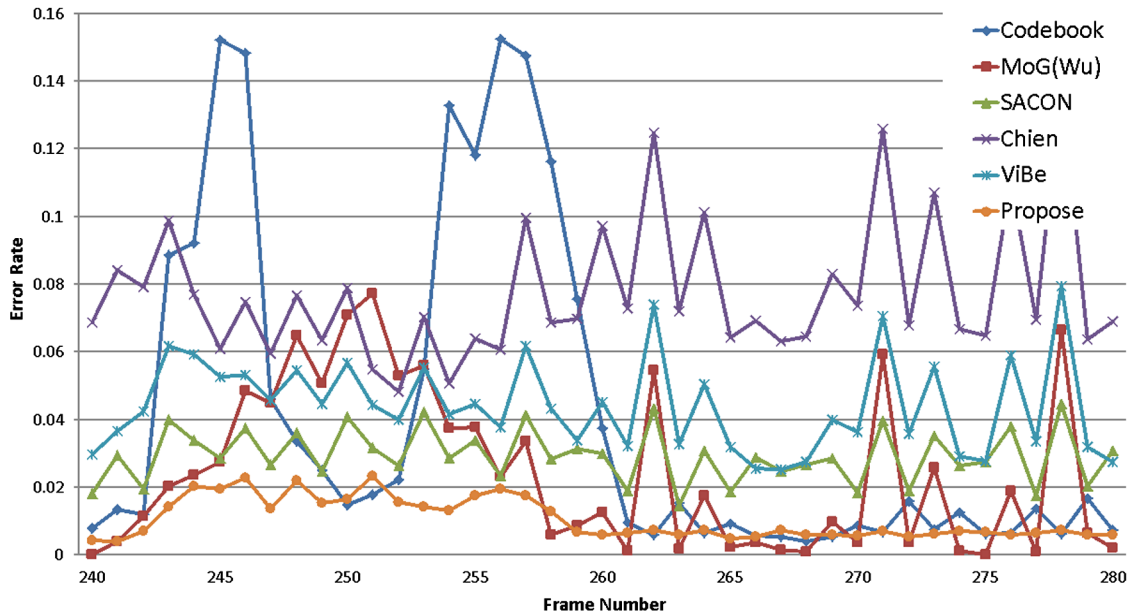


Fig. 8 Error rate in the waving tree sequence.

where  $Th_p$  is a threshold for probability. And then, the probability value can be obtained as

$$pr_{ij}^u(t) = \frac{MC_{ij}^u}{N} \times \exp\left(\frac{-MC_{ij}^u}{N} \times \Pi_{ij}^u(t)\right) \quad (7)$$

$u = 1, 2, \dots, m,$

where  $MC_{ij}^u$  is obtained in learning phase and  $N$  is the total number of learning frames. If  $x_{ij}(t)$  does not match to  $ABR_{ij}^u$ , the  $\Pi_{ij}^u(t)$  will be increased by one, and then the

$pr_{ij}^u$  will decrease exponentially.  $\Pi_{ij}^u(t)$  is defined as a counter that records the time interval of  $ABR_{ij}^u$  matching to  $x_{ij}(t)$ . It can be obtained as in Eq. (8),

$$\Pi_{ij}^u(t) = \begin{cases} \Pi_{ij}^u(t-1) + 1, & \text{if } |x_{ij}(t) - ABR_{ij}^u| \\ 0, & \text{else} \end{cases} \quad (8)$$

$u = 1, 2, \dots, m,$

where  $\Pi_{ij}^u(0) = 0$ . Thus, a large value of  $\Pi_{ij}^u(t)$  means that  $x_{ij}(t)$  and  $ABR_{ij}^u$  are not matched for a long period of time. It is resumed while  $x_{ij}(t)$  and  $ABG_{ij}^u$  are good match.

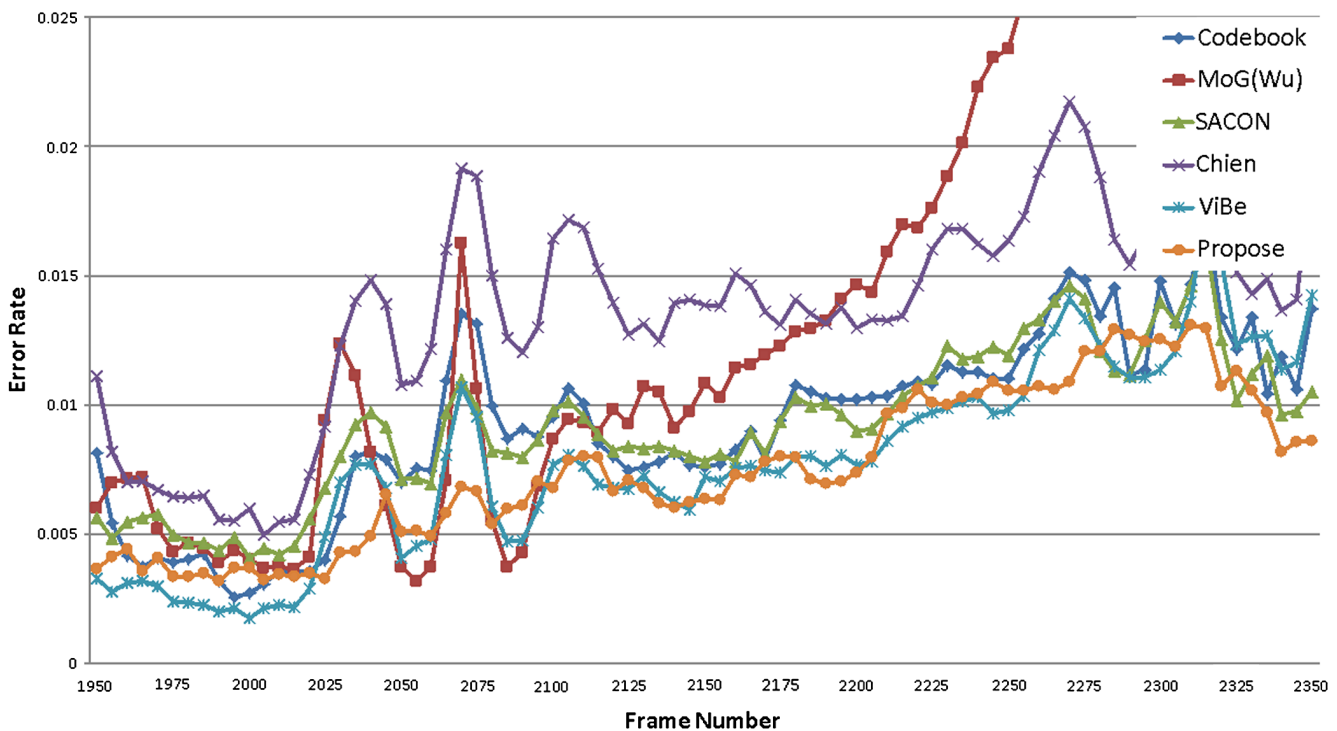


Fig. 9 Error rate in the PETS'2001 sequence.

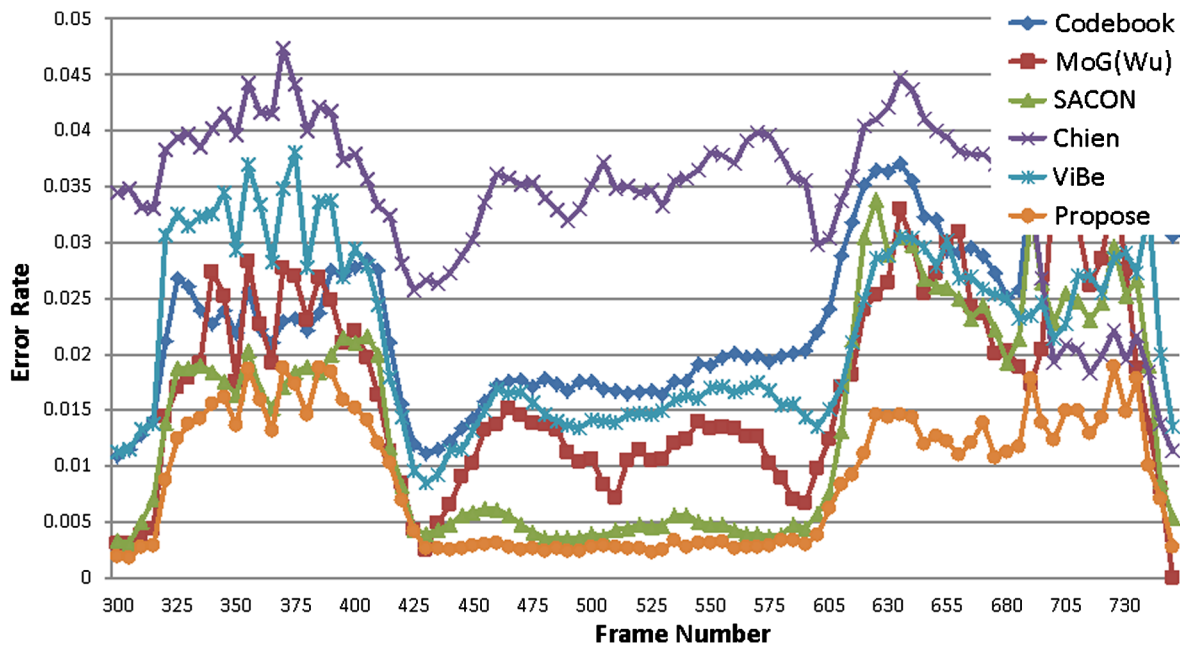


Fig. 10 Error rate in the raining sequence.

Table 1 Average of error rate in various benchmarks.

Video sequence	Codebook	MoG (Wu)	SACON	Chien	ViBe ( $N = 20$ )	Proposed
Waving tree( $160 \times 120$ )	4.11%	2.42%	2.93%	7.75%	4.41%	1.05%
Raining( $320 \times 240$ )	2.49%	1.56%	1.31%	3.19%	2.14%	0.81%
PETS 2001( $768 \times 576$ )	0.93%	1.73%	0.92%	1.37%	0.78%	0.75%

#### 4 Experimental Results and Comparison

To evaluate the performance of background subtraction, three test video sequences including waving trees, torrential rain and wind, and PETS'2001 Dataset 3<sup>33</sup> were used in the experiments. The performance of the proposed method was compared with that of Codebook,<sup>13</sup> MOG (Wu),<sup>23</sup> SACON,<sup>3</sup> Chien,<sup>26</sup> and ViBe.<sup>31</sup> A pixel-based error rate based on ground truth is a fair and often adapted assessment method,<sup>27</sup> and was used to evaluate each method's performance. The error rate is given in Eq. (9),

$$\text{Error Rate} = \frac{fp + fn}{\text{frame size}}, \quad (9)$$

where  $fp$  and  $fn$  are the sum of all false positives and the sum of false negatives, respectively. A smaller error rate means that the detected result is more similar to the ground truth.

It is important to choose a proper number of background layers to trade off between the hardware memory and the number of ABR layers based on the scene requirements. Figures 3(c)–3(g) show the foreground detection results for the waving trees video images, based on our proposed HPB model with 1 to 5-layer ABRs. In Fig. 4, the error rates of the 3-layer, 4-layer, and 5-layer ABRs are small. However, the 3-layer ABR uses much less memory.

In Figs. 5–7, we demonstrate that the proposed approach exhibits better foreground detection than the other methods in three benchmarks. Furthermore, Figs. 8–10 demonstrate that the proposed method presents a lower error rate in the ground truth comparison. The average error rates of the results from the six methods for various sequences are depicted in Table 1. It shows that the proposed approach has lower average error rate than the other methods.

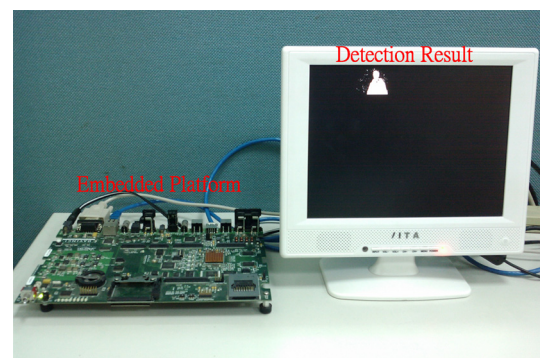


Fig. 11 Implementation of the embedded system.



**Table 2** Memory comparison of background models.

Video sequence	Codebook	MoG(Wu)	SACON	Chien	ViBe(N = 20)	Proposed
Waving tree(160 × 120)	1638 KB	1920 KB	1459 KB	1094 KB	1152 KB	1056 KB
Raining(320 × 240)	5330 KB	7680 KB	5836 KB	4377 KB	4608 KB	4224 KB
PETS 2001(768 × 576)	26.5 MB	44.2 MB	33.6 MB	25.2 MB	26.5 MB	24.3 MB

As shown in Fig. 11, we use the TI TMS320DM6446 Davinci development kit as our development platform which has a dual-core device including ARM926EJ-S and C64x+TM DSP. The resources of an embedded platform are limited, so the implementation has to consider memory consumption. Table 2 lists the real memory utilization for all six methods when applied to the different video sequences. It shows the memory requirement of our proposed method is much less than other approaches; thus, our approach can achieve the real-time operation with 23 frames per second for the waving trees video. The proposed method is suitable for implementation in an embedded platform.

## 5 Conclusions

An efficient and precise foreground object detection method was proposed in this paper. The proposed method applies a stable background record and multi-layer astable background records to construct a correct background model. While when more layers are used, more background information can be recorded to improve the precision, it also needs more memory as well as more calculation effort. Thus, it is important to choose a proper number of background layers to trade off between the memory load and the number of dynamic background layers required by the scene. To save more memory space and calculation time, the 3-layer dynamic background model was used in our approach. According to our experimental results, the error rates of the 3-layer, 4-layer, and 5-layer HPB models are similar in many video benchmarks. The results demonstrate that the proposed method has a lower error rate with ground truth than the five other models tested. Furthermore, the proposed approach has higher precision of object detection than other methods for various sequences. The final verification was done using a 2.66 GHz CPU with a video resolution of 768 × 576 and an execution speed of 21 frames/second in complex background scene. In addition, the proposed method can achieve real-time for complex scene on a Davinci embedded platform.

## Acknowledgments

We would like to thank Oliver Barnich and M. Van Droogenbroeck, who provided the C-like source code for his algorithms.

## References

- C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. IEEE Conf. Computer Vision and Pattern Recogn.*, Vol. 2, pp. 246–252 (1999).
- A. Cavallaro, O. Steiger, and T. Ebrahimi, "Tracking video objects in cluttered background," *IEEE Trans. Circuits Syst. Video Technol.* **15**(4), 575–584 (2005).
- H. Wang and D. Suter, "A consensus-based method for tracking modeling background scenario and foreground appearance," *Pattern Recogn.* **40**(3), 1091–1105 (2006).
- J. W. Hsieh et al., "Video-based human movement analysis and its application to surveillance systems," *IEEE Trans. Multimedia* **10**(3), 372–384 (2008).
- C. F. Juang et al., "Computer vision-based human body segmentation and posture estimation," *IEEE Trans. Syst. Man Cybernet. A: Syst. Hum.* **39**(1), 119–133 (2009).
- W. Lao, J. Han, and P. H. N. de With, "Automatic video-based human motion analyzer for consumer surveillance system," *IEEE Trans. Consumer Electron.* **55**(22), 591–598 (2009).
- C. H. Chuang et al., "Carried object detection using ration histogram and its application to suspicious event analysis," *IEEE Trans. Circuits Syst. Video Technol.* **19**(6), 911–916 (2009).
- S. Gupte et al., "Detection and classification of vehicles," *IEEE Trans. Intel. Transport. Syst.* **3**(1), 37–47 (2002).
- S. C. Chen et al., "Learning-based spatio-temporal vehicle tracking and indexing for transportation multimedia database systems," *IEEE Trans. Intel. Transport. Syst.* **4**(3), 154–167 (2003).
- J. W. Hsieh et al., "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Trans. Intel. Transport. Syst.* **7**(2), 175–187 (2006).
- W. Zhang and Q. M. Jonathan Wu, "Moving vehicles detection based on adaptive motion histogram," *Digital Signal Process.* **20**(3), 793–805 (2010).
- N. Buch, J. Orwell, and S. A. Velastin, "Detection and classification of vehicles for urban traffic scenes," in *Int. Conf. Vis. Inform. Eng.*, pp. 182–187 (2008).
- K. Kim et al., "Real-time foreground-background segmentation using codebook model," *Real-Time Imag.* **11**(3), 172–185 (2005).
- P. Spagnolo et al., "Moving object segmentation by background subtraction and temporal analysis," *Image Vis. Comput.* **24**(5), 411–423 (2006).
- J. C. Nascimento and J. S. Marques, "Performance evaluation of object detection algorithms for video surveillance," *IEEE Trans. Multimedia* **8**(4), 761–774 (2006).
- P. M. Jodoin, M. Mignotte, and J. Konrad, "Statistical background subtraction using spatial cues," *IEEE Trans. Circuits Syst. Video Technol.* **17**(12), 1758–1763 (2007).
- C. R. Wren et al., "Pfinder: real-time tracking of the human body," *IEEE Trans. Pattern Anal. Mach. Intel.* **19**(7), 780–785 (1997).
- I. Haritaoglu, D. Harwood, and L. S. Davis, "W<sup>4</sup>: real-time surveillance of people and their activities," *IEEE Trans. Pattern Anal. Mach. Intel.* **22**(8), 809–830 (2000).
- C. Kim and J. N. Hwang, "Object-based video abstraction for video surveillance systems," *IEEE Trans. Circuits Syst. Video Technol.* **12**(12), 1128–1138 (2002).
- S. Y. Chien, S. Y. Ma, and L. G. Chen, "Efficient moving object segmentation algorithm using background registration technique," *IEEE Trans. Circuits Syst. Video Technol.* **12**(7), 577–586 (2002).
- C. Stauffer and W. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Mach. Intel.* **22**(8), 747–757 (2000).
- D. S. Lee, "Effective Gaussian mixture learning for video background subtraction," *IEEE Trans. Pattern Anal. Mach. Intel.* **27**(5), 827–832 (2005).
- H. H. P. Wu et al., "Improved moving object segmentation by multi-resolution and variable thresholding," *Opt. Eng.* **45**(11), 117003 (2006).
- J. Cheng et al., "Flexible background mixture models for foreground segmentation," *Image Vis. Comput.* **24**(5), 473–482 (2006).
- M. Heikkila and M. Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Pattern Anal. Mach. Intel.* **28**(4), 657–662 (2006).
- W. K. Chan et al., "Efficient content analysis engine for visual surveillance network," *IEEE Trans. Circuits Syst. Video Technol.* **19**(5), 693–703 (2009).
- C. C. Chiu, M. Y. Ku, and L. W. Liang, "A robust object segmentation system using a probability-based background extraction algorithm," *IEEE Trans. Circuits Syst. Video Technol.* **20**(4), 518–528 (2010).

28. W. Wang, J. Yang, and W. Gao, "Modeling background and segmenting moving objects from compressed video," *IEEE Trans. Circuits Syst. Video Technol.* **18**(5), 670–681 (2008).
29. L. Maddalean and A. Petrosino, "A self-organizing approach to background subtraction for visual surveillance applications," *IEEE Trans. Image Process.* **17**(7), 1168–1177 (2008).
30. D. M. Tsai and W. Y. Chiu, "Motion detection using Fourier image reconstruction," *Pattern Recogn. Lett.* **29**(16), 2145–2155 (2008).
31. O. Branich and M. Van Droogenbreck, "ViBe : a universal background subtraction algorithm for video sequences," *IEEE Trans. Image Process.* **20**(6), 1709–1724 (2011).
32. A. Lanza, S. Salti, and L. Di Stefano, "Background subtraction by non-parametric probabilistic clustering," in *IEEE Int. Conf. Adv. Video Signal-Based Surveil.*, IEEE Society, Klagenfurt, pp. 243–248 (2011).
33. PETS'2001 Dataset 3. <http://www.cvg.cs.rdg.ac.uk/PETS2001/pets2001-dataset.html> Last modified 6 July (2009).



**Wen-Kai Tsai** is a PhD candidate at Graduate School of Engineering Science and Technology, National Yunlin University of Science and Technology, Taiwan. He received BS and MS degrees in electronics engineering from National Yunlin University of Science and Technology, Taiwan, in 2004 and 2006, respectively. His research interests include digital signal processing and image processing.



**Chung-Chi Lin** received the MS degree in computer science from University of Houston, TX, USA, in 1983, and the PhD degree in engineering science and technology from National Yunlin University of Science & Technology, Taiwan, in 2009. Since 2009, he has been an Associate Professor with the Department of Computer Science, Tunghai University, Taiwan. His current research interests include image processing, digital signal processing, and System-on-chip design.



**Ming-Hwa Sheu** received the BS degree in electronics engineering from National Taiwan University of Science and Technology, Taipei, Taiwan, in 1986, and the MS and PhD degrees in Electrical Engineering from National Cheng Kung University, Tainan, Taiwan, in 1989 and 1993, respectively. From 1994 to 2003, he was an Associate Professor at National Yunlin University of Science and Technology, Touliu, Taiwan. Currently, he is a Professor in the Department of Electronics Engineering, National Yunlin University of Science and Technology. His research interests include CAD/VLSI, digital signal processing, algorithm analysis, and system-on-chip design.