Mixture of Gaussians-Based Background Subtraction for Bayer-Pattern Image Sequences

Jae Kyu Suhr, *Student Member, IEEE,* Ho Gi Jung, *Senior Member, IEEE,* Gen Li, and Jaihie Kim

***Abstract*—This letter proposes a background subtraction method for Bayer-pattern image sequences. The proposed method models the background in a Bayer-pattern domain using a mixture of Gaussians (MoG) and classifies the foreground in an interpolated red, green, and blue (RGB) domain. This method can achieve almost the same accuracy as MoG using RGB color images while maintaining computational resources (time and memory) similar to MoG using grayscale images. Experimental results show that the proposed method is a good solution to obtain high accuracy and low resource requirements simultaneously. This improvement is important for a low-level task like background subtraction since its accuracy affects the performance of high-level tasks, and is preferable for implementation in real-time embedded systems such as smart cameras.**

***Index Terms*—Background subtraction, Bayer color filter array, mixture of Gaussians (MoG), visual surveillance.**

I. Introduction

Moving object segmentation is an active research topic in a visual surveillance area. Background subtraction is one of the most widely used techniques to segment moving objects for static cameras [1]–[3]. Since background subtraction is a low-level task, it should consider two aspects: accuracy and computational resources (time and memory). First, its accuracy is critical because the output of the background subtraction is used for other high-level tasks, such as tracking and recognition. Erroneous output will affect the performances of these high-level tasks. Second, computational resources used for background subtraction are critical since the resources remaining after this low-level task should be used for high-level tasks, and is preferable as a means of implementing this task in real-time embedded systems such as smart cameras [4], [5]. Therefore, it is important for the background subtraction method to obtain high accuracy and low resource requirements at the same time.

Background subtraction performance depends mainly on the background modeling technique [6]. Extensive research has been carried out regarding this task [1]–[3], [6], [7]. Of this research, a mixture of Gaussians (MoG) using online K-means approximation [7] is one of the most popular methods [1]–[3] since it can cope with global changes (illumination or camera jitter) and periodic disturbances (swaying vegetation or flickering monitors). The method in [7] can be divided into two steps: background modeling and foreground classification. This method has been applied mostly to red, green, and blue (RGB) color and grayscale images. In cases using RGB color images, both background modeling and foreground classification are conducted in the RGB domain. Since these two steps are conducted in 3-D space (RGB), its computational cost and memory requirement are relatively large. However, it can achieve high foreground segmentation accuracy due to its color information. In cases using grayscale images, both background modeling and foreground classification are conducted in the grayscale domain. Since these two steps are conducted in 1-D space (intensity), its computational cost and memory requirement are relatively small. However, the foreground segmentation accuracy inevitably decreases due to the loss of color information.

To solve the problem of accuracy and resource requirements, this letter proposes a background subtraction method by using Bayer-pattern image sequences. The proposed method conducts background modeling in a Bayer-pattern domain using MoG and foreground classification in an interpolated RGB domain. By using this approach, we achieve almost the same accuracy as the method in [7] using RGB color images while maintaining computational resources similar to the method in [7] using grayscale images. Maintaining a good performance while reducing the computational resources of the method in [7] is important since its limitations in terms of computational resources are addressed in many papers, especially for real-time embedded systems [8], [9], and numerous researchers dealing with high-level tasks are still frequently using it [10]–[13]. There has been an attempt to use Bayer-pattern images for background subtraction [14], but it does not provide the detailed method description, explicit performance evaluation and analysis.

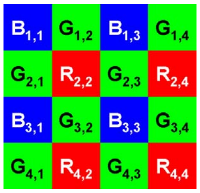
The main difference between the method in [7] using RGB color images and the proposed method is that the former conducts both background modeling and foreground classification in the RGB domain, although the latter conducts background modeling in a Bayer-pattern domain and foreground classification in an interpolated RGB domain. In this interpolated RGB domain, a pixel is classified by combining the information of

Manuscript received March 5, 2010; revised May 26, 2010; accepted July 2, 2010. Date of publication October 18, 2010; date of current version March 23, 2011. This work was supported by the National Research Foundation of Korea through the Biometrics Engineering Research Center, Yonsei University, under Grant R112002105070020(2010). This paper was recommended by Associate Editor B. Zeng.

The authors are with the School of Electrical and Electronic Engineering, Biometrics Engineering Research Center, Yonsei University, Seoul 120-749, Korea (e-mail: lfisbf@yonsei.ac.kr; hgjung@yonsei.ac.kr; leegeun@yonsei.ac.kr; jhkim@yonsei.ac.kr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2010.2087810



$B_{1,1}$	$G_{1,2}$	$B_{1,3}$	$G_{1,4}$
$G_{2,1}$	$R_{2,2}$	$G_{2,3}$	$R_{2,4}$
$B_{3,1}$	$G_{3,2}$	$B_{3,3}$	$G_{3,4}$
$G_{4,1}$	$R_{4,2}$	$G_{4,3}$	$R_{4,4}$

Fig. 1. Example of a Bayer CFA pattern.

three 1-D spaces (R, G, and B) rather than a single 3-D space (RGB). Due to this fact, the proposed method has two properties. First, this method has more chances to classify foreground as background because the background model includes false RGB combinations. However, the probability that a foreground pixel falls into these false background distributions is quite low when considering the whole 3-D RGB space. Second, this method can separately estimate variances of RGB components without increasing computational cost because one pixel has only one color component in Bayer-pattern images. This makes background modeling more accurate compared to the method in [7] which assumes that the variances of RGB components are the same for computational reasons.

In this experiment, the performance of the proposed method was quantitatively evaluated and compared with the method in [7] using three types of images (RGB color, grayscale, and Bayer-pattern images). The experimental results show that the proposed method produces similar or slightly higher accuracy compared to the method in [7] using RGB color images and requires almost the same computational resources as the case using grayscale images.

II. Method Description

A. Bayer-Pattern Image

A color image consists of three channels per pixel. Using three spatially aligned sensors to acquire color images has several disadvantages; it increases camera size and cost, and requires complicated pixel registration procedure. Consequently, most digital color cameras use a single image sensor with a color filter array (CFA) in front [15]. When using the CFA, each pixel measures only one color and spatially neighboring pixels which correspond to different colors are used to estimate unmeasured colors.

Among CFA patterns, the Bayer CFA pattern is one of the most widely used patterns [16]. As shown in Fig. 1, the Bayer CFA pattern is a 2 × 2 pattern which has two green components in diagonal locations and red and blue components in the other locations. An image produced by this pattern is called a Bayer-pattern image and the interpolation process to obtain a full-color image is called "demosaicing." One of the simplest demosaicing methods is bilinear demosaicing [17]. This method uses a bilinear interpolation to produce a full-color image. If this method is applied to the pixel location at (2, 2) in Fig. 1, green and blue values ($\tilde{G}_{2,2}$ and ($\tilde{B}_{2,2}$) at this pixel location are estimated by using

$$\tilde{G}_{2,2} = \left(G_{1,2} + G_{2,1} + G_{2,3} + G_{3,2}\right)/4$$

$$\tilde{B}_{2,2} = (B_{1,1} + B_{1,3} + B_{3,1} + B_{3,3})/4. \tag{1}$$

B. Mixture of Gaussian-Based Background Subtraction

This section briefly describes a mixture of the Gaussians-based background subtraction method proposed in [7]. This method describes the probability of observing a pixel value, X_t, at time t as follows:

$$P(X_t) = \sum_{i=1}^{k} \omega_{i,t}\eta\left(X_t, \mu_{i,t}, \Sigma_{i,t}\right) \tag{2}$$

where K is the number of Gaussians, which is usually set to be between 3 and 5. $\omega_{i,t}$, $\mu_{i,t}$, and $\Sigma_{i,t}$ are weight, mean, and the covariance matrix of the ith Gaussian in the mixture at time t, respectively. For computational efficiency, RGB pixel values are assumed to be independent and have the same variances. To update this model, the following online K-means approximation is used. Every new pixel value is checked against the K Gaussian distributions to determine whether this value is within 2.5 standard deviation of one of them. If none of the distributions includes this pixel value, the least probable distribution is replaced with a distribution whose mean, variance, and weight are set to the current pixel value, predetermined high variance, and low weight, respectively. The weights of the K distributions at time t are updated as follows:

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

where α is a learning rate, and $M_{k,t}$ is 1 for the distribution which includes the current pixel value within its 2.5 standard deviation and 0 for the other distributions. After updating the weights, they are renormalized to make their summation become one. The parameters of the distribution which includes the current pixel value within its 2.5 standard deviation are updated as follows:

$$\begin{aligned} \mu_{k,t} &= (1 - \rho)\mu_{k,t-1} + \rho X_t, \\ \sigma^2_{k,t} &= (1 - \rho)\sigma^2_{k,t-1} + \rho(X_t - \mu_{k,t})^T(X_t - \mu_{k,t}) \end{aligned} \tag{4}$$

where ρ is a learning factor for adapting distributions. The parameters of the other distributions remain the same. To decide whether X_t is included in the background distributions, the distributions are ordered by the value of $\omega_{k,t}/\sigma_{k,t}$ and the first B distributions which satisfy (5) are chosen as the background distributions as follows:

$$B = \arg\min_{b}\left(\sum_{k=1}^{b} \omega_{k,t} > T\right) \tag{5}$$

where T is a measure of the minimum portion of the data that should be accounted for by the background. If X_t is within 2.5 standard deviation of one of these B distributions, it is decided as a background pixel.

C. Proposed Method

The method in [7], mentioned in Section II-B, consists of two steps: background modeling and foreground classification. The proposed method conducts background modeling in a Bayer-pattern domain and foreground classification in an interpolated RGB domain. First, the background modeling procedure of this method is the same as the method in [7] except it is conducted in a Bayer-pattern domain so that X_t

and $\mu_{i,t}$ in (2) are scalar values rather than 3-D vectors. Second, the foreground classification procedure is conducted as follows. The means $(\mu_{b,t})$ and standard deviations $(\sigma_{b,t})$ of B distributions which satisfy (5) are chosen at each pixel location. The index (N) which gives a minimum Mahalanobis distance between X_t and $\mu_{b,t}$ is selected as follows:

$$N = \arg\min_{b} \left\{ abs\left(X_t - \mu_{b,t}\right)/\sigma_{b,t} \right\}. \tag{6}$$

After finding the index, the signed Mahalanobis distance (D_t) at that pixel location is calculated as follows:

$$D_t = \left(X_t - \mu_{N,t}\right)/\sigma_{N,t}. \tag{7}$$

The signed distance is calculated because it will be used for interpolation. Since each pixel has only one color component in Bayer-pattern images, D_t can be more explicitly notated by $D^{R}{}_{t}$, $D^{G}{}_{t}$ or $D^{B}{}_{t}$ depending on its pixel location. If D_t is assumed to be calculated at the pixel location assigned for the red channel, it can be represented by $D^{R}{}_{t}$. After obtaining the distance of the red channel $D^{R}{}_{t}$, the distances of the other two channels ($\tilde{D}^{G}{}_{t}$ and $\tilde{D}^{B}{}_{t}$) are estimated by interpolating the distances calculated from spatially neighboring pixels which correspond to different color channels (green and blue). For this interpolation process, the bilinear demosaicing technique mentioned in Section II-A is used. Finally, the pixel location of X_t is classified as background if the absolute values of all three signed distances (D_t^R, $\tilde{D}_t^G$, and $\tilde{D}_t^B$) are not larger than a predetermined threshold ($TH = 2.5$) as in (8). Otherwise, it is classified as foreground as follows:

$$X_t = \begin{cases} background, & abs\left(D_t^R\right) \le TH \wedge abs\,\tilde{D}_t^G \le TH \wedge abs\tilde{D}_t^B \le TH \\ foreground, & \text{otherwise.} \end{cases} \tag{8}$$

In this method, the Mahalanobis distance can be considered as the "backgroundness" of X_t from a view point of one channel. The backgroundness of this pixel from view points of the other two channels is estimated by interpolating the backgroundness of the spatially neighboring pixels.

The main difference between the method in [7] using RGB color images and the proposed method is that the former conducts the background modeling and foreground classification in an RGB domain, but the latter conducts the background modeling in a Bayer-pattern domain and foreground classification in an interpolated RGB domain. This difference can be explained in detail by using Fig. 2. In this figure, for convenience sake, the blue channel is omitted and the number of Gaussians which represents background distributions [B in (5)] is assumed to be two. Fig. 2(a) and (b) shows the background modeling results in the red-green domain and the interpolated red-green domain, respectively. As shown in Fig. 2(a), in the former case, combinations of red and green channels are known so that the background is modeled with two 2-D Gaussians. However, this combination cannot be known in the latter case since each pixel has a MoG for only one color and information of the other colors at that pixel location is interpolated from spatially neighboring pixels. Due to this fact, it can be said that the background is modeled with two 1-D Gaussians in each channel as shown in Fig. 2(b). Therefore, the decision boundary of the former case is defined with two circles as shown in Fig. 2(a) with solid lines, and that of the latter case is defined with four rectangles as shown in Fig. 2(b) with solid and dashed lines. The reason why the shape of the decision boundary is rectangle rather than square in Fig. 2(b) is because the variance of each channel is separately estimated in the proposed method. From this figure, two properties of the proposed method can be noticed: one is negative and the other is positive. The negative property is that this method produces false background regions as shown in Fig. 2(b) with two dashed rectangles. These regions are caused by incorrect combinations of 1-D Gaussians. Because of this property, the proposed method has more chances to classify the foreground as background. However, the probability that a foreground pixel falls into the false background regions is quite low when considering the whole 3-D RGB space because the variances of the Gaussians chosen as background distribution are usually quite small due to the Gaussian ordering and selection based on the value of $\omega_{k,t}/\sigma_{k,t}$ and (5). Also, experimental results show that the performance of the proposed method is almost the same as that of the method in [7] using RGB color images. The positive property of the proposed method is that it can separately estimate the variances of RGB channels without increasing computational costs. This can make the decision boundary of the proposed method more accurate than that of the method in [7] using RGB color images where these variances are assumed to be the same for computational reasons. Due to this property, the proposed method shows a slightly better performance compared to the method in [7] using RGB color images in the experiments.

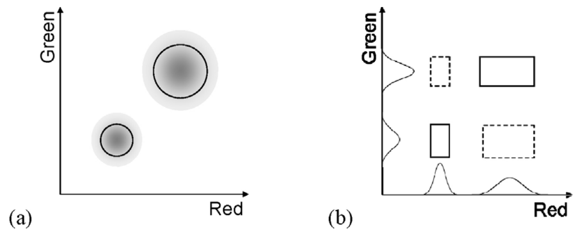


Fig. 2. Background distributions and decision boundaries. (a) In red-green domain. (b) In interpolated red-green domain.

The proposed method uses less computational resources compared to the method in [7] using RGB color images. In terms of computational costs, the proposed method conducts the background modeling in (4) with 1-D means and pixel values, but the method in [7] using RGB color images do the same operation with 3-D means and pixel values. Consequently, the proposed method requires less than half the computational costs of the method in [7] using RGB color images. Specifically, the proposed method requires 5 multiplications and 3 additions while the method in [7] using RGB color images requires 11 multiplications and 9 additions during the operation in (4). The computation of the distance interpolation is not considered here because the same operation is necessary to obtain RGB images from Bayer-pattern images. In terms of memory requirements, the proposed method requires $W \times H \times (3 \times K + 2)$ buffers ($W \times H$ for an input image, $W \times H \times K$ for means, $W \times H \times K$ for variances,

TABLE I
DESCRIPTION OF DATABASE

	Resolution (pixels)	No. of Images	No. of Ground truth	Environment	Source of Database
DB1	360 × 240	500	216	Outdoor	http://www.cs.cmu.edu/∼yaser
DB2	320 × 240	1501	194	Outdoor	http://web.eee.sztaki.hu/∼bcsaba
DB3	320 × 240	440	170	Outdoor	http://web.eee.sztaki.hu/∼bcsaba
DB4	320 × 240	300	113	Indoor	http://cvrr.ucsd.edu/aton/shadow
DB5	320 × 240	887	121	Indoor	http://web.eee.sztaki.hu/∼bcsaba
DB6	320 × 256	1286	20	Indoor	http://perception.i2r.a-star.edu.sg
DB7	320 × 240	2227	7	Outdoor	http://vision.gel.ulaval.ca/∼CastShadows
DB8	320 × 240	1800	7	Indoor	http://vision.gel.ulaval.ca/∼CastShadows
DB9	320 × 240	300	20	Indoor	Our own database
DB10	320 × 240	300	20	Indoor	Our own database
DB11	320 × 240	440	170	Outdoor	Noise-contaminated version of DB3
DB12	320 × 240	300	113	Indoor	Noise-contaminated version of DB4

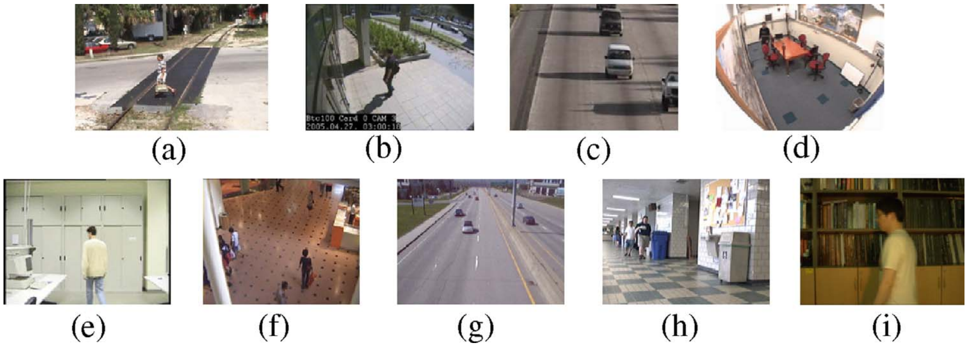


Fig. 3. Example images of databases. (a)–(h) Example images of DB1 to DB8, respectively. (i) Example image of DB9 and DB10.

TABLE II
FNRS AND FPRS (%)

	FNR				FPR			
	Proposed Method	RGB Color	Grayscale	Pseudo-Grayscale	Proposed Method	RGB Color	Grayscale	Pseudo-Grayscale
DB1	1.20	2.97	9.09	9.73	3.97	2.47	2.46	2.50
DB2	12.97	16.12	23.51	24.89	8.62	5.69	6.65	6.05
DB3	9.82	10.12	22.87	23.07	11.19	10.89	8.86	8.82
DB4	5.66	9.35	18.09	18.56	2.51	2.79	1.83	1.78
DB5	4.06	5.48	15.03	15.02	8.71	7.75	6.73	6.68
DB6	6.87	11.04	18.07	20.02	8.23	5.95	6.13	5.91
DB7	1.16	3.21	7.83	9.26	3.14	1.67	2.44	2.42
DB8	15.81	15.07	29.03	29.47	5.64	6.37	4.16	3.98
DB9	6.39	8.02	24.23	21.85	5.93	8.68	3.21	3.81
DB10	7.50	12.66	24.50	22.75	6.52	6.27	3.49	4.06
DB11	12.10	12.28	23.37	23.97	11.65	12.68	9.82	9.78
DB12	9.09	15.34	25.01	24.91	1.96	0.64	1.66	1.68
Average	7.72	10.14	20.05	20.29	6.51	5.99	4.79	4.79

and $W \times H \times K$ for weights and $W \times H$ for Mahalanobis distances). But the method in [7] using RGB color images requires $W \times H \times (5 \times K + 3)$ buffers ($W \times H \times 3$ for an input image, $W \times H \times K \times 3$ for means, $W \times H \times K$ for variances, and $W \times H \times K$ for weights). W and H are image width and height, respectively, and K is the number of Gaussians.

III. Experiments

A. Experimental Setting

Performance evaluation and comparison were conducted using 12 video sequences. Table I explains the databases in detail and Fig. 3 shows example images of them. DB9 and DB10 were acquired at the same location, but under different illumination conditions. They were taken while the illumination was flickering and swinging, respectively. DB11 and DB12 were noise-contaminated versions of DB3 and DB4, respectively (additive Gaussian noise and peak signal-to-noise ratio = 30 dB). For the experiment, three kinds of images are generated from the original color images in the databases. First, grayscale images are generated from the original color images. Second, Bayer-pattern images are generated from the original color images. Last, RGB color images are generated from the Bayer-pattern images via the bilinear demosaicing mentioned in Section II-A. The method in [7] was re-implemented by authors, and the number of Gaussians (K), the

TABLE III
AVERAGE PROCESSING TIME (S)

Proposed Method	RGB Color	Grayscale	Pseudo-Grayscale
0.50	1.25	0.44	0.44

learning rate (α) and the measure of the minimum portion (T) were set to 4, 0.005, and 0.5, respectively. These parameters were empirically chosen to have small and balanced false negative rate (FNR) and false positive rate (FPR). All experiments were run in MATLAB using a 2.8 GHz Intel Core i7 860 central processing unit.

B. Performance Evaluation

The proposed method was evaluated and compared with the method in [7] using three types of images: RGB color, grayscale, and Bayer-pattern images. When the method in [7] uses Bayer-pattern images, we refer to it as "pseudo-grayscale" since the Bayer-pattern images are used as grayscale images. This is to avoid confusion between the proposed method and the method in [7] using Bayer-pattern images. For performance evaluation and comparison, we used three criteria: FNR, FPR, and processing time. FNR and FPR were calculated in the sense of foreground detection and are shown in Table II, and the processing time is shown in Table III. These two tables show that the proposed method achieves similar or slightly higher accuracy compared to the method in [7] using RGB color images while maintaining similar computational costs as when grayscale images are used.

As shown in Table II, the FNR of the proposed method is less than that of the method in [7] using RGB color images by 2.42% in average. This result shows two things: one is that the drawback of the proposed method induced by the interpolated RGB domain mentioned in Section II-C seldom occurs in a real situation and the other is that the separate variance estimation of RGB components in the proposed method can increase the foreground detection accuracy. FNRs of the method in [7] using grayscale and pseudo-grayscale images are quite similar because these two types of images have only one channel information for each pixel. However, the FNR of the proposed method is noticeably less than those of the method in [7] using grayscale and pseudo-grayscale images by over 12.33% and 12.57%, respectively. FPRs of the proposed method and the method in [7] using RGB color images are higher than those of the method in [7] using grayscale and pseudo-grayscale images. This is because two former methods are likely to classify shadow and reflection pixels as foreground. Fig. 4 shows receiver operating characteristic (ROC) curves of four approaches which were drawn by using 12 image sequences. Fig. 5 shows examples of foreground segmentation results. It can easily be seen that the proposed method produced less holes in the foreground regions compared to the other methods.

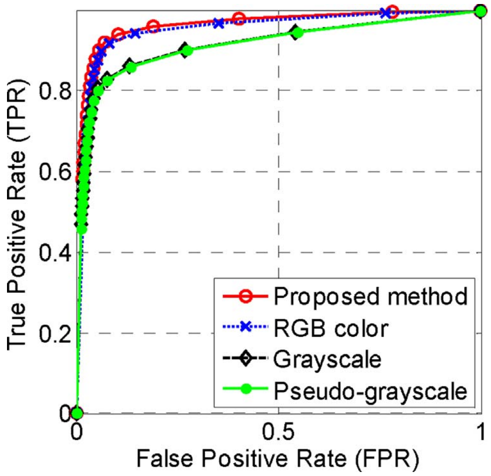


Fig. 4. ROC curves of four approaches.

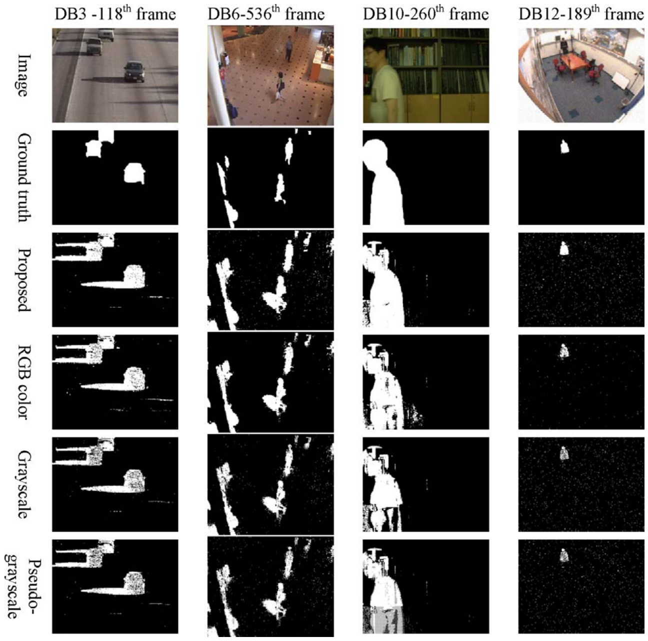


Fig. 5. Examples of foreground segmentation results.

As shown in Table III, the processing time of the proposed method is similar to that of the method in [7] using grayscale and pseudo-grayscale images. There is only a 0.06 s increase in the processing time caused by the bilinear interpolation process. However, the processing time of the proposed method is on average 2.5 times faster than that of the method in [7] using RGB color images.

The overall experimental result shows that the proposed method achieves similar or slightly higher accuracy compared to the method in [7] using RGB color images while maintaining computational resources similar to the case using grayscale images.

IV. CONCLUSION

This letter proposed a background subtraction method for Bayer-pattern image sequences. The proposed method modeled background in a Bayer-pattern domain and classified foreground in an interpolated RGB domain. The experimental results showed that this method is a good solution to obtain high accuracy and low resource requirements simultaneously.

REFERENCES

[1] M. Piccardi, "Background subtraction techniques: A review," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2004, pp. 3099–3104.

[2] R. J. Radke, S. Andra, O. Al-Kofahi, and B. Roysam, "Image change detection algorithms: A systematic survey," *IEEE Trans. Image Process.*, vol. 14, no. 3, pp. 294–307, Mar. 2005.

[3] Y. Benezeth, P. M. Jodoin, B. Emile, H. Laurent, and C. Rosenberger, "Review and evaluation of commonly-implemented background subtraction algorithms," in *Proc. IEEE Int. Conf. Patt. Recog.*, Dec. 2008, pp. 1–4.

[4] B. Rinner and W. Wolf, "An introduction to distributed smart cameras," *Proc. IEEE*, vol. 96, no. 10, pp. 1565–1575, Oct. 2008.

[5] A. N. Belbachir, *Smart Cameras*. New York: Springer, 2009.

[6] M. Heikkila and M. Pietikainen, "A texture-based method for modeling the background and detecting moving objects," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 28, no. 4, pp. 657–662, Apr. 2006.

[7] C. Stauffer and E. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 22, no. 8, pp. 747–757, Aug. 2000.

[8] F. Kristensen, H. Hedberg, H. Jiang, P. Nilsson, and V. Öwall, "An embedded real-time surveillance system: Implementation and evaluation," *J. Signal Process. Syst.*, vol. 52, no. 1, pp. 75–94, 2008.

[9] B. Kisačanin, S. S. Bhattacharyya, and S. Chai, *Embedded Computer Vision*. New York: Springer, 2009.

[10] P. Peurum, S. Venkatesh, and G. West, "A study on smoothing for particle-filtered 3-D human body tracking," *Int. J. Comput. Vision*, vol. 87, nos. 1–2, pp. 53–74, 2010.

[11] K. Huang, S. Wang, T. Tan, and S. J. Maybank, "Human behavior analysis based on a new motion descriptor," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 12, pp. 1830–1840, Dec. 2009.

[12] C. H. Chuang, J. Hsieh, L. Tsai, S. Chen, and K. C. Fan, "Carried object detection using ratio histogram and its application to suspicious event analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 6, pp. 911–916, Jun. 2009.

[13] T. Xia and S. Gong, "Video behavior profiling for anomaly detection," *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 30, no. 5, pp. 893–908, May 2008.

[14] J. Li, F. Li, and M. Zhang, "A real-time detecting and tracking method for moving objects based on color video," in *Proc. 6th Int. Conf. Comput. Graphics, Imag. Visualization*, 2009, pp. 317–322.

[15] D. Alleysson, S. Susstrunk, and J. Herault, "Linear demosaicing inspired by the human visual system," *IEEE Trans. Image Process.*, vol. 14, no. 4, pp. 439–449, Apr. 2005.

[16] B. E. Bayer, "Color imaging array," U.S. Patent 3971065, 1976.

[17] J. E. Adams, Jr., "Interactions between color plane interpolation and other image processing functions in electronic photography," *Proc. SPIE*, vol. 2416, pp. 144–151, Feb. 1995.