Background Subtraction based on Cooccurrence of Image Variations

Makito Seki[†] Toshikazu Wada[††] Hideto Fujiwara[†] Kazuhiko Sumi[†]

[†] Advanced Technology R&D Center
Mitsubishi Electric Corporation
Hyogo, 661-8661, Japan

[††] Faculty of Systems Engineering
Wakayama University
Wakayama, 640-8510, Japan

Abstract

This paper presents a novel background subtraction method for detecting foreground objects in dynamic scenes involving swaying trees and fluttering flags. Most methods proposed so far adjust the permissible range of the background image variations according to the training samples of background images. Thus, the detection sensitivity decreases at those pixels having wide permissible ranges. If we can narrow the ranges by analyzing input images, the detection sensitivity can be improved. For this narrowing, we employ the property that image variations at neighboring image blocks have strong correlation, also known as "cooccurrence". This approach is essentially different from chronological background image updating or morphological postprocessing. Experimental results for real images demonstrate the effectiveness of our method.

1. Introduction

Background subtraction is a convenient and effective method for detecting foreground objects in the stationary background. In real world scenes, however, especially in the outdoor scenes, this restriction, i.e., stationary background, often turns out to be impractical because the background scenes are not stable. For example:

- illumination variations due to sunlight and weather changes, or,
- background object motions, for instance, tree leaves swaying and flags fluttering.

For carrying out desirable foreground detection even in such a dynamic environment, two types of approaches have been proposed:

- use the static background model having a permissible range of the image variations at each pixel or local image block.
- dynamically update the background model.

Most of the former methods set up the permissible range according to the formulization of background variations (e.g., illumination variations) [1, 2], or according to the statistical analysis of training samples of background images [3, 4, 5, 6]. For instance, in Fig. 1, the range of background image variations model is represented as the hatching region in feature space of each pixel or block. When the pixel value or local image pattern is mapped to the outside of this range, it is detected as a foreground object. As a result, the detection sensitivity will be decreased at those pixels having wide ranges [7]. This problem can be improved by increasing the feature space dimension, but it is not realistic, because it requires a large number of background image training samples enough to span a high-dimensional space.

As for the latter approach, several chronological updating methods of the background image have been proposed [5, 8, 9, 10]. In these method, linear prediction, and other heuristic methods are employed, assuming that background image variations are temporally continuous. Some of them employ a special stabilization mechanism, e.g., Wiener filter [9]. Even though, none of these methods is free from erroneous updating. Moreover, these methods can not adapt the quick image variations, e.g., turning on and off the light.

We therefore propose a new method that improves detection sensitivity by dynamically narrowing the ranges of background image variations for every input image. This narrowing does not require the temporal continuity of the background images, but the cooccurrence of image variations at neighboring blocks (see Fig. 2). This approach is essentially different from chronological background image updating or morphological postprocessing based on object shape. Our method has the following advantages: 1) since our method utilize the spatial property of background image variations, it is not affected by the quick image variations, 2) our method can be applied not only to the background object motions, such as swaying tree leaves, but also to illumination variations.

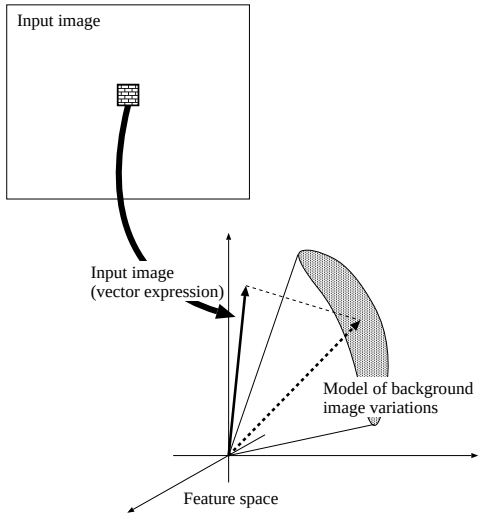


Figure 1. Relation between the model of background image variations and input image.

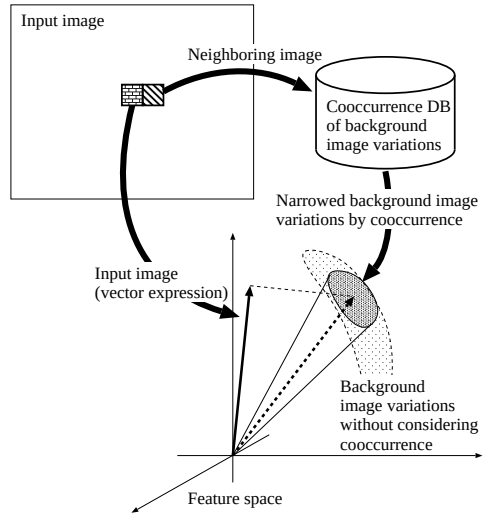


Figure 2. Improvement of detection sensitivity by narrowing range of background image variations.

In the discussions below, we first show that image variations at neighboring image blocks have the cooccurrence. Then, we extend the background subtraction method [6] using eigen spaces for each local image block, and propose a background subtraction method based on the cooccurrence of image variations. Finally, we demonstrate the effectiveness of our method by showing experimental results for real images.

2. Cooccurrence of Image Variations

In this paper, we partition an image into a set of blocks with $N \times N$ pixels and represent each block by an N^2 dimensional vector by scanning the block. Thus, each element of a vector corresponds to the intensity value of the pixel in a block. Hereafter, the image pattern (vector) of time t in a block u is represented as $i_{(u,t)}$.

First of all, we examine our assumption that image variations at neighboring blocks have cooccurrence, using the background image sequence including swaying trees, as shown in Fig. 3. For that purpose, we analyze the principal components of image patterns $i_{(u,t)}$ $(t = 1, ..., \tau)$ for each block, and project them into 2-D eigen space spanned by first and second principal components. As an example, a distribution of the image patterns $i_{(A,t)}, i_{(B,t)}$ on blocks A and B in the Fig. 3 is shown in Fig. 4. In subfigures (a) and (b), point pairs observed within the same time interval are colored with the same color.

As shown in this figure, points having the same colors are almost solidified, and the ordering of colors are the same in (a) and (b), although the right-and-left inversion. It is also confirmed by other experiment, that this property stands for higher dimensional eigen space. Almost the same property can be found not only in blocks A and B but in other neighboring blocks. It is the same as fluttering flags.

In addition, we examined the case where sunlight changes, using a background image sequence for 24 hours $(0.5 frame/min)$. For example, a distribution of the image patterns $i_{(A,t)}, i_{(B,t)}$ on blocks A and B in Fig. 5 is shown in Fig. 6. As shown in this figure related to sunlight changes, the same color points form solidified clusters, and the ordering of colors are the same correctly.

From these examples, we can assume that image variations at neighboring image blocks have strong correlation.

3. Background Subtraction Method

Here, we explain the background subtraction method based on the cooccurrence of image variations mentioned above. This method can be regarded as narrowing the background image variations by estimating the background image pattern in each image block from the neighboring image patterns in the input image.

3.1. Background Image Variations Model

First, our proposed method learns the background image patterns $i_{(u,t)}$ $(u \subset U; t = 1, ..., \tau)$. Here, U is all divided blocks, and τ is learning time. To compress the dimension of image patterns (N^2), image patterns are projected into a low dimensional eigen space spanned by several principal

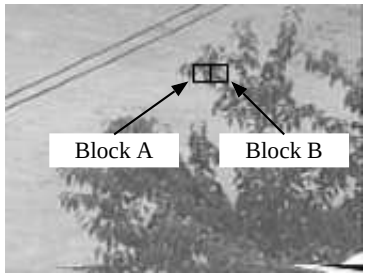
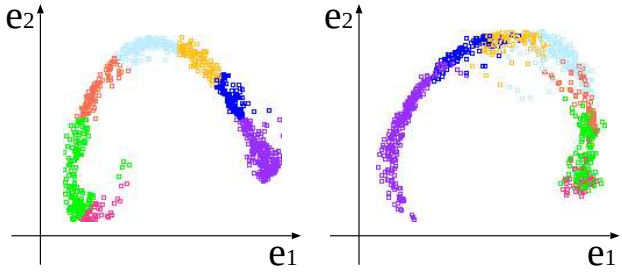


Figure 3. Two blocks examined in tree sway evaluation.



(a) Block A (b) Block B

Figure 4. Distribution of image patterns on blocks A and B in Fig. 3. (In subfigures (a) and (b), point pairs observed within the same time interval are colored with the same color.)

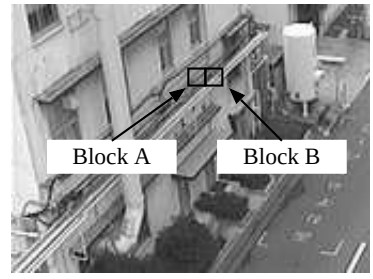
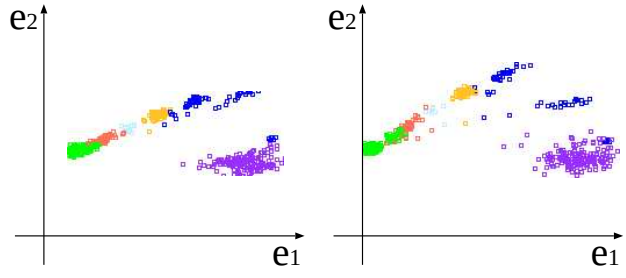


Figure 5. Two blocks examined in sunlight changes evaluation.



(a) Block A (b) Block B

Figure 6. Distribution of image patterns on blocks A and B in Fig. 5. (In subfigures (a) and (b), point pairs observed within the same time interval are colored with the same color.)

components [6]. The cooccurrence of image variations between two neighboring image blocks can be regarded as a nonlinear mapping, which is learned from examples. That is, pairs of image patterns at neighboring image blocks are stored and the mapping for new input can be estimated by interpolating these pairs. The algorithm is summarized as follows.

1. In block u, covariance matrix S_u is computed from the image patterns $i_{(u,t)}$ of the background image sequence and the average pattern $\bar{i}_u$.

$$S_u = \frac{1}{\tau}\sum_{t=1}^{\tau}(i_{(u,t)} - \bar{i}_u)(i_{(u,t)} - \bar{i}_u)^T \qquad (1)$$

2. Covariance matrix S_u is decomposed into its eigen values $\lambda_u(k)$ $(k = 1, ..., N^2)$ and eigen vectors $e_u(k)$.

$$S_u \cdot e_u(k) = \lambda_u(k) \cdot e_u(k) \qquad (2)$$

3. Eigen Space $E_u = [e_u(1), ..., e_u(K)]$ consists of K eigen vectors with a large eigen value. In this case, the dimension K of eigen spaces is common to all blocks.

4. Image patterns $i_{(u,t)}$ of a background image sequence in block u are transformed into points in the eigen space and memorized with observation time.

$$z_{(u,t)} = {E_u}^T \cdot (i_{(u,t)} - \bar{i}_u) \qquad (3)$$

3.2. Estimation of Pattern from Neighboring Block

Since image variations at neighboring blocks have cooccurrence, the image pattern of a certain block can be estimated from the image patterns of neighboring blocks. In this paper, the difference between the estimated pattern and an actual image pattern is used as one of the measures of "background-likelihood". In addition, we don't use simultaneous probability of patterns among neighboring image blocks as a measure of background-likelihood, because it is necessary to learn image patterns in the feature space of a double dimension, and is not realistic, as Chapter 1 described.

The cooccurrence relation of image patterns at neighboring blocks is usually a nonlinear mapping. Thus, we approximate an input pattern by the linear sum of the learning patterns located nearby, and estimate an image pattern

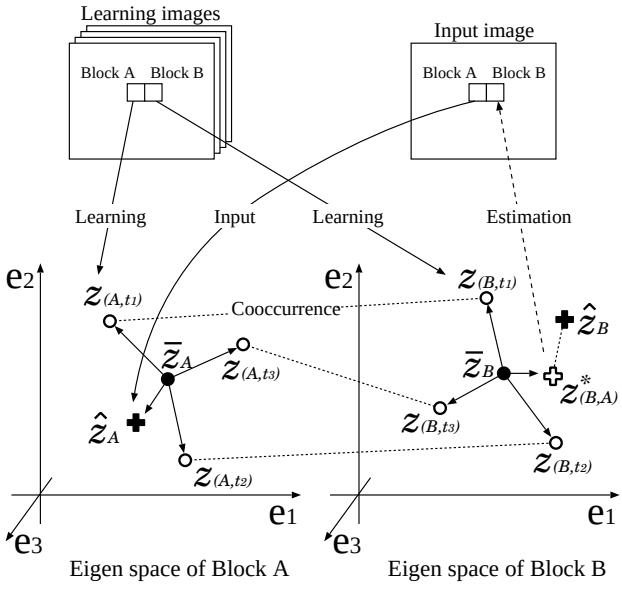


Figure 7. Estimation of a neighboring pattern. (L nearest learning patterns $z_{(A,t_l)}$ for the input pattern $\hat{z}_A$ are searched in the eigen space of block A, L learning patterns $z_{(B,t_l)}$ occurred simultaneously with them are picked up in the eigen space of block B, and the pattern $z^*_{(B,A)}$ is estimated by linear interpolation of the patterns $z_{(B,t_l)}$ in the eigen space of block B.)

at a neighboring block from the coefficient and the cooccurrence relation of learning image patterns known beforehand.

Such a local linear interpolation and local linear transformation from space to space is dealt with by reference [11]. In this research, to recover the 3D shape of an unfolded book surface from a scanner image, the local linear interpolation is performed in an "intensity eigen space". An "intensity eigen space" and a "shape eigen space" are associated by local linear transformation, and a shape is recovered in consideration of the error in interpolation and transformation. Therefore, we estimate the image pattern of a certain block from the image pattern of a neighboring block using the method of reference [11]. The algorithm estimating the image pattern of block B from the image pattern of block A is as follows. (See Fig. 7)

1. New input patterns $\hat{i}_A, \hat{i}_B$ are transformed into the points $\hat{z}_A, \hat{z}_B$ in each eigen space E_u using the average pattern $\bar{i}_u$ of learning patterns.

$$\hat{z}_u = {E_u}^T \cdot (\hat{i}_u - \bar{i}_u), \tag{4}$$

where $u = A, B$.

2. In the eigen space of block A, L learning patterns $z_{(A,t_1)}, z_{(A,t_2)}, ..., z_{(A,t_L)}$ near the $\hat{z}_A$ are picked up, then each pattern is subtracted by the average pattern $\bar{z}_A$, and matrix Z_A is computed. If $z_{(A,t_1)}, z_{(A,t_2)}, ..., z_{(A,t_L)}$ surround $\hat{z}_A$, you may set L to $K+1$, i.e., the eigen space dimension $+1$.

$$Z_A = \left[z_{(A,t_1)} - \bar{z}_A, ..., z_{(A,t_L)} - \bar{z}_A\right] \tag{5}$$

3. The coefficient vector $\Psi = [\alpha_1, \alpha_2, ..., \alpha_L]$ of linear interpolation to $\hat{z}_A$ is computed by the expression below. This expression is derived from consideration of the error of the local linear interpolation in the eigen space and the local linear transformation from eigen space to eigen space (following Step 5) in reference [11].

$$\Psi = (Z_A^T \cdot Z_A + \varsigma_A^2 KI)^{-1} \cdot Z_A^T \cdot (\hat{z}_A - \bar{z}_A), \tag{6}$$

where

$$\varsigma_A^2 = \nu \cdot \frac{tr\left[Z_A^T \cdot Z_A\right]}{KL} \tag{7}$$

when ν denotes a certain small positive number.

4. In the eigen space of block B, L learning patterns $z_{(B,t_1)}, z_{(B,t_2)}, ..., z_{(B,t_L)}$ that occurred simultaneously with $z_{(A,t_1)}, z_{(A,t_2)}, ..., z_{(A,t_L)}$ above are picked up. The average pattern $\bar{z}_B$ is then computed.

5. The pattern $z^*_{(B,A)}$, which should be inputted in block B, is estimated from the above linear interpolation coefficient and the cooccurrence relation of learning image patterns in block A and B.

$$z^*_{(B,A)} = \bar{z}_B + \sum_{j=1}^{L} \alpha_j \cdot (z_{(B,t_j)} - \bar{z}_B) \tag{8}$$

Here, the distance between this estimated pattern $z^*_{(B,A)}$ and the actual image pattern (point projected in the eigen space) $\hat{z}_B$ is used as one of the measures of background-likelihood in block B.

3.3. Calculation of Background-likelihood

Background subtraction is performed by calculating background-likelihood in each block. However, in this paper, we use the following two probability to dynamically narrow the range of background image variations in a focused block. Furthermore, when the product[1] of such probability is low, we judge that the block belongs to the foreground object.

[1] This measure is adopted as the only realization method for narrowing the range of background image variations model appropriately, and is not for calculating the simultaneous occurrence probability of two phenomena.

- Probability P_1 of background-likelihood when judging only based on the input pattern in the focused block.
- Probability P_2 of background-likelihood when judging based on the patterns estimated from some neighboring blocks.

Now, let C denotes the focused block. The probability P_1 of background-likelihood of an input pattern (strictly, a point projected in eigen space) $\hat{z}_C$ is defined as follows using L learning patterns $z_{(C,t_1)}, z_{(C,t_2)}, ..., z_{(C,t_L)}$ near the input pattern in the eigen space.

$$P_1(\hat{z}_C) = \frac{1}{L}\sum_{j=1}^{L}\frac{1}{\sqrt{2\pi}\sigma_C}\exp\left\{-\frac{(\hat{z}_C - z_{(C,t_j)})^2}{2\sigma_C^2}\right\}, \quad (9)$$

where σ_C^2 is constant.

Expression (9) assumes an isotropic normal distribution around each learning pattern point.

On the other hand, let $D_j \ (j = 1, ..., 8)$ denote the eight neighboring blocks of the focused block C. The probability P_2 is defined as follows[2].

$$P_2(\hat{z}_C) = \frac{1}{8}\sum_{j=1}^{8} P(\hat{z}_C | z^*_{(C,D_j)}) \quad (10)$$

Here, probability $P(\hat{z}_C | z^*_{(C,D_j)})$ denotes background-likelihood of an input pattern $\hat{z}_C$ to the pattern $z^*_{(C,D_j)}$ estimated from neighboring block D_j, and is defined as follows.

$$P(\hat{z}_C | z^*_{(C,D_j)}) = \frac{1}{\left(\sqrt{2\pi}\right)^K |\Phi|}\cdot \exp\left\{\frac{\left(\hat{z}_C - z^*_{(C,D_j)}\right)^T \Phi^{-1}\left(\hat{z}_C - z^*_{(C,D_j)}\right)}{2}\right\} \quad (11)$$

Expression (11) assumes a K dimensional normal distribution around the estimated pattern $z^*_{(C,D_j)}$. In addition, Φ denotes the covariance matrix of the learning patterns[3] $z_{(C,t'_1)}, z_{(C,t'_2)}, ..., z_{(C,t'_L)}$, which are used to estimate $z^*_{(C,D_j)}$.

[2]The summation in P_2 is the heuristics introduced for stability. Since the background-likelihood probabilities calculated from neighboring blocks independently may have wide variations, we got the average of them for the stabilization.

[3]Generally, the patterns $z_{(C,t'_1)}, z_{(C,t'_2)}, ..., z_{(C,t'_L)}$ are not necessarily the same as $z_{(C,t_1)}, z_{(C,t_2)}, ..., z_{(C,t_L)}$ which are used by expression (9), because the former is chosen as the cooccurrence patterns from the neighboring block D_j to estimate $z^*_{(C,D_j)}$, and the latter is chosen as the patterns located near the $\hat{z}_C$. That is, selection criteria differ.

Figure 8. Example of input image.

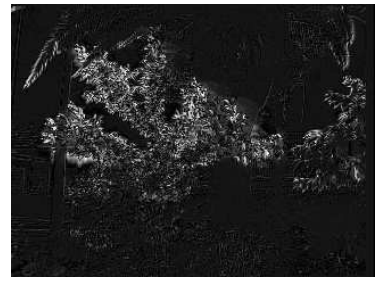


Figure 9. Sway of trees (intensity enhanced).

4. Experiments

To confirm the effectiveness of the above proposed method, we conducted experiments using various image sequences including Fig. 8. The detection target is a person[4], the image size is 320×240 pixels, and the gray level is quantized into 8 bit. Each block size is 8×8 pixels, where every two neighboring blocks overlap each other by four pixels horizontally or vertically[5]. The background image variations are learned from hundreds image in which a person does not exist. Figure 9 shows the difference between one particular image and the averaged image in the image sequence shown in Fig. 8. As shown in this figure, the image sequence contains tree swaying, some sunlight change, and some camera shaking.

Below, we explain the experiment procedure, making the case of the sequence of Fig. 8 into an example.

First, to determine the dimensions of the eigen space, we changed the dimension K gradually and investigated the ratio of effective blocks in which the cumulative contributing ratio to the Kth principal components becomes 90% or more. Figure 10 shows the result, where if the dimension K is set to 15, the ratio of effective blocks will become about 90%. Therefore, the dimension is set as $K = 15$ in the

[4]In some sequences, we composed the person in the background images.

[5]The overlapping of blocks is not necessary for the detection. It is for improving the spatial resolutions of the detection results.

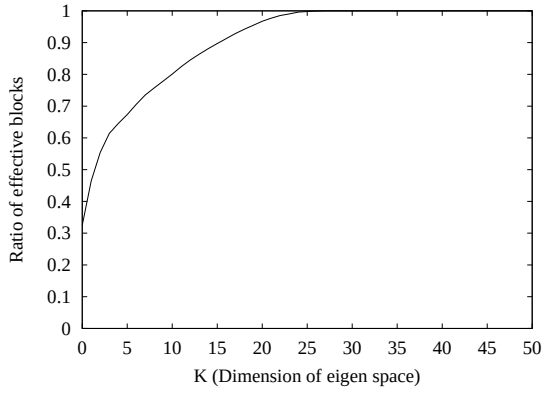


Figure 10. Ratio of effective blocks to the dimension K of eigen space.

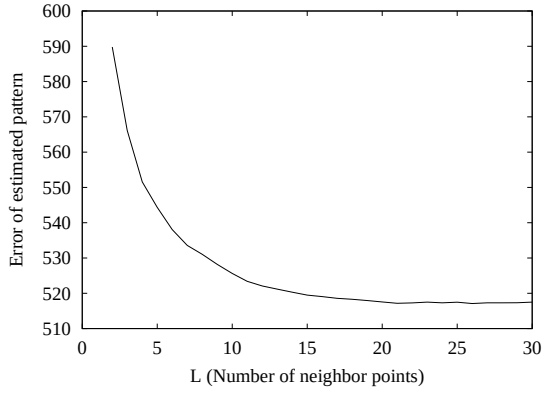


Figure 11. Error of estimated pattern to the number L of neighbor points.

following experiments[6].

Next, to verify whether the number L of neighbor points used for estimation of image pattern is appropriate at $K+1$, we investigated the error (Euclidean distance in the eigen space) of the actual pattern and the pattern estimated from the neighboring block using an image that is not used for background learning. Figure 11 shows the average value of the error in all blocks, when changing L from 2 to 30. As shown in this figure, the error is small around of $L = 16$.

Based on the above, we set $K = 15, L = 16$ and verify the performance of the proposed method, where we use the three methods shown below for comparison.

[Method 1] The method in reference [6]. In this method, the distributions of the image patterns are learned in the eigen spaces constructed for each block, and the background subtraction using the Mahalanobis distances is performed.

[Method 2] The method using the probability P_1 defined in expression (9), i.e., does not use the cooccurrence of background image variations.

[Method 3] The proposed method.

[6]If a scene changes, the number of dimensions decided under the same conditions will change. However, the value is from several dimensions to dozens of dimensions in general.

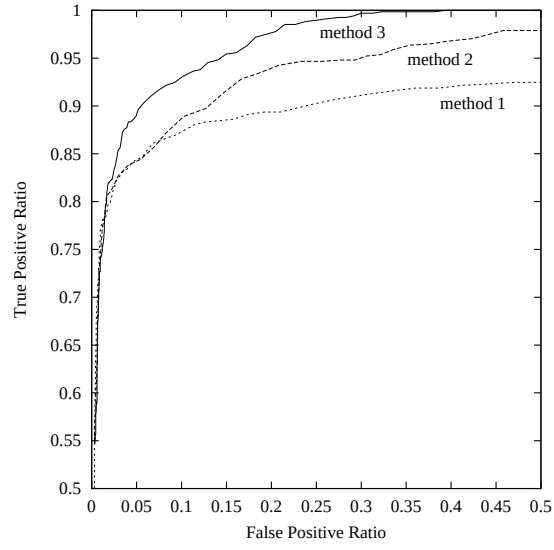
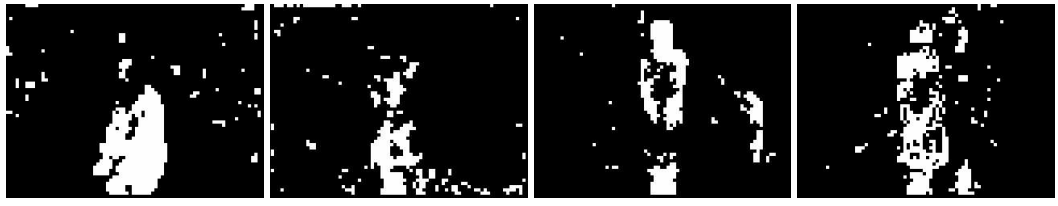


Figure 12. Performance evaluation using ROC curves.

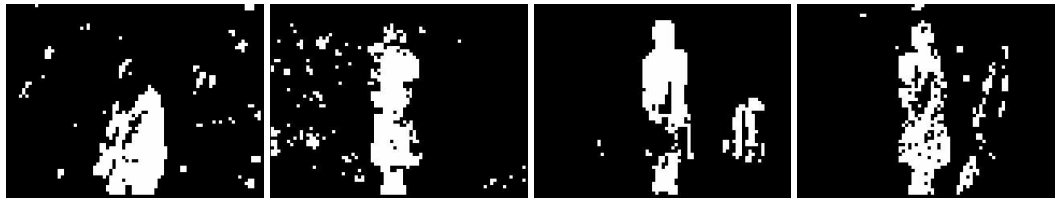
Figure 12 shows ROC curves formed by gradually changing the threshold value used for processing. The horizontal axis shows the ratio of incorrect detections in the background, and the vertical axis shows the ratio of correct detections of a person, where the true data of the person region was created manually. Since the curve of the proposed method is positioned to the upper left of the other curves as shown in this figure, the proposed method is more effective than the others in dynamic scenes involving the background image variations. In addition, the difference among methods 2 and 3 shows the effect by the use of cooccurrence.

Finally, we show examples of the detection results of each method for various image sequences including Fig. 8. Figure 13 shows the results under the False Positive Ratio of 3%. Comparing with methods 1 and 2, the proposed method 3 missed far fewer detections of the person region. In addition, Fig. 14 shows the results under the True Positive Ratio of 85%. As shown in this figure, the proposed method displayed few incorrect detections in the background.

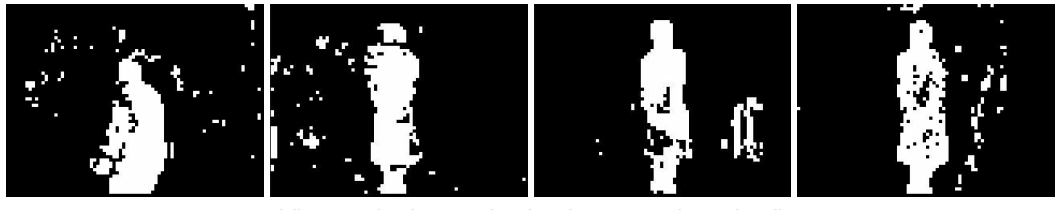
(a) Input images



(b) Results by method 1



(c) Results by method 2



(d) Results by method 3 (proposed method)

Figure 13. Detection results under the False Positive Ratio of 3%. (In proposed method 3, the holes within the person region become small.)

5. Conclusions

In this paper, we propose a novel background subtraction method for detecting objects in dynamic scenes. We first verified that image variations at neighboring blocks have cooccurrence. We then proposed a method that evaluates the background-likelihood based on the cooccurrence for dynamically narrowing the permissible range of background image variations. Finally, we demonstrated the effectiveness of our method by showing experimental results for real images.

Future works include improving the preciseness and computational speed. For example, our method can be extended to more precise method by enlarging the definition of neighboring block.

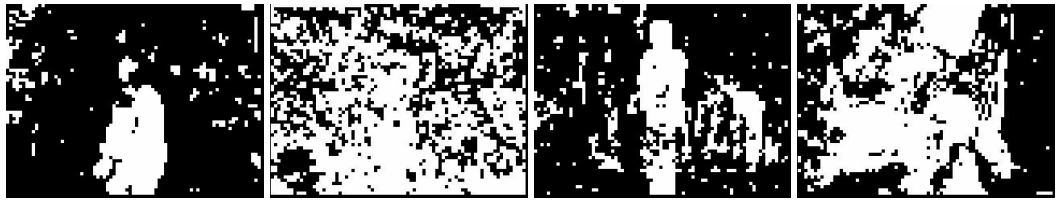
References

[1] N. M. Oliver, B. Rosario, A. P. Pentland, "A Bayesian Computer Vision System for Modeling Human Interactions," *IEEE Trans. on PAMI*, Vol.22, No.8, pp.831-843, 2000.

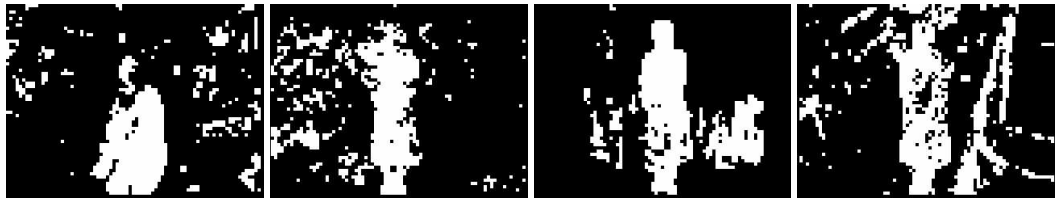
[2] T. Matsuyama, H. Habe, R. Yumiba, K. Tanahashi, "Background Subtraction under Varying Illumination," *Proc. of 3rd Int'l Workshop on Cooperative Distributed Vision*, pp.225-246, 1999.

[3] H. Nakai, "Non-Parameterized Bayes Decision Method for Moving Object Detection," *Proc. of Asian Conf. on Computer Vision*, Vol.3, pp.447-451, 1995.

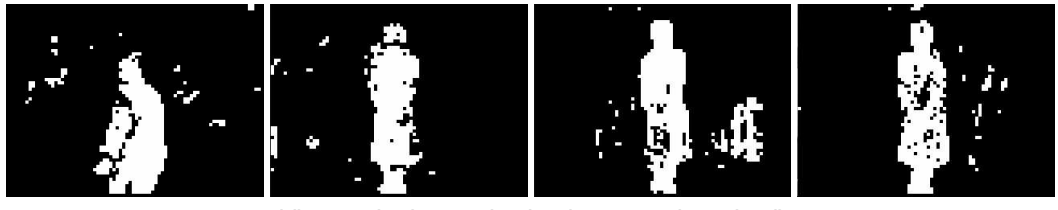
(a) Input images



(b) Results by method 1



(c) Results by method 2



(d) Results by method 3 (proposed method)

Figure 14. Detection results under the True Positive Ratio of 85%. (In proposed method 3, the false pixels in background are smaller than method 1 or 2.)

[4] W.E.L. Grimson, C. Stauffer, R. Romano, and L. Lee, "Using adaptive tracking to classify and monitor activities in a site," *Proc. CVPR*, pp.22-29, 1998.

[5] I. Haritaoglu, D. Harwood, L. S. Davis, "A Fast Background Scene Modeling and Maintenance for Outdoor Surveillance," *Proc. of ICPR*, Vol.4, pp.179-183, 2000.

[6] M. Seki, H. Fujiwara, K. Sumi, "A Robust Background Subtraction Method for Changing Background," *Proc. of IEEE Workshop on Applications of Computer Vision*, pp.207-213, 2000.

[7] A. Iketani, A. Nagai, Y. Kuno, Y. Shirai, "Detecting persons on changing background," *Proc. of ICPR*, pp.74-76, 1998.

[8] K. Karmann, A. Brandt, "Detection and Tracking of Moving Objects by Adaptive Background Extraction," *Proc. of Scandinavian Conference on Image Analysis*, Vol.2, pp.1051-1058, 1989.

[9] K. Toyama, J. Krumm, B. Brumitt, and B. Meyers, "Wallflbwer: Principles and Practice of Background Maintenance," *Proc. of ICCV*, pp.255-261, 1999.

[10] D. Wang, T Feng, H. Y. Shum, S. Ma, "A Novel Probability Model for Background Maintenance and Subtraction," *Int'l Conf. on Vision Interface*, pp.109-117, 2002.

[11] H. Ukida, K. Konishi, T. Wada, T. Matsuyama, "Recovering Shape of Unfolded Book Surface from a Scanner Image using Eigenspace Method," *Proc. of IAPR Workshop on Machine Vision Applications*, pp.463-466, 2000.