

# A Modified Expansion-Contraction Method for Mobile Object Tracking in Video Surveillance: Indoor Environment

Jin-Shig Kang

Department of Tele-Communication Engineering, Jeju National University, Jeju, Korea



## Abstract

Recent years have witnessed a growing interest in the fields of video surveillance and mobile object tracking. This paper proposes a mobile object tracking algorithm. First, several parameters such as object window, object area, and expansion-contraction (E-C) parameter are defined. Then, a modified E-C algorithm for multiple-object tracking is presented. The proposed algorithm tracks moving objects by expansion and contraction of the object window. In addition, it includes methods for updating the background image and avoiding occlusion of the target image. The validity of the proposed algorithm is verified experimentally. For example, the first scenario traces the path of two people walking in opposite directions in a hallway, whereas the second one is conducted to track three people in a group of four walkers.

**Keywords:** Object tracking, Mobile object, Object window, Background image, Modified expansion-contraction algorithm

## 1. Introduction

In recent years, an increasing number of studies have investigated video surveillance and mobile object tracking algorithms. The application areas of object tracking include

- Motion-based recognition of humans,
- Automated surveillance for monitoring a scene to detect suspicious activities or unlikely events, and
- Traffic monitoring for real-time collection of traffic statistics in order to direct traffic flow.

To detect a mobile object, the target object should be separated from the background. This can be done by using the background subtraction method or the frame differencing method for adjacent frames. The method used for object tracking depends on the representation of the target object as a point, silhouette, etc. [1]. Typical object tracking methods are point-based tracking, kernel tracking and silhouette tracking [1, 2]. Recent years have witnessed the growing use of probabilistic approaches, such as the use of a probability distribution to represent the position and color distribution of an object, for object tracking [3].

Several multiple-object tracking algorithms such as Kalman filter [4], particle filter [5-8], and mean shift [9, 10] are also available. Furthermore, a vector Kalman predictor [11] has been proposed for tracking objects. In this paper, separate methods for occlusion and merging are applied to resolve ambiguous situations. Moreover, states of the corresponding moving objects are searched using a spiral searching technique prior to tracking. Recently, Czyzewski

Received: Dec. 13, 2013  
Revised : Dec. 21, 2013  
Accepted: Dec. 23, 2013

Correspondence to: Jin-Shig Kang  
(shigkj@jejunu.ac.kr)  
©The Korean Institute of Intelligent Systems

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

and Dalka [12] used a Kalman filter with an RGB color-based approach to measure the similarity between moving objects. Zhang et al. [13] presented a particle swarm optimization-based approach for multiple-object tracking based on histogram matching. Jiang et al. [14] suggested a linear programming approach, whereas Huang and Essa [15] presented an algorithm for tracking multiple objects through occlusions.

The basic expansion-contraction (E-C) algorithm has been presented in previous papers [16, 17]. The problems discussed in these papers include

- Changes in lighting conditions,
- Failure to track fast-moving objects, and
- Difficulty in separating adjacent objects.

In this paper, a modified E-C algorithm for multiple-object tracking is presented. Modifications are made to the method of expansion and contraction for an object window in order to separate the target object from the surrounding objects and the background. The proposed algorithm includes a method for avoiding occlusion of the target image. Finally, the validity of the proposed algorithm is verified through several experiments.

## 2. Problem Formulation and Definitions

### 2.1 Summary of Some Definitions Proposed by in [18, 19]

Several parameters such as object window, object area, and expansion and contraction parameter defined in are reintroduced in this paper. The binary image is denoted by  $I$ , and  $I_x$  and  $I_y$  are defined as

$$I_x = \text{diag} [I^T I], \quad I_y = \text{diag}[I I^T] \quad (1)$$

where  $I_x$  and  $I_y$  represent the density of non-zero pixels in the  $x$ -direction and  $y$ -direction, respectively. The object window is defined as a minimized image box that includes a target object. The object area can be computed as

$$A = \sum_{k=x_{\min}}^{x_{\max}} (I_x)_k = \text{Trace} [I^T I]$$

or

$$A = \sum_{l=y_{\min}}^{y_{\max}} (I_y)_l = \text{Trace} [I I^T] \quad (2)$$

The center position  $x_p, y_p$  can be calculated as

$$x_p = \frac{[\sum_{l=x_{\min}}^{x_{\max}} l (I_x)_l]}{A}, \quad y_p = \frac{[\sum_{j=y_{\min}}^{y_{\max}} j (I_y)_j]}{A} \quad (3)$$

where  $x_p$  is the center of mass in the  $x$ -direction and  $y_p$  is the center of mass in the  $y$ -direction.

In case of object tracking with a video stream, the size of the target object changes according to its distance from the camera. Thus, the size of the object window must be changed depending on the size of the target object. To carry out this operation, the expansion and contraction parameter is defined as

$$EC_{par} = \frac{\text{Total area of object window}}{\text{Area of object}}, \quad (4)$$

which is the ratio of the object window to the target object. Note that the object window must include the target object, and  $EC_{par}$  must be greater than 1.

### 2.2 Separable, Partially Separable, and Inseparable Objects

It is important to separate the target object from other objects, in order to ensure that the resulting object window contains only the target object. Figure 1 shows a group of people walking together (left), and its corresponding  $I_x$  (top-right) and  $I_y$  (bottom-right). As shown in the left figure, it is difficult to separate the encircled person entirely as a vertical strip or horizontal strip. However, as shown in the top-right figure, the encircled person may be separated as a vertical strip i.e., partially separable on the  $x$ -coordinate. However, a woman indicated by the white arrow cannot be separated on any coordinate because its object area is relatively small and is thus absorbed in a different object's area in the course of the operation of  $I_x$  and  $I_y$ .

Even in this case, the target object lies between 100-220 pixels on the  $y$ -axis and the finally separated object window is shown in Figure 1(b). Further, the corresponding  $I_x$  and  $I_y$  are shown in the center figure and the bottom figure respectively. As shown in the center figure of Figure 1(b), the target object window is separated well and it contains the target object.

Let us now consider another example where the aim is to separate the encircled image as shown in the top-left figure of Figure 2. As shown in the middle and bottom figures, the target object (people) is partially separable on the  $x$  coordinate but is inseparable on the  $y$  coordinate. Thus, from the information obtained the middle figure, i.e., the target object lies in 120~160 pixels across, the image can be separated into the strip image, which contains 120~160 pixels across in the  $x$ -direction and all pixels in the  $y$ -direction. The resulting strip is shown by the strip box in the top-right figure of Figure 3(a). The next step is to recalculate  $I_x$  and  $I_y$  for the strip image obtained previously, which is shown in the top-right and second right

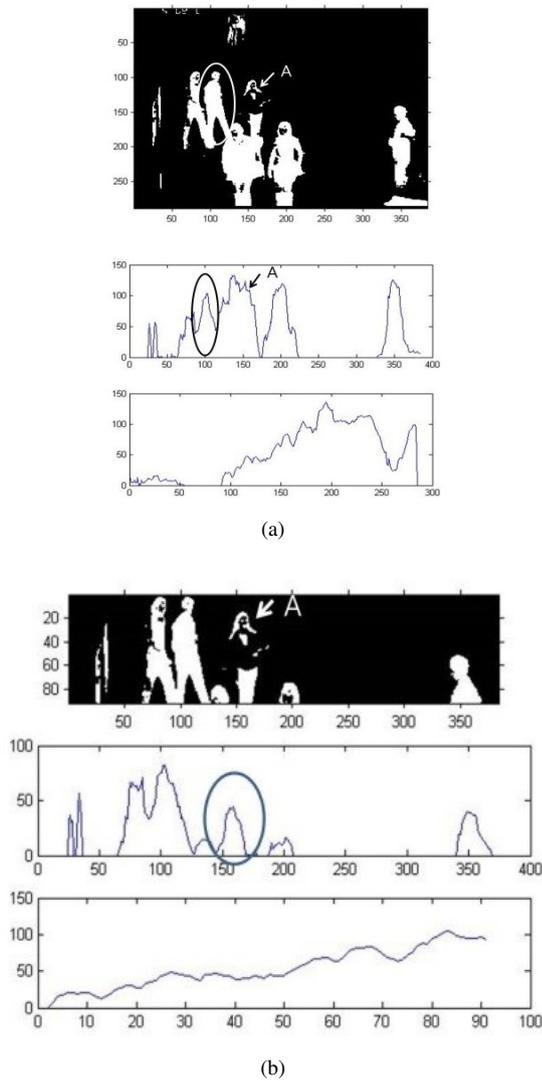


Figure 1. (a). Group of people walking together (left), corresponding  $I_x$  (top-right) and  $I_y$  (bottom-right). (b). The stripped image on  $y$ -coordinate (top), corresponding  $I_x$  (center) and  $I_y$  (bottom).

figures in Figure 3(b). The top-left figure in Figure 3(b) shows the strip image, the top-right figure is  $I_x$ , and the second-right figure is  $I_y$ . The strip image i.e., the top-left figure shows that there is some noise at the top of the strip image, which cannot be separated on the  $x$  coordinate anymore. However, as shown in the strip image or second-right figure, the target object can be separated from the noise on the  $y$  axis.

### 3. Modified E-C Method

The entire process of object tracking is described in this section. This section describes the overall system flow and suggests an algorithm for updating the background image. A method for

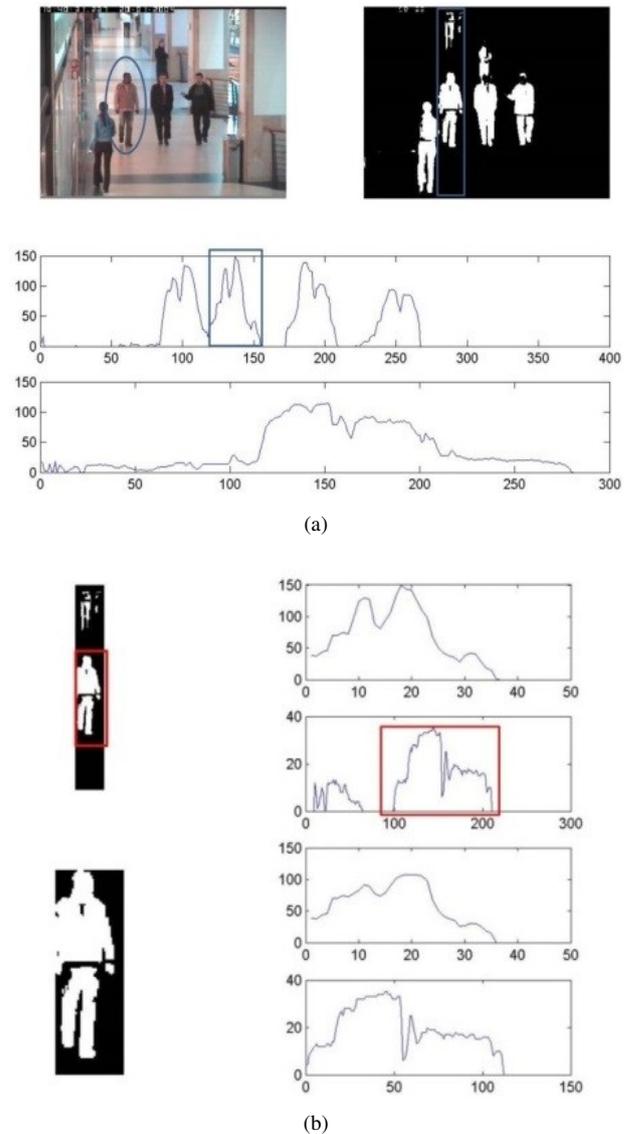


Figure 2. (a) The original image frame (top-left), binary image (top-right),  $I_x$  (middle) and  $I_y$  (bottom). (b) The strip image (top-left), the final object window (bottom-left),  $I_x$  (top-right) and  $I_y$  (second-right) for strip image and  $I_x$  (third-right) and  $I_y$  (bottom-right) for the final object window.

expansion and contraction of the object window and the process of selecting an object by color information are also described in this section.

#### 3.1 Object Tracking Procedure

The overall process of object tracking is shown in Figure 3. The first step in object tracking is the initialization process. This step involves

- Computation of the initial position of the target object,

- Selection of an extended initial object window,
- Selection of  $\Delta p_0 (\Delta x_0, \Delta y_0)$ , which is the initial value of the variation of the center of mass point of the target object, and
- Computation of the predicted center of mass position  $\hat{p}_1 (\hat{x}_1, \hat{y}_1)$  for the next frame.

Go to the first frame. Extracting the sub-image from the background frame and the current frame is the second step in this process. In this step, the predicted center of mass position  $\hat{p}_1 (\hat{x}_1, \hat{y}_1)$  is considered as the center and the size of the window is three or four times greater than that of the object window that was previously selected. In the next step, the absolute difference between the two sub-images obtained earlier is calculated and converted into a binary image using a threshold operation. The fourth step involves calculating  $\text{diag}(II^T)$  and  $\text{diag}(I^T I)$ , contracting the extended object window, and extracting the target object. The area of the target object, the actual center of mass position  $p_1 (x_1, y_1)$ , and the expansion and contraction parameter  $EC_{par}$  are calculated in this step. In the final step, the predicted center of mass position  $\hat{p}_2 (\hat{x}_2, \hat{y}_2)$  is computed. Go to the next frame.

The target tracking process described above can be summarized as three key-stages, prediction - operation - update. In prediction stage, the predicted center of mass position of the target objects are computed by using informations obtained previous frames, and expanded object window, centered at the predicted center of mass and sized three or four times larger than target object, is selected for each target. The primary role of operation stage is extraction of the target objects. This stage includes extraction of sub-image, conversion of sub-image into binary image, calculation of  $I_x$  and  $I_y$ , and contraction of object window. If it is required to separate target object from other objects, then the separation process described in Section 2.2 is performed. In update stage, the actual center of mass position for each target and  $EC_{par}$  are computed.

### 3.2 Expansion and Contraction of Object Window

The center of mass position  $p_k (x_k, y_k)$  for the  $k^{\text{th}}$  frame is described by

$$x_k = x_{k-1} + \Delta x_k \Delta t + \eta_x \tag{5a}$$

$$y_k = y_{k-1} + \Delta y_k \Delta t + \eta_y \tag{5b}$$

where,  $\eta_x, \eta_y$  are noise terms.

For the  $(k + 1)^{\text{th}}$  frame, the predicted center of mass position

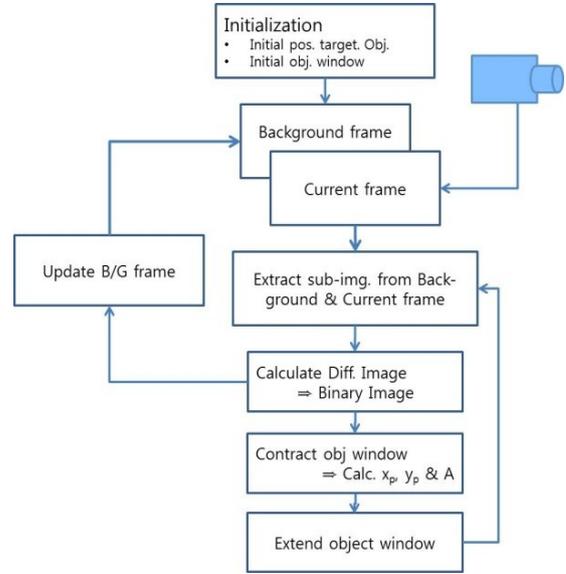


Figure 3. Overview of the system flow.

$\hat{p}_{k+1} (\hat{x}_{k+1}, \hat{y}_{k+1})$  is

$$\hat{x}_{k+1} = x_k + \Delta \hat{x}_{k+1} \Delta t + \eta_x \tag{6a}$$

$$\hat{y}_{k+1} = y_k + \Delta \hat{y}_{k+1} \Delta t + \eta_y. \tag{6b}$$

$\Delta \hat{x}_{k+1}, \Delta \hat{y}_{k+1}$  is then selected as the three-step weighted average, i.e.,

$$\Delta \hat{x}_{k+1} = \frac{1}{(\alpha + \beta)} [\alpha \Delta x_k + \beta \Delta x_{k-1}] + \eta_x \tag{7a}$$

$$\Delta \hat{y}_{k+1} = \frac{1}{(\alpha + \beta)} [\alpha \Delta y_k + \beta \Delta y_{k-1}] + \eta_y \tag{7b}$$

Eqs. (8a) and (8b) are described, in terms of measured values, as follows:

$$\Delta \hat{x}_{k+1} = \frac{1}{(\alpha + \beta)} [\alpha x_k - (\alpha - \beta)x_{k-1} + \beta x_{k-2}] + \eta_x \tag{8a}$$

$$\Delta \hat{y}_{k+1} = \frac{1}{(\alpha + \beta)} [\alpha y_k - (\alpha - \beta)y_{k-1} + \beta y_{k-2}] + \eta_y \tag{8b}$$

For the case of multiple-target tracking, the predicted position of the  $j^{\text{th}}$  object is

$$\hat{p}_{k+1}^j = p_k^j + \Delta \hat{p}_{k+1}^j \tag{9a}$$

$$\Delta \hat{p}_{k+1}^j = \frac{1}{(\alpha + \beta)} [\alpha p_k^j - (\alpha - \beta)p_{k-1}^j + \beta p_{k-2}^j] \tag{9b}$$

The calculations used in this paper to predict the center of mass point of a target object are very simple and adequate for target

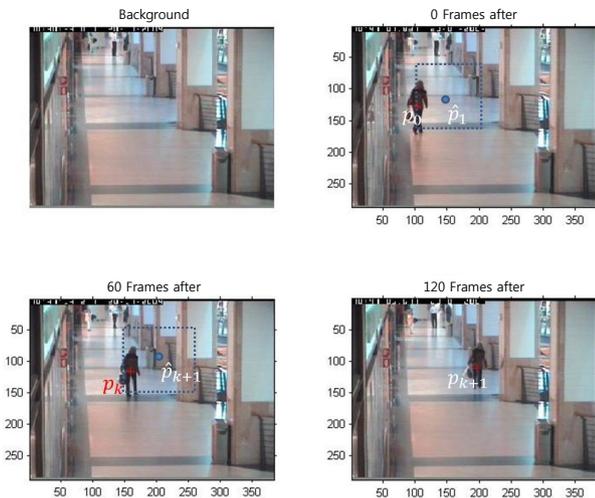


Figure 4. Background image (top-left), base frame (top-right),  $k^{\text{th}}$  frame (bottom-left), and  $(k + 1)^{\text{th}}$  frame (bottom-right) of the expansion and contraction procedure of a person walking at a 60-frames interval are shown above.

tracking in an indoor environment. Of course, the Kalman filtering method or the particle filter algorithm is also available instead of Eq. (5).

The expansion and contraction procedure, a part of the main result of this paper, is shown in Figures 4 and 5. For comparison with other studies, all video materials are borrowed from context aware vision using image-based active recognition (CAVIAR) [17]. Figure 4 shows how to extend and contract the object window. In this figure, the first image is the background image and the remaining three images show a woman walking at 60-frame intervals. The top-right image in this figure shows the expansion and contraction procedure of an object window. The first step is calculating  $p_0(x_0, y_0)$  by reducing the initially selected object window and using Eq. (3) described in the top-left and top-right figures in Figure 5. Then, the predicted center of mass  $\hat{p}_1(\hat{x}_1, \hat{y}_1)$  described in the top-right figure of Figure 4 is computed. In the current ( $k^{\text{th}}$ ) frame, obtain sub-images by extracting the background and the  $k^{\text{th}}$  frame and calculate the binary image shown in the mid-left figure of Figure 5. Then, obtain the object window by contraction (white arrow). Then, compute the predicted center of mass  $\hat{p}_{k+1}(\hat{x}_{k+1}, \hat{y}_{k+1})$ .

The operation of expansion and contraction of the object window is very simple as the actual operation is performed on the  $I_x$  and  $I_y$  axis and not on the image frame expansion. These operations are shown in the middle and bottom figures, respectively. The operation procedure is a two-step process that involves extending and contracting the object window first

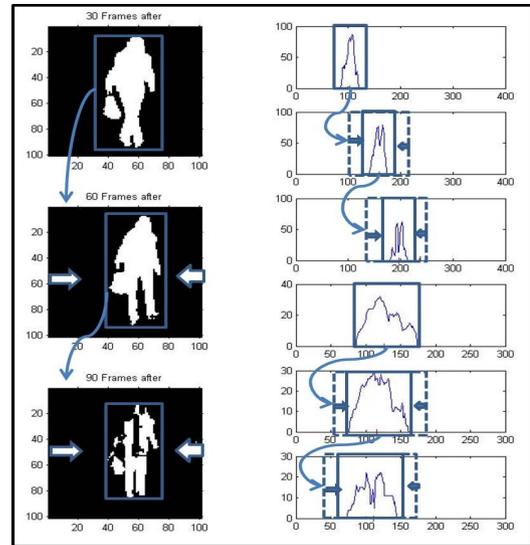


Figure 5. Expansion and contraction of object window (top-right), same operation on  $I_x$  (middle), and same operation on  $I_y$  (bottom).

on the  $I_x$  axis and then on the  $I_y$  axis.

The expansion and contraction parameter, ( $EC_{par}$ ), plays an important role in the contraction operation. Initially, the value of this parameter is greater than 1. It becomes 2 when the ratio of the object area to the total area of the object window is 50%. Further, the value becomes 3 when the ratio becomes 30%. If the expansion and contraction parameter tends to 1, this implies that the object is too large compared to the object window. When the parameter takes a value approximately 3 or 4, it implies that the object is very small compared to the object window. Thus, it is reasonable that the value of the  $EC_{par}$  variable is maintained around 2. When the value of  $EC_{par}$  is close to 1, the object window must be extended, and when it is much greater than 2, the object window must be contracted. In order to maintain the performance of the system, the appropriate  $EC_{par}$  value is around 1.5 to 2.

### 3.3 Selection of Object by Color Information [15]

If an occlusion has occurred, then the color information of the target object just before and after the occlusion is very useful. The tracking can be successfully continued if the two objects are not identical or have similar color. The study on the occlusion can be divided into two kinds. The one is using color and shape information [15] and other is movement information of target object by using particle filter or Kalman filter algorithm [16, 20]. However, when both the objects have identical or similar colors, and are of similar shape, then the tracking may fail. Such a

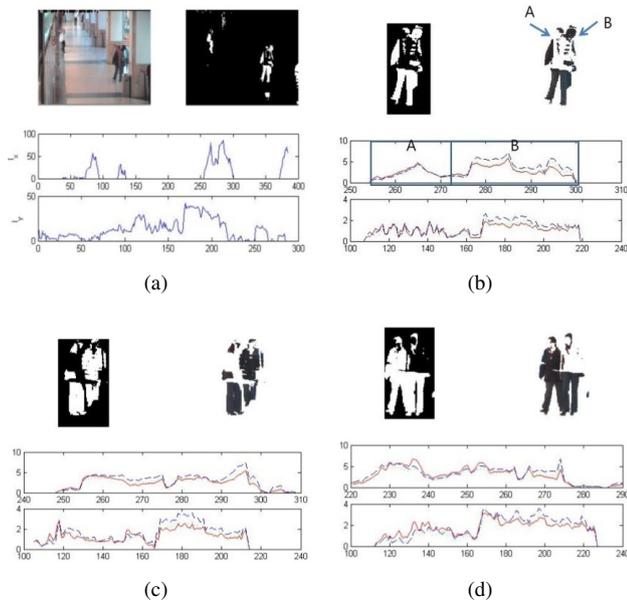


Figure 6. When occlusion has occurred (a, b), just before (c) and after (d).

scenario requires further investigation.

In order to solve this problem, this paper uses both information, i.e., color and shape of the target object and velocity information. Figure 6 shows an occlusion (a and b) occurring just before (c) and after (d). The middle and the bottom figure of (a) is  $I_x$  and  $I_y$  respectively. Middle and bottom figure of (b), (c) and (d) are  $I_x$  and  $I_y$  respectively, but each of which are computed by using color matrix, i.e., RGB matrices. Bottom figure of (b), (c) and (d) shows very similar pattern, even the position of two objects A and B are exchanged. But middle figure of (b), (c) and (d) shows different shape each other. Also two objects can be separate about 275 pixels for (c) and about 265 pixels for (d). Separated objects can be identified by using color distribution or shape.

## 4. Experimental Results and Discussion

To verify the validity of the algorithm presented in this paper, several experiments were performed using mobile images provided by CAVIAR [17]. The first experiment involved tracking one person walking from the bottom-right corner of the lobby towards the top left corner. The second experiment involved tracking two people walking in opposite directions and one person walking in a crowd. The last experimental scenario involves tracking three people walking together and another person walking in the opposite direction.

### 4.1 Scenario 1: Tracking One Person Walking in the Lobby

The first experiment involves tracking one person walking from the bottom right corner of the lobby towards the top left corner. The tracking results of this experiment are shown in Figure 7. Each frame in this figure is selected from the 10-frame steps. The calculated target positions for each frame are marked by “\*”. As shown in Figure 7, the target tracking is performed successfully and the proposed algorithm works well.

### 4.2 Scenario 2: Tracking Two People Walking With Other People

The second scenario consists of tracking two people walking in opposite directions and one person walking in a crowd. In Figure 8, the tracking procedure is shown by 13-frame intervals. In each image, the person walking in the upward direction is marked by a red cross. Further, the person walking with a group of three people from the center in the downward direction is marked by a yellow cross. As shown by the second row and fourth row, accurate tracking is performed even when these two people approach very closely.

### 4.3 Scenario 3: Tracking Three People Walking With Other People

The third experimental scenario is to track three people walking together and another person walking in the opposite direction. This scenario is the same as scenario 2, except that one person is added to the target. It is known by this scenario that the computational complexity increases in comparison with scenario 2. However, it does not significantly affect the run-time. This procedure is shown in Figure 9.

## 5. Conclusion

This paper investigated multi-human tracking in an indoor environment and presented a modified E-C method. The proposed algorithm provides the advantages of the mean-shift algorithm as well as the useful properties of particle swarm optimization and filter-based algorithms for multi-object tracking. Some useful new variables were defined, such as object window, E-C parameter (i.e. the ratio of the object area to the object window area),  $I_x$ , defined as the distribution of non-zero pixels in the horizontal direction (x-direction), and  $I_y$ , defined as the distribution of non-zero pixels in the vertical direction (y-direction). The center of mass for a human object is computed using  $I_x$

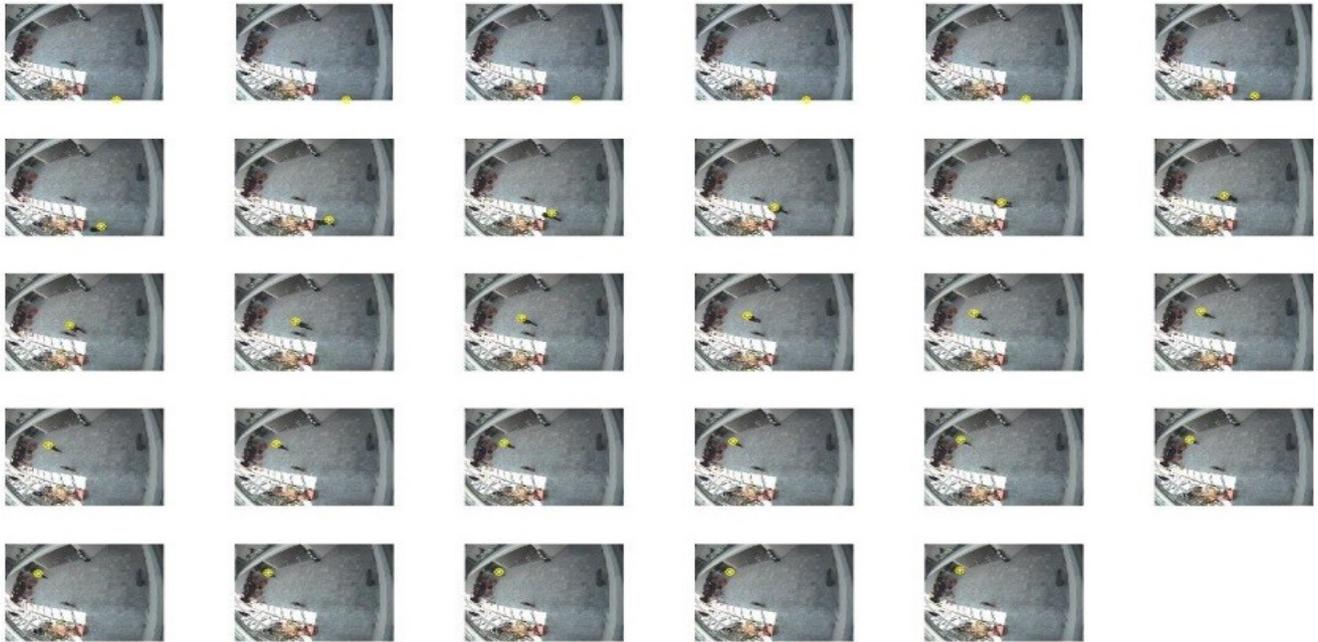


Figure 7. Tracking result for one people. Each frame in this figure is selected from the 10-frame steps.

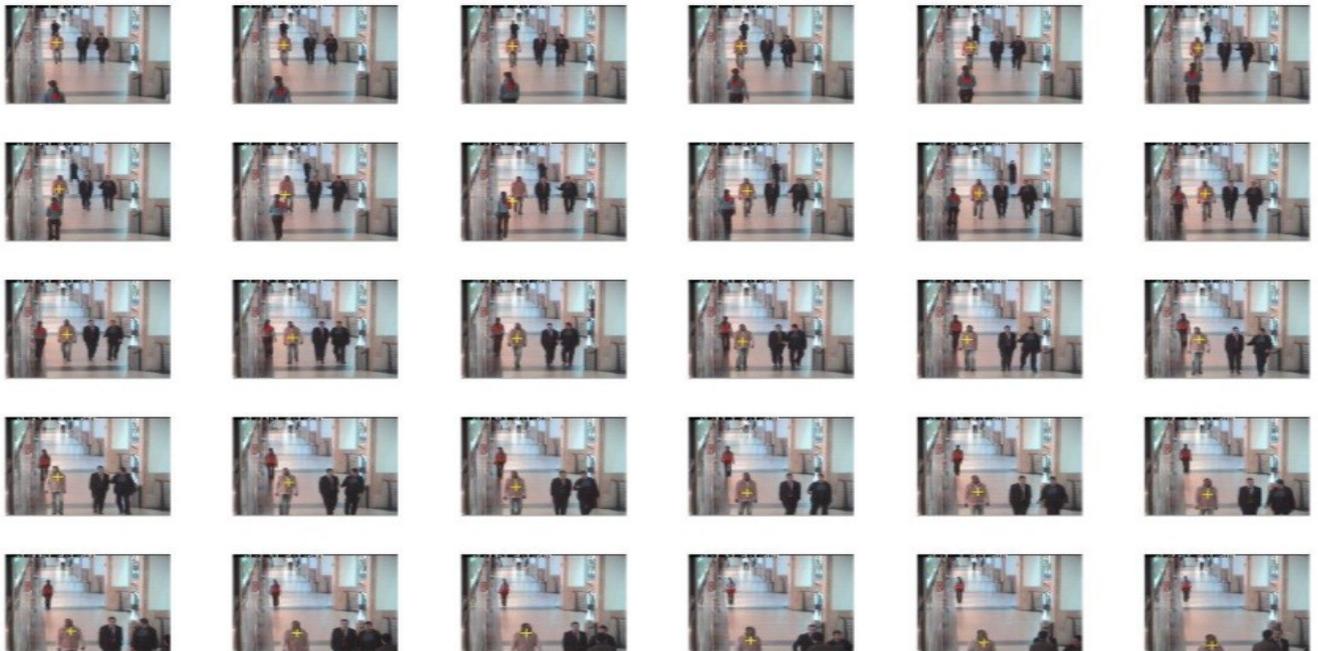


Figure 8. Tracking result for two people walking in opposite directions.

and  $I_y$ . To show that the proposed object tracking method can be efficiently applied to a variety of environment, several experiment were carried out. As stated in the experimental section,

the proposed method works well for every scenario. As the computational load is very low, the proposed method will be useful for more complex environments as well. However, in

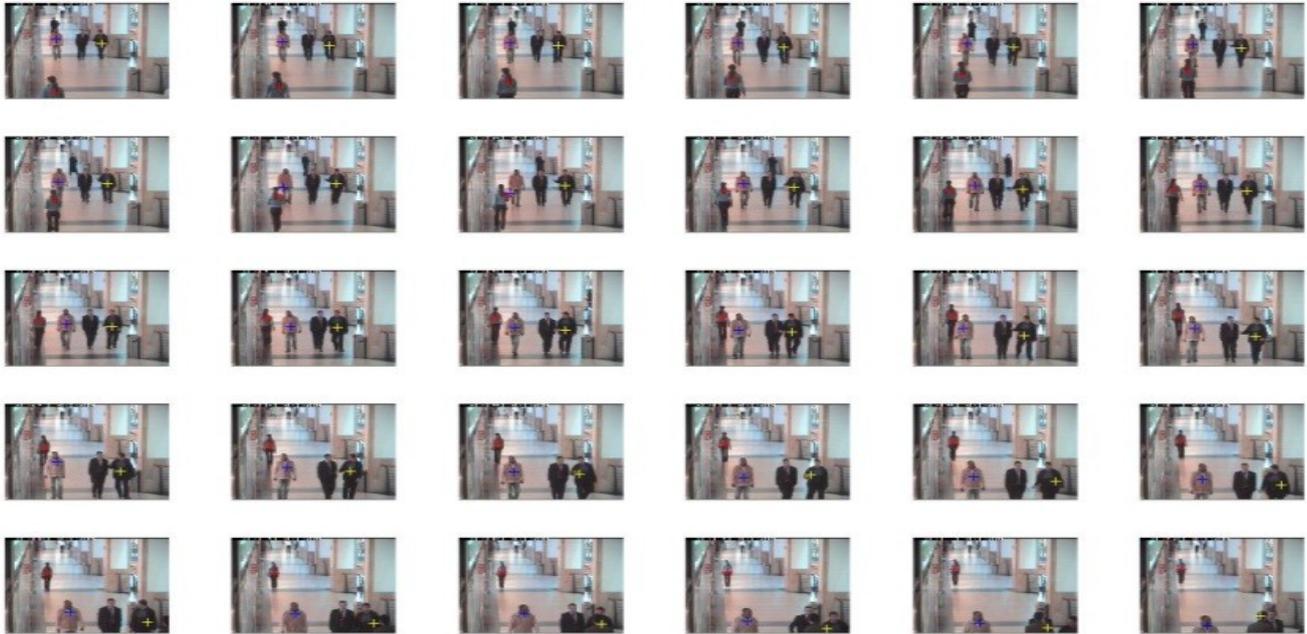


Figure 9. Tracking results for three people when four people are walking. Three people are walking together, but one is walking in the opposite direction.

case of two objects having identical or similar colors, and similar shape, the tracking may fail, and such a scenario requires further research.

### Conflict of Interest

No potential conflict of interest relevant to this article was reported.

### Acknowledgments

This research was supported by the 2013 Scientific Promotion Program funded by Jeju National University.

### References

- [1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking," *ACM Computing Surveys*, vol. 38, no. 4, pp. article number 13, Dec. 2006. <http://dx.doi.org/10.1145/1177352.1177355>
- [2] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564-577, May 2003. <http://dx.doi.org/10.1109/TPAMI.2003.1195991>
- [3] V. Takala and M. Pietikäinen, "Multi-object tracking using color, texture and motion," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, June 17-22, 2007, article number 4270504. <http://dx.doi.org/10.1109/CVPR.2007.383506>
- [4] S. M. Khan and M. Shah, "Tracking multiple occluding people by localizing on multiple scene planes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 505-519, Mar. 2009. <http://dx.doi.org/10.1109/TPAMI.2008.102>
- [5] M.S. Arulampalam, S. Maskell, N. Gordon, T. Clapp. "A tutorial on particle filters for online nonlinear non-Gaussian Bayesian tracking", *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174-188, Feb. 2002. <http://dx.doi.org/10.1109/78.978374>
- [6] C. Hue, J. P. Le Cadre, and P. Pérez, "Tracking multiple objects with particle filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, no. 3, pp. 791-812, Jul. 2002. <http://dx.doi.org/10.1109/TAES.2002.1039400>

- [7] S. Maskell, N. Gordon, "A tutorial on particle filters for on-line nonlinear/non-Gaussian Bayesian tracking," *IEEE Target Tracking: Algorithms and Applications (Ref No. 2001/174)*, October 16-17, 2001, pp. 2/1-2/15. <http://dx.doi.org/10.1049/ic:20010246>
- [8] J. Kwon, K.M. Lee, F.C. Park, "Visual tracking via geometric particle filtering on the affine group with optimal importance functions," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, Miami, FL, June 20-25, 2009, pp. 991-998. <http://dx.doi.org/10.1109/CVPRW.2009.5206501>
- [9] D. Comaniciu and P. Meet, "Mean shift analysis and applications," in *Proceedings of the 1999 7th IEEE International Conference on Computer Vision*, Kerkyra, Greece, September 20-27, 1999, pp. 1197-1203. <http://dx.doi.org/10.1109/ICCV.1999.790416>
- [10] D. Comaniciu and V. Ramesh, "Mean shift and optimal prediction for efficient object tracking," in *International Conference on Image Processing*, Vancouver, Canada, September 10-13, 2000, pp. [d]70-73. <http://dx.doi.org/10.1109/ICIP.2000.899297>
- [11] S. A. Vigus, D. R. Bull, and C. N. Canagarajah, "Video object tracking using region split and merge and a Kalman filter tracking algorithm," in *Proceedings of the International Conference on Image Processing*, October 7-10, 2001, pp. 650-653. <http://dx.doi.org/10.1109/ICIP.2001.959129>
- [12] A. Czyzewski and P. Dalka, "Examining Kalman Filters Applied to Tracking Objects in Motion," in *9th International Workshop on Image Analysis for Multimedia Interactive Services*, Klagenfurt, Austria, May 7-9, 2008, pp. 175-178. <http://dx.doi.org/10.1109/WIAMIS.2008.23>
- [13] X. Zhang, W. Hu, S. Maybank, X. Li, and M. Zhu, "Sequential particle swarm optimization for visual tracking," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition*, Anchorage, AK, June 23-28, 2008, article number 4587512. <http://dx.doi.org/10.1109/CVPR.2008.4587512>
- [14] H. Jiang, S. Fels, and J. J. Little, "A linear programming approach for multiple object tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, June 17-22, 2007, article number 4270205. <http://dx.doi.org/10.1109/CVPR.2007.383180>
- [15] Y. Huang and I. Essa, "Tracking multiple objects through occlusions," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, San Diego, CA, June 20-25, 2005, pp. 1051-1058. <http://dx.doi.org/10.1109/CVPR.2005.350>
- [16] K. E. Ko, J. H. Park, S. M. Park, J. Y. Kim, and K. B. Sim, "Occluded object motion estimation system based on particle filter with 3D reconstruction," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 12, no. 1, pp. 60-65, Mar. 2012. <http://dx.doi.org/10.5391/IJFIS.2012.12.1.60>
- [17] "CAVIAR: Context Aware Vision using Image-based Active Recognition," Available <http://homepages.inf.ed.ac.uk/rbf/CAVIAR/>
- [18] J. S. Kang, "A new mobile object tracking approach in video surveillance. Part I: Indoor environment," in *The 14th International Symposium on Advanced Intelligence Systems*, Daejeon, Korea, November 13-16, 2013, pp. 1097-1102.
- [19] S. W. Kim and J. S. Kang, "A new mobile object tracking approach in video surveillance. Part II: Outdoor environment," in *The 14th International Symposium on Advanced Intelligence Systems*, Daejeon, Korea, November 13-16, 2013, pp. 1103-1108.
- [20] S. M. Park, J. H. Park, H. B. Kim, and K. B. Sim, "Specified object tracking problem in an environment of multiple moving objects," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 11, no. 2, pp. 118-123, Jun. 2011. <http://dx.doi.org/10.5391/IJFIS.2011.11.2.118>



**Jin Shig Kang** received his Ms and Ph.D. degrees from the Department of Electro-Communication Engineering, Hanyang University, Seoul, Korea, in 1986 and 1992, respectively. From 1992 to the present, he

is a professor at the Department of Telecommunication Engineering, Jeju National University, Jeju, Korea. His research interests include industrial control application, robotics and machine vision.

Tel: +82-64-754-3637,

Fax: +82-64-755-3610

E-mail: shigkj@jejunu.ac.kr