J Inf Process Syst, Vol.15, No.2, pp.344~358, April 2019
https://doi.org/10.3745/JIPS.02.0109

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Multi-Person Tracking Using SURF and Background Subtraction for Surveillance

Juhee Yu* and Kyoung-Mi Lee*

Abstract

Surveillance cameras have installed in many places because security and safety is becoming important in modern society. Through surveillance cameras installed, we can deal with troubles and prevent accidents. However, watching surveillance videos and judging the accidental situations is very labor-intensive. So now, the need for research to analyze surveillance videos is growing. This study proposes an algorithm to track multiple persons using SURF and background subtraction. While the SURF algorithm, as a person-tracking algorithm, is robust to scaling, rotating and different viewpoints, SURF makes tracking errors with sudden changes in videos. To resolve such tracking errors, we combined SURF with a background subtraction algorithm and showed that the proposed approach increased the tracking accuracy. In addition, the background subtraction algorithm can detect persons in videos, and SURF can initialize tracking targets with these detected persons, and thus the proposed algorithm can automatically detect the enter/exit of persons.

Keywords

Background Subtraction, Feature Detection, SURF, Tracking, Video Surveillance

1. Introduction

There are many studies on systems for automatically monitoring surveillance videos, but analysis of surveillance video is very difficult due to illumination changes in monitoring areas over lengthy recording periods [1,2]. In addition, to store all recorded videos, most systems store them at low resolution, which may affect the results of video processing.

To address these issues, many studies on robust algorithms for extracting and tracking a person in the changing environments have been actively conducted. The Kalman filtering algorithm, which is robust under changes in time and lighting, was studied for tracking [3,4]. Tian et al. [5] extracted persons using the mixture of Gaussians method, which is robust to lighting changes. In addition, Du et al. [6] and Mu et al. [7] tracked a person using Scale-Invariant Feature Transform (SIFT), which is robust to lighting, time, and environmental changes. Shuo et al. [8] used Speeded Up Robust Features (SURF), which is an interest point descriptor and detector robust to rotation and scaling. However, the SURF tracking method has the disadvantage of tracking errors during abrupt change. To solve this problem and increase tracking accuracy, Zhou and Hu [9] combined SURF with a Meanshift algorithm and Li et al. [10] combined SURF

Manuscript received December 22, 2016; first revision June 5, 2017; accepted September 30, 2017.

Corresponding Author: Kyoung-Mi Lee (kmlee@duksung.ac.kr)
* Dept. of Computer Science, College of Information Media, Duksung Women's University, Seoul, Korea (juneeyoo@naver.com, kmlee@duksung.ac.kr)

with a Camshift algorithm that is affected by color.

In this study, we propose a person-tracking algorithm that combines the SURF tracking algorithm and background subtraction. We use the SURF tracking algorithm that is robust to tracking in surveillance videos with low-resolution and low quality and robust to changes in illumination, scaling and rotation [11]. However, if the shape of a tracked person is changed, SURF fails to track due to partial or no matching. The proposed person-tracking algorithm complements SURF by using an adaptive background-modeling algorithm. Unlike the previous studies on SURF tracking [12,13], the proposed algorithm can automatically initialize tracking targets and thus remove the burden of manual initialization. In particular, even a newly entering person can be automatically initialized as a tracking target.

2. Problems in Multi-Person Tracking

In general, person tracking in videos can be defined as a mapping of persons detected in each frame. Fig. 1 shows the detection of persons P_i^{t-1} and P_j^{t-1} ($i \neq j$) in the $(t-1)^{th}$ frame, and the mapping of the persons P_i^t and P_j^t detected in the t^{th} frame. Such mapping allows the tracking of multiple persons in a video.

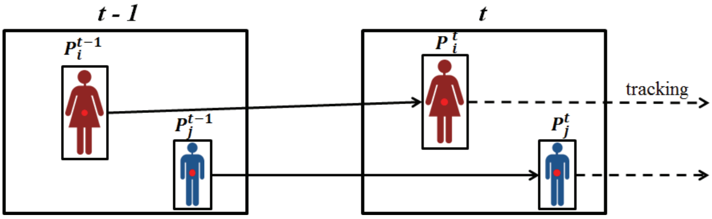


Fig. 1. Mapping between consecutive two frames.

However, from a mapping standpoint, tracking can fail due to various mapping errors such as no mapping, partial mapping or incorrect mapping. Fig. 2 shows a mapping error where a detected person P_i^{t-1} in the $(t-1)^{th}$ frame cannot be mapped to the same person P_i^t in the t^{th} frame. Such mapping error results in the cessation of tracking in subsequent frames.

Fig. 3 shows a mapping error where a detected person P_i^{t-1} in the $(t-1)^{th}$ frame is partially mapped to the same person P_i^t in the t^{th} frame. Such mapping error can cause no mapping or incorrect mapping in subsequent frames.

Fig. 4 shows a mapping error where a detected person P_i^{t-1} in the $(t-1)^{th}$ frame is incorrectly mapped to a different person P_j^t ($i \neq j$) in the t^{th} frame. Such mapping error results in tracking of the different target person.

Generally, many persons enter or exit the fields of view in surveillance videos. Fig. 5 shows a situation where a new person enters into the t^{th} frame but is not tracked in the $(t-1)^{th}$ frame. Such a situation creates a mapping error for tracking the new person in subsequent frames.

On the contrary, Fig. 6 shows a situation where a detected person P_i^{t-1} in the $(t-1)^{th}$ frame does not exits the t^{th} frame. While Fig. 6 shows no mapping because no person is detected in the t^{th} frame, Fig. 2 shows no mapping even though there is a detected person in the t^{th} frame.

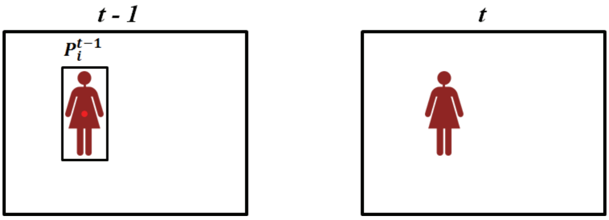


Fig. 2. Mapping error 1: no mapping.

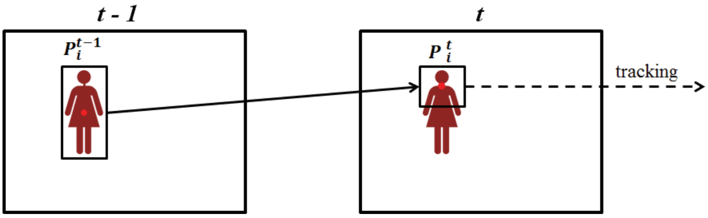


Fig. 3. Mapping error 2: partial mapping.

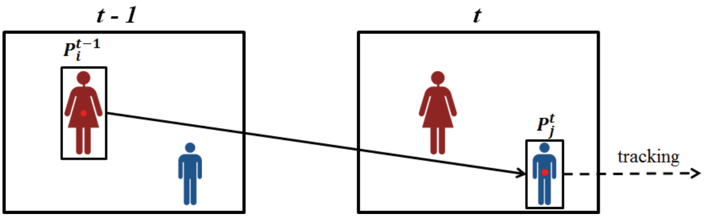


Fig. 4. Mapping error 3: incorrect mapping.

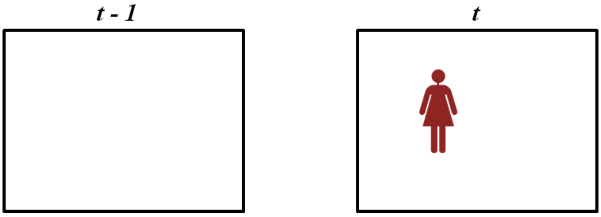


Fig. 5. Mapping error 4: entering.

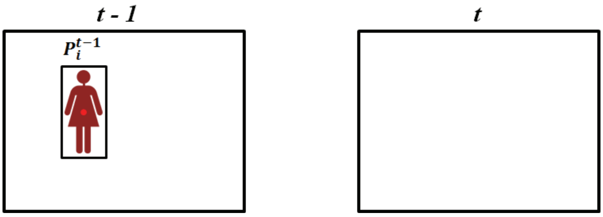


Fig. 6. Mapping error 5: exiting.

3. Proposed Multi-Person Tracking Algorithm

In order to solve the tracking problem as mapping errors in Section 2, the proposed multi-person-tracking system combines the SURF tracking algorithm and background subtraction. Fig. 7 shows a framework of the proposed system, where t is the number of incoming frames (time), and i is the number of tracked persons. At the beginning of the video ($t = 0$), the system initializes tracking using background subtraction and person detection in order to extract the tracked persons as tracking targets. Then, at $t \geq 1$, the system matches the persons tracked in $(t - 1)^{th}$ frame to the persons tracked in the t^{th} frame using SURF tracking. To reduce tracking errors, the proposed system determine a type of tracking errors and correct them using background subtraction and person detection. Finally, the matching results become the targets for the next frame. This process is repeatedly conducted in each frame, as shown in Fig. 7.

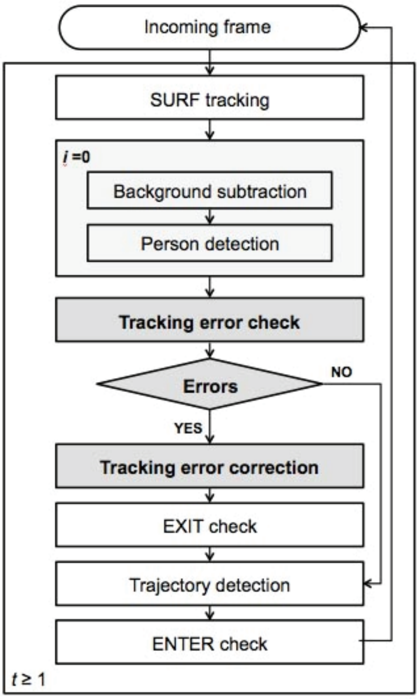


Fig. 7. Framework of the proposed tracking system.

3.1 SURF Tracking

The SURF algorithm is a scale- and rotation-invariant feature extraction algorithm and is therefore suitable for image matching. In addition, SURF can find partially matched images regardless of results of preprocessing. The SURF feature points are computed as follows [9]:

1) To accelerate the computing speed of all rectangles, we compute an integral image by summing the intensity values of each point $X = (x, y)$ in the input image F.

$$F_{\Sigma}(X) = \sum_{a=0}^{a \leq x} \sum_{b=0}^{b \leq y} I(a,b) \tag{1}$$

2) To detect candidate feature points, we compute a Hessian matrix for $X = (x, y)$ in the input image F at scale σ,

$$H(X,\sigma)=\begin{bmatrix} L_{xx}(X,\sigma) & L_{xy}(X,\sigma) \\ L_{xy}(X,\sigma) & L_{yy}(X,\sigma) \end{bmatrix} \tag{2}$$

where $L_{xx}(X,\sigma)$, $L_{xy}(X,\sigma)$, and $L_{yy}(X,\sigma)$ are convolutions of the Gaussian second order derivatives at the point X.

3) Using the Gaussian second order derivatives, we denote the approximation filters as D_{xx}, D_{xy}, and D_{yy} and compute the Hessian determinant

$$\det\left(H_{appx}\right)=D_{xx}D_{yy}-\left(0.9D_{xy}\right)^2 \tag{3}$$

Eq. (3) is computed for various filter sizes and scales.

4) After computing the Hessian determinant with various filter sizes as feature points, we detect interest points k that are local maxima of the region and scale.

5) To determine the directions of feature points, we compute the Haar transform within a circle of radius $6\times\sigma$ centered at each feature point and add weights with a Gaussian function within a circle of radius $2\times\sigma$ centered at each feature point. Then, we rotate the circle $\pi/3$ degrees and determine the direction d with a maximum sum of weights as the direction k_d of the feature points.

6) The SURF algorithm can track persons through image matching [8]. In other words, SURF tracking is a process for finding a person P_i^t in the t^{th} frame F^t by matching P_i^{t-1} as follows:

$$SURF(P_i^{t-1},F^t)\rightarrow P_i^t \tag{4}$$

Fig. 8 shows the examples of results of SURF tracking. The small images on the left are persons in the previous frame, and the rectangles in the right images are persons detected by SURF in the current frame.

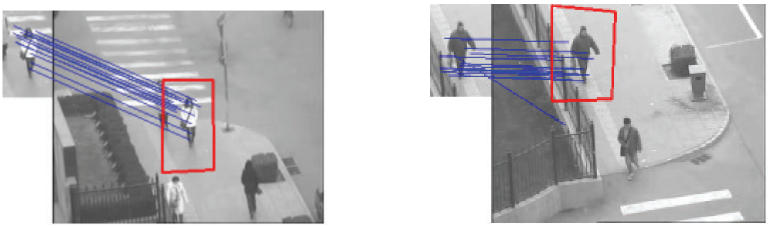

(a) (b)

Fig. 8. Results of SURF matching: (a) Example 1 and (b) Example 2.

3.2 Illumination Adaptive Background Subtraction and Person Detection

To start a tracking algorithm, it needs to initialize a tracking target person. In this study, we use background subtraction to separate the foreground (moving objects) and the background in the frames. However, frames obtained from surveillance cameras may involve various changes in time and illumination, even if the cameras record at the same location and in the same environment. It is difficult to extract the foreground from continuously changing frames. Generally, the foreground can be extracted using the difference between the background frame and the input frame. However, such simple difference images also contain noise due to illumination changes. Thus, the difference image minimizes the time and illumination changes and reduces the changes in the background.

In this study, we use adaptive background modeling to separate a frame *F* to a background image and a foreground image [14]. This adaptive method creates the foreground image using Eq. (5). At time *t*, each frame F^t calculates a Mahalanobis distance δ^t with an average background image μ_B^t and a standard deviation background image σ_B^t. The average image is initialized as the first frame, $\mu_B^0 = F^0$, and the standard deviation image is also initialized at 0, $\sigma_B^0 = 0$.

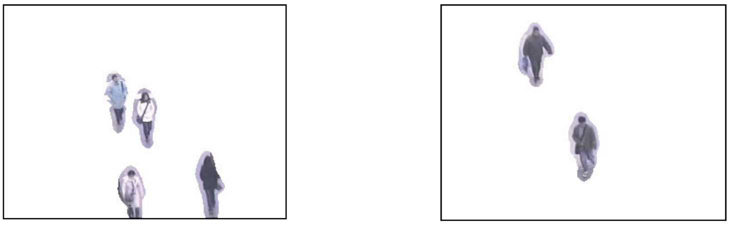
$$\delta^t = \frac{\left|F^t - \mu_B^t\right|}{\sigma_B^t} \tag{5}$$

The background can be considered if the Mahalanobis distance δ^t is small than a thresholding value θ^t. If there is a background, Eq. (6) updates the average and standard deviation background images as follows:

$$\mu_B^t = \alpha^{t-1}\mu_B^{t-1} + (1-\alpha^{t-1})F^t, \qquad \sigma_B^t = \sqrt{\alpha^{t-1}W + (1-\alpha^{t-1})\left(\mu_B^t - F^t\right)^2} \tag{6}$$

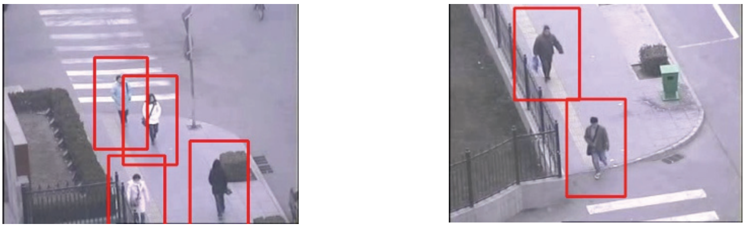
where $W = \left(\sigma_B^{t-1}\right)^2 + \left(\mu_B^t - \mu_B^{t-1}\right)^2$ and $\alpha^{t-1} = \frac{t-1}{t}$. Eq. (6) can make background images adaptive to gradual changes. Whenever a new frame arrives, Eq. (5) calculates the distance with the updated background image by Eq. (6).

Then, we group the foreground frame according to connected foreground pixels and obtained a list of regions. To detect a person, we assume the person stands or walks (runs). Thus, we filter the regions by removing noise-like regions such as small-area regions and non-person-like regions such as regions with horizontally long shapes. Fig. 9(a) and (b) show the result images of adaptive background modeling and noise filtering. Fig. 10(a) and (b) show rectangles of persons as tracking targets.



(a) (b)

Fig. 9. Results of background subtraction: (a) Example 1 and (b) Example 2.



(a) (b)

Fig. 10. Results of person detection: (a) Example 1 and (b) Example 2.

3.3 SURF Tracking Error Detection and Correction

The SURF algorithm continues tracking using image matching between frames. The advantage of using the SURF algorithm for tracking is that the SURF matching is not affected by fine changes between frames. However, if there is a sudden change between frames, SURF results in no match or partial match. Thus, SURF loses a tracking target and unfortunately is no longer able to track.

In this study, to reduce tracking errors, we propose error determination and correction of SURF matching. We consider three types of SURF tracking errors based on mapping errors. The first type of tracking error is no mapping. In other words, P_i^t does not exist as a mapping target of P_i^{t-1}.

$$P_i^t = NULL \tag{7}$$

In Fig. 2, there is no mapping of P_i^{t-1} (box area) with P_i^t, even though P_i^t exists in the t^{th} frame.

The second type of tracking error is partial mapping, in other words, the overlap between the area A_i^t of P_i^t and the area A_i^{t-1} of P_i^{t-1} is larger than θ_A.

$$\left| A_i^t \cap A_i^{t-1} \right| > \theta_A \tag{8}$$

In Fig. 3, P_i^{t-1} (box area) is mapped with a part of P_i^t and the overlapped area of A_i^t and A_i^{t-1} is smaller than θ_A.

The third type of tracking error is incorrect mapping, in other words, the distance between the center C_i^t of P_i^t and the center C_i^{t-1} of P_i^{t-1} is larger than θ_C.

$$\sqrt{\left(C_i^t.x - C_i^{t-1}.x\right)^2 + \left(C_i^t.y - C_i^{t-1}.y\right)^2} > \theta_C \tag{9}$$

In Fig. 4, P_i^{t-1} (box area) is incorrectly mapped with P_j^t. SURF considers P_j^t is a more similar than P_i^t in the t^{th} frame, even though the distance between C_i^{t-1} and C_i^t is greater than θ_C.

To overcome these three types of tracking errors, we correct them by assigning a tractable path. Fig. 11 shows the proposed error determination and correction.

Step 1 (error determination): This step identifies three types of SURF tracking errors using Eqs. (7)–(9).

Step 2 (temporary path assigned): This step assigns a temporary path to the wrong path due to the tracking errors. Since the movement distance between frames is generally short, we can consider the center C_i^{t-1} in the $(t-1)^{th}$ frame as the temporary path of P_i^t in the t^{th} frame.

Step 3 (path correction): C_i^{t-1}, however, is not a correct path in the t^{th} frame. After adaptive background subtraction, we check if the foreground includes C_i^{t-1}. If does, the t^{th} path is corrected to the center of the foreground. If not, we consider the tracked person to exit the view of the camera.

Figs. 12–14 show the proposed method corrects three types of tracking errors. Fig. 12(a) does not map persons located in the frame; however, Fig. 12(b) correctly maps the persons using the correction steps. The partial map in Fig. 13(a) and incorrect map in Fig. 14(a) are corrected to properly map the persons in Fig. 13(b) and Fig. 14(b), respectively.

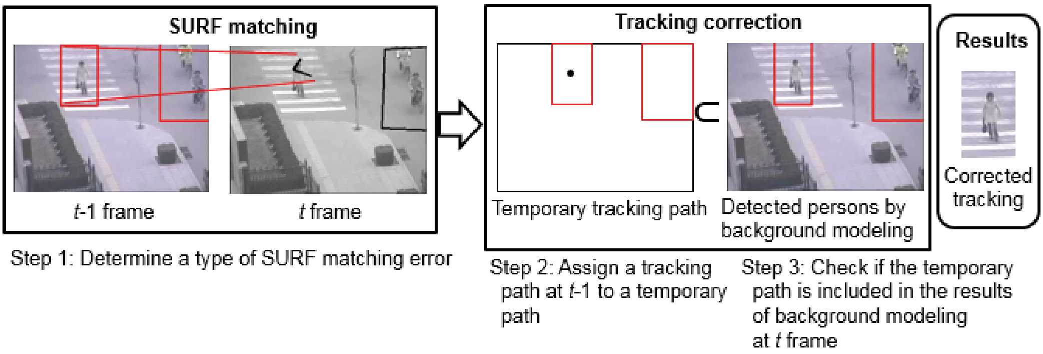
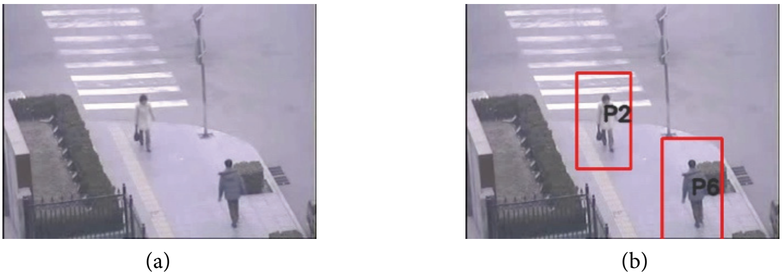
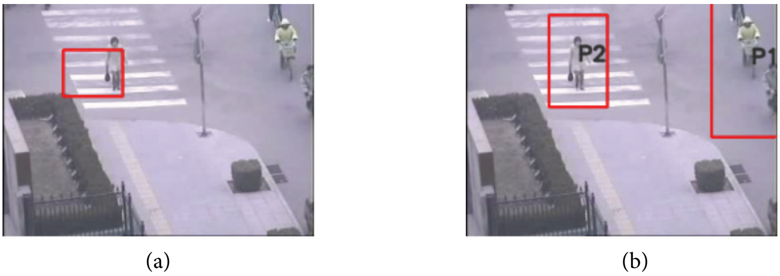

SURF matching
t-1 frame
t frame
Step 1: Determine a type of SURF matching error
Tracking correction
Temporary tracking path
⊂
Detected persons by background modeling
Step 2: Assign a tracking path at t-1 to a temporary path
Step 3: Check if the temporary path is included in the results of background modeling at t frame
Results
Corrected tracking

Fig. 11. Determination and correction of SURF tracking errors.



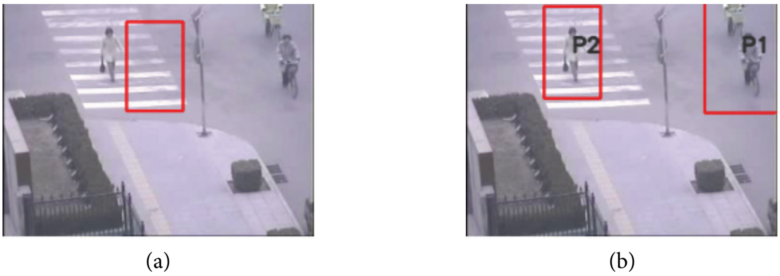
(a) (b)

Fig. 12. Correction of tracking error 1: (a) no-mapping error by SURF tracking and (b) correction by the proposed tracking.



(a) (b)

Fig. 13. Correction of tracking error 2: (a) partial-mapping error by SURF tracking and (b) correction by the proposed tracking.



(a) (b)

Fig. 14. Correction of tracking error 3: (a) incorrect-mapping error by SURF tracking and (b) correction by the proposed tracking.

4. Experimental Results and Analysis

4.1 Experimental Dataset

The proposed tracking system was tested on NLPR_MCT dataset 1 [15]. The dataset includes three synchronous videos recorded from three non-overlapping cameras—two videos were produced by outdoor cameras and one video was by an indoor camera. The cameras recorded in real scenes during daily time, which make the dataset a good representation of normal life. The level of illumination between outdoor and indoor videos varies greatly and the number of persons in the dataset is 235. Also, the problem of occlusion is serious in the indoor video. The dataset contains 15 videos; each is nearly 20 minutes with 20 fps. The video frames are gray-scaled with a spatial resolution of 320×240 pixels.

4.2 Tracking Results

To compare the performance of the proposed person tracking system, we also implemented and tested the conventional SURF tracking [8], and the complementary method with Meanshift [9]. Figs. 15–17 show the cases that an error occurs in the conventional SURF tracking. Figs. 18–20 show the complement results with Meanshift. Figs. 21–23 show the complement results with the proposed method.

Figs. 15, 18, and 21 show the results of applying the three tracking method into the case of no-mapping error. Fig. 15 includes a no-mapping error associated with the conventional SURF tracking. SURF does not track a person at the 6th frame in Fig. 15 and then does no longer track in subsequent frames. Fig. 18 shows Meanshift helps the SURF tracking at the 6th frame and then correctly tracks in the subsequent frames. And Fig. 21 shows the results of the proposed method that can track P2 well in all frames.

Fig. 16, 19, and 22 show the results of applying into the case of partial-mapping error. Fig. 16 includes a partial-mapping error associated with the conventional SURF tracking. SURF partially tracks a person at the 3rd frame and then does no longer track in subsequent frames due to the previous partial-mapping error. Fig. 19 shows Meanshift helps the SURF tracking at the 3rd frame. However, the partial mapping target at the 3rd frame becomes a wrong mapping target from the 8th frame. Fig. 22 shows the proposed method correctly tracks even partial-mapping errors.

Fig. 17, 20, and 23 show the results of applying into the case of incorrect-mapping error. Fig. 17 includes an incorrect-mapping error associated with the conventional SURF tracking. SURF tracks a wrong target at the 5th frame and does no longer track after the 7th frame. Fig. 20 shows Meanshift tracks a wrong target at 5th frame and continues tracking with the wrong target in subsequent frames. However, the proposed method did not show incorrect-mapping errors in Fig. 23.

In addition, SURF requires initialization for tracking. If a person enters during a video, the person cannot be tracked without initialization. However, the proposed system applies automatic initialization with background subtraction.

To calculate a tracking precision, we compute an average error rate E as follows

$$E = \frac{\sum_{t=1}^{f} E^t}{f} \tag{10}$$

$$E^t = \frac{\sum_{i=1}^{n} E_i^t}{n} \tag{11}$$

$$E_i^t = \begin{cases} 0, & \text{Correct tracking} \\ 1. & \text{Others} \end{cases} \tag{12}$$

where f is the number of frames and E^t is the error rate in the t^{th} frame. n is the number of persons in the t^{th} frame, and E_i^t is the number of tracking errors of the i^{th} person.

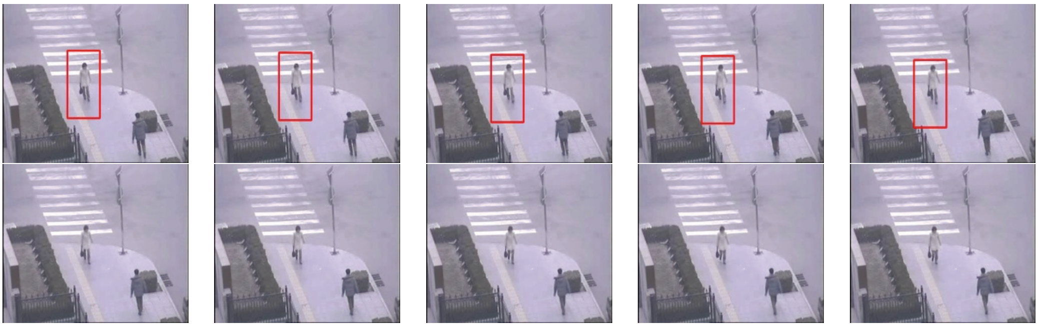


Fig. 15. SURF tracking [8] with a no-mapping error in the top-right to bottom-left direction.

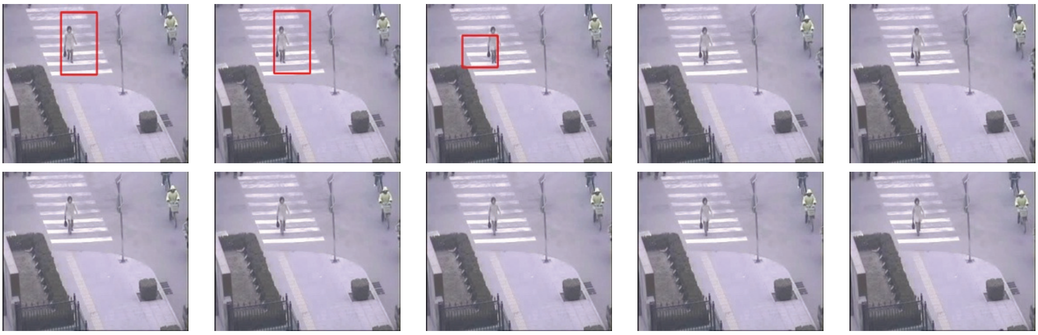


Fig. 16. SURF tracking [8] with a partial-mapping error in the top-right to bottom-left direction.

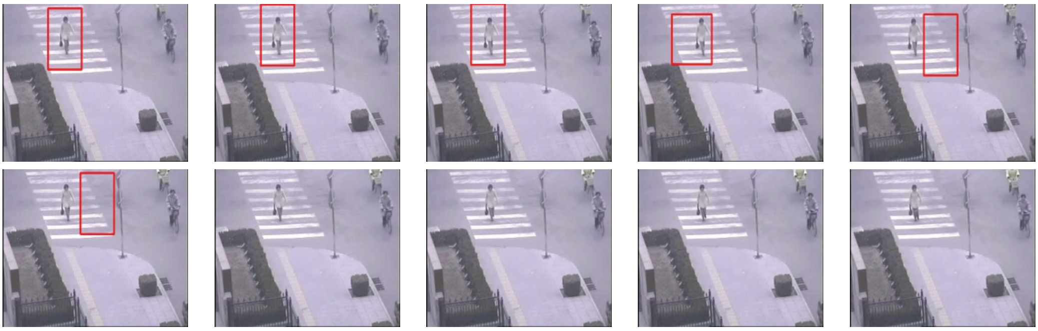


Fig. 17. SURF tracking [8] with an incorrect-mapping error in the top-right to bottom-left direction.

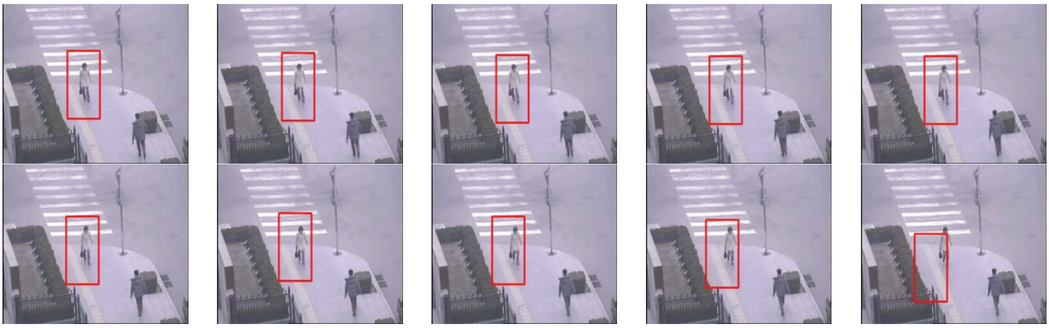


Fig. 18. Meanshift tracking [9] to correct the tracking error in Fig. 15.

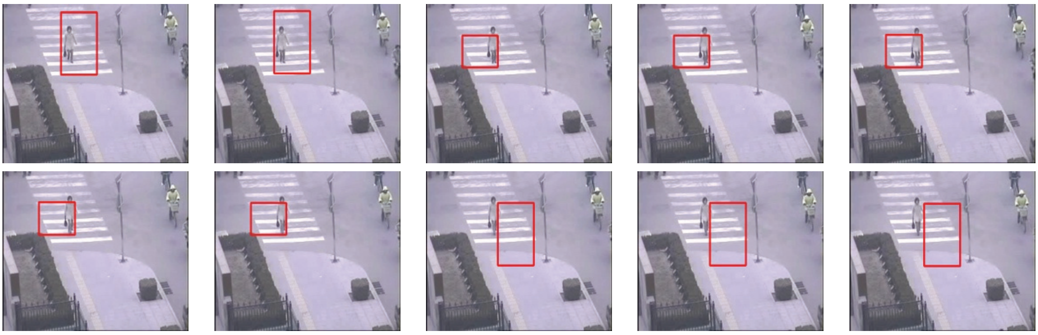


Fig. 19. Meanshift tracking [9] to correct the tracking error in Fig. 16.

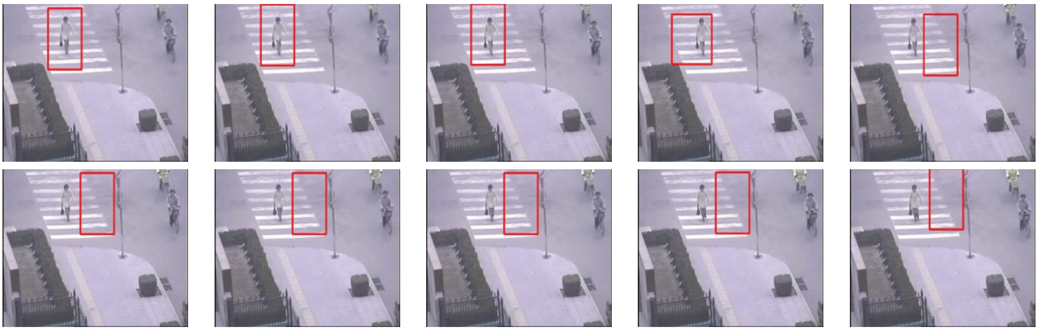


Fig. 20. Meanshift tracking [9] to correct the tracking error in Fig. 17.

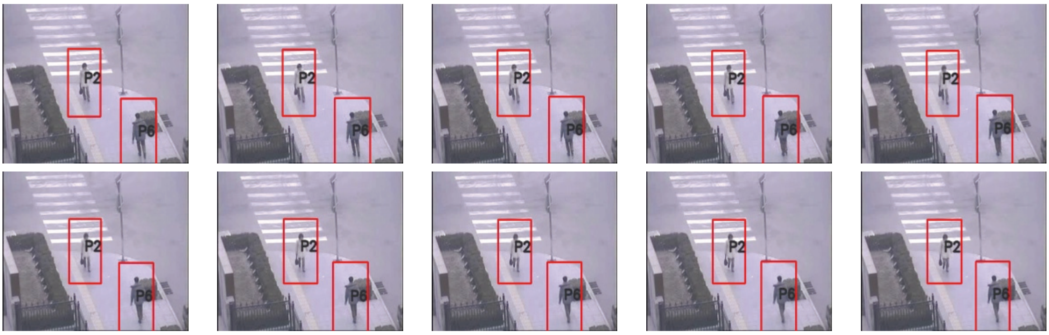


Fig. 21. The proposed tracking to correct the tracking error in Fig. 15.

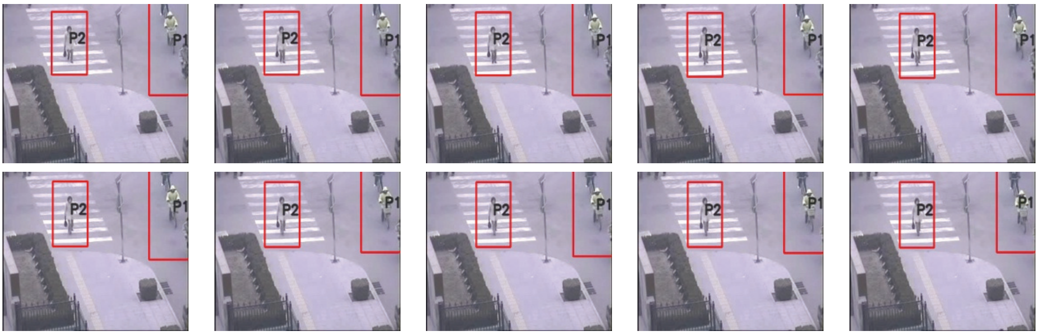


Fig. 22. The proposed tracking to correct the tracking error in Fig. 16.

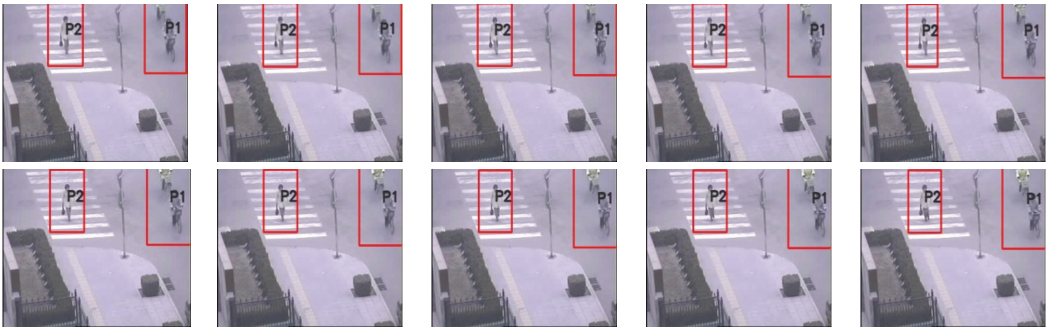


Fig. 23. The proposed tracking to correct the tracking error in Fig. 17.

Table 1 show the precision results using Eq. (10). In Table 1, SURF tracking [8] is 83.6%, and illustrates the three types of errors in addition to other errors. In our experiments, no-mapping error is the most frequent error. Also, the complementary method with Meanshift [9] is 90.4%. Meanshift overcomes the no-mapping error and reduces partial-mapping errors, while the incorrect-mapping errors and others remain. And the proposed tracking method is 96%, which is the highest of all tested methods.

Table 1. Results analysis of three methods

	SURF tracking [8]	SURF tracking + Meanshift [9]	Proposed tracking
Precision (%)	83.6	90.4	96
Error rate (%)			
No mapping	5.1	-	-
Partial mapping	4	3	-
Incorrect mapping	3.2	3.2	-
Others	4	3.4	4

4.3 Result Analysis

In the Subsection 4.1, we show that the proposed tracking method reduces most tracking errors compared with the conventional SURF tracking and the complementary method with Meanshift. We also tested time efficiency and Table 2 shows the averaged processing time per frame, illustrating that the proposed system does not take much more time than the complementary method with Meanshift.

Table 2. Time efficiency

	SURF [8]	SURF+Meanshift [9]	Proposed
CPU time (ms)	0.375	0.398	0.400

While the proposed tracking algorithm increases the tracking precision for most tracking errors, it fails to track some tracking errors, such as overlapping targets, as shown in Fig. 24. Assume that two persons are overlapped in the $(t - 1)^{th}$ frame and thus are recognized as one person, P_i^{t-1}. If two persons are separated in the t^{th} frame and thus are recognized as two persons P_j^t and P_k^t, then P_j^t, and P_k^t are recognized as different persons from P_i^{t-1} and there is no mapping between P_i^{t-1} and P_j^t or P_k^t.

Fig. 25 shows some examples of overlapping targets. Rectangles P2 in Fig. 25(a) and P1 in Fig. 25(b) include two overlapping persons, but they are recognized as one person. Sometimes, two overlapping persons are not even recognized as one person, such as the ellipse in Fig. 25(c).

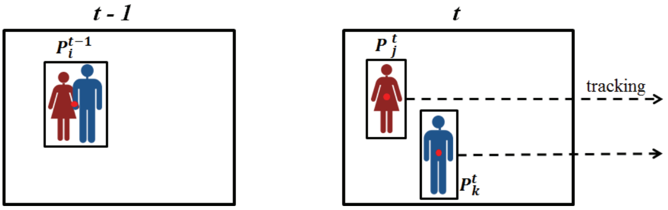
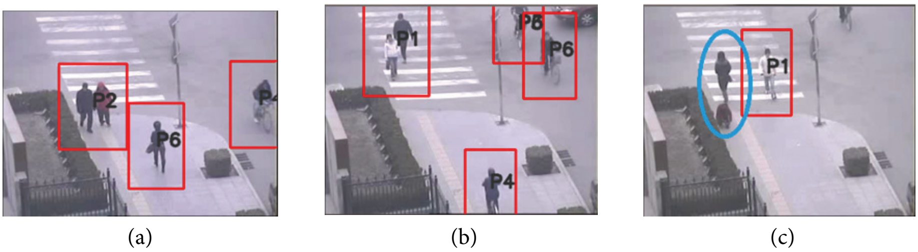


Fig. 24. Example of other mapping error: connected persons.



(a) (b) (c)

Fig. 25. Examples of other errors: (a) P2, two persons are overlapped and are recognized as one person, (b) P1, two persons are overlapped and are recognized as one person, and (c) persons in an oval are unrecognized because of overlapping.

5. Conclusion

SURF is a suitable algorithm for image matching with scale-invariance and rotation-invariance. However, SURF cannot correctly match in the presence of drastic changes between two consecutive frames. To resolve such SURF disadvantages, this study proposes a tracking system with adaptive background subtraction, SURF tracking, and error correction. We also automate the initialization of tracking targets and allow for tracking of multiple persons.

One issue to be addressed further is to overcome overlapping errors in Fig. 25, by considering tracking as one-to-many or many-to-one mapping, instead of one-to-one mapping between two consecutive frames. We are also extending the proposed tracking algorithm for a mobile camera that contains severe image blurs, drastic viewpoint variations, and occlusions.

References

[1] U. Joshi and K. Patel, "Object tracking and classification under illumination variations," *International Journal of Engineering Development and Research*, vol. 4, no. 1, pp. 667-670, 2016.

[2] H. Patel and M. P. Wankhade, "Human tracking in video surveillance," in *Advances in Computing and Information Technology*. Heidelberg: Springer, 2012, pp. 749-756.

[3] C. Li, L. Guo, and Y. Hu, "A new method combining HOG and Kalman filter for video-based human detection and tracking," in *Proceedings of the 3rd International Congress on Image and Signal Processing*, Yantai, China, 2010, pp. 290-293.

[4] C. T. Chu, J. N. Hwang, S. Z. Wang, and Y. Y. Chen, "Human tracking by adaptive Kalman filtering and multiple kernels tracking with projected gradients," in *Proceedings of the 5th ACM/IEEE International Conference on Distributed Smart Cameras*, Ghent, Belgium, 2011, pp. 1-6.

[5] Y. Tian, R. S. Feris, H. Liu, A. Hampapur, and M. T. Sun, "Robust detection of abandoned and removed objects in complex surveillance videos," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 41, no. 5, pp. 565-576, 2011.

[6] K. Du, Y. Ju, Y. Jin, G. Li, Y. Li, and S. Qian, "Object tracking based on improved MeanShift and SIFT," in *Proceedings of the 2nd International Conference on Consumer Electronics, Communications and Networks*, Yichang, China, 2012, pp. 2716-2719.

[7] K. Mu, F. Hui, and X. Zhao, "Multiple vehicle detection and tracking in highway traffic surveillance video based on SIFT feature matching," *Journal of Information Processing Systems*, vol. 12, no. 2, pp. 183-195, 2016.

[8] H. Shuo, W. Na, and S. Huajun, "Object tracking method based on SURF," *AASRI Procedia*, vol. 3, pp. 351-356, 2012.

[9] D. Zhou and D. Hu, "A robust object tracking algorithm based on SURF," in *Proceedings of International Conference on the Wireless Communications & Signal Processing*, Hangzhou, China, 2013, pp. 1-5.

[10] J. Li, J. Zhang, Z. Zhou, W. Guo, B. Wang, and Q. Zhao, "Object tracking using improved Camshift with SURF method," in *Proceedings of International Workshop on Open-Source Software for Scientific Computation*, Beijing, China, 2011, pp. 136-141.

[11] H. Bay, T. Tuytelaars and L. Van Gool, "SURF: speeded up robust features," in *Computer Vision–ECCV 2006*. Heidelberg: Springer, 2006, pp. 404-471.

[12] N. Ren, J. Du, S. Zhu, L. Li, D. Fan, and J. Lee, "Robust visual tracking based on scale invariance and deep learning," *Frontiers of Computer Science*, vol. 11, no. 2, pp. 230-242, 2017.

[13] R. Cao, Q. Li, W. Zhang, Z. Pei, and Y. Liu, "Adaptive block-based target tracking method fusing color histogram and SURF features," in *Proceedings of the 2016 Chinese Intelligence System Conference*. Singapore: Springer, 2016, pp. 193-200.

[14] K. M. Lee, "Adaptive model-based multi-person tracking," in *Computational and Information Science*. Heidelberg: Springer, 2005, pp. 1201-1207.

[15] NLPR_MCT dataset [Online]. Available: http://mct.idealtest.org/Datasets.html.

Juhee Yu https://orcid.org/0000-0002-8093-2339

She received B.S. and M.S. degrees in Department of Computer Science from Duksung Women's University in 2013 and 2015, respectively. She is currently a researcher in the Intelligent Multimedia laboratory, Duksung Women's University, Seoul, Korea.

Kyoung-Mi Lee https://orcid.org/0000-0001-8417-8479

She received B.S. degree in computer science from Duksung Women's University in 1993, M.S. degree in computer science from Yonsei University in 1996, and Ph.D. degree in computer sciences from the University of Iowa, Iowa City, in 2001. She is currently a professor in the Department of Computer Science, Duksung Women's University, Seoul, Korea. Her research interests include multimedia information processing, in particular, image and video processing, multimedia indexing and retrieval, and multimedia mining.