

Disjoint Particle Filter to Track Multiple Objects in Real-time

YoungJoon Chai¹, Hyunki Hong¹, and TaeYong Kim¹

¹GSAIM, Chung-Ang University, Seoul, Republic of Korea
[e-mail: chai1014@yahoo.co.kr, honghk@cau.ac.kr, kimty@cau.ac.kr]

*Corresponding author: TaeYong Kim

*Received February 4, 2014; revised April 3, 2014; revised April 9, 2014; accepted April 14, 2014;
published May 29, 2014*

Abstract

Multi-target tracking is the main purpose of many video surveillance applications. Recently, multi-target tracking based on the particle filter method has achieved robust results by using the data association process. However, this method requires many calculations and it is inadequate for real time applications, because the number of associations exponentially increases with the number of measurements and targets. In this paper, to reduce the computational cost of the data association process, we propose a novel multi-target tracking method that excludes particle samples in the overlapped predictive region between the target to track and marginal targets. Moreover, to resolve the occlusion problem, we define an occlusion mode with the normal dynamic mode. When the targets are occluded, the mode is switched to the occlusion mode and the samples are propagated by Gaussian noise without the sampling process of the particle filter. Experimental results demonstrate the robustness of the proposed multi-target tracking method even in occlusion.

Keywords: Visual tracking, multiple target tracking, particle filter, real-time surveillance

A preliminary version of this paper appeared in IEEE ICCE 2011, Jan. 10-12, LasVegas, USA [14]. This version includes expansion of the key ideas and supporting experimental results for multiple object tracking. This research was supported by Basic Science Research Program of the National Research Foundation of Korea, funded by the Ministry of Education, Science and Technology (2013- 009166).

<http://dx.doi.org/10.3837/tiis.2014.05.012>

1. Introduction

The main goal of visual tracking for video surveillance is to track the position of multiple moving targets. For example, trajectories of multiple targets are useful for analyzing traffic on roads, consumer flow in department stores, sport events, and so on. The Kalman filter [1, 2] or particle filter [3, 4, 5] are widely used for tracking objects. The Kalman filter supports an effective method by estimating a target state according to the time sequence. However, the target state is limited to the linear Gaussian model. The particle filter achieves superior performance in complicated environments. It is possible to apply the particle filter to the non-linear and non-Gaussian models. However, the traditional particle filter is not adequate to independently track multiple targets, since the samples are centralized to the modal with the highest likelihood according to the importance sampling.

Joint Probabilistic Data Association Filter (JPDAF) [6], based on the data association technique, is a useful method for multi-target tracking. This method can independently track multiple targets by evaluating the association probabilities between measurements and targets. The particle filter-based data association methods [7, 8], modified from JPDAF based on the Kalman filter, are robust in tracking multiple targets with nonlinear property. However, these techniques are originally developed for radar data, and they require modification to apply to the visual tracking problem. There are many methods to track objects including the color histogram method [9, 10], which is an extended version of the color histogram method [9], kernel particle filter [11], boosted particle filter [12], and detector confidence particle filter [13], and these methods combine feature detection and tracking for the video sequence.

Despite these efforts, there are some problems in visual tracking based on the data association technique: 1) The tracking results cannot keep the identification of each target after occlusion, because of the lack of clarity of mutual exclusion of each target; and 2) the data association process of JPDAF is too complicated to apply to visual tracking in real-time.

In this paper, we propose the disjoint particle filter (DPF) for multi-target tracking. The proposed method assumes that each measurement cannot be identified by the feature of observed data, but is only disjointed by the spatial information. DPF is faster than JPDAF, which considers every association probability between the targets and measurements, since the proposed method updates weights by only comparing the predictive target probabilities

with those of marginal targets. However, the proposed approach is weak in accuracy under occlusion, because we use the spatial information of samples with propagation to evaluate the predictive target probability. To solve these problems, we use a dynamic mode which can overcome the occlusion problem. When occlusions occur, the mode is changed into the occlusion mode, and the samples are propagated by Gaussian random noise. This method enables the filter to keep track of a target even under temporary occlusion.

This paper is organized as follows. In Section 2, we explain the general particle filter for visual tracking. In Section 3, we present the prediction and the update process of the proposed DPF, and how to change a filter state in dynamic mode. Experimental results are presented in Section 4, which show speedups for real time tracking and robustness in occlusion. The conclusions are presented in Section 5.

2. Visual Tracking using the Particle Filter

The particle filter is also known as the Condensation algorithm [12], and has become famous in the vision-based tracking area. When the state of a tracked object is described by the vector \mathbf{x}^k , while the vector \mathbf{Z}^k denotes all the observations up to time k , to track a target of interest via the particle filter, the posterior probability density function $p(\mathbf{x}^k | \mathbf{Z}^k)$, known as the filtering distribution, should be calculated at each time step k . It can be calculated by two-step recursion in Bayesian sequential estimation;

1) Prediction step:

$$p(\mathbf{x}^k | \mathbf{Z}^{k-1}) = \int p(\mathbf{x}^k | \mathbf{x}^{k-1}) p(\mathbf{x}^{k-1} | \mathbf{Z}^{k-1}) d\mathbf{x}^{k-1}, \quad (1)$$

2) Update step:

$$p(\mathbf{x}^k | \mathbf{Z}^k) \propto p(\mathbf{z}^k | \mathbf{x}^k) p(\mathbf{x}^k | \mathbf{Z}^{k-1}). \quad (2)$$

Because the integrals of the optimal Bayesian solution are intractable, the discrete sums of weighted samples drawn from the posterior distribution are used in the particle filter algorithm. However, it is often difficult to directly sample from the true posterior density; and thus, the particles are sampled from a known proposal distribution.

In order to explain the details of the particle filter, let $\{\mathbf{x}^{i,k}, w^{i,k}\}_{i=1}^N$ denote a weighted set of samples, where $\{\mathbf{x}^{i,k}, i=0, \dots, N\}$ is a set of support points with associated weights

$\{w^{i,k}, i = 0, \dots, N\}$, and $\mathbf{x}^k = \{\mathbf{x}^j, j = 1, \dots, k\}$ is the set of all states up to time step k . The samples are repeatedly generated from the proposal distribution, and the new importance weights are updated by:

$$w^{i,k} \propto w^{i,k-1} \frac{p(\mathbf{z}_k | \mathbf{x}^{i,k}) p(\mathbf{x}^{i,k} | \mathbf{x}^{i,k-1})}{q(\mathbf{x}^k | \mathbf{x}^{i,k-1}, \mathbf{z}^k)}, \quad (3)$$

where, $\sum_{i=1}^N w^{i,k} = 1$, and q is the proposal density function, where new samples are generated from that distribution.

For successful tracking in the particle filter, which particular proposal distribution is used is important. In this paper, we use the state evolution model $p(\mathbf{x}^k | \mathbf{x}^{k-1})$ as a proposal distribution for simple implementation as in [14]. So, the weights of samples are only dependent on the likelihood of measurements.

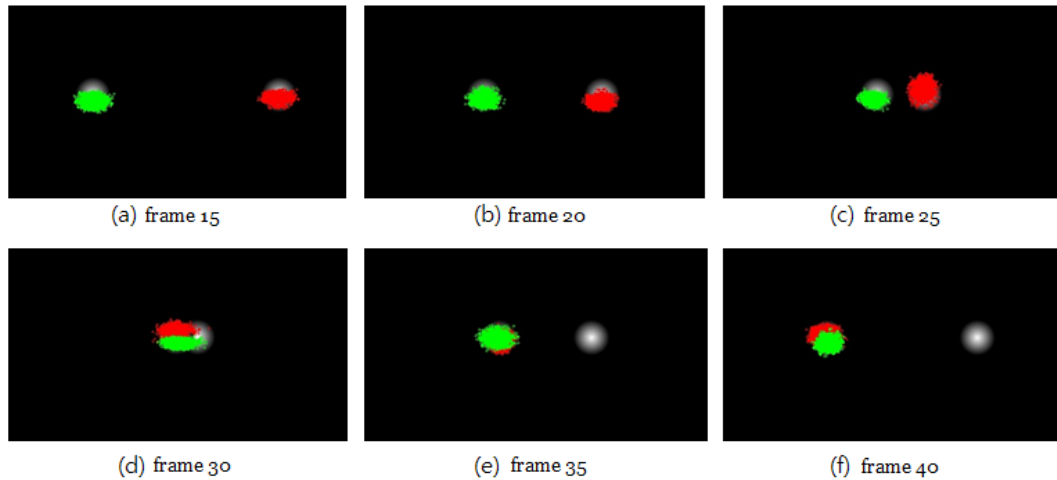


Fig. 1. An example of tracking two targets using two particle filters

However, in this simplicity of using only the likelihood for object tracking, there is a critical disadvantage for tracking multiple targets. **Fig. 1** shows a problem that is created by tracking two targets with two simple particle filters. The particle samples, which independently track each of the two targets, are centralized into the left target, after overlapping at d) of the 30th frame as shown in **Fig. 1**. This problem occurs when the samples are driven to one target with

higher likelihood on the neighboring two targets. Recently, many strategies have been proposed to solve the problem associated with multi-target tracking. The key idea of the literature is to estimate each object's position separately, and the measurements should be assigned to each target. Therefore, data association has been introduced to relate each target to its associated measures. Among the various methods, the Joint Probabilistic Data Association Filter (JPDAF) algorithm [8] is taken into account as one of the most attractive approaches.

3. Disjoint Particle Filter

One of the shortcomings of the data association method [8] is that the computational complexity exponentially increases as the numbers of targets and measurements are increased. Therefore, the objectives of our proposed DPF are reducing the computational complexity and tracking each target efficiently.

3.1 Prediction

Consider the problem of tracking T targets. We use the same number of particle filters as targets. $\mathbf{X}^k = \{(\mathbf{x}_1^k, \pi_1^k), \dots, (\mathbf{x}_T^k, \pi_T^k)\}$, where \mathbf{x}_t^k and π_t^k denote the state vector of the t -th particle filter and the confidence of the filter at time k , respectively. Confidence π_t^k is an indicator that determines the mode of the particle filter. We assume that each measurement data is not identified. Our state vector, \mathbf{x}_t^k , is simply the location, $\mathbf{x}_t^k = [x_t^k, y_t^k]^T$, which is the center of the rectangle with a predefined size for color histogram matching. The rectangle size is not related to the target size and appearance, because the rectangle region is just used for histogram matching. The size of a target is described by the distribution of samples, while the dynamics of the target are described by the simple linear model:

$$\mathbf{x}_{t_i}^k = \mathbf{x}_{t_i}^{k-1} + \Delta \mathbf{x}_{t_i}^{k-1} + G, \quad (4)$$

where, G denotes the Gaussian noise that is in charge of spreading the samples and the variation is determined according to the spreading degree of samples, which is adaptively set by a filter mode.

3.2 Weight Update

As shown in **Fig. 1**, when two targets are overlapped, all samples are centralized into only one target due to the higher likelihood of that target. JPDAF, the data association based technique,

is an efficient way of independently tracking targets by calculating the intersection of each other. However, the computation complexity is inappropriate for a real-time tracker, because the number of associations grows exponentially as the number of targets and the number of measurements increase. To improve the performance, we define the predictive target region (PTR) and propose a method to exclude the samples in the overlapped region at the re-sampling step.

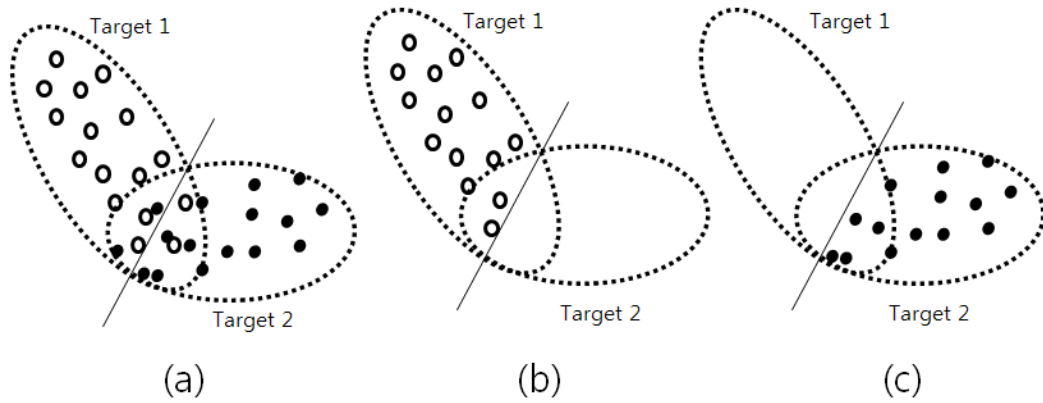


Fig. 2. The concept of mutual exclusion of disjoint particle filters

Fig. 2 shows the boundary of two overlapped PTRs for mutual exclusion at the particle sampling stage. If two particles are overlapped at the PTR, particles from different filters are depicted as dots or circles in Fig. 2, and are used separately for each particle filter. We can compute the PTR centers with $\mathbf{x}_t^{i,k}$, $i=1,2,\dots,N$ that are generated in the re-sampling process.

$$\mu_t^k = \frac{1}{N} \sum_{i=1}^N \mathbf{x}_t^{i,k}. \quad (5)$$

The distance of an i^{th} sample from the PTR center of the t^{th} target is defined by:

$$d_t^{i,k} = (\mathbf{x}_t^{i,k} - \mu_t^k)^T (S_t^k)^{-1} (\mathbf{x}_t^{i,k} - \mu_t^k), \quad (6)$$

where, S_t^k is a covariance matrix of all the sample positions for the t^{th} target. We assume a predictive target distribution function for the i^{th} sample as the Gaussian distribution.

$$P(\mathbf{x}_t^{i,k}) = \frac{1}{(2\pi)^{|S_t^k|^{1/2}}} \exp\left\{-\frac{(d_t^{i,k})^2}{2}\right\}, \quad (7)$$

where, $|S_t^k|$ is the determinant of S_t^k . This is the predictive target distribution function by

sampled from the probability density function. After a few iterations the particle set will contain many particles with weight close to zero and a few particles with high weights. In order to avoid this effect we re-sample the particles that new particle set likely contains multiple copies of particles with high weights whereas particles with low weights are likely to be discarded from the set as shown in Fig.3. The particles with higher weights (larger circles) are chosen with a higher probability, but the total number of samples stays the same.

Sample weights are updated by the observation likelihood, if the predictive target probability of the target being tracked is higher than the predictive target probability of other targets, as follows:

$$\begin{aligned} & \text{if } P(\mathbf{x}_t^{i,k}) > \max(P(\mathbf{x}_{t'}^{i,k}))_{t' \neq t, 1 \leq t' \leq T} \\ & \quad w_t^{i,k} = o_t^{i,k} \\ & \text{else} \\ & \quad w_t^{i,k} = 0. \end{aligned} \quad (9)$$

If the weight is updated for all particle samples in (9), the state vector of the t^{th} target can be estimated as follows:

$$E[\mathbf{x}_t^k] = \sum_{j=1}^N w_t^{j,k} \mathbf{x}_t^{j,k}. \quad (10)$$

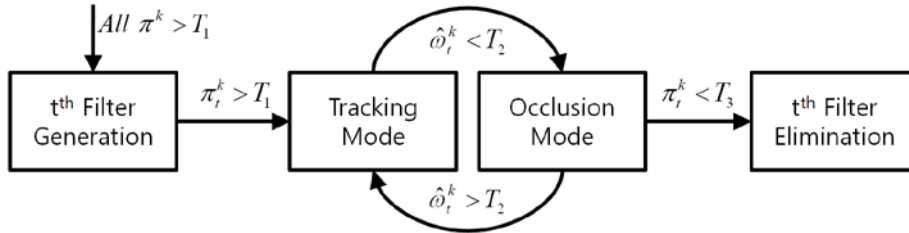


Fig. 4. Mode transition diagram for DPF

3.3 Dynamic Mode for Occlusion

Fig. 4 shows the transition diagram of the DPF containing the dynamic mode. As defined in Section 2, the DPF has the filter confidence π_t^k for each target, and is computed by averaging the weight of all samples.

$$\pi_t^k = \alpha \times \pi_t^{k-1} + (1 - \alpha) \times \hat{w}_t^k, \quad (11)$$

where $\hat{w}_t^k = \sum_{i=1}^N \frac{w_{t,i}^k}{N}$, and α is the weight that adjusts the importance between the previous

confidence π_i^{k-1} and the average of current observations \hat{w}_i^k . π_i^k represents the tracking confidence for the t^{th} target. If this value is high, then the tracking is stable in recent frames. If this value is low, then tracking is not stable up to recent frames. Each process in Fig. 4 is described below.

1) Generation: If π_i^k of all particle filters are higher than the threshold T_1 , then a new particle filter is generated to track targets that are not chased. If there is no target remaining to be tracked, then the generated particle filter will be eliminated. The samples in the generated particle filter are uniformly distributed in the global region of the image at the initial time, and are then converged in one target by applying SIR (sampling importance re-sampling). In this process, if the π_i^k of the generated particle filter is higher than T_1 , then the status of the generated particle filter is changed into the tracking mode.

2) Tracking Mode: Targets are tracked by applying the SIR with (9). In this process, if the \hat{w}_i^k is lower than T_2 , then it will be changed into the occlusion mode.

3) Occlusion Mode: This process is under the situation that most samples missed the target. In this step, instead of the SIR process, all samples are propagated by dynamics with the Gaussian noise, as described in (4). After particles are propagated by (4) and updated by (9), if \hat{w}_i^k is higher than T_2 , then the mode will be changed into the tracking mode.

4) Elimination: If the filter confidence π_i^k is lower than T_3 , then the particle filter will be eliminated.

Our method has tracking and occlusion modes to distinguish between the cases of the lost target and temporary occlusion. If the average of all sample weights is lower than the threshold, then the mode changes into the occlusion mode and keeps tracking by the propagation of particles according to the dynamics of the Gaussian noise. Because the filter confidence is updated by the mean of weights of all the samples in the filter, as shown in (9), the filter confidence decreases in the occlusion mode. If the filter confidence is lower than the threshold, then the filter restarts the tracking for the new target by uniformly distributed samples.

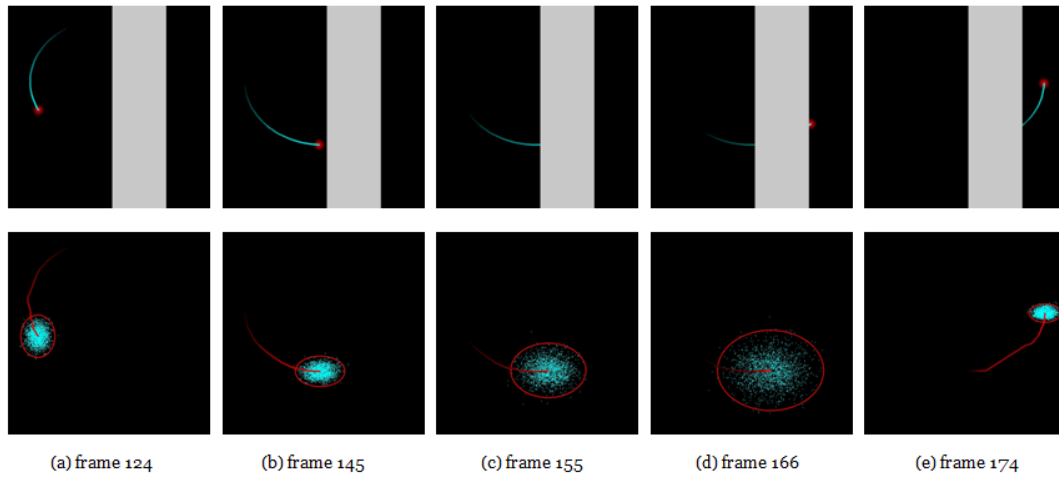


Fig. 5. Occluded target tracking using the dynamic mode

Fig. 5 shows tracking results of the target occluded by an obstacle using the dynamic mode. The images in the first row show that the red color denotes a target moving in a circular trajectory, the cyan line is a trace of the target, and the gray region is an artificially made obstacle. In the second row images, the cyan points mean the particle samples tracking a target, and the predictive target region (PTR) that occurs by particles is presented as a red ellipse.

As shown in **Fig. 5**, the moving direction of the target is changed from (a) the bottom of the 124th frame to (b) the right of the 145th frame. The particles are also propagated from horizontally to vertically, since all the samples are moving according to their dynamics in (4). This property makes it easy to track the target in occlusion mode. The particle filter with the tracking mode at (b) transfers into the occlusion mode at (c), because the target is lost due to the occlusion. Samples in the occlusion mode are diffused by keeping the previous dynamics, so that at (d) the target can be tracked at the appropriate location of the 166th frame. The 174th frame in (e) shows that the filter has changed back into the tracking mode.

4. Experiment

The experiments are performed on a PC with 2GB memory, 2.0 GHz Quad-Core CPU, and the resolution of an input image is 640×480. To apply real-time surveillance systems, we compare the performance of the proposed method with that of JPDAF. In the second experiment, we test the multi-target tracking for soccer players that occlude each other.

4.1 Pixel based Tracking

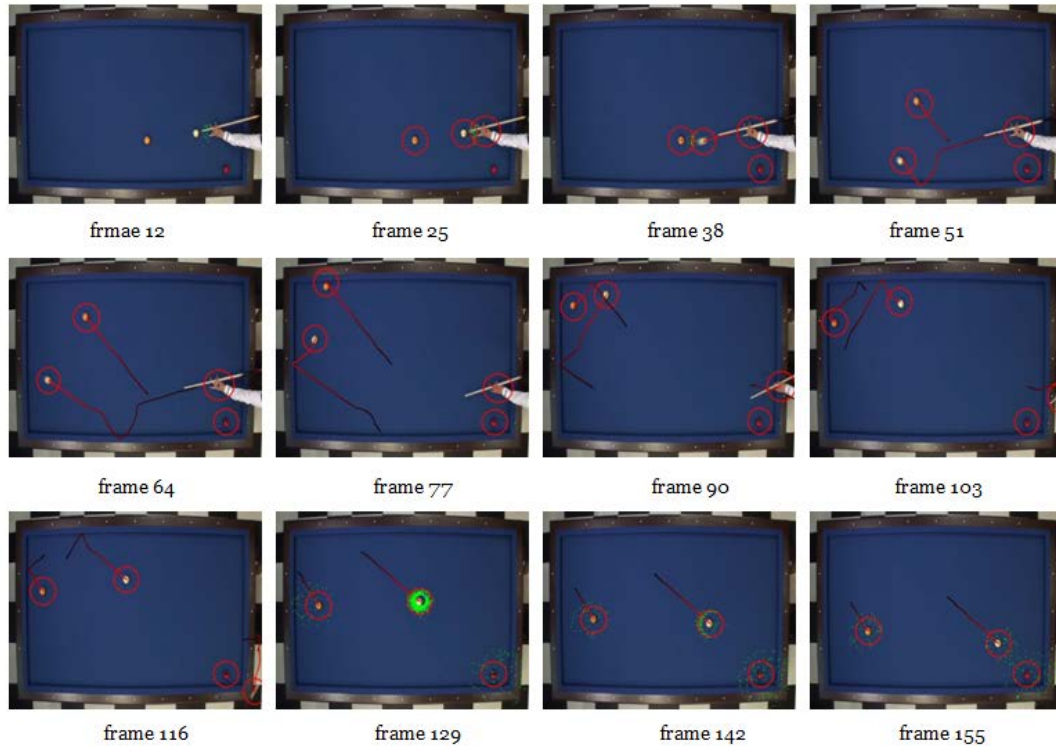


Fig. 6. Multi-target tracking for real-time application

The experiment is performed by increasing the number of targets from one to nine with $N=2000$ for each target, with $T_1=0.9$, $T_2=0.7$, $T_3=0.1$, and $\alpha=0.9$, in (11). For the initialization of the particle filter, we manually select the starting values of the state vector x_k , and the number of particle filters increases from one to the number of targets. The samples in the generated particle filter are uniformly distributed in an image at the initial time then converged in one target by applying the importance re-sampling. After the particles of a filter are converged to a target, a new particle filter is generated according to the scenario in Section 3.3.

Fig. 6 shows the tracking result of billiard ball targets from the reference colors. By using the histogram matching technique described in (8), the particle filter can provide a fine matching result based on various feature vectors. We use the similarity of Cr and Cb in the YCrCb color space as the likelihood instead of the histogram matching, for fair comparison between the data association technique and the proposed technique. **Fig. 6** shows the 12th frame starting with one particle filter, and from 38th to 116th frames there are four particle filters in tracking mode.

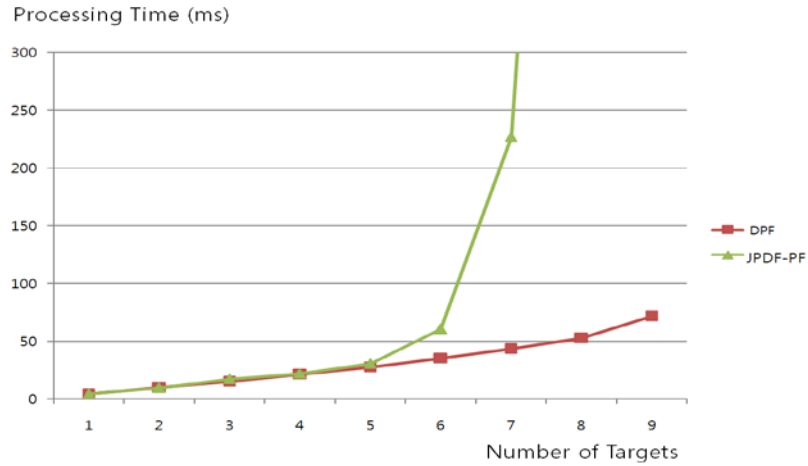


Fig. 7. Comparison of the processing time with DPF and JPDAF-PF

Fig. 7 shows the average processing time of one frame for the particle filter based on JPDAF [9] and the proposed DPF. In JPDAF-PF, we increase the number of targets and the number of measurements equally. Targets are likely to be tracked in real-time until up to five in both methods, even when using 2,000 samples for each target. However, since the processing time of JPDAF-PF exponentially increases according to the number of targets and the number of measurements, it is inappropriate for a real-time tracker. The computational cost of the proposed method is efficient for real-time tracking, because the processing time is only linearly increased.

4.2 Block-based Tracking

Fig. 8 shows the tracking results using DPF for soccer players that occlude each other. In **Fig. 8**, the 130th~200th frames show two soccer players overlapping one another, and the DPF excludes particle samples in the two overlapped PTRs at the re-sampling step. Despite occlusion, the 236th frame shows successful tracking of each player, because DPF can spatially distinguish the two targets in the occluded PTR by sample exclusion. This result shows that the proposed algorithm assures tracking accuracy as well as computational efficiency for real-time surveillance applications.

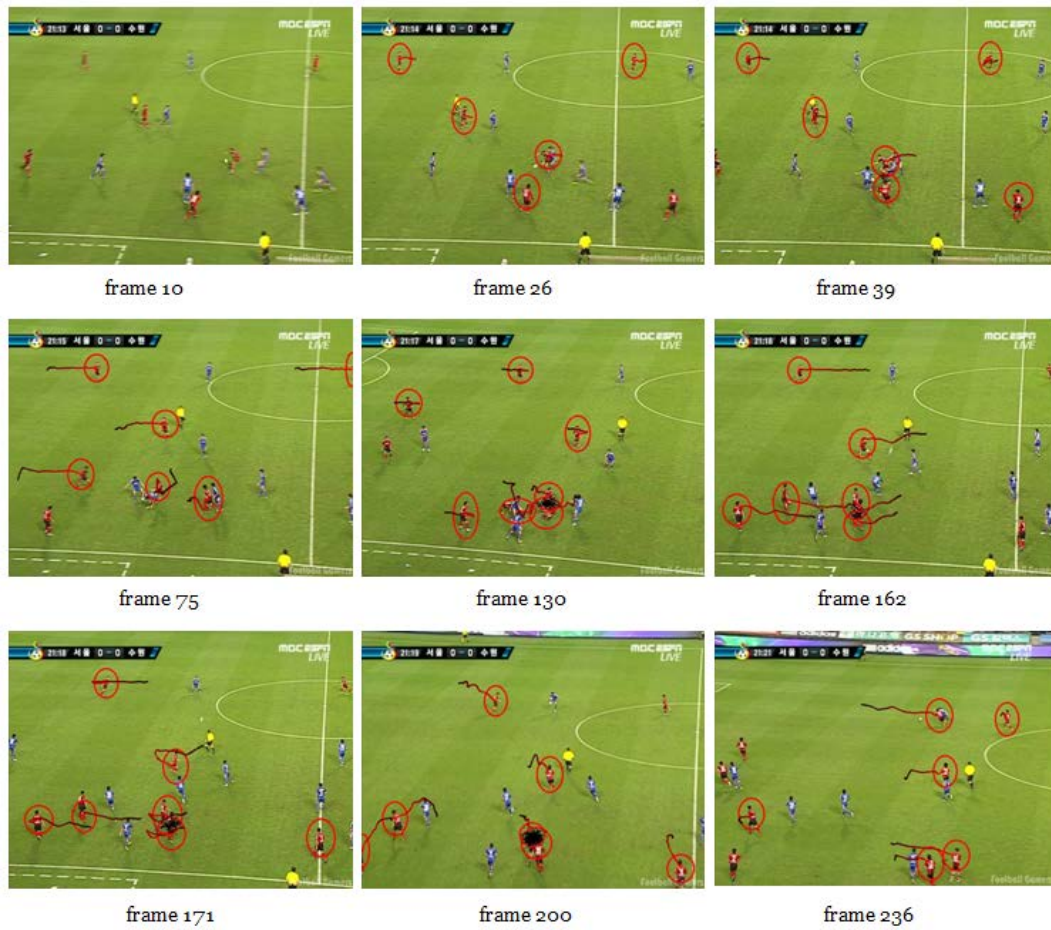


Fig. 8. Tracking soccer players with occlusion

5. Conclusion

The disjoint particle filter intuitively and efficiently tracks multi-target by excluding particles in the overlapped tracking region between the tracking target and other objects. In the case of occlusion, DPF changes the tracking mode with the SIR process into the occlusion mode with Gaussian propagation, and tracking is stable despite the occlusion. In the experiment of tracking targets of billiards and soccer players, we show the robustness in tracking targets and the efficiency in computation. In future work, we will combine the DPF with textural features and it will be applied to real-time surveillance systems.

References

- [1] Y. Bar-Shalom, X. Li and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, J. Wiley and Sons, 2001. [Article \(CrossRef Link\)](#).
- [2] J.-Y. Kim, C.-H. Yi and T.Y. Kim, "ROI-Centered Compression by Adaptive Quantization for Sports Video," *IEEE Transaction on Consumer Electronics*, vol. 56, no. 2, pp. 951-956, 2010. [Article \(CrossRef Link\)](#).
- [3] M. S. Arulampalam, S. Maskell, N. Gordon and 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, 2002. [Article \(CrossRef Link\)](#).
- [4] YoungJoon Chai, SeungHo Shin, Kyusik Chang and TaeYong Kim, "Real-time user interface using particle filter with integral histogram," *IEEE Transaction on Consumer Electronics*, vol. 56, no. 2, pp. 510-515, 2010. [Article \(CrossRef Link\)](#).
- [5] Y. Bar-Shalom and W. D. Blair, *Multitarget-Multisensor Tracking: Applications and Advances, Vol. III.*, Norwood, MA: Artech House, 2000.
- [6] Y. Bar-Shalom, Fred Daum and Jim Huang, "The Probabilistic Data Association Filter," *IEEE Control Systems Magazine*, vol. 29, no. 6, pp. 82-100, 2009. [Article \(CrossRef Link\)](#).
- [7] W. Ng, J. Li, S. Godsill and J. Vermaak, "Tracking variable number of targets using sequential monte carlo methods," in *Proc. of IEEE Statistical Signal Processing Workshop*, pp.1286-1291, July 17-20, 2005. [Article \(CrossRef Link\)](#).
- [8] A. Gorji, M. B. Menhaj and S. Shiry, *Multiple Target Tracking for Mobile Robots Using the JPDAF Algorithm*, Tools and Applications with Artificial Intelligence, pp. 51-68, 2009. [Article \(CrossRef Link\)](#).
- [9] M. Jaward, L. Mihaylova, N. Canagarajah and D. Bull, "Multiple object tracking using particle filters," in *Proc. of the AERO*, 2006. [Article \(CrossRef Link\)](#).
- [10] J. Czyz, B. Ristic and B. Macq, "A color-based particle filter for joint detection and tracking of multiple objects," in *Proc. of the ICASSP*, pp.217-220, 2005. [Article \(CrossRef Link\)](#).
- [11] Cheng Chang, R. Ansari and A. Khokhar, "Multiple Object Tracking with Kernel Particle Filter," in *Proc. of the CVPR*, pp. 566-573, June 20-25, 2005. [Article \(CrossRef Link\)](#).
- [12] K. Okuma, A. Taleghani, N. de Freitas, J. J. Little and D. G. Lowe. "A boosted particle filter: Multitarget detection and tracking," in *Proc. ECCV*, pp. 28-39, May 11-14, 2004. [Article \(CrossRef Link\)](#).
- [13] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier and L. Van Gool, "Markovian tracking-by-detection from a single, uncalibrated camera," in *Proc. IEEE Int. Workshop Performance Evaluation of Tracking and Surveillance*, December 3, 2009. [Article \(CrossRef Link\)](#).
- [14] YoungJoon Chai, JinYong Park, KwangJin Yoon and TaeYong Kim, "Multi Target Tracking Using Multiple Independent Particle Filters For Video Surveillance," in *Proc. of IEEE International Conference on Consumer Electronics*, pp.756-757, January 9-12, 2011. [Article \(CrossRef Link\)](#).



YoungJoon Chai received a BS degree in Information and Communication Engineering from Korea University of Technology and Education in 2003 and a MS degree in Advanced Imaging Engineering at Chung-Ang University, Seoul, Korea. Currently, he is working toward a PhD in Advanced Imaging Engineering at Chung-Ang University.



Hyunki Hong received his BS, MS, and PhD degrees in electronic engineering from Chung-Ang University, Seoul, Korea, in 1993, 1995, and 1998, respectively. From 1998 to 1999 he worked as a researcher in the Automatic Control Research Center, Seoul National University, Korea. From 2000 to 2014, he was a professor in the Department of Imaging Science and Arts, Graduate School of Advanced Imaging Science, Multimedia and Film at Chung-Ang University. Since 2014, he has been a professor in the School of Integrative Engineering, Chung-Ang University. His research interests include computer vision, augmented reality, and multimedia application.



TaeYong Kim received a BS degree in Electrical Engineering and MS degree in Communication Engineering from Han-Yang University, Seoul, Korea, in 1986 and 1988, respectively, and a PhD in Computer Science and Engineering from Pohang University of Science & Technology, Korea, in 1998. He has been a professor at the Graduate School of Advanced Imaging Science, Multimedia and Film, Chung-Ang University, Seoul, Korea since 2003. His research interests are in Computer Vision, Image Processing, and A.I.