DUAL-LAYER PARTICLE FILTERING FOR SIMULTANEOUS MULTIPLE OBJECTS DETECTION AND TRACKING

Kyungwon Jung, Nahyun Kim, Seungwon Lee, and Joonki Paik

Image Processing and Intelligent Systems Laboratory
Graduate School of Advanced Imaging Science, Multimedia, and Film
Chung-Ang University, Seoul, Korea
fjqmsa@gmail.com, skgus1433@gmail.com, superlsw@gmail.com,
and paikj@cau.ac.kr

ABSTRACT

In this paper, we present a novel method for simultaneous detection and tracking of multiple objects using dual-layer particle filtering. For detecting and tracking multiple moving objects, the proposed dual-layer particle filter (DLPF) consists of parent-particles (PPs) in the first layer for detecting multiple objects and child-particles (CPs) in the second layer for tracking objects that are detected in the first layer. In the first layer, PPs detect persons using a classifier prior trained by the intersection kernel support vector machine (IKSVM) at each particle under a randomly selected scale. If a certain PP detects a person, it generates CPs, and makes an object model in the detected object region for tracking the detected object. While PPs that have detected objects generate CPs for tracking, the rest of PPs still move for detecting objects. Experimental results show that the proposed method can automatically detect and track multiple objects and efficiently reduce the computational time using the sampled particles based on motion distribution in video sequences.

***Index Terms*—** Object detection, pedestrian detection, particle filtering, object tracking, motion estimation

1. INTRODUCTION

Tracking one or more moving objects is a fundamental problem in computer vision, and has broad applications, such as object-based auto-focusing, traffic monitoring, vehicle navigation, human computer interaction, augmented reality and intelligent surveillance [1][2].

Conventional tracking methods can be classified as either stochastic or deterministic [3][4]. Adaboost is a popular deterministic approach that has been widely used for the detection of targets, and making a trajectory by connecting successive locations of identified objects. However, the Adaboost detector may fall into a local minimum when there are occlusions or object deformations. Two well-known, popular stochastic approaches include Kalman and the particle filters, both of which recursively estimate the state of a dynamical system from measurements or observations [5]. Kalman filter assumes a linear model for the state dynamics and the measurement equation, and is the optimal estimator when the noise processes are Gaussian. On the other hand, particle filters work for both linear and non-linear dynamical systems, and do not require a Gaussian assumption on the estimation noise. However, conventional particle filtering requires the initially specified region for tracking an object. Okuma et al. have proposed simultaneous multi-target detection and tracking using boosting based particle filtering [6]. However, it requires high computational loads because of the exhaustive search of objects in the entire image. To solve this problem, Gualdi have proposed efficient object detection method in the single image using particle-windows [7]]. It is, however, unsuitable for real-time video object detection and tracking because of the resampling structure.

In this paper, we propose a simultaneous multiple objects detection and tracking method using dual-layer particle filters (DLPFs), such as parent-particles (PPs) in the first layer and child-particles (CPs) in the second layer. The proposed detection and tracking algorithm is shown in Fig. 1.

In the first layer, PPs detect persons using a classifier trained in advance by the intersection kernel support vector machine (IKSVM) at each particle under a randomly selected scale [9]. PPs should be located at the proper object region for accurate, fast detection of objects. Under assumption that objects of interest usually move around in the image, the DLPF estimates the motion in consecutive images, and resamples PPs from the estimated motion distribution.

In the second layer, if a certain PP detects a person, it generates CPs, and makes an object model in the detected object region for tracking the detected object. While PPs that have detected objects generate CPs for tracking, the rest of PPs still move for detecting objects. When an object disappears, the corresponding PP is reset and resampled.

Unlike conventional particle filtering where the tracking starts by manually specifying the initial region of an object,

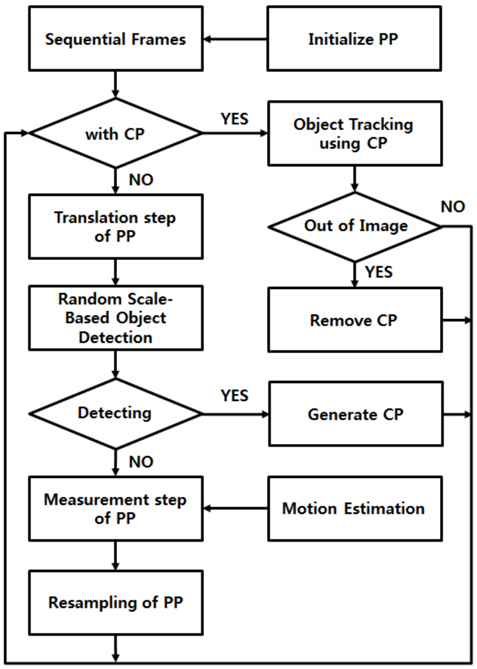


Fig. 1. Proposed dual-layer particle filtering algorithm for detecting and tracking multiple objects.

the proposed method automatically detects the initial region of multiple objects and performs tracking with significantly reduced amount of computation.

Based on experimental results, the proposed simultaneous multiple objects detection and tracking method can be employed in various visual investigation applications such as intelligent video surveillance, human computer interaction, and intelligent transportation systems.

2. DUAL-LAYER PARTICLE FILTERING

This section describes the simultaneous objects detection and tracking method using dual-layer particle filtering (DLPF). Parent-particles (PPs) in the first layer detect objects, and child-particles (CPs) in the second layer tracks object that are detected in the first layer. Fig. 2 illustrates the concept of the proposed DLPF. PPs marked by yellow particles on top are weighted by motion distribution, and resampled to detect an object as shown in the left column of Fig. 2. The PP of a detected object generates CPs marked by brown particles, which are weighted by motion distribution as shown in the right column Fig. 2. This process repeats to track the object until it disappears.

2.1. First layer particle filtering for object detection

In the first layer, PPs detect objects using a classifier prior trained by intersection kernel support vector machines (IKSVM) at each particle under a randomly selected scale.

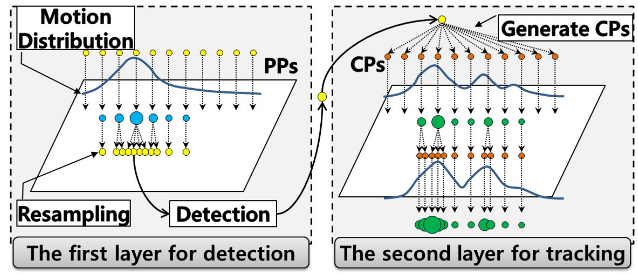


Fig. 2. Concept of the DLPF.

For fast object detection by PPs in video, particles should be properly distributed and located. We estimate the motion of objects being tracked, and resample the PPs by motion distribution-based weighting.

Lucas-Kanade (LK) algorithm has been widely used for estimating motion in the optical flow [10]. From the optical flow equation, motion vector $\mathbf{M} = [d_x, d_y]^T$ for resampling PPs in the neighborhood of the object is calculated by least-square problem as

$$\mathbf{M} = \begin{bmatrix} \sum_\Omega I_x^2 & \sum_\Omega I_x I_y \\ \sum_\Omega I_x I_y & \sum_\Omega I_y^2 \end{bmatrix}^{-1} \begin{bmatrix} -\sum_\Omega I_x I_t \\ -\sum_\Omega I_y I_t \end{bmatrix}, \quad (1)$$

where Ω represents the window region, and I_x and I_y represent spatial derivatives of I with respect to horizontal and vertical directions, respectively, and I_t the temporal derivative of I.

For resampling PPs around of the moving objects, we should consider a particle filtering model in the state space. The state of particles can be represented as $\mathbf{s} = \left[x, y, v^x, v^y\right]^T$, and the state transition model with constant velocity can also be defined as

$$\begin{aligned} x_k &= x_{k-1} + v_{k-1}^x \Delta k, \\ y_k &= y_{k-1} + v_{k-1}^y \Delta k, \\ v_k^x &= v_{k-1}^x, \\ v_k^y &= v_{k-1}^y, \end{aligned} \quad (2)$$

where (v^x, v^y) represents the velocity of the particle. The discrete version of the state transition equations with $\Delta k = 1$ can be expressed as

$$\mathbf{s}_k^- = \mathbf{A}\mathbf{s}_{k-1}^+ + \mathbf{w}_k, \quad (3)$$

where

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \mathbf{and w_k} = \begin{bmatrix} N(0, Q_{pos}) \\ N(0, Q_{pos}) \\ N(0, Q_{vel}) \\ N(0, Q_{vel}) \end{bmatrix}, \quad (4)$$

(5)

where Q_{pos} and Q_{vel} are, separately, position and velocity covariance of process noise. The measurement m_k can be represented in the predicted state $\mathbf{s}_k^-$ with

$$m_k = M(x_k, y_k), \tag{6}$$

where $M(x, y) = \sqrt{d_x^2 + d_y^2}$ represents the magnitude of the motion, and the likelihood is calculated as

$$L = -\log(2\pi R) - \frac{\|y_D\|^2}{\sqrt{2\pi}}, \tag{7}$$

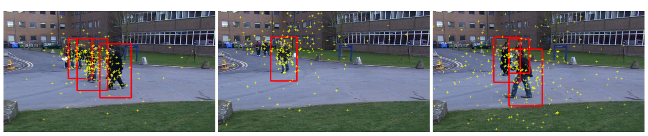
where R represents the standard deviation of the motion estimation noise, $\|y_D\| = y_o - y_k$, where y_0 represents the target motion magnitude which is experimentally set to 5, and the maximum of $M(x, y)$ is set to 5. We then compute the cumulative sum using regularized log-likelihood as

$$c_i = \sum_{m=1}^{i} \frac{L_i}{L_{\text{total}}}, \quad \text{for } i = 1, ..., N, \tag{8}$$

where N is the total number of PPs, L_{total} represents the total sum of L. We generate real random numbers u_i that uniformly distribute in [0,1].

Finally, PPs randomly select the scale for determining the size of a detected object at s_k^+. For $i = 1, \ldots, N$, we find a positive integer j such that $c_{j-1} < u_i$ and $c_j \geq u_i$, and then update the current state as $\mathbf{s}_{k,i}^+ \leftarrow \mathbf{s}_{k,j}^-$.

The result of the first layer particle filtering is shown in Fig. 3. PPs are resampled by the objects movements and the pedestrian detection can be performed at each PP using IKSVM.



(a) Frame 101 (b) Frame 206 (c) Frame 273

Fig. 3. Result of the first layer particle filtering. (red: detected objects, yellow: PPs)

2.2. Second layer particle filtering for object tracking

In the second layer, if a certain PP detects the pedestrian, the corresponding PP generates CPs, and makes an object model in the detected object region for tracking the detected object. The Bayesian approach to the object tracking is to recursively update an estimate of the state of an object $\mathbf{x}_k$ at time k given the measurements or observations, $\mathbf{y}_{1:k}$, of the object up to time k. The state vector may consist of a number of states, such as the two- or three-dimensional position, $\mathbf{x}_k^P$, and the scale $\mathbf{x}_k^s$ of an object. If $p(\mathbf{x}_k|\mathbf{x}_{k-1})$, the probability density function of the state at time k given all of the measurements up to time $k-1$, is known, then the state at time k may be predicted using the Chapman-Kolmogorov equation as

$$p(\mathbf{x}_k|\mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k|\mathbf{x}_{k-1})p(\mathbf{x}_{k-1}|\mathbf{y}_{1:k-1})d\mathbf{x}_{k-1}, \tag{9}$$

where $p(\mathbf{x}_k|\mathbf{y}_{1:k-1})$ is the prior density. Given a new measurement, $\mathbf{y}_k$, at time k, the prior is updated using posterior density computed by Bayes' rule as

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) = \frac{p(\mathbf{y}_k|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k-1})}{p(\mathbf{y}_k|\mathbf{y}_{1:k-1})} \tag{10}$$

where $p(\mathbf{y}_k|\mathbf{x}_k)$ represents the observation model. Equations (7) and (8) form the basis of the optimum Bayes' solution. Unfortunately, the recursive propagation of these densities is generally intractable, and other approaches are used to approximate this solution. One of these is the particle filter, which is a Monte Carlo (MC) method that uses sequential importance sampling.

In the PPs, the posterior $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is approximated by a finite set of N samples $\{x_t^i\}_{i=1,\ldots,N}$ with importance weights w_t^i. The candidate samples $\tilde{x}_k^i$ are drawn independently by sampling from an importance distribution $g(x_k|x_{1:k-1}, y_{1:k})$ and the weight of the samples are computed as

$$w_k^i = \frac{p(\mathbf{y}_k|\tilde{\mathbf{x}}_k^i)p(\tilde{\mathbf{x}}_k^i|\mathbf{x}_{k-1}^i)}{g(\mathbf{x}_k|\mathbf{x}_{1:k-1}, \mathbf{y}_{1:k})}. \tag{11}$$

To avoid degeneracy, the resampling process generates a set of unweighted particles according to their importance weights to avoid degeneracy. In the bootstrap filter, $g(\mathbf{x}_k|\mathbf{x}_{1:k-1}, \mathbf{y}_{1:k}) = p(\mathbf{x}_k|\mathbf{x}_{k-1})$ and the weights become the observation model $p(\mathbf{y}_k|\mathbf{x}_k)$. The basic idea of the particle filter is to represent the posterior density by a set of random samples with associated weights and to compute estimates of these states, such as expected values, using these weights and samples.

Prediction model $p(\mathbf{x}_k|\mathbf{x}_{k-1})$ uses the second-order autoregressive process, and the noise model is defined by the Gaussian function. The observation model uses the Hue-Saturation-Value (HSV) color histogram, which decouples chromatic information from shading effects, for robust illumination changes in environments of moving camera [8]. HSV histogram is composed of $M = M_h M_s + M_v$ bins and $h_k(z)$ is defined to the bin index at location z in time k.

The color distribution $\mathbf{q_k}(\mathbf{x}) = \{q_k(m; \mathbf{x})\}_{m=1\ldots M}$ at time k is given as

$$q_k(m; \mathbf{x}) = K \sum_{z \in R(\mathbf{x})} \delta[h_k(\mathbf{z}) - m], \tag{12}$$

where δ represents the Kronecker delta function, K is a normalization constant, and $R(x)$ is the sampled object region. For measuring the similarity between the observation and the reference models, Bhattacharyya distance measurement is used as

$$\mathbf{B}[\mathbf{q}^*, \mathbf{q_k}(\mathbf{x})] = \left[1 - \sum_{m-1}^{M} \sqrt{q^*(m)q_k(m; \mathbf{x})}\right]^{\frac{1}{2}}, \tag{13}$$

where $\mathbf{q}^*$ represents the reference color model. The reference distribution is obtained at the initial time k_0 .

Similarity measurement of the observation model is defined by Bhattacharyya distance as

$$w_k^i \propto e^{-\lambda \mathbf{B}^2[\mathbf{q}^*, \mathbf{q_k}(\mathbf{x})]} \tag{14}$$

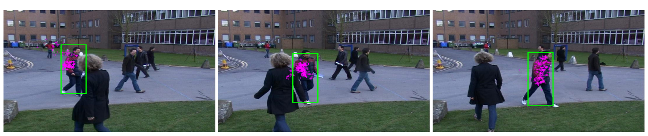
For example, the maximum *a posteriori* (MAP) estimate of the state at time k may be found from N particles (samples) as follows

$$\mathbf{x}_k^{\mathrm{MAP}} = \underset{\mathbf{x}_k^i}{\arg\max}\, w_k^i \quad \text{for} \quad i = 1, ..., N. \tag{15}$$

Finally, x_k^{MAP} among the CPs is selected as the state of the object at the time k. The CPs generated by the PP can continuously track the detected object, and the other PPs keeps moving for detecting objects. When an object being tracked disappears, the corresponding PP is reset and resampled.

In the first layer, if PPs are resampled by motion distribution in both tracked and untracked regions, PPs can be stuck in the tracked moving object by CPs. To overcome this problem, the motion in the tracked object region is set to zero. Some of false positive detection can be solved by selecting the object, which is sequentially detected 4 times.

As shown Fig. 4, result of the second layer particle filtering show that a moving object can be tracked by CPs.



(a) Frame 21 (b) Frame 25 (c) Frame 29

Fig. 4. Result of second layer particle filtering. (magenta: CPs, green: tracked objects region)

3. EXPERIMENTAL RESULTS

To evaluate the effectiveness of the proposed approach for simultaneous multiple objects detection and tracking, we tested the algorithm in PETS2009 and PETS2006 dataset of 2000 frames whose resolution are 768×576 [11][12] .

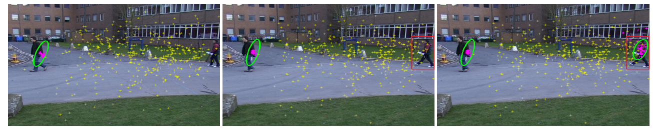
Experimental results of performance of simultaneously detection and tracking are shown in Fig. 5 and Fig. 6. Experimental results show that the proposed method can robustly track the object and be worked well for general video image.

4. CONCLUSION

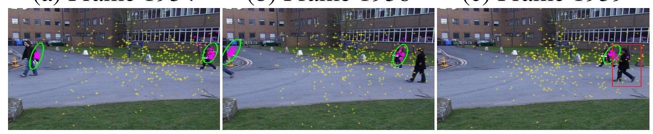
In this paper, we presented multiple targets tracking method using dual-layer-based particle filtering in the visual surveillance system. Conventional object tracking methods can track the initially specified region. A boosting-based tracking method requires an intractably long processing time because objects should be detected in the entire image. To solve these problems, we used DLPF that consist of detection and tracking layers. To simultaneously detect and track multiple targets, parent-particles (PP) and child-particles (CPs) based on DLPF ware generated and used.

Based on experiment results, the proposed simultaneous multiple objects detection and tracking method can be applied such as the PTZ camera in video surveillance system, the human computer interaction, and the intelligent transport system.

In the future research, we will provide comparison results between the proposed and existing methods to evaluate the performance.



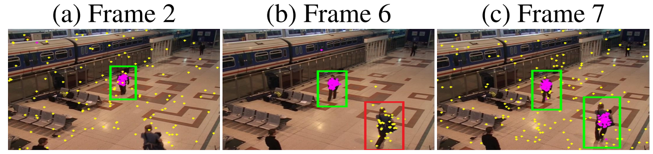
(a) Frame 1934 (b) Frame 1938 (c) Frame 1939



(d) Frame 1943 (e) Frame 1950 (f) Frame 1954

Fig. 5. Experimental results of performance of simultaneously detection and tracking (yellow: PPs, magenta: CPs, red: detected objects, green: tracked objects region)

(a) Frame 2 (b) Frame 6 (c) Frame 7



(d) Frame 37 (e) Frame 59 (f) Frame 63

Fig. 6. Experimental results of performance of simultaneously detection and tracking (yellow: PPs, magenta: CPs, red: detected objects, green: tracked objects region)

5. ACKNOWLEDGEMENT

This research was supported by Basic Science Research Program through National Research Foundation (NRF) of Korea funded by the Ministry of Education, Science and Technology (2009-0081059) and by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2012 (00045420-1).

6. REFERENCES

[1] J. Kim, D. Yeom, and Y. Joo, "Fast and robust algorithm of tracking multiple moving objects for intelligent video surveillance systems," *IEEE Trans. Consumer Electronics*, vol. 57, no. 3, pp. 1165-1170, August 2011.

[2] Y. Chai, S. Shin, K. Chang, and T. Kim, "Real-time user interface using particle filter with integral histogram," *IEEE Trans. Consumer Electronics*, vol. 56, no. 2, pp. 510-515, May 2010.

[3] Y. Rui and Y. Chen, "Better proposal distributions: object tracking using unscented particle filter," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, vol. 2, pp. 786-793, 2001.

[4] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," *Proc. British Machine Vision Conference*, vol. 1, pp. 47-56, 2006.

[5] Z. Chen "Bayesian filtering: From Kalman Filters to particle filters, and beyond," *Technical Report McMasters University, Hamilton*, 2003

[6] K. Okuma, A. Taleghani, N. Freitas, J. Little, and D. Lowe, "A boosted particle filter: multitarget detection and tracking", *Proc. European Conference on Computer Vision*, vol. 3021, pp. 28-39, 2004

[7] G. Gualdi, A. Prati, R. Cucchiara "Multi-stage particle windows for fast and accurate object detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.34, no. 8 , pp. 1589-1604, August 2012

[8] P. Peerez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," *Proc. European Conference on Computer Vision*, pp. 661-675, 2002.

[9] S. Maji, A. Berg, and J. Malik, "Classification using intersection kernel support vector machines is efficient," *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 1-8, 2008

[10] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," *Proc. DARPA Image Understanding Workshop*, pp. 121-130, 1981.

[11] PETS 2006 Benchmark Data, http://www.cvg.rdg.ac.uk/PETS2006/data.html.

[12] PETS 2009 Benchmark Data, http://www.cvg.rdg.ac.uk/PETS2009/a.html.