

Switching Particle Filters for Efficient Real-time Visual Tracking

Takashi Bando^{1,3}, Tomohiro Shibata^{1,2,3}, Kenji Doya^{1,2,3} and Shin Ishii^{1,3}

{takash-b, tom, ishii}@is.naist.jp, doya@atr.jp

¹ Nara Institute of Science and Technology, Nara, 630-0192 JAPAN

² Computational Neuroscience Laboratories, ATR International

³ CREST, Japan Science and Technology Agency

Abstract

Particle filtering is an approach to Bayesian estimation of intractable posterior distributions from time series signals distributed by non-Gaussian noise. A couple of variant particle filters have been proposed to approximate Bayesian computation with finite particles. However, the performance of such algorithms has not been fully evaluated under circumstances specific to real-time vision systems.

In this article, we focus on two filters: Condensation and Auxiliary Particle Filter (APF). We show their contrasting characteristics in terms of accuracy and robustness. We then propose a novel filtering scheme that switches these filters, according to a simple criterion, for realizing more robust and accurate real-time visual tracking. The effectiveness of our scheme is demonstrated by real visual tracking experiments. We also show that our simple switching method significantly helps online learning of the target dynamics, which greatly improves tracking accuracy.

1. Introduction

Particle Filtering is an approach to Bayesian estimation of intractable posterior distributions from time series signals with non-Gaussian noise, as such that generalizes the traditional Kalman filtering methods. This approach has been attracting attention in various research areas, including real-time visual processing [1][3][6], which deals with images contaminated by non-Gaussian noises due to not only signal noises but also obstacles and/or distracters. Several particle filters have been proposed for approximating Bayesian computation with finite particles. The performances of such algorithms have not, however, been fully evaluated under circumstances specific to real-time vision systems.

In this article, we focus on two filters: Condensation [3] and Auxiliary Particle Filter [4] (APF). Condensation was the first successful implementation of a particle filter for real-time visual tracking, which employs the Sampling Importance Resampling (SIR) method [2], and can run on a

cheap standard PC. The APF was proposed to solve a certain problem with the basic particle filter. We show their contrasting characteristics; actually, they can be considered compensatory in terms of accuracy and robustness under the circumstances specific to real-time vision systems. To exploit their advantages, we propose a novel filtering scheme that switches these filters according to a simple criterion. We demonstrate the effectiveness of our scheme with real experiments in which the task is to track a moving ball passing behind an occluder. We also show that our simple switching method significantly helps online learning of target dynamics, which greatly improves tracking accuracy.

2. Switching Particle Filters

2.1. Particle Filters

Particle filters are based upon point-mass representation of probability densities. Let $\mathbf{x}_t \in \mathcal{R}^{N_x}$ denote the continuous state vector of a tracking target at time step t , and $\mathbf{z}_t \in \mathcal{R}^{N_z}$ a measurement vector. Using a point-mass set $\{(\mathbf{x}_{t-1}^{(n)}, \pi_{t-1}^{(n)}), n = 1, \dots, N\}$ at time step $t-1$, the filtered posterior at time step t is approximated as

$$p(\mathbf{x}_t|\mathbf{z}_t) \approx k_t p(\mathbf{z}_t|\mathbf{x}_t) \sum_n \pi_{t-1}^{(n)} p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(n)}), \quad (1)$$

in which the prediction step $p(\mathbf{x}_t|\mathbf{z}_{t-1})$ is approximated as $\sum_n \pi_{t-1}^{(n)} p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(n)})$, and $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ is the state transition prior. For the next iteration step, sampling from the posterior described by Eq. (1) is called *resampling*.

In Condensation, $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ used as the proposal distribution is independent of the observations \mathbf{z}_t at each time step t , and the state space is explored regardless of \mathbf{z}_t . This property suffers from the outlier problem [4], i.e., model-implausible observations that may occur when there are unexpected occluders, distracters, and changes in the target motion. For example, Fig. 1 (left) shows that most of the particles drawn from the transition prior have low likelihood. Then the position becomes inaccurate which produces poor tracking.

Since the particle filters use a sampling method with a limited number of particles, its performance is largely attributed to which resampling method is used. Furthermore, a couple of algorithms have been proposed to solve Condensation's outlier problem [4].

The Auxiliary Particle Filter [4] proposed by Pitt and Shepherd includes elegant resampling method that solves the outlier problem. In their approach, the likelihoods are calculated as weights at any likely point that characterizes $p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(n)})$. Then, by resampling with the weights, we can accurately estimate the posterior depicted in Fig. 1 (left). By conducting simulations using artificial images, we tested the accuracy of the APF. Figure 1 (right) presents the averaged finding speed (upper), and the standard deviation σ (lower) over 500 trials, with the abscissa denoting the number of particles. The finding speed denotes the number of frames necessary for almost all particles to find the target after at least one of them has found it. This figure shows that the APF has a faster finding speed regardless of the number of particles.

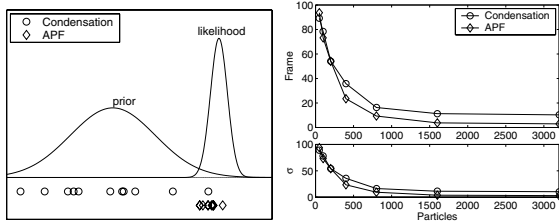


Figure 1. Particles drawn from the proposal distribution (left), and finding speed (right).

2.2. Switching the Particle Filters

Although it seems that the APF provides better performance than Condensation, we found that it often loses the target when it is occluded and/or distracted (See Section 4.2), which is a natural condition in the real world. Because the APF weights its observation strongly, it tends to lose diversity among particles. In contrast, we found that it is very difficult to distract Condensation, since it gives greater weight to its prediction than to its observation. Thus, Condensation and the APF seem compensatory in terms of accuracy and robustness under realistic circumstances. To exploit their advantages, we propose a novel scheme that switches these algorithms dynamically. In this scheme a confidence matrix, $V_{\text{est}} \in \mathcal{R}^{N_x \times N_x}$, the covariance matrix of the current estimated target state, is used for the switching. The confidence matrix that shows the confidence level of the current estimated target state in the target space is $V_{\text{est},ij} = \sigma_{\text{est},ij}^2(i, j = 1, \dots, N_x)$, where

$$\sigma_{\text{est},ij}^2 = \frac{\sum_n \pi_t^{(n)} (\hat{x}_{t,i} - x_{t,i}^{(n)})(\hat{x}_{t,j} - x_{t,j}^{(n)})}{N - 1}, \quad (2)$$

and \hat{x}_t is the expectation of a target state \mathbf{x}_t at time step t . Equipped with a threshold matrix γ , the switching scheme is depicted in Fig. 2 and described as:

if $\sigma_{\text{est},ij} > \gamma_{ij} (\exists i, j)$, then use APF,
otherwise, use Condensation.

The strategy of the switching scheme can be stated as:

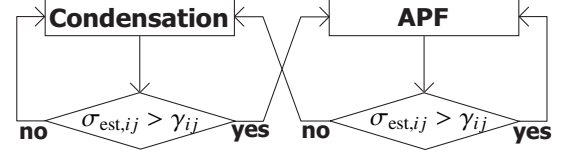


Figure 2. A flowchart of switching method.

“Don’t believe the thing you see too much if you are confident with your estimation.” We also emphasize that this switching scheme contributes to reducing computational costs, since it maintains only N particles, rather than $2 \times N$.

3. Experimental Setup

The task in our experiments is to track a red ball moved sinusoidally either by hand or by an industrial manipulator. Real video images were taken by a digital video camera and processed by a Pentium 4 (2 GHz) Linux PC. These images were downloaded to the PC by Video for Linux at a resolution of 640×480 pixels, then reduced to 320×240 . Our system processed one frame in only about 8 msec.

3.1. Likelihood of the Target

For the particle filters, we need to prepare the observation model, or the likelihood function, of the ball. In general, the more precise and complicated the model is, the less robustly the model would fit and find its actual target, i.e., the likelihood tends to be too peaked. Moreover, it is time-consuming for each particle to calculate such complicated model. Thus, we employ a simple model that roughly approximates the likelihood with color information. The red ratio at pixel $\mathbf{x} = (x_1, x_2)$ in the image coordinates is given by

$$r(\mathbf{x}) = \frac{R(\mathbf{x})}{R(\mathbf{x}) + G(\mathbf{x}) + B(\mathbf{x})}, \quad (3)$$

where $R(\mathbf{x})$, $G(\mathbf{x})$, and $B(\mathbf{x})$ are observed colors of red, green, and blue, respectively. Then, the weight $\pi_t^{(n)}$ of a sample point $\mathbf{x}_t^{(n)}$ is given by

$$\pi_t^{(n)} \propto \alpha + \frac{1}{\sqrt{2\pi}\sigma_h} \exp\left\{-\frac{(r(\mathbf{x}_t^{(n)}) - \mu_h)^2}{2\sigma_h^2}\right\}, \quad (4)$$

where μ_h and σ_h^2 are the mean and the variance of a red ratio, respectively, which are calculated for the pixels composing the actual red ball beforehand. We set α to 10% of

the maximum value of the above Gaussian term, such as to represent the background level, cf. [3].

3.2. Switching Threshold and Particle Size

For simplicity we assume that the 2D-size of the ball is constant, which means the distance between the camera and the ball is fixed. We also assume that the random variables x_i , ($i = 1, 2$) are independent. Under this assumption, the covariance matrix $V_{\text{est}}(t)$ becomes diagonal. Then, all we need to consider for the switching threshold are γ_{ii} , ($i = 1, 2$). Since we confirmed that the radius of the ball can be used as the standard deviation of the well-estimated target state, we preset the switching threshold γ_{ii} to the ball radius (8 pixels) in our actual experiments.

We also have to determine the number of particles. From simulation results using artificial images, we set the number of particles to 800 to maintain the accuracy and real-time quality, cf. Fig. 1 (right).

3.3. Target Dynamics Models and Learning

Since the prediction step $p(\mathbf{x}_t|\mathbf{z}_{t-1})$ is approximated as $\sum_n \pi_{t-1}^{(n)} p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(n)})$, it is better to have correct knowledge of the dynamics model of the target for appropriate tracking. A previous research effort attempted to learn it by a batch EM algorithm [5]; however, the target dynamics is unknown in general, thus one needs to learn the dynamics model in an online fashion. Moreover, online learning usually includes a forgetting scheme, which is suitable for tracking of a target whose dynamics is smoothly changing.

In our experiments, we assume that target dynamics is a second-order linear system. In the case of no learning as well as initial parameters for learning, we assume the target dynamics is $\mathbf{x}_{t+2} = 2\mathbf{x}_{t+1} - \mathbf{x}_t + \mathbf{w}_t$, i.e., constant velocity motion, where \mathbf{w}_t is white gaussian noise with mean=0, variance=1. See the Appendix for our learning algorithm.

4. Experimental Results

First of all we show how online learning of the target dynamics contributes to tracking, with or without the switching. We then show how our switching method outperforms other filters in a realistic task: tracking with an occluder and distracters.

4.1. Learning the Target Dynamics

Figure 3 (left) is a sample image depicting the task environment, in which a red ball was moved sinusoidally by a human. Successful learning was achieved except in the case of the too-narrow time window for Kalman smoothing. Empirically, we set the window size to 60 frames.

Figure 3 (right) shows representative tracking trajectories acquired by Condensation with or without learning.

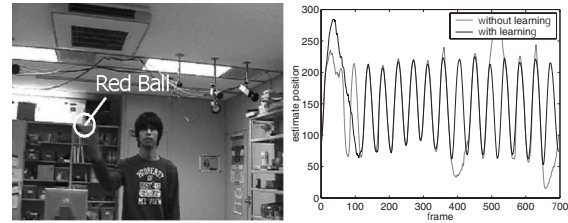


Figure 3. Online learning of a dynamics model. A sample image (left), and estimated trajectories (right).

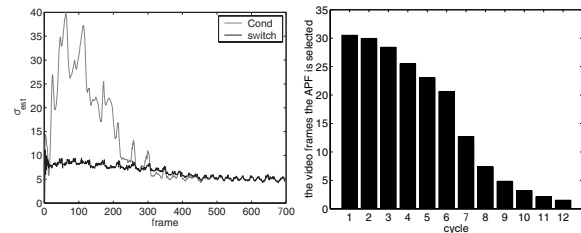


Figure 4. Learning progress. Standard deviations of the target state (left), and the number of frames that APF selected (right).

With learning, the tracking improves its robustness compared to the tracking without learning, although it sometimes loses the target in the early phase of learning.

Figure 4 (left) shows the relationship between the learning progress and the standard deviation of the estimated target state. As the learning progresses, σ_{est} got smaller and the tracker became armed with confidence. However, with the switching, it kept small from the beginning.

Figure 4 (right) shows the number of frames the APF selected according to the cycle of the ball's sinusoidal motion. The number of frames decreased as learning progressed, which means that the observation was more weighted in the early phase of learning, whereas the prior was more weighted as the learning converged.

4.2. Tracking with an Occluder and distracters

To statistically compare the performance of different particle filters under normal circumstances for vision systems, we prepared an environment in which we placed a board as an occluder between the red ball and the camera, cf. Fig. 5 (left). There were also some distracters, i.e., red objects around the trajectory and the board.

The bar graph in Fig. 5 (right) represents the mean absolute position errors acquired over 50 trials by different particle filters without learning dynamics. The true ball position behind the board was linearly interpolated based on the observed data.

Table 1 shows success rates in the same task with or without learning dynamics. “Success” means that a tracking did not fail for 420 frames. The denominators are the trial numbers and the numerators are success counts.

These figures show that our switching particle filter outperforms other filters, including Unscented Particle Filter (UPF) [6], whose efficiency for visual tracking has already been shown. Because the APF took particular note of the current observation, it sometimes failed to track the target as it crossed behind the board due to distracters. Condensation was more robust for crossing the board, but subsequent convergence to the target was very slow and not promising. The UPF was more robust and accurate than those two filters, but its accuracy was much worse than that of the switching method.

We found that it was very difficult to learn the target dynamics in this type of severe circumstance: surprisingly, the success rates of Condensation and the APF were 0 in the learning task. In general, an occluder and distracters tend to degrade tracking accuracy, leading to learning of irrelevant parameters. This vicious circle accounts for those low success rates. Despite that, our switching method worked very well, presumably because of one of its functions that automatically controls the balance between observation and prediction (see 4.1)

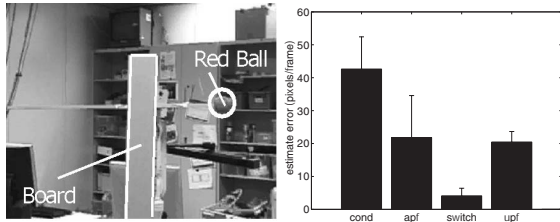


Figure 5. A sample image in the occlusion task (left), and mean absolute position errors (right).

	with learning	without learning
Cond	0/50	2/50
APF	0/50	25/50
UPF	45/50	47/50
Switch	44/50	49/50

Table 1. Success rate in occlusion task.

5. Conclusion

We proposed a novel particle filtering scheme, “switching particle filter,” for real-time vision systems. The switching particle filter uses two particle filters that incorporate different resampling algorithms. The two particle filters are Condensation and Auxiliary Particle Filter (APF). Through simulations and real experiments where

distracters and occluders exist, we found that their inherent properties are compensatory in terms of accuracy and robustness. To exploit their advantages, our proposed filter switched them dynamically according to a simple criterion: the confidence level of the current estimated target state. We demonstrated that the switching particle filter outperforms other well-known particle filters through real visual tracking experiments. We have also shown that the switching scheme significantly helps online learning of target dynamics, which greatly improves tracking accuracy. In this article, the threshold parameter for switching was preset to the ball radius. Automatic determination of this parameter can be regarded as learning of a hyper-parameter, which will be our future work.

Appendix

For online learning, fixed-lag Kalman smoothing (window size T) is applied to filtering estimates to time step t , obtaining

$$S_i(t) = (1 - \eta)S_i(t-1) + \eta S_i(t), \quad (5)$$

$$S_{ij}(t) = (1 - \eta)S_{ij}(t-1) + \eta S_{ij}(t), \quad (6)$$

where η is a learning coefficient, and

$$S_i = \sum_{t=1}^{T-K} \hat{\mathbf{x}}_{t+i}, \quad S_{ij} = \sum_{t=1}^{T-K} \hat{\mathbf{x}}_{t+i} \hat{\mathbf{x}}_{t+j}^T - \frac{1}{T-K} S_i S_j^T, \quad (7)$$

which are the first-order moments and the autocorrelations calculated in the window. This scheme can be achieved online. If learning coefficient η is scheduled, the online learning will progress smoothly; in the experiments, however, we fix $\eta = 0.01$ because the target may behave aperiodically in the real world.

References

- [1] M. J. Black and A. D. Jepson. Recognizing temporal trajectories using the condensation algorithm. In *IEEE Int. Conf. Automat Face Gesture*, pages 16–21, 1998.
- [2] N. Gordon, J. Salmond, and A. Smith. Novel approach to nonlinear non-gaussian bayesian state estimation. In *IEEE Proc. Radar Signal Processing*, volume 140, pages 107–113, 1993.
- [3] M. Isard and A. Blake. Condensation-conditional density propagation for visual tracking. *IJCV*, 29(1):5–28, 1998.
- [4] M. k. Pitt and N. Shephard. Filtering via simulation: Auxiliary particle filter. *Journal of American Statistical Assosiation*, 94:590–599, 1999.
- [5] B. North and A. Blake. Learning dynamical models using expectation-maximisation. In *ICCV*, 1998.
- [6] Y. Rui and Y. Chen. Better proposal distributions: Object tracking using unscented particle filter. In *CVPR*, 2001.