

VISUAL TRACKING USING SEQUENTIAL IMPORTANCE SAMPLING WITH A STATE PARTITION TECHNIQUE

Yan Zhai¹, Mark Yeary¹, Joseph P. Havlicek¹, Jean-Charles Noyer², Patrick Lanvin²

¹School of Electrical and Computer Engineering
University of Oklahoma, Norman, OK USA

²Laboratoire d'Analyse des Systèmes du Littoral
Université du Littoral Côte d'Opale, Calais Cedex, France

Corresponding Author's E-mail: yan.zhai@ieee.org, yeary@ieee.org

ABSTRACT

Sequential importance sampling (SIS), also known as particle filtering, has drawn increasing attention recently due to its superior performance in nonlinear and non-Gaussian dynamic problems. In the SIS framework, estimation accuracy depends strongly on the choice of proposal distribution. In this paper we propose a novel SIS algorithm called *PF-SP-PEKF* that is based on a state partition technique and a parallel bank of extended Kalman filters designed to improve the accuracy of the proposal distribution. Our results show that this new approach yields a significantly improved estimate of the state, enabling the new particle filter to effectively track human subjects in a video sequence where the standard condensation filter fails to maintain track lock. Moreover, because of the improved proposal distribution, the new filter can achieve a given level of performance using fewer particles than its conventional SIS counterparts.

1. INTRODUCTION

Robust, accurate visual object tracking is fundamental to a variety of computer vision applications including robotics, human tracking and biometric identification, intelligent transportation systems, smart rooms, and military targeting systems. Usually, the objects of interest in a video sequence are represented by state space models that may involve strong nonlinearities. In addition, the noise and background clutter that are almost always present in real-world video sequences make the visual tracking problem particularly challenging. Conventional methods for dealing with nonlinear and non-Gaussian problems include the extended Kalman filter (EKF), Bayesian multiple-hypothesis filters, and approximate variants including the probability data association filter (PDAF) and joint probability data associa-

tion filter (JPDAF) [1]. Hidden Markov models (HMM) have also been widely used [2, 3].

More recently, a sequential importance sampling (SIS) technique, *viz.* particle filtering (PF), has been recognized as a significant target tracking algorithm because of its accuracy, robustness and flexibility in non-linear and non-Gaussian scenarios [4]. Various PF tracking approaches have been reported, including the condensation method [5], switching PF [6], unscented PF [7] and others. However, performance of these techniques is strongly influenced by the choice and accuracy of the proposal distribution (PD). In this paper, we introduce a new PF approach where the PD is generated using the state partition technique and multiple weighted EKFs. This technique provides highly accurate and robust state estimates in the presence of noise and strong clutter, which improves tracking performance while simultaneously reducing the need for a large number of particles. As an illustrative example, we apply the new approach to track human perambulation in a highly cluttered environment.

2. SEQUENTIAL IMPORTANCE SAMPLING

The conventional PF technique [4] is based on a nonlinear system modeled in state space according to

$$\begin{aligned}\mathbf{x}(t) &= f(\mathbf{x}(t-1)) + \mathbf{v}(t-1) \\ \mathbf{y}(t) &= h(\mathbf{x}(t)) + \mathbf{n}(t)\end{aligned}\quad (1)$$

where $\mathbf{x}(t)$ and $\mathbf{y}(t)$ denote the state vector and observations and where $\mathbf{v}(t)$ and $\mathbf{n}(t)$ are the process and observation noises. The filtering objective is to simulate the posterior distribution $p(\mathbf{x}(0:t)|\mathbf{y}(1:t))$ via a large set of randomly generated samples called *particles*. Applying Bayes' theorem, this distribution is given by

$$p(\mathbf{x}(0:t)|\mathbf{y}(1:t)) = \frac{p(\mathbf{y}(1:t)|\mathbf{x}(0:t))p(\mathbf{x}(0:t))}{\int p(\mathbf{y}(1:t)|\mathbf{x}(0:t))p(\mathbf{x}(0:t)) d\mathbf{x}(0:t)}.$$

This work was supported in part by the U.S. Army Research Laboratory and the U.S. Army Research Office under grant W911NF-04-1-0221.

SIS approximates the posterior distribution according to

$$p(\mathbf{x}(0:t)|\mathbf{y}(1:t)) \approx \sum_{i=1}^{N_s} \tilde{\omega}^{(i)}(t) \delta(\mathbf{x}(0:t) - \mathbf{x}^{(i)}(0:t)), \quad (2)$$

where N_s is the number of particles. The normalized importance weight $\tilde{\omega}^{(i)}(t)$ is given by

$$\tilde{\omega}^{(i)}(t) = \frac{\omega^{(i)}(t)}{\sum_{j=1}^{N_s} \omega^{(j)}(t)}, \quad (3)$$

where $\omega^{(i)}(t)$ denotes the importance weight associated with the i^{th} particle at time t , given by

$$\omega^{(i)}(t) = \omega^{(i)}(t-1) \frac{p(\mathbf{y}(t)|\mathbf{x}^{(i)}(t)) p(\mathbf{x}^{(i)}(t)|\mathbf{x}^{(i)}(t-1))}{q(\mathbf{x}^{(i)}(t)|\mathbf{x}^{(i)}(0:t-1), \mathbf{y}(1:t))}. \quad (4)$$

The particles $\mathbf{x}^{(i)}(t)$ are drawn from the PD $\mathbf{x}^{(i)}(t) \sim q(\mathbf{x}(t)|\mathbf{x}^{(i)}(0:t-1), \mathbf{y}(1:t))$. Designing a proper PD $q(\cdot)$ associated with the SIS is one of the greatest challenges in implementing a PF and can profoundly affect the filter performance. In the next section, we introduce a new, robust approach to this important task.

3. NEW PF ALGORITHM (PF-SP-PEKF)

It is well-known that using the state transition prior as the proposal (also known as *condensation*) is not robust in strongly cluttered environments, since this proposal does not include the most recent observations [4, 7]. In this section, we suggest a new PD based on the state partition technique and a parallel bank of EKFs (SP-PEKF) [8]. We briefly review the SP-PEKF method and then show how it can be incorporated within the PF framework.

SP-PEKF is a technique for calculating the statistics of the states of a nonlinear system. The rationale is to generate a set of samples x_i , $i \in [1, N]$, associated with each state x according to the given initial distribution $\mathcal{N}(x(0), R(0))$. These samples are partitioned as $x(t) \triangleq x_n(t) + x_r(t)$, where $x_n(t)$ and $x_r(t)$ denote the nominal and residual parts of the true state, respectively. After partitioning, the samples are propagated through a parallel filter bank and the estimated states are generated as a weighted sum of the filtered samples. Initially, samples $x_i(0)$ are generated according to $\hat{x}(0) = x_{n_i}(0) + \hat{x}_{r_i}(0)$ and $R(0) = R_{n_i}(0) + R_{r_i}(0)$, where $x_{n_i}(0) = \hat{x}(0)$, $\hat{x}_{r_i}(0) = 0$, $R_{n_i}(0) = R(0)$, and $R_{r_i}(0) = 0$. Here, R_{n_i} and R_{r_i} represent the covariance of the nominal state and the residual state, respectively.

The nominal state is updated according to $x_{n_i}(t) = f(x_{n_i}(t-1)) + v_{n_i}(t-1)$, where $v_{n_i}(t)$ is distributed identically to $v(t)$ but is a different realization. The system state trajectory is then linearized about $x_{n_i}(t)$ according to

$$x(t) \approx f(x_{n_i}(t-1)) + F(x_{n_i}(t-1))x_{r_i}(t-1) + v(t-1), \quad (5)$$

where $F(\cdot)$ denotes the Jacobian. From (5), we have that

$$x_{n_i}(t) + x_{r_i}(t) \approx f(x_{n_i}(t-1)) + v_{n_i}(t-1) - v_{n_i}(t-1) + F(x_{n_i}(t-1))x_{r_i}(t-1) + v(t-1),$$

which can be simplified to obtain

$$x_{r_i}(t) \approx F(x_{n_i}(t-1))x_{r_i}(t-1) + v(t-1) - v_{n_i}(t-1). \quad (6)$$

Manipulating the output equation in a similar way yields

$$y(t) \approx h(x_{n_i}(t)) + H(x_{n_i}(t))x_{r_i}(t) + n(t), \quad (7)$$

where $H(\cdot)$ is the Jacobian. Note that the dynamic model given by (6) and (7) approximates the nonlinear system (1) but is linear in the residual part of the state $x_{r_i}(t)$. Thus, a bank of EKFs can be applied to update $x_{r_i}(t)$ and the overall system state is estimated by the weighted sum

$$\hat{x}(t|t) = \sum_{i=1}^N \hat{x}_i(t|t) w_i(t), \quad (8)$$

where $\hat{x}_i(t|t) = x_{n_i}(t) + \hat{x}_{r_i}(t|t)$. The filter weights $w_i(t)$ in (8) are given by

$$w_i(t) = \frac{L_i(t|t)w_i(t-1)}{\sum_{i=1}^N L_i(t|t)w_i(t-1)}, \quad (9)$$

where

$$L_i(t|t) = |R_{y_i}(t|t-1)|^{-0.5} \exp\left[-\frac{1}{2} \|y(t) - \hat{y}_i(t|t-1)\|^2 R_{y_i}^{-1}(t|t-1)\right]. \quad (10)$$

Finally, the overall estimation error covariance R_{total} is

$$R_{total}(t|t) = \sum_{i=1}^N \{R_i(t|t) + [\hat{x}(t|t) - \hat{x}_i(t|t)] \times [\hat{x}(t|t) - \hat{x}_i(t|t)]^T\} w_i(t). \quad (11)$$

Our unique contribution in this paper is developing a new PF algorithm (PF-SP-PEKF) which uses the SP-PEKF state estimates for its PD $q(\cdot)$. At each iteration, particles are drawn from a normal distribution with mean (8) and covariance (11). These particles are then propagated as in the conventional PF according to (2)-(4) with a resampling step. In order to take maximum advantage of the state estimates derived from the particle filter, the final estimates from the combined approach are recursively fed back into the SP-PEKF filter. This update stage serves as a ‘‘correction’’ step for the SP-PEKF filter at each iteration. We typically choose the number of filter bank channels N in (8)-(11) much smaller than the number of particles N_s ; our experience is that taking $10 \leq N \leq 20$ is generally sufficient to obtain significantly improved performance relative to, e.g., a comparable condensation filter. The overall PF-SP-PEKF algorithm is illustrated in Fig. 1 and summarized below.

The PF-SP-PEKF algorithm:

- Sequential Importance Sampling (SIS) Step:
 - Sample from the proposal distribution according to (8) and (11):

$$x^{(i)}(t) \sim q(x^{(i)}(t)|x^{(i)}(0:t-1), y(1:t)) = \mathcal{N}(\hat{x}(t|t), R_{total}(t|t));$$
 - Evaluate and normalize the importance weights according to (3) and (4);
- Resampling Step: Generate a new set of particles $x^{i*}(t)$ from $x^{(i)}(t)$ by sampling N_s times the approximate distribution of $p(x(t)|y(1:t))$ so that

$$Pr(x^{i*}(t) = x^{(j)}(t)) = \tilde{\omega}^{(j)}(t);$$
- Output and Update Step: Estimate $x(t)$ according to $x(t) \approx \frac{1}{N_s} \sum_{i=1}^{N_s} x^{i*}(t)$ and update the proposal.

4. PF-SP-PEKF VIDEO TRACKING

As an illustrative example, we apply the PF-SP-PEKF algorithm proposed in Section 3 to track the head of a perambulating human in a real-time video sequence. We adopt a Langevin process [2, 7] to model the head as a moving ellipse centered at image coordinates (row, col) = (r, s) . The state dynamics are given by

$$\begin{bmatrix} r(t+1) \\ s(t+1) \\ \dot{r}(t+1) \\ \dot{s}(t+1) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & a_r & 0 \\ 0 & 0 & 0 & a_s \end{bmatrix} \begin{bmatrix} r(t) \\ s(t) \\ \dot{r}(t) \\ \dot{s}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ b_r \\ b_s \end{bmatrix} m(t),$$

where $x(t) = [r(t) \ s(t) \ \dot{r}(t) \ \dot{s}(t)]^T$ is the state vector, $a_r = \exp(\beta_r \Delta T)$, $a_s = \exp(\beta_s \Delta T)$, $b_r = \bar{v}_r \sqrt{1 - a_r^2}$, $b_s = \bar{v}_s \sqrt{1 - a_s^2}$, parameters β_r and β_s are rate constants, ΔT is the discretization time step, \bar{v}_r and \bar{v}_s are the steady-state root-mean-square velocity, and $m(t)$ is the process noise.

K equally spaced rays are drawn from the center of the ellipse, which serves as the origin of a local coordinate system (u, v) . The intersections of these rays with the ellipse boundary are taken as observations. In local coordinates, the intersections along the k^{th} ray are obtained according to

$$\begin{aligned} u_k &= \sqrt{a^2 b^2 / (b^2 + a^2 \tan^2 \theta_k)} \\ v_k &= \tan \theta_k \cdot \sqrt{a^2 b^2 / (b^2 + a^2 \tan^2 \theta_k)} \end{aligned} \quad (12)$$

by solving the ellipse equation $\frac{(u_k - m)^2}{a^2} + \frac{(v_k - n)^2}{b^2} = 1$ and the ray equation $v_k = u_k \tan \phi_k$, where a and b denote

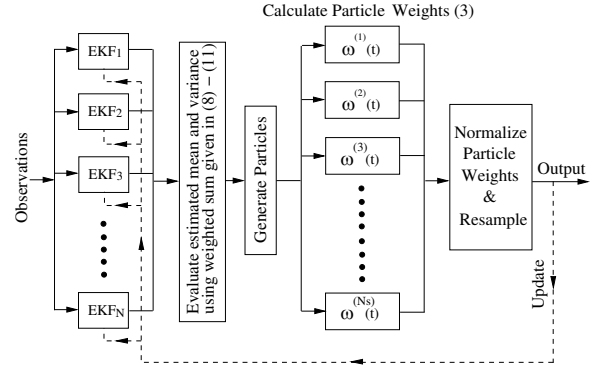


Fig. 1. Block diagram of the PF-SP-PEKF algorithm.

the major and minor axes of the ellipse, respectively. Let y represent the observation and convert the local coordinates (u_k, v_k) back to image coordinates. The observation equation is then given by

$$y(t) = [(u_k + r(t), v_k + s(t))] + n(t), \quad (13)$$

where $n(t)$ is the measurement noise. To reject clutter, we apply 1-D edge detection along each ray. The edges are labeled as $j = 1 \dots J_k$, where J_k is the total number of edges detected along the k^{th} ray. The likelihood of each ray is then $p_k(y_t|x_t) = N_m \sum_{j=1}^{J_k} \mathcal{N}((u_k, v_k), \sigma_{kj}^2)$, where N_m is a normalizing factor. The overall likelihood computed across all rays is given by $p(y(t)|x(t)) = \prod_{k=1}^K p_k(y(t)|x(t))$. This likelihood model is a simplified version of the one given in [7]. The innovation along each ray is $y_k(t) - \bar{y}(t|t-1) = \sum_j \pi_{kj}((u_k, v_k)_{t,j} - (u_k, v_k)_{t|t-1})$.

5. RESULTS AND DISCUSSION

Digital video data were acquired with a frame size of 480×720 pixels in a highly cluttered laboratory environment using a high-quality full motion video camera manufactured by Cohu Electronics. The *condensation* method and the PF-SP-PEKF algorithm were implemented for comparison. Tracking results for the two filters are shown in Fig. 2 and Fig. 3, respectively, where the estimated target centroid is indicated by a white cross. For the first 20 frames the motion was nearly rectilinear with minimal acceleration and both filters performed well. Beginning around frame 25, however, the motion became more complicated with nontrivial accelerations and increased background clutter around the head. While the PF-SP-PEKF filter maintained a consistent track lock throughout the video sequence, it may be seen in Fig. 2 that the condensation filter progressively diverged in frames 30 through 50, ultimately locking onto the clutter structure and losing the tracked object all together. These results, which are typical of those we

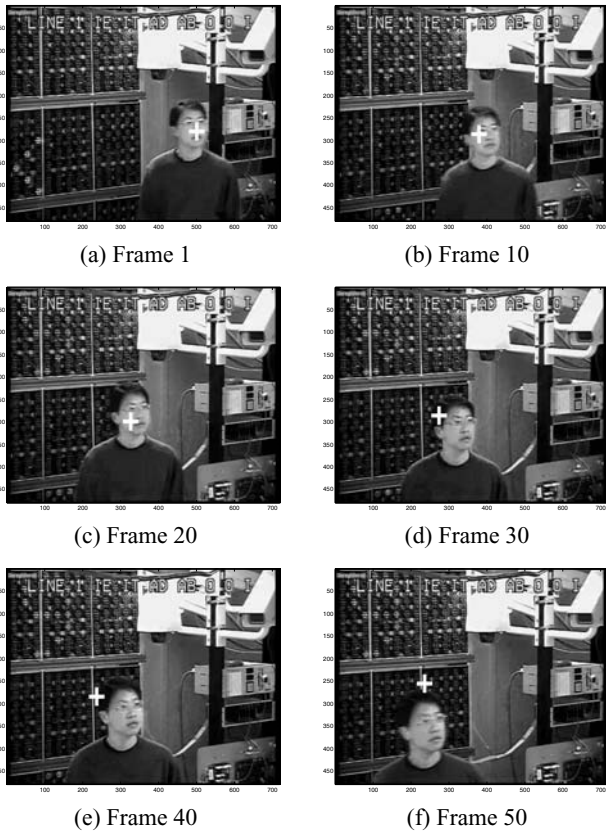


Fig. 2. Condensation method results. Divergence is observed in frame 30 with complete track loss by frame 50.

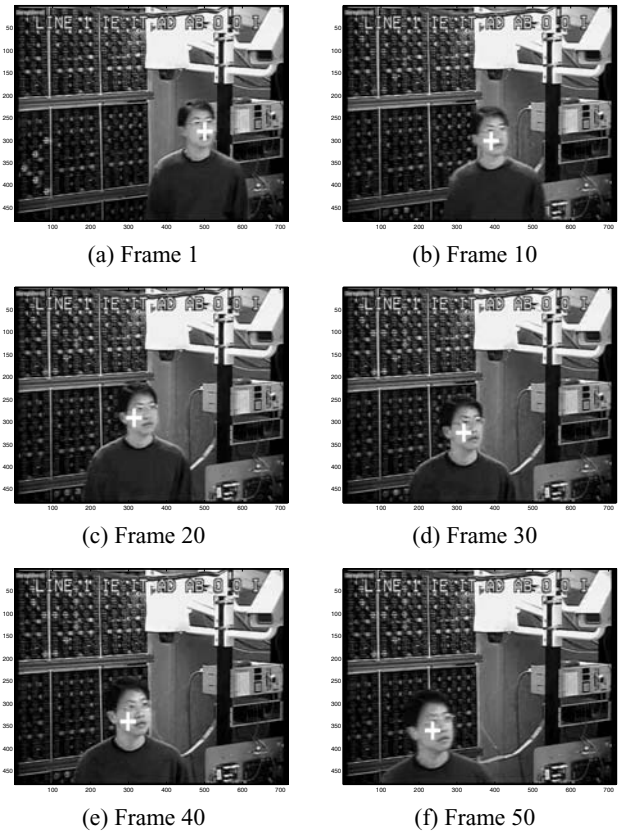


Fig. 3. PF-SP-PEKF tracking result: a consistent track lock is maintained throughout the entire video sequence.

have obtained, demonstrate the performance advantage that can be gained with the PF-SP-PEKF approach by explicitly considering the most recent observation when constructing the proposal distribution. Due to its improved accuracy and robustness, the PF-SP-PEKF is also capable of achieving a given level of tracking performance with fewer particles than its conventional PF counterparts, which constitutes a significant advantage in real-time implementations. Our future research is focused on combining the SP-PEKF approach with Markov Chain Monte Carlo methods such as the Metropolis-Hastings algorithm to address the well-known “sample impoverishment” problem that is typically seen in conventional particle filters when a small number of heavily weighted particles cause a loss of sample diversity.

6. REFERENCES

- [1] Y. Bar-Shalom and T. Fortmann, *Tracking and Data Association*, Academic Press, 1998.
- [2] Y. Chen, Y. Rui and T. Huang, “JPDAF Based HMM for Real-Time Contour Tracking”, *Proc. of IEEE CVPR*, pp.I-543 to 550, Dec., 2001.
- [3] M. Bruno and J. Moura, “Multiframe detector/tracker: optimal performance,” *IEEE Trans. Aerosp. Electron. Syst.*, vol. 37, pp. 925-946, July 2001.
- [4] M. Arulampalam, S. Maskell, N. Gordon and T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking,” *IEEE Trans. Signal Process.*, vol. 50, no.2, pp. 174-188, Feb., 2002.
- [5] M. Isard and A. Blake, “Contour tracking by stochastic propagation of conditional density,” *Proc. Eur. Conf. Computer Vision*, 1996.
- [6] T. Bando, T. Shibata, K. Doya and S. Ishii, “Switching particle filters for efficient real-time visual tracking”, *Proc. Int. Conf. Pattern Recognition*, pp.720-723, Aug., 2004.
- [7] Y. Rui and Y. Chen, “Better proposal distributions: object tracking using unscented particle filter,” *Proc. of IEEE CVPR*, pp. 786-793, Dec., 2001.
- [8] D. Lainiotis and P. Paparaskeva, “A new class of efficient adaptive nonlinear filters (ANLF),” *IEEE Trans. Signal Process.*, vol. 46, pp. 1730-1737, June, 1998.