Enhanced Particles With Pseudolikelihoods for Three-Dimensional Tracking

Huiying Chen and Youfu Li, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a new method to fuse sensing data of the most current observation into a 3-D visual tracker using pseudolikelihood functions with particle filtering techniques. With the proposed approach, the importance density function in particle filter can be modified to represent posterior states by particle crowds in a better way. Thus, it makes the tracking system more robust to noise and outliers. On the other hand, because the particle interpretation is performed in a much more efficient fashion, the number of particles used in tracking is greatly reduced, which improves the real-time performances of the system. Simulation and experimental results verified the effectiveness of the proposed method.**

***Index Terms*—Importance density, particle filtering, pseudolikelihood, 3-D tracking.**

I. Introduction

VISUAL tracking is to continuously estimate and update the position and orientation of a target. In a visual tracking context, much effort has been devoted to tracking problems in feature space, i.e., image features on the image plane have been adopted, analyzed, and tracked to achieve tracking tasks. This kind of tracking, without recovering the depth information, is considered as two-dimensional (2-D) tracking. In 2-D tracking, no depth information is recovered, which may lead to undesirable tracking performance. For instance, vision systems need the depth information to adjust their optical parameters to suitable values for tracking for better detection of the target object [1]. In silhouette tracing, depth information is needed for defining the size of the target or deciding a proper targeting area [2].

Three-dimensional (3-D) tracking, which reveals the targets' information in one more dimension than 2-D tracking, can provide more degrees of freedom in its application. For example, it can be directly used in dynamic 3-D estimation or, with the depth information recovered, it can help in collision avoidance in 3-D space [3].

From the viewpoint of depth information recovery, 3-D visual tracking can be performed with three main kinds of methods: 1) using two or multiple simultaneous views (e.g., stereopsis or multicamera network [4]–[6]); 2) using two or multiple nonsimultaneous views (e.g., a moving camera [7]); and 3) using a single view with an active setup [8], [9].

As 3-D visual tracking [10], [11] is one of the typical non-Gaussian and nonlinear state-space estimation problems [12], the popular way to tackle it is by using the particle filter (PF), also known as condensation or sequential Monte Carlo method [13]–[15], which is a powerful simulation-based method.

However, a general 3-D tracking problem with 6 DOF often requires thousands of particles [16], which can run foul of computational complexity and further interfere real-time performance for tracking agile motion. On the other hand, in generic particle techniques, the most common strategy is to choose the transition prior $P(x_k|x_{k-1})$ as the *importance density function* [13], [17], [18]. As a result, the proposed density function used for sampling is independent of the most current measurement, and thus, it is sensitive to outliers. In order to tackle the problem, various approaches have been taken using variational optimization or local linearization methods. Gordon *et al.* [13] presented a prior editing method, in which an *ad hoc* acceptance test was used to propose particles in regions of high likelihood. There were some other approaches [19], [20] that used the extended Kalman filter (EKF) [21] to calculate the approximation to the true posterior as the importance density. van der Merwe *et al.* [22] improved the method by employing more accurate approximations from the unscented Kalman filter (UKF) instead. Although the aforementioned approaches enhanced the robustness of PF algorithm, they did not provide the way to release the computational burden.

In this paper, we propose a new method to fuse sensing data of the most current observation into a 3-D visual tracker with PF techniques. Incorporating with the sensing data, the importance density function becomes well modified so that particle crowds can represent the posterior states in a much more efficient fashion. Consequently, the number of particles used in tracking is greatly reduced. We have proved that only about 100 particles are enough to perform a real-time 3-D tracking process with an active vision system. In our previously reported work [23], a data fusion method for active sensing was presented. This paper further explores the fusion factor and criterion for optimizing the importance density function. Some new results are also given.

II. Three-Dimensional Tracking With an Active Vision System Under Particle Frameworks

The 3-D tracking task for rigid body tracking here is performed with an active vision system [24], [25] using pattern

Manuscript received August 14, 2008; revised May 18, 2009. First published June 2, 2009; current version published July 24, 2009. This work was supported by the Research Grants Council of Hong Kong under Projects CityU117106 and CityU 117507.

H. Chen is with the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: velvet.chen@polyu.edu.hk).

Y. Li is with the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong (e-mail: meyfli@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2009.2024099

projection, which is similar to a passive stereo vision system with one of the cameras replaced by a projector. Using a color-encoded structured light pattern [26], the active vision system can yield good results in 3-D visual sensing with a single view. Compared with other 3-D sensing systems, such as stereo vision systems, active vision systems have the merit that they can find feature points (points on the projected pattern) easily, even when the target object has a smooth and continuous surface. With an active vision system, the tracking task is formulated in the framework of particle filtering as follows.

A. State Transition

Suppose that the target object is rigid. We define the state vector of the object as $(\mathbf{T}, \Omega)^{\mathrm{T}}$, with $\mathbf{T} = (X, Y, Z)$ being the position of the object frame in a fixed world frame and $\Omega = (\phi, \theta, \psi)$ being the orientation of the object frame in the world frame.

At the kth state, the next object position and orientation can be computed using velocities with an independent first-order Gaussian random walk model (see [23] for details)

$$\Omega_{k+1} = \Omega_k + \Delta t \cdot \dot{\Omega}_k \tag{1}$$

$$T_{k+1} = T_k + \Delta t \cdot R(\Omega_k) \cdot \dot{\Omega}_k \tag{2}$$

where Δt is a short sampling time, and R is a rotation matrix.

B. Observation

For the camera, the relationship between 3-D coordinates of an object point from the view of the camera $\mathbf{X}^{\mathrm{cam}} = (\, X^{\mathrm{cam}} \quad Y^{\mathrm{cam}} \quad Z^{\mathrm{cam}} \quad 1 \,)^{\mathrm{T}}$ and its projection on the image $\mathbf{m}^{\mathrm{cam}} = (\, \lambda u^{\mathrm{cam}} \quad \lambda \nu^{\mathrm{cam}} \quad \lambda \,)^{\mathrm{T}}$ is given by

$$\mathbf{m}^{\mathrm{cam}} = \mathbf{M}^{\mathrm{cam}} \mathbf{X}^{\mathrm{cam}} \tag{3}$$

where $\mathbf{M}^{\mathrm{cam}}$ is a 3×4 perspective matrix of the camera, λ is a scale factor, and u^{cam} and ν^{cam} are the image feature coordinates in the image plane. Equation (3) expresses the direct measurement by the camera. We term this process as *passive sensing*. Such data are a kind of midway results in the active visual system.

The observation of the 3-D target location by active sensing can be determined by (see [23] for detailed derivation)

$$\mathbf{X}^{\mathrm{cam}} = \mathbf{Q}^{-1} \mathbf{m}_{+}^{\mathrm{cam}} \tag{4}$$

where $\mathbf{Q}$ is a 4×4 projection matrix with camera and projector parameters, and $\mathbf{m}_{+}^{\mathrm{cam}} = (\, \lambda u^{\mathrm{cam}} \quad \lambda \nu^{\mathrm{cam}} \quad \lambda \quad 0 \,)^{\mathrm{T}}$.

C. Prediction

At the kth state, let the unknown state vector be x_k. Assume that the probability function of the prior state vector is represented by $P(x_k|x_{k-1})$, which can be derived from (1) and (2). Then, assume that we have k independent observations of the state vector $\mathbf{y} = (y_1, y_2, \ldots, y_k)$ drawn from a normal population with unknown mean and known variance. According to Bayes' rule, the posterior probability function of the state vector can be calculated using the following:

$$P(x_k \,|\, \mathbf{y}) = \frac{P(x_k \,|\, x_{k-1}) P(\mathbf{y} \,|\, x_k)}{P(\mathbf{y})}. \tag{5}$$

The value of $P(\mathbf{y})$, which is a reciprocal of the normalizing constant for the probability density function in (5), can be calculated as $P(\mathbf{y}) = \int P(x_k) P(\mathbf{y} \,|\, x_k)\, \mathrm{d}x_k$. Because $P(\mathbf{y})$ is a constant, (5) can be written as

$$P(x_k \,|\, \mathbf{y}) \propto P(x_k \,|\, x_{k-1}) P(\mathbf{y} \,|\, x_k). \tag{6}$$

The weight of particle i can be defined from the likelihood function as

$$w_k^i \propto w_{k-1}^i \, \frac{P(x_k \,|\, x_{k-1}) P(\mathbf{y} \,|\, x_k)}{q(x_k \,|\, x_{k-1}, \mathbf{y})} \tag{7}$$

where $q(x_k \,|\, x_{k-1}, \mathbf{y})$ is the importance density function.

The particle weights can be normalized as $\tilde{w}_k^i$

$$\tilde{w}_k^i = \frac{w_k^i}{\sum_{i=1}^{N} w_k^i} \tag{8}$$

where N is the total number of particles. Consequently, the approximation of the posterior density is expressed as

$$P(x_k \,|\, \mathbf{y}) \approx \sum_{i=1}^{N} \tilde{w}_k^i \delta \left(x_k - x_k^i \right) \tag{9}$$

with δ being a Dirac's delta function [14].

III. Data Fusion for Importance Density Optimization

A. Data Fusion Algorithm and the Pseudolikelihood

In the particle framework, the posterior density is approximated by a weighted sum of N particles that are drawn from an *importance density*. Thus, the choice of the importance density directly influences the property of sampling, and it is a crucial factor. If the importance density is too small for some part of the support state of the posterior, one will need very large sample size to approximate the state properly. On the contrary, if the importance density is well designed, only a small number of particles will be needed to draw a good representation of the state. In generic particle techniques, the transition prior $P(x_k \,|\, x_{k-1})$ has often been employed as the importance density. This is a reasonable choice because, first, the transition prior $P(x_k \,|\, x_{k-1})$ is handy and easy to be calculated that it will not increase the computational complexity of the system. Second, according to continuity, the distribution of state x_k may not differ much from its prior state x_{k-1} in short-time propagation; thus, $P(x_k \,|\, x_{k-1})$ can represent the posterior state intuitively. However, because $P(x_k \,|\, x_{k-1})$ is independent of the most current observation information, it is sensitive to outliers. For instance, when the new measurements appear in the tail of the prior or if the likelihood is too peaked in comparison to the prior [13], $P(x_k \,|\, x_{k-1})$ will fail in interpreting the posterior

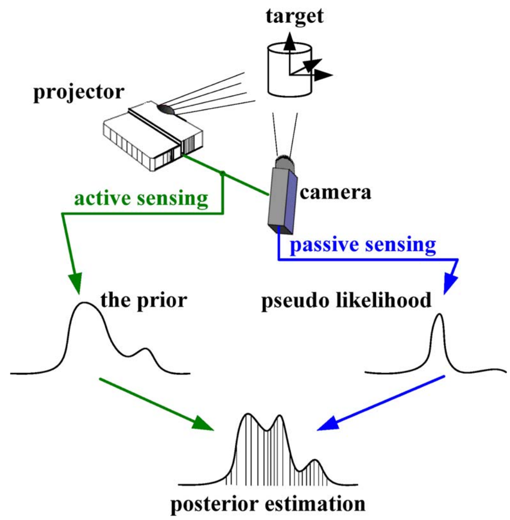


Fig. 1. Data fusion with updated passive sensing data.

properly. Here, we propose a method to fuse passive sensing data into the importance density function for optimizing particle sampling to reduce the number of required particles. We use the passive sensing data rather than the active sensing data for reference because they are handy to be used and easier to be modeled.

In order to obtain better expression of the posterior state, importance density function should be moved toward the region of high likelihood. Notwithstanding, because the support valuables are different, likelihood functions cannot be used directly to modify the importance density. To surmount this problem, a *pseudolikelihood* function is first generated with the latest passive sensing data. Then, the pseudolikelihood is projected to the importance density space, and the importance density is modified by fusing the sensing data in it. The pseudolikelihood function is generated with the most current observation of certain reference feature points through passive sensing. It is a subset of the likelihood function, and it can represent the likelihood function to certain extent. The advantage of the pseudolikelihood function is that it can be projected to the importance density space easily by using the inverse procedure of passive sensing observation model (a monocular camera model). The basic idea of the proposed approach is shown in Fig. 1. Suppose that, in a set of reference points, the observation of passive sensing (of a monocular camera) can be expressed as a function of the current state with noise

$$y_{k_j}^R = g_j(x_k, \zeta), \qquad j = 1, \ldots, N_R \tag{10}$$

where ζ is the noise, and N_R is the number of reference points.

Equation (10) can be looked on as a pseudolikelihood function

$$y_k^R \sim \hat{L}_y = P\left(\mathbf{y} \mid x_k^R\right). \tag{11}$$

On the contrary, the current state can be estimated using the inverse function of (10) as

$$x_k = g^{-1}\left(x_{k-1}, y_k^R, \zeta\right) \tag{12}$$

TABLE I
PASSIVE DATA FUSION ALGORITHM IN PF

Assume that at the previous state *k-1*, we have the particle crowd $\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^N$, then proceed as following at time k
1. Sampling: simulate $x_k^i \sim P(x_k \mid x_{k-1})$
2. Calculate weights: compute the weights according to likelihood function and conduct normalization
3. Pseudo likelihood computation: calculate $\hat{L}_x = P(x_k \mid x_{k-1}, y_k^R, \zeta)$
4. Data fusion: simulate x_k^i, draw αN samples from the prior $P(x_k \mid x_{k-1})$ and $(1-\alpha)N$ samples from the pseudo likelihood projection $P(x_k \mid x_{k-1}, y_k^R, \zeta)$, where α is a data fusion factor, $0 \le \alpha \le 1$
5. Update weights: compute the weights according to the new likelihood function and conduct normalization
6. Resampling

which is, in fact, a projection of the pseudolikelihood to the importance density (x_k) space

$$\hat{L}_x = P\left(x_k \mid x_{k-1}, y_k^R, \zeta\right). \tag{13}$$

Then, (13) can be used to achieve the algorithm for data fusion, as shown in Table I.

B. Importance Density Optimization

Degeneracy is a common phenomenon with PFs (see [19]). As a result of degeneracy, all but one particle will have negligible weight after a few state transitions. Degeneracy implies the wastage of computational resources that much effort is devoted to updating particles whose contribution to the approximation to posterior states is almost zero.

We can here adopt the *effective sampling size* N_k^{eff} [27], which is a suitable measure of degeneracy, as a criterion to guide the optimization process for data fusion.

As N_k^{eff} cannot be evaluated exactly [28], an estimate $\hat{N}_k^{\text{eff}}$ of N_k^{eff} can be calculated by

$$\hat{N}_k^{\text{eff}} = \frac{1}{\sum_{i=1}^N \left(\tilde{w}_k^i\right)^2} \tag{14}$$

where $\tilde{w}_k^i$ is the normalized weight indicated in (8).

A large N_k^{eff} implies that the likelihood is located closely to the prior $P(x_k \mid x_{k-1})$, so that the particle crowd from the prior x_k^i can be relied on better. Thus, the percentage of effective sampling $\hat{N}_k^{\text{eff}}/N$ can be used to define the data fusion factor α (see Table I) for importance density optimization by choosing

$$\alpha = \rho \frac{\hat{N}_k^{\text{eff}}}{N} \tag{15}$$

where ρ is a positive scale factor $\rho \geq 1$.

According to our previous study [29], when the configuration of the vision system (i.e., relative location to the object, optical and physical parameters of the camera and the projector) is not well designed, the percentage of effective sampling $\hat{N}_k^{\text{eff}}/N$ can be very small, which even may drop down to 5% sometimes.

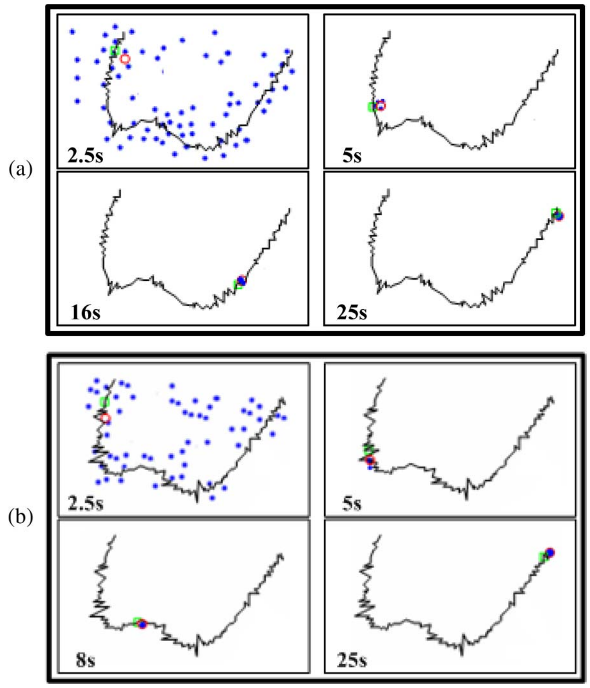


Fig. 2. Active visual tracking. (a) Without passive data fusion. (b) With passive data fusion. (Red circles) True object locations. (Green boxes) Estimations from active sensing. (Blue points) Estimations from particle filtering.

Since the effective sampling size is corresponding to the tracking error to some extent [29], a small effective sampling size may cause a large tracking error by particle estimation. In this case, the transition prior $P(x_k \mid x_{k-1})$ is not suitable to guide the importance density on its own. The pseudolikelihood, which represents the most current sensing data, will help the PF obtain better sampling and reduce tracking error.

The value of ρ can be determined empirically or it can simply be chosen as one.

Then, the new particle crowd after data fusion is

$$x_k^{*i} = \alpha\, x_k^i + (1-\alpha)\hat{x}_k^i \tag{16}$$

where x_k^i is drawn from the prior, and $\hat{x}_k^i$ is drawn from the pseudolikelihood projection.

IV. Simulation and Experimental Results

A. Simulation Results

1) Effectiveness of Data Fusion: We first studied and verified the effectiveness of passive data fusion. Fig. 2(a) and (b) shows the simulations of the proposed PF for active tracking with and without the fusion of passive information. In Fig. 2(a), lacking of the fusion of passive sensing information, the particle tracker used about 16 s to converge to the true object location. Fig. 2(b) shows that, under the same condition, the fusion of passive sensing data helps the PF to achieve faster convergence. In this case, the active tracker took about 8 s to converge.

2) Tracking Accuracy: Fig. 3 shows the estimation errors for 3-D location tracking by different methods. The generic PF (GPF), which employed 800 particles, performed the best, while the EKF performed the worst because of its disadvantage in dealing with multimodality. The proposed PF with data fusion, even only with 100 particles, achieved performance approximately as good as the GPF.

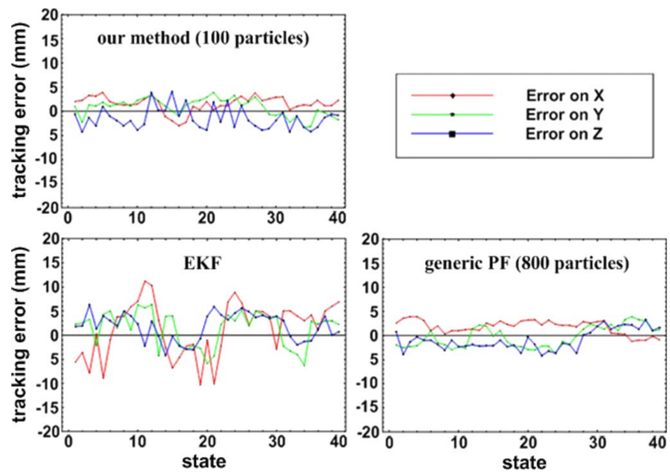


Fig. 3. Tracking accuracy comparison.

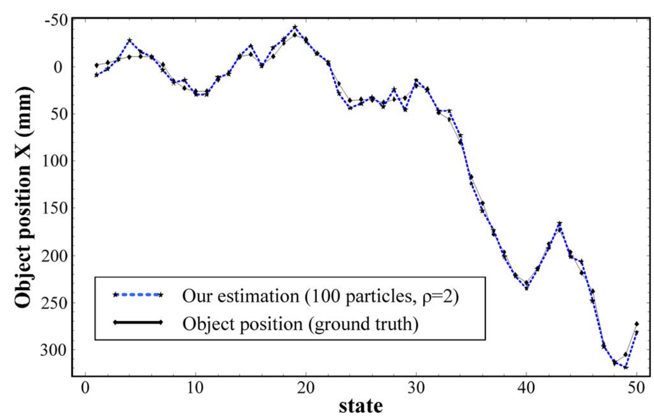


Fig. 4. Visual tracking using a PF with 100 particles and $\rho = 2$.

3) Tracking Speed: With a well-expressed importance density, the proposed PF can achieve better real-time performance with expedition. Simulation results demonstrate the superiority of the proposed method in comparison with EKF and GPF. Because the EKF does not involve calculations of sampling, it can achieve faster real-time performance with an average of 0.0146 s for each state. The algorithm running times for GPF with 200, 400, and 800 particles are 0.0271, 0.0522, and 0.0998 s, respectively. With only 100 particles, the proposed PF excels the GPFs in running time, and it only spent 0.0150 s in average.

4) Choice of the Data Fusion Factor: We then compared the tracking errors with different data fusion factors. A PF with 100 particles was adopted for the tracking shown in Fig. 4. In this simulation, the average percentage of effective sampling $\hat{N}_k^{\mathrm{eff}}/N$ (before resampling) is about 36%.

As shown in Fig. 5, the data fusion factor affects the tracking performance. According to our simulation study, even with the same vision system configuration, the effective sampling size may change during the tracking process, and it is not necessary to adopt the same ρ when performing visual tracking. Table II indicates some possible choices of ρ (or α) according to our simulation results.

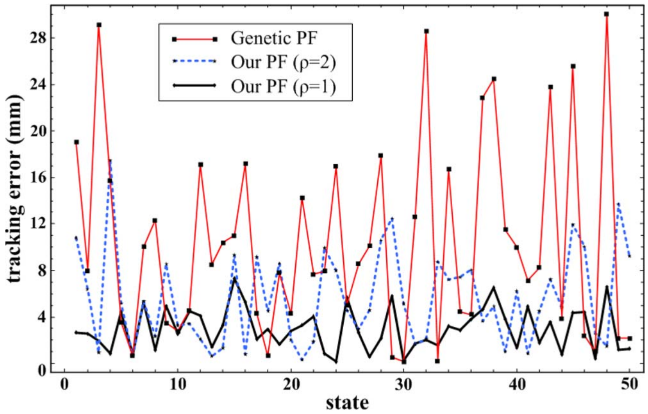
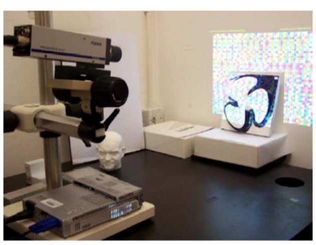


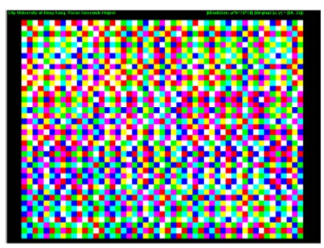
Fig. 5. Tracking error comparison.

TABLE II
SOME CHOICES OF ρ OR α

$\hat{N}_k^{eff} / N$ **value**	<0.05	0.1~0.2	0.2~0.3	>0.3
ρ or α **value**	$\alpha \approx 0.1$	$\rho \approx 3$	$\rho \approx 2$	$\rho \approx 1$



(a)



(b)

Fig. 6. Active vision system using color-encoded structured light.

B. Experiment

The proposed tracking method was implemented with an active vision system which consists of a PULNIX TMC-9700 CCD camera and a PLUS V131 DLP projector [as shown in Fig. 6(a)]. When the system is used in a visual tracking task, the projector projects a color-encoded structured light (see Fig. 6(b), [9]) onto the surface of the target object. Via triangulation, the system returns a time sequence of 3-D object positions and orientations. This provides the *observation* (given in Section II-B) for the tracking formulation.

We used a concave object as the target, which was moved arbitrarily by hand in 3-D space to give motions with 3-DOF translational and 2-DOF rotational. The tracker (formulated in Sections II and III) was used to estimate the target's 5-DOF position. Here, a PF with 100 particles and $\rho = 1$ was employed. Since the object was moved randomly and the tracking was performed in real time, quantitative results on tracking accuracy were hard to obtain due to lack of the ground truth. We thus reprojected the estimated object positions and orientations onto the camera image (the red circles shown in Fig. 7) for qualitative evaluation. Some examples of snapshots in the tracking are shown in Fig. 7. With a sampling rate of about 12 ft/s, correct and reliable tracking was observed in the implementation. In Fig. 7, the tracking errors were mainly caused by the sensing itself, rather than the tracker. For example, in the frame shown in the bottom right, relatively larger tracking error is observed. This is because the target happened to move to a position where the structured light pattern could not be detected clearly. When reprojecting the estimation from active sensing of the feature points onto the image, the comparison in Fig. 8 shows that good active sensing quality actually corresponds to good tracking results. In some situations, e.g., when the relative depth to the projector was not large enough or when the light pattern started to blur on the target surface, the sensing for target tracking is also influenced.

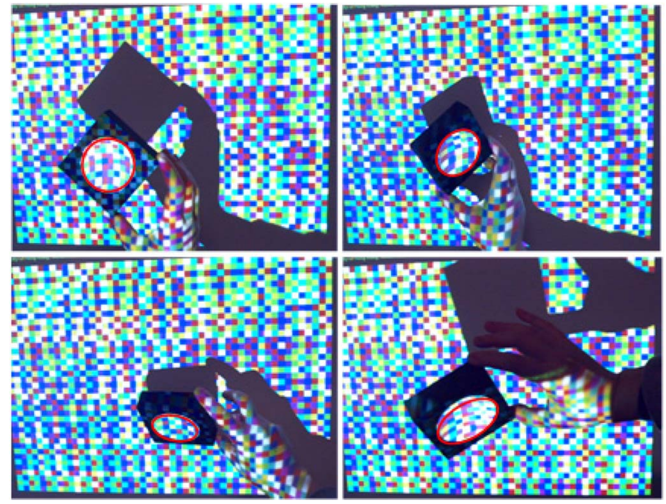
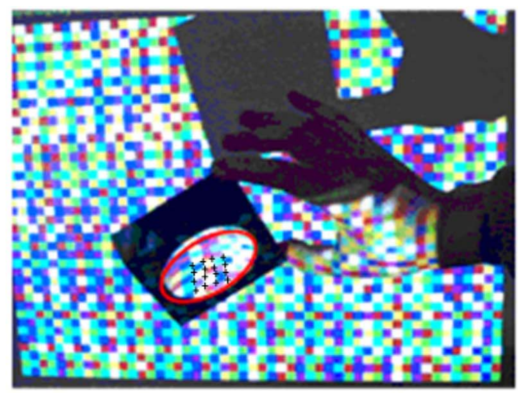
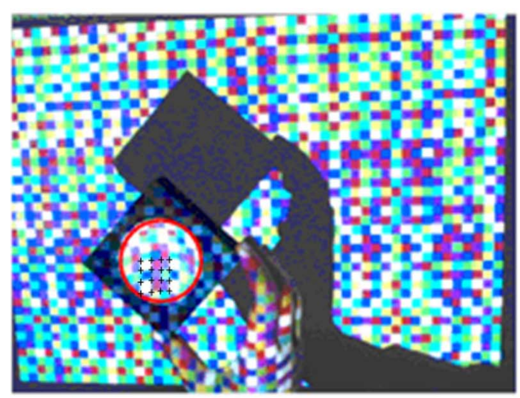


Fig. 7. Tracking a concave object with the proposed method.



(a)



(b)

Fig. 8. Visual tracking error and active sensing error.

The aforementioned example of tracking in 3-D (with 5-DOF motions) is useful in many practical applications such as in motion capturing of actions where the orientations and 3-D positions of an actor's arm/leg need to be tracked, for filmmaking in the entertainment industry.

V. Conclusion

In this paper, a new PF approach to fuse passive sensing data into an active 3-D visual tracker has been presented. With the enhanced PF, the importance density in particles can be modified to interpret the posterior state in an efficient way, using pseudolikelihoods. Therefore, the number of particles used in tracking is greatly reduced. Simulation and experimental results have shown that the proposed method significantly improved the tracking accuracy, as well as tracking speed in comparison with other methods. Possible applications of the proposed method can be found in motion capturing for filmmaking in entertainment industry.

The future work will include studying the data fusion factor and its effect on the tracking performance, as well as the implementation of the proposed method with a reconfiguration active vision system.

References

[1] K. C. Ng, H. Ishiguro, M. Trivedic, and T. Sogo, "An integrated surveillance system—Human tracking and view synthesis using multiple omni-directional vision sensors," *Image Vis. Comput.*, vol. 22, no. 7, pp. 551–561, Jul. 2004.

[2] Z. Zhu and Q. Ji, "3D face pose tracking from an uncalibrated monocular camera," in *Proc. 17th Int. Conf. Pattern Recog.*, 2004, vol. 4, pp. 400–403.

[3] A. Eidehall and F. Gustafsson, "Combined road prediction and target tracking in collision avoidance," in *Proc. IEEE Intell. Veh. Symp.*, Parma, Italy, 2004, pp. 619–624.

[4] R. Hartley and P. Sturm, "Triangulation," *Comput. Vis. Image Underst.*, vol. 68, no. 2, pp. 146–157, Nov. 1997.

[5] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2000.

[6] R. Hartley, R. Gupta, and T. Chang, "Stereo from uncalibrated cameras," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 1992, pp. 761–764.

[7] H. Y. Chen and Y. F. Li, "Metric measurements of 3D object features by a moving camera," in *Proc. IEEE Int. Conf. IMTC*. Como, Italy, May 2004, pp. 498–503.

[8] T. Kanade, "Recovery of the three-dimensional shape of an object from a single view," *Artif. Intell.*, vol. 17, no. 1–3, pp. 409–460, 1981.

[9] B. Zhang, Y. F. Li, and Y. Wu, "Self-recalibration of a structured light system via plane-based homography," *Pattern Recognit.*, vol. 40, no. 4, pp. 1368–1377, Apr. 2007.

[10] H. Y. Chen and Y. F. Li, "Enhanced 3D tracking using nonsingular constraints," *Opt. Eng.*, vol. 45, no. 10, pp. 107 202-1–107 202-12, Oct. 2006.

[11] S. Rovetta and R. Zunino, "A multiprocessor-oriented visual tracking system," *IEEE Trans. Ind. Electron.*, vol. 46, no. 4, pp. 842–850, Aug. 1999.

[12] J. Stephant, A. Charara, and D. Meizel, "Virtual sensor: Application to vehicle sideslip angle and transversal forces," *IEEE Trans. Ind. Electron.*, vol. 51, no. 2, pp. 278–289, Feb. 2004.

[13] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *Proc. Inst. Elect. Eng.—Radar Signal Process.*, vol. 140, no. 2, pp. 107–113, Apr. 1993.

[14] A. Doucet, N. D. Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. Berlin, Germany: Springer-Verlag, Jan. 2001.

[15] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis. (IJCV)*, vol. 29, no. 1, pp. 2–28, 1998.

[16] D. Fox, "KLD-sampling: Adaptive particle filters," in *Proc. Conf. Neural Inform. Process. Syst.*, 2001, pp. 713–720.

[17] D. Avitzour, "A stochastic simulation Bayesian approach to multitarget tracking," *Proc. Inst. Elect. Eng.—Radar, Sonar Navig.*, vol. 142, no. 2, pp. 41–44, Apr. 1995.

[18] E. R. Beadle and P. M. Djuric, "A fast-weighted Bayesian bootstrap filter for nonlinear model state estimation," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 33, no. 1, pp. 338–343, Jan. 1997.

[19] A. Doucet, "On sequential simulation-based methods for Bayesian filtering," Dept. Eng., Cambridge Univ., Cambridge, U.K., Tech. Rep. CUED/F-INFENG/TR 310, 1998.

[20] M. K. Pitt and N. Shephard, "Filtering via simulation: Auxiliary particle filters," *J. Amer. Stat. Assoc.*, vol. 94, no. 446, pp. 590–599, 1999.

[21] B. Terzic and M. Jadric, "Design and implementation of the extended Kalman filter for the speed and rotor position estimation of brushless DC motor," *IEEE Trans. Ind. Electron.*, vol. 48, no. 6, pp. 1065–1073, Dec. 2001.

[22] R. van der Merwe, A. Doucet, N. de Freitas, and E. Wan, "The unscented particle filter," Dept. Eng., Cambridge Univ., Cambridge, U.K., Tech. Rep. CUED/F-INFENG/TR 380, 2000.

[23] H. Y. Chen and Y. F. Li, "Particles coupled with data fusion for 3D tracking," in *Proc. IEEE ICIT*, Chengdu, China, Apr. 2008, pp. 1–6.

[24] P. Besl, "Active, optical range imaging sensors," *Mach. Vis. Appl.*, vol. 1, no. 2, pp. 127–152, Apr. 1988.

[25] S. Segvic and S. Ribaric, "Determining the absolute orientation in a corridor using projective geometry and active vision," *IEEE Trans. Ind. Electron.*, vol. 48, no. 3, pp. 696–710, Jun. 2001.

[26] P. M. Griffin, L. S. Narasimhan, and S. R. Yee, "Generation of uniquely encoded light patterns for range data acquisition," *Pattern Recognit.*, vol. 25, no. 6, pp. 609–616, Jun. 1992.

[27] J. S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamical systems," *J. Amer. Stat. Assoc.*, vol. 93, pp. 1032–1044, 1998.

[28] M. S. 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.

[29] H. Y. Chen and Y. F. Li, "Dynamic view planning by effective particles for three-dimensional tracking," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 1, pp. 242–253, Feb. 2009.

Huiying Chen received the B.S. degree in mechatronics engineering from South China University of Technology, Guangzhou, China, the M.S. degree in precision machinery engineering from The University of Tokyo, Tokyo, Japan, and the Ph.D. degree from the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong.

She is currently with the Department of Industrial and Systems Engineering, The Hong Kong Polytechnic University, Hong Kong. Her research interests include robot vision, visual tracking, and dynamic view planning.

Youfu Li (M'91–SM'01) received the Ph.D. degree from the Robotics Research Group, Department of Engineering Science, University of Oxford, Oxford, U.K., in 1993.

From 1993 to 1995, he was a member of the Postdoctoral Research Staff in the Department of Computer Sciences, University of Wales, Aberystwyth, U.K. He is currently an Associate Professor in the Department of Manufacturing Engineering and Engineering Management, City University of Hong Kong, Hong Kong. His research interests include robot sensing, robot vision, 3-D vision, and visual tracking.

Dr. Li is an Associate Editor of the IEEE Transactions on Automation Science and Engineering.