Dynamic View Planning by Effective Particles for Three-Dimensional Tracking

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

***Abstract*—In this paper, we propose a new approach to dynamically manage the viewpoint of a vision system for optimal 3-D tracking using particle techniques. We adopt the effective sample size in the proposed particle filter as a criterion for evaluating tracking performance and employ it to guide the view-planning process for finding the best viewpoint configuration. In our approach, the vision system is designed and configured to achieve the largest number of effective particles, which minimizes tracking error by revealing the system to a better swarm of importance samples and interpreting posterior states in a better way. Superiorities of our method are shown by comparison with the resampling particle filter and other view-planning methods.**

***Index Terms*—Best viewpoint configuration, dynamic view planning, effective sample size, particle filter, 3-D tracking.**

I. Introduction

A. Background

THREE-DIMENSIONAL tracking deals with continuous 3-D state estimation and update of a moving object [1]. The task of 3-D tracking is of paramount importance for many applications and has been considered from widely different perspectives of various theoretical backgrounds and interests. As one of the state-space estimation problems, 3-D tracking can be modeled with the aid of parametric models. However, due to varying degrees of uncertainty inherent in system modeling and complexity of system noise, visual system is often subject to elements of non-Gaussianity, nonlinearity, and high dimensionality, which unfortunately, usually precludes analytic solutions. It is a strong belief that the issue of state measurement ultimately remains best handled within the framework of statistical inference. Instead of using linearization techniques, the estimation problem is directly solved with Bayesian methodology [2], [3]. However, the Bayesian paradigm involves calculation of high-order integrals of the time state estimation. Thus, in the last few decades, many approximation filtering schemes, which are well-known as methods of particle filtering (PF), also known as condensation or sequential Monte Carlo methods [4]–[7], have been developed to seek a simulation-based way to surmount the problems.

In this paper, an improved particle filter, which always has the largest effective sampling size, is employed to fulfill 3-D tracking task. Such improvement is attained with a reconfigurable vision system, where the viewpoint can actively be controlled in 3-D tracking. We name this viewpoint control process as dynamic view planning.

The objective of view planning is to find a suitably short view plan satisfying the specified goals and achieve this within an acceptable computation time for a given imaging environment and a target object [8], [9]. In a visual tracking context, view planning dynamically involves determining a time sequence of the best viewpoint configurations (or camera configurations) which optimize certain tracking indices [10]. Dynamic view planning endows the observer with the ability of actively placing the camera at different viewpoints for compensating the limitation in sensing scope as well as achieving better tracking performance. The main thrust of this paper is focused on how to accomplish dynamic view planning for 3-D tracking.

B. Related Work

In view-planning literature, much effort has been devoted into the research on static view planning for object recognition, scene reconstruction, inspection, feature detection, robot navigation, etc. [8], [9], [11], [12], whereas research that has been conducted on dynamic view planning with a self-configuring sensor space is relatively less. Furthermore, in the majority cases when dynamic view planning for tracking has been studied, the view-planning strategies employed were sensor driven, that is to say, view-planning algorithms were designed for specific sensor settings (such as sensor network setting) and configurations, which may not be easy to apply to other systems. Gupta and Das [12] developed a simple algorithm that detected and tracked a moving target, with Berkeley motes [13], a kind of radio sensor network. Besides, Schiff and Goldberg [14] introduced an estimation method based on PF with a binary motion sensor network to automatically track and capture photos of an intruder. These are two examples of sensor-driven strategies.

In nonsensor-driven strategies, research on how to explore the property of Kalman filter (or extended Kalman filter) to set up criteria for optimization in dynamic view planning has been studied a lot [15]. Denzler *et al.* [16] presented an approach

Manuscript received October 1, 2007; revised January 30, 2008 and May 23, 2008. This work was supported by the Research Grants Council of Hong Kong under Project CityU117605. This paper was recommended by Associate Editor Q. Ji.

H. Chen was with the City University of Hong Kong, Kowloon, Hong Kong. She is now with the Department of Industrial and Systems Engineering, Hong Kong Polytechnic University, Kowloon, 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, Kowloon, 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/TSMCB.2008.2005113

that allowed the optimal selection of the focal lengths of two cameras during active 3-D object tracking. The selection was based on the uncertainty in the 3-D estimation with Kalman filtering techniques. Tordoff and Murray [17] described a practical method of zoom control for object tracking by a camera with a variable focal-length lens, with the aid of the tuning of a constant velocity Kalman filter. Clemente *et al.* [18] presented a method for simultaneous localization and mapping with a monocular camera using the extended Kalman filter. Martinez and Bullo [19] proposed a method of motion coordination for range sensors in a tracking application. They investigated the determinant of Fisher information matrix [20] to guarantee the increase of sensitivity with respect to sensor measurements and implemented the algorithm within an extended Kalman filtering framework. However, the limitation of the above methods may lie in the point that Kalman filter, even the extended Kalman filter, cannot handle problems in a system with non-Gaussian models. Therefore, some other researchers have pursued view-planning methods within the particle framework to accurately model the underlying dynamics. Spletzer and Taylor [21] presented an approach to the tracking problem of actively controlling the configuration of a team of mobile agents equipped with cameras to optimize the quality of estimates derived from their measurements. Their idea was inspired by work on PF, and they defined a quality function which corresponded to the expected tracking error in their estimation. This method is in fact an error-orientated view-planning method. Nevertheless, their approach involved direct backward calculations from the measurement space to the parameter space. As it was not a one to one projection, unique solutions might not be obtained in those calculations. Moreover, random search steps in their algorithm may harm the real-time property of tracking.

C. Our Approach

In this paper, we propose a new approach to dynamic view planning for optimal 3-D tracking with particle techniques. We adopt the effective sample size as a criterion for evaluating tracking performance and employ it to guide the view-planning process for finding the best viewpoint configuration. In our approach, the vision system is designed and configured actively to achieve the largest effective sample size for particle sampling. On one hand, this technique minimizes tracking error by revealing the system to a better swarm of importance samples and interpreting the posterior state in a better way. On the other hand, it prevents particles' degeneracy, reduces total number of particles, and eliminates the necessity of resampling procedure as well, thus reinforces real-time properties and increases tracking speed. Simulation results have proved that the maximum effective sample size actually corresponds to the minimum tracking error. Superiorities of our method are also shown by comparison with the generic resampling PF and other view-planning methods.

In Section II, we present the task of 3-D tracking using particle techniques. Dynamic view-planning constraints and strategies are addressed in Section III. We then give some discussions on simulation results in Section IV. In Section V, experimental results are given to verify the effectiveness of the proposed method.

II. 3-D Tracking With Particle Techniques

A. Problem Statement

Consider a tracking task conducted by a reconfigurable vision system, whose viewpoint can be modified during the tracking process. Let $\mathbb{C}^v$ denote the configuration space of the viewpoints in the vision system. Considering the evolution of system state sequence, let $\zeta_k \in \mathbb{C}^v$ denote an element of this configuration space at state k $(k \in \mathbb{N})$. Without loss of generality, ζ_k can be written in a vector form of "generalized viewpoint" [22] as $(x_k^v, y_k^v, z_k^v, \alpha_k^v, \beta_k^v, \gamma_k^v)^{\mathrm{T}}$ which indicates the 3-D spatial position and orientation of the viewpoint with viewpoint configuration v at state k. The objective of dynamic view planning, in this case, is to find a the best configuration ζ_k^* of the vision system, which optimizes certain performance index PI_k of tracking $\zeta_k^* = \arg\min_{\zeta_k}(\mathrm{PI}_k)$.

A 3-D tracking problem is formulated as an estimation of state vector of the target object given by

$$\mathbf{x}_k = \left(x_k^t, y_k^t, z_k^t\right)^{\mathrm{T}} \tag{1}$$

where $(x_k^t, y_k^t, z_k^t)^{\mathrm{T}}$ is 3-D spatial position of the target. Note that increasing the number of tracked points can introduce other state variables (such as orientation parameters) into the definition. When taking the continuity of target motion into account, we employ a second-order autoregressive model

$$\mathbf{x}_k = A_1\mathbf{x}_{k-1} + A_2\mathbf{x}_{k-2} + \varepsilon_k \tag{2}$$

as the dynamic model and it can be rewritten as

$$X_k = AX_{k-1} + \mathrm{E}_k \tag{3}$$

where $X_k = \begin{bmatrix} \mathbf{x}_k \\ \mathbf{x}_{k-1} \end{bmatrix}$, $A = \begin{bmatrix} A_1 & A_2 \\ \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}$, $\mathrm{E}_k = \begin{bmatrix} \varepsilon_k \\ \mathbf{0}_{3\times1} \end{bmatrix}$, and ε_k is an independent identically distributed (i.i.d.) process Gaussian noise sequence with zero mean and covariance Q_k, i.e., $\varepsilon_k \sim N(\mathbf{0}, Q_k)$.

The objective of tracking is to recursively estimate $\mathbf{x}_k$ from measurements

$$\mathbf{y}_k = h_k(\mathbf{x}_k, \eta_k) \tag{4}$$

where h_k is a nonlinear function in general and η_k is an i.i.d. process Gaussian noise sequence.

According to camera perspective projection rule [23], the following observation relation can be employed among the augmented vectors of $\mathbf{x}_k$, $\mathbf{y}_k$, and η_k, which can be obtained with homogeneous coordinates

$$\tilde{\mathbf{y}}_k = sFD^{-1}\tilde{\mathbf{x}}_k + C(\tilde{\mathbf{x}}_k) + \tilde{\eta}_k \tag{5}$$

where s is a scale factor, F is the projection matrix of the vision system, D is the rigid transformation matrix of camera motion, $D = \begin{bmatrix} R & t \\ \mathbf{0}_3^{\mathrm{T}} & 1 \end{bmatrix}$ with R and t being the rotational motion and the translational motion, and C is a function for

modeling nonlinearity (such as lens distortion). Obviously, D is a function of viewpoint configuration ζ_k.

B. Developing the Particle Framework

At time step k, when a measurement $\mathbf{y}_k$ becomes available, according to the Bayes' rule [2], the posterior probability function of the state vector can be calculated using the following equation:

$$p(\mathbf{x}_k|\,\mathbf{y}_{1:k}) = \frac{p(\mathbf{x}_k|\,\mathbf{x}_{k-1})p(\mathbf{y}_k|\,\mathbf{x}_k)}{p(\mathbf{y}_k|\,\mathbf{y}_{1:k-1})}. \tag{6}$$

$P(\mathbf{y})$ is a normalizing constant. It can be calculated as $p(\mathbf{y}_k|\,\mathbf{y}_{1:k-1}) = \int p(\mathbf{x}_k|\,\mathbf{x}_{k-1})p(\mathbf{y}_k|\,\mathbf{x}_k)\,d\mathbf{x}_k$. Because $P(\mathbf{y})$ is a constant, (6) can be written as

$$p(\mathbf{x}_k|\,\mathbf{y}_{1:k}) \propto p(\mathbf{x}_k|\,\mathbf{x}_{k-1})p(\mathbf{y}_k|\,\mathbf{x}_k). \tag{7}$$

Suppose at time step k there is a set of particles, $\{\mathbf{x}_k^i, i = 1,\ldots,N_\mathrm{s}\}$ with associated weights $\{w_k^i, i = 1,\ldots,N_\mathrm{s}\}$ randomly drawn from *importance sampling* [24], [25], where N_s is the total number of particles. The weight of particle i can be defined as

$$w_k^i \propto w_{k-1}^i \frac{p\left(\mathbf{x}_k^i|\,\mathbf{x}_{k-1}^i\right) p\left(\mathbf{y}_k|\,\mathbf{x}_k^i\right)}{q\left(\mathbf{x}_k^i|\,\mathbf{x}_{k-1}^i, \mathbf{y}_{1:k}\right)} \tag{8}$$

where $q(\mathbf{x}_k^i|\,\mathbf{x}_{k-1}^i, \mathbf{y}_{1:k})$ is the importance density function. In this paper, we use the transition prior $p(\mathbf{x}_k|\,\mathbf{x}_{k-1})$ as the importance density function. Then, (8) can be simplified as

$$w_k^i \propto w_{k-1}^i p\left(\mathbf{y}_k|\,\mathbf{x}_k^i\right). \tag{9}$$

Furthermore, if we use Grenander's factored sampling algorithm [26], (9) can be modified as

$$w_k^i = p\left(\mathbf{y}_k|\,\mathbf{x}_k^i\right). \tag{10}$$

The particle weights then can be normalized using

$$w_k^{*i} = \frac{w_k^i}{\sum_{i=1}^{N_\mathrm{s}} w_k^i} \tag{11}$$

to give a weighted approximation of the posterior density in the following form:

$$p(\mathbf{x}_k|\,\mathbf{y}_{1:k}) \approx \sum_{i=1}^{N_\mathrm{s}} w_k^{*i}\delta\left(\mathbf{x}_k - \mathbf{x}_k^i\right) \tag{12}$$

where δ is the Dirac's delta function.

In order to calculate $p(\mathbf{y}_k|\,\mathbf{x}_k^i)$ in (10), $\mathbf{x}_k^i$ is projected according to (5), and the reprojection error on the image plane is calculated. The error is defined as a Euclidean distance between the observed location and the reprojected location in the image space. A Gibbs distribution was employed to transform this to a probability.

III. Best Configuration via Dynamic View Planning

A. View-Planning Strategies

Degeneracy phenomenon is a common problem with particle filter. 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 a large effort is engaged to update particles whose contribution to the approximation to posterior states is almost zero. Doucet [25] has shown that the variance of the importance weights can only increase over time so that degeneracy is an inevitable phenomenon with general sequential importance sampling scheme. There are commonly three methods to tackle the degeneracy problem [27]: 1) brute force approach; 2) good choice of importance density; and 3) use of resampling. The brute force approach uses a large enough sampling size to cover the effect of weight degeneration. However, it is often impractical in real-time estimation system. The method of choosing the *optimal importance density* [25] can maximize the *effective sampling size* [24], [28], which is a suitable measure of degeneracy. However, it involves drawing samples from the importance density coupled with the latest observation, which may not be straightforward. The third method involves using the resampling process to reduce degenerate effects [4], [29], [30]. Notwithstanding, it introduces additional computation complexity. Although resampling has been employed a lot in generic particle filter to avoid degeneracy as one of the most popular methods, it cannot directly be used to guide the view-planning process, and it is not our focus in this paper. Dynamic view planning gives us additional degrees of freedom in system optimization, and we are interested in dealing with the degeneracy problem by taking benefit of that freedom. We will later compare the tracking results of our approach to those of the resampling method in the simulation and experiment part.

In this paper, we intend to use the reconfigurability of the visual tracking system to reduce those effects of degeneracy in particle filter. This approach is based on the thought that optimizing the algorithm of PF is equivalent to optimizing tracking performance, as we employ particle techniques for tracking. We develop a method to guide dynamic view planning for optimizing viewpoint configuration via maximizing the *effective sampling size* in particle filter. We will discuss later in next section that the best viewpoint configuration at which the effective sampling size reaches its maximum is actually consistent with the one at which the tracking system achieves its minimum tracking error.

According to [24] and [28], the effective sampling size N_k^{eff} at state k is defined as

$$N_k^{\mathrm{eff}} = \frac{N_\mathrm{s}}{1 + \mathrm{Var}\left(w_k^i\right)} \tag{13}$$

TABLE I
DYNAMIC VIEW-PLANNING ALGORITHM

Assume that at the previous state *k-1*, we have the particle crowd $\{\mathbf{x}_{k-1}^{i}, w_{k-1}^{*i}\}_{i=1}^{N_s}$ and a viewpoint configuration ζ_{k-1}^{*}, then proceed as following at state *k*

DO when $\zeta_k \in \mathbb{C}_k^v$ (search the *k*th viewpoint configuration space)

{

Sampling:

simulate $\mathbf{x}_k^i \sim P(\mathbf{x}_k^i \mid \mathbf{x}_{k-1}^i)$

Weights calculating and normalizing:

make observation at each ζ_k, calculate w_k^i and w_k^{*i} according to (10), (11),

calculate the rate of effective particles λ_k^{eff}

}

View planning:

use the least-squared method to choose the best view plan ζ_k^* according to (16), and mark its corresponding particles and particle weights

Estimation:

use the marked particles and weights corresponding to ζ_k^* to calculate $p(\mathbf{x}_k \mid \mathbf{y}_{1:k})$ according to (12)

View plan executing:

move the current viewpoint to ζ_k^*

Then continue to make observations and calculations at state *k+1*.

where w_k^i is referred to as the "true weight" indicated in (10) and N_s is the number of samples. As N_k^{eff} cannot exactly be evaluated [27], an estimate $\widehat{N_k^{\text{eff}}}$ of N_k^{eff} can be calculated by

$$\widehat{N_k^{\text{eff}}} = \frac{1}{\sum_{i=1}^{N_s} \left(w_k^{*i}\right)^2} \tag{14}$$

where w_k^{*i} is the normalized weight indicated in (13).

We then define the rate of effective particles as

$$\lambda_k^{\text{eff}} = \frac{\widehat{N_k^{\text{eff}}}}{N_s}. \tag{15}$$

Finally, the view-planning task is achieved by computing the best configuration ζ_k^* in the viewpoint configuration space $\mathbb{C}_k^v$ through the following equation:

$$\zeta_k^* = \arg\min_{\zeta_k} \left(\sum_{i=1}^{N_s} \left(w_k^{*i}\right)^2 \right) \Bigg|_{\zeta_k \in \mathbb{C}_k^v} = \arg\max_{\zeta_k} \left(\widehat{N_k^{\text{eff}}} \right) \Bigg|_{\zeta_k \in \mathbb{C}_k^v}. \tag{16}$$

TABLE II
CAMERA PARAMETERS

f	k_u	k_v	ϑ	(u_0, v_0)	K_1	K_2
0.025	8.6×10^{-6}	8.3×10^{-6}	89°	(320, 240)	-0.1	0.1

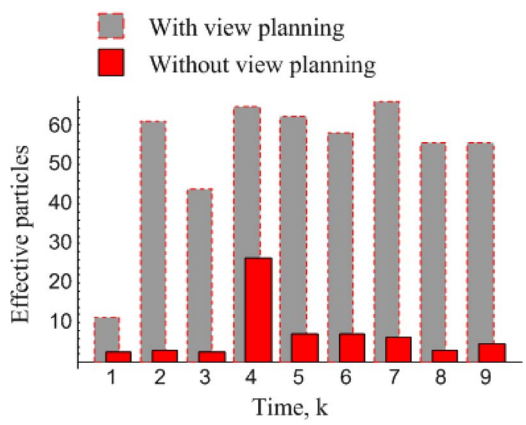

Fig. 1. Number of effective particles.

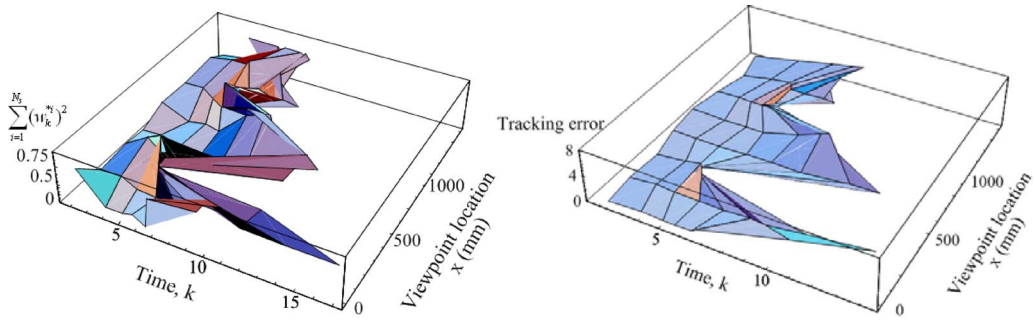

Fig. 2. Testing different evaluation criteria with position parameter x (3-D).

Our dynamic view-planning algorithm is described in Table I. At state k, first, for every candidate of viewpoint configuration ζ_k, we sample $\mathbf{x}_k^i$ from the prior $P(\mathbf{x}_k^i|\,\mathbf{x}_{k-1}^i)$ and calculate its corresponding particle weights w_k^{*i} and rate of effective particles λ_k^{eff}. Then, we use the least-squared method to search for the best configuration ζ_k^* which maximizes λ_k^{eff}, and mark its particle weights and samples. Finally, these samples and weights are used to calculate the posterior by a weighted sum indicated in (12).

B. View-Planning Constraints

The follow constraints can be considered in dynamic view-planning process for tracking.

1) Detectability Constraint: This constraint is to guarantee that the target object can be detected by the vision system. Denote A as a point on the object surface, $\mathbf{n}$ as its normal, S as the vision sensor, $\mathbf{v}$ as its pose, and $\mathbf{v}_{\text{a}}$ as the viewing direction from S to A. We say point A is visible if the dot product of its normal and the viewing direction is negative. That is

$$\mathbf{G1}: \qquad \mathbf{n}\cdot\mathbf{v}_{\text{a}} < 0. \tag{17}$$

This means that the point is visible if the angle (θ) between its normal and the view direction is less than 90°. However, we should set a limit $(\theta_{\max})$ for this angle as the sampling will not be reliable when it is close to 90°. According to (17), we have

$$\mathbf{G2}: \qquad \theta = \pi - \cos^{-1}\frac{\mathbf{n}\cdot\mathbf{v}_{\text{a}}}{\|\mathbf{n}\|\times\|\mathbf{v}_{\text{a}}\|} < \theta_{\max}. \tag{18}$$

Most charge-coupled device (CCD) cameras have a field of view limited by the size of the viewing area and the focal length of the lens. An object point beyond the sensor's field of view will be projected outside the sensor area and will not be detectable. The location, which satisfies the field of view constraint for a set of surface features enclosed by a circumscribing sphere, is given by the following equation:

$$\mathbf{G3}: \qquad \mathbf{v}\cdot\mathbf{v}_{\text{a}} - \|\mathbf{v}\|\cdot\|\mathbf{v}_{\text{a}}\|\cos(\alpha_{\text{s}}/2) \geq 0 \tag{19}$$

where α_{s} is the field-of-view angle of the sensor.

2) In-Focus Constraint: This constraint is to guarantee that the target object is in focus. It introduces a tolerance in position for which the target is still considered acceptably focused based on the resolution of the image sensor [11]. If a point is imaged to a blur circle of a given size c, it is sufficiently considered in focused for a range of depths from $d_{\max}$, the far limit of the depth of field, to $d_{\min}$, the near limit [31], thus the in-focus constraint can be expressed as

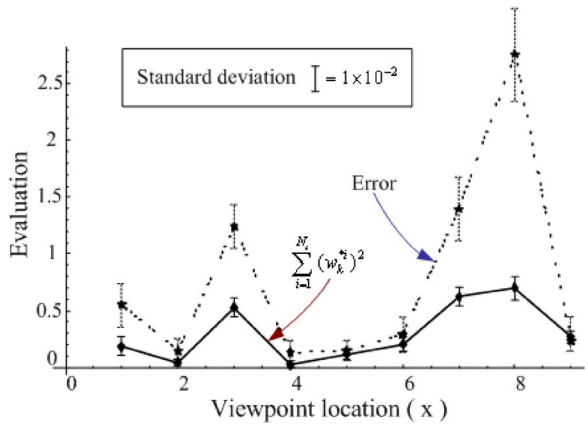

Fig. 3. Testing different evaluation criteria with position parameter x (2-D).

$$\mathbf{G4}: \qquad f < d_{\min} < d_{\max} < 2f. \tag{20}$$

3) View Plan Cost Constraint: This constraint is to guarantee that the time used for view planning (computational complexity) and view plan executing meets the real-time requirement of tracking. Assume that z is the target depth location, N_{res} is resolution alone the shortest direction on the image plane, and $v_{\max}$ is the maximum target velocity, then we can define a constraint for view plan cost time T_{vp} as

$$\mathbf{G5}: \qquad T_{\text{vp}} \leq \kappa\cdot\frac{N_{\text{res}} f}{v_{\max} z} \tag{21}$$

where κ is a scale factor $0 < \kappa < 1$.

4) Mechanical Feasibility Constraint: Because a view plan must be executed by vision system's reconfiguration, the kinematical reachability, feasibility of manipulator of the camera, and constraint for avoiding collision with the environment should also be considered. We thus define a constraint for the viewpoint configuration space as

$$\begin{aligned}\mathbf{G6}: \quad \mathbb{C}^v \in \{&x_{\min} < x^v < x_{\max}, y_{\min} < y^v < y_{\max}\\ &z_{\min} < z^v < z_{\max}, \alpha_{\min} < \alpha^v < \alpha_{\max}\\ &\beta_{\min} < \beta^v < \beta_{\max}, \gamma_{\min} < \gamma^v < \gamma_{\max}\}.\end{aligned} \tag{22}$$

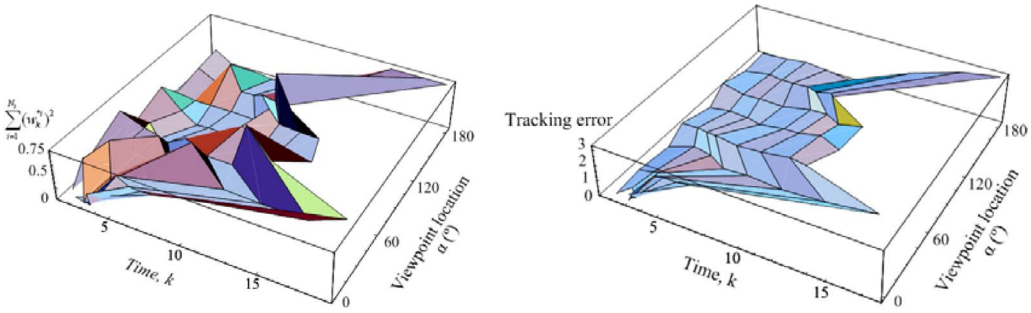

Fig. 4. Testing different evaluation criteria with orientation parameter α (3-D).

IV. Simulation Results and Discussions

In the following simulations, we use a fixed camera model given in (5) for observation. Considering lens distortion, we can modify function F in (5) with

$$F = \begin{pmatrix} f\cdot(1+d)\cdot k_u & -f\cdot(1+d)\cdot k_u\cdot\cot\vartheta & u_0 & 0 \\ 0 & f\cdot(1+d)\cdot k_v/\sin\vartheta & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{23}$$

where f is the focal length, k_u and k_v are the pixel transform factor along u and v axes, respectively, ϑ is the angle between u and v axes, (u_0, v_0) is the location of image center, and d represents nonlinear distortion with

$$d = K_1 r^2 + K_2 r^2. \tag{24}$$

K_1 and K_2 are distortion parameters, r^2 can be calculated with the object location in 3-D space, with $r^2 = (x^2 + y^2)/z^2$.

The values of these parameters are listed in Table II.

A. Effectiveness of the Proposed View-Planning Method

1) Number of Effective Particles: In this simulation, we compared the effective particles (sample size) $\widehat{N_k^{\text{eff}}}$ of the proposed view-planning method to the generic particle filter with a fixed viewpoint. We used 100 particles for each methods and ran the simulation for 100 times with nine state transitions. The average number of effective particles of the two methods are plotted, as shown in Fig. 1. The generic PF with view planning obtained a very low rate of effective particles (7%), while our approach maximized the rate of effective particles $\overline{\lambda^{\text{eff}}}$ at about 53% via dynamic view planning.

2) Tracking Error: Our algorithm realizes view planning and achieves the best configurations of the vision system by maximizing $\widehat{N_k^{\text{eff}}}$ (or minimizing $\sum_{i=1}^{N_s}(w_k^{*i})^2$). In this simulation, we checked tracking errors of different viewpoint configurations (camera locations) to prove that the best configuration in the sense of sampling efficiency is consistent with the best configuration in the sense of minimizing tracking error. As position and orientation parameters of viewpoints may have different properties in view planning, we tested view-planning results with these two kinds of parameter separately. For position parameters, as shown in Fig. 2, average values of $\sum_{i=1}^{N_s}(w_k^{*i})^2$ and tracking errors are plotted with different

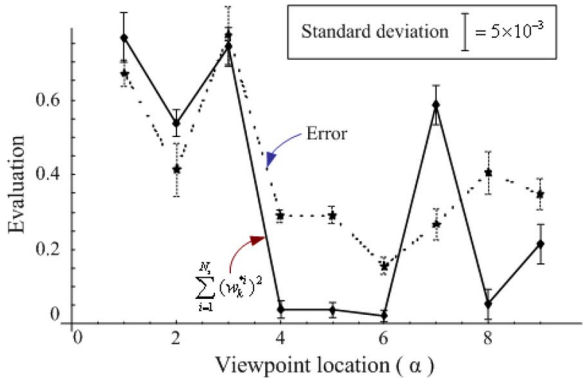

Fig. 5. Testing different evaluation criteria with orientation parameter α (2-D).

configurations and estimation states. Here, the tracking error is defined as the distance between the estimated location and its true location. 100 tests each with 100 particles were employed. Nine viewpoint locations in camera coordinates, $x_i = 170 \times i(\text{mm})$, when $i = 1, \ldots, 9$, were employed. These locations were empirically chosen considering both the sensitivity and the kinematics constraints of the system.

As shown in Fig. 2 that these two evaluation criteria shared the same tendency in viewpoint configuration. The comparison in 2-D figure at the sixth estimation state is shown in Fig. 3. Different evaluation values with their values of standard deviation are plotted. Fig. 3 clearly shows that tracking error reaches its minimum (at the fourth x location, $x = 680$ mm) when $\sum_{i=1}^{N_s}(w_k^{*i})^2$ reaches its minimum value. In other words, the view planning driven by optimizing particle sampling actually minimizes the tracking error and improves tracking performance.

Similar results can be found in the simulation with orientation parameters shown in Figs. 4 and 5. At this time, also nine viewpoint locations in camera orientation parameter, $\alpha_i = 20^\circ \times i(\text{mm})$, when $i = 1, \ldots, 9$, were employed. As shown in Fig. 5, both tracking error and $\sum_{i=1}^{N_s}(w_k^{*i})^2$ reach their minimum at the sixth α location ($\alpha = 120^\circ$), which indicates that the effect of maximizing the effective particle rate is consistent with the effect of minimizing tracking error.

B. Comparison With Other Methods

We first compared our view-planning method to the generic resampling PF with a fixed viewpoint. 100 tests were

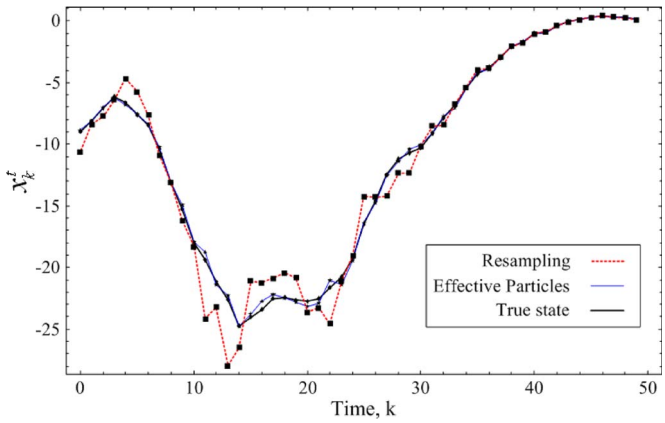

Fig. 6. Comparing with resampling PF (100 particles), when $A_1 = \mathrm{diag}(2,0,0)$, $A_2 = \mathrm{diag}(-1,0,0)$, and $Q_k = \mathrm{diag}(0.9,0,0)$.

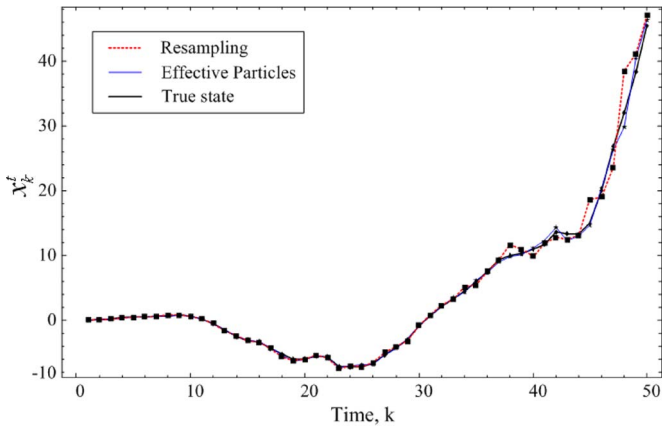

Fig. 7. Comparing with resampling PF (1000 particles), when $A_1 = \mathrm{diag}(1.5,0,0)$, $A_2 = \mathrm{diag}(-1,0,0)$, and $Q_k = \mathrm{diag}(0.9,0,0)$.

run, respectively, with a 100-particle resampling PF and a 1000-particle resampling PF. Only 100 particles were used in our view-planning algorithm for the best configuration as comparison. Tracking results are shown in Figs. 6 and 7. Because our view planning with effective particles actually minimized the tracking error, it showed its advantage even over both resampling PFs.

Then, we compared our method with the "centering" view-planning method. The centering method has usually been adopted in visual servoing [32], which controls the viewpoint to keep the image feature of the target object always at the center point of the image screen. Considering distortion errors and other errors caused by decentering, the centering view-planning method helps in minimizing the sensing error of vision system. One hundred tests with 100 particles for each method were conducted. The average estimation results are shown in Fig. 8. Our method clearly showed its superiority to the centering method.

Finally, we compared our best configuration method with the "error-orientated" view-planning method [21] reviewed in Section I, in which the view-planning process was directly driven by minimizing the estimated tracking error. Fig. 9 shows the average tracking results with 100 tests.

The tracking errors of the aforementioned methods are shown in Fig. 10, and their tracking performances are evaluated

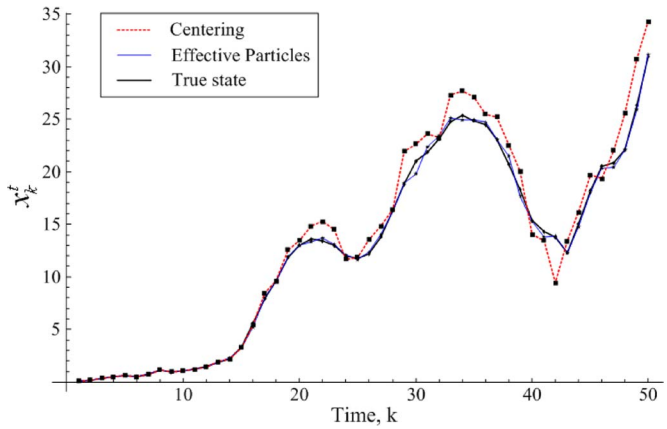

Fig. 8. Comparing with the centering view planning, when $A_1 = \mathrm{diag}(1.5,0,0)$, $A_2 = \mathrm{diag}(-0.5,0,0)$, and $Q_k = \mathrm{diag}(0.9,0,0)$.

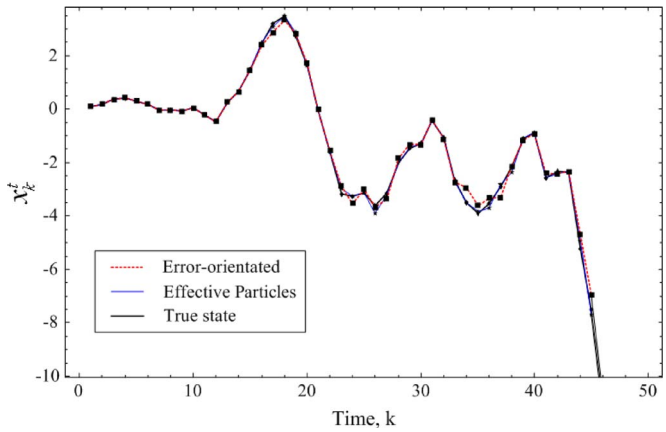

Fig. 9. Comparing with the error-orientated view planning, when $A_1 = \mathrm{diag}(0.95,0,0)$, $A_2 = \mathrm{diag}(-0.5,0,0)$, and $Q_k = \mathrm{diag}(0.9,0,0)$.

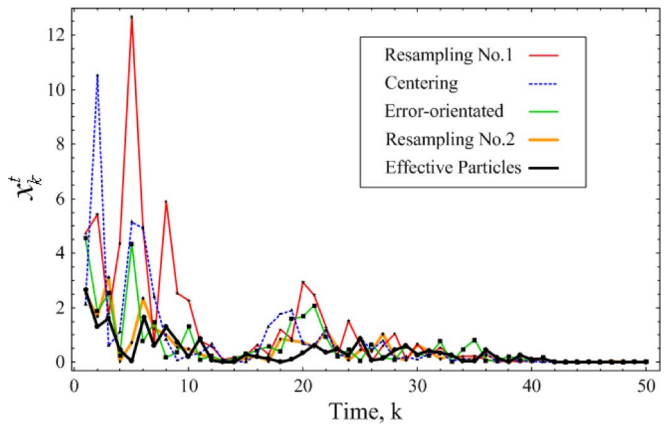

Fig. 10. Tracking errors using different methods, when $A_1 = \mathrm{diag}(0.95,0,0)$, $A_2 = \mathrm{diag}(0.5,0,0)$, and $Q_k = \mathrm{diag}(0.5,0,0)$.

in Table III. In these tests, our view-planning method by effective particles was superior to others in tracking performance with the smallest tracking error and reasonable tracking speed. When the resampling method was used, although it could reduce the effects of weight degeneracy, the tracking error was large when using a small number of particles, whereas tracking speed was slow when using a large number of particles. Without testing the particle weights to obtain the best

TABLE III
TRACKING PERFORMANCE WITH DIFFERENT METHODS

Evaluation / Method	Total number of particles	Average relative tracking error	Average estimation time (s)	Average rate of effective particles
Effective particles	100	2.7%	0.032	49.5%
Resampling PF No. 1	100	7.5%	0.030	6.6%*
Resampling PF No. 2	1000	4.5%	0.266	7.6%*
Centering	100	6.5%	0.027	6.1%
Error-orientated	100	5.5%	0.111	7.1%

(* This is the rate of effective particles $\overline{\lambda^{eff}}$ before resampling. After resampling, the rate of effective particles is compulsory modified to 100%. Even though, it shows no improvement in tracking performance.)

configuration, the centering method showed its advantage in tracking speed. However, because it can only compensate for a part of the sensing error and because it still suffers from particle degeneracy, its tracking error was larger than our method. The error-orientated method was supposed to be the one that could achieve good results in tracking accuracy. However, practically, it needs direct backward calculations and random search in the parameter space which may preclude unique solutions and, yet, cannot overcome the degeneration, which affects its tracking accuracy and the achievable tracking speed as well.

Our method with effective particles minimizes tracking error by revealing the system to a better swarm of importance samples and interpreting the posterior state in a better way. Furthermore, it significantly reduces particles' degeneracy so that a relative smaller particle crowd can be used to achieve the same level of tracking performance and, thus, increases possible tracking speed.

Note that the setup of this simulation was different from that of the simulation shown in Fig. 6. Although the object did not slow down in the end, all tracking errors happened to converge to near zero values after 42 steps.

V. EXPERIMENT

A. System Setup

The implementation of the proposed dynamic view-planning method was conducted using our reconfigurable vision system, with a PC-based IM-PCI system and a variable scan frame grabber. This system supports many real-time processing functions including some feature extraction such as edge detection. Our algorithms were developed in VC++ programming language and run as imported functions by ITEX-CM. The system setup consists of a color CCD camera (model Pulnix TMC-6), with a resolution of 640 × 480 pixels, a pan-tilt unit (model PTU-46-17.5) for two-axis angular motion, and a linear motion system with a guideway (model KK86-20) and motion controller (model Elmo BAS-3/320-2). A photo of this 3-DOF system is shown in Fig. 11.

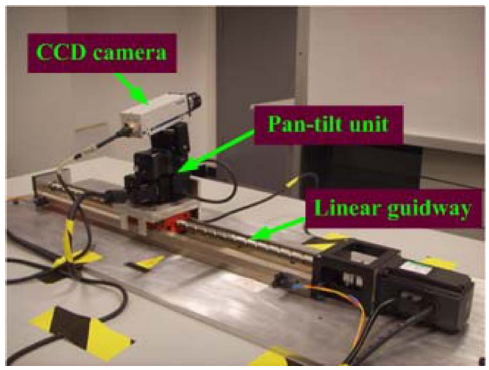

Fig. 11. Reconfigurable vision system with 3 DOF.

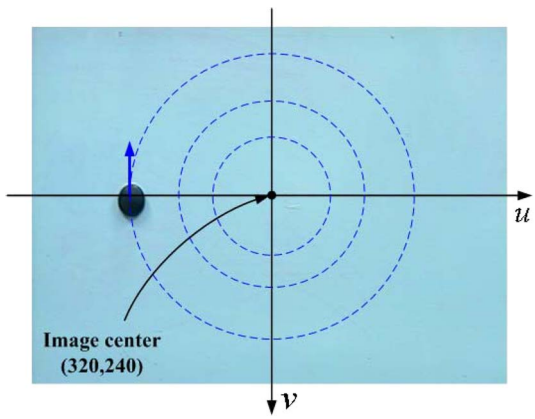

Fig. 12. Uniform circular motion at different diameters.

B. Tracking Error

For simplicity's sake, we used a point object for the experiment and made the object undergo uniform circular motion around the center with different diameters on a plane perpendicular to the optical axis of the camera at its original location (see Fig. 12). Using this motion, we can eliminate both influence of image feature location and influence of velocity on tracking error (see [1] for detailed discussion). The object was made to move along different circles at 0.6 m/s. Then, the magnitude

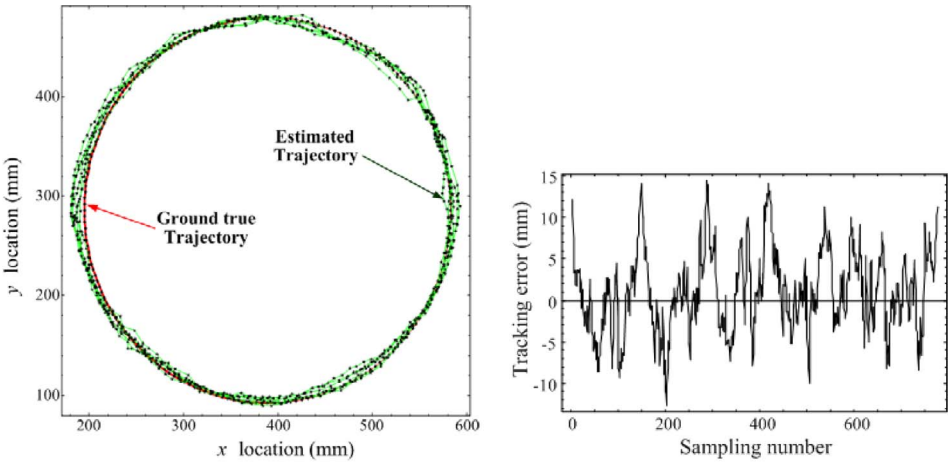

Fig. 13. Tracking error on a circle.

TABLE IV
ABSOLUTE TRACKING ERRORS WITH DIFFERENT VIEW-PLANNING METHODS

Test Diameter $2r$ (pixel)	Effective Particles		Centering		Error-Orientated	
	Mean Tracking Error (mm)	Standard Deviation (mm2)	Mean Tracking Error (mm)	Standard Deviation (mm2)	Mean Tracking Error (mm)	Standard Deviation (mm2)
400	6.2	19.1	7.2	18.5	8.8	21.6
360	6.1	21.1	9.1	19.7	9.0	19.1
320	4.1	18.8	8.3	16.1	8.9	19.6
280	5.6	17.8	6.6	18.8	7.0	19.6
240	5.9	19.5	6.5	15.5	5.5	18.5
200	5.3	19.3	6.2	16.5	8.4	19.5
160	4.6	17.1	7.0	17.3	4.7	18.1
120	5.2	16.6	6.3	15.6	5.9	19.6
80	5.8	18.3	7.5	16.1	5.9	18.6
Mean	5.4	18.6	7.2	17.1	7.1	19.4

of tracking error was defined as the spatial distance between the estimated value of the object location and its true value (see Fig. 13). The target object was detected and tracked by the segmentation based on color and contour cues.

Since the generic resampling PF does not change viewpoint configuration, it is not a view-planning method. We compared our view-planning method with the centering method and error-orientated view-planning method discussed in the previous section. We used 100 particles for each method, and the depth location (z location to the camera) was fixed at 1300 mm. Then, we made the object moved along each of the concentric circles repeatedly and tracked it using different methods. The average tracking errors on different concentric circles with different methods were calculated and are given in Table IV. These results show that our view-planning method with effective particles can achieve small tracking error then other two methods. This superiority in tracking performance can clearly be found in Fig. 14.

We also investigated the number of effective particles during those tracking procedures with different methods. The results are listed in Table V. Using dynamic view planning to maximize the number of effective particles, our method obtained much more effective particles in tracking, and it is also considered the reason why our method was superior to other methods in tracking performance.

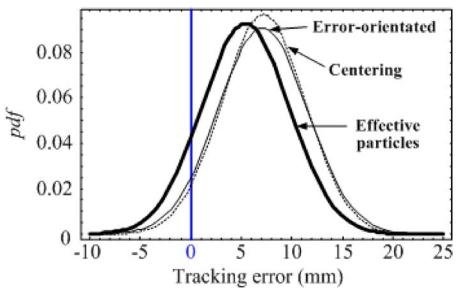

Fig. 14. Probability density functions of estimated tracking errors (constructed from their means and standard deviations).

C. Tracking Speed

Average tracking speeds with different methods are listed in Table VI. Because our method and centering method do not involve random search procedure that the error-orientated method uses, they both achieved very nice real-time performance.

TABLE V
NUMBER OF EFFECTIVE PARTICLES WITH DIFFERENT VIEW-PLANNING METHODS

Test Diameter	Mean Number of Effective Particles (among 100 particles)		
$2r$ (pixel)	Effective Particles	Centering	Error-Orientated
400	39	9	5
360	41	3	8
320	68	16	6
280	55	6	8
240	60	6	19
200	54	27	17
160	72	7	6
120	50	10	15
80	40	7	7
Mean	53	7	10

TABLE VI
TRACKING SPEED WITH DIFFERENT VIEW-PLANNING METHODS

Method	Effective Particles	Centering	Error-Orientated
Tracking Speed (fps)	24	24	7

D. Tracking Constraints

1) BFL: Our previous study [33] on the depth (z) locations shows that there is an optimal z location at which the tracking error reaches its global minimum. This location was called the best focus location (BFL). This is because the tracking resolution is inversely proportional to z, which indicates that a large z is desirable for better tracking. However, when z becomes large, the vision system becomes less sensitive to the object motion. Therefore, the optimal z location is based on the best compromise between the positioning uncertainty and the sensitivity of the vision system. In a dynamic view-planning theme, BFL can be used to provide a constraint for the view plan as well as to guide the reconfiguration of vision system to obtain better tracking performance.

We here used experiment to identify the value of BFL. We made a point object undergo uniform circular motion on a plane perpendicular to the optical axis of the camera at its original location at different z locations (note that according to the aforementioned experimental results, tracking error seemed to have no relation with the diameter of the circular motion). Then, we used our view-planning method to track the object. By comparing to the ground true trajectory, the tracking error was obtained and is shown in Fig. 15. A z location at 1320 mm can be considered as the BFL in our experiment setup.

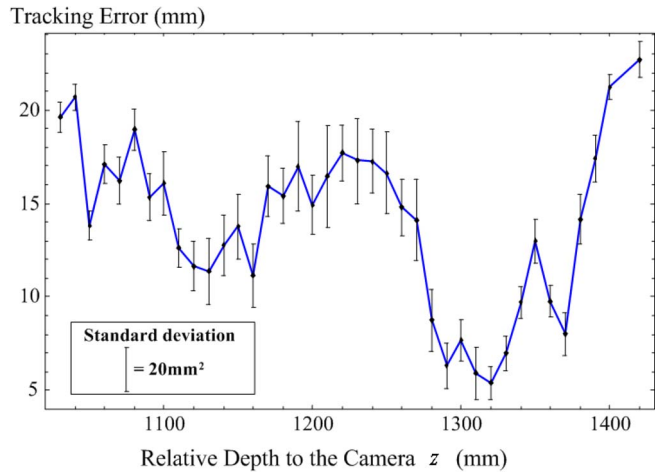

Fig. 15. Effect of z on tracking error.

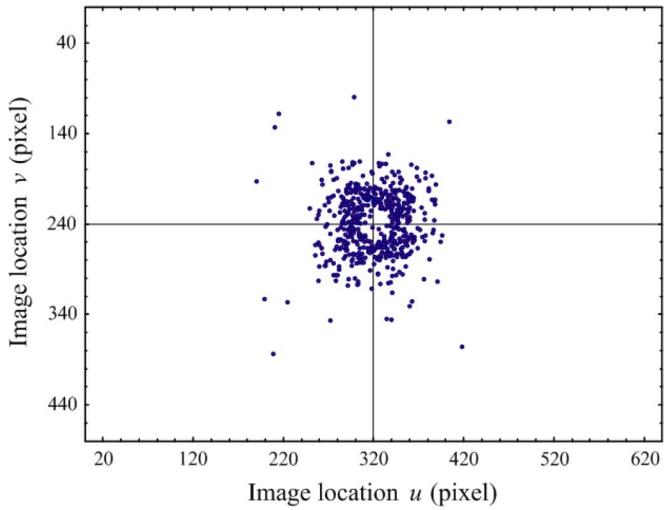

Fig. 16. Image feature location during view planning with our method of effective particles.

2) Optimal Targeting Area: Intuitively, the centering method that always keeps the target object on the image center can reduce the influence of image distortion, particularly radial distortion, and it has been widely used under visual servoing categories. Unlike the centering method, our view-planning method and the error-orientated method are not necessary to keep the object on the image center. With our setup of the reconfigurable vision system, we can try to keep the target feature point at a specific position on the image to minimize tracking error, and this position was called the optimal targeting area (see [1]).

Figs. 16 and 17 show the feature location of the target object on the image during dynamic view planning. Both our method (Fig. 16) and the error-orientated method (Fig. 17) kept the target feature in the neighborhood of the image center, which help reduce influences of distortion. However, as the result of using our method, image feature was located in a donut shape avoiding the image center. This result is in fact consistent with our previous research on optimal targeting area [1], which helps to minimize tracking error.

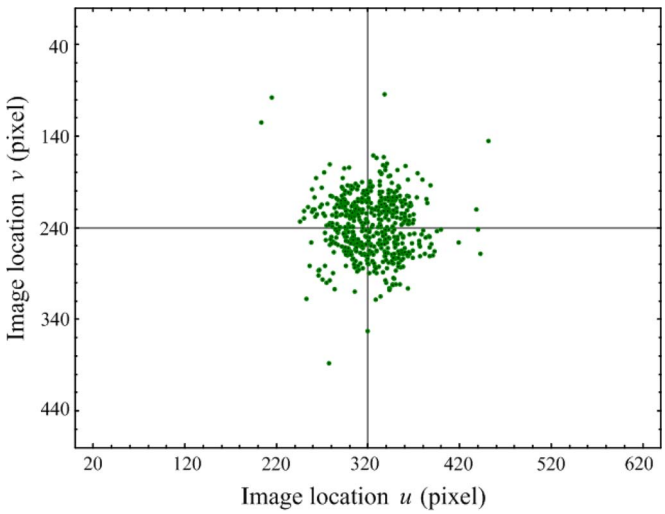

Fig. 17. Image feature location during view planning with the error-orientated method.

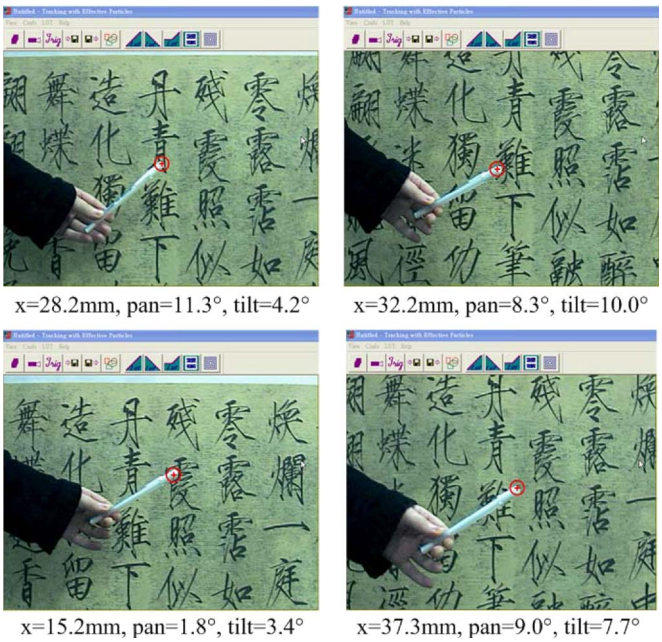

Fig. 18. Dynamic view planning in 3-D tracking by our method.

E. Dynamic View Planning in 3-D Tracking

Our dynamic view-planning method with effective particles was then implemented to track a pen tip which was moving randomly with an average speed about 1 m/s. The pen tip was detected and tracking based on the segmentation with color and contour cues. Some examples of snapshots in the tracking with their corresponding viewpoint locations (best configurations) are shown in Fig. 18. In this experiment, every pair of two sequential frames was employed and compared to calculate the depth information z, and beside the current configuration information, the Chinese calligraphy background was used as correspondence between every two frames for further modification. Even with this process, a tracking rate of about 17 ft/s was achieved in the implementation. We then reprojected the estimated 3-D locations of the pen tip onto the image space for tracking error analysis. The red target marks in Fig. 18 represent those estimations from our tracking algorithm. Experimental results show that the tracking was conducted with very good accuracy, with an average tracking error of 2.8 pixels.

VI. Conclusion

In this paper, a new method of dynamic view planning for 3-D tracking has been presented. The proposed view-planning method is based on the use of an improved particle filter, whose effective sample size has been maximized. In our approach, the vision system has been designed and configured to achieve the largest number of effective particles, which actually minimizes the tracking error by revealing the system to a better swarm of importance samples and interpreting the posterior state in a better way. Simulation and experimental results verified the effectiveness of the proposed method and showed superiorities of our method in accuracy and tracking speed as well by comparing to other view-planning methods. In particular, for a real-time 3-D tracking case, the proposed method can obtain a tracking rate of about 17 ft/s with very accurate tracking results.

References

[1] 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.
[2] J. M. Bernardo and A. F. M. Smith, *Bayesian Theory*. Hoboken, NJ: Wiley, 1994.
[3] Y. Zhang and Q. Ji, "Active and dynamic information fusion for multisensor systems with dynamic Bayesian networks," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 2, pp. 467–472, Apr. 2006.
[4] 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.
[5] A. Doucet, N. D. Freitas, and N. Gordon, *Sequential Monte Carlo Methods in Practice*. New York: Springer-Verlag, Jan. 2001.
[6] M. Isard and A. Blake, "CONDENSATION—Conditional density propagation for visual tracking," *Int. J. Comput. Vis. (IJCV)*, vol. 29, no. 1, pp. 5–28, 1998.
[7] W. R. Scott *et al.*, "Performance-oriented view planning for automatic model acquisition," in *Proc. 31th Int. Symp. Robot.*, 2000, pp. 314–319.
[8] T. Matsumoto and K. Yosui, "Adaptation and change detection with a sequential Monte Carlo scheme," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 37, no. 3, pp. 592–606, Jun. 2007.
[9] H. Y. Chen and Y. F. Li, "Non-model-based view planning for active vision," in *Proc. 11th IEEE Int. Conf.* M^2VIP, Macau, China, Nov. 2004, pp. 7–15.
[10] K. A. Tarabanis *et al.*, "A survey of sensor planning in computer vision," *IEEE Trans. Robot. Autom.*, vol. 11, no. 1, pp. 86–104, Feb. 1995.
[11] S. Y. Chen and Y. F. Li, "Vision sensor planning for 3-D model acquisition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 5, pp. 894–904, Oct. 2005.
[12] R. Gupta and S. R. Das, "Tracking moving targets in a smart sensor network," in *Proc. IEEE Veh. Technol. Conf.*, Oct. 2003, vol. 5, pp. 3035–3039.
[13] J. Hill *et al.*, "System architecture directions for network sensors," in *Proc. Architectural Support Program. Languages Operating Syst.*, Nov. 2000, pp. 93–104.
[14] J. Schiff and K. Goldberg, "Automated intruder tracking using particle filtering and a network of binary motion sensors," in *Proc. IEEE Int. CASE*, Oct. 2006, pp. 580–587.
[15] R. Araujo and A. T. de Almeida, "Learning sensor-based navigation of a real mobile robot in unknown worlds," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 29, no. 2, pp. 164–178, Apr. 1999.
[16] J. Denzler, M. Zobel, and H. Niemann, "Information theoretic focal length selection for real-time active 3-D object tracking," in *Proc. ICCV*, Nice, France, 2003, pp. 400–407.
[17] B. J. Tordoff and D. W. Murray, "A method of reactive zoom control from uncertainty in tracking," *Comput. Vis. Image Understanding*, vol. 105, no. 2, pp. 131–144, Feb. 2007.

[18] L. A. Clemente, A. J. Davison, I. D. Reid, J. Neira, and J. D. Tardós, "Mapping large loops with a single hand-held camera," in *Proc. RSS*, Atlanta, GA, Jun. 2007.
[19] S. Martinez and F. Bullo, "Optimal sensor placement and motion coordination for target tracking," in *Proc. IEEE ICRA*, 2005, pp. 1368–1373.
[20] Y. Bar-Shalom *et al.*, *Estimation with Applications to Tracking and Navigation*. New York: Wiley, 2001.
[21] J. R. Spletzer and C. J. Taylor, "Dynamic sensor planning and control for optimally tracking targets," *Int. J. Robot. Res.*, vol. 22, no. 1, pp. 7–20, Jan. 2003.
[22] C. Giraud and B. Jouvencel, "Sensor selection: A geometrical approach," in *Proc. IEEE Int. Conf. Intell. Robots Syst.*, 1995, pp. 555–560.
[23] D. A. Forsyth and J. Ponce, *Computer Vision: A Modern Approach*. Englewood Cliffs, NJ: Prentice-Hall, 2003.
[24] N. Bergman, "Recursive Bayesian estimation: Navigation and tracking applications," Ph.D. dissertation, Linköping Univ., Linköping, Sweden, 1999.
[25] 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.
[26] U. Grenander, Y. Chow, and D. Keenan, *HANDS. A Pattern Theoretical Study of Biological Shapes*. New York: Springer-Verlag, 1991.
[27] M. S. Arulampalam *et al.*, "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.
[28] J. S. Liu and R. Chen, "Sequential Monte Carlo methods for dynamic systems," *J. Amer. Stat. Assoc.*, vol. 93, no. 443, pp. 1032–1044, Sep. 1998.
[29] J. Carpenter, P. Clifford, and P. Fearnhead, "Improved particle filter for nonlinear problems," *Proc. Inst. Elect. Eng.—Radar, Sonar, Navig.*, vol. 146, no. 1, pp. 2–7, Feb. 1999.
[30] B. Ripley, *Stochastic Simulation*. New York: Wiley, 1987.
[31] S. Abrams *et al.*, "Dynamic sensor planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1993, vol. 2, pp. 605–610.
[32] E. Malis, F. Chaumette, and S. Boudet, "2-1/2-D visual servoing," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, pp. 238–250, Apr. 1999.
[33] H. Y. Chen and Y. F. Li, "Metric measurements of 3D object features by a moving camera," in *Proc. IEEE Int. Conf. Instrum. Meas. Technol.*, Como, Italy, May 2004, vol. 1, pp. 498–503.

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, Hong Kong Polytechnic University, Kowloon, Hong Kong. Her research interests include robot vision, visual tracking, and dynamic view planning.

Youfu Li (S'91–M'92–A'95–SM'01) received the B.S. and M.S. degrees in electrical engineering from Harbin Institute of Technology, Harbin, China, and the Ph.D. degree from the Robotics Research Group, Department of Engineering Sciences, University of Oxford, Oxford, U.K., in 1993.

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

Dr. Li is an Associate Editor of IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING.