Visual Object Tracking based on Real-time Particle Filters

Dong- Hun Lee*, Yong-Gun Jo**, and Hoon Kang***

* School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, Korea
(Tel : +82-2-816-8234; E-mail:dhjames@sirius.cie.cau.ac.kr)
** School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, Korea
(Tel : +82-2-816-8234; E-mail: iceman@sirius.cie.cau.ac.kr)
*** School of Electrical and Electronics Engineering, Chung-Ang University, Seoul, Korea
(Tel : +82-2-816-8234; E-mail: hkang@cau.ac.kr)

Abstract: Particle filter is a kind of conditional density propagation model. Its similar characteristics to both selection and mutation operator of evolutionary strategy (ES) due to its Bayesian inference rule structure, shows better performance than any other tracking algorithms. When a new object is entering the region of interest, particle filter sets which have been swarming around the existing objects have to move and track the new one instantaneously. Moreover, there is another problem that it could not track multiple objects well if they were moving away from each other after having been overlapped. To resolve reinitialization problem, we use competitive-AVQ algorithm of neural network. And we regard interfarme difference (IFD) of background images as potential field and give priority to the particles according to this IFD to track multiple objects independently. In this paper, we showed that the possibility of real-time object tracking as intelligent interfaces by simulating the deformable contour particle filters.

Keywords: Particle Filter**,** CONDENSATION, Active Contour, Object Tracking

1. INTRODUCTION

Object tracking is one of the most intriguing problems in the computer vision area. It is not only related to the problem of the shape and the color tracking of an object but also linked to that of object recognition or identification of a particular object. Recently, in the computer vision area, visual tracking of moving objects is an active research issue since the particle filter approaches [1][2][3] were proved to be efficient in object tracking especially in the cluttered environment. Particle filters are based upon the Bayesian conditional probabilities such as prior distributions and posterior ones. It is strongly believed that intelligence in robot vision would be enhanced through particle filters or the CONDENSATION(CONditional DENSity PropagATION) algorithm [3] [4] since the tracking information will eventually increase the machine intelligent quotient (MIQ) in terms of target object context, visual planning and control. In addition to clutter, visual uncertainties such as nonlinear dynamics, non-Gaussian density, occlusion and lightness change make particle filters a necessary component in robot vision.

In this paper, we address a robust object tracking algorithm based on multiple observation models in particle filters. First of all, we choose a static webcam as the vision sensor, and assume that a human is moving in clutter. The research goal is to track a moving face or hand with the vision sensor. Now, we apply our particle filter, namely, CONDENSATION algorithm [5] to keep track of the moving face or hand. The particle filters are a class of stochastic approximations of the state posterior with a set of N weighted particles or samples $\{X_t^{(i)}, X_t^{(i)}\}_{i=1,...,N}$ where $X_t^{(i)}$ is a possible state of the ith particle at time t and $\pi_t^{(i)}$ is the associated weight. Therefore, we perform the Monte-Carlo simulation of underlying probability distributions which may take arbitrary form, instead of deriving the analytic solution. The state $X_t^{(i)}$ may be any measurable contexts such as position, velocity, rotating angle, scaling factor, tilting, etc. of a moving object. Moreover, the measure Z_t may be any measurable

quantities such as image contrast, digital image subtraction, edge-detected silhouette, 2D or 3D contours, RGB or HSV colors, etc. Each particle (contour as a particle) represents a possible state with the associated weight of a likelihood, which is measurable or computable.

In particle filters, we keep track of the state samples with the non-zero posterior probability approximated by the ensemble of the weights on all of these sampled particle sets and the idea is based upon factored sampling [6] in Bayesian inference rule described by the conditional posterior density as

$$P(X_t \mid Z^t) = \kappa P(Z^t \mid X_t)P(X_t \mid Z^{t-1})$$
$$= \kappa P(Z_t \mid X_t)\int_{X_{t-1}} P(X_t \mid X_{t-1})P(X_{t-1} \mid Z^{t-1})$$
$$\cong \kappa P(Z_t \mid X_t)\sum_{i=1}^{N} \pi_{t-1}^{(i)}P(X_t \mid X_{t-1}^{(i)}) \qquad (1)$$
$$= \kappa P(Z_t \mid X_t)q(X_t)$$

where X_t is the state at time t , $Z^t = \{Z_1,...,Z_t\}$ is the observation up to time t , κ is a proportional constant, $P(X_t \mid X_{t-1})$ is the Markov-chain motion model, $P(Z_t \mid X_t)$ is the measurement model, and $\pi_{t-1}^{(i)}$ is the weight for particle $X_{t-1}^{(i)}$. Here, $q(X_t) = \sum_{i=1}^{N} \pi_{t-1}^{(i)}P(X_t \mid X_{t-1}^{(i)})$ is the proposal distribution from which N samples $X_t^{(i)}$ are drawn by Monte-Carlo approximation of the integral. In brief, the posterior density $P(X_{t-1} \mid Z^{t-1})$ is recursively approximated by the particle set $\{X_{t-1}^{(i)}, \pi_{t-1}^{(i)}\}$, and then these samples are weighted by the likelihood $\pi_{t-1}^{(i)} = P(Z_{t-1} \mid X_{t-1}^{(i)})$.

Primarily, the particle filter algorithm consists of three elementary steps: sampling, predicting, and measuring. The basic principle is the conditional probability propagation between the prior density in sampling and predicting, and the posterior density in measurement. Thus, the prior and the posterior conditional densities are calculated by taking turns in a recursive form. Existing algorithms of particle filters include CONDENSATION[4][7], Kalman particle filter (KPF) [8], and Unscented Kalman particle filter (UPF) [9]. Both KPF and UPF add the predicting models by using the state estimate with the Gaussian and non-Gaussian assumptions respectively

and thus improve the tracking characteristics of the prior in terms of better proposal distributions. Three updating steps of tracking in particle filters are summarized as follows:

     1) Sampling (Selection)
     2) Predicting (Particle Dynamics)
     3) Measuring (Output)

In the sampling phase, the importance sampling [7] incorporates hints or other sources by introducing samples at more likely spots in the state space while partitioned sampling [10] suggests that particles may be sampled hierarchically for efficiency in order to obtain equivalent results with fewer samples. Therefore, in the subsequent section, we investigate two observation models in particle filters: the point clouds of skin colors and the active contours of the B-spline snakes.

2. Particle Filters with Multiple Observation Models for Visual Object Tracking

The CONDENSATION algorithm provides a plausible way of tracking a moving object with some features for the vision sensors in terms of a probabilistic propagation process. If the number of the object features is K, the state of the ith particle or contour becomes $X_t^{(i)} \in R^k$, and if the particle dynamics is considered, the number of the state is increased twice as much. The basic tracking algorithm is represented as follows:

CONDENSATION Tracking Algorithm

• Given the old particle or contour set $\{X_{t-1}^{(i)}, \pi_{t-1}^{(i)}\}$, compute a new particle or contour set $\{X_t^{(i)}, \pi_t^{(i)}\}$ for i=1,…, N Initialize $\pi_0^{(i)} = 1/N$, and its cumulative distribution $c_0^{(i)} = i/N$.

1) Select a sample $S_t^{(i)}$ as follows:
   a) Generate a uniformly distributed random number $r \in [0,1]$.
   b) Find the smallest j for which $c_{t-1}^{(j)} \geq r$.
   c) Set $S_t^{(i)} = X_{t-1}^{(j)}$.

2) Predict from sampling to obtain $S_t^{(i)}$.

$$P(X_t \mid X_{t-1} = S_t^{(i)}) \qquad (2)$$

Here, we imbed the particle dynamics by choosing one of many different equations of motion and the estimation procedure in order to find $\hat{S}_{t|t-1}^{(i)}$. Note that deterministic drift and stochastic diffusion are included together by the random number generation.

3) Measure the object features and update the weights in terms of the multiple observation models

$$\pi_t^{(i)} = P(Z_t \mid X_t = \hat{S}_{t|t-1}^{(i)}) \qquad (3)$$

And normalize the posterior so that $\sum_{i=1}^{N} \pi_t^{(i)} = 1$ and compute the cumulative distribution $c_t^{(i)}$. Update $\{X_t^{(i)}, \pi_t^{(i)}\}$. For the multiple observation models, the conjunctive or disjunctive operators may be used in order to combine the multiple object features together by using AND or OR in $P(Z_t \mid X_t)$ multiplication in $\pi_t^{(i)}$.

• Estimate the moments of the tracked positions or other estimates after all N particles or contours are updated as follows:

$$E[g(X_t)] = \sum_{i=1}^{N} \pi_t^{(i)} \cdot g(\hat{S}_{t|t-1}^{(i)}) \qquad (4)$$

For the state estimate $E[g(X_t)]$, it is noted that there is no explicit state estimate maintained, that is, the state information is just a cloud of particle points or samples of active contours. Therefore, we may arbitrarily choose either the mean value of the particles or the confidence region of an inverse covariance matrix. Moreover, one element of the state, for example, the average position of the target object is estimated for updating the next position of the target object centroid.

2.1 Sampling (Selection)

In the sampling stage, we select N particle samples by picking them randomly and uniformly in [0,1] from the normalized weights given by $\sum_{i=1}^{N} \pi_t^{(i)} = 1$. This selection approach is the same as the roulette-wheel selection as in the genetic algorithm (GA), which is equivalent to spinning a roulette wheel with the arc-lengths of bins equal to the particle weights. Then, the particles are resampled according to the score, that is, highly likely particles are duplicated while unlikely ones are eliminated. This results in adaptive focusing on more promising areas of the state space.

2.2 Prediction (Particle dynamics)

The prediction step is to place each particle or contour from the generative form of dynamics given by

$$S_t^{(i)} = f(S_{t-1}^{(i)}) + \varepsilon_{t-1} \qquad (5)$$

Where f() is a deterministic component and ε_t is a random component. Usually, f() represents a nonlinear drift of each particle with different displacement and ε_t is an independent diffusion of each particle typically modeled with a white noise. Therefore, we often use the extended Kalman filters (EKF) as the prediction model in order to estimate $\hat{S}_{t|t-1}^{(i)}$ by assuming that noise characteristics complies with Gaussian. The particle or contour dynamics is related to the observation model in such a way that it might lead to the features of higher likelihood observation as shown in Figure 1.
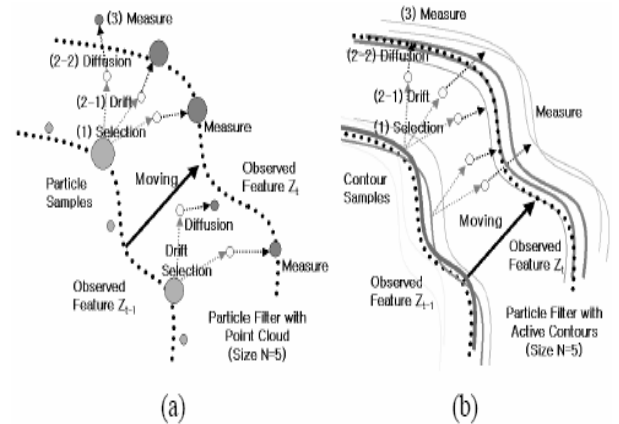

Figure 1. Observation Mechanisms in the Predicting and the Measuring Stages: (a) 5 Particle Samples of Point Cloud and (b) 5 Particle Samples of Deformable Contours

2.3 Measurement (Output)

For each particle or contour $X_t^{(i)}$, we compute new weight $\pi_t^{(i)}$ using the measurement likelihood

$$\pi_t^{(i)} = \mathrm{P}(\mathrm{Z}_t|\mathrm{X}_t^{(i)}) \qquad (6)$$

and find the plausibility conditions such that the particles with impossible configurations are given zero likelihood. For example, the positions outside of the image features of interest are given $\pi_t^{(i)} = 0$.

2.4 Multiple Observation Models

In this paper, we choose two object features for face or hand tracking. One is the contrast information from the digitally subtracted images and the other is the skin-color information from the HSV color model. The former measure of the subtracted images ΔP_t is defined by

$$\Delta P_t(x,y) = |\mathrm{I}_t(\mathrm{x,y}) - \mathrm{I}_{t-\Delta}(\mathrm{x,y})| \qquad (7)$$

where $I_t(x,y)$ is the $(x,y)^{th}$ pixel value of the image at time t in [0,1] and $I_{t-\Delta}$ is the delayed image at time frame $t-\Delta$. Also, the latter measure of the skin color is based on the HSV model where the hue and the saturation information are used by computing the Gaussian density function

$$C_t(x,y) = \exp(\frac{-(\mathrm{h(x,y)}-\mathrm{h_R(x,y)})^2 - (s(x,y)-s_R(x,y))^2}{2\sigma_{hs}^2}) \quad (8)$$

Where σ_{hs} is the equivalent standard deviation for the HS features;

$C_t(x,y)$ is the $(x,y)^{th}$ feature of the skin color; $h(x,y)$ and $s(x,y)$ are the hue and the saturation values of a color-measured control point at (x,y), and $h_R(x,y)$ and $s_R(x,y)$ are the hue and the saturation references for the initial particle subset, respectively. Now we consider the observation models for the convectional particle sets of point cloud and for the deformable contours [5] of the B-spline snakes.

• Point Cloud as Particles:

- For the ith particle's point $(x_p,y_p) \in \Delta P_i$ of both the subtracted pixel and the associated skin-color point, the weight of the prior is computed by using the disjunctive operator (OR):

$$w_t^{(i)} = \sum_{(\mathrm{x_P,y_P}) \in \Delta P_i} \{\Delta P_t(\mathrm{x_P,y_P}) + \lambda \mathrm{C}_t(\mathrm{x_P,y_P})\} \qquad (9)$$

- The state vector for one particle is defined as $X_t^{(i)} = [x_P(t), y_P(t), x_P(t-1), y_P(t-1)]^T$ at time t and the dynamics is represented by

$$X_t = \mathrm{AX}_{t-1} + B_{v_t} \qquad (10)$$

where *A, B* are the appropriate matrices, and v_t is the white noise with diffusion covariance matrix about the two axes.

• Deformable Contours as Particles:

- For the control points of the ith contour, $(x_P,y_P) \in P_i$ of the subtracted image, and for the corresponding i^{th} skin-color control points $(x_C,y_C) \in C_i$, the weight of the prior is computed by using the conjunctive operator (AND):

$$w_t^{(i)} = \sum_{(x_P,y_P) \in \Delta P_i} \Delta P_t(x_P,y_P) \sum_{(x_C,y_C) \in C_i} \lambda C_t(\mathrm{x_C,y_C}) \quad (11)$$

- The state vector for one particle is defined as

$X_t^{(i)} = [\overline{\mathrm{x}_P}(\mathrm{t}), \overline{\mathrm{y}_P}(\mathrm{t}), \theta(\mathrm{t}), \alpha(\mathrm{t}), \overline{\mathrm{x}_P}(\mathrm{t\text{-}1}), \overline{\mathrm{y}_P}(\mathrm{t\text{-}1}), \theta(\mathrm{t\text{-}1}), \alpha(\mathrm{t\text{-}1})]^{\mathrm{T}}$ is the centroid of the control points in ΔP_i; θ, α are the rotating angle and the scaling factor of the B-spline snake, respectively; and the dynamics is similarly described by (10).

Note that λ is the balancing coefficient between two independent object features as the conjunctive or disjunctive operators are used for them. Also, the posterior π_t is obtained by resampling and normalization such as

$$\pi_t = \mathrm{f}_\mathrm{N}[f_R(f_N(w_t))] \qquad (12)$$

where, f_N, f_R are the normalization and the resampling functions, respectively, and w_t is given in (9) and (11). The deformable contour of the ith particle has the form of the planar affine shape space as

$Q' = \mathrm{WY}_t + Q$

$$W = \begin{bmatrix} 1 & 0 & \mathrm{Q_x} & 0 & 0 & \mathrm{Q_y} \\ 0 & 1 & 0 & \mathrm{Q_y} & \mathrm{Q_x} & 0 \end{bmatrix} \qquad (13)$$

$Y_t = [\overline{\mathrm{x}_P}, \overline{\mathrm{y}_P}, \cos\ \theta-1+\alpha, \cos\ \theta-1+\alpha, \sin\ \theta, \sin\ \theta]^{\mathrm{T}}$

where W is the shape matrix, Y_t is the shape-space vector computed from X_t, and $Q', Q = [\mathrm{Q_x^T}, \mathrm{Q_y^T}]^T$ are the updated and the reference control point vector, respectively. Here, the updated control point vector Q' is represented by

$$Q' = [\mathrm{x_{P,1}(t),...,x_{P,n}(t);y_{P,1}(t),...,y_{P,n}(t)}]^{\mathrm{T}} \qquad (14)$$

where *n* is the total number of the control points in a B-spline snake.

3. Experimental Results of Visual Object Tracking with Particle Filters

Now three cases of the multiple observation models are applied to visual object (face or hand) tracking based on the particle filters in the experiment.

3.1 Robust Face Tracking Based On Particle Filters of Point Cloud

For the simulation of face tracking with a particle cloud, we choose the number of particles, 50 $N =$ and the particle observation model, (9) in order to combine both the contrast and the skin-color features together. Here, $\lambda = 1/\mathrm{N}$, and the threshold of $\pi^{(i)}$ for the effective particle size is $2/N$ for adaptive resampling. The standard deviations for diffusion in both axes are $\sigma_x = \sigma_y = 30$ and those for the hue and the saturation are selected as $\sigma_{hs} = 0.05$. Figure 2 shows visual face tracking with particle filters of point cloud using both subtracted images and skin colors.
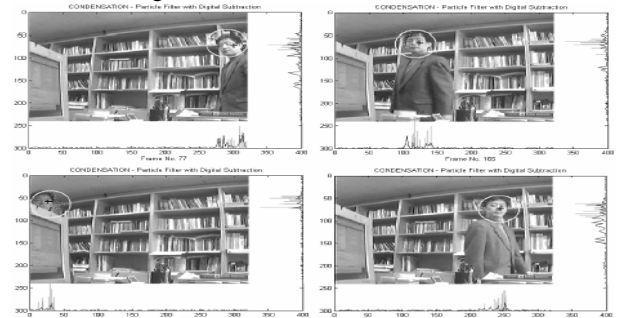

Figure 2. Experiment of Visual Face Tracking with Particle Filters of Point Cloud using Both Subtracted Images and Skin Colors (N=50)
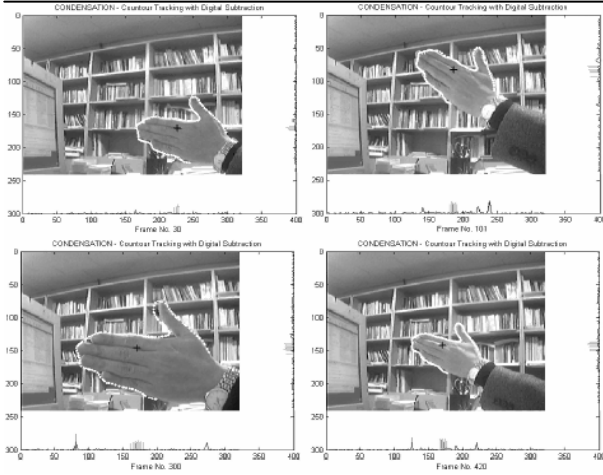
Figure 3. Experiment of Visual Hand Tracking with Deformable Contours using Subtracted Images Only
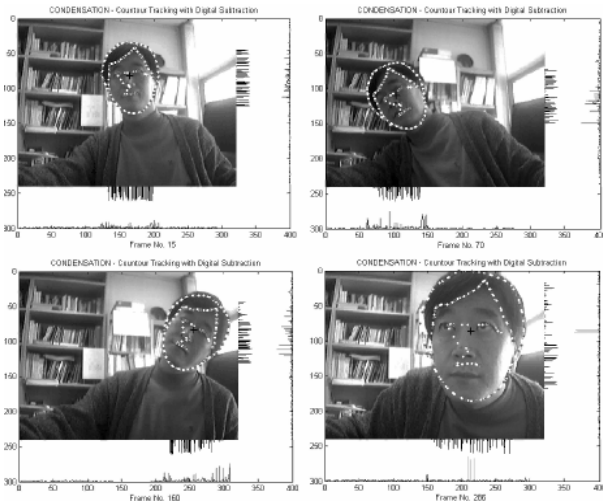

Figure 4. Experiment of Visual Face Tracking with Deformable Contours using Both Subtracted Images and Skin Colors (N=100)
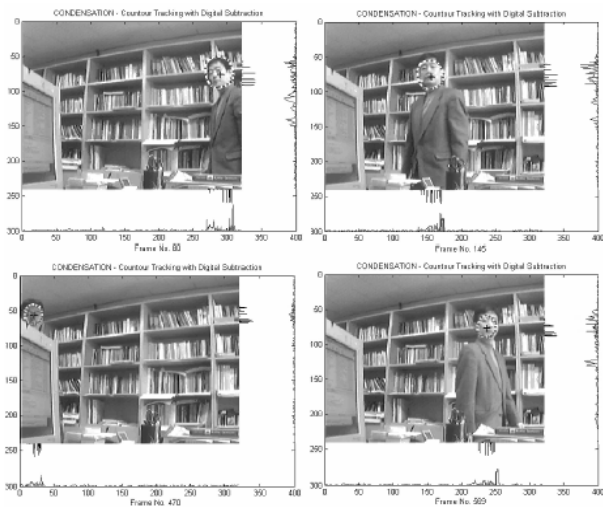

Figure 5. Experiment of Visual Face Tracking with Deformable Contours using Both Subtracted Images and Skin Colors (N=50)

3.2 Robust Hand Tracking Based On Deformable Active Contours

For the simulation of hand tracking with contours, we choose the number of particles, 200 $N = 200$ and the deformable observation model, (9). But, in this case, we take only the contrast features into account by setting $\lambda = 0$, the number of control points in the B-spline snake is n=45, and the threshold of $\pi^{(i)}$ for the effective particle size is $1.5/N$ during adaptive resampling. The standard deviations for diffusion in both axes are $\sigma_x = \sigma_y = 5$ and those for the rotating angle and the scaling factor are selected as $\sigma_\theta = 3(\text{deg.}), \sigma_\alpha = 0.05$, respectively. In Figure 3, it is demonstrated that visual hand tracking is performed with the B-spline snake of deformable contours using subtracted images only.

3.3 Robust Face Tracking Based On Deformable Active Contours and Skin-Colors

Finally, for the simulation of human face tracking with both point clouds and contours, we choose the number of particles, N=50 or N=100 and the observation model, (11) in order to combine both the contrast and the skin-color features together.

Here, the number of control points is n=21 and the threshold of $\pi^{(i)}$ for the effective particle size is $1.5/N$ for adaptive resampling. The standard deviations for diffusion in both axes are $\sigma_x = \sigma_y = 5$, those for the rotating angle and the scaling factor are selected as $\sigma_\theta = 1(\text{deg.}), \sigma_\alpha = 0.01$, respectively, and those for the hue and the saturation are selected as $\sigma_{hs} = 0.05$. However, by using the multiple observation models $(\lambda = 1/N)$ such as deformable contours and a constrained point cloud of both subtracted images and skin colors under the same conditions, we could significantly reduce the number of particle sets in the above cases by half with better tracking results as shown in Figures 4 and 5, respectively.

4. Competitive-AVQ for Reinitialization Procedure

Particle filter of point cloud can be realized in real-time. However, particle filter has inherent problem which is initialization point. When a new object is detected somewhere part of interface, particle filter Set in pause position has to move and tracking the new object instantaneously. In order to do that, we use competitive-AVQ algorithm of neural network. This algorithm defines an area which made stripe of image plane boundary. And it is used when IFD data exceeds a threshold value of stripe area. The c-AVQ algorithm is represented as follows:

---

c-AVQ for PF Reinitialization

- Check boundary stripes P_B with threshold (3%~4% IFD data), if the condition holds,
- For all $p_t(k)$ in $P_B >$ threshold at time t, find the winner (the ith cluster)

$$w_i^*(k) = \arg \cdot \min_j \left\{ \left\| w_j(k) - p_t(k) \right\|_2 \right\} \qquad j \to \#(\text{cluster})$$

$$w_i^*(k+1) = w_i^*(k) + \mu \left(p_t(k) - w_i^*(k) \right) \qquad k \to \#(P_B)$$

- Update the ith particle state s.t. the average is shifted to the ith cluster center (the winner)

$$E\left\lceil x_t^i \right\rceil = w_i^*$$

---

5. Priority-based Subtraction of Background IFD & Effective Particle Size

Multiple object tracking regards interframe difference (IFD) as potential field (V). After regarding that, it determines the priority for many particles and repeats a sampling (selection), particle dynamic, output one by one how to subtract with priority at IFD buffer. This subtraction method is like subtractive clustering algorithm. If this method is used, object tracking can be done at different direction though two objects cross each other.

$$V^0 = \Delta P$$

$$V^K = \Delta P - V^{k-1} \,, \quad k = 1 \sim \# \text{ (PF Set)} \tag{15}$$

When several measurement values or a uniform posterior distribution is appeared, effective particle size, N_{eff}, is used in resampling stage and is defined as follows:

$$N_{eff} = 1/(\sum_{i=1}^{N} \pi_i^2) \tag{16}$$

Therefore, if N_{eff} is very small or large, resampling is attempted. In this case will be happen when the object is paused and when the tracking is failed. Figure 6 shows experiment result using real-time multiple object tracking based on point cloud method.

6. Conclusions and Discussion

In this paper, the CONDENSATION algorithm is extended to combining the multiple observation models such as particle clouds, contour tracking, digital subtraction and skin color, with application to human face or hand tracking, where multiple object features are measured with the prior and posterior probability densities are propagated in order to obtain the state estimates such as the target average position, the angles of principal axis, the scaling factors, etc. From the experimental results, the suggested particle filters with multiple observation models demonstrated robust visual tracking in uncertainties such as clutter and nonlinear motion.

The advantages of particle filters with multiple observation models are demonstrated here with application to face or hand tracking in robot vision. First, the particle filters can have target object dynamics embedded easily by combining nonlinear dynamics and multiple observation models with a smaller particle size. Second, Clutter in the measurement likelihood that causes the multi-modal posterior can be dealt with robustly. Third, point clouds, control points, and statistical parameters in particle or contour tracking may be simply converted to the image segmentation. And finally, it is possible to apply particle filters to high-level inference of context semantics by using the 2D or 3D contour information as shown in Figure 7 where the tracking result of picture drawing is demonstrated. For the future works, we will investigate the multiple object particle filters as well as the real-time implementation of particle filters for contour tracking.
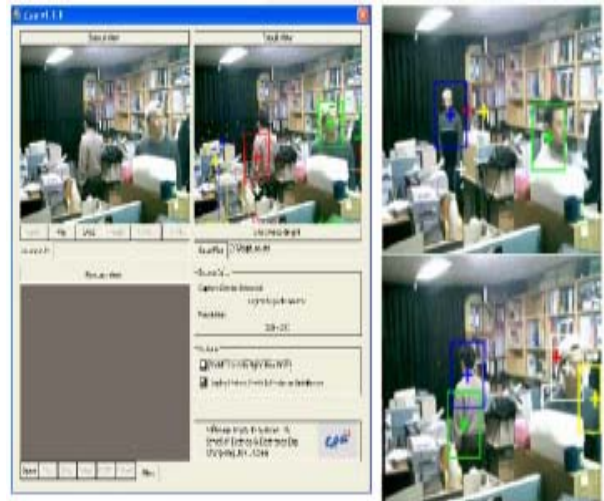

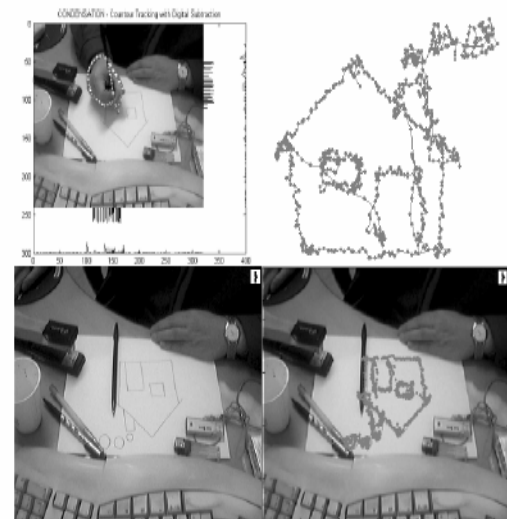Figure 6. Real-time particle filter for multiple objects

Figure 7. Picture Drawing Results with Particle Filters of Multiple Observation Models (N=200)

REFERENCES

[1] J.M. Hammersley and K.W. Morton, "Poor man's monte carlo," J. of the Royal Statistical Society B, vol. 16, pp.23-38, 1954.

[2] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," IEEE Proc. F, vol. 140, no.2, pp. 107-113, 1993.

[3] M. Isard and A. Blake, "Contour tracking by stochastic propagation of conditional density," in Proc. 4th European Conf. on Computer Vision (ECCV), pp. 343-356, April 1996

[4] M. Isard and A. Blake, "Condensation-conditional density propagation for visual tracking," Int. J. Computer Vision, vol 29, no. 1, pp. 5-28, 1998.

[5] A. Blake and M. Isard, Active Contours, London: SpringerVerlag, 1998.

[6] A. Popoulis, Probability and Statistics, New York: PrenticeHall, Inc., 1990.

[7] M. Isard and A. Blake, "Icondensation: Unifying low-level and high-level tracking in a stochastic framework," in Proc. 5th European Conf. on Computer Vision (ECCV), pp. 893-908, April 1998.

[8] A. Blake, M. Isard, and D. Reynard, "Learning to track the visual motion of contours," J. Artificial Intelligence, vol 78, pp. 101-134, 1995.

[9] S.J. Julier and J.K. Uhlmann, "A new extension of kalman filter to linear system," in Proc. Aerosense: 11th Int. Symp. On Aerospace/Defense Sensing, Simulation, and Control, Orlando, FL, pp. 182-193, 1997.

[10] J.M. Jurada, Introduction to Artificial Neural Systems, New York: West Pub.Co., 1992

[11] R. Malladi, J. A. Sethian and B. C. Vemuri, "Shape Modeling with Front Propagation: A Level Set Approach", IEEE Trans. On Pattern Analysis and Machine Intelligence, vol.17,no. 2,pp.158-175,Feb.1995

[12] J. MacCormick and A. Blake, "Partitioned sampling, articulated objects and interface-quality hand tracking," in Proc. 7th European Conf. on Computer Vision (ECCV), April 2000.

[13] Freedman, D and Tao Zhang "Active contours for tracking distributions", IEEE Trans. on Image Processing, vol. 13, no. 4, PP. 518 - 526, 2004.

[14] C. Gentile, O. Camps and M. Sznaier, "Segmentation for robust tracking in the presence of severe occlusion", IEEE Trans. on Image Processing, vol. 13, no. 2, pp. 166 - 178, 2004.

[15] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson and P.-J. Nordlund, "Particle filters for positioning, navigation, and tracking", IEEE Trans. on Signal Processing, vol. 50, no. 2, pp. 425 - 437, Feb. 2002.

[16] C. Hue, J.-P. Le Cadre and P. Perez, "Tracking multiple objects with particle filtering", IEEE Trans. on Aerospace and Electronic Systems, vol. 38, no. 3, pp. 791 - 812, 2002.