Journal of Management Science and Statistical Decision
Vol.7 No1.(2010) p35-40

# Particle Filter Optimized by CamShift for Scale Adaptive Face Tracking

Xinmei Li, Yi Cao*
Computer School, Northeast Normal University
Changchun, China
{caoyi370, lixm851}@nenu.edu.cn

Jun Kong, Jin Zhang, Danni Yang
Computer School, Northeast Normal University
Key Laboratory for Applied Statistics of MOE
Changchun, China

***Abstract*—In this context, a novel algorithm is proposed for scale adaptive face tracking. Mean-Shift process is applied to Particle Filter (PF) framework, where PF is optimized by Mean-Shift procedure due to its property of accurate tracking and fast convergence. And the accuracy position determination makes window size computation possible in the proposed method. On the other hand, the algorithm maintains multi-hypothesis tracking, so it is still robust. Additionally, an effect mechanism is proposed for checking tracking results, which provides useful information for next frame. In this paper, we have described and evaluated the performance of the proposed algorithm. The experimental results verify its efficiency and robustness for face tracking.**

***Keywords-face tracking; particle filter; Mean-Shift; histogram; scale adaptive***

## I. INTRODUCTION

Human face tracking (HFT) is one of several technologies useful in vision-based interaction (VBI), which is one of several technologies useful in the broader area of perceptual user interfaces (PUI)[7].Robust face tracking is a prerequisite for face analysis and recognition. However, it is a challenging issue in computer vision that tacking objects efficiently and robustly in complex environment. Nowadays, there are two main categories algorithms for visual tracking. The first category is probabilistic method, such as CONDESATION [2], Particle Filter [3], and Monte Carlo tracking [4], which is multi-hypothesis tracking method. Due to the particle filter's property of non-Gaussian, non-linear assumption and multi-hypothesis, the tracker is robust and the method can deal with tasks of background clutters, partial occlusion and complete occlusion for several frames. However, it has disadvantage of huge computational cost. The second category is deterministic method. The representative method is Mean-Shift algorithm [1], which is a non-parametric method of finding the local maximum of possibility gradient. As Mean-Shift keeps single hypothesis, it is computationally efficient. Also it is usually more accurate than that of PF algorithm. But it may run into trouble when similar objects are presented in background or when complete occlusion occurs [10]. So it becomes an interesting task that how to integrate the two algorithms successfully to apply to tracking problem. The main of our work is proposing a novel method combining the two algorithms and applying it to face tracking successfully.

Additionally, an efficient mechanism is proposed for checking tracking result. If the tracking is failure, the target should be rechecked and the particles should be initialized again. So the mechanism provides important information for tracking in the next frame. As verified by several videos, the experimental result shows accuracy and robustness for tracking.

In this paper, Mean-Shift process is applied to PF framework. The PF is optimized by Mean-Shift procedure due to its property of accuracy and fast convergence. And the accuracy posterior probability, which is the target face position in this work, makes window size computation possible. It also benefits resample stage. On the other hand, the proposal method keeps multi-hypothesis, so it is still robust for tracking.

We organize the rest of the paper as follows. Section 2 presents PF framework. The proposed method is described in detail in Section3. The experiments focus on robustness, precision and computational cost respectively in Section4. Finally, Section 5 illustrates the conclusion.

## II. PARTICLE FILTER

Sequential Monte Carlo algorithms (also called particle filters) are a special kind of filters in which theoretical distributions on the state-space are approximated by simulated random measures (also called particles)[7]. The state and the observation in the *t*-th frame are denoted as $x_t$ and $z_t$ respectively. Then the posterior probability $p(x_t|z_t)$ can be computed by using the Bayes theorem as follows:

$$p(x_t|z_t) \propto p(z_t|x_t) \int p(x_t|x_{t-1})p(x_{t-1}|z_{t-1})dx_{t-1} \tag{1}$$

In a particle filter, $p(x_t|z_t)$ is estimated by using the set of weighted particle $\{(x_t^0, \pi_t^0), \cdots, (x_t^N, \pi_t^N)\}$ . Each particle $i$ stores a system state $X_t^i$ and a quality measure $\pi_t^i$ called weight corresponding to the state $X_t^i$ at time t. Since the probability is represented by the discrete particles, it can represent an arbitrary non-linear distribution.

Fig. 1 depicted the scheme of particle filter and it contains four steps. The posterior probability $p(x_t \mid z_t)$ is estimated recursively as follows:

*Corresponding author.
E-mail addresses:Caoyi370@nenu.edu.cn
Telephone number: +86 13500810143
Fax number: +86 431 85696533

Step 1: The state $s_{t-1}$ is approximated by $N$ particles with the weights $N^{-1}$.
Step 2: By the system observation, we obtain each particle's weight measuring the quality of particle accord with the matter of fact. Then the posterior probability $p(x_t \mid z_t)$ is computed.
Step 3: Update the samples by resample process. Taking into account that particles with larger weight values can be chosen several times and the particles with lower values are replaced.
Step 4: The states $s_t$ of the particles are propagated by transition probability $p(x_t \mid x_{t-1} = s_{t-1})$.

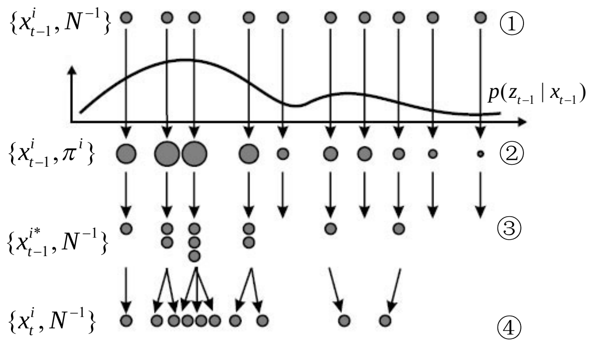



Figure 1. Particle filter scheme.

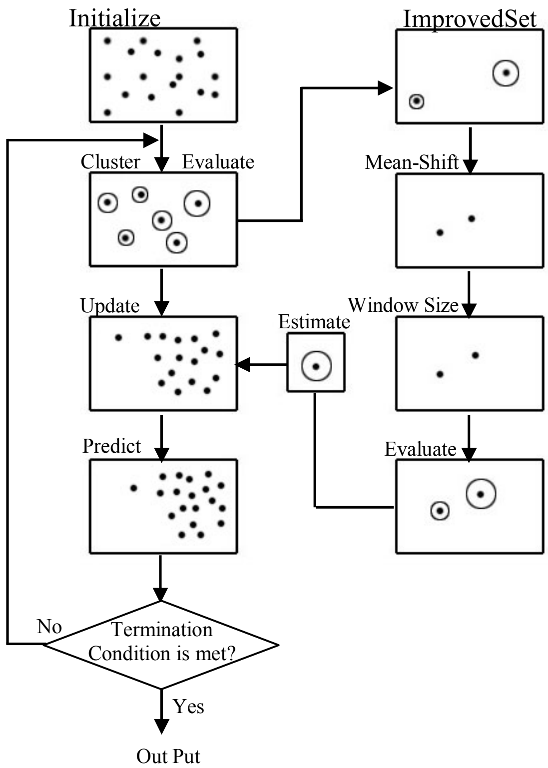



Figure 2. The framework of the proposed approach.

## III. OUR PROPOSED MENTHOD

In this paper a novel method is proposed which applies the Mean-Shift process to the PF framework. By this method the computational cost is reduced since we select only several particles involved in the computation of the posterior probability. However, it is still robust because the algorithm maintains the PF multi-hypothesis properties. Moreover, the tracking result is more accuracy due to Mean-Shift procedure. Here, the Mean-Shift process is used as optimization strategy. Fig. 2 presents a graphical template of the proposed algorithm. The detail of every stage is described as follows:

### A. Initialization

It is color-based tracking in this paper. RGB video images are converted into the HSV color space so that it is robust against illumination variations. And the hue channel is interested only. Accuracy target face is got as a model from the first frame of video sequence. Then when tracking begins, prior at time $t_0$ contains model face region and state $s_0$ denoted as $M(a,b),(a=1\cdots m, b=1\cdots n)$, $X_0=(x_0,y_0,c_0,d_0)$ respectively.

The target face is modeled by a state vector:

$$P=(x,y,c,d)^T \tag{2}$$

where $(x,y)$ is the position of the target face. Tracking window's size is represented by $(c,d)$. Particles are randomly selected from uniform distribution.

To use Mean-Shift technology, there are two works must be done. Firstly, we need to calculate the histogram $H$ of the region $M(a,b),(a=1\cdots m, b=1\cdots n)$, and then a reference color histogram $H$ is got:

$$H=\{h_u; u=1\cdots 256\} \tag{3}$$

where

$$h_u = C\sum_{a=1,b=1}^{m,n}\delta(b(x_{ab})-u) \tag{4}$$

$\delta$ is the Kronecker delta function, b is the mapping function which associates with the pixel at location $x_{ab}$, the index $b(x_{ab})$ of the histogram bin corresponding to the color of that pixel . $C$ is the normalization constant derived by imposing the condition $\sum_{u=1}^{256} h_u = 1$. Here, another histogram $H^*$ is also given for stage F.

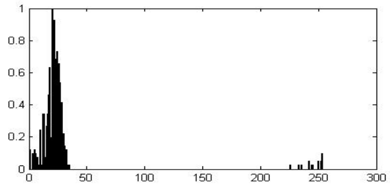
$$H^*=\{h'_u; u=1\cdots 256\} \tag{5}$$

$$h'_u=\begin{cases}0, & h_u<\alpha \\ f(h_u), & h_u\ge\alpha\end{cases} \tag{6}$$

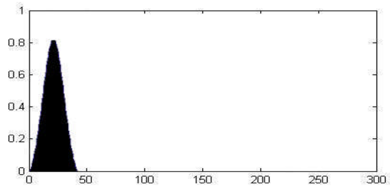
Where

$$f(x) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{7}$$

Fig. 3 describes the process how $H^*$ is generated:




(a) Histogram of $H$




(b) Histogram of $H^*$

Figure 3. Histogram of $H$ and $H^*$

Secondly, a color probabilistic map $P$, which is the same size with frame, is calculated by the histogram $H^*$ back projection. First we calculate the color histogram of the target and stored in a look-up table. Then while a new frame wants to update, the table is looked up for each pixel's color, and a probability value is assigned to each pixel. Consequently, a probabilistic distribution map is got. Mean-Shift procedure can be employed to find the nearby dominant distribution peak afterwards. The probabilistic map is updated every frame before applied Mean-Shift algorithm, and for the purpose of less computational load the probabilistic map is only updated in smaller region but not all of the frame. The size of the region is on the basis of the speed and size of the target.

*B. Clustering and Evaluation*

Particles are clustered by k-mean clustering. Every class is denoted by a particle which is the mean value of the all particles contained in this class. Then the clustering result consists of ClusterSet. We evaluate every particle in ClusterSet according to follow formula:

$$w_t^k \propto \Sigma_{(x,y)\in S(x^k, y^k)} f(p_t(x,y)-\alpha) \tag{8}$$

$f$ and $\alpha$ are same as formula(6),(7), $p_t(x,y)$ is the probability value in probabilistic map $P$ at location $(x,y)$ at time $t$. $S(x^k, y^k)$ is the 8-neighborhood coordinate set as depicted in Fig. 4. $w_t^k$ is the weight of particle $k$ in the CulsterSet, the larger is $w_t^k$ the higher quality of that particle.

| | | |
|---|---|---|
| 0.8 | 0.4 | 0.3 |
| 0.5 | 0.6 | 0.4 |
| 0.9 | 0.3 | 0.2 |

(a)

→

| | | |
|---|---|---|
| 1 | 0 | 0 |
| 1 | 1 | 0 |
| 1 | 0 | 0 |

(b)

Figure 4. The values of (a) are 8-neighborhood coordinate set with the center $p(x,y) = 0.6$ in probabilistic $P$, the values of (b) are the results done with function $f$ which the threshold $\alpha = 0.5$.

*C. Selecting Particles for Mean-Shift*

To optimize the PF algorithm, $N^i$ particles are selected for Mean-Shift. We don't need more particles because particles prone to moving to their neighboring local maxima actively after mean shift analysis. To maintain a trade-off between quality and diversity, $N^i/2$ best particles from ClusterSet are selected to constitute as set A and $N^i/2$ diverse ones consist of set B. Set A and set B both constitute as ImprovedSet. $\forall particle_i \in A$, the diverse particles in set B are selected according to the following formulas: .

$$particle_j = \underset{i\in A, j\in B}{\arg} \max(-e^{\frac{(p_i - p_j)^2}{2}}) \tag{9}$$

Where $p_i$ and $p_j$ denoted the position of the particles in Set A and B respectively. Hence, the diverse particles, which are farthest from the best quality particles in Set A, are got.

*D. Optimization Using Mean-Shift Procedure*

Mean-Shift is applied to due to its property of accuracy and fast convergence. And the accuracy posterior probability, which is the target face position in this work, makes window size computation possible. It also benefits resample stage. So it can optimize particle filter. And the optimization makes reduce the quantity of the particles, therefor the computational load is also reduced. Each particle $\{(x_t^k, y_t^k) \mid k = 1 \cdots N^i\}$ in the ImprovedSet is applied for Mean-Shift according to the following formulas:

$$x_t^k = \frac{M01}{M00} = \frac{\Sigma_{i,j} x_i p_t(x_i, y_j)}{\Sigma_{i,j} p_t(x_i, y_j)} \tag{10}$$

$$y_t^k = \frac{M10}{M00} = \frac{\Sigma_{i,j} y_j p_t(x_i, y_j)}{\Sigma_{i,j} p_t(x_i, y_j)} \tag{11}$$

where $p_t(x_i, y_j)$ is the probability value at location $(x_i, y_j)$ at time $t$. Accuracy position of the target is obtained through finite iteration by Mean-Shift algorithm. Two time iteration is satisfied in this paper.

### *E. Window Size Determinization*

With the prior of target face's geometrical structure, we consider that the target face approximates to ellipse. So the size of target face is modeled as ellipse. We determine the size of target face by semimajor and semiminor of the modeled ellipse. Due to the property of the accuracy of Mean-Shift algorithm, the particles are apt to the center of target after Mean-Shift process. This makes us apply the ellipse to determine the size of the target. For every particle $(x_t^k, y_t^k)_{k=1\cdots N^i}$, we model the size of target face as $T_t^k(c,d)$, $c$ and $d$ respectively represent semimajor axis and semiminor axis of the ellipse. Meanwhile, a canny detector is used for the edge detection. An edge map $E_t$ is generated. We obtain semimajor axis and semiminor axis of the ellipse as following formulas:

$$(c_t^k, d_t^k) = \arg\max_{c\in[25,31],d\in[14,18]} (T_t^k(c,d) \wedge E_t) \tag{12}$$

### *F. Mechanism for cheching tracking result*

First, the weight of every particle in ImprovedSet is calculated bye the following formulas:

$$\pi_t^k = H_t^k \cdot H^* \tag{13}$$

Then the posterior probability of the current frame is computed by:

$$x_t = \sum_{k=1}^{N^i} \pi_t^k x_t^k \bigg/ \sum_{k=1}^{N^i} \pi_t^k \tag{14}$$

$$y_t = \sum_{k=1}^{N^i} \pi_t^k y_t^k \bigg/ \sum_{k=1}^{N^i} \pi_t^k \tag{15}$$

Once the posterior probability is got, the histogram $H_t$, corresponding to the posterior probability $p_t = \{x_t, y_t, c_t, d_t\}$, is calculated as in (3), and then a mechanism for checking tracking result is applied for determining if the tracking fails. The tracking result is determined by the fowling formulas:

$$L = \begin{cases} 1 & H_t \cdot H^* \geq \theta \\ 0 & H_t \cdot H^* < \theta \end{cases} \tag{16}$$

Where $\theta$ is a threshold, we can got the vale of $\theta$ from the experiment.

If the tracking fails, the target will be departed again for tracking. Fig. 5 is the processing chart according to the tracking result.

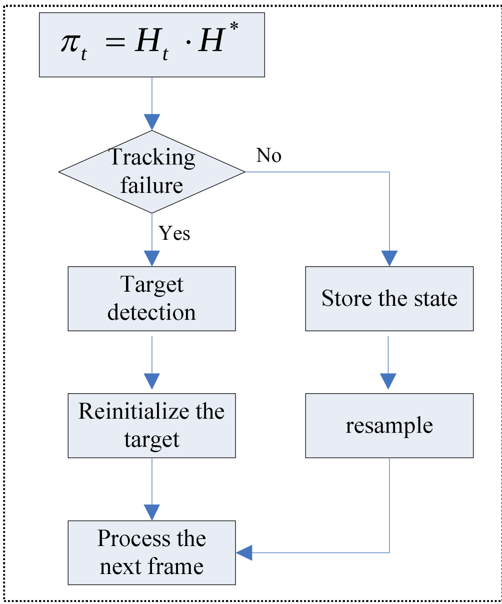



Figure 5. The processing chart according to the tracking result.

### *G. Update*

The SupportSet is updated by taking into account that the particles with larger value can be chosen several times. So the SupportSet is updated partly from the particles in ClusterSet, and partly from the posterior possibility.

### *H. Predict*

The updated particles are predicted the position of the next time. In this paper particles are propagated by first-order system model, $B$ is the propagation radius. The value of the propagation is revenant to the speed of the target.

$$X_t = X_{t-1} + Bw \tag{17}$$

Where $X_t$ and $X_{t-1}$ are respectively represent the particles position of time $t$ and $t-1$.

Finally, the algorithm of the proposed method is described as follows:

1) Depart the target face and initialize the target region with particles $\{(x_t^k, y_t^k) \mid k = 1\cdots N\}$, called as SupportSet. And the particles are selected randomly.

2) Cluster the particles in SupportSet and generated a new particle set called with ClusterSet $\{(x_t^k, y_t^k) \mid k = 1\cdots N^c\}$.

3) Select diverse particles from ClusteringSet for Mean-Shift. The selected particle set $\{(x_t^k, y_t^k) \mid k = 1\cdots N^i\}$ called as ImprovedSet. After Mean-Shift process, the particles are apt to at the center of the target duo to the accuracy property of Mean-Shift algorithm.

4) For every particle in ImprovedSet, the possible size of target face is found according to (12).

5) The weight of particles is calculated by (13), and the posterior possibility is computed as a weighted sum of the particles. Then the mechanism for checking tracking result is

apllied in the tracking system. If the tracking is failure, go to setp 1).

6) Update the particles from the posterior possibility and ClusterSet partly. Sequently the particles of $t+1$ are predited according to (17).

7) Step 2)- step 6) are followed continuously until tracking process ends.

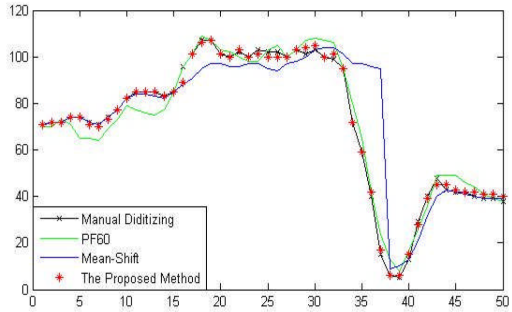
## IV. EXPERIMENT

To demonstrate the effectiveness of our proposed algorithm, it was applied to face tracking problem under partial occlusion and target face bothered by another face because it is a challenging work in face tracking. Many videos are used in the experiments and the results have verified the validity, accurate tracking and less computational load of our proposed method. Here the representative results of the experiment are shown. One experiment is face tracking under partial occlusion; the other is face tracking bothered by another face. The all algorithm were implemented by the same programmer, language programming (matlab) and platform. This section is devoted to show the obtained results.

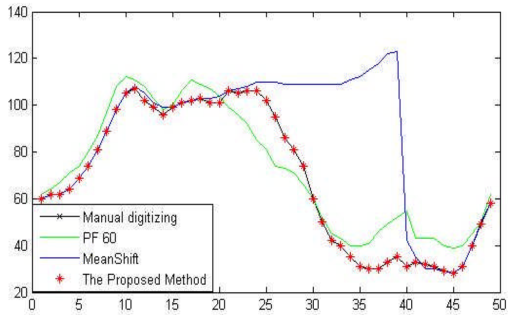
We have compared the performance of the proposed algorithm against CONDENSATION (Particle Filter) [2], Mean-Shift [1]. As a measurement of the algorithms performance, we concerned the robustness, precision and computational load. The results have been showed in the following.

- Robustness: we evaluate the robustness according to the number of frames which are failure in tracking of all frames. As depicted in Table I, It shows that the Mean-Shift algorithm is easy to lose, however, particle filter algorithm and our proposed method is robust. It also shows in Fig. 7 that the tracker is lost by Mean-Shift algorithm in frame 36 and frame 37 in video Cubicle but the tracker is robust by our proposed method.

- Precision: comparison of accuracy is experiment with three algorithms: particle filter algorithm with 60 particles, Mean-Shift and our proposed method. Here the location of the target face is given by manual digitizing, which is used as benchmark. In Fig. 6 the x-coordinate and y-coordinate represents the position of the target face. It shows that particle filter is robust but it is not accuracy, Mean-Shift is accuracy for tracking, but it is easy to lose. However, the proposed method is not only robust but also accuracy.




(a) The tracking result of the Cubicle video sequence. The coordinate denotes the position of the tracker.




(b) The tracking result of Djb video sequence The coordinate denotes the position of the tracker.

Figure 6. Comparition of the precision.

Fig. 8 also shows that the proposed method is more accurate than the particle filter algorithm. Frame 3, 30, 43, 48 of video Djb demonstrated it. By comparison, it presents clearly that our proposed method can track not only robust but also accurate.

- Computational load: The most expensive operation in the standard CONDENSATION algorithm is the evaluation of the likelihood function [5]. It is less computational load in the proposed method than standard CONDENSATION algorithm because that only $N^i$ particles need to be evaluated of all $N$ particles. To verify the efficiency achieved by our proposed method, we compare the time consumption of the three algorithms. It is shown that it is the fastest in tracking using Mean-Shift algorithm, and slow in particle filter.

TABLE I. COMPARISON OF THE ROBUSTNESS

| **Video sequences** | **Algorithm** | | |
|---|---|---|---|
| | ***Our proposed method*** | ***Particle filter*** | ***Mean-Shift*** |
| Cubicle | 0/51 | 0/51 | 4/51 |
| djb | 0/50 | 0/50 | 2/50 |
| sb | 12/500 | 12/500 | 43/500 |

2/50 means that 2 frames are failure in tracking of all 50 frames.

TABLE II. COMPARION OF TRACKING SPEED

| **Video sequences** | **Algorithm** | | |
|---|---|---|---|
| | ***Our proposed method*** | ***Particle filter*** | ***Mean-Shift*** |
| Cubicle | 4.6635 | 2.548 | 11.8439 |
| djb | 4.9410 | 2.712 | 12.0048 |
| sb | 45.357 | 21.463 | 12.824 |

The data in the table is tracking speed.4.6635 means the tracking speed is 4.6635 frames per second.

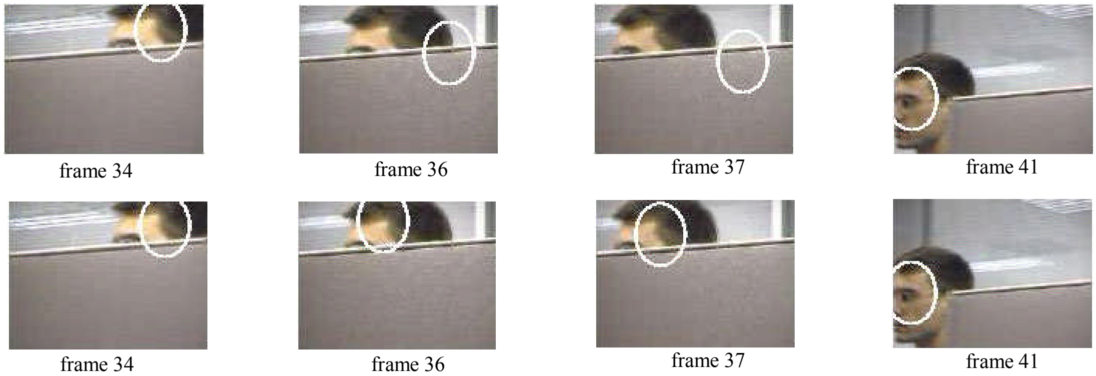



Figure 7. Comparision of robustness in tracking. The first line is the tracking result with Mean-Shift algorithm, and the second line is the tracking result with our proposed method. It shows that the tracker is failure with Mean-Shift method, but the tracker is robust with our proposed method.


Figure 8. Comparision of precision in tracking. The first line is the tracking result with PF(60) algorithm, and the second line is the tracking result with our proposed method. The experimet result shows that the proposed algorithm is more accurate than that of PF algorithm.

As expected, the combination maintains a trade-off between the tracking speed and robustness. The Mean-Shift process is contributed to the tracking accuracy, and the PF framework maintains the multi-hypothesis property so that the tracker is robust.

## V. CONCLUSION

In this paper we proposed a novel method that combines PF and Mean-Shift algorithm, which maintains the property of the PF's multi-hypothesis and Mean-Shift's accuracy. The combination makes the algorithm computational efficient and tracking robust, precise. Moreover, the accuracy posterior probability benefits resample stage.

## ACKNOWLEDGMENT

This work was supported by the Science Foundation for Young Teachers of Northeast Normal University (No. 20090301), the Key the Key Project of MOE (No.109052), the Training Fund of NENU'S Scientific Innovation Project (No. NENU-STC07018), the Fund of Jilin Provincial Science & Technology Department (No.20070322), and the NSFC (No. 10871037).

## REFERENCES

[1] Bradskin, G.R., "Computer vision face tracking for use in a perceptual user interface". IEEE Workshop Appl.Comput.Vision, 1998.214-219

[2] M. Isard and A. Blake, "ICONDENSATION: Unifying low-level and high-level tracking in a stochastic framework", *ECCV,* 1998.893-908

[3] Arulampalam,M.S.,Maskell,S.,Gordon,N.,Clapp,T. "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking." IEEE trans. Signal Process.2002

[4] Perez,P.,Hue,C.,Vermaak,J.,Gangnet,M. "Color-based probabilistic tracking." Eur. Conf. Comput. Vision. 2002. 661-675

[5] Deutscher, J., Blake, A., Reid, I. " Articulated body motion capture by annealed particle filtering." IEEE Conf. Comput Vision Pattern Recognition . 2000.126-133

[6] Shan, C., Wei, Y., Tan, T., & Ojardias, F. "Real time hand tracking by combining particle filtering and mean shift, Sixth IEEE International Conference on Automatic Face and Gesture Recognition." 2004.

[7] Carpenter, J., Clifford, P. "Building robust simulation based filters for evolving data sets." Technical Report, Department of Statistics, University of Oxford, Oxford, UK. 1999.

[8] Charay Lerdsudwichai, Mohamed Abdel-Mottaleb, and A-Nasser Ansari. "Tracking multiple people with recovery from partial and total occlusion." Patter Recognition. 2005.1059-1070

[9] J.J. Pantrigo, A. Sanchez, A. S. Montemayor, A. Duarte. "Multi-dimensional visual tracking using scatter search particle filter." Pattern Recognition Letters. 2008.1160-1174

[10] Wang, Z., Yang, X., Xu, Y., Yu, S. "CamShift guided particle filter for visual tracking." Pattern Recognition Letters . 2008.10

[11] Hong liu, Ze Yu, Hongbin Zha, Yuexian Zou, Liu Zhang. "Robust human tracking based on multi-cue integration and mean-shift." Pattern Recognition Letters 2009.827-837