# Scatter Search Particle Filter for 2D Real-Time Hands and Face Tracking

Juan José Pantrigo, Antonio S. Montemayor, Raúl Cabido

Universidad Rey Juan Carlos, c/ Tulipán s/n
28933 Móstoles, Spain
{juanjose.pantrigo, antonio.sanz}@urjc.es, rcabido@gmail.com

**Abstract.** This paper presents the scatter search particle filter (SSPF) algorithm and its application to real-time hands and face tracking. SSPF combines sequential Monte Carlo (particle filter) and combinatorial optimization (scatter search) methods. Hands and face are characterized using a skin-color model based on explicit RGB region definition. The hybrid SSPF approach enhances the performance of classical particle filter, reducing the required evaluations of the weighting function and increasing the quality of the estimated solution. The system operates on 320x240 live video in real-time.

## 1 Introduction

Automatic visual analysis of human motion is an active research topic in Computer Vision and its interest has been growing during the last decade [1]. Human-computer interaction is going to non-contact devices, using perceptual and multimodal user interfaces. That means the system allows the user to interact without physical contact, using voice or gesticulation capture [2]. The gesture tracking by monocular vision is an important task for the development of many systems. Recently, the field of computer vision has devoted considerable research effort to the detection and recognition of faces and hand gestures [3]. To locate the regions of interest, this kind of systems needs for previous multiple objects tracking procedure. Tracking human movement is a challenging task which strongly depends on the application [4].

Recent research in human motion analysis makes use of the particle filter (PF) framework. The particle filter algorithm enables the modelling of a stochastic process with an arbitrary probability density function (pdf), by approximating it numerically with a set of points called particles in a state-space process [5]. Unfortunately, particle filter fails in high dimensional estimation problems such as articulated objects [6] or multiple object tracking [7]. Hands and head tracking is an instance of multiple object tracking, involving a six-dimensional state space. In these kind of problems, particle filters may not be enough.

The main contribution of this paper is the development of a multiple object visual tracker based on the scatter search particle filter (SSPF) algorithm [8]. The scatter search (SS) metaheuristic proposed by Glover [9][10] is a population based metaheuristic applied in the context of combinatorial optimization problems. SSPF hybridizes both PF and SS frameworks in two different stages. In the PF stage, a

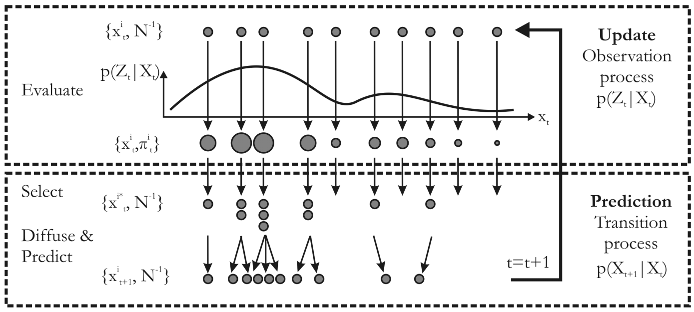



**Fig. 1.** Particle Filter scheme

particle set is propagated and updated to obtain a new particle set. In the SS stage, an elite set from the particle set is selected, and new solutions are obtained by combining them. SSPF significantly improves the performance of particle filters. This algorithm has been effectively applied to real-time hands and face tracking. Results show the system operates in real-time on a standard PC computer.

The rest of the paper is organized as follows. Next section presents the basic particle filter framework. Section 3 shows the scatter search procedure itself while section 4 describes the hybridization of scatter search and particle filter (called SSPF). Section 5 and 6 demonstrates its application to a 2D face and hands tracking problem. Finally, we present our conclusions and future works in section 7.

## 2 Particle Filter Framework

Sequential Monte Carlo algorithms (also called particle filters) are filters in which theoretical distributions on the state space are approximated by simulated random measures (called particles) [5]. The state-space model consists of two processes: (i) an observation process $p(\boldsymbol{Z}_t|\boldsymbol{X}_t)$, where $\boldsymbol{X}$ denotes the system state vector and $\boldsymbol{Z}$ is the observation vector, and (ii) a transition process $p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1})$. Assuming that observations $\{\boldsymbol{Z}_0, \boldsymbol{Z}_1,\ldots, \boldsymbol{Z}_t\}$ are known, the goal is to recursively estimate the posterior pdf $p(\boldsymbol{X}_t|\boldsymbol{Z}_t)$ and the new system state $\{\chi_0, \chi_1, \ldots, \chi_t\}$ at each time step. In Sequential Bayesian Modelling framework, posterior pdf is estimated in two stages:

(i) Evaluation: posterior pdf $p(\boldsymbol{X}_t|\boldsymbol{Z}_t)$ is computed at each time step by applying Bayes theorem, using the observation vector $\boldsymbol{Z}_t$:

$$p(\boldsymbol{X}_t \mid \boldsymbol{Z}_t) = \frac{p(\boldsymbol{Z}_t \mid \boldsymbol{X}_t)p(\boldsymbol{X}_t \mid \boldsymbol{Z}_{t-1})}{p(\boldsymbol{Z}_t)} \tag{1}$$

(ii) Prediction: the posterior pdf $p(\boldsymbol{X}_t|\boldsymbol{Z}_{t-1})$ is propagated at time step $t$ using the Chapman-Kolmogorov equation:

$$p(\boldsymbol{X}_t \mid \boldsymbol{Z}_{t-1}) = \int p(\boldsymbol{X}_t \mid \boldsymbol{X}_{t-1})p(\boldsymbol{X}_{t-1} \mid \boldsymbol{Z}_{t-1})d\boldsymbol{X}_{t-1} \tag{2}$$

A predefined system model is used to obtain an updated particle set.

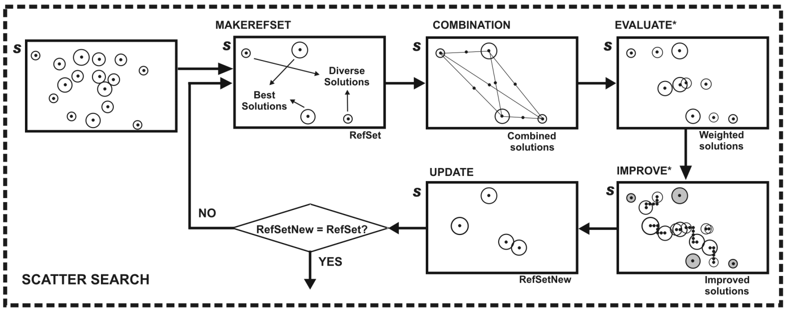



**Fig. 2**. Scatter Search scheme

In Fig. 1 an outline of the particle filter scheme is shown. The aim of the PF algorithm is the recursive estimation of the posterior pdf $p(\boldsymbol{X}_t|\boldsymbol{Z}_t)$, that constitutes the complete solution to the sequential estimation problem. This pdf is represented by a set of weighted particles $\{(\boldsymbol{x}_t^0, \pi_t^0)\ldots (\boldsymbol{x}_t^N, \pi_t^N)\}$.

PF starts by setting up an initial population $\boldsymbol{X}_0$ of $N$ particles using a known pdf. The measurement vector $\boldsymbol{Z}_t$ at time step $t$, is obtained from the system. Particle weights $\boldsymbol{\Pi}_t$ are computed using a weighting function. Weights are normalized and a new particle set $\boldsymbol{X}^*_t$ is selected. As particles with larger weight values can be chosen several times, a diffusion stage is applied to avoid the loss of diversity in $\boldsymbol{X}^*_t$. Finally, particle set at time step $t$+1, $\boldsymbol{X}_{t+1}$, is predicted using the motion model. Therefore, particle filters can be seen as algorithms handling the particles time evolution. Particles move according to the state model and are multiplied or died according to their weights or fitness values as determined by the likelihood function [5].

## 3 Scatter Search

Scatter search (SS) [9][10] is a population-based metaheuristic that provides unifying principles for recombining solutions based on generalized path construction in Euclidean spaces. In other words, SS systematically generates disperse set of points (solutions) from a chosen set of reference points throughout weighted combinations. This concept is introduced as the main mechanism to generate new trial points on lines jointing reference points. SS metaheuristic has been successfully applied to several hard combinatorial problems.

In Fig. 2 an outline of the SS is shown. SS procedure starts by choosing a subset of solutions (called *RefSet*) from a set *S* of initial feasible ones. The solutions in *RefSet* are the *h* best solutions and the *r* most diverse ones of *S*. Then, new solutions are generated by making combinations of subsets (pairs typically) from *RefSet*. The resulting solutions, called trial solutions, can be infeasible. In that case, repairing methods are used to transform these solutions into feasible ones. In order to improve the solution fitness, a local search from trial solutions is performed. SS ends when the new generated solutions do not improve the quality of the *RefSet*.

## 4 Scatter Search Particle Filter

Visual tracking of articulated motion is a complex task with high computational costs. Due to the dynamic nature of the problem, sequential estimation algorithms are usually applied to visual tracking. Unfortunately, particle filter fails in high dimensional estimation problems such as articulated objects or multiple object tracking. These problems can be seen as *dynamic optimization problems*. In our opinion, dynamic optimization problems deal with optimization and prediction tasks. This assumption is supported by the fact that the optimization method for changing conditions needs from adaptive strategies. On the other hand, in dynamic optimization problems it is not good enough to predict, and high quality solutions must be found.

Scatter search particle filter (SSPF) integrates both SS and PF frameworks in two different stages:

- In the PF stage, a particle set is propagated over the time and updated to obtain a new one. This stage is focused on the evolution in time of the best solutions

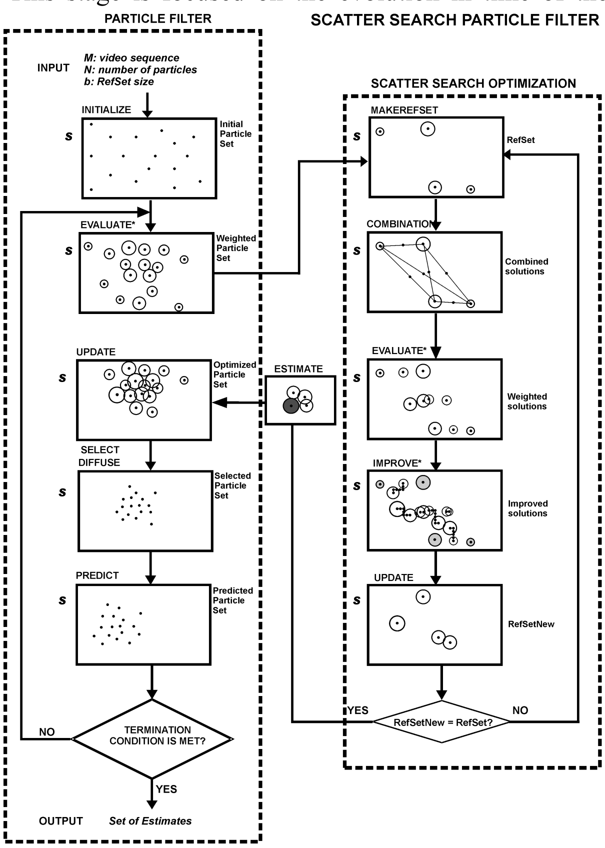



**Fig. 3**. Scatter Search Particle Filter scheme. Weight computation is required during EVALUATE and IMPROVE stages (*)

found in previous time steps. The aim for using PF is to avoid the loss of diversity in the solution set and to adapt the system to changing conditions.
- In the SS stage, a fixed number of solutions from the particle set are selected and combined to obtain better ones. This stage is devoted to improve the quality of a reference subset of good solutions in such a way that the final solution is also improved.

Fig. 3 shows a graphical template of the SSPF algorithm. Dashed lines separate the two main components in the SSPF scheme: PF and SS optimization, respectively. SPF starts with an initial population of $N$ particles drawn from a known pdf (Fig. 3: INITIALIZE). Each particle represents a possible solution of the problem. Particle weights are computed using a weighting function (Fig. 3: EVALUATE). SS stage is later applied to improve the best obtained solutions of the particle filter stage. A Reference Set (*RefSet*) is created selecting a subset of $b$ ($b<<N$) particles from the particle set (Fig. 3: MAKEREFSET). This subset is composed by the $b/2$ best solutions and the $b/2$ most diverse ones of the particle set. New solutions are generated and evaluated, by combining all possible pairs of particles in the *RefSet* (Fig. 3: COMBINE and EVALUATE). In order to improve the solution fitness, a local search from each new solution is performed (Fig. 3: IMPROVE). Worst solutions in the *RefSet* are replaced when there are better ones (Fig. 3: UPDATEREFSET). SS stage ends when new generated solutions *RefSetNew* do not improve the quality of the *RefSet*. Once the SS stage is finished, the "worst" particles in the particle set are replaced with the *RefSetNew* solutions (Fig. 3: INCLUDE). Then, a new population of particles is created by selecting the individuals from particle set with probabilities according to their weights (Fig. 3: SELECT and DIFFUSE). Finally, particles are projected into the next time step by following the update rule (Fig. 3: PREDICT).

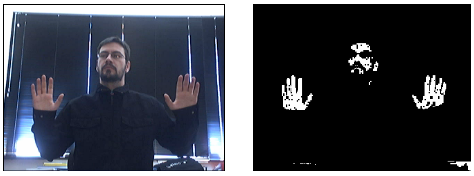
The SSPF leads the search process to a region of the search space in which it is highly probable to find new better solutions than the initial computed ones. SSPF search in state-space is not performed randomly like in a general particle filter. Unlike the PF algorithm, SSPF is time-adaptive since the number of evaluations of the weighting function changes in each time step.

## 5 SSPF Implementation to Hands and Face Tracking

Our proposed SSPF system is applied to determine the position of hands and face in 2D image sequences. Each particle in the particle set describes a possible solution for the tracking problem. The *particle structure* used in this work is:

$$\left[x_F, y_F, x_{RH}, y_{RH}, x_{LH}, y_{LH}, \dot{x}_F, \dot{y}_F, \dot{x}_{RH}, \dot{y}_{RH}, \dot{x}_{LH}, \dot{y}_{LH}\right] \tag{3}$$

where $x$ and $y$ are the spatial positions, and $\dot{x}$ represents the first derivative of magnitude (velocity). Subscripts $F$, $RH$ and $LH$ are referred to face, right and left hands respectively. The number of particles $N$ in the particle set $S$ is chosen to be 100. The *RefSet* is created by selecting the 6 best solutions in $S$.



(a) (b)

**Fig. 4** Observation model: (a) input image, (b) skin color detection

The *observation model* specifies the image features to be extracted. To construct the weighting function it is necessary to use adequate image features. In this work we have used a pixel based skin color detection method (see Fig. 4b). In this method an explicit region of RGB color space is defined as skin. A pixel (R,G,B) is classified as skin if [11]:

$$\begin{gathered}(R > 45) \,\&\, (G > 40) \,\&\, (B > 20) \,\&\, (\max(R,G,B) - \min(R,G,B) > 15) \\ \&\, (|R - G| > 15) \,\&\, (R > G) \,\&\, (R > B)\end{gathered} \tag{4}$$

The obvious advantage of this method is the simplicity of skin detection rules that leads to construction of a very rapid classifier. Particle weights are computed as the number of skin pixels belonging to three rectangular windows located in $[x_F, y_F, x_{RH}, y_{RH}, x_{LH}, y_{LH}]$.

*System model* describes how particles evolve over the time. The update rule used in this work is:

$$\begin{cases} x_{t+\Delta t} = x_t + \dot{x}_t \Delta t \\ \dot{x}_{t+\Delta t} = \dot{x}_t + G_x \Delta t \end{cases} \tag{5}$$

where $x$ represents some spatial variable, $\Delta t$ is the temporal step and $G_X$ is a random Gaussian variable with zero mean and normal deviation $\sigma_X$.

The *combination procedure* consists of exploring all possible combinations using partitions of solutions. Lets consider two solutions *(F1, R1, L1)* and *(F2, R2, L2)*, where *F*, *R* and *L* represent the kinematic state of face and left and right hands respectively. Six new solutions *(F1, R2, L2)*, *(F2, R1, L1)*, *(F2, R1, L2)*, *(F1, R2, L1)*, *(F2, R2, L1)* and *(F1, R1, L2)*, are obtained by combining parts of solutions. Finally, the best solution is chosen as result of the combination procedure.

A standard *local search* was employed as an *improvement stage* in the SS scheme. Given a solution, a neighborhood is explored until a new high quality solution is found. Then, this new solution replaces the old one and the procedure is repeated until no improvement is produced.

## 6 Experimental Results

To analyze the performance of the proposed algorithm, implementations of PF and SSPF have been developed. The experiments were evaluated in a 1.7 GHz Pentium 4, 256 MB RAM under Microsoft Windows XP Home SP2. Real-time video capture and

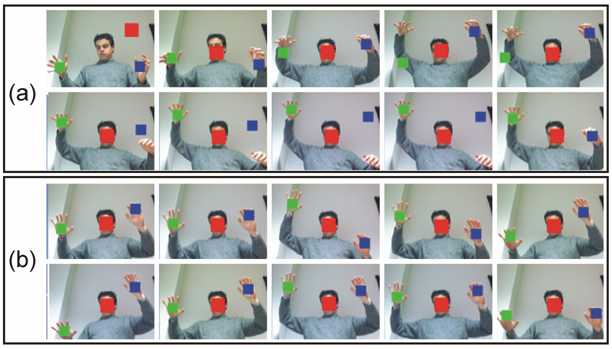
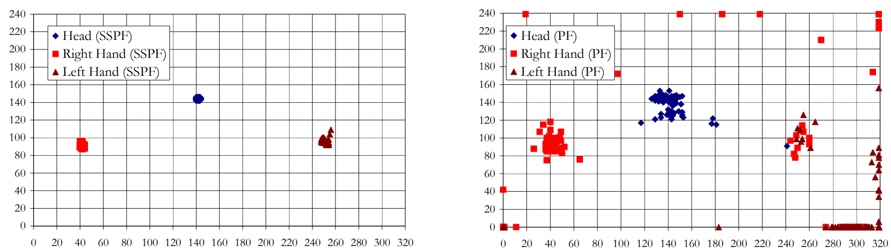



**Fig. 5**. Hands and face tracking using (a) PF and (b) SSPF.

processing have been done using Microsoft DirectShow API. In particular, the algorithms have been implemented as a DirectShow transform filter. The SSPF based system successfully operates on 320x240 live video in real time (28Hz). Fig. 5 shows the performance of the tracker using PF and SSPF. Fig. 6 shows the performance of SSPF vs. PF in the estimation of hands and head tracking during a video sequence. In this experiment the standard deviation using SSPF was dramatically lower than using PF as it can be seen in Table 1. These results demonstrate that the SSPF based approach increases the performance of general PF, improving the quality of the estimated solution and reducing the required evaluations of the weighting function. As results, the frame rate obtained using SSPF is higher than using PF, as it can be seen in Fig. 7.




(a) (b)

**Fig. 6** Estimation of hands and head position using (a) SSPF and 282 particles and (b) PF and 400 particles, during 133 video frames for the static pose shown in fig 4.

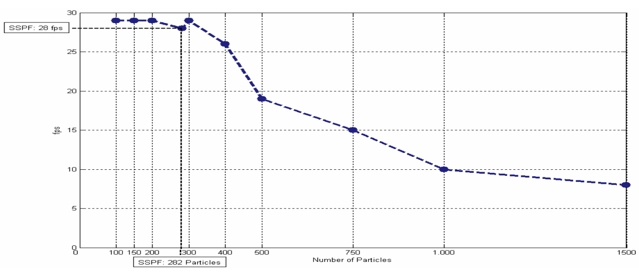



**Fig. 7** Frame rate using different number of particles in PF and using SSPF

**Table 1.** Standard deviation ($\sigma$) for SSPF and PF for $x$ and $y$ coordinates during 133 video frames for the static pose shown in fig 4.

| *Algorithm* | $N_{part}$ | *Head* | | *Right Hand* | | *Left Hand* | |
|---|---|---|---|---|---|---|---|
| | | $\sigma_x$ | $\sigma_y$ | $\sigma_x$ | $\sigma_y$ | $\sigma_x$ | $\sigma_y$ |
| *SSPF* | 282 | 1.1 | 1.1 | 1.2 | 1.7 | 1.8 | 2.1 |
| *PF* | 400 | 11.9 | 9.2 | 125.2 | 72.3 | 119.6 | 33.3 |
| | 1000 | 7.2 | 5.8 | 17.9 | 57.5 | 91.7 | 35.0 |

## 7 Conclusion

The main contribution of this work is the development of a 2D face and hands tracker based on the scatter search particle filter (SSPF) algorithm. SSPF hybridizes the scatter search metaheuristic and the particle filter framework to solve dynamic problems. Experimental results show how SSPF appreciably increases the performance of PF without losing quality in the estimation procedure. As a result, the tracker works in real time on a standard PC. Future works will deal with the management of occlusions and the correspondence problem to develop a vision based human-computer interface.

## References


1. Wang, L., Weiming, H., Tieniu, T.: Recent developments in human motion analysis. *Pattern Recognition* 36, 585–601 (2003)
2. Buades, J.M., Perales, F.J., Varona, J. Real time segmentation and tracking of face and hands in VR Applications. *AMDO LNCS 3179*, 259-268 (2004)
3. MacLean, J., et al. Fast Hand Gesture Recognition for Real-Time Teleconferencing Applications. *In proc. of the 2nd Int Workshop on Recognition, Analysis and Tracking of Faces and Gestures in Real-time Systems* (2001)
4. Argyros, A.A., Lourakis, M.I.A. Real time Tracking of Multiple Skin-Colored Objects with a Possibly Moving Camera. *In proc. of the European Conference on Computer Vision (ECCV'04)*, Springer-Verlag, vol. 3, 368-379 (2004)
5. Arulampalam, M., et al.: A Tutorial on Particle Filter for Online Nonlinear/Non-Gaussian Bayesian Tracking. IEEE Trans. On Signal Processing, V 50 (2): 174–188 (2002)
6. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. IEEE Conf. Computer Vision and Pattern Recognition, Vol. 2 (2000) 126–133
7. C. Hue, J.-P. Le Cadre, P. Pérez. A particle filter to track multiple objects. In *IEEE Workshop on Multi-Object Tracking*, 61-68 (2001)
8. Pantrigo, J.J., Duarte, A., Sánchez, A.: Scatter Search Particle Filter to Solve the Dynamic Travelling Salesman Problem. *EVOCOP 2005 LNCS* 3448, 177-189 (2005)
9. Glover, F.: A Template for Scatter Search and Path Relinking. *LNCS*, 1363, 1-53 (1997)
10. Laguna, M., Marti, R.: Scatter Search methodology and implementations in C. Kluwer Academic Publisher (2003)
11. Peer, P., Kovac, J., Solina, F. Human skin colour clustering for face detection. *In proc. of the InternationalConference on Computer as a Tool EUROCON* (2003)