Combining Particle Filter and Population-based Metaheuristics for Visual Articulated Motion Tracking

Juan José Pantrigo*, Ángel Sánchez*, Kostas Gianikellis+ and Antonio S. Montemayor*

* Dpto. de Informática, Estadística y Telemática. Universidad Rey Juan Carlos. c/ Tulipán s/n. 28933. Móstoles. Spain
+ Dpto. de Activ. Musical, Plástica y Corporal. Universidad de Extremadura. Av Universidad s/n. 10071. Cáceres. Spain

Abstract

Visual tracking of articulated motion is a complex task with high computational costs. Because of the fact that articulated objects are usually represented as a set of linked limbs, tracking is performed with the support of a model. Model-based tracking allows determining object pose in an effortless way and handling occlusions. However, the use of articulated models generates a multidimensional state-space and, therefore, the tracking becomes computationally very expensive or even infeasible.

Due to the dynamic nature of the problem, some sequential estimation algorithms like particle filters 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 are called *dynamic optimization problems*. Metaheuristics, which are high level general strategies for designing heuristics procedures, have emerged for solving many real world combinatorial problems as a way to efficiently and effectively exploring the problem search space. Path relinking (PR) and scatter search (SS) are evolutionary metaheuristics successfully applied to several hard optimization problems. PRPF and SSPF algorithms respectively hybridize both, particle filter and these two population-based metaheuristic schemes.

In this paper, We present and compare two different hybrid algorithms called Path Relinking Particle Filter (PRPF) and Scatter Search Particle Filter (SSPF), applied to 2D human motion tracking. Experimental results show the proposed algorithms increase the performance of standard particle filters.

Key Words: Image Sequence Analysis, Articulated Motion Tracking, Particle Filter, Optimization, Population-based Metaheuristics, Scatter Search, Path Relinking.

---

1 Introduction

Automatic visual analysis of human motion is an active research topic in Computer Vision and its interest has been growing in the last decade [20][15][6] [11]. Analysis and synthesis of human motion has numerous applications. In Visual Surveillance, gait recognition has been used for controlling the access of persons to restricted areas [20]. In Advanced User Interfaces, visual analysis of human movement is applied in detecting human presence and interpreting human behaviour [20]. Human motion analysis in Medicine can be employed to characterize and diagnose certain types of disorders [11]. Finally, visual analysis of human movement is

---

also used in Biomechanics, studying human body behavior subject to mechanical loads in three main areas: medical, sports and occupational.

Human body is usually represented as a set of limbs linked one to each other at joints [18]. Most studies in human motion analysis are based on articulated models that properly describe the human body [15][18][10][19]. Model-based tracking allows extracting body posture in an effortless way and handling occlusions.

2D contour representation of human body is relevant in the extraction of the human body projection in the image plane. In this description, human body segments are similar to 2D ribbons or blobs. In the work by Ju [10] a cardboard people model was proposed. Human body segments were modelled by planar patches. Leung and Yang [13] used a 2D ribbons with U-shaped edge segments. Rohr [18] proposed a 2D motion model in which a set of analytically motion curves represented the postures.

One particular pose of the subject can be expressed as a single point in a state-space. In this multidimensional space each axis represents a degree of freedom (DOF) of a joint in the model. Thus, all possible solutions to the pose estimation problem are represented as points in this state-space. The goal of the model is to connect the state-space with the 2D image space. This is achieved by creating a set of synthetic model images and comparing them to measurements taken at each frame of the video sequence thus obtaining a similarity frame estimate. Low level features such as blobs (silhouette), edges (contours), colour and movement have been widely used in diverse approaches [15].

There are several methods for the comparison between synthetic data and frame measurements. A usual approach, given by a Kalman Filter, predicts just one state and estimates the difference between the synthetic data and the measurements data [15]. Another approach, given by a Particle Filter algorithm, predicts the most likely states using a multiple hypothesis framework. The Particle Filter (PF) algorithm, (also termed as Condensation algorithm) enables the modelling of a stochastic process with an arbitrary probability density function (pdf), by approximating it numerically with a set of points (particles) in a process state-space [21].

The problem with using an articulated model for human body representation is the high dimensionality of the state-space and the high computational effort it supposes [4]. Also, in the Condensation approach, the number of required particles grows with the size of the state-space, as demonstrated in [14]. To address this difficulty, several optimized PF algorithms have been proposed. They use different strategies to improve their performance. Deutscher [4][5] developed an algorithm termed Annealed Particle Filter (APF) for tracking people. This filter works well for full-body models with 30 DOFs. Partitioned Sampling (PS) [14] is a statistical approach to tackle hierarchical search problems. PS consists by dividing the state space into two or more partitions, and sequentially applying the stated dynamic model for each partition followed by a weighted re-sampling stage. Ning [16] use learned motion models and motion constraints integrated into a dynamic model to concentrate factored sampling in the areas of state-space with most posterior information.

Optimization problems consist of the search for a "best" configuration of a set of variables to achieve some goals. Metaheuristics are a kind of approximate general methods that can be applied to solve complex optimization problems. Metaheuristics try to combine basic heuristic methods in higher level frameworks aimed at efficiently and effectively exploring a search space [2].

Due to the dynamic nature of the problem, sequential estimation algorithms are usually applied to visual tracking. Unfortunately, particle filter are not effective in high dimensional estimation problems such as articulated objects or multiple object tracking. These problems can be seen as a sequence of optimization problems, and they are called *dynamic optimization problems*. In order to avoid the limitations to the particle filters, we propose a general framework to develop hybrid optimization algorithms which combine both sequential estimation algorithms and population-based metaheuristics.

In this paper we consider two different instances of this method: *Path Relinking Particle Filter* (PRPF) and *Scatter Search Particle Filter* (SSPF). These algorithms are inspired by the Path Relinking and the Scatter Search Metaheuristics proposed by Glover [9][7]. These algorithms hybridizes both Particle Filter (PF) and Population-based Metaheuristic (PBM) frameworks in two different stages. In the PF stage, a particle set is propagated and updated to obtain a new particle set. In PBM stage, an optimized subset (called RefSet) from the particle set is selected, according to quality and diversity criteria, and new solutions are constructed using
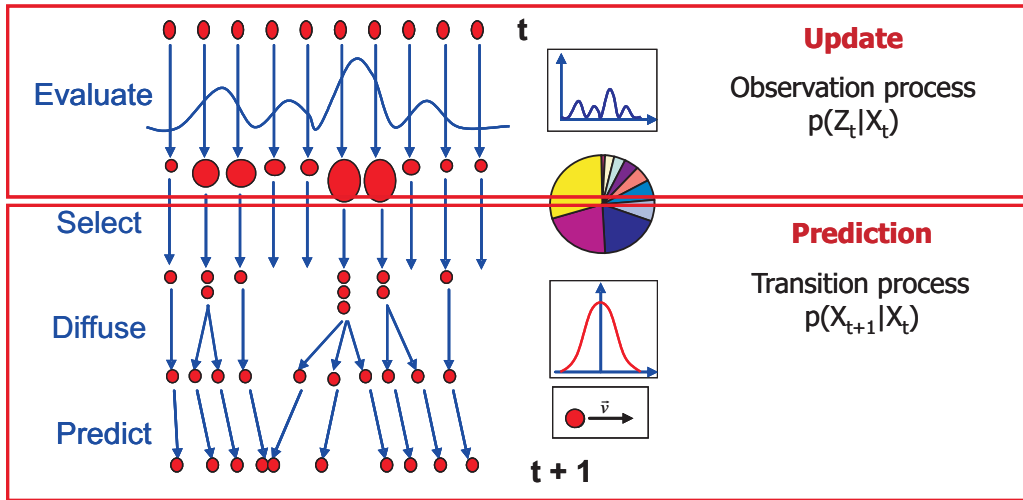
Figure 1: Particle Filter scheme

different combination methods.

We have applied the PRPF and SSPF algorithms to 2D human pose estimation in different movement tracking activities such as running and jumping. Experimental results show that the proposed algorithms increase the performance of standard particle filters by improving the quality of the estimate, adapting the computational load to problem constraints and reducing the number of required evaluations of the weighting function.

2   Particle Filters

Sequential Monte Carlo algorithms (also called Particle Filters) are a special class of filters in which theoretical distributions on the state-space are approximated by simulated random measures (also called particles) [3]. The state-space model consists of two processes: (i) an observation process $p(Z_t|X_t)$, where X denotes the system state vector and Z is the observation vector, and (ii) a transition process $p(X_t|X_{t-1})$). Assuming that observations $\{Z_0, Z_1, \ldots, Z_t\}$ are sequentially measured in time, the goal is the estimation the new system state $\{\chi_0, \chi_1, \ldots, \chi_t\}$ at each time step. In the framework of Sequential Bayesian Modelling, posterior pdf is estimated in two stages:

(i) Evaluation: posterior pdf $p(X_t|Z_t)$ is computed using the observation vector Z_t:

$$p(X_t|Z_t) = \frac{p(Z_t|X_t)p(X_t|Z_{t-1})}{p(Z_t)} \tag{1}$$

(ii) Prediction: the posterior pdf $p(X_t|Z_{t-1})$ is propagated at time step t using the Chapman-Kolmogorov equation:

$$p(X_t|Z_{t-1}) = \int p(X_t|X_{t-1})p(X_{t-1}|Z_{t-1})dX_{t-1} \tag{2}$$

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

In Figure 1 an outline of the Particle Filter scheme is shown. The aim of the PF algorithm is the recursive estimation of the posterior pdf $p(X_t|Z_t)$, that constitutes a complete solution to the sequential estimation problem. This pdf is represented by a set of weighted particles $\{(x_t^0, p_t^0), \ldots, (x_t^N, p_t^N)\}$, where the weights $p_t^n = p(Z_t|X_t = x_t^n)$ are normalized.

PF algorithm starts by setting up an initial population X_0 of N particles using a known pdf. The measurement vector Z_t at time step t, is obtained from the system and particle weights Π_t are computed using a fitness
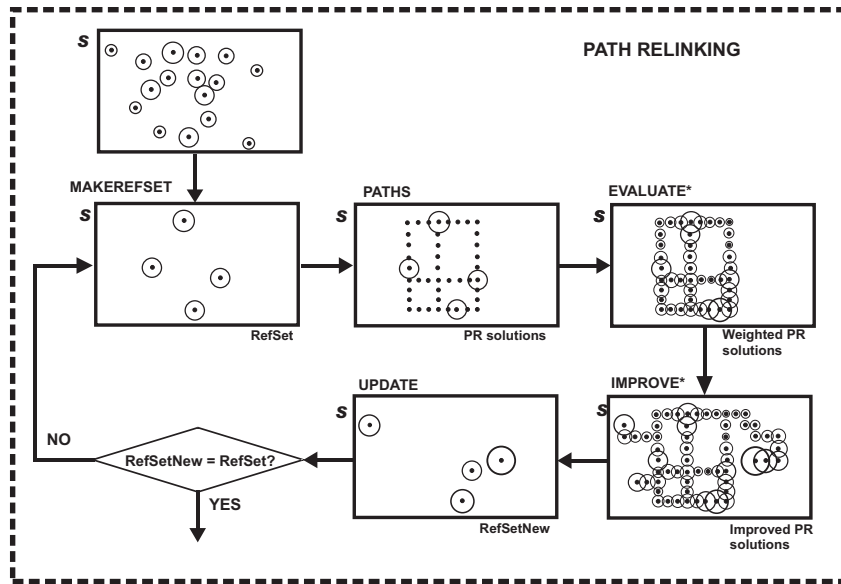
Figure 2: Path Relinking scheme

function. Weights are normalized and a new particle set 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 X_t^*. Finally, particle set at time step $t + 1$, X_{t+1}, is predicted using the motion model. A pseudocode of a general PF is detailed in [1][17].

Therefore, Particle Filters can be seen as algorithms handling the particles evolution. Particles in PF move according to the state model and are multiplied or died according to their weights or fitness values as determined by the pdf [3].

3   Population-based Metaheuristics

Metaheuristics are a kind of approximate algorithm which basically tries to combine basic heuristic methods in higher level frameworks aimed at efficiently exploring a search space [2]. Metaheuristics are applied successfully in optimization problems, which consist of the search for a "best" configuration of a set of variables to achieve some goals. Population-based metaheuristics (PBM) [2] are algorithms that works with a set of solution at the same time. Thus, this kind of methods perform search processes which describe the refinement of a set of solutions in the search space. This section is devoted to present the considered population-based metaheuristics.

3.1   Path Relinking

Path Relinking (PR) [9][7] is an evolutionary metaheuristic in the context of the combinatorial optimization problems. PR constructs new high quality solutions by combining other previous solutions based on the exploration of paths connecting them. To yield better solutions than the original ones, PR starts from a given set of elite candidates, called $RefSet$ (short for "Reference Set"). These solutions are selected through a search process and are ordered according to their corresponding qualitative values. New candidates are then generated, exploring trajectories that connect solutions in the $RefSet$. The metaheuristic starts with two of these solutions x' and x'', and it generates a path $x' = x(l), x(2), \ldots, x(r) = x''$ in the neighbourhood space that leads
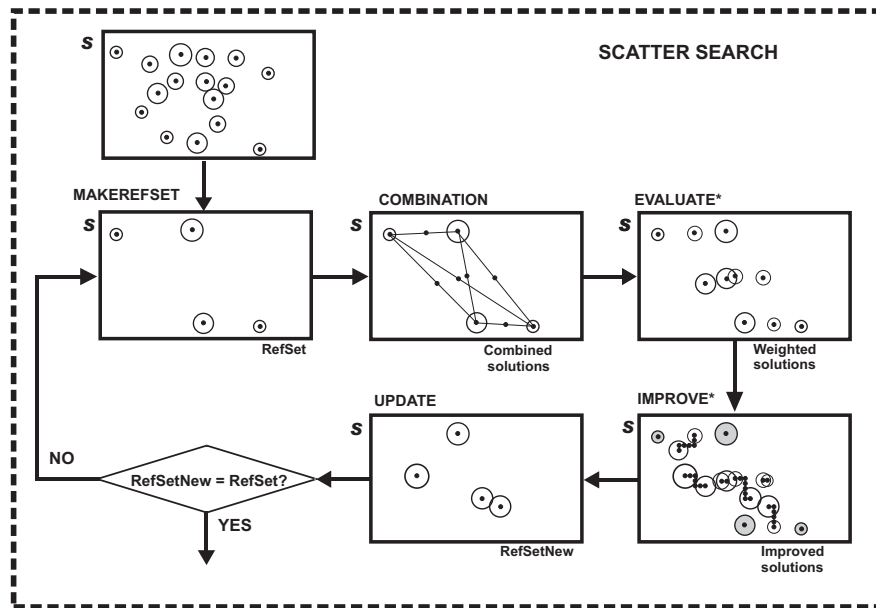
Figure 3: Scatter Search scheme

toward the new sequence. In order to produce better quality solutions, it is convenient to add a local search optimization phase. In Figure 2 an outline of the PR is shown.

3.2 Scatter Search

Scatter Search (SS) [8][12] 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 (never randomly) 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 solutions. SS metaheuristic has been successfully applied to several hard combinatorial problems. A recent review of this method can be found in [12].

In Figure 3 an outline of the SS is shown. SS procedure starts by choosing a solutions subset (called $RefSet$) from a set S of $PopSize = |S|$ initial feasible ones. The solutions in $RefSet$ are obtained by choosing the h best solutions and the r most diverse ones in S. Then, new solutions are generated by making combinations of solution 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 $RefSet$ quality.

4 Particle Filter and Population-based Metaheuristics Hybrid Algorithms

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 a sequence of optimization problems, and they are called *dynamic optimization problems*. In our opinion, dynamic optimization problems deals with optimization and prediction tasks. This assumption is supported by the fact that the optimization method for changing conditions needs from adaptive
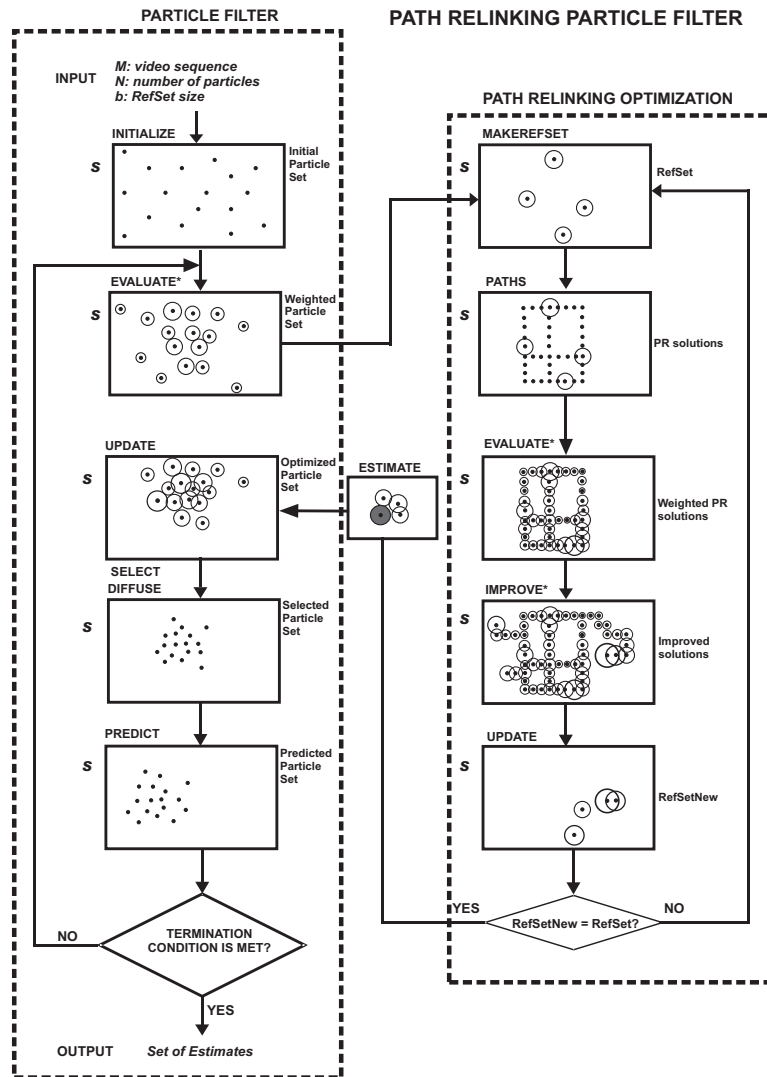
Figure 4: Path Relinking Search Particle Filter scheme. Weight computation is required during EVALUATE and IMPROVE stages (*)

strategies. On the other hand, in dynamic optimization problems it is not good enough to predict, and high quality solutions must be found.

Therefore, it could be not too appropriate to use optimization procedures in the prediction stage. Analogously, sequential estimation algorithms are well-suited in prediction stages, but they are not good enough for solving dynamic optimization problems. Then, dynamic optimization problems needs from both, optimization and prediction tasks. The key question is how to hybridize these two kinds of algorithms to obtain a new one which combines both techniques. In order to answer this question, two different hybrid algorithm called *Path Relinking Particle Filter* (PRPF) and *Scatter Search Particle Filter* (SSPF) are presented in this section.

4.1 Path Relinking Particle Filter

Path Relinking Particle Filter (PRPF) algorithm was introduced in [17] to be applied to estimation problems in sequential processes that can be expressed using the state-space model abstraction. PRPF integrates both Path Relinking (PR) and Particle Filter (PF) frameworks in two different stages:
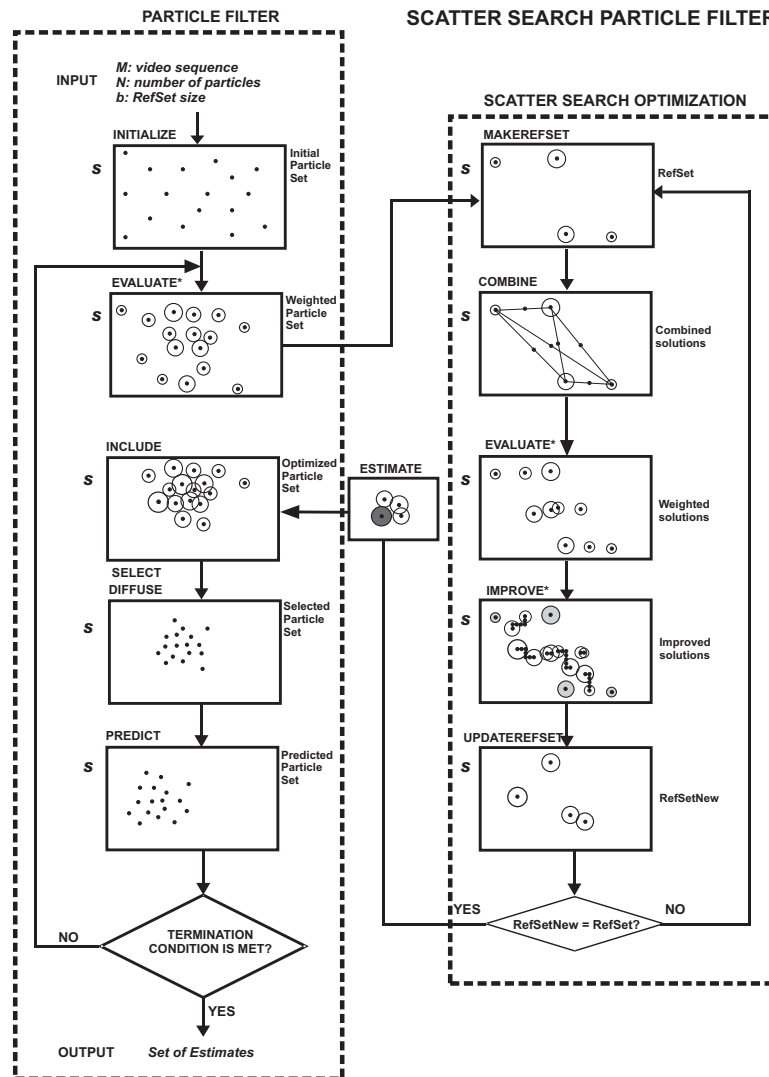
Figure 5: Scatter Search Particle Filter scheme. Weight computation is required during EVALUATE and IM-PROVE stages (*)

- In the *Particle Filter stage*, a particle (solution) set is propagated over the time and updated with measurements to obtain a new one. This stage is focused on the evolution of the best solutions found in previous time steps. The main aim for using PF is to avoid the loss of needed diversity in the solution set.

- In the *Path Relinking 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 set of good solutions in such a way that the final solution is also improved.

Figure 4 shows a graphical template of the PRPF method. Dashed lines separate the two main components in the PRPF scheme: PF and PR optimization, respectively. PRPF starts with an initial population of N particles drawn from a known pdf (Figure 4: INITIALIZE stage). Each particle represents a possible solution of the problem. Particle weights are computed using a weighting function and a measurement vector (Figure 4: EVALUATE stage). PR stage is later applied improving the best obtained solutions of the particle filter stage. A $RefSet$ is created selecting the b ($b << N$) best particles (Figure 4: MAKEREFSET stage). New solutions are generated and evaluated by exploring trajectories that connect all possible pairs of particles in the $RefSet$

(Figure 4: PATHS and EVALUATE stages). In order to improve the solution fitness, a local search from some of the generated solutions within the PR procedure is performed (Figure 4: IMPROVE stage). PR stage ends when the new generated solutions do not improve the quality of the $RefSet$. Once the PR stage is finished, the "worst" particles are replaced with the $RefSet$ solutions (Figure 4: UPDATE stage). Then, a new population of particles is created by selecting the individuals from the whole particle set with probabilities according to their weights (Figure 4: SELECT stage). To avoid the loss of diversity, a diffusion stage is applied to the particles of the new set (Figure 4: DIFFUSE stage). At the end, particles are projected into the next time step by making use of the update rule (Figure 4: PREDICT stage).

4.2   Scatter Search Particle Filter

The Scatter Search Particle Filter (SSPF) algorithm is introduced in this paper to be applied to dynamic optimization problems. SSPF integrates both Scatter Search (SS) and Particle Filter (PF) frameworks in two different stages.

- The *Particle Filter stage* proceed in the same manner as in the PRPF algorithm

- In the *Scatter Search 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 set of good solutions in such a way that the final solution is also improved.

Figure 5 shows a graphical template of the SSPF algorithm. Dashed lines separate PF and SS stages. PF stages works in the same way than in the PRPF (Figure 5: INITIALIZE, EVALUATE, INCLUDE, SELECT, DIFFUSE and PREDICT stages). SS stage is applied before the evaluation stage to improve the best obtained solutions of the particle filter. A $RefSet$ is created selecting a subset of b ($b << N$) particles from the particle set (Figure 5: MAKEREFSET stage). 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 $RefSet$ (Figure 5: COMBINE and EVALUATE stages). To improve the solution fitness, a local search from each new solution is performed (Figure 5: IMPROVE stage). Worst solutions in the $RefSet$ are replaced when there are better ones (Figure 5: UPDATEREFSET stage). 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 (Figure 5: INCLUDE stage) and subsequent filter stages are performed (Figure 5: SELECT, DIFFUSE and PREDICT stages).

4.3   PRPF and SSPF Main Features

The SSPF and PRPF algorithms are centered on a delimited region of the state-space in which it is highly probable to find new better solutions than the initial ones. PRPF increases the performance of general PF by improving the quality of the estimate, adapting computational load to constraints and reducing the number of required evaluations of the particle weighting function.

PF performs two tasks over the set $S(t)$ to obtain the solution set $S(t + 1)$: selecting the best solutions and predicting new solutions from the best ones. Firstly, the selection procedure selects particles with larger weight values more likely than those with lower weights. Secondly, PF performs a prediction procedure over these best solutions to obtain the set $S(t + 1)$. In this way, PF beats to problem changes by predicting the best solution time evolution. As results, solutions in $S(t + 1)$ will be closer to global optimum than another ones obtained randomly. On the other hand, a diffusion procedure is applied to the selected solutions to include diversity in the set $S(t + 1)$.

To summarize, the main advantages of the PRPF and SSPF hybrid algorithms are:

- Hybrid estimator quality is improved with respect to PF and the required number of evaluations for the weighting function is also reduced. This is due to the fact that PRPF and SSPF search is not performed randomly like in a general particle filter.

- Both hybrid algorithms are time-adaptive since the number of evaluations of the weighting function changes in each time step. If the initial solutions in the $RefSet$ are far away one from each other, paths connecting solutions become long enough, and the number of explored solutions increases.

- The number of individuals in the particle filter does not change during the algorithm execution. PRPF algorithm reduces the total required number of evaluations of the weighting function when increasing the number of total time steps.

Population-based Metaheuristics (PBM) and PF are related in such a way that when the PBM improves, the PF performance also improves and vice versa. PF allows parameter tuning in order to adjust the quality and the diversity of the set S, used by PBM. On the other hand PBM improves the quality of the particle set allowing the better estimation of the pdf, by including $RefSet$ solutions in the set S. This fact yields to a highly configurable algorithm. The main considered hybrid algorithm parameters are:

- The size of the particle set N is the number of particles in the particle set. There should be enough particles to support a set of diverse solutions, avoiding the loss of diversity in the particle set. Thus, N influences on the performance of the SS stage. The value of N depends on the problem instance complexity.

- The size of the reference set b is the number of solutions in the $RefSet$. A typical $RefSet$ size value recommended is $b = 10$ [12].

- The diffusion stage is applied to avoid the loss of diversity in S. It is performed by applying a random displacement with maximum amplitude A. This amplitude A is a measure of the diversity produced in the new particle set. Therefore, A influences the performance of the SS by tuning the diversity of the initial solution set, and hence, the diversity of the $RefSet$.

5   Models for Human Pose Estimation

Each one of the involved models in our framework is detailed in this section. A geometrical model is required to link solutions in the state-space with $2D$ image feature extraction. Observation and system models respectively define the observation and transition processes in the state-space model abstraction.
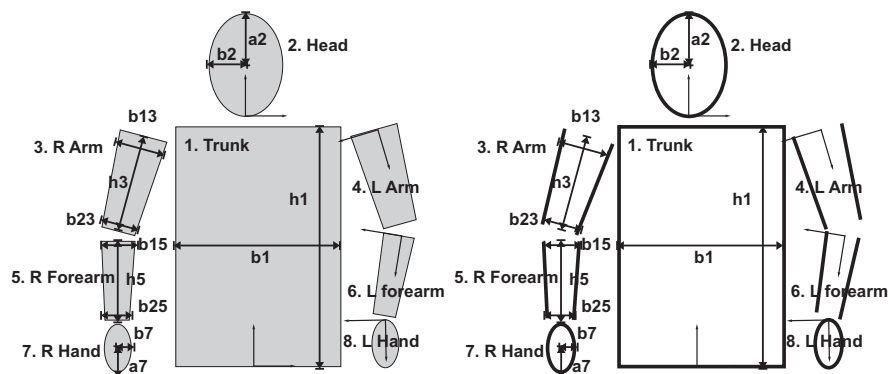

Figure 6: Proposed blob (left) and edge (right) configuration for human upper-body model
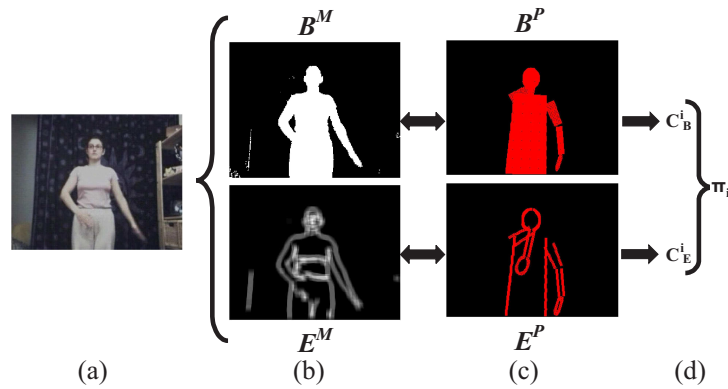
Figure 7: Observation process: (a) initial image, (b) feature extraction, (c) particle prediction and (d) particle weight computation

5.1 Geometrical Model

We use an a priori $2D$ geometrical model to represent the observed subject. It consists of a hierarchical set of articulated limbs. This model stores geometrical (time-independent) parameters describing the body components. Figure 6 illustrates the proposed blobs and edge models for upper-body tracking. As shown in the experiments section, this model can be easily extended to describe the whole human body.

	Trunk	Head	Right Arm	Right Forearm	Right Hand
Identifier	1	2	3	5	7
Shape	T	E	T	T	E
Level	1	2	2	3	4
Father	-	1	1	3	5
Size	$[h1, b1, b1]$	$[a2, b2]$	$[h1, b13, b23]$	$[h1, b15, b25]$	$[a7, b7]$
Position	-	$[0, h1 + \Delta]$	$[-b1/2, h1 - b12/2]$	$[0, h3]$	$[0, h5]$

Table 1: Limb properties in a human upper-body model

Body limbs are represented by a set of trapezium-shaped (trunk, arms, legs, and feet) and ellipse-shaped (head and hands) ribbons which are connected by joints. Size of trapeziums (T) is described by three parameters: one for the length and two for the axes. Size of ellipses (E) is described by two axes. Each limb is jointed with a father limb except trunk. Position and orientation of each body part is described in his father frame. The coordinate system for the body parts are aligned with the natural axes. The origin of a coordinate system is located at the point in which each limb is jointed with his father limb. The level of the limb is related to the distance from the body center, and it is useful to calculate position and orientation of body parts in the global reference system. Several examples of limb descriptions in the proposed model are shown in Table 1.

Particles store time-dependent values relating to limb positions, orientations and velocities. The state x_t^i of a particle (x_t^i, π_t^i) in an eight-limb model is described as:

$$[x_1, y_1, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8, \dot{x}_1, \dot{y}_1, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6, \dot{\theta}_7, \dot{\theta}_8] \quad (3)$$

where x and y are the spatial positions, θ_i is the i limb orientation in the father's system of reference and $\dot{x}$, $\dot{y}$ and $\dot{\theta}$ represents the first derivative of its corresponding variable. The goal of the geometrical model is to relate solutions in the multi-dimensional state-space with the 2D image features. Thus, the method predicts the pose of the model for the next frame and creates synthetic edge and blobs images. Note that these parameters are defined
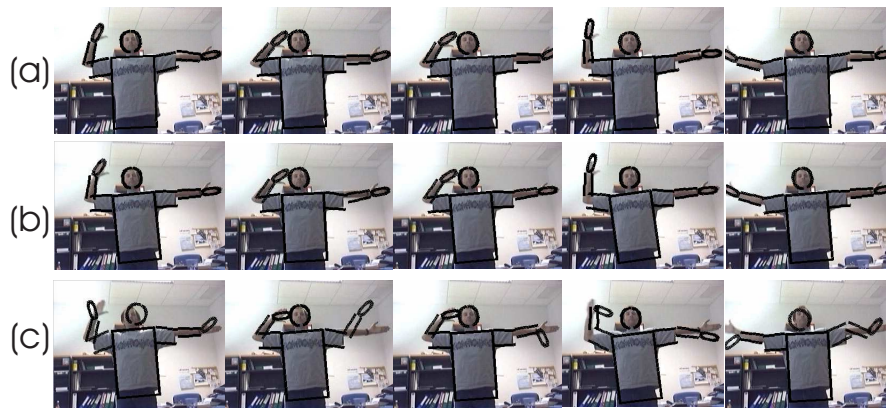
Figure 8: Visual model adjustment for a subject performing planar movements (frames 10, 20, 30, 40 and 50) using (a) PRPF, (b) SSPF and (c) one-layered PF

with respect to the camera view point. Features, those extracted from each frame in the video sequence and those predicted by the PRPF and SSPF algorithms, are compared in order to obtain a corresponding similarity measure. This similarity value is iteratively used to establish the weights of the different particles for the following frame during the tracking stage.
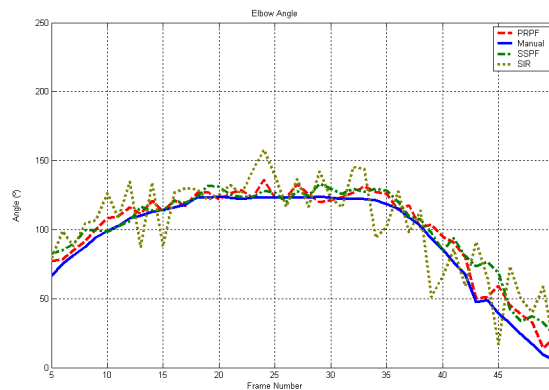

Figure 9: Right elbow angle estimation in frontal movement shown in Figure 8 using PRPF, SSPF, One-Layered PF and manual digitizing

5.2   Observation Model and Weighting Function

The observation model specifies the image features to be extracted. To construct the weighting function it is necessary to use adequate image features. In controlled environments, edges and silhouette are relatively easy to extract from both, the image and the geometrical model. Continuous edges extracted from a human image usually provide a good measure of visible body limbs. However, they are sensitive to noise. A region-based feature such as silhouette has the advantage over edges of being less sensitive to noise [15]. On the other hand, details may be lost in the extraction of silhouettes. In order to overcome these difficulties both a silhouette and an edge based model are used.

Figure 7 represents the observation process that leads to the particle weights computation. Continuous edges extracted from a human image usually provide a good measure of visible body limbs. A Canny edge method
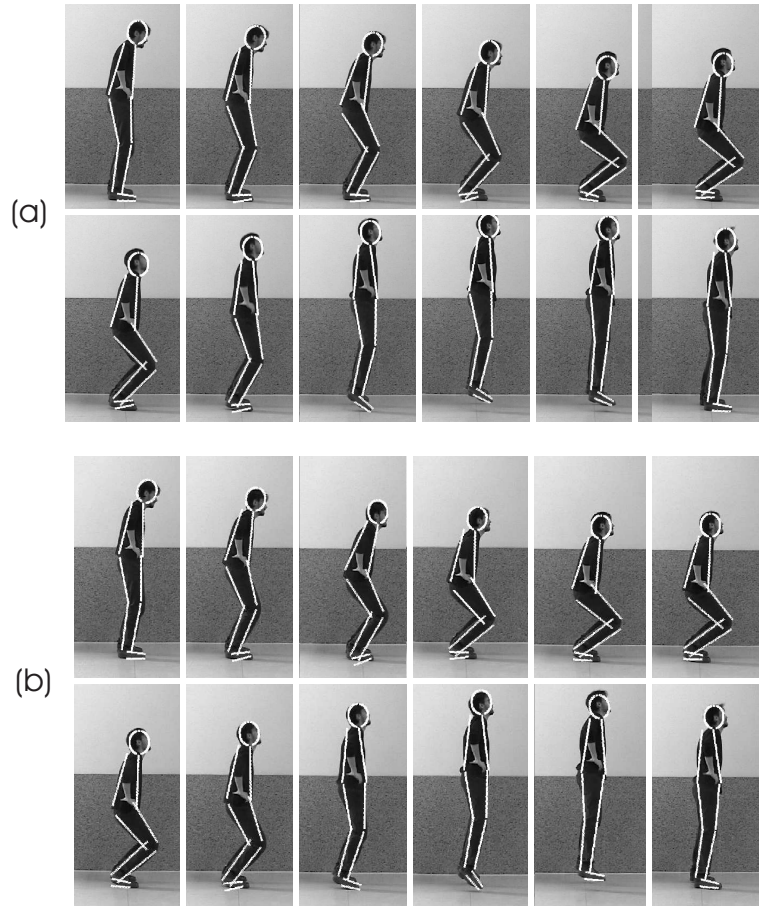
Figure 10: Visual model adjustment for a jumping man using (a) PRPF and (b) SSPF

is used in this work, although any other edge detector could be employed. The resulting human body edges are then smoothed using a convolution operation. This produces a pixel map E^M in which each pixel is set to a value related to its proximity to an edge. Another pixel map E^P is built extracting edges produced by the geometrical model of the configuration predicted by the ith particle, for each pixel j in the pixel map. Similarly, a background subtraction was used to obtain human silhouette. Two pixel maps B^M and B^P are built and compared to compute the corresponding values of C_j^B. Differences between these two maps are computed by:

$$\forall i \in \{1, \ldots, N_{particles}\}, \forall j \in \{1, \ldots, N_{pixel}\} \rightarrow C_E^i = \sum_j |E_j^M - E_j^P| \tag{4}$$

$$\forall i \in \{1, \ldots, N_{particles}\}, \forall j \in \{1, \ldots, N_{pixel}\} \rightarrow C_B^i = \sum_j |B_j^M - B_j^P| \tag{5}$$

Finally, edges and blobs coefficients are combined to obtain i^{th} particle weight at each frame using an exponential weighting function as follows:

$$\forall i \in \{1, \ldots, N_{particles}\} \rightarrow \pi^i = e^{-\alpha(C_E^i + C_B^i)} \tag{6}$$

where α is an experimental parameter which allow us to tune the influence of peaks in the weighting function. This weighting function give a measure of the model fitting quality, so that larger weights mean better fits.
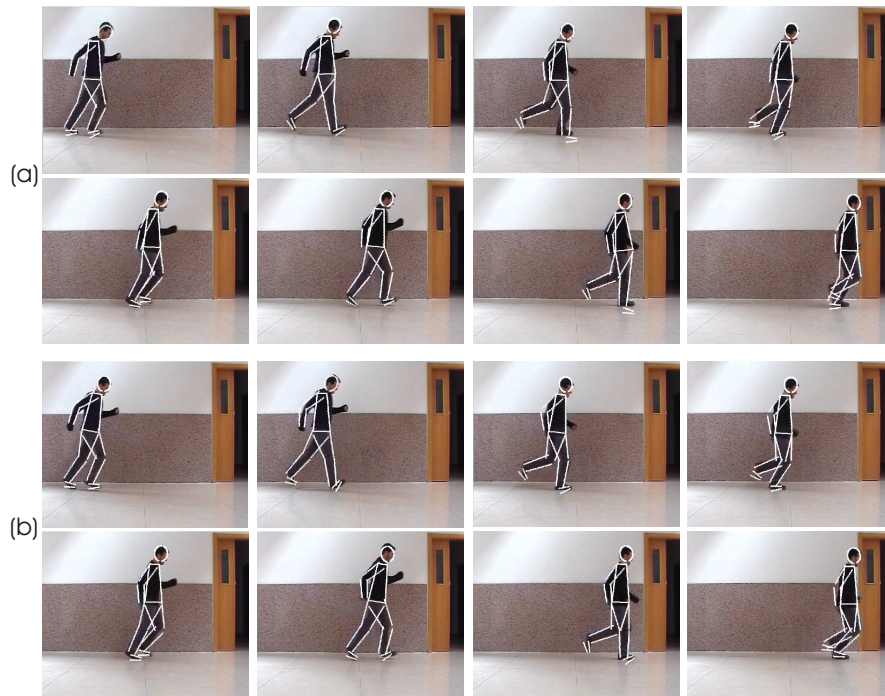
Figure 11: Visual model adjustment for a running man using (a) PRPF and (b) SSPF

5.2.1 System Model

The system model describes the temporal update rule for the system state [21]. The tracked object state consists of a given number of spatial (linear or angular) coordinates and the corresponding velocities, deriving in a first-order motion model. Two excitation forces, F and G, that are modeled by random Gaussian variables with zero mean and normal deviation σ_F and σ_G respectively allow changes in the object state (position and velocity). The value of σ_F and σ_G depend on expected changes in the position and velocity of the tracked object. The update rule used in this work is performed by these two equations:

$$
\begin{aligned}
x_{t+\Delta t} &= x_t + \dot{x}_t \Delta t + F_x \\
\dot{x}_{t+\Delta t} &= \dot{x}_t + G_x
\end{aligned}
\tag{7}
$$

where x represents some spatial (linear or angular) variable, Δt is the time step and F_x and G_x are random Gaussian variables with zero mean and normal deviation σ_F and σ_G, respectively.

6 Experimental Results

To analyze the performance of the proposed model-based PRPF and SSPF algorithms, people performing different activities were recorded in several scenarios. These algorithms were implemented using MATLAB 6.1. Figure 8 shows the model adjustment for a subject performing planar movements. Upper-body model consists of eight limbs. A visual comparison leads to a very good estimation between the PRPF and SSPF results. Right elbow angle estimation using PRPF, SSPF are compared against the One-Layered Particle FIlter (1LPF) and manual digitizing curves in Figure 9. One-layered Particle Filter algorithm is an improved version of classical Particle Filter. A description of this algorithm can be found in [5].

Table 2 shows the mean values of several angles from frontal (Figure 8) and jump (Figure 10) sequences. In

SEQUENCES		1LPF	PRPF	SSPF
	$N_{part}/frame$	1600	1363	999
JUMP	*Knee Angle (MSE/fr)*	10.53	5.10	8.72
(Figure 10)	*Hip Angle (MSE/fr)*	6.81	5.15	5.86
FRONTAL	$N_{part}/frame$	4000	2401	1626
MOVEMENT	*Right Elbow Angle (MSE/fr)*	17.43	8.27	7.99
(Figure 8)	*Left Elbow Angle (MSE/fr)*	45.22	11.31	8.67

Table 2: $MSE/frame$ values with respect to manual digitizing and $N_{part}/frame$ of one-layered PF, PRPF and SSPF for two different motion sequences
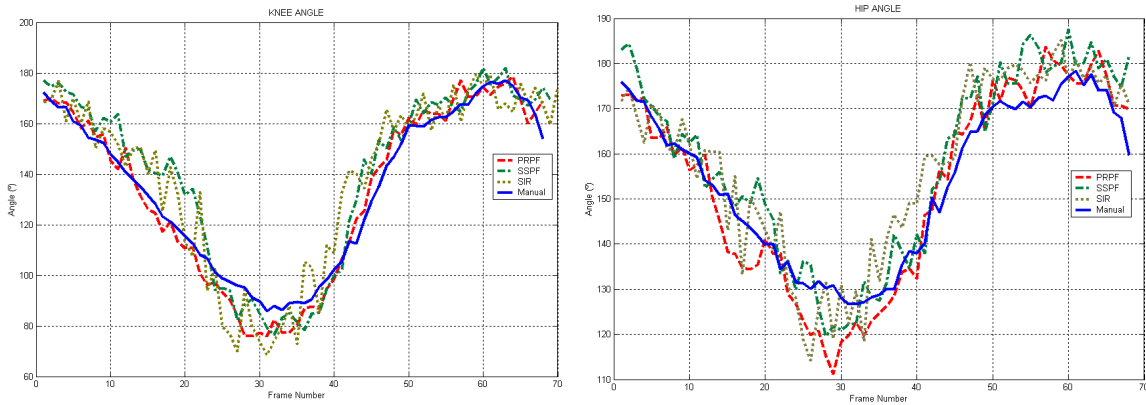

Figure 12: Right hip (left) and knee (right) angle estimation in the jump sequence shown in Figure 10 using PRPF ($N_{part}/frame$ = 2838), SSPF ($N_{part}/frame$ = 1626), 1LPF ($N_{part}/frame$ = 4000) and manual digitizing

order to give a measure of the performance of three methods, we calculate a performance factor (P_f) given by

$$P_f = \frac{1}{N_{part} * MSE/fr} \qquad (8)$$

where N_{part} is the number of particles and MSE/fr is the mean square error per frame. P_f increases when N_{part} or MSE/fr decrease. Thus, a greater value of p indicates greater performance of the approach. Table 3 shows the P_f obtained for one-layered PF, PRPF and SSPF in absolute and relative terms. In these experiments SSPF obtain the best performance factor.

SEQUENCES		1LPF	PRPF	SSPF	PRPF/1LPF	SSPF/1LPF
JUMP	P_f *Knee Angle*	5.9×10^{-5}	1.4×10^{-4}	1.1×10^{-4}	2.4	1.9
	P_f *Hip Angle*	9.2×10^{-5}	1.4×10^{-4}	1.7×10^{-4}	1.5	1.9
FRONTAL	P_f *Right Elbow Angle*	1.4×10^{-5}	5.0×10^{-5}	1.2×10^{-5}	3.5	8.7
MOVEMENT	P_f *Left Elbow Angle*	5.5×10^{-6}	3.7×10^{-5}	1.1×10^{-4}	6.7	20.8

Table 3: Performance Factor obtained for one-layered PF, PRPF and SSPF for two different motion sequences

Figure 11 shows a runner tracked with a ten limbs body model using PRPF and SSPF algorithm. Both sequences demonstrate an accurate model adjustment. Right arm is not included into the geometrical model because it remains completely occluded during most video sequence. Figure 10 shows the same countermovement jump sequence tracked by PRPF and SSPF. A full-body model formed by only five limbs is employed.

Selected non-consecutive frames are shown in both figures. Right knee (left) and hip (right) angle estimation using PRPF, SSPF, 1LPF and manual digitizing curves are shown in Figure 12.

7   Conclusion

The main contribution of this work is the application of the Path Relinking Particle Filter (PRPF) and the Scatter Search Particle Filter (SSPF) algorithms to the model-based human motion tracking. Both algorithms were originally developed for general dynamic optimization and complicated sequential estimation problems. Experimental results have shown that PRPF and SSPF frameworks can be very efficiently applied to the 2D human pose estimation problem. We have estimated a performance factor taking into account the number of particles and the MSE of the corresponding methods against the manual digitizing. By means of this factor we observe that the SSPF algorithm has the best performance hit in terms of MSE and computational load. The proposed geometrical human model is flexible and easily adaptable to the different analyzed human motion activities. However, it depends on the view-point and it is only suitable for planar movements. In this way, quite energetic planar activities such as running and jumping in different environment have been effectively tracked.

References

[1] M. Arulampalam, "A Tutorial on Particle Filter for Online Nonlinear/Non-Gaussian Bayesian Tracking", *IEEE Trans. On Signal Processing*, 50(2):174-188 (2002)

[2] C. Blum, A., Roli, "Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison", *ACM Computing Surveys*, 35(3):268-308, 2003.

[3] J. Carpenter, P. Clifford, P. Fearnhead, "Building robust simulation based filters for evolving data sets", *Tech. Rep., Dept. Statist., Univ. Oxford, Oxford, U.K*, 1999.

[4] J. Deutscher, A. Blake, I. Reid, "Articulated body motion capture by annealed particle filtering", *IEEE Conf. Computer Vision and Pattern Recognition* 2:126-133, 2000

[5] J. Deutscher and I. Reid, "Articulated Body Motion Capture by Stochastic Search", IJCV 61(2):185-205, 2005.

[6] D. Gavrila, "The visual analysis of human movement: a review", *Computer Vision and Image Understanding* 73(1):82-98 ,1999.

[7] F. Glover "A Template for Scatter Search and Path Relinking", *Artificial Evolution, Lecture Notes in Computer Science* 1998(1363):13-54, 1998

[8] F. Glover, G. Kochenberger, "Handbook of metaheuristics", *Kluwer Academic Publishers*, 2002.

[9] F. Glover, M. Laguna, R. Mart "Scatter Search and Path Relinking: Foundations and Advanced Designs", *In New Optimization techniques in Engineering*, 2003

[10] S. Ju, M. Black, Y. Yaccob "Cardboard people: a parameterized model of articulated image motion", *IEEE Int. Conf. on Automatic Face and Gesture Recognition* 1:38-44, 1996

[11] I. Kakadiaris, R. Sharma, "Editorial Introduction to the special issue on human modelling, analysis and syntesis", *Machine Vision and Applications* 14: 197-198, 2003

[12] M. Laguna, R. Marti, "Scatter Search methodology and implementations in C", *Kluwer Academic Publisher*, 2003.

[13] M.K. Leung, Y.H., Yang, "First sight: a human body outline labeling system" *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17(4):359-377, 1995

[14] J. MacCormick, A. Blake "Partitioned sampling, articulated objects and interface-quality hand tracking", *Proceedings of the 7^{t}h European Conference on Computer Vision* 2:3-19 ,2000

[15] B. Moeslund, E. Granum, "A Survey on Computer Vision-Based Human Motion Capture", *Computer Vision and Image Understanding* 81(3): 231-268, 2001.

[16] H. Ning, T. Tan, L. Wang, W. Hu, "People Tracking based on Motion model and motion constraints with automatic initialization", Pattern Recognition 37, 1423-1440, 2004.

[17] J. J., Pantrigo, A. Sánchez, K. Gianikellis, A. Duarte, "Path Relinking Particle Filter for Human Body Pose Estimation", *Proceedings of the Joint IAPR international workshops SSPR 2004 and SPR 2004*, LNCS 2004(3138): 653-661, 2004.

[18] K. Rohr, "Human movement analysis based on explicit motion models", *In Motion-based Recognition, Kluwer Academic Publishers* 171-198, 1997

[19] S. Wachter, H. Nagel, "Tracking persons in monocular image sequences", *Computer Vision Image Understanding* 74(3):174-192, 1999

[20] L. Wang, H. Weiming, T. Tieniu, "Recent developments in human motion analysis", *Pattern Recognition* 36(3): 585-601, 2003.

[21] D. Zotkin, R. Duraiswami, L. Davis, "Joint Audio-Visual Tracking Using Particle Filters", *EURASIP Journal on Applied Signal Processing* 11:1154-1164, 2002