2D Human Tracking by Efficient Model Fitting using a Path Relinking Particle Filter

Juan José Pantrigo[1], Ángel Sánchez[1], Kostas Gianikellis[2], Antonio S. Montemayor[1]

[1]Universidad Rey Juan Carlos, c/ Tulipán s/n
28933 Móstoles, Spain
{j.j.pantrigo, an.sanchez, a.sanz}@escet.urjc.es
[2]Universidad de Extremadura, Avda. Universidad s/n
10071 Cáceres, Spain
kgiannik@unex.es

Abstract. This paper presents a 2D model-based Path Relinking Particle Filter (PRPF) algorithm for human motion tracking and analysis applications. PRPF algorithm hybridizes both Particle Filter and Path Relinking frameworks. The proposed algorithm increases the performance of general Particle Filter by improving the quality of the estimate, by adapting computational load to problem constraints and by reducing the number of required evaluations of the weighting function. A 2D human body model was used to develop tracking with PRPF. This model consists of a simple hierarchical set of articulated limbs, which is described by geometrical parameters. It is easily adaptable to the tracking application requirements. We have applied the PRPF algorithm to 2D human pose estimation in different movement tracking activities such as walking and jumping. Qualitative experimental results show that the model-based PRPF is appropriate in 2D human pose estimation studies.

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 [1][2][3][4]. Analysis and synthesis of human motion has numerous applications. In Visual Surveillance, gait recognition has been used in access control systems [1]. In Advanced User Interfaces, visual analysis of human movement is applied in detecting human presence and interpreting human behaviour [1]. Human motion analysis in Medicine can be employed to characterize and diagnose certain types of disorders [4]. 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 [5].

Human body is usually represented as an articulated model. This is due to the fact that it consists of a set of limbs linked one to each other at joints which allow different movements of these limbs [6]. Most studies in human motion analysis are based on articulated models that properly describe the human body [2][6][7][8]. 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 [7] a cardboard people model was proposed. Human body segments were modelled by planar patches. Leung and Yang [9] used a 2D ribbons with U-shaped edge segments. Rohr [6] 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 n-dimensional 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 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 [2].

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 [2]. 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 called particles in a process state-space [10].

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 [11]. Also, in the Condensation approach, the number of required particles grows with the size of the state-space, as demonstrated in [12]. To address this difficulty, several optimized PF algorithms have been proposed. They use different strategies to improve their performance. Deutscher [11] developed an algorithm termed Annealed Particle Filter (APF) for tracking people. This filter works well for full-body models with 29 DOFs. Partitioned Sampling (PS) [12] 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 resampling stage.

We propose a model-based Path Relinking Particle Filter (PRPF) [13] applied to the visual tracking. This algorithm is inspired by the Path Relinking Metaheuristic proposed by Glover [14][15] as a way to integrate intensification and diversification strategies in the context of combinatorial optimization problems. PRPF hybridizes both Particle Filter (PF) and Path Relinking (PR) frameworks in two different stages. In the PF stage, a particle set is propagated and updated to obtain a new particle set. In PR stage, an elite subset (called *RefSet*) from the particle set is selected, and new solutions are constructed by exploring trajectories that connect each of the particles in the *RefSet*. Also, a geometrical model is used to represent the human body in a 2D space as a hierarchical set of articulated limbs, which allows us to obtain pose from images at any time. We represent the shape of the human body by a set of plane trapeziums and ellipses connected by joints. Moreover, this model is simple, the number of required parameters is small and it is easily adaptable to describe different articulated objects.

2 Particle Filter

General particle filters (PF) are sequential Monte Carlo estimators based on particle representations of probability densities, which can be applied to any state-space model [16]. The state-space model consists of two processes: (i) an observation process $p(\mathbf{Z}_t|\mathbf{X}_t)$, where $\mathbf{X}$ denotes the system state vector and $\mathbf{Z}$ is the observation vector, and (ii) a transition process $p(\mathbf{X}_t|\mathbf{X}_{t-1})$. Assuming that observations $\{\mathbf{Z}_0, \mathbf{Z}_1, \ldots, \mathbf{Z}_t\}$ are sequentially measured in time, the goal is to estimate the new system state $\{\chi_0, \chi_1, \ldots, \chi_t\}$ at each time. In the framework of Sequential Bayesian Modelling, posterior pdf is estimated in two stages:

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

$$p(\mathbf{X}_t \mid \mathbf{Z}_{t-1}) = \int p(\mathbf{X}_t \mid \mathbf{X}_{t-1}) p(\mathbf{X}_{t-1} \mid \mathbf{Z}_{t-1}) d\mathbf{X}_{t-1} \qquad (1)$$

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

(ii) Evaluation: posterior pdf $p(\mathbf{X}_t|\mathbf{Z}_t)$ is computed using the observation vector $\mathbf{Z}_t$:

$$p(\mathbf{X}_t \mid \mathbf{Z}_t) = \frac{p(\mathbf{Z}_t \mid \mathbf{X}_t) p(\mathbf{X}_t \mid \mathbf{Z}_{t-1})}{p(\mathbf{Z}_t \mid \mathbf{Z}_{t-1})} \qquad (2)$$

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 $p(\mathbf{X}_t|\mathbf{Z}_t)$, that constitutes a complete solution to the sequential estimation problem. This pdf is represented by a set of weighted particles $\{(\mathbf{x}_t^0, \pi_t^0) \ldots (\mathbf{x}_t^N, \pi_t^N)\}$, where the weights $\pi_t^n \propto p(\mathbf{Z}_t| \mathbf{X}_t = \mathbf{x}_t^n)$ are normalised. The state χ_t can be estimated by the equation:

$$\chi_t = \sum_{n=1}^{N} \pi_t^n \mathbf{x}_t^n \qquad (3)$$

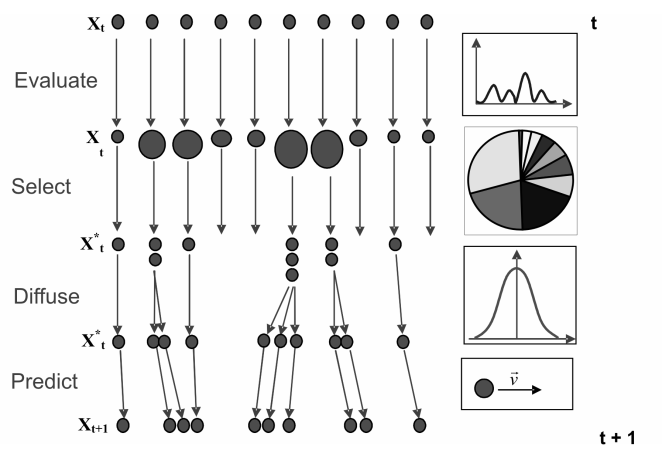


Fig. 1. Particle Filter scheme

PF starts by setting up an initial population $\mathbf{X}_0$ of N particles using a known pdf. The measurement vector $\mathbf{Z}_t$ at time step t, is obtained from the image. Particle weights $\mathbf{\Pi}_t$ are computed using the weighting function. Weights are normalized and a new particle set $\mathbf{X}^*_t$ is selected. As particles with larger weight values can be chosen several times, a diffusion stage are applied to avoid the loss of diversity in $\mathbf{X}^*_t$. Finally, particle set at time step $t+1$, $\mathbf{X}_{t+1}$, is predicted using the motion model. A pseudocode of a general PF is detailed in [13][16].

3 Path Relinking

Path Relinking (PR) [14][15] 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 that connect 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, by exploring trajectories that connect solutions in the *RefSet*. The metaheuristic starts with two of these solutions x' and x'', and it generates a path $x' = x(l), x(2),..., x(r) = x''$ in the neighbourhood space that leads toward the new sequence. In order to produce better quality solutions, it is convenient to add a local search optimization phase. A pseudocode of PR can be found in [13][14].

4 Path Relinking Particle Filter

Path Relinking Particle Filter (PRPF) algorithm was introduced in [13] to be applied to estimation problems in sequential processes that can be expressed using the state-space model abstraction. As pointed out in section 1, PRPF integrates both PF and PR frameworks in two different stages. The PRPF algorithm is 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. Figure 2 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. Each particle represents a possible solution of the problem. Particle weights are computed using a weighting function and a measurement vector. 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. New solutions are generated and evaluated, by exploring trajectories that connect all possible pairs of particles in the *RefSet*. In order to improve the solution fitness, a local search from some of the generated solutions within the PR procedure is performed. PR stage ends when the news generated solutions do not improve the quality of the *RefSet*.

Once the PR stage is over, the "worst" particles are replaced with the *RefSet* solutions. Then, a new population of particles is created by selecting the individuals from the whole particle set with probabilities according to their weights. In order to avoid the loss of diversity, a diffusion stage is applied to the particles of the new set. At the end, particles are projected into the next time step by making use of the update rule. The pseudocode of PRPF algorithm for visual tracking is detailed in [13].

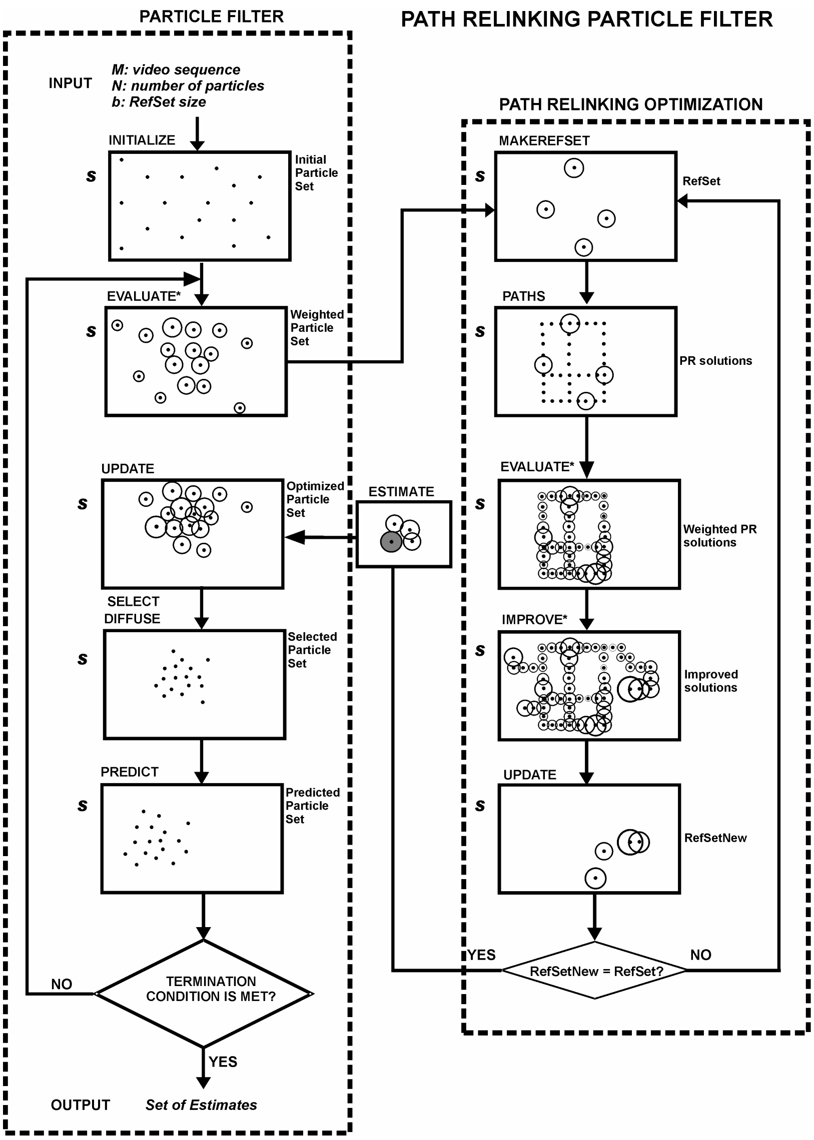


Fig. 2. Path Relinking Particle Filter scheme. Actual frame measures are required during EVALUATE and IMPROVE stages (*)

PRPF estimator quality is improved with respect to PF and the required number of evaluations for the weighting function is also reduced. Therefore, PRPF search in the state-space is not performed randomly like in a general particle filter. PRPF is 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, then paths connecting solutions became long enough, and the number of explored solutions increases. It is not possible to have any estimate of the previous state of the system at the beginning of the visual tracking, therefore the particle filter is usually randomly initialized. 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.

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 process in the state-space model abstraction.

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 3 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 full human body.

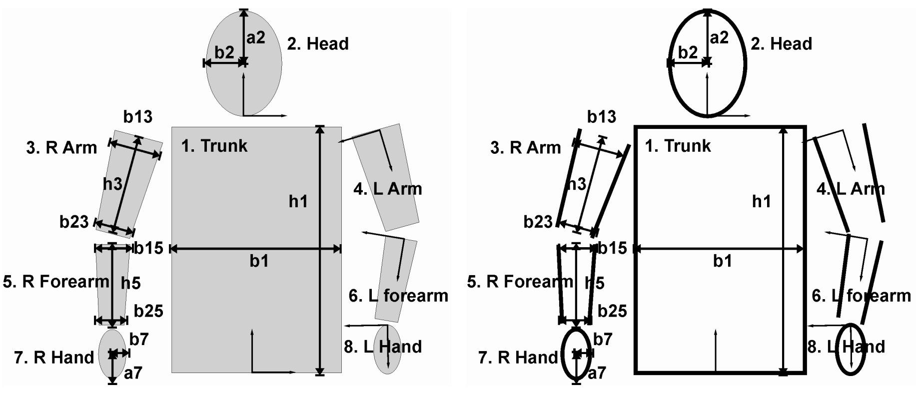


Fig. 3. Proposed blob (left) and edge (right) configuration for human upper-body model

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

	Trunk	Head	R. arm	R. forearm	R. hand
Ident.	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]
Pos.		[0, h1+Δ]	[-b1/2, h1-b12/2]	[0, h3]	[0, h5]

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. *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 limbs descriptions in the proposed model are shown in Table 1.

Particles store time-dependent values relating to limb positions, orientations and velocities. The state $\mathbf{x}_t^i$ of a particle $(\mathbf{x}_t^i, \pi_t^i)$ in an eight-limb model is described as:

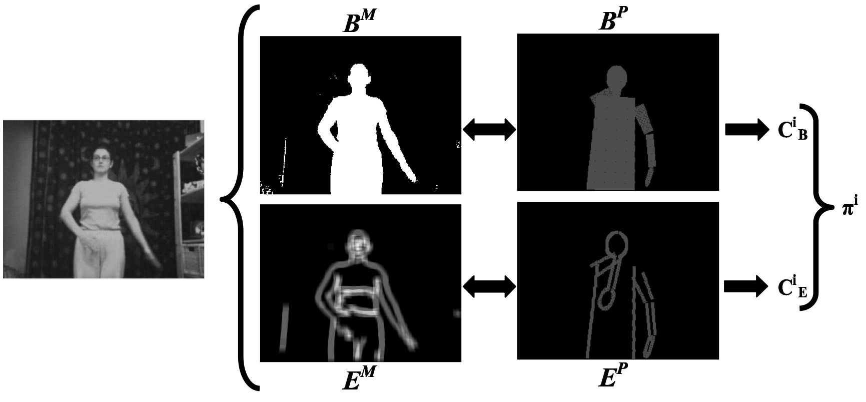
$$\left[x_1, y_1, \theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6, \theta_7, \theta_8, \dot{x}_1, \dot{y}_1, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5, \dot{\theta}_6, \dot{\theta}_7, \dot{\theta}_8\right] \quad (4)$$

where x and y are the spatial positions, θ is the limb orientation in the father's system of reference and $\dot{x}$, $\dot{y}$ and $\dot{\theta}$ represents the first derivative of it's corresponding variable.

The goal of the geometrical model is to connect 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 edge and blobs synthetic images. Note that used tracking model parameters are defined with respect to the camera view point. Features, those extracted from each frame in the video sequence and those predicted by the PRPF, 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.

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 [2]. 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.



(a) (b) (c) (d)

Fig. 4. Observation process: (a) initial image, (b) feature extraction, (c) particle prediction and (d) particle weight computation

Figure 4 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. We use a Canny edge method to extract edges in the human body image. The resulting 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, 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^i_B. Differences between these two maps are computed by:

$$\forall\, \mathrm{i} \in \{1..\mathrm{N}_{\mathrm{part}}\}, \forall\, \mathrm{j} \in \{1..\mathrm{N}_{\mathrm{pixel}}\}:\ C_E^i = \sum_j \left| E_j^M - E_j^B \right| \tag{5}$$

$$\forall\, \mathrm{i} \in \{1..\mathrm{N}_{\mathrm{part}}\}, \forall\, \mathrm{j} \in \{1..\mathrm{N}_{\mathrm{pixel}}\}:\ C_B^i = \sum_j \left| B_j^M - B_j^B \right| \tag{6}$$

Finally, edges and blobs coefficients are combined to obtain *ith* particle weight at each frame using the following weighting function:

$$\forall\, \mathrm{i} \in \{1..\mathrm{N}_{\mathrm{part}}\}, \pi^i = e^{-\alpha(C_E^i + C_B^i)} \tag{7}$$

where α is an experimental parameter.

5.3 System Model

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

$$\begin{cases} x_{t+\Delta t} = x_t + \dot{x}_t \Delta t \\ \dot{x}_{t+\Delta t} = \dot{x}_t + G_x \Delta t \end{cases} \tag{8}$$

where x represents some spatial (linear or angular) variable, Δt is the temporal step and G_X is a random Gaussian variable with zero mean and normal deviation σ_X.

6 Experimental Results

In order to analyse the performance of the proposed model-based PRPF system, people performing different activities were recorded in different scenarios. PRPF algorithm was implemented using MATLAB 6.1. Figure 5 shows the model adjustment for a subject performing planar movements. Upper-body model consists of eight limbs. Right elbow angle estimation using PRPF, Sampling Importance Resampling (SIR) and manual digitizing curves are shown in Figure 6.

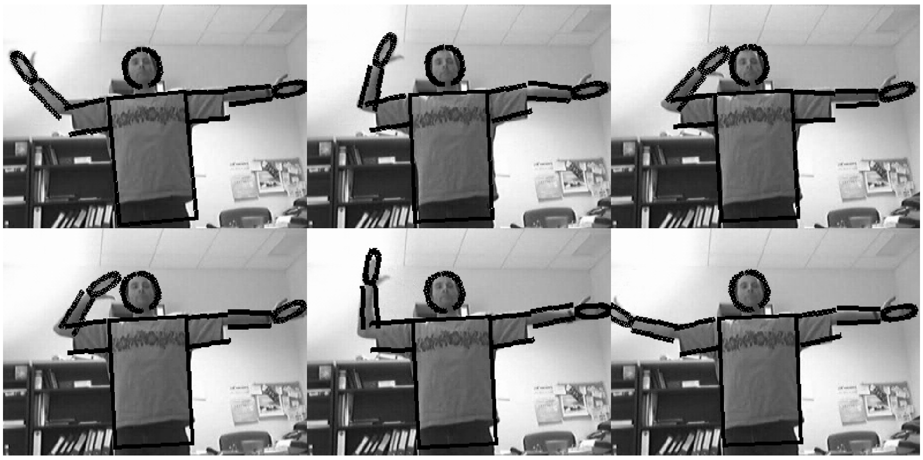


Fig. 5. Visual model adjustment for a subject performing planar movements (frames # 1, 10, 20, 30, 40 and 50)

Figure 7 shows a runner tracked with a ten limbs full-body model. This sequence demonstrates that the model adjustment is accurate. In Figure 8 a countermovement jump sequence is shown. 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 and manual digitizing curves are shown in Figure 9. Finally, Table 2 shows the MSE values of some calculated angles from frontal and jump sequences.

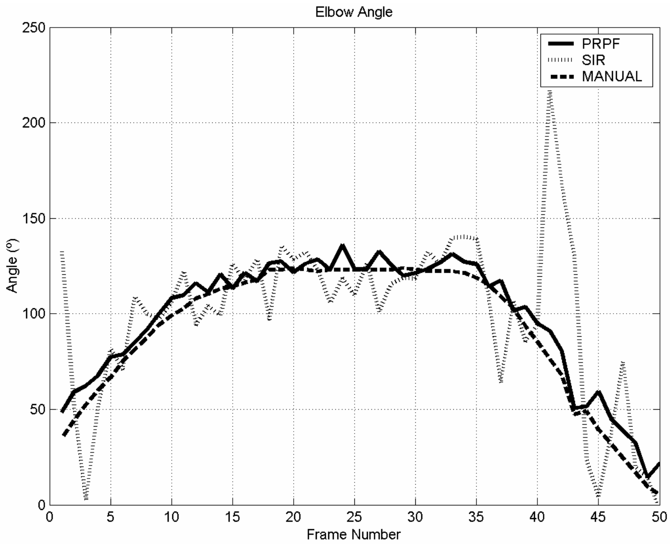


Fig. 6. Right elbow angle estimation using PRPF, SIR and manual digitizing

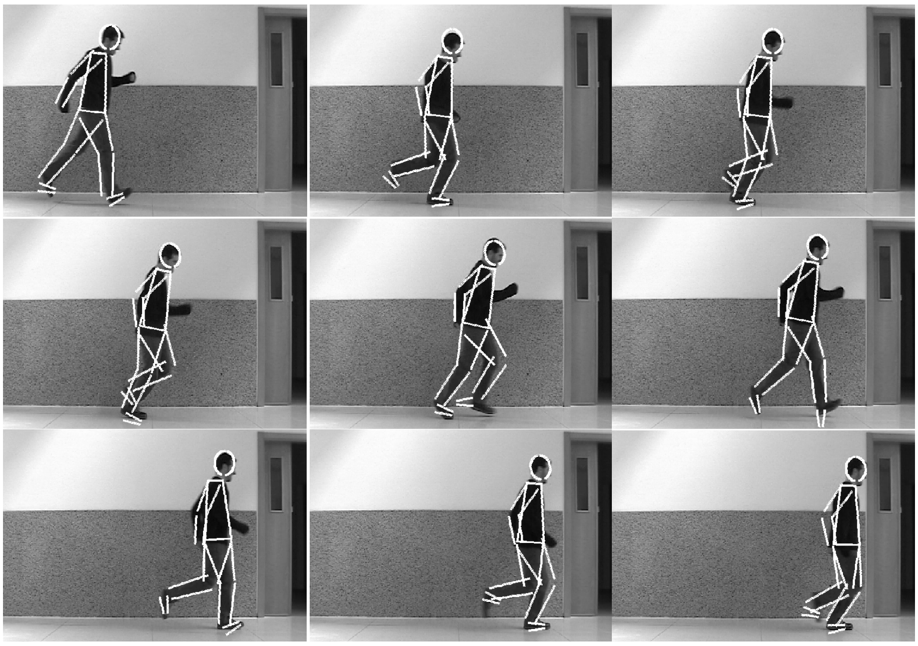


Fig. 7. Visual model adjustment for a running man

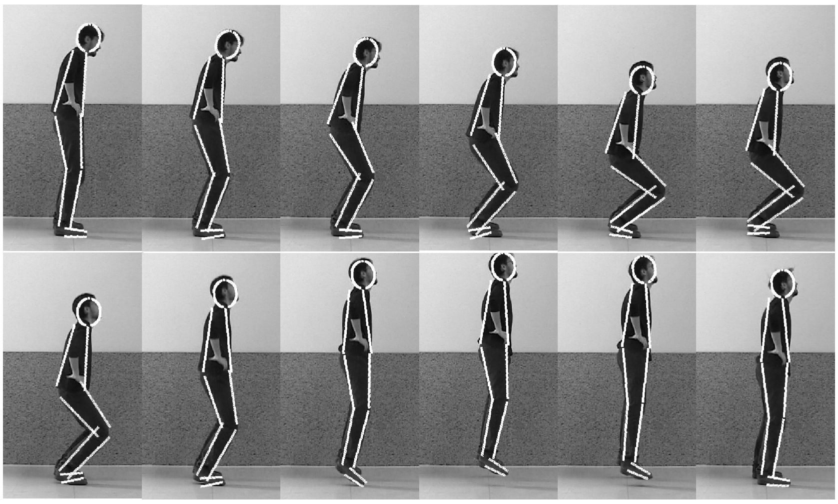


Fig. 8. Visual model adjustment for a jumping man

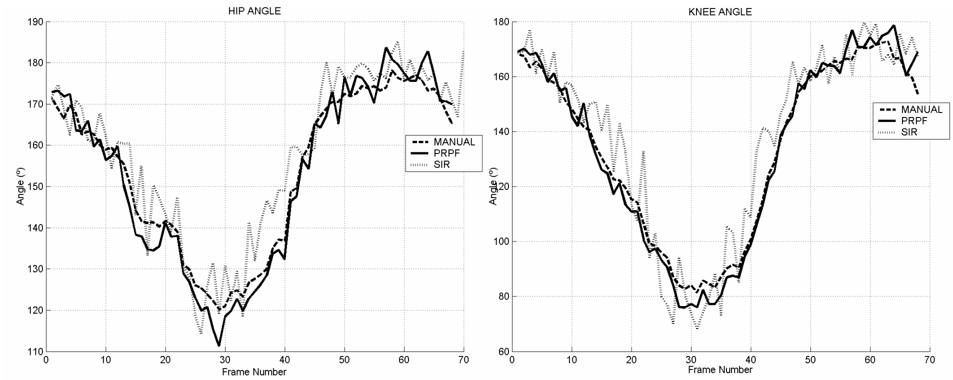


Fig. 9. Right hip (left) and knee (right) angle estimation using PRPF, SIR and manual digitizing

Table 2. *MSE / frame* values with respect to manual digitizing and N_{part} */ frame* of SIR and PRPF for two different motion sequences

SEQUENCES		**SIR**	**PRPF**
JUMP (figure 8)	N_{part} / frame	1600	1363
	MSE Knee Angle / frame	124,4	20,6
	MSE Hip Angle / frame	45,9	15,4
FRONTAL MOVEMENT (figure 5)	N_{part} / frame	4000	2838
	MSE R Elbow Angle / frame	1220,4	71,7
	MSE L Elbow Angle / frame	6912,4	194,2

7 Conclusion

The main contribution of this work is the application of the Path Relinking Particle Filter (PRPF) algorithm to model-based human motion tracking and analysis. PRPF was originally developed for general estimation problems in sequential processes that are represented by the state-space model abstraction. Experimental results have shown that this model-based PRPF framework can be very efficiently applied to the 2D human pose estimation problem, as demonstrated by Table 2. The proposed geometrical human model is flexible and has been designed to be easily adapted to the different analyzed human motion activities. In this way, quite energetic activities such as running and jumping in different environment have been easily and effectively tracked. Managing partial occlusions in different human tracking problems with the PRPF approach will be improved. We also propose the study of PRPF in 3D scenarios for human pose estimation in biomechanics applications. Finally, the authors are working in a robust colour and edge based PRPF applied to tracking articulated objects.

References

1. Wang, L., Weiming, H., Tieniu, T.: Recent developments in human motion analysis. Pattern Recognition 36 (2003) 585–601
2. Moeslund, B., Granum, E.: A Survey on Computer Vision-Based Human Motion Capture. Computer Vision and Image Understanding 81 (2001) 231–168
3. Gavrila, D: The visual analysis of human movement: a review. CVIU 73 1: (1999)
4. Kakadiaris, I., Sharma, R.: Editorial Introduction to the special issue on human modelling, analysis and syntesis. Machine Vision and Applications Vol. 14 (2003) 197–198
5. IBV: Biomechanics. http://www.ibv.org/ingles/ibv/index2.html (1992)
6. Rohr, K.: Human movement analysis based on explicit motion models. Chapter 8 in Motion-based Recognition, Kluwer Academic Publishers (1997) 171–198
7. Ju, S., Black, M, Yaccob, Y.,: Cardboard people: a parameterized model of articulated image motion. IEEE Int. Conf. on Automatic Face and Gesture Recognition (1996) 38–44
8. Wachter, S., Nagel, H.-H.: Tracking persons in monocular image sequences. Computer Vision Image Understanding 74 Vol 3 (1999) 174–192
9. Leung, M.K., Yang, Y.H.: First sight: a human body outline labeling system, IEEE Trans. Pattern Anal. Mach. Intell. 17 (4) (1995) 359–377.
10. Zotkin, D., Duraiswami, R., Davis, L.: Joint Audio-Visual Tracking Using Particle Filters. EURASIP Journal on Applied Signal Processing, Vol, 11 (2002) 1154–1164
11. 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
12. MacCormick, J.; Blake, A.: Partitioned sampling, articulated objects and interface-quality hand tracking. Proc European Conf. Computer Vision (2000)
13. Pantrigo, J.J., Sánchez, A., Gianikellis, K., Duarte, A.: Path Relinking Particle Filter for Human Body Pose Estimation. Accepted paper to appear in LNCS (2004)
14. Glover, F., Laguna, M., Martí, R.: Scatter Search and Path Relinking: Foundations and Advances Designs. To appear in New Optimization techniques in Engineering (2003)
15. Glover, F.: A Template for Scatter Search and Path Relinking. LNCS, 1363, (1997) 1-53
16. 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)