Articulated Body Motion Tracking by Combined Particle Swarm Optimization and Particle Filtering

Tomasz Krzeszowski, Bogdan Kwolek, and Konrad Wojciechowski

Polish-Japanese Institute of Information Technology
Koszykowa 86, 02-008 Warszawa
http://www.pjwstk.edu.pl

Abstract. This paper proposes the use of a particle filter with embedded particle swarm optimization as an efficient and effective way of dealing with 3d model-based human body tracking. A particle swarm optimization algorithm is utilized in the particle filter to shift the particles toward more promising configurations of the human model. The algorithm is shown to be able of tracking full articulated body motion efficiently. It outperforms the annealed particle filter, kernel particle filter as well as a tracker based on particle swarm optimization. Experiments on real video sequences as well as a qualitative analysis demonstrate the strength of the approach.

1 Introduction

Human body tracking has many applications, including, but not limited to, visual surveillance, human computer communication and recognizing human activities [1]. One problem of human body tracking is to estimate the joint angles of a human body at any time. This is one of the most challenging problems in the area of computer vision because of self-occlusions, a high dimensional search space and high variability in human appearance. The high dimensionality of the configuration space and the exponentially increasing computational cost are the main challenges in full articulated motion tracking [2]. An articulated human body can be thought of as including at least 11 body parts. This may involve around 26 parameters to describe the full body articulation. By building a mapping from configuration space to observation space, 3d model-based approaches rely on searching the pose space to find the body configuration that best-matches the current observations [3]. Matching such complex and self-occluding model to human silhouette might be especially difficult in cluttered scenes. In monocular image sequence this matching problem is under constrained. The major problems with monocular 3d body tracking arise due to depth ambiguities, movements perpendicular to the image plane and occlusion. Multiple cameras and simplified backgrounds are often employed to ameliorate some of such practical difficulties.

Particle filtering is one of the most popular algorithms for tracking human body motion. After the seminal work [4] the particle filter has been utilized in human motion tracking in [2]. In a particle filter each particle corresponds to some hypothesized set of model parameters. Given the number of parameters needed to describe a realistic articulated model of the human body is larger than twenty, the number of particles of that are required to adequately approximate the underlying probability distribution in the body space might be huge. Hence, a considerable amount of approaches has been proposed to overcome the course of dimensionality inherent in the particle filtering. Given the number of allowable configurations of the human body is restricted by biomechanical constraints, some approaches to human motion tracking are based on learning a low-dimensional representation of the high-dimensional configuration space. Tracking of human motion in such a low-dimensional manifold results in lower numbers of required particles. Gaussian Process Latent Variable Models (GPLVM) [5] provide inverse mapping from the pose space to the latent space. However, manifolds can only be learned for specific activities, such as walking, jumping or running, and it unclear how this approach can be extended to broader classes of human motion.

The likelihood function in human motion tracking can be very peaky with multiple local maxima. In [6], to cope with multiple local maxima the particles are repositioned via a deterministic search in order to attain a better concentration around significant modes of the importance function. A different approach has been proposed in [7] where the promising areas in the probability distribution are identified through iterative mode-seeking using mean-shift. Experiments on real monocular image sequences demonstrated that the algorithm is capable of tracking two arms of upper human body at 7.5 Hz on a laptop computer. Another approach that has gained considerable interest in this type of problems consists in a coarse to fine search on the importance function of the particle filter [2]. Very good results were obtained in a setup with three cameras. As reported in [8] the annealed particle filter has good performance if the images are acquired with frame rate of 60 fps and the tracking performance of such a filter drops when the frame rate is below 30. Another disadvantage of the annealed particle filter is that it often fails to represent multiple hypotheses. In [9] it has been shown that particle swarm optimization outperforms the simulated annealing in terms of accuracy and consistency of the results.

One drawback of using particle filter in human motion tracking is the inability of samples to explore the probability distribution efficiently. This arises because the particles do not move according to their former experience and their relationship to other particles. Hence, they have reduced capability to escape the local minima. Therefore, in this work we propose an algorithm which combines particle swarm optimization (PSO) [10] and particle filtering as an effective way for human motion tracking. The interactions between particles in the course of swarm-based optimization lead to the emergence of global and collective behavior, which allows the particles to gravitate towards the global extremum, whereas the particle filter is responsible for maintaining multi-modal densities.

In the reminder of this paper we briefly outline particle filtering and particle swarm optimization. We then show our algorithm. Afterwards we discuss our results in more detail. Finally, a summary is presented.

2 The algorithm

Particle swarm optimization is a population based optimization technique, which differs from other evolutionary techniques by inclusion of particle velocity. Particles can be attached to each other by any kind of neighborhood topology represented by a graph. In the fully connected neighborhood topology, which is represented by fully connected graph all particles in the swarm are connected to one another. Each particle in a swarm represents a candidate solution of the problem. With respect to a fitness function, the best location that has been visited thus far by a particle is stored in the particles memory. The fitness values corresponding to such best positions are also stored. Additionally the particles have access to the best location of the whole swarm, i.e. a position that yielded the highest fitness value. A particle therefore employs the best position encountered by itself and the best position of the swarm to move itself toward an optimal value of the objective function.

Each particle i maintains the current position $\mathbf{x}_i$, current velocity $\mathbf{v}_i$, and its best position $\mathbf{y}_i$. For every iteration, the jth-component of particle velocity vector $\mathbf{v}_i$ is updated as follows:

$$v_{i,j} \leftarrow w v_{i,j} + c_1 r_{1,j}(y_{i,j} - x_{i,j}) + c_2 r_{2,j}(\hat{y}_j - x_{i,j}) \tag{1}$$

where w is the positive inertia weight, $v_{i,j}$ is the velocity of particle i in dimension j, $r_{1,j}$ and $r_{2,j}$ are uniquely generated random numbers in the interval $(0,1)$, c_1, c_2 are positive, cognitive and social constants, respectively. The position of each particle is updated according to the following equation:

$$x_{i,j} \leftarrow x_{i,j} + v_{i,j} \tag{2}$$

Given the above rules of position and velocity update, the particle swarm optimization algorithm can be expressed as follows:

Function $\hat{\mathbf{y}}$ = PSO ($\mathbf{x}_i$)
  If $\mathbf{x}_i$ == [] (empty matrix), initialize $\mathbf{x}_i$
  Initialize $\mathbf{v}_i$, $\mathbf{y}_i = \mathbf{x}_i$, $\hat{\mathbf{y}} = \arg\min_{\mathbf{x}_i} f(\mathbf{x}_i)$
  Repeat
    For each particle i
      Apply (1) to update velocity of each particle
      Apply (2) to update position of each particle
      Evaluate function $f(\cdot)$ at updated positions $\mathbf{x}_i$
      If $f(\mathbf{x}_i) < f(\mathbf{y}_i)$, update the local best values, $\mathbf{y}_i \leftarrow \mathbf{x}_i$
      If $f(\mathbf{y}_i) < f(\hat{\mathbf{y}})$, update the global best, $\hat{\mathbf{y}} \leftarrow \mathbf{y}_i$
  Until number of function evaluations < *max_iter*

The fitness value of each particle is evaluated by a predefined observation model as follows:

$$f(\mathbf{x}_i) = p(\mathbf{o}_i|\mathbf{x}_i) \tag{3}$$

where $\mathbf{o}_i$ is the observation corresponding to $\mathbf{x}_i$.

The presented above particle swarm optimizer can be employed to carry out a global gradient-less stochastic search for the best configuration of the model parameters. The tracking of the human figure can also be formulated as the computation of the posterior probability distribution over the parameters of the model at time t given a sequence of images. Due to the nonlinearity of the likelihood function over model parameters the computation of the probability distribution is complicated. For these reasons the posterior is typically represented as a weighted set of particles, which are then propagated via a particle filter.

Particle filters approximate stochastically the state posterior with a set of N weighted particles, (s, π), where s is a sample state and π is its weight. This set of particles is propagated over time. At each time t the particles undergo selecting, predicting and re-weighting. In the select stage the algorithm randomly selects N particles from $\{s_{t-1}\}$ based on weights $\pi_{t-1}^{(n)}$. In the predict phase the particles undergo moving according to deterministic motion model. After the drift the particles are perturbed individually. Afterwards, based on observation model $p(\mathbf{o}_i|\mathbf{x}_i)$ the likelihood for each new sample is calculated, and finally weights are updated to obtain $\{s_t^{(n)}, \pi_t^{(n)}\}$.

In our approach to articulated motion tracking we exploit the power of the particle filter to represent multimodal distributions, which arise due to strong nonlinearity of the likelihood function over model parameters. Through the use of the particle filter less likely model configurations are not discarded immediately, but have chance to be considered in the next time. In articulated motion tracking the weakness of the particle filter consists in that the particles typically do not cluster around the true state of the figure and instead they concentrate around local maximas in the posterior distribution. In consequence, if particles are too diffused the tracking can be lost. In order to cope with this we employ particle swarm optimization in the particle filter to shift the particles toward more promising regions in the configuration space. The modified particle filter can be expressed as follows:

1. Select: Randomly select N particles from $\{s_{t-1}\}$ based on weights $\pi_{t-1}^{(n)}$.
2. Predict: Perturb individually particles
3. Shift: Shift the particles via the PSO
4. Re-weight: Get the likelihood for each new sample. Update the weights to obtain $\{s_t^{(n)}, \pi_t^{(n)}\}$
5. Estimate: Estimate the state using the mean $\mathbf{E}[\mathbf{x}_t] \approx \sum_{n=1}^{N} \pi_t^{(n)} s_t^{(n)}$

3 Experimental results

The model of the human body has a form a kinematic chain consisting of 11 segments and the configuration of the model is defined by 26 DOF. The articulated model consists of cuboids modeling pelvis, torso, head, upper and lower arm and legs. A pose configuration is determined by position and orientation of the pelvis in the global coordinate system as well as relative angles between connected limbs. Given the parameters of the camera each cuboid can be projected into 2d image plane. To simplify the projection onto the image we project the corners via perspective projection and afterwards a rendering of the cuboids takes place. A regular rectangular grid is used to extract pixel values for each body part in such a rendered image.

Successful approaches to articulated object tracking typically rely on accurate extraction of foreground silhouettes using background subtraction. In [8][11][7] the edges have additionally been utilized in tracking of the human motion. In order to carry out qualitative analysis we construct foreground silhouettes through manual fitting of the 3d model to the person on the input images and then we render the model. This way we have in disposal the configuration of the human as well as the foreground image. Such a configuration reflecting the current human pose determines the reference image, which undergoes matching via the tracking algorithm. The tracking algorithm operates in 26 dimensional state space and generates pixel maps that are employed in computing the likelihoods.

The experiments were conducted on images acquired from surveillance cameras that are situated in a student hostel, see Fig. 1. For visualization purposes the reference sub-images were placed at the bottom left part of the input images. By projecting the body model into the images we can extract information for each body part as shown in the mentioned sub-images. The size of the input images is 720x576 and they were acquired at 6 fps. The low frequency of the input sequence comprises considerable challenge for the examined algorithms.

Fig. 1. 3d model-based human body tracking, frames #5, #15, #25 and #35, left bottom: appearance image of person undergoing tracking. The degree of overlap between the appearance image and the projected model into 2d image plane of the camera is 0.86, 0.80, 0.80 and 0.79, respectively.

A comparison among particle filter with embedded particle swarm optimization (PF+PSO), ordinary particle filter (PF), particle swarm optimization (PSO), kernel particle filter (KPF) [7], and annealed particle filter (APF) can be seen in Tab. 1. It can be observed that the PF+PSO algorithm is better in comparison to all remaining algorithms in term of the accuracy of body motion tracking. Both PSO+PF and PSO are superior to the remaining algorithms in terms of the computation time. Using 200 particles and 5 iterations in an unoptimized C/C++ implementation of the PF+PSO-based algorithm, a 2.0 GHz PC requires about 1.36 sec. per image to perform the motion tracking, most of the time being spent in the evaluation of the fitness function. In such a configuration of the tracker the estimates of the human pose in the sequence from Fig. 1 have acceptable accuracy. At the mentioned figure we demonstrate some experimental results, which were obtained using 500 particles and in 10 iterations. The results obtained via PSO are superior in comparison to results produced by KPF and APF. In the employed test sequence, which has been acquired with relatively low frequency, the KPF behaved better than APF. The discussed results are averages from three independent runs of the algorithms.

Table 1. Computation time (4-th column) and average degrees of overlap between the reference image of human body and the estimated body pose (3-rd column) for particle filter (PF), particle swarm optimization (PSO), particle filter with particle swarm optimization (PF+PSO), kernel particle filter (KPF) and annealed particle filter (APF).

	#particles	#it.	overlap [%]	time [sec.]
PF	20000		0.75	22.92
	10000		0.74	11.39
	5000		0.73	5.69
	2000		0.67	2.71
PSO	1000	10	0.83	12.45
	500	10	0.80	6.20
	200	10	0.76	2.49
	1000	5	0.81	6.79
	500	5	0.78	3.40
	200	5	0.76	1.36
PF+PSO	1000	10	0.84	12.63
	500	10	0.83	6.24
	200	10	0.80	2.50
	1000	5	0.82	6.90
	500	5	0.81	3.41
	200	5	0.78	1.36
KPF	2000	3	0.76	7.26
	1000	3	0.74	3.50
APF	2000	10	0.79	22.38
	1000	10	0.78	11.16
	500	10	0.75	5.60

In Fig. 2 we demonstrate the degree of overlap versus frame number for the algorithms utilized in our experiments. It can be observed that the results obtained via PF+PSO algorithm are better. The PSO-based tracker is superior to PF+PSO tracker in the initial part of the image sequence. The magnitude of change of the overlap degree for PF+PSO is smaller in comparison to other curves. For PF+PSO the degree of overlap does not drop below 0.78, particularly in the end part of the sequence, where the remaining trackers achieve worse overlap degrees. The results for PF+PSO, PSO and APF were obtained using 500 particles and 10 iterations. The results for KPF were obtained with 2000 particles and 3 iterations, whereas 10000 particles were employed in PF. In Fig. 1 we can see how well the rendered model with configuration determined by PF+PSO fits the human silhouette, which has been shot in frames #5, #15, #25 and #35. In the experiments we have used simple motion models and we expect that prior model of the human motion can improve further the robustness as well as the accuracy of the motion tracking.

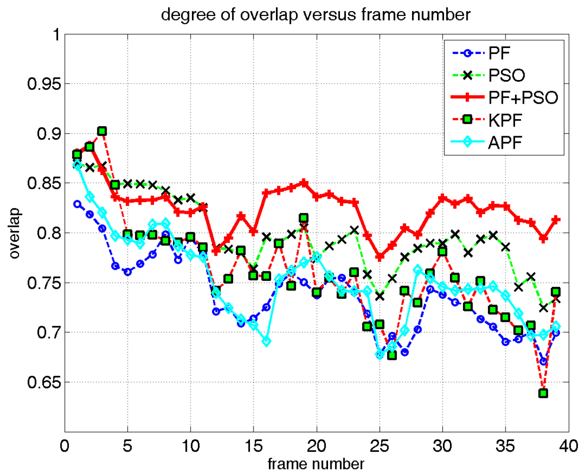

Fig. 2. Degree of overlap between the reference human body and the rendered image on the basis of the best particle

4 Conclusions

The most important general reason for the weakness of the particle filter is that in high dimensional state spaces the particles can not cluster around the true state and instead they have tendency to migrate towards local maximas in the posterior distribution. In this work, an effective algorithm for tracking human motion has been presented. The experimental results suggest that particle filter combined with particle swarm optimization achieves the best results in terms of the accuracy of the tracking. The algorithm is capable of tracking full articulated body motion efficiently. We have demonstrated the behavior of the algorithm on challenging human motion sequence. The algorithm has been compared with particle swarm optimization, kernel particle filter, annealed particle filter and ordinary particle filter. The tracker using particle swarm optimization achieves slight worse tracking accuracy than particle filter combined with particle swarm optimization. In image sequences acquired at low frame rates it outperforms both the annealed particle filter and the kernel particle filter.

Acknowledgment

This paper has been supported by the project "System with a library of modules for advanced analysis and an interactive synthesis of human motion" co-financed by the European Regional Development Fund under the Innovative Economy Operational Programme - Priority Axis 1. Research and development of modern technologies, measure 1.3.1 Development projects.

References

1. Poppe, R.: Vision-based human motion analysis: an overview. Computer Vision and Image Understanding **108** (2007) 4–18
2. Deutscher, J., Blake, A., Reid, I.: Articulated body motion capture by annealed particle filtering. In: IEEE Int. Conf. on Pattern Recognition, Hilton Head, South Carolina, USA (2000) vol. 2, 126–133
3. Sidenbladh, H., Black, M., Fleet, D.: Stochastic tracking of 3d human figures using 2d image motion. In: European Conference on Computer Vision. (2000) 702–718
4. Isard, M., Blake, A.: Condensation - conditional density propagation for visual tracking. Int. J. of Computer Vision **29** (1998) 5–28
5. Lawrence, N.D.: Gaussian process latent variable models for visualisation of high dimensional data. In: Advances in Neural Information Processing Systems (NIPS), Vancouver, Canada. (2003) 329–336
6. Sminchisescu, C., Triggs, B.: Covariance scaled sampling for monocular 3d body tracking. In: IEEE Int. Conf. on Computer Vision and Pattern Recognition. Volume 1. (2001) 447–454
7. Fritsch, J., Schmidt, J., Kwolek, B.: Kernel particle filter for real-time 3d body tracking in monocular color images. In: IEEE Int. Conf. on Face and Gesture Rec., Southampton, UK, IEEE Computer Society Press (2006) 567–572
8. Balan, A., Sigal, L., Black, M.: A quantitative evaluation of video-based 3d person tracking. In: IEEE Workshop on VS-PETS. (2005) 349–356
9. Ivekovic, S., Trucco, E., Petillot, Y.R.: Human body pose estimation with particle swarm optimisation. Evolutionary Computation **16** (2008) 509–528
10. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proc. of IEEE Int. Conf. on Neural Networks, IEEE Press, Piscataway, NJ (1995) 1942–1948
11. Deutscher, J., Reid, I.: Articulated body motion capture by stochastic search. Int. J. Comput. Vision **61** (2005) 185–205