MONOCULAR HUMAN MOTION TRACKING WITH THE DE-MC PARTICLE FILTER

Ming Du, Ling Guan

Ryerson Multimedia Research Lab, Dept. of Electrical and Computer Engineering Ryerson University, Toronto, Canada

ABSTRACT

A key to accomplish articulated human motion tracking and other high-dimensional visual tracking tasks is to have an efficient way to draw samples from the state space. The typical particle filter method and most of its variants do not perform well in achieving this goal. To solve the problem we present a novel algorithm, namely the Differential Evolution - Markov Chain (DE-MC) particle filtering. It substantially improves the core of traditional particle filter, i.e. the sampling strategy. As a result, we can obtain reasonably distributed samples in an efficient way thus translating into reliable tracking performance. Experimental results demonstrate the power of the proposed approach.

1. INTRODUCTION

Human motion tracking plays a critical role in human movement understanding and analysis in that it quantifies the motion for any following processing. Although the human vision system can locate another person and his body parts easily, accurately and quickly, the same task remains arduous for computers. In research, 3-D articulated human body models are extensively used. Model-based approaches have the advantage that they can give natural interpretation and description of human body motion. As a trade-off, one is confronted with the curse of high dimensionality.

The CONDENSATION [5] or particle filter algorithm has become a popular choice for solving visual tracking problems because of its ability to deal with multi-model distributions. However, it turns out to be inadequate when the dimensionality of the state space becomes large. The reason is largely due to its defective sampling strategy. In recent years, many different approaches have been attempted to improve the performance of particle filter based human motion tracker, such as imposing physical constraints [2], training a dynamical model or building HMM for certain type of human motion [11] [12], or decomposing the state space by template matching [8]. However, they either lack sufficient influence on the overall tracking performance or set too rigid restrictions to be used in general tracking scenario. It has been realized that improving the sampling or global optimization strategy is more decisive to the success of articulated model based human motion tracking. Preceding works on this topic include Deutscher et. al.'s annealed particle filter (APF) [3] and Sminchisescu et.al.'s covariance scaled sampling [4] and hyperdynamics importance sampling [6]. Our work belongs to this category.

In this paper we present a new algorithm, which makes substantial changes to the core structure of the traditional particle filter. An information exchange scheme is built between the sampling and the weighting steps to implicitly adjust the estimation of posterior distribution. Moreover, this adjustment also has the purpose of global optimization, which can save considerable energy within the filter. The Differential Evolution-Markov Chain (DE-MC), a statistical method for approximating target density functions, provides necessary assistance in this regard, and is integrated into the framework of Sequential Monte Carlo Sampling. We name the the proposed algorithm DE-MC particle filter and demonstrate its superiority with the application in monocular 3D human pose tracking. However, the generality of the algorithm allows it to be readily used in any visual tracking context.

2. DE-MC PARTICLE FILTER

The visual tracking problem can be deemed a filtering process $p(\mathbf{X}_k | \mathbf{Y}_{1:k})$, where \mathbf{X}_k is the state vector (joint angles in the articulated human motion tracking case) at time *k* and $\mathbf{Y}_{1:k} = \{\mathbf{Y}_1, \dots, \mathbf{Y}_k\}$ are the observations (images in the case of visual tracking) up to and including time *k*. By Bayesian Inference [5]:

$$p(\mathbf{X}_k \mid \mathbf{Y}_{1:k}) = \lambda_k p(\mathbf{Y}_k \mid \mathbf{X}_k) \int p(\mathbf{X}_k \mid \mathbf{X}_{k-1}) p(\mathbf{X}_{k-1} \mid \mathbf{Y}_{1:k-1}) d\mathbf{X}_{k-1}$$
(1)

By applying the Monte Carlo principle, the distribution can be represented by discrete samples through particle filtering. The N samples (particles) are drawn from a proposed distribution $q(\mathbf{X}_{i}^{(k)})$ $\mathbf{X}_{i}^{(k-1)}, \mathbf{Y}_{k}$ $i = 1, 2, \dots, N$ and each of them will later be assigned a weight. Since measurement happens after sampling, in practice the \mathbf{Y}_k is an unavailable factor of the proposal distribution. Generic particle filter tackles the problem by simply ignoring this factor when drawing samples, solely depending on $p(\mathbf{X}_{i}^{(k)} \mid$ $\mathbf{X}_{i}^{(k-1)}$), i.e. the dynamical model. The dynamical models may help the samples head for positions relatively near the solution in regard to their starting points, but then the CONDENSATION or generic particle filter diffuses them in a blind way and then the success of tracking becomes a matter of luck as there are no available clues. Our motivation to propose the DE-MC particle filter is to utilize the feedback from the measurement function to compensate for the missing \mathbf{Y}_k . According to this information particles can move to the regions where a solution is more possible to be found. Figure 1 illustrates the different results of the two sampling strategies. In the following subsections we will first introduce some background knowledge. Then we will propose the DE-MC particle filter.

2.1. Monte Carlo Markov Chain and Differential Evolution Algorithm

Regardless of in which initial state it starts, a Markov chain (MCMC) will always reach a steady state distribution $p(\mathbf{X})$ if it possesses irreducibility and aperiodicity[]. They guarantee a finite path from each state to every other state with non-zero transition probability, which is the so-called ergodicity. Markov Chain Monte Carlo (MCMC) method takes aim at constructing a MC which has the given target



Fig. 1. Sampling of CONDENSATION and DE-MC particle filter. A: Dynamical Model. B: Gaussian Noise Diffusion C: DE-MC.

distribution as its invariant distribution [10]. Normally we ensure the stationarity by making the chain satisfy the reversibility property (detailed balance):

$$p(\mathbf{X}_k)\mathbf{T}(\mathbf{X}_{k-1} \mid \mathbf{X}_k) = p(\mathbf{X}_{k-1})\mathbf{T}(\mathbf{X}_k \mid \mathbf{X}_{k-1})$$
(2)

A very popular MCMC implementation method is the Metropolis algorithm [10]. According to this, the transition probability of the chain is given by:

$$\mathbf{T}(\mathbf{X}_k \mid \mathbf{X}_{k-1}) = \alpha(\mathbf{X}_{k-1}, \mathbf{X}_k)g(\mathbf{X}_k \mid \mathbf{X}_{k-1})$$
(3)

where $g(\mathbf{X}_{k} | \mathbf{X}_{k-1}) = g(|\mathbf{X}_{k} - \mathbf{X}_{k-1}|)$ is a symmetric random walker proposal distribution and:

$$\alpha(\mathbf{X}_{k-1}, \mathbf{X}_k) = \min(1, \frac{p(\mathbf{X}_k)}{p(\mathbf{X}_{k-1})})$$
(4)

is called the acceptance rate.

The Differential Evolution Algorithm (DE) is an global optimization algorithm for high dimensional state space [13]. Similar to other evolutionary program methods, it is based on evolution theory and competition mechanism. Compared with the Genetic Algorithm, it is defined in real parameter space and hence is simpler to use. Experiments have verified its excellent performance in convergence through comparison with other optimization methods [13].

Assume that a function $f(\mathbf{E})$ is defined over a *D*-dimensional state space ε . Assume also that we do not know the analytical form of this function, but can evaluate the value indirectly. We can use the DE algorithm to search the global maximum with an initial population $\mathbf{E}_{n,0}$, $n = 0, 1, \ldots, N - 1$. *N* is the number of population. The DE algorithm generates a new generation of population in time step k+1 according to:

$$\mathbf{E}_{n,k+1}^{*} = \mathbf{E}_{n,k} + \lambda (\mathbf{E}_{r1,k} - \mathbf{E}_{r2,k})$$
(5)

where r1, r2 are random integers drawn from [0,1,2,...,N-1] and mutually different, and $\lambda > 0$. Whether a newly generated state vector will be accepted is dependent on the value evaluated by the target function: if $\mathbf{E}_{i,n+1}^*$ yields a larger function value than $\mathbf{E}_{i,n}$ the state will be updated, otherwise it will be kept intact. Several variants of the DE are available and the use of crossover operator is optional [13].

2.2. DE-MC Algorithm

By examining the characteristics of the MCMC and the DE algorithm we find that they can help each other in search of an optimal



(a) Muller potential surface.

(b) Sampling result for the Muller potential surface (200 samples, 30 iterations).

Fig. 2. The DEMC algorithm simulation results. Black cross marks the starting position of samples.

solution. The acceptance rule in the DE part is controlled by the MCMC acceptance mechanism, whilst the step size and orientation of the random walk of the MCMC part is produced by the DE algorithm. Multiple MCMCs can interact with each other, sharing information with the aid of the DE algorithm, and under the guidance of the DE algorithm, the MCMCs will gradually concentrate on the important regions of the posterior distribution without being trapped around one local extrema for a long time. The DE-MC algorithm is summarized as follows:

Start with a target function $f(\mathbf{E})$ and an initial population $(\mathbf{E}_{0,0}, \mathbf{E}_{1,0}, \ldots, \mathbf{E}_{N-1,0})$, whose members are *D*-dimensional vectors.In the *k*th iteration:

- 1. For each member of the population $\mathbf{E}_{n,k-1}$, $n = 0, 1, \dots, N-1$, randomly choose two integers r1 and r2 so that $r1 \neq r2 \neq n$.
- 2. Create a new member $\mathbf{E}_{n,k}^*$ by:

$$\mathbf{E}_{n,k}^{*} = \mathbf{E}_{n,k-1} + \lambda (E_{r1,k-1} - E_{r2,k-1}) + g.$$
(6)

 λ is a scalar whose value is found to be optimal when: $\lambda = 2.38/\sqrt{2D}$ and g is drawn from a symmetric distribution with small variance compared to that of **E**.

- 3. Compute the ratio: $R = \frac{f(\mathbf{E}_{n,k}^*)}{f(\mathbf{E}_{n,k-1})}$.
- 4. Choose a number h from U(0,1), if R > h, $\mathbf{E}_{n,k} = \mathbf{E}_{n,k}^*$; otherwise $\mathbf{E}_{n,k} = \mathbf{E}_{n,k-1}$.
- 5. Repeat steps 2-5 for iteration k + 1 until a convergence or a preset end point is reached.

It is easy to verify that detailed balance is retained with the DE-MC algorithm. We run simulations on Muller potential surface to illustrate the performance of the DE-MC algorithm in approximating target distributions. In the literature [6], it is shown that under the same settings the samples are still trapped in the starting cost basin even after 6000 iterations with the standard dynamic sampling, while from Figure 2 we can see the important regions of target distribution are well sampled by 200 samples with the DE-MC algorithm after only 30 iterations.

2.3. DE-MC Particle Filter

Based on the Differential Evolution-Monte Carlo (DE-MC) algorithm, we propose a novel sequential Monte Carlo sampling approach, namely the DE-MC particle filter. The DE-MC particle filtering iteration at time step k can be summarized as follows:

Starting from the set of particles which are the filtering result of time step k - 1: $\{\mathbf{X}_{k-1}^{(i)}, w(\mathbf{X}_{k-1}^{(i)})\}_{i=1}^{N}$.

- 1. Selection: select a set of samples $\{\mathbf{X}_{k}^{(i)}\}_{i=1}^{N}$ from $\{\mathbf{X}_{k-1}^{(i)}\}_{i=1}^{N}$ with the probability proportional to $w(\mathbf{X}_{k-1}^{(i)})$.
- 2. **Prediction and Measurement:** Apply a constant velocity dynamical model to the samples: $\mathbf{X}_{k}^{(i)-} = \mathbf{X}_{k}^{(i)} + \mathbf{V}_{k-1}$, where \mathbf{V}_{k-1} is the velocity vector computed in time step k-1. The particle set $\{\mathbf{X}_{k}^{(i)-}\}_{i=1}^{n}$ then acts as the initial population for a *T*-iteration DE-MC processing. The processing follows the procedure we listed in the preceding subsection. The fitness function is the image likelihoods in the case of visual tracking (see next section). Then the weights of particles are subject to update by the DE-MC. For Equation (6) in step 3 of the DE-MC algorithm we choose $g \sim U(-c\sigma, c\sigma)$ and $\sigma = [\sigma_0, \sigma_1, \dots, \sigma_{D-1}]^T$ is a vector with the elements equal to the standard deviations of the elements in **X**. Normal distribution can be used here instead of uniform distribution. c is a small number which can be flexibly chosen. Also in the same equation, the optimal value of λ is now determined by $\lambda = (1 c) \times \frac{2.38}{\sqrt{2D}}$. At the end of this step, we take the output population as the particle set of current time step: $\{\mathbf{X}_{k}^{(i)}, w(\mathbf{X}_{k}^{(i)})\}_{i=1}^{N}$.
- 3. Representation and Velocity Updating: Estimate the state at time step k as: $\mathbf{X}_k = \arg \max_{\mathbf{X}_k^{(i)}; i=1,...,N} w(\mathbf{X}_k^{(i)})$ and update the velocity vector of current time step: $\mathbf{V}_k = \mathbf{X}_k \mathbf{X}_{k-1}$.

We adopt an adaptive strategy inspired by [3] to help determine the value of σ in step 2 by:

$$\sigma_{k,t} \propto \frac{1}{\sum_{i=1}^{n} (w(\mathbf{X}_{k,t-1}^{(i)}))^2}$$
(7)

where *t* denotes the DE-MC iteration index. The right hand side of the equation is called the survival diagnostic. The step size of random jumping for current DE-MC iteration is reduced if the survival rate of the last DE-MC iteration is high and is inflated otherwise.

The most evident improvement of the DE-MC particle filter with respect to the CONDENSATION algorithm is that the prediction (sampling) step and the measurement step are now integrated together instead of functioning separately. As a result, the observation lost in traditional particle filters is retrieved. The dynamical model is still necessary to accomplish part of the sampling task, however it is the DE-MC algorithm that will make the proposed algorithm work more effectively than its predecessors.

3. IMAGE LIKELIHOOD

The image likelihood or observation model is used to measure the correctness of particles (hypothesis state vectors). It is built by comparing the resemblance of the hypothesis images to the ground-truth ones. A 3D articulated human body model is adopted for generating hypothesis images. It has 14 body segments which are modeled as truncated cones or spheres. The 24 DOFs (Degrees of Freedom) associated with the model, including joint angles and global translation and rotation parameters, give a description of the human pose and form the vector \mathbf{X} . Joint angle limits are imposed as constraints for the reason of physical consistency. After solving necessary parameters through initialization and calibration, hypothesis images can be produced by the perspective projection of the human model.



Fig. 3. Snapshots of the tracking results for walking, hopping, running and jumping sequences, respectively

We consider three image features for the evaluation of image likelihoods. Here we only give a very brief description of them. Details are available in our earlier work [7]. In the following denotations, S, C, and B represent silhouette, color histogram and boundary, respectively; the subscript G and H denote ground-truth and hypothesis, respectively:

Area of silhouette We count the pixels in the overlapped region of S_G and S_H and use their ratio as the density $p_1(\mathbf{Y}_k \mid \mathbf{X}_k^{(i)})$

Color histogram We compute the Bhattacharyya distance between C_G and C_H as $p_2(\mathbf{Y}_k \mid \mathbf{X}_k^{(i)})$. Here we propose to use a weighting scheme for calculation [7].

boundary descriptor We compute the Euclidean distance between the Fourier Descriptors of B_G and B_H as $p_3(\mathbf{Y}_k \mid \mathbf{X}_k^{(i)})$.

The overall observation model is set as the multiplication of the three individual density functions.

4. EXPERIMENTS

We carry out experiments on four monocular human motion video sequences: Most experiments are conducted on Sequence 1, which is a walking sequence; Sequence 2, 3, 4 are hopping sequence, jumping sequence and running sequence, respectively. Their lengths are between 1.2s and 2.0s and the size is 720×480 . Monocular video sequences offer a huge challenge for any human motion tracking algorithm. To our knowledge there are very few successful tracking results reported under a similar situation in the literature. Many preceding works ease the tracking task by using multiple cameras [3], restricting application scenario to a particular motion type [11] [12] and having the subjects wear clothes which have distinguished feature for different body parts [4], while our experiments are conducted in a quite general situation. The tracking results are shown in Figure 3. For tracking Sequence 1, a 7-layer (or iteration) DE-MC particle filter is used (Here we use "layer" to represent a DE-MC iteration). The number of particles is 500, which can lead to a satisfactory balance between the reliability and the computational cost of



Fig. 4. Comparison of the tracking results by using the DE-MC particle filter with that by using other popular human tracking methods. (a) The images are the results obtained by the CONDENSATION (the second row), the annealed particle filter (the third row) and the DE-MC particle filter(the fourth row), respectively. (b) The quantitative analysis in terms of the average sample weights.

the tracker. For Sequence 2, 3 and 4, a 9-layer DE-MC particle filter with 600 particles is used. Although some tracking errors can be found due to motion blur, the proposed tracker achieves satisfactory performance. In Figure 4, we compare the tracking result obtained by a 7-layer DE-MC particle filter with those obtained by other popular particle filtering-based algorithms. In terms of fair comparison, the experiment is based on almost the same number of measurement function evaluations (DE-MC:3500 times/frame, Condensation and APF: 5000 times/frame) because it is the most time-consuming part for particle filtering. The other factors, such as the initialization result, initial covariance of the state vector and measurement function are given the same settings for all the algorithms. Despite consuming only 70% of the computations spent by the other two algorithms, DE-MC particle filter still outperforms them. Here average sample weights are regarded as a measure of tracker's performance since to a large extent it can reflect how many of the samples are considered "valid". We also compare the tracking results for Sequence 1 obtained by using 2-layer, 5-layer and 7-layer DE-MC particle filters, in Figure 5. The numbers of samples per layer for these three trackers are 1750, 700, and 500, respectively.

5. CONCLUSIONS

We have developed a novel human motion tracking method, the DE-MC particle filter, based on the fact that the posterior depends on both the previous state and the current observation in a visual tracking application. This method improves the traditional particle-filterbased approach by building a path which connects sampling and measurement. This leads to a more reasonable approximation to the proposal distribution and hence an greatly improved accuracy for tracking. Future work will include automatic initialization of the human body model parameters and the application of more prior knowledge about human kinematics in tracking.

6. REFERENCES

[1] C. J. F. T. Braak, "Genetic Algorithms and Markov Chain Monte Carlo: Differential Evolution



Fig. 5. (a) Comparison of the performance of the DE-MC particle filters with 2 layers, 5 layers and 7 layers (from left to right, respectively). (b) The quantitative analysis in terms of the average sample weights.

Markov Chain Makes Bayesian Computing Easy," http://www.biometris.nl/Markov%20Chain.pdf.

- [2] H. Sidenbladh and M. J. Black, "Learning Image Statistics for Bayesian Tracking," *Proc. of Interntional Conference on Computer Vision*, 2001.
- [3] J. Deutscher, A. Blake and I. Reid, "Articulated Body Motion Capture by Annealed Particle Filtering," *Proc. of Interntional Conference on Computer Vision and Pattern Recognition*, 2000.
- [4] C. Sminchisescu and B. Triggs, "Covariance Scaled Sampling for Monocular 3D Body Tracking," *Proc. of Interntional Conference on Computer Vision and Pattern Recognition*, 2001.
- [5] M. Isard and A. Blake, "Condensation: Conditional Density Propogation for Visual Tracking," *International Journal of Computer Vision*, vol. 29, pp. 5-28, January, 1998.
- [6] C. Sminchisescu and B. Triggs, "Hyperdynamics Importance Sampling," *Proc. of European Conference on Computer Vision*, 2002.
- [7] M. Du and L. Guan, "Multi-Information Fusion for Human Motion Tracking by Particle Filter," *Visual Communications* and Image Processing, 2005.
- [8] M. W. Lee, I. Cohen and S. K. Jung, "Particle Filter with Analytical Inference for Human Body Tracking," *Proc. of IEEE Workshop on Motion and Video Computing*, 2002.
- [9] Y. Rui and Y. Chen, "Better Proposal Distributions: Object Tracking using Unscented Particle Filter," *Proc. of Interntional Conference on Computer Vision and Pattern Recognition*, 2001.
- [10] C. Andrieu, N. D. Freitas A. Doucet and M. I. Jorden, "An introduction to MCMC for Machine Learning," *Machine Learning*, vol. 50, pp. 5-43, 2002.
- [11] H. Sidenbladh, M. J. Black and L. Sigal, "Implicit Probabilistic Models of Human Motion for Synthesis and Tracking," *Proc.* of European Conference on Computer Vision, 2002.
- [12] R. Urtasun and P. Fua, "3D Human Body Tracking using Deterministic Temporal Motion Models," *Proc. of European Conference on Computer Vision*, 2004.
- [13] R. Storn, "On the Usage of Differential Evolution for Function Optimization," Proc. of Biennial Conference of the North American Fuzzy Information Processing Society, 1996.