

# Particle Filter Based Monocular Human Tracking with a 3D Cardbox Model and a Novel Deterministic Resampling Strategy

Ziyuan Liu, Dongheui Lee and Wolfgang Sepp

**Abstract**—The challenge of markerless human motion tracking is the high dimensionality of the search space. Thus, efficient exploration in the search space is of great significance. In this paper, a motion capturing algorithm is proposed for upper body motion tracking. The proposed system tracks human motion based on monocular silhouette-matching, and it is built on the top of a hierarchical particle filter, within which a novel deterministic resampling strategy (DRS) is applied. The proposed system is evaluated quantitatively with the ground truth data measured by an inertial sensor system. In addition, we compare the DRS with the stratified resampling strategy (SRS). It is shown in experiments that DRS outperforms SRS with the same amount of particles. Moreover, a new 3D articulated human upper body model with the name 3D cardbox model is created and is proven to work successfully for motion tracking. Experiments show that the proposed system can robustly track upper body motion without self-occlusion. Motions towards the camera can also be well tracked.

## I. INTRODUCTION

Human motion capture is the process in which the configuration of body parts is estimated over time from sensor input. Marker-based human motion capture systems use markers attached to the human body to capture human motion. Despite the good accuracy of marker-based motion capture systems, they are obtrusive and expensive. Many applications, such as surveillance and human-computer interaction applications, however, require the capture system to be markerless, and to capture human motion just by analyzing image sequences. In recent years, these systems have attracted tremendous attention, and they are considered to be an active research area in the future as well.

In this paper, a motion capture system is proposed for upper body motion tracking. The proposed system tracks human motion based on monocular silhouette-matching, and it is built on the basis of a hierarchical particle filter. In order to tackle the high dimensionality of the search space, an efficient *deterministic resampling strategy* (DRS) is applied within the particle filter framework. Using the ground truth data which is obtained by the Moven inertial motion capture suit [18], we evaluate the proposed system quantitatively. In experiments, we compare our DRS with the stratified resampling strategy (SRS) [12]. Experiments show that the proposed system achieves stable real-time tracking of human motion without self-occlusion in 15 fps (frames per second) using 600 particles. Motions towards the camera can also be

well tracked. Additionally, a new 3D articulated human upper body model with the name *3D cardbox model* is created and is proven to work successfully for motion tracking.

The remainder of this paper is structured as follows: in Section II, we review the related work on markerless human motion tracking and human modeling. Subsequently, we describe our system with respect to its two parts: modeling and estimation. In Section III, we introduce the modeling part, where our new human model and other basic components of the modeling part are explained. In Section IV, the estimation part of the proposed system is introduced, where we discuss model initialization and online motion tracking respectively. In Section V, the proposed system is evaluated in experiments, where we compare our DRS with SRS.

## II. RELATED WORK

The main challenge of markerless human motion capturing lies at the high dimensionality of the search space. So far, many approaches have been proposed. Particle filtering [3] samples the configuration space according to stochastic models of the observation. However, the bottleneck of applying particle filtering in markerless motion tracking is that the number of particles needed for a successful tracking increases exponentially with the number of degrees of freedom (DOF) of the underlying human model. According to [2], a realistic human model contains 14 DOF for upper body or 25 DOF for full body, and this results in an extremely high-dimensional search space. In order to address this problem, the annealed particle filter [4] and the partitioned particle filter [5] have been proposed, in which the number of particles needed is effectively reduced, however, the processing speed is slow. Another way to solve the high-dimensionality is to add constraints to human movements, as done in [6]. Nevertheless, the power of such systems is limited due to the incorporated constraints. L. Sigal et. al. [7] apply optimized algorithms to search for a global or local estimation of the pose configuration, but the applicability of the system is limited to situations, in which multiple cameras are used. Gall et. al. [8] combine the stochastic sampling and the optimization to search for a global estimate of the human posture. Despite the good tracking accuracy, their system needs no less than 76 seconds to process one frame. Other systems are based on additional 3D sensor, such as the Swiss Ranger, and employ Iterative Closest Point (ICP) algorithm [9]. Recently, systems using the Kinect sensor have also been proposed [10]. As pointed out by [11], one limitation for most of the existing systems is the lack of quantitative error analysis, which is important for system comparison.

Z. Liu and D. Lee are with Institute of Automatic Control Engineering, Technische Universität München, D-80290, München, Germany. {ziyuan.liu, dhlee}@tum.de

W. Sepp is with Institute of Robotics and Mechatronics, German Aerospace Center (DLR), Germany. wolfgang.sepp@dlr.de

In addition to different tracking methods, there are also several ways to model human body. According to [1], there exist nowadays three main types of human models: 2D shape model [13], 3D volumetric model [14] and 3D surface-based shape model [15]. In 2D models, body parts are modeled as 2D patches. Such models usually do not represent the geometry and kinematics of the human body explicitly. Normally, scaling factors are calculated for 2D models, so that they can better match the input images. In contrast to 2D models, 3D models incorporate physical rotations and translations of the body parts to produce the final body postures. In 3D volumetric models, body parts are modeled as separate rigid objects that are articulated through joints. In 3D surface-based models, the human body is modeled as a single surface which is built by a mesh of deformable polygons.

### III. MODELING

#### A. Physical Model

In this paper, a new 3D articulated human model with the name “3D cardbox model” is proposed. Our 3D cardbox model represents each body part with two or three 2D patches that are perpendicular to each other. A comparison of the 3D volumetric model and our 3D cardbox model is illustrated in Fig. 1. In the 3D cardbox model, the head, upper arm, forearm and waist are represented by two central patches, whereas the torso is modeled by three patches. Compared with the 3D volumetric model, our model has less planes to project while rendering the human model image.

Our human model contains two types of parameters: *posture parameters* and *size parameters*. The posture parameters refer to the kinematic model of our 3D cardbox model, which contains 12 DOF in total; 2 DOF for rotations of each elbow, 2 DOF for rotations of each shoulder, 3 DOF for the orientation of the torso and 1 DOF for the position of the torso. Intuitively, 3 DOF are needed for determining torso position, however, at current implementation, 2 DOF (along the X and Z direction in Fig. 1) are set constant after an initialization step. Thus, we estimate 12 DOF during online motion tracking. Given all the DOF information, the 3D model is mapped to the image plane of a camera observing the scene. Size parameters of our human model reflect the physical dimension of the observed person. These parameters are initialized through offline model initialization. Fig. 2 illustrates all size parameters of our human model.

#### B. Observation Model

In our approach, we observe human motion by the silhouette of the person. Silhouette images are obtained via foreground-background segmentation. Here, the background and the camera position are both assumed static. The brightness statistics of the background is obtained by capturing first a set of images of the background and then using the averaging background method [17]. In this way, the mean brightness  $I_B(x, y)$  and standard deviation  $\sigma_B(x, y)$  of the background is learned. Having built the background model, pixels of subsequent images  $I_t(x, y)$ , whose brightness is

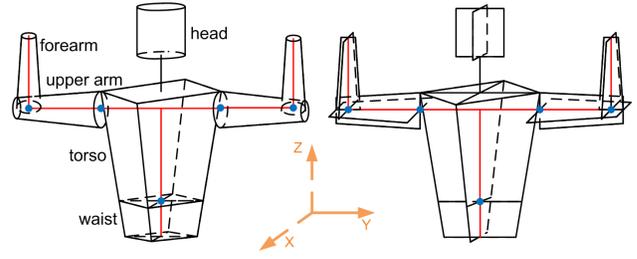


Fig. 1. The 3D volumetric model (left) and the 3D cardbox model (right). Red lines denote skeleton and blue circles denote joints. Coordinate system used for torso position is shown as orange arrows. Axis X is the opposite direction of the optical axis of the camera. Axis Y is horizontally parallel to the camera plane, and axis Z is determined by the right-hand rule.

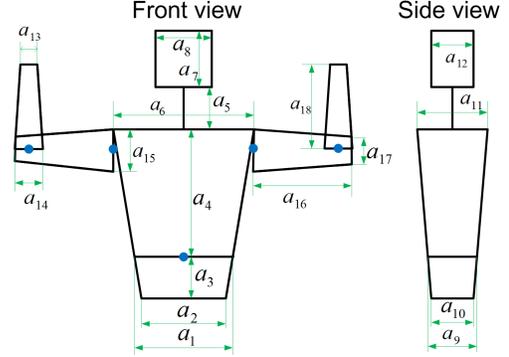


Fig. 2. Size parameters of the 3D cardbox model.

different from that of the learned background, are identified as foreground. The input image is thresholded into a binary silhouette image:

$$f(x, y, I_t) = \begin{cases} 0, & |I_B(x, y) - I_t(x, y)| < 2\sigma_B(x, y) \\ 1, & \text{otherwise} \end{cases} \quad (1)$$

where 0 indicates the background and 1 the foreground. Fig. 3 shows one example of foreground segmentation.

A certain human posture  $\mathbf{d}$  is assigned a cost  $W(\mathbf{d}, I_t)$  by matching the perspectively projected 3D model  $h(x, y, \mathbf{d})$  with the silhouette image  $f(x, y, I_t)$ . The cost denotes the amount of distinct pixels between  $h(x, y, \mathbf{d})$  and  $f(x, y, I_t)$  and is given as:

$$W(\mathbf{d}, I_t) = \sum_{x, y} h(x, y, \mathbf{d}) \oplus f(x, y, I_t) \quad ,$$

$$\text{with } h(x, y, \mathbf{d}) = \begin{cases} 1, & (x, y) \in D \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

where  $D$  denotes the set of image coordinates that are located within the projected human model. The symbol  $\oplus$  denotes the logic XOR function. The function  $h(x, y, \mathbf{d})$  indicates the intensity value of pixel  $(x, y)$  of the projected human model that is generated by the set of posture parameters  $\mathbf{d}$ :

$$\mathbf{d} = [d_1, d_2, \dots, d_{12}]^T \quad , \quad (3)$$

where each element  $d_i$  in  $\mathbf{d}$  refers to one DOF. Fig. 4 shows one example of silhouette matching, where the distinct pixels (right) between the projected human model (left) and the extracted silhouette (middle) are shown in white. In Section IV, we explain how to estimate posture parameters and size parameters by minimizing the cost function in (2).

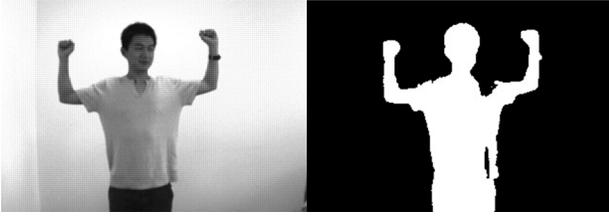


Fig. 3. Foreground segmentation: observation image (left) and extracted silhouette image (right).

#### IV. ESTIMATION

The estimation part of the proposed system contains two phases, i.e. offline model initialization and online motion tracking as described in the following.

##### A. Offline Model Initialization

Before the human model is used for online motion tracking, it must be initialized, which means the body size parameters of the human model are tuned for the observed person. Fig. 2 illustrates the body size parameters that need to be initialized. These parameters are iteratively estimated by local optimization around an initial guess using multiple hypotheses. A hypothesis of size parameters  $\mathbf{A}$  is generated by adding Gaussian noise vector  $\delta_\alpha$  to initial guess  $\mathbf{A}_{\text{ini}}$  of the body size of the 3D model:

$$\mathbf{A} = \mathbf{A}_{\text{ini}} + \delta_\alpha, \delta_\alpha \sim \mathcal{N}(0, \sigma_\alpha^2 E), \quad (4)$$

where  $E$  is the identity matrix. The standard deviation  $\sigma_\alpha$  used to generate the Gaussian noise is 5 cm for each size parameter. Each hypothesis  $\mathbf{A}$  represents one set of possible values of the body size parameters (Fig. 2):  $\mathbf{A} = [a_1, a_2, \dots, a_{18}]^T$ . Because of the large number of parameters, the parameters are estimated in a three-step hierarchical way. Let say  $\mathbf{A} = [\mathbf{O}^1, \mathbf{O}^2, \mathbf{O}^3]^T$ . Size of torso, waist and head in front view  $\mathbf{O}^1 = [a_1, \dots, a_8]^T$ , size of torso, waist and head in side view  $\mathbf{O}^2 = [a_9, \dots, a_{12}]^T$ , and size of arms in front view  $\mathbf{O}^3 = [a_{13}, \dots, a_{18}]^T$  are estimated separately. According to the subset  $\mathbf{O}^j, j = 1, 2, 3$ , that is being estimated, reference images  $I_{\text{ref}}^j$  observing the test person doing corresponding reference postures are captured and used to estimate the size parameters. Reference postures  $\mathbf{d}_{\text{ref}}^j$  of the test person are assumed known. The entire process of offline model initialization is described in Algorithm 1.

##### B. Online Motion Tracking

Once the human model is initialized, the body posture parameters of the observed person are continuously estimated in the online motion tracking phase. We apply a hierarchical particle filter to perform online motion tracking as described



Fig. 4. Silhouette matching: projected human model  $h(x, y, \mathbf{d})$  (left), extracted silhouette  $f(x, y, I_t)$  (middle) and distinct pixels shown in white (right).

---

#### Algorithm 1 Offline Model Initialization

---

**Require:**  $\mathbf{A}_{\text{ini}}$ .  
Generate  $M$  hypotheses for estimating size parameters:  
**for**  $i = 1 : 1 : M$  **do**  
 $\mathbf{A}_i = \mathbf{A}_{\text{ini}} + \delta_\alpha, \delta_\alpha \sim \mathcal{N}(0, \sigma_\alpha^2 E)$ ;  
**end for**  
3-step hierarchical estimation of subsets:  
**for**  $j = 1 : 1 : 3$  **do**  
**while** Initialization accuracy not reached **do**  
Find the best subset  $\tilde{\mathbf{O}}^j$  using corresponding reference posture  $\mathbf{d}_{\text{ref}}^j$   
and reference image  $I_{\text{ref}}^j$ :  
 $\tilde{\mathbf{O}}^j = \underset{\mathbf{O}_i^j, i=1, \dots, M}{\text{argmin}} W(\mathbf{d}_{\text{ref}}^j, I_{\text{ref}}^j)$ ;  
Generate  $M$  new hypotheses of corresponding subset:  
**for**  $i = 1 : 1 : M$  **do**  
 $\mathbf{O}_i^j = \tilde{\mathbf{O}}^j + \delta_\alpha, \delta_\alpha \sim \mathcal{N}(0, \sigma_\alpha^2 E)$ ;  
**end for**  
**end while**  
**end for**

---

in Algorithm 2: first, initialize particles for tracking torso, left arm and right arm separately; then resample these particles with DRS, using Algorithm 4. Here, each particle of the particle filter represents a certain human posture. The general

---

#### Algorithm 2 Online Motion Tracking

---

**Require:** Body size parameters are initialized.  
Initialize particles:  
Set the index of estimation cycle  $k$  to 0,  $k = 0$ ;  
**for**  $i = 1 : 1 : N$  **do**  
 $\mathbf{d}_i^0 = \mathbf{d}_{\text{ref}} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E)$ ;  
**end for**  
Assign each particle  $\mathbf{d}_i^0, i = 1, \dots, N$  a cost using (2);  
**while** New observation image  $I_t$  **do**  
Sample and resample with DRS, as described in Algorithm 4;  
**end while**

---

idea of our hierarchical particle filter is to divide the whole estimation procedure into three steps that are listed below:

- First, estimate the orientation of the torso (3 DOF) and the position of the torso (1 DOF).
- Second, estimate the orientation of the left arm (2 DOF for left shoulder, 2 DOF for left elbow).
- Third, estimate the orientation of the right arm (2 DOF for right shoulder, 2 DOF for right elbow).

Note that the second and third step can be processed parallel.

Particles  $\mathbf{d}_i^0$  of the initialization step are generated by adding Gaussian noise  $\delta_\beta$  to the reference posture  $\mathbf{d}_{\text{ref}}$  from which online motion tracking starts:

$$\mathbf{d}_i^0 = \mathbf{d}_{\text{ref}} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E), \quad i = 1, \dots, N, \quad (5)$$

where  $N$  denotes the total number of the used particles. Here, the superscript is the index of estimation cycle which starts with “0” indicating the initialization. The subscript is the particle index.

##### C. Sampling and Resampling

Before discussing DRS, we introduce first a standard particle filter that uses SRS (stratified resampling strategy).

1) *SRS*: For tracking with SRS, the sampling and resampling part of online motion tracking is executed using SRS, as described in Algorithm 3. In the  $k$ th estimation cycle of SRS, each particle  $\mathbf{d}_i^k, i = 1, \dots, N$ , is assigned a probability  $p_i^k$ :

$$p_i^k = \exp\left(-\frac{W_i^{k2}}{2\sigma_\gamma^2}\right), \quad (6)$$

where  $W_i^k$  denotes the cost of particle  $\mathbf{d}_i^k$ . For the tracking of torso, left arm and right arm,  $\sigma_\gamma$  has different values, as listed in Table I. Different values of  $\sigma_\gamma$  are used for tracking of torso, left arm and right arm, as listed in Table I. The calculated probability  $p_i^k$  is normalized so that the sum of probabilities over all particles becomes 1. The normalized probability for each particle is denoted as  $\omega_i^k$ . Using the normalized probability, we calculate a cumulative probability  $L_i^k$  for each particle  $\mathbf{d}_i^k$  at estimation cycle  $k$  as follows:

$$L_i^k = \sum_{j=1}^i \omega_j^k. \quad (7)$$

### Algorithm 3 SRS

**Require:** Particles are initialized.  
 $\mathbf{d}_1^k = \operatorname{argmin}_{\mathbf{d}_i^{k-1}, i=1 \dots N} W(\mathbf{d}_i^{k-1}, I_t)$ ;  
**for**  $i = 2 : 1 : N$  **do**  
 Generate a random number  $r$  with the uniform distribution  $\mathcal{U}(0, 1)$  according to the stratified sampling [12];  
 $\mathbf{d}_i^k = \mathbf{d}_\psi^{k-1} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E)$ , with  $\psi = \min\{j | r \leq L_j^{k-1}\}$ ;  
 Assign each particle  $\mathbf{d}_i^k, i = 1 \dots N$  a normalized probability and calculate the cumulative probability for each particle using (7);  
**end for**  
 $k++$ ;

2) *DRS*: The standard particle filter with SRS uses a Gaussian distribution for the observation probability density. In this case, the selectivity of particles degrades as the modeling error of the observation model increases, especially when the physical model cannot perfectly match the appearance of the human body. We propose a deterministic resampling strategy (DRS) that generates new particles according to a survival-rate  $e$ , as described in Algorithm 4 and illustrated in Fig. 5. Let  $\tilde{\mathbf{d}}_i^{k-1}$  denote the ranked list of posture parameter with  $W(\tilde{\mathbf{d}}_i^{k-1}, I_t) < W(\tilde{\mathbf{d}}_{i+1}^{k-1}, I_t)$ . The first particle  $\mathbf{d}_1^k$  in the  $k$ th estimation cycle is the same as the best particle in the  $(k-1)$ th estimation cycle  $\tilde{\mathbf{d}}_1^{k-1}$ :

$$\mathbf{d}_1^k = \tilde{\mathbf{d}}_1^{k-1} = \operatorname{argmin}_{\mathbf{d}_i^{k-1}, i=1 \dots N} W(\mathbf{d}_i^{k-1}, I_t). \quad (8)$$

By using the survival-rate  $e$ , first  $eN$  particles survive from the  $(k-1)$ th estimation cycle. The survived particles are added with a random Gaussian noise vector  $\delta_\beta$  so as to produce new particles:  $\mathbf{d}_i^k = \tilde{\mathbf{d}}_i^{k-1} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E)$ ,  $1 < i \leq eN$ . The rest particles are drawn from the neighbourhood of the best particle:  $\mathbf{d}_i^k = \tilde{\mathbf{d}}_1^{k-1} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E)$ ,  $eN < i \leq N$ . The particle with the lowest cost is considered to be the estimation result for each estimation cycle.

### Algorithm 4 DRS

**Require:** Particles are initialized.  
 Rank particles  $\tilde{\mathbf{d}}_i^{k-1}, i = 1, \dots, N$ , according to their costs, then store the ranked particles into the particle list  $\tilde{\mathbf{d}}_i^{k-1}$ ;  
**for**  $i = 1 : 1 : N$  **do**  
**if**  $i = 1$  **then**  
 $\mathbf{d}_1^k = \tilde{\mathbf{d}}_1^{k-1}$ ;  
**else if**  $1 < i \leq e \cdot N$  **then**  
 $\mathbf{d}_i^k = \tilde{\mathbf{d}}_i^{k-1} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E)$ ;  
**else**  
 $\mathbf{d}_i^k = \tilde{\mathbf{d}}_1^{k-1} + \delta_\beta, \delta_\beta \sim \mathcal{N}(0, \sigma_\beta^2 E)$ ;  
**end if**  
 Assign each particle a cost using (2);  
**end for**  
 $k = k + 1$ ;

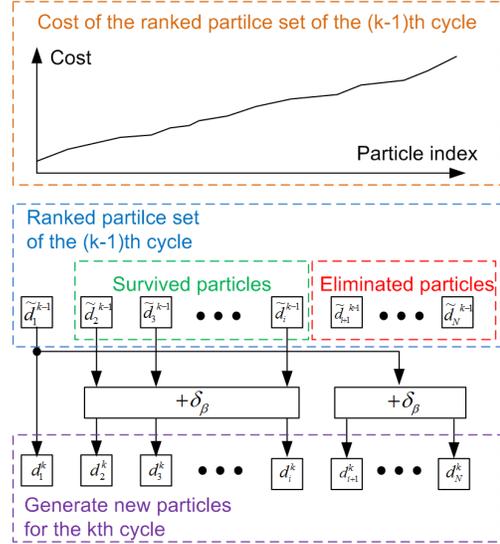


Fig. 5. Sample and resample with DRS.

## V. EXPERIMENTS AND DISCUSSIONS

The experiments are performed with a BumbleBee Stereo Camera BB-COL-20, whereas only one camera is used. A quad-core AMD Phenom™ II X4 945 (3.0 GHz) processor is used as CPU. However, the current implementation is single-threaded. The “Sequential Monte Carlo Template Class” [16] library in C++ is used as the framework for implementing particle filter. The OpenCV library [17] is used for image processing and camera calibration. The resolution of input images is  $320 \times 240$ .

For the choice of the survival-rate  $e$ , imagine two extreme situations. In an extreme case of  $e \approx 0$ , the procedure becomes a greedy local optimization, which redistributes the particles just around the best one. If the observed human motion is fast, the estimation tends to get lost due to the limited search range. In the case of  $e = 1$ , particles are

TABLE I  
PARAMETERS USED FOR ALL EXPERIMENTS.

	Torso	Left arm	Right arm
$N$	200	200	200
$e$	0.2	0.4	0.4
$\sigma_\beta$ for rotation (radian)	0.1	0.25	0.25
$\sigma_\beta$ for translation (cm)	2	-	-
$\sigma_\gamma$ (pixel)	8000	4000	4000

just distributed randomly without convergence. Thus, the reasonable choice of  $e$  should be a compromise of these two cases, which ensures a big search range and a satisfactory accuracy. The used values are summarized in Table I.

Motion used in the following experiments consists of a sequence of three motions: “wave hands” (Fig. 10), “bend aside” (Fig. 11) and “bow forward” (Fig. 12). These figures show results of online motion tracking using DRS with 600 particles ( $N = 600$ ). One can see that the proposed system can stably track human motions without self-occlusion. With 600 particles, we achieve real-time tracking in 15 fps.

#### A. Ground Truth Evaluation

The Movens inertial motion capture suit uses 17 inertial sensors and an advanced articulated human model to track human motion. It is utilized to measure the underlying body posture parameters that is considered to be the ground truth of experiments. The Movens system provides Cartesian positions of the body segments. For the comparison, the tracking result in the joint space of the proposed method is converted to Cartesian position of corresponding body parts by forward kinematics using the estimated body size parameters. Here, 3D positions of shoulder, elbow and hand are compared. It should be mentioned that we could not achieve accurate comparison of both systems but rough comparison only. The reason is that different human kinematic models are used in both systems and the real positions of corresponding body parts are not defined in the same way.

The systems are synchronized with the image captured by the camera. Human motion tracking is repeated 100 times on an example video. As tracking error, we measure the average 3D distance between the corresponding body parts of the Movens and the proposed system over 100 repetition. The results are shown in Fig. 6 and Fig. 7. DRS achieves better 3D tracking accuracy than SRS in hand and elbow tracking. The tracking accuracy of hand is worse than that of elbows and shoulders, because of large difference of hand position in Movens human model and our human model. The hand in Movens system are located at wrists, whereas the hand in the proposed system are at the end of fists. This difference results in an extra error of 10 cm approximately for hand tracking. Taking into account that the tracking is only based on monocular silhouette-matching, and considering the difference of both systems, the proposed system shows good 3D tracking performance.

#### B. 2D Error and Reproducibility of the System Output

The system output of each estimation cycle refers to the best estimate (particle) that has the lowest cost. In the following, we compare DRS with SRS regarding two aspects of the system output: 2D error and reproducibility.

1) *2D error*: 2D error of the system output is represented by its cost. Fig. 8 depicts the mean of the cost of the system output over 100 runs. The figure shows that DRS outperforms SRS with respect to 2D error, although the same number of particles are used in both cases. Since we track human motion just based on silhouette-matching, the DRS is a better choice compared with SRS.

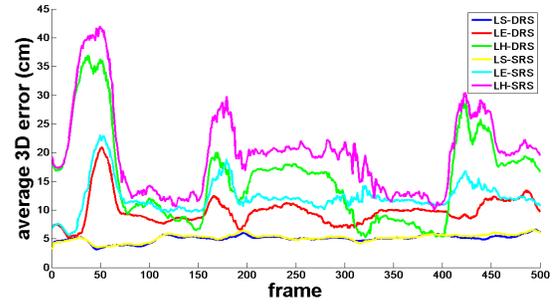


Fig. 6. Average 3D error over 100 repetition: left shoulder (LS), left elbow (LE) and left hand (LH).

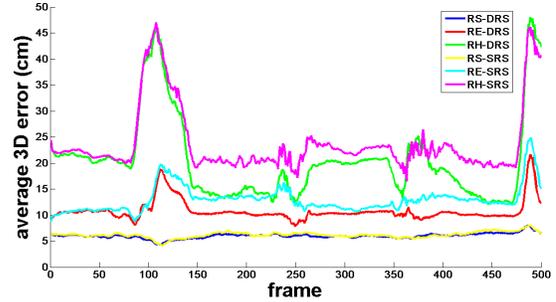


Fig. 7. Average 3D error over 100 repetition: right shoulder (RS), right elbow (RE) and right hand (RH).

2) *Reproducibility*: Ideally, we expect that the system output should show the same performance on the same input. Here, we test the reproducibility of the system output by measuring the mean and standard deviation of each DOF ( $d_i, i = 1, \dots, 12$ ) of the system output over 100 repetition. One example of the results is shown in Fig. 9, in which one rotational DOF of left elbow is demonstrated. Here, we can see that the mean value (solid lines) is similar in both cases, but the standard deviation (dashed lines) of the system output obtained by DRS is smaller throughout the entire tracking. This result indicates that the system output estimated by DRS is better reproducible than that by SRS. The DRS works in a survival of the fittest paradigm, and it contains no non-linear processing of the error value, which results in reliable tracking with a small number of particles (600 particles for 12 dimensions).

## VI. CONCLUSIONS

This paper presents an upper body motion capture system based on monocular silhouette-matching, which is built on the basis of a hierarchical particle filter. A novel deterministic

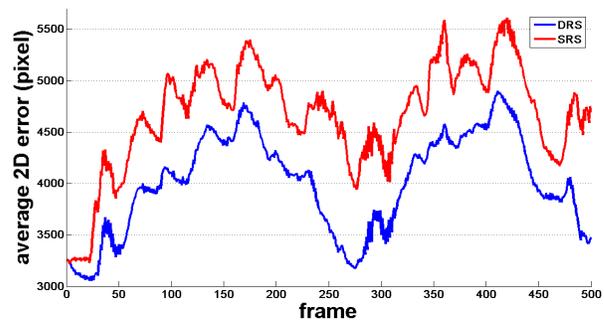


Fig. 8. Average 2D error over 100 runs

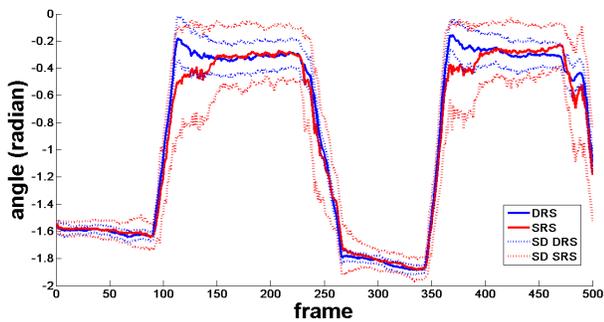


Fig. 9. Reproducibility test: one rotational parameter of left elbow, dashed lines are corresponding standard deviation (SD) lines.

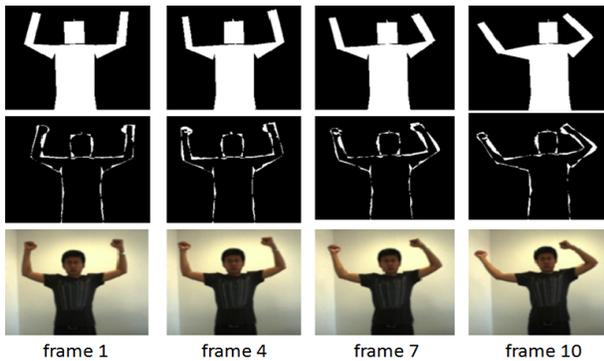


Fig. 10. Online motion tracking with DRS: wave hands. (top) the best estimate; (middle) matching result; (bottom) raw image.

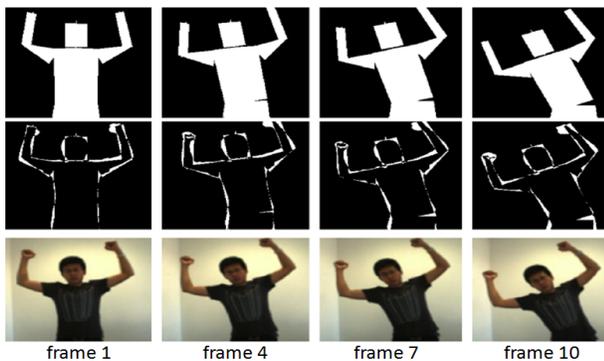


Fig. 11. Online motion tracking with DRS: bend aside. (top) the best estimate; (middle) matching result; (bottom) raw image.

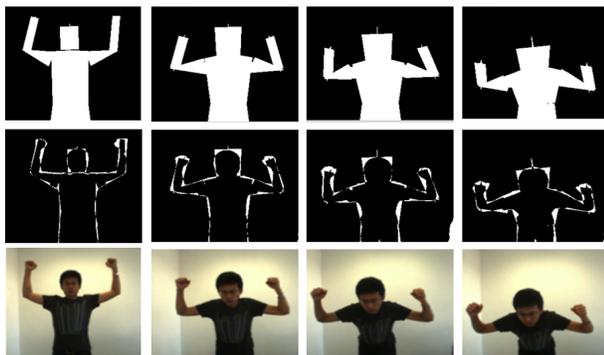


Fig. 12. Online motion tracking with DRS: bow forward. (top) the best estimate; (middle) matching result; (bottom) raw image.

resampling strategy (DRS) is applied so as to explore the search space efficiently. Using the ground truth data collected by Moven system, the tracking performance is quantitatively evaluated. Experimental results show that the DRS outperforms SRS regarding the 3D and 2D tracking error, and reproducibility. A new 3D articulated human upper body model with the name 3D cardbox model is proposed and is proven to work successfully for motion tracking. Compared with the 3D volumetric model, our model has less planes to project while rendering the human model image, which leads to less computational cost. The proposed system achieves a stable real-time tracking of human motion in 15 fps, using 600 particles. It is worth mentioning that motions towards the camera, such as “bow forward”, is also well tracked by the proposed system.

## REFERENCES

- [1] R. Poppe, “Vision-based human motion analysis: An overview”, *Computer Vision and Image Understanding*, 2007, pp. 4-18.
- [2] P. Azad et al., “Stereo-based Markerless Human Motion Capture for Humanoid Robot Systems”, *IEEE International Conference on Robotics and Automation*, 2007, pp. 3951-3956.
- [3] P. Azad, T. Asfour and R. Dillmann, “Robust Real-time Stereo-based Markerless Human Motion Capture”, *IEEE-RAS International Conference on Humanoid Robots*, 2008, pp. 700-707.
- [4] J. Deutscher, A. Blake, and I. Reid, “Articulated Body Motion Capture by Annealed Particle Filtering”, *International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000, pp. 2126-2133.
- [5] J. Deutscher, A. Davison, and I. Reid, “Automatic partitioning of high dimensional search spaces associated with articulated body motion capture”, *International Conference on Computer Vision and Pattern Recognition*, 2001, pp. 669-676.
- [6] M. A. Brubaker and D. Fleet, “The kneed walker for human pose tracking”, *IEEE International Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1-8.
- [7] L. Sigal, S. Bhatia, S. Roth, M. Black, and M. Isard, “Tracking loose-limbed people”, *IEEE International Conference on Computer Vision and Pattern Recognition*, 2004, pp. 421-428.
- [8] J. Gall, B. Rosenhahn, T. Brox, and H. P. Seidel, “Optimization and filtering for human motion capture”, *International Journal of Computer Vision*, 2008, pp. 75-92.
- [9] S. Knoop, S. Vacek, and R. Dillmann, “Modeling Joint Constraints for an Articulated 3D Human Body Model with Artificial Correspondences in ICP”, *International Conference on Humanoid Robots*, 2005, pp. 74-79.
- [10] <http://www.openni.org>.
- [11] Z. Li and D. Kulić, “Particle Filter Based Human Motion Tracking”, *International Conference on Control, Automation, Robotics and Vision*, 2010, pp. 555-560.
- [12] G. Kitagawa, “Monte Carlo filter and smoother for non-Gaussian nonlinear state space models”, *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1-25, 1996.
- [13] Y. Huang, T. S. Huang, “Model-based human body tracking”, *International Conference on Pattern Recognition*, 2002, pp. 552-555.
- [14] R. Kehl, L. V. Gool, “Markerless tracking of complex human motions from multiple views”, *Computer Vision and Image Understanding*, vol. 104, no. 2, pp. 190-209, 2006.
- [15] J. Carranza, C. Theobalt, M. A. Magnor, H. P. Seidel, “Free-viewpoint video of human actors”, *ACM Transactions on Computer Graphics* vol. 22, no. 3, pp. 569-577, 2003.
- [16] A. M. Johansen, “SMCTC: Sequential Monte Carlo in C++”. *Journal of Statistical Software*, vol. 30, no. 6, pp. 1-41, 2009.
- [17] G. Bradski, A. Kaehler, *Learning OpenCV*. O’Reilly Media, 2008.
- [18] Xsens Technologies B.V., “Moven User Manual”. 2008.