Real-Time Decentralized Articulated Motion Analysis and Object Tracking From Videos

Wei Qu, *Member, IEEE*, and Dan Schonfeld, *Senior Member, IEEE*

***Abstract*—In this paper, we present two new articulated motion analysis and object tracking approaches: the decentralized articulated object tracking method and the hierarchical articulated object tracking method. The first approach avoids the common practice of using a high-dimensional joint state representation for articulated object tracking. Instead, we introduce a decentralized scheme and model the interpart interaction within an innovative Bayesian framework. Specifically, we estimate the interaction density by an efficient decomposed interpart interaction model. To handle severe self-occlusions, we further extend the first approach by modeling high-level interunit interaction and develop the second algorithm within a consistent hierarchical framework. Preliminary experimental results have demonstrated the superior performance of the proposed approaches on real-world videos in both robustness and speed compared with other articulated object tracking methods.**

***Index Terms*—Articulated motion analysis, Bayesian density propagation, object tracking, video analysis.**

I. Introduction and Related Work

ARTICULATED motion analysis and object tracking from videos has received a significant amount of attention in recent years driven by its wide applications such as human–computer interaction, patient rehabilitation, biomechanics, human activity analysis, computer animation, etc. Articulated object tracking is a challenging task because of the exponentially increased computational complexity in terms of the degrees of freedom of the object and the severe image ambiguities incurred by frequent self-occlusions.

Many approaches have been studied to circumvent the problems inherent in articulated object tracking. Most earlier efforts of articulated motion analysis took advantage of 2-D or 3-D object models [1]. Drummond and Cipolla [2] presented an algorithm which propagated statistics of probability distributions through a kinematic chain to obtain maximum *a posteriori* estimates. Erol *et al.* reviewed the existing hand motion estimation methods in [3]. A unified spatio-temporal articulated model was proposed by Lan and Huttenlocher [4]. Kalman filters have been employed by many researchers to combat occlusions in articulated object tracking [5], [6]. In the context of multiview body tracking, Bregler and Malik [7] proposed to track people with twists and exponential maps. Kehl *et al.* [8] presented an approach based on a volumetric reconstruction and a stochastic meta descent optimization.

Sequential Monte Carlo method or particle filter [9] was demonstrated to be efficient for object tracking in clutter [10] and has also been introduced for articulated motion analysis. Deutsher *et al.* [11] modified the Condensation algorithm [10] by an annealed particle filter. Choo and Fleet [12] described a filter that used hybrid Monte Carlo to obtain efficient samples in high-dimensional spaces. Chang *et al.* [13] proposed an appearance-based particle filter for articulated hand tracking. Although promising and effective in different aspects, the successful application of particle filtering was limited to situations where the dimension of the joint state is relatively small. For high-dimensional state spaces, many algorithms become computationally inefficient and, thus, ineffective. Recently, graphical models such as Bayesian networks have been used to facilitate the analysis of articulated object tracking. Sudderth *et al.* [14] developed a nonparametric belief propagation algorithm based on the regularized particle filter and structured graphical models. Isard combined belief propagation with ideas from particle filtering and proposed a PAMPAS algorithm in [15]. Sigal *et al.* [16] applied the PAMPAS algorithm to a 3-D loose-limbed model for people tracking. Wu *et al.* [17] proposed a mean field Monte Carlo algorithm based on a dynamic Markov network for 2-D articulated body tracking.

In this paper, we present two novel approaches for articulated motion analysis and object tracking from videos, which are referred to as decentralized articulated object tracking (DAOT) and hierarchical articulated object tracking (HAOT). Distinct from the existing approaches, DAOT exploits an innovative decentralized framework based on a graphical model analysis. Facilitated by graph decomposition, we derive a novel Bayesian conditional density propagation rule, which models the interpart interaction of articulated objects explicitly in a consistent Bayesian framework. To efficiently handle the severe self-occlusion problem, HAOT further extends the DAOT approach by modeling high-level interunit interaction and proposes a robust hierarchical framework for articulated object tracking.

The paper is organized as follows. Section II presents the proposed DAOT framework. Section III describes the extended hierarchical framework of the HAOT approach. In Section IV, we provide experimental results.

Manuscript received April 20, 2006; revised March 9, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Zoltan Kato.

W. Qu is with Motorola Labs, Schaumburg, IL 60196 USA (e-mail: wei.qu@motorola.com).

D. Schonfeld is with the Department of Electrical and Computer Engineering, University of Illinois at Chicago, Chicago, IL 60607 USA (e-mail: dans@uic.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2007.899619

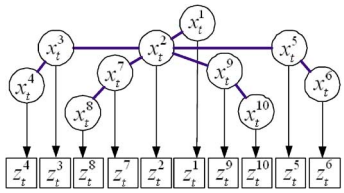

Fig. 1. Graphical model for an articulated object.

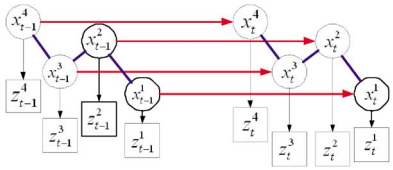

Fig. 2. Dynamical graphical model for articulated motion analysis.

II. Decentralized Framework for Articulated Motion Analysis and Object Tracking

A. Articulated Object Representation

An articulated object can be represented by a graphical model such as shown Fig. 1. It has two layers: the hidden state layer (circle nodes) and the observation layer (square nodes). Each circle node corresponds to a *part* of the articulated object. For example, considering a human body, a *part* can be the torso, or a thigh, etc. The undirected links represent physical constraints among different articulated parts. Each individual part is associated with its observation. The directed link from a part's state to its associated observation represents the local observation likelihood.

Instead of using the joint state representation for the whole articulated object, we denote the state of each part at time t by $\mathbf{x}_t^i$, where $i = 1, \ldots, M$ is the index of parts. $\mathbf{x}_t^i$ can be the part's model or motion parameters. For example, in our implementation, since we adopt a cardboard model [1] for each part, the state is chosen as $\mathbf{x} = (cx, cy, a, b, \theta)$ where (cx, cy) is a characteristic point, such as the center or a joint; a, b are the width and height; θ is the rotation angle of the cardboard around the characteristic point with respect to the Y axis. For a 3-D articulated object, other models such as the loose-limbed body model [16] can be used instead. Moreover, we denote the observation of $\mathbf{x}_t^i$ by $\mathbf{z}_t^i$, the set of all history states up to time t by $\mathbf{x}_{0:t}^i$ where $\mathbf{x}_0^i$ is the initialization prior, the set of all observations up to time t by $\mathbf{z}_{1:t}^i$. Furthermore, different parts are obviously not independent. Each part only interacts with its neighbor. We denote the neighborhood parts of $\mathbf{x}_t^i$ by $\mathcal{N}(i)$, the joint state of all neighbors by $\mathbf{X}_t^{\mathcal{N}(i)} = \{\mathbf{x}_t^j, j \in \mathcal{N}(i)\}$, the observations associated with $\mathbf{X}_t^{\mathcal{N}(i)}$ by $\mathbf{Z}_t^{\mathcal{N}(i)} = \{\mathbf{z}_t^j, j \in \mathcal{N}(i)\}$.

In order to describe the motion of an articulated object, we accommodate the state dynamics by a dynamical graphical model such as shown in Fig. 2. It contains two consecutive time frames. The directed links between consecutive states represents the dynamics which is assumed as a *Markov* chain. Directly analyzing the dynamical graphical model is challenging due to the complicated correlations and mixed undirected/directed links. In order to facilitate the analysis and achieve real-time implementation,

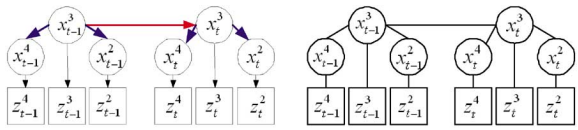

Fig. 3. Graphical model decomposition. (a) Decomposed graphical model for part 3 in Fig. 2. (b) Corresponding moral graph.

we adopt a decentralized framework for each part instead of analyzing the articulated object as a whole. Therefore, we perform a graphical decomposition based on the following steps. 1) The original graphical model is decomposed into M models, each of which formulates one part of the articulated object. 2) All non-neighboring state nodes and their observations corresponding to the current analyzed part's state are removed. That is, only the neighboring state nodes, their associated observations, and the directed links between them are preserved. 3) Each undirected link between two neighboring state nodes is further decomposed into a couple of directed links. The direction is chosen to point from the current part's state to other states. For example, Fig. 3(a) shows the decomposition result for part 3 in Fig. 2. Although the directed links of other parts' dynamics are neglected for simplicity when analyzing a particular part, this information is not lost but is considered when analyzing the other parts. Thus, when using M simultaneous trackers, one tracker per decomposed model for each part, all M decomposed models together capture all of the information (nodes and links) of the original graphical model. Furthermore, since each submodel preserves the correlation between neighboring state nodes, the proposed framework is not a simple product of M submodels for the original model, which distinguishes this decomposition from the use of M independent trackers.

The decomposed graphical models are *directed acyclic independence graphs* [18, pp. 56–82]. By exploiting the *separation theorem* [18] to the associated *moral graph* [18] such as shown Fig. 3(b), it is easy to verify the following *markov properties* [18], i.e., conditional independence properties, for each decomposed graphical model

$$p\left(\mathbf{z}_t^i|\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{x}_{0:t-1}^i, \mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}\right) = p\left(\mathbf{z}_t^i|\mathbf{x}_t^i\right) \tag{1a}$$

$$p\left(\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}|_{0:t-1}^i, \mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}\right) = p\left(\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}|\mathbf{x}_{0:t-1}^i\right) \tag{1b}$$

$$p\left(\mathbf{Z}_t^{\mathcal{N}(i)}|\mathbf{x}_t^i, \mathbf{x}_{0:t-1}^i\right) = p\left(\mathbf{Z}_t^{\mathcal{N}(i)}|\mathbf{x}_t^i\right) \tag{1c}$$

$$p\left(\mathbf{x}_t^i|\mathbf{x}_{0:t-1}^i\right) = p\left(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i\right) \tag{1d}$$

$$p\left(\mathbf{Z}_t^{\mathcal{N}(i)}|\mathbf{x}_t^i\right) = \prod_{j \in \mathcal{N}(i)} p\left(\mathbf{z}_t^j|\mathbf{x}_t^i\right) \tag{1e}$$

$$p\left(\mathbf{z}_t^j|\mathbf{x}_t^j, \mathbf{x}_t^i\right) = p\left(\mathbf{z}_t^j|\mathbf{x}_t^j\right). \tag{1f}$$

Properties (1b) and (1c) indicate that the observations in different time frames are conditionally independent given the states. Properties (1a) and (1f) can be interpreted to state that a part's observation is conditionally independent of other state nodes, given the parent state node. Property (1d) is consistent

with the first-order hidden *Markov* property usually imposed in conventional Bayesian tracking on the state dynamics. Property (1e) states that, given a part's state, its neighboring observations are conditionally independent.

B. Bayesian Conditional Density Propagation

The graphical model of articulated objects presented in the previous section has the potential to be used in different applications of motion analysis such as estimation, prediction, and filtering. In this section, we formulate the motion estimation problem. In other words, given the observations, we want to determine the underlying object state. We will use multiple trackers, one tracker per object part, simultaneously for the whole articulated object. This is achieved in a decentralized scheme by inferring the posterior $p(\mathbf{x}^i_{0:t}|\mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t})$ for each decomposed graphical model associated with an object part

$$
\begin{aligned}
&p\left(\mathbf{x}^i_{0:t}|\mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t}\right)\\
&\quad= \frac{p\left(\mathbf{z}^i_t|\mathbf{x}^i_t, \mathbf{Z}^{\mathcal{N}(i)}_t, \mathbf{x}^i_{0:t-1}, \mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)}{p\left(\mathbf{z}^i_t, \mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)}\\
&\qquad\times p\left(\mathbf{x}^i_t, \mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{x}^i_{0:t-1}, \mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)\\
&\qquad\times p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right) \qquad (2)\\
&\quad= \frac{p\left(\mathbf{z}^i_t|\mathbf{x}^i_t\right) p\left(\mathbf{x}^i_t, \mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{x}^i_{0:t-1}\right)}{p\left(\mathbf{z}^i_t, \mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)}\\
&\qquad\times p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)\\
&\quad= \frac{p\left(\mathbf{z}^i_t|\mathbf{x}^i_t\right) p\left(\mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{x}^i_t, \mathbf{x}^i_{0:t-1}\right) p\left(\mathbf{x}^i_t|\mathbf{x}^i_{0:t-1}\right)}{p\left(\mathbf{z}^i_t, \mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)}\\
&\qquad\times p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right) \qquad (3)\\
&\quad= \frac{1}{k_t} p\left(\mathbf{z}^i_t|\mathbf{x}^i_t\right) p\left(\mathbf{Z}^{\mathcal{N}(i)}_t|\mathbf{x}^i_t\right) p\left(\mathbf{x}^i_t|\mathbf{x}^i_{t-1}\right)\\
&\qquad\times p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right) \qquad (4)\\
&\quad= \frac{1}{k_t} p\left(\mathbf{z}^i_t|\mathbf{x}^i_t\right) \prod_{j\in\mathcal{N}(i)} p\left(\mathbf{z}^j_t|\mathbf{x}^i_t\right) p\left(\mathbf{x}^i_t|\mathbf{x}^i_{t-1}\right)\\
&\qquad\times p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right) \qquad (5)\\
&\quad= \frac{1}{k_t} p\left(\mathbf{z}^i_t|\mathbf{x}^i_t\right) \prod_{j\in\mathcal{N}(i)} \int p\left(\mathbf{z}^j_t|\mathbf{x}^j_t, \mathbf{x}^i_t\right) p\left(\mathbf{x}^j_t|\mathbf{x}^i_t\right) d\mathbf{x}^j_t\\
&\qquad\times p\left(\mathbf{x}^i_t|\mathbf{x}^i_{t-1}\right) p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right)\\
&\quad= \frac{1}{k_t} p\left(\mathbf{z}^i_t|\mathbf{x}^i_t\right) \prod_{j\in\mathcal{N}(i)} \int p\left(\mathbf{z}^j_t|\mathbf{x}^j_t\right) p\left(\mathbf{x}^j_t|\mathbf{x}^i_t\right) d\mathbf{x}^j_t\\
&\qquad\times p\left(\mathbf{x}^i_t|\mathbf{x}^i_{t-1}\right) p\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right). \qquad (6)
\end{aligned}
$$

In (2), we apply the *Markov* properties (1a) and (1b). The denominator in (3) can be regarded as a normalization constant k_t since it is not related to the state. In (4), we use the properties (1c) and (1d). In (5), the *Markov* property (1e) is applied. In (6), the property (1f) is used.

The above Bayesian stochastic formulation explicitly models the physical interaction among neighboring parts of an articulated object. Clearly, the posterior of part i at time t is affected by five factors: 1) the posterior $p(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1})$ in the previous frame at time $t-1$; 2) the local likelihood $p(\mathbf{z}^i_t|\mathbf{x}^i_t)$ of the analyzed part i; 3) the dynamics $p(\mathbf{x}^i_t|\mathbf{x}^i_{t-1})$; 4) the "interpart interaction" density $p(\mathbf{x}^j_t|\mathbf{x}^i_t)$, which models the prior physical interaction between part i and its neighboring parts $j \in \mathcal{N}(i)$; and 5) the local likelihood $p(\mathbf{z}^j_t|\mathbf{x}^j_t)$ of neighboring part j, which can be regarded as a weighted bias of the associated interpart interaction $p(\mathbf{x}^j_t|\mathbf{x}^i_t)$.

C. Sequential Monte Carlo Approximation

In this section, we use the sequential Monte Carlo (SMC) method [9] to approximate the generic Bayesian updating framework presented in the previous section. We refer to this algorithm as DAOT.

The basic idea of SMC approximation is to use a weighted sample set $\{\mathbf{x}^{i,n}_{0:t}, w^{i,n}_t\}^{N_s}_{n=1}$ to estimate the posterior density, which is $p(\mathbf{x}^i_{0:t}|\mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t})$ in our formulation, where $\{\mathbf{x}^{i,n}_{0:t}, n = 1, \ldots, N_s\}$ are the samples, $\{w^{i,n}_t, n = 1, \ldots, N_s\}$ are the associated normalized weights, and $\sum_n w^{i,n}_t = 1$. According to the *importance sampling theory* [9], it is possible to generate the samples $\mathbf{x}^{i,n}_{0:t}$ from an importance density $q(\cdot)$ with associated importance weights

$$
w^{i,n}_t \propto \frac{p\left(\mathbf{x}^i_{0:t}|\mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t}\right)}{q(\cdot)}. \qquad (7)
$$

For the sequential case, if the importance density $q(\cdot)$ is chosen to factorize such that

$$
\begin{aligned}
&q\left(\mathbf{x}^i_{0:t}|\mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t}\right)\\
&\quad= q\left(\mathbf{x}^i_t|\mathbf{x}^i_{0:t-1}, \mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t}\right) q\left(\mathbf{x}^i_{0:t-1}|\mathbf{z}^i_{1:t-1}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t-1}\right) \qquad (8)
\end{aligned}
$$

then by substituting (6) and (8) into (7), we have

$$
\begin{aligned}
w^{i,n}_t \propto w^{i,n}_{t-1} &\frac{p\left(\mathbf{z}^i_t|\mathbf{x}^{i,n}_t\right) p\left(\mathbf{x}^{i,n}_t|\mathbf{x}^{i,n}_{t-1}\right)}{q\left(\mathbf{x}^{i,n}_t|\mathbf{x}^{i,n}_{0:t-1}, \mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t}\right)}\\
&\times \prod_{j\in\mathcal{N}(i)} \left\{\sum_{l=1}^{N^j_s} p(\mathbf{z}^j_t|\mathbf{x}^{j,l}_t) p(\mathbf{x}^{j,l}_t|\mathbf{x}^{i,n}_t)\right\}. \qquad (9)
\end{aligned}
$$

In (9), the integral has been approximated by a summation, where N^j_s is the total number of samples of part j. The density $p(\mathbf{x}^{j,l}_t|\mathbf{x}^{i,n}_t)$ models the interaction between two neighboring parts' samples $\mathbf{x}^{i,n}_t$ and $\mathbf{x}^{j,l}_t$. The local likelihood $p(\mathbf{z}^j_t|\mathbf{x}^{j,l}_t)$ acts as a weight to the associated interaction. They work together to constrain the neighboring parts and prevent different parts of the articulated object from separating over time. The dynamics $p(\mathbf{x}^i_t|\mathbf{x}^i_{t-1})$ can be estimated using a random walk model [19], a second-order constant acceleration model [17], etc. For the local likelihood, we use the same models as those used in conventional Bayesian tracking [19], [20]. Estimation of the interpart interaction density $p(\mathbf{x}^j_t|\mathbf{x}^i_t)$ and the choice of the importance density $q(\mathbf{x}^i_t|\mathbf{x}^i_{0:t-1}, \mathbf{z}^i_{1:t}, \mathbf{Z}^{\mathcal{N}(i)}_{1:t})$ are not trivial and will be discussed in the following sections.

Isard [15], Sudderth *et al.* [14], and Wu *et al.* [17] have independently developed algorithms to model the interaction of articulated structure with the aid of multiple particle sets. Sigal *et al.* [16] applied the PAMPAS algorithm proposed in [15] with a loose-limbed model for people tracking. These methods share a similarity with our DAOT in that, although in different ways, an object part can "*communicate*" with its neighbors by a "*message*" term which incorporates the prior knowledge of the physical interaction constraints. The main difference between our DAOT and these algorithms is that the methods in [14]–[16] are based on belief propagation and the approach in [17] uses probabilistic variational analysis and mean field iteration, whereas our DAOT is based on *directed acyclic independence graph analysis* [18] and Bayes' rule. Moreover, we propose a new interpart interaction model of articulated constraints, which is presented in the following section.

1) Interpart Interaction Model: The interpart interaction density $p(\mathbf{x}_t^j|\mathbf{x}_t^i)$ models the constraints between analyzed part i and its neighboring part j. Estimation of this density should adapt to different applications and is usually critical in practical implementation. Wu *et al.* [17] proposed a constraint model for the 2-D human body. It is fast and easy to implement, although it does not handle the pose relation between two adjacent parts and, thus, does not provide satisfactory results for self-occlusion as shown in Section IV. Sigal *et al.* [16] proposed a Gaussian mixture model for 3-D human body. In principle, this learning-based model is powerful since it implicitly incorporates different constraint information. However, in practice, due to its consideration of different cues together in one model by a joint state representation, this approach has high computational requirements for model training. Inspired by these methods, we propose an efficient decomposed interpart interaction model. It can be observed that the relative locations and poses of two adjacent parts are independent. Therefore, by temporarily discarding the time index, we have

$$\begin{aligned} &p(\mathbf{x}^j|\mathbf{x}^i) \\ &\quad = p(cx^j, cy^j, a^j, b^j, \theta^j|cx^i, cy^i, a^i, b^i, \theta^i) \\ &\quad = p(cx^j, cy^j|cx^i, cy^i)p(\theta^j|\theta^i)p(a^j, b^j|a^i, b^i) \end{aligned} \quad (10)$$

where we assume that the size of an object part is independent of its location and pose. Without considering the size relation between two parts, $p(a^j, b^j|a^i, b^i)$ becomes uniformly distributed. Thus, we can further simplify (10) to be

$$p(\mathbf{x}^j|\mathbf{x}^i) \propto p(\mathbf{c}^j|\mathbf{c}^i)p(\theta^j|\theta^i) \quad (11)$$

where $\mathbf{c}^i = (cx^i, cy^i), \mathbf{c}^j = (cx^j, cy^j)$ are the coordinates of the characteristic points. $p(\mathbf{c}^j|\mathbf{c}^i)$ models the location interaction of two adjacent parts. As a specific example, we can estimate it by a "*spring-joint*" model similar to [17]

$$\begin{aligned} &p(\mathbf{c}^j|\mathbf{c}^i) \\ &\quad = \frac{1}{2\pi|\Sigma_{\mathbf{c}}|^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{c}^j - \mathbf{c}^i)^T\Sigma_{\mathbf{c}}^{-1}(\mathbf{c}^j - \mathbf{c}^i)\right\} \end{aligned} \quad (12)$$

where $\Sigma_{\mathbf{c}}$ is the covariance matrix of this bivariate normal distribution. In (11), $p(\theta^j|\theta^i)$ models the pose relation of two adjacent parts. It can be estimated either by some prior knowledge in particular applications, or by learning from training data. For better comparison, we used a Gaussian mixture model similar to [16] in the experiments.

TABLE I
PSEUDO-CODE OF DAOT ALGORITHM

```
For  i=1:M
*  Sampling x_t^{i,n} ~ q(x_{0:t}^i | z_{1:t}^i, Z_{1:t}^{N(i)});
*  Calculate the initial weights w_t^{i,n} by Equ.(13)
*  Normalize the weights w_t^{i,n};
*  Temporary estimate  x̂_{t,0}^i = Σ_{n=1}^{N_s} w_t^{i,n} x_t^{i,n};
*  FOR  j=1:i-1
   -  IF  j ∈ N(i)
      // Inter-Part Interaction Weighting
      ⋆  FOR k=1:K
         // Weighting part i
         ◇  Interaction weighting for part i;
         ◇  Update weights w_t^{i,n} by Equ.(17);
         ◇  Normalize the weights w_t^{i,n};
         ◇  Estimate x̂_{t,k}^i = Σ_{n=1}^{N_s} w_t^{i,n} x_t^{i,n} ;
         // Weighting part j
         ◇  Interaction weighting for part j;
         ◇  Update weights w_t^{j,n};
         ◇  Normalize the weights w_t^{j,n};
         ◇  Estimate x̂_{t,k}^j = Σ_{n=1}^{N_s} w_t^{j,n} x_t^{j,n} ;
         ◇  Resample {x_t^{i,n}, w_t^{i,n}}, {x_t^{j,n}, w_t^{j,n}};
      ⋆  END FOR k
   -  END IF
*  END FOR j
END FOR i
```

2) Importance Density Selection: The efficiency of a sequential Monte Carlo-based approach is strongly dependent on the selected importance density $q(\cdot)$. When $q(\cdot)$ is close to the true posterior, the samples are more effective. A natural choice of the importance density which has been widely used in the literature [10], [17] is the state dynamics $p(\mathbf{x}_t^j|\mathbf{x}_t^i)$. In our experiments, we choose this importance density in order to highlight the effectiveness of the proposed framework and for better comparison with other approaches.

3) Pseudo-Code: We summarize the main steps of the proposed DAOT using Sequential Monte Carlo approximation in Table I. We only illustrate the implementation of one object part at one time step. In our experiments, we chose the iteration number of the interpart interaction $K = 2 \sim 5$.

III. Hierarchical Decentralized Framework for Articulated Motion Analysis and Object Tracking

Although the proposed DAOT is very effective in terms of speed and robustness in most cases for articulated object tracking, we found that it has limitations in solving severe self-occlusion. These problems motivated us to propose a more sophisticated HAOT framework.

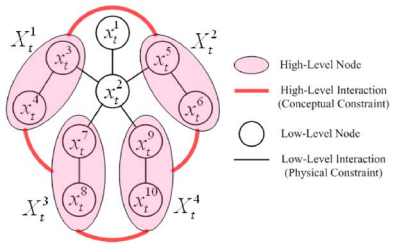

Fig. 4. Hierarchical graphical model for an articulated object.

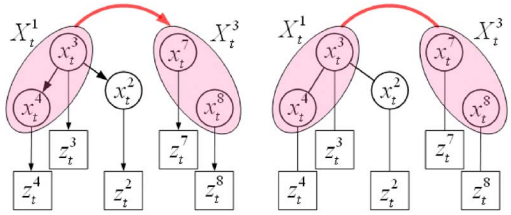

Fig. 5. Decomposed results. (a) The decomposed graphical model for part 3. (b) The corresponding moral graph.

A. Hierarchical Graphical Modeling

The interaction inside an articulated object lies not only in the adjacent *parts* but also some "*high-level*" nonadjacent "*part groups*." For clarity, we define a group of *parts* as a *unit*, which is denoted by $\mathbf{X}_t^{\mathcal{I}}$, where $I = 1, \ldots, \mathcal{M}'$; $\mathcal{M}'$ is the total number of units. For instance, each limb of a human body contains two parts and can, thus, be regarded as a unit. Similar to the model in Section II-A, but considering the "*high-level*" unit interaction as well, we represent the same articulated object in Fig. 1 by a hierarchical graphical model as illustrated in Fig. 4. The observation layer is not shown for simplicity and clarity. Compared with the model in Fig. 1, the difference of this hierarchical model is that it introduces a high-level layer containing big purple ellipse nodes and red curve links. Each big ellipse node corresponds to a unit of the articulated object. The undirected curve links between units represent "*high-level*" interaction. A good example of such kind of interaction appears in the human body. It can be observed that when a person walks normally, there is a specific phase relation among arms and legs. For instance, when the left leg moves forward, the right arm also moves forward while the left arm moves backward. This information can be used to solve self-occlusion. We denote the related units of $\mathbf{X}_t^{\mathcal{I}}$ by $\mathcal{R}(\mathcal{I})$, the joint state of all these related neighboring units by $\mathbf{X}_t^{\mathcal{R}(I)} = \{\mathbf{X}_t^{\mathcal{K}}, \mathcal{K} \in \mathcal{R}(\mathcal{I})\}$, and the corresponding observations by $\mathbf{Z}_t^{\mathcal{R}(I)} = \{\mathbf{Z}_t^{\mathcal{K}}, \mathcal{K} \in \mathcal{R}(\mathcal{I})\}$.

Similar to DAOT, we adopt a decentralized framework and, therefore, decompose the graphical model for each part. Fig. 5(a) presents the decomposition result for part 3 from Fig. 4. Fig. 5(b) shows the corresponding *Moral Graph*. From the decomposed graphs, it is easy to verify the following *Markov Properties* [18] by exploiting the *separation theorem* [18]

$$p\left(\mathbf{z}_t^i|\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}, \mathbf{x}_{0:t-1}^i, \mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right) = p\left(\mathbf{z}_t^i|\mathbf{x}_t^i\right) \tag{13a}$$

$$p\left(\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{0:t-1}^i, \mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right) = p\left(\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{0:t-1}^i\right) \tag{13b}$$

$$p\left(\mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_t^i, \mathbf{x}_{0:t-1}^i\right) = p\left(\mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_t^i\right) \tag{13c}$$

$$p\left(\mathbf{x}_t^i|\mathbf{x}_{0:t-1}^i\right) = p\left(\mathbf{x}_t^i|\mathbf{x}_{t-1}^i\right) \tag{13d}$$

$$p\left(\mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_t^{\mathcal{N}(i)}, \mathbf{X}_t^{\mathcal{R}(\mathcal{I})}, \mathbf{x}_t^i\right) = p\left(\mathbf{Z}_t^{\mathcal{N}(i)}|\mathbf{X}_t^{\mathcal{N}(i)}\right) p\left(\mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}\right) \tag{13e}$$

$$p\left(\mathbf{X}_t^{\mathcal{N}(i)}|\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}, \mathbf{x}_t^i\right) = p\left(\mathbf{X}_t^{\mathcal{N}(i)}|\mathbf{x}_t^i\right) \tag{13f}$$

$$p\left(\mathbf{Z}_t^{\mathcal{N}(i)}|\mathbf{X}_t^{\mathcal{N}(i)}\right) = \prod_{j \in \mathcal{N}(i)} p\left(\mathbf{z}_t^j|\mathbf{x}_t^j\right) \tag{13g}$$

$$p\left(\mathbf{X}_t^{\mathcal{N}(i)}|\mathbf{x}_t^i\right) = \prod_{j \in \mathcal{N}(i)} p\left(\mathbf{x}_t^j|\mathbf{x}_t^i\right) \tag{13h}$$

$$p\left(\mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}\right) = \prod_{\mathcal{K} \in \mathcal{R}(\mathcal{I})} p\left(\mathbf{Z}_t^{\mathcal{K}}|\mathbf{X}_t^{\mathcal{K}}\right) \tag{13i}$$

$$p\left(\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_t^{\mathcal{I}}\right) = \prod_{\mathcal{K} \in \mathcal{R}(\mathcal{I})} p\left(\mathbf{X}_t^{\mathcal{K}}|\mathbf{X}_t^{\mathcal{I}}\right) \tag{13j}$$

$$p\left(\mathbf{Z}_t^{\mathcal{K}}|\mathbf{X}_t^{\mathcal{K}}\right) = \prod_{k \in \mathcal{K}} p\left(\mathbf{z}_t^k|\mathbf{x}_t^k\right). \tag{13k}$$

In addition to the above *Markov properties*, we also make an assumption to facilitate the derivation.

Assumption 1: $p(\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_t^i) = p(\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_t^{\mathcal{I}})$ for all $\mathbf{x}_t^i \in \mathbf{X}_t^{\mathcal{I}}$.

This assumption assumes all the parts $\mathbf{x}_t^i \in \mathbf{X}_t^{\mathcal{I}}$ share the same "relation" with the neighboring units $\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}$, which is the interaction between high-level units $\mathbf{X}_t^{\mathcal{I}}$ and $\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}$. It a reasonable simplification whose soundness can be ascertained a simple analogy. For simplicity, we may use the relation among universities as a substitute. Imagining that $\mathbf{X}_t^{\mathcal{I}}$ represents Harvard University and $\mathbf{X}_t^{\mathcal{R}(\mathcal{I})}$ corresponds to other IV League schools such as Yale University. Let us consider $\mathbf{x}_t^i$ to be a specific student at Harvard. Therefore, without any prior knowledge, we assume that the relation between this specific student and Yale University is equivalent to the relation of any other student at Harvard to Yale University.

B. Hierarchical Bayesian Conditional Density Propagation

Similar to DAOT, we present a Bayesian conditional density propagation framework for each decomposed graphical model. Facilitating by the derived *Markov properties* (13a)–(13j), the posterior density $p(\mathbf{x}_{0:t}^i|\mathbf{z}_{1:t}^i, \mathbf{Z}_{1:t}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t}^{\mathcal{R}(\mathcal{I})})$ for part i can be estimated as follows:

$$\begin{aligned} &p\left(\mathbf{x}_{0:t}^i|\mathbf{z}_{1:t}^i, \mathbf{Z}_{1:t}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t}^{\mathcal{R}(\mathcal{I})}\right) \\ &= \frac{p\left(\mathbf{z}_t^i|\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}, \mathbf{x}_{0:t-1}^i, \mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)}{p\left(\mathbf{z}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)} \\ &\quad \times p\left(\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{0:t-1}^i, \mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right) \\ &\quad \times p\left(\mathbf{x}_{0:t-1}^i|\mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right) \\ &= \frac{p\left(\mathbf{z}_t^i|\mathbf{x}_t^i\right) p\left(\mathbf{x}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{0:t-1}^i\right)}{p\left(\mathbf{z}_t^i, \mathbf{Z}_t^{\mathcal{N}(i)}, \mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{z}_{1:t-1}^i, \mathbf{Z}_{1:t-1}^{\mathcal{N}(i)}, \mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)} \end{aligned} \tag{14}$$

$$\times p\left(\mathbf{x}_{0:t-1}^{i}|\mathbf{z}_{1:t-1}^{i},\mathbf{Z}_{1:t-1}^{\mathcal{N}(i)},\mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)$$
$$=\frac{p\left(\mathbf{z}_{t}^{i}|\mathbf{x}_{t}^{i}\right)p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i},\mathbf{x}_{0:t-1}^{i}\right)p\left(\mathbf{x}_{t}^{i}|\mathbf{x}_{0:t-1}^{i}\right)}{p\left(\mathbf{z}_{t}^{i},\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{z}_{1:t-1}^{i},\mathbf{Z}_{1:t-1}^{\mathcal{N}(i)},\mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)}$$
$$\times p\left(\mathbf{x}_{0:t-1}^{i}|\mathbf{z}_{1:t-1}^{i},\mathbf{Z}_{1:t-1}^{\mathcal{N}(i)},\mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)$$
$$=\frac{p\left(\mathbf{x}_{0:t-1}^{i}|\mathbf{z}_{1:t-1}^{i},\mathbf{Z}_{1:t-1}^{\mathcal{N}(i)},\mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)}{p\left(\mathbf{z}_{t}^{i},\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{z}_{1:t-1}^{i},\mathbf{Z}_{1:t-1}^{\mathcal{N}(i)},\mathbf{Z}_{1:t-1}^{\mathcal{R}(\mathcal{I})}\right)}$$
$$\times p\left(\mathbf{z}_{t}^{i}|\mathbf{x}_{t}^{i}\right)p\left(\mathbf{x}_{t}^{i}|\mathbf{x}_{t-1}^{i}\right)p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i}\right). \quad (15)$$

In (14), we use *Markov properties* (13a) ande (13b). In (15), we apply the properties (13c) and (13d). The density $p(\mathbf{Z}_t^{\mathcal{N}(i)},\mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_t^i)$ in (15) models both the low-level and high-level interaction between part-neighbors $\mathbf{Z}_t^{\mathcal{N}(i)}$ and unit-neighbors $\mathbf{Z}_t^{\mathcal{R}(\mathcal{I})}$ for $\mathbf{x}_t^i$. We further derive this likelihood density as follows:

$$p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i}\right)$$
$$=\int\int p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_{t}^{\mathcal{N}(i)},\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})},\mathbf{x}_{t}^{i}\right)$$
$$\times p\left(\mathbf{X}_{t}^{\mathcal{N}(i)},\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i}\right)d\mathbf{X}_{t}^{\mathcal{N}(i)}d\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}$$
$$=\int\int p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)},\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_{t}^{\mathcal{N}(i)},\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})},\mathbf{x}_{t}^{i}\right)$$
$$\times p\left(\mathbf{X}_{t}^{\mathcal{N}(i)}|\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})},\mathbf{x}_{t}^{i}\right)p\left(\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i}\right)d\mathbf{X}_{t}^{\mathcal{N}(i)}d\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}$$
$$=\int\int p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)}|\mathbf{X}_{t}^{\mathcal{N}(i)}\right)p\left(\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}\right)$$
$$\times p\left(\mathbf{X}_{t}^{\mathcal{N}(i)}|\mathbf{x}_{t}^{i}\right)p\left(\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i}\right)d\mathbf{X}_{t}^{\mathcal{N}(i)}d\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})} \quad (16)$$
$$=\int p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)}|\mathbf{X}_{t}^{\mathcal{N}(i)}\right)p\left(\mathbf{X}_{t}^{\mathcal{N}(i)}|\mathbf{x}_{t}^{i}\right)d\mathbf{X}_{t}^{\mathcal{N}(i)}$$
$$\times\int p\left(\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}\right)p\left(\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{x}_{t}^{i}\right)d\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})} \quad (17)$$
$$=\int p\left(\mathbf{Z}_{t}^{\mathcal{N}(i)}|\mathbf{X}_{t}^{\mathcal{N}(i)}\right)p\left(\mathbf{X}_{t}^{\mathcal{N}(i)}|\mathbf{x}_{t}^{i}\right)d\mathbf{X}_{t}^{\mathcal{N}(i)}$$
$$\times\int p\left(\mathbf{Z}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}\right)p\left(\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})}|\mathbf{X}_{t}^{\mathcal{I}}\right)d\mathbf{X}_{t}^{\mathcal{R}(\mathcal{I})} \quad (18)$$
$$=\underbrace{\prod_{j\in\mathcal{N}(i)}\int p\left(\mathbf{z}_{t}^{j}|\mathbf{x}_{t}^{j}\right)p\left(\mathbf{x}_{t}^{j}|\mathbf{x}_{t}^{i}\right)d\mathbf{x}_{t}^{j}}_{\text{Low-level inter-part weighting}}$$
$$\times\underbrace{\prod_{\mathcal{K}\in\mathcal{R}(\mathcal{I})}\int p\left(\mathbf{Z}_{t}^{\mathcal{K}}|\mathbf{X}_{t}^{\mathcal{K}}\right)p\left(\mathbf{X}_{t}^{\mathcal{K}}|\mathbf{X}_{t}^{\mathcal{I}}\right)d\mathbf{X}_{t}^{\mathcal{K}}}_{\text{High-level inter-unit weighting}}. \quad (19)$$

In (16), we use *Markov properties* (13e) and (13f). In (17), the double integral is separated into two single integrals for the different variables. In (18), we use **Assumption 1**. In (19), we apply the properties (13g) and (13j). As we can see in (18), the joint interaction likelihood of $\mathbf{x}_t^i$ is finally decomposed into two different factors: the *low-level interpart weighting* and the *high-level interunit weighting*.

Compared with DAOT, HAOT extends DAOT by introducing an additional high-level interunit weighting factor as shown in (19). Without considering the high-level interunit interaction, HAOT degrades to DAOT. This can be easily achieved by assuming the interunit weighting densities to be uniformly distributed.

C. Sequential Monte Carlo Implementation

We can also use the sequential Monte Carlo method [9] to approximate the conditional density propagation rule derived in the previous section.

In the context of HAOT, if the importance density $q(\cdot)$ is chosen to be

$$q\left(\mathbf{x}_{0:t}^{i}|\mathbf{z}_{1:t}^{i},\mathbf{Z}_{1:t}^{\mathcal{N}(i)},\mathbf{Z}_{1:t}^{\mathcal{R}(I)}\right)$$
$$=q\left(\mathbf{x}_{t}^{i}|\mathbf{x}_{0:t-1}^{i},\mathbf{z}_{1:t}^{i},\mathbf{Z}_{1:t}^{\mathcal{N}(i)},\mathbf{Z}_{1:t}^{\mathcal{R}(I)}\right)$$
$$\times q\left(\mathbf{x}_{0:t-1}^{i}|\mathbf{z}_{1:t}^{i},\mathbf{Z}_{1:t-1}^{\mathcal{N}(i)},\mathbf{Z}_{1:t-1}^{\mathcal{R}(I)}\right) \quad (20)$$

then, according to the *importance sampling theory* [9], the sample weights can be updated by

$$w_t^{i,n}\propto\frac{p\left(\mathbf{x}_{0:t}^{i}|\mathbf{z}_{1:t}^{i},\mathbf{Z}_{1:t}^{\mathcal{N}(i)},\mathbf{Z}_{1:t}^{\mathcal{R}(I)}\right)}{q(\cdot)}. \quad (21)$$

By substituting (15), (19), and (20) into (21) and approximating the integrals by summations, we have

$$w_t^{i,n}\propto w_{t-1}^{i,n}\frac{p\left(\mathbf{z}_{t}^{i}|\mathbf{x}_{t}^{i,n}\right)p\left(\mathbf{x}_{t}^{i,n}|\mathbf{x}_{t-1}^{i,n}\right)}{q\left(\mathbf{x}_{t}^{i}|\mathbf{x}_{0:t-1}^{i},\mathbf{z}_{1:t}^{i},\mathbf{Z}_{1:t}^{\mathcal{N}(i)},\mathbf{Z}_{1:t}^{\mathcal{R}(I)}\right)}$$
$$\times\prod_{j\in\mathcal{N}(i)}\left\{\sum_{l=1}^{N_s^j}p\left(\mathbf{z}_{t}^{j}|\mathbf{x}_{t}^{j,l}\right)p\left(\mathbf{x}_{t}^{j,l}|\mathbf{x}_{t}^{i,n}\right)\right\}$$
$$\times\prod_{\mathcal{K}\in\mathcal{R}(\mathcal{I})}\left\{\sum_{\mathcal{L}=1}^{N_s^{\mathcal{K}}}p\left(\mathbf{Z}_{t}^{\mathcal{K}}|\mathbf{X}_{t}^{\mathcal{K},\mathcal{L}}\right)p\left(\mathbf{X}_{t}^{\mathcal{K},\mathcal{L}}|\mathbf{X}_{t}^{\mathcal{I},n}\right)\right\} \quad (22)$$

where n is the sample index of part i (unit $\mathcal{I}$), l is the sample index of part $j,\mathcal{L}$ is the sample index of unit $\mathcal{K}$, N_s^j is the total sample number of part j, and $N_s^{\mathcal{K}}$ is the total sample number of unit $\mathcal{K}$. By using *Markov property* (13k), the density $p(\mathbf{Z}_t^{\mathcal{K}}|\mathbf{X}_t^{\mathcal{K},\mathcal{L}})$ in (22) can be further approximated by a product of all parts' local observation likelihoods in unit $\mathcal{K}$

$$p\left(\mathbf{Z}_{t}^{\mathcal{K}}|\mathbf{X}_{t}^{\mathcal{K},\mathcal{L}}\right)=\prod_{k\in\mathcal{K}}p\left(\mathbf{z}_{t}^{k}|\mathbf{x}_{t}^{k,\mathcal{L}}\right). \quad (23)$$

Noticing that all the parts of an articulated object are tracked simultaneously in our framework. By first calculating all parts' local observation likelihood, we do not have to calculate the interunit observation likelihood $p(\mathbf{Z}_t^{\mathcal{K}}|\mathbf{X}_t^{\mathcal{K},\mathcal{L}})$ separately, but can use the available local likelihoods to estimate the interunit weighting directly as shown in (23). This saves a lot of computation and makes the implementation very fast.

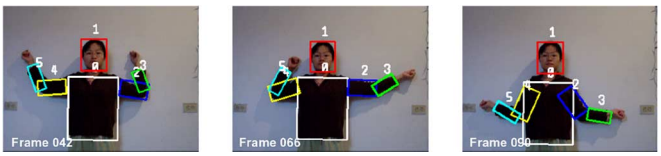

Fig. 6. Tracking results of DAOT on the sequence GIRL.

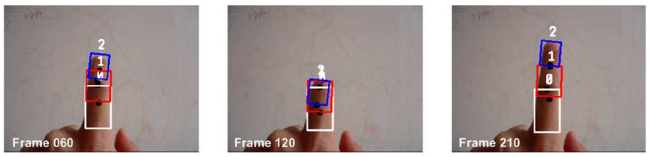

Fig. 7. Tracking results of DAOT on the sequence 3D_FINGER.

D. High-Level Interaction Model

The modeling complexity of high-level interunit interaction depends on different factors such as the dimensionality of units, the video scene, the object motion regulation, etc. Without using any prior knowledge, the interunit interaction model could be learned from video data by using *density estimation* techniques [21]. As a paradigm, we used a Gaussian mixture model [22] in our experiments to estimate the density $p(\mathbf{X}_t^{\mathcal{K}}|\mathbf{X}_t^{\mathcal{I}})$ from training data for a walking person. The distribution is learned using 3715 ground truth frames of a walking person collected at 25 frames per second. A standard iterative expectation–maximization (EM) algorithm [22] with K-means initialization is used to learn Gaussian mixture model.

IV. Experimental Results

In this section, we report some of the experimental results. The tracking performance of the proposed two methods were compared both qualitatively and quantitatively with the multiple independent trackers (MIT) [10], joint particle filter (JPF) [23], mean field Monte Carlo (MFMC) [17], and loose-limbed people tracking (LLPT), respectively. All experiments were performed using C++ on a 3.2-GHz Pentium IV PC without code optimization.

A. Qualitative Tracking Results

The video GIRL contains a girl moving her arms. It has 122 frames and was captured by 25 fps with a resolution of 320 × 240 pixels. The uniform color of the sweater and fast motion make it challenging for articulated object tracking. The proposed DAOT achieved robust tracking results even when the arms moved rapidly as shown in Fig. 6.

The video 3D_FINGER has a finger bending into the image plane as illustrated in Fig. 7. It was captured by 15 fps with a resolution of 240 × 180 pixels and has 345 frames. We use this video to demonstrate the ability of DAOT to handle self-occlusions and show its potential for 3-D applications. The black points indicate the characteristic joint points used for interpart interaction.

The WALKING sequence contains a person walking forward inside a classroom. It has 66 frames and was captured by 25 fps with a resolution of 320 × 240 pixels. The identical color of the torso and arms, and the frequent severe self-occlusions among limbs make it difficult for articulated object tracking. To compare DAOT and HAOT against the state of the art, we independently implemented the MFMC and LLPT. In Fig. 8, we illustrate the sample result frames. MFMC kept the connection

Fig. 8. Comparison of MFMC [17], LLPT [16], the proposed DAOT and HAOT for the sequence WALKING.

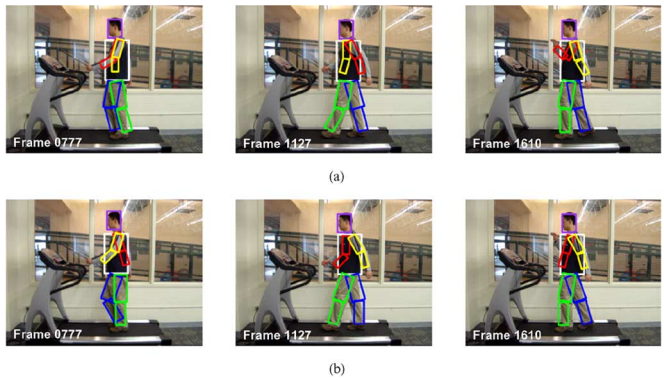

Fig. 9. Comparison of DAOT and HAOT on the sequence GYM.

TABLE II
Computational Cost and Speed Comparisons of a Centralized Approach JPF and the Proposed Decentralized Approach DAOT

Tracked Parts	JPF		DAOT	
	Samples	Speed (fps)	Samples	Speed (fps)
3	300	$13 \sim 14$	$50 \times 3 = 150$	$25 \sim 26$
4	800	$7 \sim 9$	$50 \times 4 = 200$	$20 \sim 22$
5	1500	$1.8 \sim 3$	$50 \times 5 = 250$	$16 \sim 17$
6	2500	$0.3 \sim 0.4$	$50 \times 6 = 300$	$14 \sim 15$

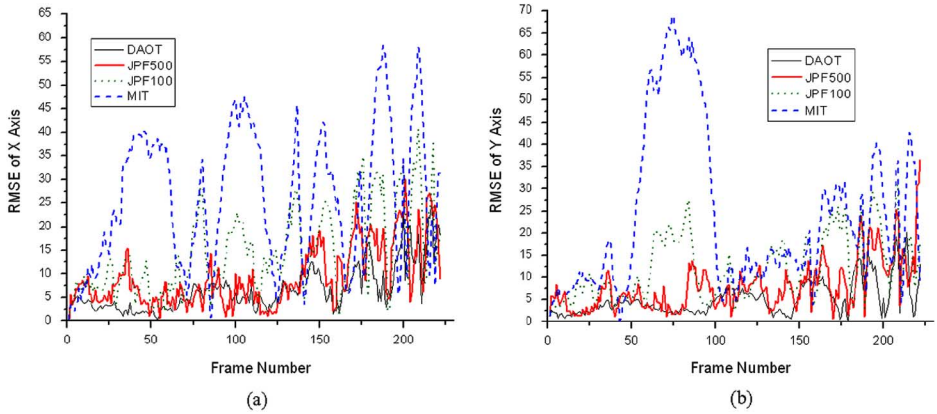

Fig. 10. Comparison of RMSE using MIT [10], JPF [23] with 100 samples and 500 samples, respectively, and the proposed DAOT on the synthetic sequence. (a) RMSE of the x axis. (b) RMSE of the y axis. For both MIT and DAOT, we use 100 samples.

among parts. However, it could not produce satisfactory results when self-occlusion presented. For simplicity and better comparison we used a 2-D instead of 3-D learned model to handle the interaction among limb parts in the LLPT, which presented improved results. However, since the location and rotation relations were considered jointly and learned implicitly together in a single model, with a fixed number of samples, the tracking results of LLPT were still not accurate. Compared with LLPT, DAOT improved the performance in that the connections among parts were preserved well. This is because DAOT uses separated interaction models for the location and rotation. Similar to LLPT, DAOT only exploits the physical adjacent constraints of the human body. This lead it to still suffer from the self-occlusion problem. By handling the high-level interaction among arms and legs and using a learned model of limb poses, HAOT gave the best results.

The video sequence GYM was captured in a gym from a side-view of a person on a walking machine. Compared with the WALKING sequence, this video is much longer (1716 frames) and has a very cluttered background. The person's movement were not purely cyclic since he occasionally raised his right arm. As we can see in Fig. 9(a), DAOT tracked the object parts well in most frames for both cyclic and irregular motions (such as the right arm in Frame 1610). However, since it only models the interpart interaction and lacks an effective scheme to handle the high-level interaction among limbs, it could not solve the self-occlusion among limbs robustly. HAOT successfully solved the self-occlusion problem by handling both the high-level interaction among arms and legs and the low-level physical constraints between parts in each limb as shown in Fig. 9(b). Since the interaction model did not include the information for irregular motion, HAOT lost such moving parts as shown in Frame 1610.

B. Quantitative Performance Analysis and Comparisons

In this section, we present a quantitative speed and accuracy analysis of DAOT and HAOT compared with other approaches.

As discussed in the introduction, the computational complexity of many existing particle filter-based articulated object tracking approaches increases exponentially in terms of the number of tracked object parts since they rely on a joint state representation. The proposed DAOT and HAOT adopt a decentralized framework and, thus, share the advantage that the computational cost increases linearly with the number of tracked parts of the articulated object. In Table II, we compare the computational cost and speed of a centralized approach JPF and DAOT. The data is obtained by applying these two methods with various numbers of tracked parts in the GIRL sequence. As we can see, to achieve a reasonable robust performance, JPF requires an exponentially increasing computational cost (in terms of samples) when the number of tracked parts increases, while our DAOT needed much less samples. Compared with JPF whose speed decreased rapidly, the speed of the DAOT decreased linearly with the number of tracked parts.

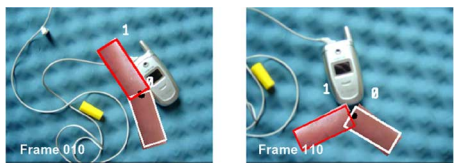

Fig. 11. Tracking results of DAOT on the synthetic sequence.

Quantitative evaluation of the performance of articulated object tracking is an unresolved matter. We have used different criteria to evaluate the performance of the proposed approach.

1) Synthetic Data: Two parts are joined together and rotate with different speeds in the image plane simulating the movement of a finger. Using the ground truth data, we can calculate the root-mean-square-error (RMSE) of the tracked parts' centers. In Fig. 10, we compare the RMSE of MIT, JPF and DAOT on the synthetic video. Fig. 11 also illustrates the visual tracking results of DAOT. As we can see, without modeling the interaction, the RMSE of MIT is much larger than JPF and DAOT. Even though JPF could achieve comparable RMSE with DAOT, it needs many more samples (500 samples). Thus, the computational cost is much higher and the speed is significantly lower. On the other hand, with the same number of samples, DAOT achieved robust tracking results with much smaller RMSE.

2) Real-World Data: Due to absence of ground truth data, it is difficult to compute the RMSE for real-world videos. Instead, we compare the tracking accuracy of different approaches by

TABLE III
SPEED AND ACCURACY COMPARISONS OF DIFFERENT PARTICLE FILTER-BASED ARTICULATED OBJECT TRACKING APPROACHES

Method	Samples per Tracker	Speed (fps)	FPR	FLR
MIT	200	$1.3 \sim 2$	44.7%	51.2%
MFMC	100	$2.1 \sim 2.6$	15.9%	18.7%
LLPT	150	$0.8 \sim 1.7$	8.3%	13.6%
DAOT	50	$9 \sim 10.2$	3.1%	5.6%
HAOT	45	$7.4 \sim 8.3$	1.5%	0.7%

defining the *false position rate* (FPR) and *false label rate* (FLR); i.e.,

$$\text{FPR} = \frac{\text{The number of position failures}}{\text{The total number of articualted object parts}} \tag{24}$$

$$\text{FLR} = \frac{\text{The number of label failures}}{\text{The total number of articualted object parts}} \tag{25}$$

where a *position failure* is defined as the absence of a tracker associated with one of the tracked parts and a *label failure* is defined as a tracker associated with a false object part.

In Table III, we compare both the speed and accuracy data of different particle filter-based approaches on the WALKING sequence. The reported optimal number of samples of each method was selected by trial-and-error. We chose the smallest number of samples which did not degrade the tracking performance significantly. As we can see, even with a much bigger number of samples than other approaches, MIT could not produce satisfactory results. MFMC and LLPT improved the performance in that both FPR and FLR became lower. This was achieved by paying additional computational cost to model the interaction. Compared with MFMC and LLPT, the proposed DAOT and HAOT model the interaction constraints of an articulated object more efficiently. With a much smaller number of samples, DAOT and HAOT achieved more robust tracking results. Moreover, even for the whole human body tracking which included ten parts, DAOT and HAOT obtained quasi-real-time running speed.

REFERENCES

[1] S. Ju, M. J. Black, and Y. Yacoob, "Cardboard people: A parameterized model of articulated image motion," presented at the Int. Conf. Automatic Face and Gesture Recognition, 1996.

[2] T. Drummond and R. Cipolla, "Real-time tracking of highly articulated structures in the presence of noisy measurements," presented at the Int. Conf. Computer Vision, Vancouver, BC, Canada, 2001.

[3] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "A review on vision-based full DOF hand motion estimation," presented at the IEEE Conf. Computer Vision and Pattern Recognition, 2005.

[4] X. Lan and D. Huttenlocher, "A unified spatio-temporal articulated model for tracking," presented at the IEEE Conf. Computer Vision and Pattern Recognition, 2004.

[5] K. Nickels and S. Hutchinson, "Model-based tracking of complex articulated objects," *IEEE Trans. Robot. Autom.*, vol. 17, no. 1, pp. 28–36, Jan. 2001.

[6] N. Jojic, M. Turk, and T. Huang, "Tracking selfoccluding articulated objects in dense disparity maps," presented at the Int. Conf. Computer Vision, Corfu, Greece, 1999.

[7] C. Bregler and J. Malik, "Tracking people with twists and exponential maps," presented at the IEEE Conf. Computer Vision and Pattern Recognition, Washington, DC, 1998.

[8] R. Kehl, M. Bray, and L. V. Gool, "Full body tracking from multiple views using stochastic sampling," in *IEEE Conf. Computer Vision and Pattern Recognition*, San Diego, CA, 2005.

[9] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Baysian tracking," *IEEE Trans. Signal Process.*, vol. 50, no. 2, pp. 174–188, Feb. 2002.

[10] M. Isard and A. Blake, "Condensation—Conditional density propagation for visual tracking," *Int. J. Comput. Vis.*, vol. 29, no. 1, pp. 5–28, 1998.

[11] J. Deutscher, A. Blake, and I. D. Reid, "Articulated body motion capture by annealed particle filtering," presented at the IEEE Conf. Computer Vision and Pattern Recognition, 2000.

[12] K. Choo and D. J. Fleet, "People tracking using hybrid monte carlo filtering," presented at the IEEE Conf. Computer Vision, Vancouver, BC, Canada, 2001.

[13] W.-Y. Chang, C.-S. Chen, and Y.-P. Hung, "Appearance-guided particle filtering for articulated hand tracking," presented at the IEEE Conf. Computer Vision and Pattern Recognition, 2005.

[14] E. Sudderth, A. Ihler, W. Freeman, and A. Willsky, "Nonparametric belief propagation," presented at the IEEE Conf. Computer Vision and Pattern Recognition, Madison, WI, 2003.

[15] M. Isard, "PAMPAS: Real-valued graphical models for computer vision," presented at the IEEE Conf. Computer Vision and Pattern Recognition, Madison, WI, 2003.

[16] L. Sigal, S. Bhatia, S. Roth, M. J. Black, and M. Isard, "Tracking loose-limbed people," presented at the IEEE Conf. Computer Vision and Pattern Recognition, Washington, DC, 2004.

[17] Y. Wu, G. Hua, and T. Yu, "Tracking articulated body by dynamic Markov network," presented at the IEEE Conf. Computer Vision, 2003.

[18] J. Whittaker, *Graphical Models in Applied Mathematical Multivariate Statistics*. New York: Wiley, 1990.

[19] W. Qu and D. Schonfeld, "Detection-based particle filtering for realtime multiple head tracking applications," presented at the SPIE Electronic Imaging Conf. Image and Video Communications and Processing, San Jose, CA, Jan. 2005.

[20] W. Qu and D. Schonfeld, "Real-time interactively distributed multi-object tracking using a magnetic-inertia potential model," presented at the Int. Conf. Computer Vision, Beijing, China, Oct. 2005.

[21] B. W. Silverman, *Density Estimation for Statistics and Data Analysis*. New York: Chapman & Hall, 1986.

[22] J. Deutscher, M. Isard, and J. MacCormick, "Automatic camera calibration from a single manhattan image," presented at the Eur. Conf. Computer Vision, Copenhagen, Denmark, 2002.

[23] M. Isard and J. MacCormick, "BraMBLe: A Bayesian multiple-blob tracker," in *Proc. Int. Conf. Computer Vision*, Vancouver, BC, Canada, Jul. 2001, vol. 2, pp. 34–41.

Wei Qu (S'04–M'06) received the B.S. degree in electrical and computer engineering from the Beijing Institute of Technology (BIT), Beijing, China, in 2000, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Illinois at Chicago (UIC), Chicago, in 2005 and 2006, respectively.

He was a Research Assistant with the Institute of Automation, Chinese Academy of Science, Beijing, from 2000 to 2002, and a Research Assistant at the Multimedia Communications Laboratory, UIC, from 2003 to 2006. During the summers of 2005 and 2006, he was a Research Intern with Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA, and Siemens Medical Solutions, Inc., Malvern, PA, respectively. He is currently a Senior Researcher at the Networks and System Center of Excellence, Motorola Labs, Schaumburg, IL. He has authored over 20 technical papers in various journals and conferences, and has two U.S. patents pending.

Dr. Qu received the Best Student Paper Award at the IEEE International Conference on Image Processing in 2006. He has also served regularly as a reviewer for different journals and conferences, such as the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, the IEEE TRANSACTIONS ON NEURAL NETWORKS, etc.

Dan Schonfeld (M'90–SM'05) was born in Westchester, PA, in 1964. He received the B.S. degree in electrical engineering and computer science from the University of California, Berkeley, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Johns Hopkins University, Baltimore, MD, in 1986, 1988, and 1990, respectively.

In 1990, he joined the University of Illinois at Chicago, where he is currently an Associate Professor in the Department of Electrical and Computer Engineering. He has authored over 100 technical papers in various journals and conferences. His current research interests are in signal, image, and video processing; video communications; video retrieval; video networks; image analysis and computer vision; pattern recognition; and genomic signal processing.

Dr. Schonfeld was coauthor of a paper that won the Best Student Paper Award in Visual Communication and Image Processing 2006. He was also coauthor of a paper that was a finalist in the Best Student Paper Award in Image and Video Communication and Processing 2005. He has served as an Associate Editor of the IEEE TRANSACTIONS ON IMAGE PROCESSING (Nonlinear Filtering) as well as an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING (Multidimensional Signal Processing and Multimedia Signal Processing).