Analyzing and Capturing Articulated Hand Motion in Image Sequences

Ying Wu, *Member*, *IEEE*, John Lin, *Member*, *IEEE*, and Thomas S. Huang, *Fellow*, *IEEE*

Abstract—Capturing the human hand motion from video involves the estimation of the rigid global hand pose as well as the nonrigid finger articulation. The complexity induced by the high degrees of freedom of the articulated hand challenges many visual tracking techniques. For example, the particle filtering technique is plagued by the demanding requirement of a huge number of particles and the phenomenon of particle degeneracy. This paper presents a novel approach to tracking the articulated hand in video by learning and integrating natural hand motion priors. To cope with the finger articulation, this paper proposes a powerful sequential Monte Carlo tracking algorithm based on importance sampling techniques, where the importance function is based on an initial manifold model of the articulation configuration space learned from motion-captured data. In addition, this paper presents a divide-and-conquer strategy that decouples the hand poses and finger articulations and integrates them in an iterative framework to reduce the complexity of the problem. Our experiments show that this approach is effective and efficient for tracking the articulated hand. This approach can be extended to track other articulated targets.

Index Terms—Motion, tracking, video analysis, statistical computing, probabilistic algorithms, face and gesture recognition.

1 Introduction

The use of hand gestures is a natural way for communications and it has attracted many research efforts aiming at the development of intelligent human computer interaction systems [24], [40], in which gesture commands may be captured and recognized by computers, and computers may even synthesize sign languages to interact with humans. For example, in some virtual environment applications, gesture interfaces may facilitate the use of bare hands for direct manipulation of virtual objects [17], [23].

One technology bottleneck of gesture-based interfaces lies in the difficulty of capturing and analyzing the articulated hand motion. Although glove-based devices can be employed to directly measure the finger joint angles and spatial positions of the hand by using a set of sensors (e.g., electromagnetic or fiber-optical sensors), they are intrusive, cumbersome, and expensive for natural interactions. Since the video sensors are cost-effective and noninvasive, a promising alternative to glove-based devices is to estimate the hand motion from video. Most existing vision-based motion capturing systems require reflective markers to be placed on the target to ease the motion tracking tasks; thus, they are not truly noninvasive. This motivates our research of developing markerless methods for tracking hand articulation.

Capturing hand and finger motions in video sequences is a highly challenging task due to the large number of degrees of freedom (DoF) of the hand kinematic structure. Fig. 1 shows the skeleton of a hand and the names of the joints. Except for the thumb, each finger has 4 DoF (2 for MCP, 1 for PIP and DIP). The thumb has 5 DoF. Adding the rigid global hand motion, the human hand has roughly 27 DoF. The high dimensionality of this problem makes the estimation of these motion parameters from images prohibitive and formidable. In addition, the rigid hand rotation may incur self-occlusion that causes fingers to become invisible, introducing large uncertainties to the estimation of the occluded parts.

Fortunately, the natural human motion is often highly constrained and the motions among various joints are closely correlated [18], [41]. Although the DoF of the hand is large, the intrinsic and feasible hand motion seems to be constrained within a subset in a lower-dimensional subspace (or the configuration space). Once the configuration space is characterized, it can be utilized to dramatically reduce the search space in capturing hand articulation. While some simple and closed form constraints have been found in biomechanics and applied to hand motion analysis [6], [15], [16], [38], further investigations on the representations and utilizations of complex motion constraints and the configuration space have not yet been conducted.

This paper presents a novel approach to capturing articulated hand motion by learning and integrating natural hand motion priors. The approach consists of three important components: 1) *The divide-and-conquer strategy*. Instead of estimating the global rigid motion and the articulated finger motion simultaneously, we decouple the hand poses and finger articulations and integrate their estimations in an iterative divide-and-conquer framework that greatly reduces the complexity of this problem. 2) *Capturing the nonrigid finger articulation*. We initiate the study of the hand articulation configuration space and provide a manifold model to characterize it. To utilize this model in tracking hand articulation, we propose a powerful importance sampling-based sequential Monte Carlo tracking algorithm that can tolerate the inaccuracy of this learned manifold model.

- *Y. Wu is with the Department of Electrical and Computer Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208. E-mail: yingwu@ece.northwestern.edu.*
- *J. Lin is with Proximex Corporation, 6 Results Way, Cupertino, CA 95014. E-mail: john.lin@proximex.com.*
- *T.S. Huang is with the Beckman Institute and the Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, 405 N. Mathews, Urbana, IL 61801. E-mail: huang@ifp.uiuc.edu.*

Manuscript received 7 July 2004; revised 24 Mar. 2005; accepted 4 Apr. 2005; published online 13 Oct. 2005.
Recommended for acceptance by Z. Zhang.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0339-0704.

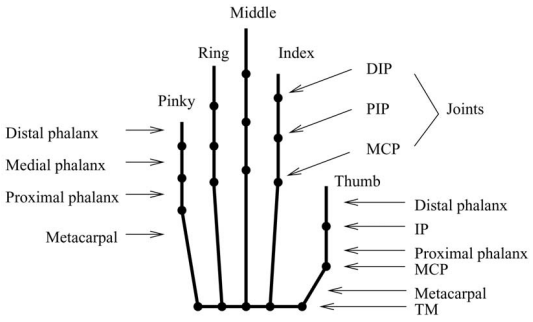

Fig. 1. Hand skeleton structure. The hand has roughly 27 DOFs.

3) *Determining the rigid hand pose*. Although many matured pose determination methods can be applied, we employ the Iterative Closed Point (ICP) algorithm and the factorization method for this purpose.

This work has three main contributions to the state-of-the-art research: 1) By learning from training data, the hand configuration space is modeled as the union of a set of linear manifolds in a lower-dimensional space ($\mathbb{R}^7$). This manifold model provides an effective prior for very efficient motion capturing. 2) Such a prior model is incorporated in the tracking process by the importance sampling scheme that redistributes the particles to more meaningful regions in order to greatly enhance valid ratio of the particles, thus leading to a very efficient computation. 3) The divide-and-conquer framework that alternates the capturing of finger articulation and the determination of the global rigid pose is practically flexible and theoretically rigorous.

In addition to the advantages of the proposed system validated in our experiments, we also discuss the limitations of our current system. It requires user-specific hand model calibration that measures the dimensions of the fingers in order to calculate the image likelihoods. Currently, this process is manually done. In addition, because of the limitation of our method for global pose estimation, our current system cannot handle large out of plane rotations and scale changes very well.

We briefly state the problem in Section 3. We describe our algorithm for capturing finger articulation in Section 4, our method for global pose determination in Section 5, and the details of the divide-and-conquer scheme in Section 6. We report our experiment results in Section 7 and conclude the paper in Section 8.

2 Related Work

Two general approaches have been explored to capture the hand articulation. The first one is the *3D model-based* approach, which takes advantage of 3D hand models and the second one is the *appearance-based approach*, which directly associates 2D image features with hand configurations.

The 3D model-based approach recovers the hand motion parameters by aligning a projected 3D model and observed image features, and minimizing the discrepancy between them. This is a challenging optimization problem in a high-dimensional space. To construct the correspondences between the model and the images, different image observations have been studied. For example, the fingertips [16], [29], [38] can be used to construct the correspondences between the model and the images. However, the robustness and accuracy largely depend on the performance of fingertip detection. The use of line features was proposed in [25], [27] to enhance the robustness. An exact hand shape model can be built by splines [15] or truncated quadrics [30] and the hand states can be recovered by minimizing the difference between the silhouettes. Since the silhouettes may not change smoothly, a Markov model can be learned in order to characterize the allowable shapes [10]. A method for combining edge and silhouette observations was reported recently for human body tracking [7].

Besides the articulated models, deformable models can also be employed to analyze hand motion. For example, one approach makes use of deformable hand shape models [9], in which the hand shape deformation can be governed by Newtonian dynamics or statistical training method such as the Principal Component Analysis (PCA). However, it is difficult to obtain accurate estimates of hand poses by these methods. An elastic graph [36] can also be used to represent hand postures. Another approach exploits a 3D deformable model in which generalized forces can be derived to integrate multiple cues including edge, optical flow, and shading information [21].

The second approach to analyzing the hand articulation is the *appearance-based* approach, which estimates hand states directly from images after learning the mapping from the image feature space to the hand configuration space. The mapping is highly nonlinear due to the variation in the hand appearances under different viewing angles. A discrete hand configuration space was proposed in [39]. Other appearance-based methods were also reported in [1], [26], [35] to recover body postures. In addition, motion capture and graphics can also be integrated in machine learning methods for human tracking [3], [4], [11]. This approach generally involves a quite difficult learning problem and it is not trivial to collect large sets of training data. The 3D model-based approach and the 2D appearance-based approach can also be combined for rapid and precise estimation [28].

3 The Problem

We denote by $\mathbf{Z}$ the feature (or image observation) and $\tilde{\mathbf{Z}}$ the hypothesized image observation given the motion $\mathbf{M} = (\boldsymbol{\Theta}, \mathbf{G})$ that consists of the local finger articulation $\boldsymbol{\Theta}$, and the global motion $\mathbf{G} = (\mathbf{R}, \mathbf{t})$, where $\mathbf{R}$ denotes the rotation and $\mathbf{t}$ the translation. The essence of capturing hand motion is to find the best motion parameters that minimize the discrepancy between $\mathbf{Z}$ and $\tilde{\mathbf{Z}}$, i.e.,

$$(\boldsymbol{\Theta}^*, \mathbf{G}^*) = \arg\min_{(\boldsymbol{\Theta}, G)} E(\mathbf{Z}, \tilde{\mathbf{Z}}(\boldsymbol{\Theta}, \mathbf{G})), \tag{1}$$

where E is the error measure. When a video sequence is given, we denote the history of the motion and the observation by $\underline{\mathbf{M}}_t = \{\mathbf{M}_1, \ldots, \mathbf{M}_t\}$ and $\underline{Z}_t = \{\mathbf{Z}_1, \ldots, \mathbf{Z}_t\}$. A Bayesian formulation of the tracking task is to recover the posterior in a recursive fashion:

$$p(\mathbf{M}_{t+1}|\underline{Z}_{t+1}) \propto p(\mathbf{Z}_{t+1}|\mathbf{M}_{t+1}) p(\mathbf{M}_{t+1}|\underline{Z}_t), \tag{2}$$

where

$$p(\mathbf{M}_{t+1}|\underline{Z}_t) = \int_{\mathbf{M}_t} p(\mathbf{M}_{t+1}|\mathbf{M}_t) p(\mathbf{M}_t|\underline{Z}_t) d\mathbf{M}_t. \tag{3}$$

The motion parameters $\mathbf{M}$ may be estimated by gradient-based nonlinear programming techniques [25] or a heuristic greedy search [15]. However, these methods rely on good starting points and are prone to local minima, due to the high dimensionality and the complexity of the search space. To enhance the robustness, particle filters [2], [12] are suggested and widely used in many tracking tasks.

Particle filters represent the posteriori $p(\mathbf{M}_t|\underline{Z}_t)$ by a set of N weighted particles $\{(s_t^{(n)}, \pi_t^{(n)})|_{n=1}^N\}$, where s denotes the sample and π denotes its weight. The recursive estimation (in (2) and (3)) is reflected by the propagation of the particle set. Specifically, the CONDENSATION algorithm [2], [12] generates particles from the dynamic prediction $p(\mathbf{M}_t|\underline{Z}_{t-1})$, and weights them by their measurements, i.e., $\pi_t^{(n)} = p(\mathbf{Z}_t|\mathbf{M}_t = s_t^{(n)})$. In this algorithm, the sampling, propagating, and reweighting process of the particles strictly follow the probabilistic derivation of the recursive estimation. It can achieve quite robust tracking results for some applications.

However, this particle filtering technique is challenged by the problem of tracking hand articulation, mainly because of:

- **High dimensionality.** This is induced by the complexity of the motion itself. Since the computational cost of particle filters comes mainly from the image measurement processes, the number of samples directly determines the accuracy and the speed of the tracker. In CONDENSATION, the number of samples needed is, in general, exponential to the dimensionality of the motion. Thus, this method is fine for rigid motion with 6 DoF, but demands formidable computations for articulated targets such as the hand with 27 DoF.
- **Particle degeneracy.** A more serious problem is caused by the sampling process. CONDENSATION uses stochastic integration to sample the prediction prior $p(\mathbf{M}_t|\underline{Z}_{t-1})$. This is correct in theory, but often leads to tracking failure, in practice, if the dynamics model $p(\mathbf{M}_t|\mathbf{M}_{t-1})$ used in tracking is not accurate. As a result, most of the samples may receive negligible weights and a large computation effort is wasted by just maintaining them. This is called *particle degeneracy*, as also noticed in the study of statistics [8], [19], [20].

In the literature, there are several approaches alleviating these challenges: For example, a semiparametric approach was taken in [5]. It retains only the modes (or peaks) of the probability density and models the local neighborhood surrounding each mode with a Gaussian distribution. Different sampling techniques were also investigated to reduce the number of samples, such as partitioned sampling scheme [22], annealed particle filtering scheme [7], tree-based filtering [31], [33], and nonparametric belief propagation [32].

Our approach is different from these methods. To address the first difficulty, our method embeds two mechanisms: a divide-and-conquer strategy and a dimension reduction procedure. Both the global rigid pose $\mathbf{G}$ and the local finger articulation $\boldsymbol{\Theta}$ contribute to the high dimensionality of the motion, but they cannot be estimated independently. In this paper, rather than solving $\mathbf{G}$ and $\boldsymbol{\Theta}$ simultaneously, we propose a more feasible and more efficient divide-and-conquer procedure that alternates the estimation of $\mathbf{G}$ and $\boldsymbol{\Theta}$ iteratively. As described later, this iterative process leads to convergence. Since the pose determination problem for rigid objects has received extensive studies, this divide-and-conquer strategy provides a framework to integrate these well-studied rigid pose determination methods with the efficient approach to articulated motion proposed in this paper.

In addition, since the motion of the finger phalanxes are correlated and constrained, the actual dimensionality of the finger articulation is less than its DoF. Thus, we apply a dimension reduction technique to find the intrinsic dimension that reduces the searching space for motion capturing.

To address the second difficulty, we learn from motion-captured data to obtain a prior of the finger articulation that leads to a more efficient tracking method based on importance sampling techniques. The learned motion prior is not necessarily accurate, but it suffices to be used as the importance function to redistribute the particles to more meaningful regions while maintaining the true underlying probability density represented by the particles. As a result, we can use a much smaller number of particles for a more efficient motion capturing.

4 Capturing Finger Articulation

This section presents our method to cope with the local finger articulation based on the importance sampling technique and a learned importance function of the hand articulation. After briefly introducing sequential Monte Carlo techniques in Section 4.1, we describe in Section 4.2 our method of characterizing the configuration space of the natural hand articulation, which is used as the importance function in the proposed sampling-based tracking algorithm in Section 4.3. The calculation of the image likelihood is described in Section 4.4.

4.1 Sequential Monte Carlo Techniques

Sampling techniques are widely used to approximate a complex probability density. A set of weighted random samples (or particles) $\{s^{(n)}, \pi^{(n)}|_{n=1}^N\}$ is *properly weighted* with respect to the distribution $f(\mathbf{X})$ if for any integrable function h of the random vector $\mathbf{X}$,

$$\lim_{N\to\infty} \frac{\sum_{k=1}^N h(s^{(k)})\pi^{(k)}}{\sum_{k=1}^N \pi^{(k)}} = E_f(h(\mathbf{X})).$$

In this sense, the distribution is approximated by a set of discrete random samples, $s^{(k)}$ with each having a probability proportional to its weight $\pi^{(k)}$.

These sampling techniques can also be used for simulating dynamic systems as long as the particle sets are properly weighted. They are called sequential Monte Carlo techniques in statistics [8], [19], [20]. The CONDENSATION algorithm [2], [12] is an example. Denote by $\mathbf{X}_t$ the motion to be inferred from estimating the posterior $p(\mathbf{X}_t|\underline{Z}_t)$. CONDENSATION draws a set of samples $\{s_t^{(n)}|_{n=1}^N\}$ from the dynamics prediction prior $p(\mathbf{X}_t|\underline{Z}_{t-1})$, and weights them by their measurements, i.e., $\pi_t^{(n)} = p(\mathbf{Z}_t|\mathbf{X}_t = s_t^{(n)})$. The particles of $p(\mathbf{X}_t|\underline{Z}_{t-1})$ are obtained through *stochastic integration* by propagating the particle set that represents the posterior at time $t-1$, i.e., $p(\mathbf{X}_{t-1}|\underline{Z}_{t-1})$. It can be shown that such a particle set is properly weighted. As described in Section 3, this method encounters two challenges when applied to tracking articulated targets: computationally demanding and particle degeneracy.

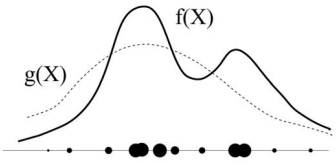

Fig. 2. Importance sampling. To represent the desired distribution $f(\mathbf{X})$, samples can be drawn from an importance function $g(\mathbf{X})$ but with compensated weights.

In fact, to represent a distribution $f(\mathbf{X})$, it is not necessary to draw samples from this distribution directly. We may generate particles from a proposal density $g(\mathbf{X})$, provided that we adjust or reweight the samples. This is the basic idea of the *importance sampling* scheme. When particles $\{s^{(n)}, \tilde{\pi}^{(n)}\}$ are generated from $g(\mathbf{X})$, their weights are compensated as

$$\pi^{(n)} = \frac{f(s^{(n)})}{g(s^{(n)})}\tilde{\pi}^{(n)},$$

where $\tilde{\pi}^{(n)}$ are the uncompensated weights associated with the sampling of $g(\mathbf{X})$. It can be proven that the sample set $\{(s^{(n)},\pi^{(n)})|_{n=1}^{N}\}$ is still *properly weighted* with respect to $f(\mathbf{X})$. This is illustrated in Fig. 2.

To employ the importance sampling technique in dynamic systems, we let $f_t(\mathbf{X}_t^{(n)}) = p(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\underline{Z}_{t-1})$, where $f_t(\cdot)$ is the tracking prediction prior (as used in CONDENSATION). We can draw samples from a proposal distribution $g_t(\mathbf{X}_t)$ (e.g., [13] used color-segmented regions for tracking the positions of hand blobs as a simple case), while compensating the weights by:

$$\pi_t^{(n)} = \frac{f_t(\mathbf{X}_t^{(n)})}{g_t(\mathbf{X}_t^{(n)})} p\Big(\mathbf{Z}_t|\mathbf{X}_t = \mathbf{X}_t^{(n)}\Big). \tag{4}$$

To evaluate $f_t(\mathbf{X}_t)$, we have:

$$\begin{aligned} f_t(\mathbf{X}_t^{(n)}) &= p\Big(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\underline{Z}_{t-1}\Big) \\ &= \sum_{k=1}^{N} \pi_{t-1}^{(k)} p\Big(\mathbf{X}_t = \mathbf{X}_t^{(n)}|\mathbf{X}_{t-1} = \mathbf{X}_{t-1}^{(k)}\Big). \end{aligned}$$

In this importance sampling scheme, no matter what importance function is used, the particle propagation always exactly follows the probability deduction of the dynamic systems. Thus, this sequential Monte Carlo method is provably correct. At the same time, it provides a powerful clue and a flexible way to overcome the challenges to CONDENSATION by constructing a proper proposal distribution (or the importance function) $g_t(\mathbf{X}_t)$ to minimize the risk of particle degeneracy and reduce the number of particles significantly. Because the importance function can be arbitrarily chosen what would be an appropriate one for tracking the articulated hand motion? We propose a method in the next section.

4.2 Learning the Importance Function for Sampling

Although the finger motion is highly articulated, its kinematics is constrained. Only certain hand configurations are feasible and natural, which form a subspace of the entire finger joint angle space. By *natural*, we mean, the configurations that should not induce much muscle tension. In general, these set of natural motion can be covered by all the combinations of extending and curling the five fingers, but exclude finger crossing. Thus, the natural motions actually include a large variety of gestures. Of course, people can make arbitrary hand configurations, but only these natural configurations need to be considered in most gesture interface applications. Fortunately, the natural hand configurations for most people are similar; therefore, having such strong articulation priors can greatly improve the motion estimation. However, these priors are very difficult to model explicitly. Finding an effective representation of the feasible hand configuration space (C-space) is not well addressed in the literature. In this section, we present an initial model of the natural hand configuration subspace including its dimensionality and topology.

Feasible hand articulation does not span the entire joint angle space $\Theta \subset \mathbb{R}^{20}$. We generally observe three types of constraints. One type of constraints, usually referred to as the static constraints in previous work, are the limits of the range of finger motions as a result of the hand anatomy, such as $0^0 \leq \theta_{MCP} \leq 90^0$. The second type of constraints describes the correlations among different joints and, thus, reduces the dimensionality of hand articulation. For example, the motions of the DIP and PIP joints are generally not independent and they can be characterized by $\theta_{DIP} = \frac{2}{3}\theta_{PIP}$ from the study of biomechanics [6]. Although this constraint can be intentionally made invalid, it has been shown to provide a good approximation to natural finger motion [15], [16]. The third class of constraints can be called *purposive constraints* since it is imposed by the naturalness of the common hand motions which are subtle to describe. Unfortunately, not all of such constraints can be quantified in closed forms. This motivates us to model the constraints using other alternatives.

Instead of using the joint angle space $\Theta \subset \mathbb{R}^{20}$, we employ the hand configuration space Ξ to represent natural hand articulations. We are particularly interested in the dimensionality of the configuration space Ξ and the behaviors of the hand articulation in Ξ. To investigate these problems, we propose a learning approach to model hand motion constraints in Ξ from a large set of hand motion data collected using a right-handed 18-sensor CyberGlove. We have collected a set of more than 30,000 joint angle measurements $\{\theta_k, k = 1, \ldots, N\}$ by performing various natural finger motions that include all combinations of extending and curling the five fingers but exclude crossing fingers. The correlations of different joints are assumed to be well represented by such a data set. Since only the finger articulation is of concern here in natural motion, the global pose data are not used in learning. PCA is applied to project the joint angle space to the configuration space by eliminating the redundancy, i.e.,

$$\mathbf{X} = \mathbf{U}^T(\theta - \theta_0), \tag{5}$$

where $\mathbf{U}$ is constructed by the eigenvectors corresponding to large eigenvalues of the covariance matrix of the data set and $\theta_0 = \frac{1}{N}\sum_{k=1}^{N}\theta_k$ is the mean of the data set. The result shows that we can project the original joint angle space into a seven-dimensional subspace, while maintain 95 percent of the variance. We plot the percentage of the variance preserved with respect to the number of eigenvalues in Fig. 3. Thus, $\mathbf{X} \in \Xi \subset \mathbb{R}^7$.

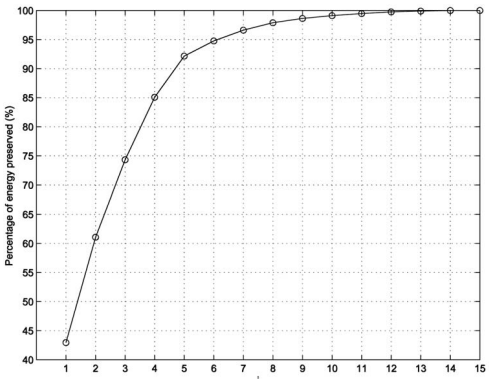

Fig. 3. The plot of the percentage of energy (i.e., variance) preserved with respect to the number of eigenvalues shows that the first 7D subspace preserves 95 percent of the variance.

Since the natural hand articulation only covers a subset of $\mathbb{R}^7$, to characterize the configuration space Ξ, we define 28 basis configurations $\mathcal{B} = \{\mathbf{b}_1, \ldots, \mathbf{b}_M : \forall \mathbf{b}_k \in \Xi, M = 28\}$. Since the feasible finger motions are bounded roughly by two extremal states, fully extended or curled, the five fingers together defines 32 states that roughly characterize the entire natural hand motion. Considering not everyone is able to bend the pinky without bending the ring finger, four unnatural states are not included in our set of basis states. Similar configurations are considered as the same state. For each basis state, we collect a set of joint angle data and project its mean to $\mathbb{R}^7$ as the basis configuration. All 28 bases are shown in Fig. 4.

Surprisingly, after examining the data in Ξ, we found that natural hand articulation lies largely in the set of linear manifolds spanned by any two basis configurations. For example, if the hand moves from a basis configuration $\mathbf{b}_i$ to another basis $\mathbf{b}_j$, the intermediate hand configuration lies approximately on the linear manifold spanned by $\mathbf{b}_i$ and $\mathbf{b}_j$, i.e.,

$$\mathbf{X} \in \mathcal{L}_{ij} = s\mathbf{b}_i + (1-s)\mathbf{b}_j, \quad 0 \le s \le 1. \tag{6}$$

Consequently, the hand articulation can be characterized in Ξ by:

$$\Xi \approx \bigcup_{i,j} \mathcal{L}_{ij}, \text{where } \mathcal{L}_{ij} = \mathtt{span}(\mathbf{b}_i, \mathbf{b}_j). \tag{7}$$

Since it is impossible for us to visualize data in high-dimensional space such as $\mathcal{R}^7$, we take a subset of the basis states and the corresponding hand motion trajectories and performed the same analysis as described earlier in order to visualize the result. A lower-dimensional visualization of the subset is shown in Fig. 5, in which each point represents a real hand configuration in Ξ.

In this example, the movements involving index, middle, and ring fingers are chosen. The corresponding basis states lie roughly at the corner of the cube whose edges are formed by the collection of the motion trajectories between the basis states. In this plot, the interior of the cube is shown to be almost empty due to staged performance. In reality, since the finger movements are largely covered by such motion trajectories among the bases, the density inside the convex hull is indeed very low. Thus, such an union of the

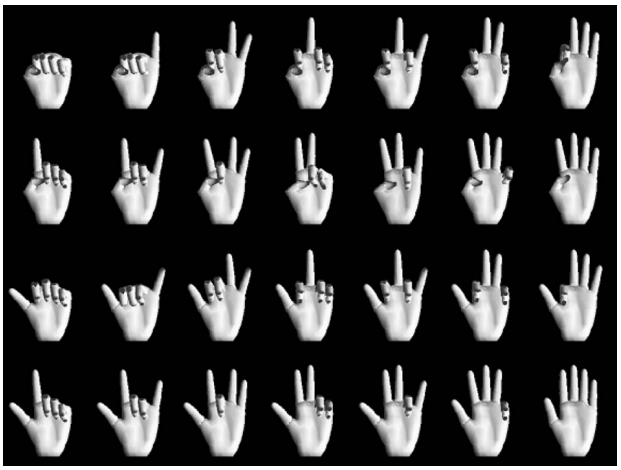

Fig. 4. The 28 basis configurations.

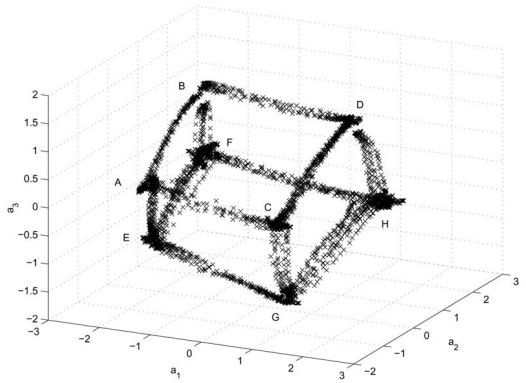

Fig. 5. A lower-dimensional visualization of a subset of the hand articulation configuration space, which is characterized by a set of basis configurations and linear manifolds. The basis states are located roughly at the corner of the cube. Each data point collected with the data glove is plotted as a "×."

set of linear manifolds actually capture the high density regions of the configuration space. As a result, it provides an effective importance function for sampling.

We noticed that [9] proposed a PCA-based approach to characterize the hand shape deformations that lie in the space spanned by a set of eigen shapes. Our method is different from theirs since our representation characterizes hand articulation in more details. Besides describing a subspace, our representation actually describes the structure of the articulation subset in the configuration space by an union of linear manifolds. Also, our representation of hand articulation is view-independent, since it is derived from the joint angle space.

4.3 Importance Sampling for Hand Articulation

One important part of sequential Monte Carlo tracking is to generate samples $\{(\mathbf{X}_{t+1}^{(n)}, \pi_{t+1}^{(n)})|_{n=1}^{N}\}$ at time $t+1$ from the samples $\{(\mathbf{X}_{t}^{(n)}, \pi_{t}^{(n)})|_{n=1}^{N}\}$ at time t. Instead of directly sampling from the prior $p(\mathbf{X}_{t+1}|\mathbf{Z}_t)$, we propose an importance sampling technique by taking the hand articulation manifolds (in Section 4.2) as the importance function.

Each hand configuration $\mathbf{X}$ should be either around a basis state $\mathbf{b}_i, i = 1, \ldots, M$, or on a manifold $\mathcal{L}_{ij}$, where $i \neq j, i, j = 1, \ldots, M$. Suppose at time frame t, the hand

configuration is $\mathbf{X}_t$. We find the projection $\bar{\mathbf{X}}_t$ of $\mathbf{X}_t$ onto the nearest manifold $\mathcal{L}_{ij}^*$, i.e.,

$$\begin{aligned}\mathcal{L}_{ij}^* &= \arg\min_{\mathcal{L}_{ij}} D(\mathbf{X}_t, \mathcal{L}_{ij}) \\ \bar{\mathbf{X}}_t &= \texttt{Proj}(\mathbf{X}_t, \mathcal{L}_{ij}^*) \\ &= \mathbf{b}_i + \frac{(\mathbf{X}_t - \mathbf{b}_i)^T(\mathbf{b}_j - \mathbf{b}_i)}{||(\mathbf{b}_j - \mathbf{b}_i)||}(\mathbf{b}_j - \mathbf{b}_i).\end{aligned}$$

Accordingly,

$$s_t = 1 - \frac{(\mathbf{X}_t - \mathbf{b}_i)^T(\mathbf{b}_j - \mathbf{b}_i)}{||(\mathbf{b}_j - \mathbf{b}_i)||}.$$

Random samples are drawn from the manifold $\mathcal{L}_{ij}$ according to the density p_{ij}, i.e.,

$$s_{t+1}^{(n)} \sim p_{ij} = N(s_t, \sigma_0), \tag{8}$$

$$\bar{\mathbf{X}}_{t+1}^{(n)} = s_{t+1}^{(n)}\mathbf{b}_i + \left(1 - s_{t+1}^{(n)}\right)\mathbf{b}_j, \tag{9}$$

where σ_0 controls the changes of the gestures within two consecutive frames. In our experiments, we set $\sigma_0 = 0.2$. Noticing $0 \le s \le 1$, we forcefully project $s_{t+1}^{(n)}$ to $[0, 1]$ by $\min(1, \max(0, s_{t+1}^{(n)}))$. Then, perform random walk on $\bar{\mathbf{X}}_{t+1}^{(n)}$ to obtain hypothesis $\mathbf{X}_{t+1}^{(n)}$, i.e.,

$$\mathbf{X}_{t+1}^{(n)} \sim N\left(\bar{\mathbf{X}}_{t+1}^{(n)}, \Sigma_1\right), \tag{10}$$

where Σ_1 reflects the uncertainty of the linear manifolds, thus controls the diffusion (or the deviation) of the particles from the manifolds. We let $\Sigma_1 = \sigma_1^2\mathbf{I}$ and set $\sigma_1 = 0.5$ in our experiments. This process is illustrated in Fig. 6a. Although, in principle, this covariance can be estimated from training data, we found in our experiments that our treatment performs better since the training data from the data glove were very noisy and the outliers affect the estimation accuracy. Based on this sampling process, the importance function can be written as:

$$\begin{aligned} g_{t+1}\left(\mathbf{X}_{t+1}^{(n)}\right) &= p\left(s_{t+1}^{(n)}|s_t\right)p\left(\mathbf{X}_{t+1}^{(n)}|\bar{\mathbf{X}}_{t+1}^{(n)}\right) \\ &\propto \exp\left\{-\frac{(s_{t+1}^{(n)} - s_t)^2}{2\sigma_0^2} - \frac{||(\mathbf{X}_{t+1}^{(n)} - \bar{\mathbf{X}}_{t+1}^{(n)})||^2}{2\sigma_1^2}\right\}. \end{aligned} \tag{11}$$

If the previous hand configuration is close to one of the basis configurations, say $\mathbf{X}_t = \mathbf{b}_k$, then it is reasonable to assume that it takes any one of the manifolds of $\{\mathcal{L}_{kj}, j = 1, \ldots, M\}$ with an equal probability, as shown in Fig. 6b. Once a manifold is selected, the same steps shown in (8)-(10) are performed.

Suppose at time t, the tracking posteriori $p(\mathbf{X}_t|\underline{Z}_t)$ is approximated by a set of weighted random samples or hypotheses $\{(\mathbf{X}_t^{(n)}, \pi_t^{(n)})|_{n=1}^N\}$. For a dynamic system, the prior is $p(\mathbf{X}_{t+1}|\underline{Z}_t)$, and we have

$$\begin{aligned} f_{t+1}\left(\mathbf{X}_{t+1}^{(n)}\right) &= p\left(\mathbf{X}_{t+1} = \mathbf{X}_{t+1}^{(n)}|\underline{Z}_t\right) \\ &= \sum_{k=1}^{N} \pi_t^{(k)} p\left(\mathbf{X}_{t+1} = \mathbf{X}_{t+1}^{(n)}|\mathbf{X}_t = \mathbf{X}_t^{(k)}\right). \end{aligned}$$

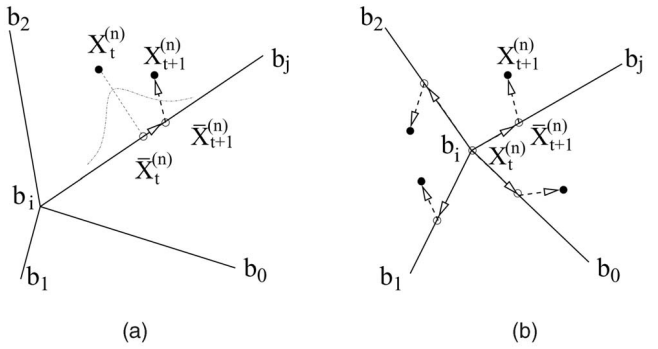

Fig. 6. Generating particles: (a) When $\mathbf{X}_t^{(n)} \neq \mathbf{b}_i$, the nearest manifold $\mathcal{L}_{ij}$ is chosen. The particle is generated by projecting to the manifold, random walking along the manifold, and diffusing away from the manifold. (b) When $\mathbf{X}_t^{(n)}$ is close to $\mathbf{b}_i$, randomly take a manifold and generate particle as (a).

Let the dynamics model be

$$p\left(\mathbf{X}_{t+1}^{(n)}|\mathbf{X}_t^{(k)}\right) = N\left(\mathbf{C}\mathbf{X}_t^{(k)}, \Sigma_2\right),$$

where $\mathbf{C}$ is the state transition matrix of the dynamic system and Σ_2 is the uncertainty of the dynamics. For simplicity, here we adopt a random walk model and set $\mathbf{C}$ to an identity matrix. Higher order models such as the constant acceleration model can also be used. In our experiments, we let $\Sigma_2 = \sigma_2^2\mathbf{I}$ and set $\sigma_2 = 0.5$. Instead of sampling directly from the prior $p(\mathbf{X}_{t+1}|\underline{Z}_t)$, samples are drawn from the proposal distribution $g_t(\mathbf{X}_{t+1})$ in (11) and the weight of each sample is compensated by:

$$\pi_{t+1}^{(n)} = \frac{f_{t+1}(\mathbf{X}_{t+1}^{(n)})}{g_{t+1}(\mathbf{X}_{t+1}^{(n)})} p\left(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1} = \mathbf{X}_{t+1}^{(n)}\right). \tag{12}$$

4.4 Model Matching: $p(\mathbf{Z}_t|\mathbf{X}_t)$

The likelihood of the image observation $p(\mathbf{Z}_t|\mathbf{X}_t)$ plays an important role in reweighting the particles (4). To calculate the likelihood, we use a *cardboard* model [14], in which each finger is represented by a set of three connected planar patches. The length and width of each patch should be calibrated according to each individual person. The kinematical chain of one finger is shown in Fig. 7a and the cardboard model in Fig. 7b. Although it is a simplification of the real hand, it offers a good approximation for motion capturing.

We measure the likelihood based on both edge and silhouette observations. Since the hand is represented by a cardboard model, it is expected to observe two edges for each planar patch. In our algorithm, a particle encodes a specific configuration of the fingers, thus determining the set of joint angles for this configuration. The global pose and the configuration of the hand determine the 3D depth of all the planar patches of the cardboard model and their occlusion relationship, based on which we compute the edges and silhouette of the model projection. As illustrated in Fig. 8, the cardboard model is sampled at a set of K points on the laterals of the patches. For each such sample, edge detection is performed on the points along the normal of this sample. When we assume that m edge points $\{z_i, 1 \le i \le m\}$ are

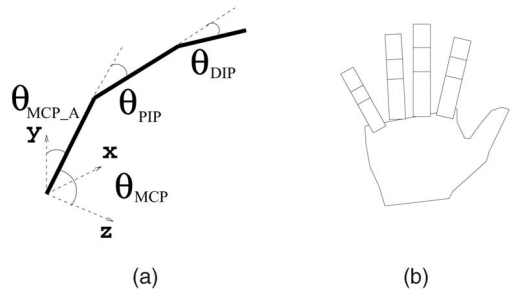

Fig. 7. (a) Kinematical chain of one finger. (b) Cardboard hand model.

observed and the clutter is a Poisson process with density λ [2], [37], then the edge likelihood is:

$$p_k^e(\mathbf{z}|x_k) \propto 1 + \frac{1}{\sqrt{2\pi}\sigma_e q\lambda}\sum_{i=1}^{m}\exp -\frac{(z_i - x_k)^2}{2\sigma_e^2}.$$

We noticed that edge points alone may not provide a good likelihood estimation, because the nearby fingers generate clutters. Therefore, we also consider the silhouette measurement. The color segmented foreground region A_I are XORed with the projected silhouette image A_M and the likelihood is computed as $p^s \propto \exp -\frac{(A_I - A_M)^2}{2\sigma_s^2}$. Thus, the total likelihood can be written as:

$$p(\mathbf{Z}|\mathbf{X}) \propto p^s \prod_{k=1}^{K} p_k^e. \tag{13}$$

4.5 Algorithm Summary

The algorithm for tracking the local finger articulation is summarized in Fig. 9.

Monte Carlo Tracking: Generate $\{(\mathbf{X}_{t+1}^{(n)}, \pi_{t+1}^{(n)}|_{n=1}^{N}\}$ from $\{(\mathbf{X}_t^{(n)}, \pi_t^{(n)}|_{n=1}^{N}\}$ based on importance sampling.

for $n = 1 : N$
  // Step(1): Selecting a manifold
  if $\mathbf{X}_t^{(n)} \neq \mathbf{b}_i, i = 1, \ldots, M$
    $\mathcal{L}_{ij}^* = \arg\min_{\mathcal{L}_{i,j}} D(\mathbf{X}_t^{(n)}, \mathcal{L}_{ij})$;
    $s_t^{(n)} = 1 - \frac{(\mathbf{X}_t^{(n)} - \mathbf{b}_i)^T(\mathbf{b}_j - \mathbf{b}_i)}{\|\mathbf{b}_j - \mathbf{b}_i\|}$;
  else
    randomly_pick $\mathcal{L}_{ij}^*$;
    $s_t^{(n)} = 0$;

  // Step(2): Sampling from $g_t(\cdot)$
  $s_{t+1}^{(n)} \sim N(s_t^{(n)}, \sigma_0)$;
  $\bar{\mathbf{X}}_{t+1}^{(n)} = s_{t+1}^{(n)}\mathbf{b}_i + (1 - s_{t+1}^{(n)})\mathbf{b}_j$;

  // Step(3): Drifting and diffusing
  $\mathbf{X}_{t+1}^{(n)} \sim N(\bar{\mathbf{X}}_{t+1}^{(n)}, \Sigma_1)$;

  // Step(4): Observing
  $\tilde{\pi}_{t+1}^{(n)} = p(\mathbf{Z}_{t+1}|\mathbf{X}_{t+1} = \mathbf{X}_{t+1}^{(n)})$;

  // step(5): Correcting the weights
  calculate $f(\mathbf{X}_{t+1}^{(n)})$;
  calculate $g(\mathbf{X}_{t+1}^{(n)})$;
  $\pi_{t+1}^{(n)} = \frac{f(\mathbf{X}_{t+1}^{(n)})}{g(\mathbf{X}_{t+1}^{(n)})}\tilde{\pi}_{t+1}^{(n)}$;
end
normalize $\{\pi_{t+1}^{(n)}|_{n=1}^{N}\}$;

Fig. 9. Pseudocode of the sequential Monte Carlo-based tracking algorithm.

5 Estimating the Global Poses

We define the global rigid hand motion by the pose of the palm. In this paper, we treat the palm as a rigid planar object. The pose determination is formulated under scaled orthographic projection in Section 5.1 and the global motion is computed via the Iterative Closed Point (ICP) approach in Section 5.2.

5.1 Hand Pose Determination

In this section, we assume the correspondences have been constructed for pose determination. The process of building the correspondences will be presented in Section 5.2. Let a point on the plane be $\mathbf{x}_i = [x_i, y_i]^T$, and its image point be $\mathbf{m}_i = [u_i, v_i]^T$. Under the scaled orthographic projection, we have

$$s\begin{bmatrix} u_i \\ v_i \\ 1 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & t_1 \\ R_{21} & R_{22} & R_{23} & t_2 \\ 0 & 0 & 0 & t_3 \end{bmatrix}\begin{bmatrix} x_i \\ y_i \\ 0 \\ 1 \end{bmatrix}.$$

That is:

$$t_3\begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}\begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} = \mathbf{A}\mathbf{x}_i + \mathbf{t},$$

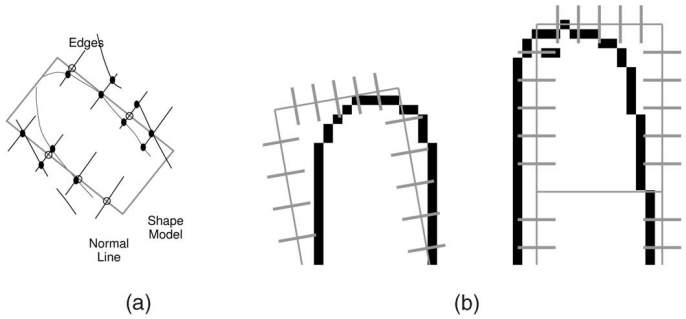

Fig. 8. Shape measurements. A hypothesized cardboard model is projected and the edge measurements are collected along the laterals of the patches.

where

$$\mathbf{A} = \begin{bmatrix} R_{11} & R_{12} \\ R_{21} & R_{22} \end{bmatrix}, \quad \text{and} \quad \mathbf{t} = \begin{bmatrix} t_1 \\ t_2 \end{bmatrix}.$$

By subtracting the centers of the projection points and model points, i.e., $\hat{\mathbf{m}}_i = \mathbf{m}_i - \bar{\mathbf{m}}$ and $\hat{\mathbf{x}}_i = \mathbf{x}_i - \bar{\mathbf{x}}$, and letting $\mathbf{B} = \mathbf{A}/t_3$, we can write:

$$\hat{\mathbf{m}}_i = \mathbf{B}\hat{\mathbf{x}}_i.$$

This is an affine transform. We denote by $[\hat{u}_i^k, \hat{v}_i^k]^T$ the ith image point (centroid subtracted) at the kth frame. If we have K corresponding frames, we can write:

$$\mathbf{W} = \begin{bmatrix} \hat{u}_1^1 & \hat{u}_2^1 & \dots & \hat{u}_N^1 \\ \hat{v}_1^1 & \hat{v}_2^1 & \dots & \hat{v}_N^1 \\ \vdots & \vdots & \vdots & \vdots \\ \hat{u}_1^K & \hat{u}_2^K & \dots & \hat{u}_N^K \\ \hat{v}_1^K & \hat{v}_2^K & \dots & \hat{v}_N^K \end{bmatrix} = \mathbf{MS}, \tag{14}$$

where

$$\mathbf{M} = \begin{bmatrix} \mathbf{B}^1 \\ \vdots \\ \mathbf{B}^K \end{bmatrix} \quad \text{and} \quad \mathbf{S} = \begin{bmatrix} \hat{x}_1 & \hat{x}_2 & \dots & \hat{x}_N \\ \hat{y}_1 & \hat{y}_2 & \dots & \hat{y}_N \end{bmatrix}.$$

Once the 3D model is calibrated, i.e., $\mathbf{S}$ is given, calculating the motion $\mathbf{M}$ is straightforward (i.e., $\mathbf{M}=\mathbf{W}\mathbf{S}^\dagger = \mathbf{W}\mathbf{S}^T(\mathbf{S}\mathbf{S}^T)^{-1}$, where $\mathbf{S}^\dagger$ is the pseudoinverse of $\mathbf{S}$). If it is not calibrated, the factorization method [34] can be taken to solve $\mathbf{M}$ and recover $\mathbf{S}$. Once $\mathbf{M}$ is solved, it is easy to figure out the pose $\mathbf{R}$ and $\mathbf{t}$. For simplicity, we can use the first frame that shows the front palm for calibration, and take the image points along the palm contour as the model points.

5.2 Iterative Closed Points

The pose determination method presented in the previous section assumes point correspondences. In this section, we describe a method for establishing point correspondences by adapting the idea of the Iterative Closed Point (ICP) algorithm. A comprehensive description of ICP for free-form curve registration can be found in [42]. The basic idea is to refine the correspondences and the motion parameters iteratively.

Since we treat the palm as a rigid planar object, it can be represented by its contour curve, which in turn can be described by a set of chained points. Let $\mathbf{x}_j (1 \le j \le N)$ be the N chained points on the 3D curve model $\mathcal{C}$ and $\mathcal{C}'$ be the edge points observed in the image. The objective is to construct the correspondences between the two curves, such that

$$e(\mathbf{R}, \mathbf{t}) = \sum_{j=1}^{N} \mathtt{D}\Big(\mathbf{P}\Big(\mathbf{R}\mathbf{x}_j^t + \mathbf{t}\Big), \mathcal{C}'\Big) w_j \tag{15}$$

is minimized, where $\mathtt{D}(\mathbf{x}, \mathcal{C}')$ denotes the distance of the point $\mathbf{x}$ and the curve $\mathcal{C}'$, w_j takes value 1 if there is a match for $\mathbf{x}_j$ and 0 otherwise, and $\mathbf{P}$ is the projection matrix given by camera calibration.

The ICP algorithm takes the image edge point that is closest to the projected 3D model point i.e., $\mathbf{P}(\mathbf{R}\mathbf{x}_k^t + \mathbf{t})$, as its correspondence. When all image edge points are far enough from the projection, the model point $\mathbf{x}_k$ is considered to have no matching point and w_k is set to 0. Motion $(\mathbf{R}, \mathbf{t})$ is computed from such a temporary correspondence using the pose determination method presented in Section 5.1. The computed motion will result in a new matching. By iteratively applying this procedure, ICP continues to refine the pose estimation. It should be pointed out that the ICP procedure converges only to local minima, which means that we need a fairly close initial start. Obviously, the ICP algorithm can be easily extended to two-frame registration.

It is worth mentioning that there is a limitation of this method for determining the global pose. Our method treats the pose of the palm as the pose of the hand (without using the fingers) and use the edges of the palm as features. Although it simplifies the pose estimation by assuming the palm to be a rigid planar object, it induces errors in practice. One reason is that the palm also undergoes substantial nonrigid motion in certain gestures. In addition, the image edges are not true edges of the palm but the projection edges when the palm is not frontal. As a result, the correspondences will not be accurate when the palm presents large out-of-plane rotation and scaling and when the palm is partially occluded. Although there have been many pose determination methods for rigid objects, accurate pose estimation of nonrigid objects such as the hand remains a quite difficult problem.

6 Divide and Conquer

The divide-and-conquer method alternates two operations:

$$\mathbf{G} = \mathcal{R}(\boldsymbol{\Theta}) = \arg\min_{G} E(\mathbf{Z}, \tilde{\mathbf{Z}}(\boldsymbol{\Theta}, \mathbf{G})),$$

and

$$\boldsymbol{\Theta} = \mathcal{A}(\mathbf{G}) = \arg\min_{\boldsymbol{\Theta}} E(\mathbf{Z}, \tilde{\mathbf{Z}}(\boldsymbol{\Theta}, \mathbf{G})),$$

where the operation $\mathcal{R}(\boldsymbol{\Theta})$ estimates the global rigid motion $\mathbf{G}$ given a fixed local motion $\boldsymbol{\Theta}$ (e.g., using the method in Section 5), and the operation $\mathcal{A}(\mathbf{G})$ estimates the local articulation $\boldsymbol{\Theta}$ given a fixed rigid global motion $\mathbf{G}$ (e.g., using the method in Section 4).

The alternation between these two operations converges to a stationary point (as proven in Appendix A). This divide-and-conquer approach has the following advantages: 1) the two decoupled estimation problems (i.e., the rigid motion and nonrigid articulation estimation) are much less difficult than the original problem and 2) many existing methods for rigid pose determination can be adopted, which makes our approach more flexible.

Sections 4 and 5 treat global rigid hand poses and local finger articulations independently. The method for finger articulation is based on global hand poses, because the 3D model projection depends on both the rigid pose and the finger joint angles. Inaccurate global poses will cause the method for local articulation estimation to mistakenly stretch and bend finger models in order to match the image observations.

Unfortunately, the pose determination method in Section 5 may induce inaccuracies since the method assumes the rigidity of the palm and matches the palm to the edges observed in the images. The inaccuracy occurs especially when the index or the little finger is straight, resulting in

(a) (b)

Fig. 10. Sample of our results on synthetic sequences. (a) A synthetic image. (b) The image with model aligned.

wrong scaling and rotation. We do observe such a phenomenon in our experiments.

We propose to tackle this difficulty by introducing more feature points for pose estimation in order to greatly reduce ambiguities. Some of these points are selected when the local finger motion is computed. For example, if we know the MCP (refer to Fig. 7a) joint of the index or the pinky finger is nonzero, we use the point at the MCP joint. If we know any of the fingers is straight, its fingertip is used. The principle is that those points lie on the same plane as the palm (on or outside the palm region certainly). Generally, these points provide bounds of the model for matching. Our extensive experiments have verified the usefulness of these extra points. Obviously, we can only find such extra points after we compute the local finger articulation.

7 Experiments

To validate and evaluate the proposed algorithms, we first performed several validation experiments on synthesized data (Section 7.1). Then, we applied our algorithm to real image sequences (Section 7.2 and 7.3). This section reports our experiments.

7.1 Validation

Since it is generally difficult to obtain the ground truth of the articulated hand motion from real video sequences, we have produced a synthetic sequence of 200 frames containing typical hand articulations. This synthetic sequence will facilitate quantitative evaluations of our algorithm.

Some examples are shown in Fig. 10. Fig. 11 shows some of the motion parameters for comparison. The solid curves are our estimates and the dash curves are the ground truth. The figure plots the x translation with an average error of 3.98 pixels, the rotation with an average error of 3.42 degrees, the PIP joint of the index finger with an average error of 8.46 degrees, the MCP flexion of the middle finger with an average error of 4.96 degrees, the PIP joint of the ring finger with an average error of 5.79 degrees, and the MCP abduction of the ring finger with an average error of 1.52 degrees. We can see from this figure that our method performs quite well.

7.2 Real Sequences: Pure Finger Articulation

In all of our experiments with real sequences, the gesturing speed is faster than what a regular camera can crisply handle. (The dataglove captures data at about 100sets/sec which is fast enough for hand gestures, but the camera can not achieve such a high rate.) Thus, when we recorded the testing video sequences, we intentionally reduced the gesturing speed of the hand in order to minimize the motion blurs produced in the recorded video. This is equivalent to using a high-speed camera.

In this set of experiments, we assume the hand has very little global motion, and allow translations in a small range. Thus, the hand motion is $(\mathbf{d}_t, \mathbf{X}_t)$, where $\mathbf{d}_t$ is global 2D translation and $\mathbf{X}_t$ is finger articulation.

We have compared three different methods for both joint angle space $\mathbb{R}^{20}$ and the configuration space $\Xi \subset \mathbb{R}^7$. The first one is a random search algorithm, which generates articulation hypotheses based on the previous estimate and a fixed Gaussian distribution without considering any constraints in the joint angle space. The second method is the Condensation algorithm. The third one is our proposed method based on learned articulation priors and importance sampling.

Some experiment results are shown in Fig. 12. Fig. 12a shows the results of random search in $\mathbb{R}^{20}$. We treat each dimension independently with a standard deviation of 5 degrees, and produce 5,000 hypotheses at each frame. However, it hardly succeeds due to the high dimensionality. When we perform random search in the reduced space $\mathbb{R}^7$ and again with 5,000 hypotheses, it loses track after several frames. The results are shown in Fig. 12b.

Fig. 12c shows some frames of the CONDENSATION algorithm in $\mathbb{R}^{20}$, in which 5,000 samples are used. The results show that it is still difficult to handle such a high dimensionality. When performing CONDENSATION in the reduced space $\mathbb{R}^7$, the algorithm can track up to 200 frames using 3,000 samples, which is shown in Fig. 12d, but cannot handle long sequences. In addition, since thousands of particles are used in both random search method and the CONDENSATION algorithm, they are computationally expensive and, thus, quite inefficient.

Finally, in our proposed algorithm, we use only 100 samples, and the algorithm is able to track hand articulations throughout the entire sequence, which is shown in Fig. 12e.[1] The joints plotted in black indicates they are bent down (i.e., showing the other side of the finger.) Our algorithm is robust and efficient since the learned articulation priors provide a strong guidance to the search and tracking process and largely reduce the search complexity. The importance sampling step in our algorithm produces particles with large weights and enhances the valid ratio of the particles. On the other hand, most of the particles will not survive the weighting process that evaluates the image measurements in both random search method and the CONDENSATION algorithm. We implemented our algorithm on a Pentium 2GHz PC and have obtained a real-time performance (about 15Hz) without code optimization.

7.3 Real Sequences: With Global Motion

We have also performed our motion capturing algorithm on real sequences with global motions. We again compared different schemes for local motion capturing. Sample results are shown in Fig. 13. The first one is a random search scheme in the $\mathbb{R}^7$ space. Our experiment used 5,000 random samples. Since this scheme does not consider the finger motion constraints, it performed poorly for local motion estimation, and it even ruined the global pose determination. The second scheme is the CONDENSATION with 3,000 samples in $\mathbb{R}^7$. It performed better than the first method, but it was not robust. We found that 3,000 samples is still not enough for this task,

1. The demo sequences of our algorithm can be obtained from http://www.ece.northwestern.edu/~yingwu/research.

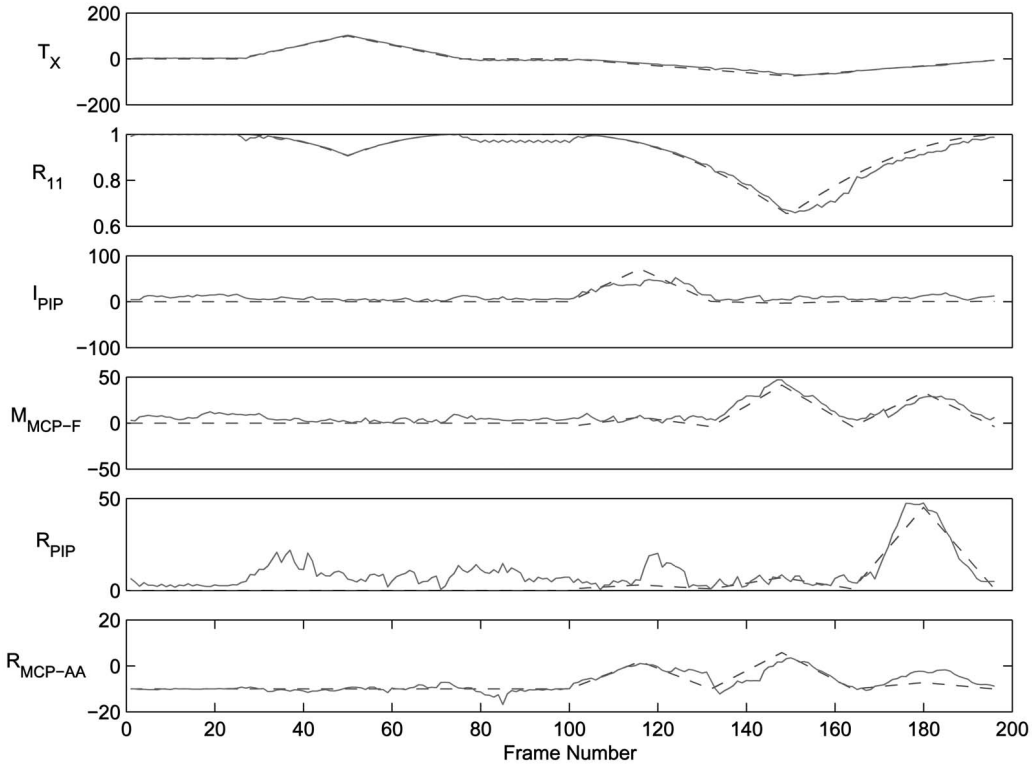

Fig. 11. The comparison of our results and the ground truth on a synthetic sequence. The dash curves are the ground truth and the solid curves are our estimates.

noticing the failure mode of the fifth one in Fig. 13b. The third scheme is our proposed method, which worked accurately and robustly. The articulation model makes the computation more efficient and the local motion estimation enhances the accuracy of hand pose determination.

7.4 Real Sequences: Using a 3D Quadric Model

Besides the cardboard model, we have also tested the proposed method with a 3D quadric model. In the testing video sequence, the fingers bend and extend while the hand moves simultaneously (Fig. 14). In addition to the superimposed model projection, a reconstructed 3D quadric model is shown below each corresponding image for better visualizations. The experiment results show that our algorithm is robust and successful in tracking complex hand motions in a cluttered environment. However, using this 3D quadric model induces much more computational cost than using the cardboard model. Our current implementation takes about 2-3s to process a frame on a Pentium 2GHz PC.

8 Conclusions

Capturing both global hand poses and local finger articulations in video sequences is a quite challenging task because of the high DoF of the articulate hand. This paper presents a divide-and-conquer approach to this problem by decoupling hand poses and finger articulations and integrating them in an iterative framework. We treat the palm as a rigid planar object and use a 3D cardboard hand model to determine the hand pose based on the ICP algorithm. Since the finger articulation is also highly constrained, we propose an articulation prior model that reduces the dimensionality of the joint angle space and characterizes the articulation manifold in the lower-dimensional configuration space. To effectively incorporate this articulation prior into the tracking process, we propose a sequential Monte Carlo tracking algorithm by using the important sampling technique. The alteration between the estimations of global hand pose and that of local finger motion results in accurate motion capturing and the proof of convergence is also given in this paper.

Our current technique assumes that the hand region can be segmented based on color from the background, which can help the image observation process. The use of a cardboard model largely simplifies the image measurement process, with the cost of sacrificing the accuracy when processing more cluttered backgrounds. We shall extend our current method to handle more clutter backgrounds. It is worth mentioning that our current global pose determination method can not handle large out-of-plane rotations and scaling very well. We will employ a better 3D model for this problem in our future work. In addition, our current system requires an user-specific calibration of the hand model which is manually done. Recently, we have developed an automatic method for tracking initialization [17] by detecting the palm and the fingers. Based on the structure from motion techniques, we shall utilize this automatic tracking initialization for automatic model calibration.

Appendix A
Proof of Convergence

Proof. Since $\boldsymbol{\Theta}^{2k} = \boldsymbol{\Theta}^{2k-1}$, apply the operation $\mathcal{R}$ to estimate global motion at the $2k\text{th}$ iteration.

$$\mathbf{G}^{2k} = \mathcal{R}\left(\boldsymbol{\Theta}^{2k-1}\right) = \arg\min_{G} E\left(\mathbf{Z}, \tilde{\mathbf{Z}}\left(\mathbf{G}, \boldsymbol{\Theta}^{2k-1}\right)\right). \quad (16)$$

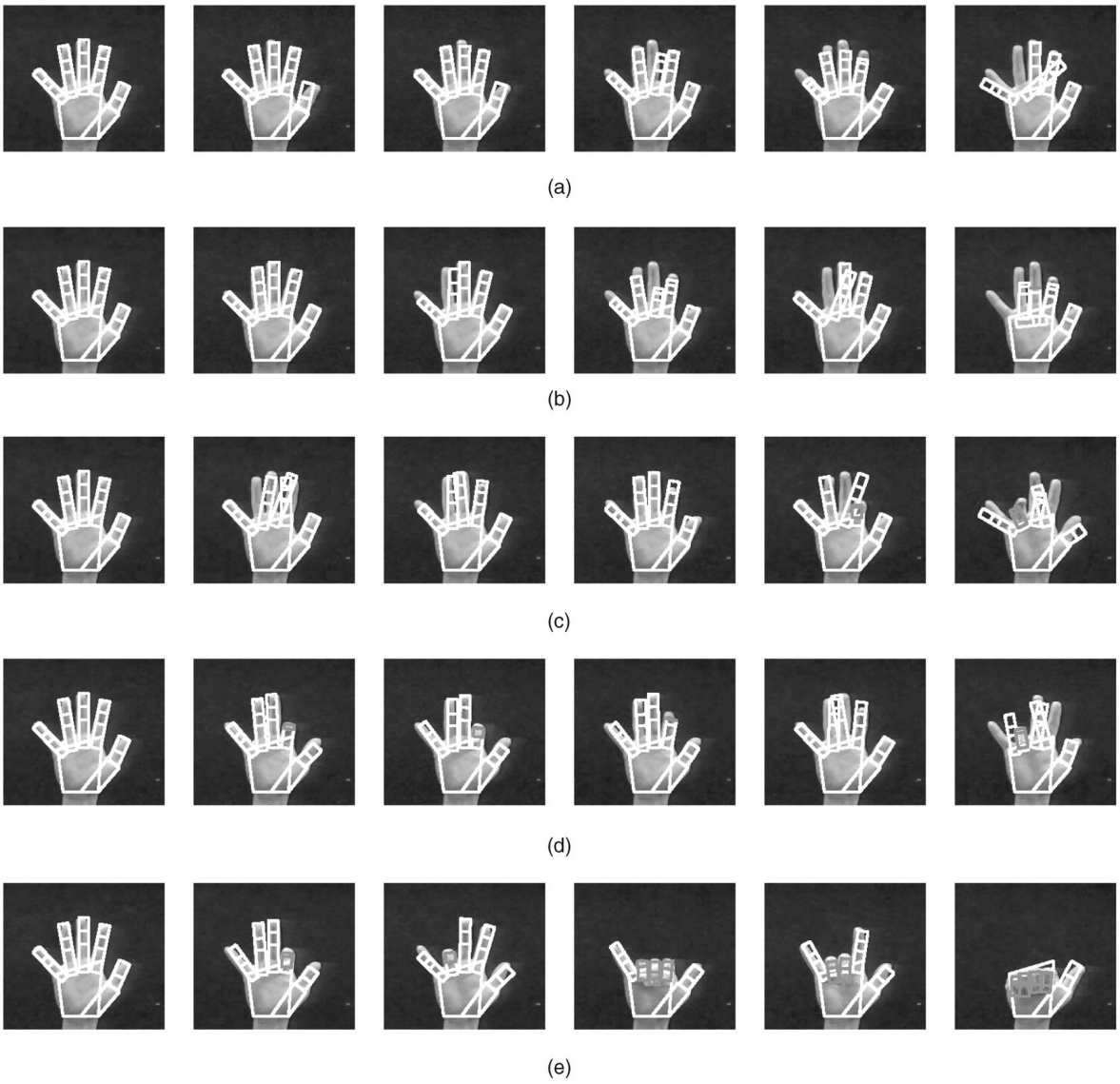

Fig. 12. Comparison of different methods. The projections of the hand model are drawn on the images. When the fingers bend and their backsides appear, the corresponding pieces are drawn in black, otherwise in white. (a) Random search 5,000 points in $\mathbb{R}^{20}$. It quickly losses track due to the high dimensionality of search space. (b) Random search 5,000 points in $\mathbb{R}^{7}$. Although dimension is reduced, the performance is still poor. (c) CONDENSATION with 5,000 samples in $\mathbb{R}^{20}$. It does not work well due to the high dimensionality of search space. (d) CONDENSATION with 3,000 samples in $\mathbb{R}^{7}$. It works fairly well without considering natural motion constraints. (e) Our approach with only 100 particles. Using our model, it can track hand articulations in a long sequence.

The error of the $2k\text{th}$ iteration is:

$$E^{2k} = E\big(\mathbf{Z}, \tilde{\mathbf{Z}}\big(\mathbf{G}^{2k}, \boldsymbol{\Theta}^{2k-1}\big)\big) = \min_{G} E\big(\mathbf{Z}, \tilde{\mathbf{Z}}\big(\mathbf{G}, \boldsymbol{\Theta}^{2k-1}\big)\big).$$

Obviously, $E^{2k} \leq E^{2k-1}$. Then, the operation $\mathcal{A}$ is applied to estimate local motion at the $(2k+1)\text{th}$ iteration:

$$\boldsymbol{\Theta}^{2k+1} = \mathcal{A}\big(\mathbf{G}^{2k}\big) = \arg\min_{\boldsymbol{\Theta}} E\big(\mathbf{Z}, \tilde{\mathbf{Z}}\big(\mathbf{G}^{2k}, \boldsymbol{\Theta}\big)\big). \quad (17)$$

Since we keep the global motion $\mathbf{G}^{2k+1} = \mathbf{G}^{2k}$, the error of the $(2k+1)\text{th}$ iteration is:

$$E^{2k+1} = E\big(\mathbf{Z}, \tilde{\mathbf{Z}}\big(\mathbf{G}^{2k}, \boldsymbol{\Theta}^{2k+1}\big)\big) = \min_{\Theta} E\big(\mathbf{Z}, \tilde{\mathbf{Z}}\big(\mathbf{G}^{2k}, \boldsymbol{\Theta}\big)\big).$$

Obviously, $E^{2k+1} \leq E^{2k}$. Thus, we have:

$$0 \leq E^{2k+1} \leq E^{2k} \leq E^{2k-1}, \quad \forall k. \quad (18)$$

Since the error measurement cannot be negative, the lower bound occurs. Because the error sequence is nonincreasing and bounded below, this two-step iterative algorithm should converge to a limit point. Furthermore, it can be shown that the algorithm converges to a stationary point. □

ACKNOWLEDGMENTS

This work was supported in part by US National Science Foundation (NSF) Grants IIS-0138965 at UIUC and NSF IIS-0347877 (CAREER) at Northwestern. The authors also greatly thank Dr. Zhengyou Zhang for the inspiring discussions and the reviewers for the constructive comments and suggestions.

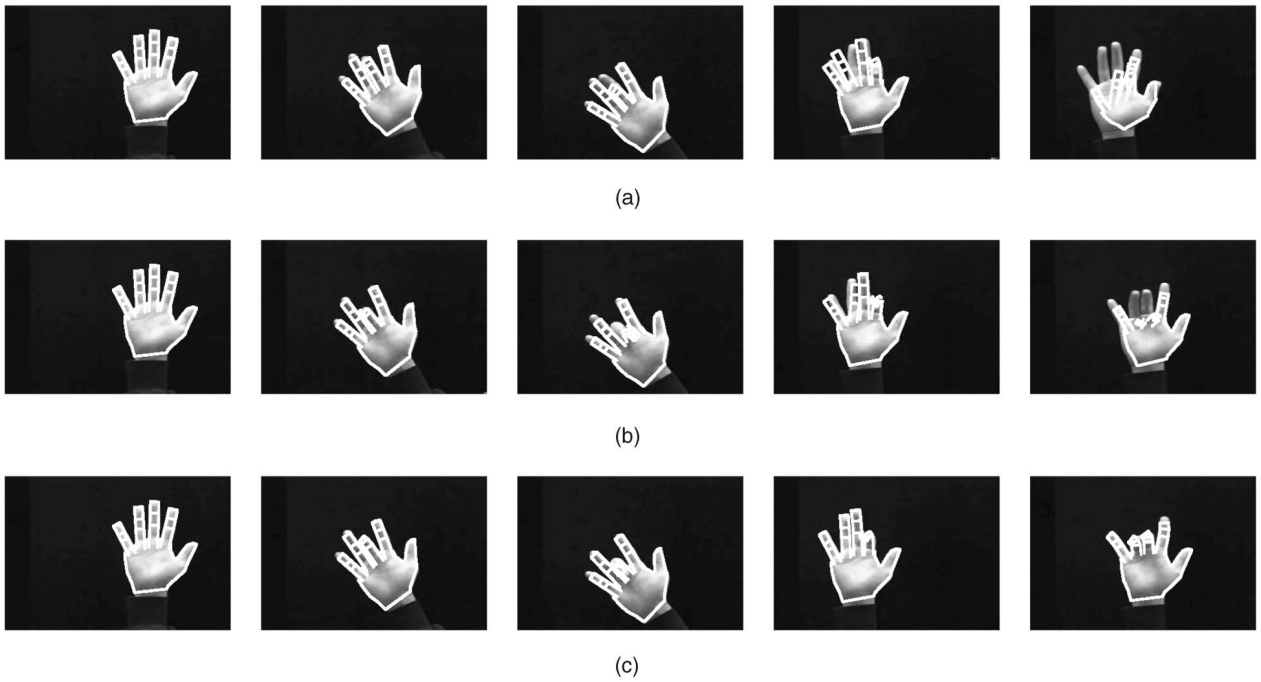

Fig. 13. Comparison of different methods on real sequences. Our method is more accurate and robust than the other two methods in our experiments. (a) Random search 5,000 points in $\mathbb{R}^7$. (b) CONDENSATION with 3,000 samples in $\mathbb{R}^7$. (c) Our approach with 100 samples.

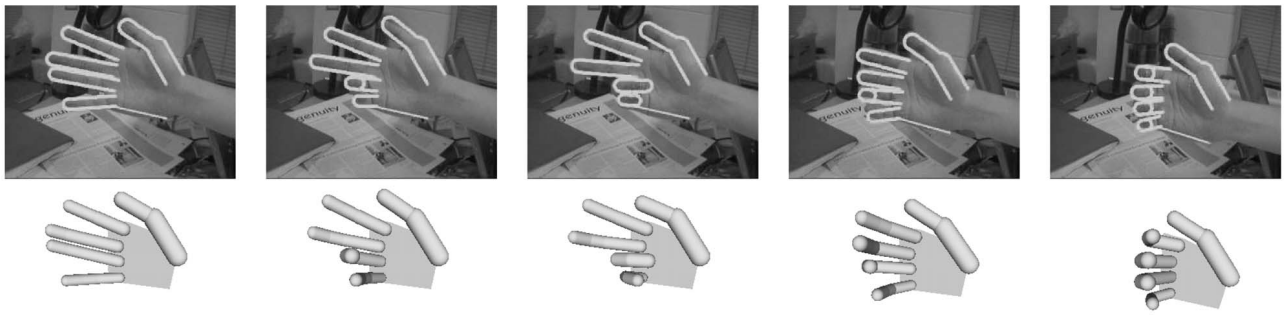

Fig. 14. Simultaneously tracking finger articulation and global hand motion. The projected edge points are superimposed with the real hand image. Below each real hand image, a corresponding reconstructed 3D hand model is shown for better visualization.

REFERENCES

[1] V. Athitsos and S. Sclaroff, "Estimating 3D Hand Pose from a Cluttered Image," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. II, pp. 432-439, June 2003.

[2] A. Blake and M. Isard, *Active Contours.* London: Springer-Verlag, 1998.

[3] M. Brand, "Shadow Puppetry," *Proc. IEEE Int'l Conf. Computer Vision,* vol. II, pp. 1237-1244, 1999.

[4] C. Bregler and S. Omohundro, "Nonlinear Image Interpolation Using Manifold Learning," *Advances in Neural Information Processing Systems 7,* G. Tesauro, D. Touretzky, and T. Leen, eds., Cambridge, Mass.: MIT Press, 1995.

[5] T.-J. Cham and J. Rehg, "A Multiple Hypothesis Approach to Figure Tracking," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 239-244, 1999.

[6] E. Chao, K. An, W. Cooney, and R. Linscheid, *Biomechanics of the Hand: A Basic Research Study.* Mayo Foundation, Minn.: World Scientific, 1989.

[7] J. Deutscher, A. Blake, and I. Reid, "Articulated Body Motion Capture by Annealed Particle Filtering," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. II, pp. 126-133, 2000.

[8] *Sequential Monte Carlo Methods in Practice,* A. Doucet, N.D. Freitas, and N. Gordon, eds., New York: Springer-Verlag, 2001.

[9] T. Heap and D. Hogg, "Towards 3D Hand Tracking Using a Deformable Model," *Proc. IEEE Int'l Conf. Automatic Face and Gesture Recognition,* pp. 140-145, 1996.

[10] T. Heap and D. Hogg, "Wormholes in Shape Space: Tracking through Discontinuous Changes in Shape," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 344-349, Jan. 1998.

[11] N. Howe, M. Leventon, and W. Freeman, "Bayesian Reconstruction of 3D Human Motion from Single-Camera Vision," *Proc. Neural Information Processing Systems,* 2000.

[12] M. Isard and A. Blake, "Contour Tracking by Stochastic Propagation of Conditional Density," *Proc. European Conf. Computer Vision,* pp. 343-356, 1996.

[13] M. Isard and A. Blake, "ICONDENSATION: Unifying Low-Level and High-Level Tracking in a Stochastic Framework," *Proc. European Conf. Computer Vision,* vol. 1, pp. 767-781, June 1998.

[14] S. Ju, M. Black, and Y. Yacoob, "Cardboard People: A Parametrized Model of Articulated Motion," *Proc. Int'l Conf. Automatic Face and Gesture Recognition,* pp. 38-44, Oct. 1996.

[15] J.J. Kuch and T.S. Huang, "Vision-Based Hand Modeling and Tracking for Virtual Teleconferencing and Telecollaboration," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 666-671, June 1995.

[16] J. Lee and T. Kunii, "Model-Based Analysis of Hand Posture," *IEEE Computer Graphics and Applications,* vol. 15, pp. 77-86, Sept. 1995.

[17] J. Lin, "Visual Hand Tracking and Gesture Analysis," PhD thesis, Dept. of Electrical and Computer Eng., Univ. of Illinois at Urbana-Champaign, Urbana, 2004.

[18] J. Lin, Y. Wu, and T.S. Huang, "Capturing Human Hand Motion in Image Sequences," *Proc. IEEE Workshop Motion and Video Computing,* pp. 99-104, Dec. 2002.

[19] J. Liu and R. Chen, "Sequential Monte Carlo Methods for Dynamic Systems," *J. Am. Statistical Assoc.,* vol. 93, pp. 1032-1044, 1998.
[20] J. Liu, R. Chen, and T. Logvinenko, "A Theoretical Framework for Sequential Importance Sampling and Resampling," *Sequential Monte Carlo in Practice,* A. Doucet, N. de Freitas, and N. Gordon, eds. New York: Springer-Verlag, 2000.
[21] S. Lu, D. Metaxas, D. Samaras, and J. Oliensis, "Using Multiple Cues for Hand Tracking and Model Refinement," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. II, pp. 443-450, June 2003.
[22] J. MacCormick and M. Isard, "Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracking," *Proc. European Conf. Computer Vision,* vol. 2, pp. 3-19, 2000.
[23] A. Mulder, "Design of Three-Dimensional Virtual Instruments with Gestural Constraints for Musical Applications," PhD thesis, Simon Fraser Univ., Canada, 1998.
[24] V. Pavlovic, R. Sharma, and T.S. Huang, "Visual Interpretation of Hand Gestures for Human Computer Interaction: A Review," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 7, pp. 677-695, July 1997.
[25] J. Rehg and T. Kanade, " Model-Based Tracking of Self-Occluding Articulated Objects," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 612-617, 1995.
[26] R. Rosales and S. Sclaroff, "Inferring Body Pose without Tracking Body Parts," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. 2, pp. 721-727, 2000.
[27] J. Segen and S. Kumar, "Shadow Gesture: 3D Hand Pose Estimation Using a Single Camera," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* pp. 479-485, 1999.
[28] N. Shimada, K. Kimura, Y. Shirai, and Y. Kuno, "Hand Posture Estimation by Combining 2-D Appearance-Based 3-D Model-Based Approaches," *Proc. Int'l Conf. Pattern Recognition,* vol. 3, pp. 709-712, 2000.
[29] N. Shimada, Y. Shirai, Y. Kuno, and J. Miura, "Hand Gesture Estimation and Model Refinement Using Monocular Camera-Ambiguity Limitation by Inequality Constraints," *Proc. Third Conf. Face and Gesture Recognition,* pp. 268-273, 1998.
[30] B. Stenger, P. Mendonca, and R. Cipolla, "Model Based 3D Tracking of an Articulated Hand," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. II, pp. 310-315, Dec. 2001.
[31] B. Stenger, A. Thayananthan, P. Torr, and R. Cipolla, "Filtering Using a Tree-Based Estimator," *Proc. IEEE Int'l Conf. Computer Vision,* vol. II, pp. 1063-1070, Oct. 2003.
[32] E. Sudderth, M. Mandel, W. Freeman, and A. Willsky, "Visual Hand Tracking Using Nonparametric Belief Propagation," *Proc. Workshop Generative Model Based Vision,* June 2004.
[33] A. Thayananthan, B. Stenger, P. Torr, and R. Cipolla, "Learning a Kinematic Prior for Tree-Based Filtering," *Proc. British Machine Vision Conf.,* vol. 2, pp. 589-598, 2003.
[34] C. Tomasi and T. Kanade, "Shape and Motion from Image Streams under Orthography—A Factorized Method," *Int'l J. Computer Vision,* vol. 9, pp. 137-154, 1992.
[35] C. Tomasi, S. Petrov, and A. Sastry, "3D Tracking = Classification + Interpolation," *Proc. IEEE Int'l Conf. Computer Vision,* vol. 2, pp. 1441-1448, Oct. 2003.
[36] J. Triesch and C. von der Malsburg, "Classification of Hand Postures against Complex Backgrounds Using Elastic Graph Matching," *Image and Vision Computing,* vol. 20, pp. 937-943, 2002.
[37] Y. Wu, G. Hua, and T. Yu, "Switching Observation Models for Contour Tracking in Clutter," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. I, pp. 295-302, June 2003.
[38] Y. Wu and T.S. Huang, "Capturing Articulated Human Hand Motion: A Divide-and-Conquer Approach," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 606-611, Sept. 1999.
[39] Y. Wu and T.S. Huang, "View-Independent Recognition of Hand Postures," *Proc. IEEE Conf. Computer Vision and Pattern Recognition,* vol. II, pp. 88-94, June 2000.
[40] Y. Wu and T.S. Huang, "Hand Modeling, Analysis and Recognition for Vision-Based Human Computer Interaction," *IEEE Signal Processing Magazine,* vol. 18, pp. 51-60, May 2001.
[41] Y. Wu, J. Lin, and T.S. Huang, "Capturing Natural Hand Articulation," *Proc. IEEE Int'l Conf. Computer Vision,* vol. II, pp. 426-432, July 2001.
[42] Z. Zhang, "Iterative Point Matching for Registration of Free-Form Curves and Surfaces," *Int'l J. Computer Vision,* vol. 13, pp. 119-152, 1994.

Ying Wu (M'01) received the BS degree from the Huazhong University of Science and Technology, Wuhan, China, in 1994, the MS degree from Tsinghua University, Beijing, China, in 1997, and the PhD degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC), Urbana, in 2001. From 1997 to 2001, he was a research assistant at the Beckman Institute for Advanced Science and Technology at UIUC. During the summer of 1999 and 2000, he was a research intern with Microsoft Research, Redmond, Washington. Since 2001, he has been an assistant professor in the Department of Electrical and Computer Engineering at Northwestern University, Evanston, Illinois. His current research interests include computer vision, computer graphics, machine learning, multimedia, and human-computer interaction. He received the Robert T. Chien Award at UIUC in 2001 and is a recipient of the US National Science Foundation CAREER award. He is a member of the IEEE and the IEEE Computer Society.

John Lin (M'04) received the BS, MS, and PhD degrees in electrical and computer engineering from the University of Illinois at Urbana-Champaign (UIUC), Urbana, in 1998, 2000, and 2004, respectively. He is currently a member of technical staff at Proximex Corp., California. He was an intern with the Mitsubishi Electric Research Lab and the IBM T.J. Watson Research Center in 2001 and 2002, respectively. His current research interests focus on issues involved in understanding and tracking articulate hand motions, surveillance systems, vision-based human computer interactions, statistical learning, and computer graphics. He is a member of the IEEE and the IEEE Computer Society.

Thomas S. Huang (S'61-M'63-SM'71-F'79) received the BS degree in electrical engineering from the National Taiwan University, Taipei, Taiwan, China, and the MS and ScD degrees in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge. He was on the faculty of the Department of Electrical Engineering at MIT from 1963 to 1973, on the faculty of the School of Electrical Engineering, and director of its Laboratory for Information and Signal Processing at Purdue University from 1973 to 1980. In 1980, he joined the University of Illinois at Urbana-Champaign, where he is now William L. Everitt Distinguished Professor of Electrical and Computer Engineering, and a research professor at the Coordinated Science Laboratory, and head of the Image Formation and Processing Group at the Beckman Institute for Advanced Science and Technology and cochair of the Institute's major research theme Human Computer Intelligent Interaction. Dr. Huang's professional interests lie in the broad area of information technology, especially the transmission and processing of multidimensional signals. He has published 21 books and more than 600 papers in network theory, digital filtering, image processing, and computer vision. He is a member of the National Academy of Engineering, a foreign member of the Chinese Academies of Engineering and Sciences, and a fellow of the International Association of Pattern Recognition, the IEEE, and the Optical Society of America, and has received a Guggenheim Fellowship, an A.V. Humboldt Foundation Senior US Scientist Award, and a Fellowship from the Japan Association for the Promotion of Science. He received the IEEE Signal Processing Society's Technical Achievement Award in 1987, and the Society Award in 1991. He was awarded the IEEE Third Millennium Medal in 2000. Also, in 2000, he received the Honda Lifetime Achievement Award for "contributions to motion analysis." In 2001, he received the IEEE Jack S. Kilby Medal. In 2002, he received the King-Sun Fu Prize, International Association of Pattern Recognition, and the Pan Wen-Yuan Outstanding Research Award. In 2003, he was appointed a professor in the Center for Advanced Study at the University of Illinois at Urbana-Champaign, the highest honor the University bestows on its faculty. In 2005, he received from UIUC School of Engineering the Tau Beta Pi D. Drucker Eminent Faculty Award. He is a founding editor of the *International Journal Computer Vision, Graphics, and Image Processing* and editor of the *Springer Series in Information Sciences*, published by Springer Verlag.