Articulated Human Body Pose Inference from Voxel Data Using a Kinematically Constrained Gaussian Mixture Model

Shinko Y. Cheng and Mohan M. Trivedi
Computer Vision and Robotics Research Laboratory
University of California, San Diego
9500 Gilman Drive MC 0434
La Jolla CA 92093-0434
{sycheng, mtrivedi}@ucsd.edu
http://cvrr.ucsd.edu/

Abstract

We present a novel method for learning and tracking the pose of an articulated body by observing only its volumetric reconstruction. We propose a probabilistic technique that utilizes a multi-component Gaussian mixture model to describe the spatial distribution of voxels in a voxel image. Each component describes a segment or rigid body, and the collection of components are kinematically constrained according to a pre-specified skeletal model. This model we refer to as a kinematically constrained Gaussian mixture model (kc-gmm). Pairs of components connected at a common joint are encouraged to assume a particular spatial configuration, forming a 1, 2 or 3 degree-of-freedom (DOF) joint. The pose learning algorithm, based on the EM algorithm, is evaluated using synthesized hand data, and the HumanEvaII dataset for facilitating algorithm comparison among different algorithms. Both datasets contain ground-truth information for accuracy measurements. A 16 component, 27 DOF mixture hand model and an 11 segment, 19 DOF mixture human body model were used. The results show that utilizing volume data, aided only by the degrees-of-freedom constraints, show accuracies of joint location estimates within 0.5cm mean-absolute-error from ground-truth with the hand data set and 17cm from subjects S2 and S4 from the HumanEvaII datasets.

1. Introduction

We present a novel method for learning and tracking the pose of an articulated body by observing only its volumetric reconstruction or "voxel image". These voxel images are the kind that can be derived from a set of images of the subject captured from various perspectives and calculated via shape-from-silhouette or other techniques [18]. The primary challenge is devising a method to efficiently and robustly extract joint parameters that describe the pose of the articulated body from unlabeled 3D voxel image sequences.

This work stems from the desire to develop accurate tether-less, vision-based articulated body pose estimation systems. These bodies may be the human body, hand, or even other creatures. Such a system have several foreseeable applications, including marker-less motion capture for human-computer interfaces, physiotherapy, 3D animation, ergonomics studies, robot control and surveillance. One of the major difficulties in recovering pose from images is the high number of degrees-of-freedom (DOF) in movement that needs to be recovered. Any rigid object requires 6 DOF to fully describe its pose. Each additional rigid object connected to it adds at least 1 DOF. A human body contains no less than 10 large body parts, equating to more than 20 DOFs. The difficulty is compounded by with the problem of self-occlusion, where body parts occlude each other depending on the configuration. Other challenges involve dealing with varying illumination which affect appearance, varying subject attire or body type, required camera configuration, required computation time.

One framework is to first extract the volume representation of the articulated body from image data. Images of the subject may be segmented, generating a silhouette image of the subject. The resulting silhouettes can then be back-projected into a series of camera rays through the silhouette back onto scene. The intersection of these rays from many images from different perspectives of the subject would constitute the visual-hull of the subject, an upper-bound to the actual volume of the articulated body [11, 12]. Even tighter bounds can be achieved if the color of the surfaces are taken into account [11, 4]. From here, the volumes are used to extract the pose of the body. This isolates the problem of finding the pose of the articulated body from

voxel data from the problem of computing the volume reconstruction. The proposed pose learning method assumes the use of only the voxel data acquired using shape-from-silhouette or other volumetric reconstruction technique.

We propose a probabilistic technique that utilizes a multi-component Gaussian mixture model to describe the spatial distribution of voxels in a voxel image. Each component describes a segment or rigid body, and the collection of components are kinematically constrained according to a pre-specified skeletal model. This model we refer to as a kinematically constrained Gaussian mixture model (kc-gmm). The kinematic constraints are in the form of a probability density function that gives a high probability when pairs of components connected at a common joint satisfies a particular spatial configuration, forming a 1, 2 or 3 degree-of-freedom (DOF) joint. This is done by incorporating a constraint function as a prior on the component means, which represent the components' location in $\mathbb{R}^3$, and covariance matrix, which represents orientation of the component. Component rotation is parameterized in terms of Euler angles. All parameters are learned using the EM algorithm.

The pose learning algorithm is evaluated using synthesized hand data, and the HumanEvaII dataset for facilitating algorithm comparison among different algorithms. Both datasets contain ground-truth information for accuracy measurements. For the case of the hand, we illustrate hand pose learning using a 16 component, 27 DOF mixture model. For the human body in the HumanEvaII dataset, we illustrate human body pose learning using a 11 segment, 19 DOF mixture model. The results show that utilizing volume data and aided by the degrees-of-freedom constraints only, this approach attain accuracies of joint location estimates within 0.5cm mean-absolute-error from ground-truth with the hand data set and 17cm from subjects S2 and S4 from the HumanEvaII datasets.

The statistical model lends itself easily to the estimation of the two attributes of the articulated body simultaneously: body structure and body pose, as both are parameters within this model. The model parameters describe 1) the dimensions of each component in height, width and depth and 2) the location and orientation of each segment or equivalently, the joint angles and component position. For this paper, the focus is on recovering body pose. However, we hope this paper will serve as an indicator of the promise that statistical clustering techniques of volume data can be used to resolve more than body pose.

In the following sections, we lay the foundation for the proposed algorithm, describing the previous research in this area in sec. 1.1. The model and learning procedure are described in sec. 2 and 3. To demonstrate the effectiveness of the proposed technique, the pose learning algorithm is evaluated on two sets of data with ground-truth: synthetic hand data and HumanEvaII Dataset. The results of these tests are described in sec. 4. Finally, the paper concludes with discussion in sec. 5.

1.1. Previous Work

There has been a tremendous amount of work in image-based recovery of articulated body pose. Several surveys of such techniques can be found in [14, 8, 9, 16]. Numerous ways have been devised to represent pose as a function of volumetric data. Each consists of a model and a fitting procedure to fit the model to the data. One of the earlier works is by Cheung *et al*. [6], where a simple k-means like algorithm is used to estimate the torso and 5 major appendages of the body (head, arms and legs). Largely to demonstrate the real-time volume reconstruction technique, no actual kinematic model was assumed.

Mikic *et al*. [13] devised a method of tracking articulated human body hierarchically, starting by detecting the head, then fitting a torso attached to the head, and then segmenting the remaining voxels to locate the upper and lower legs and arms. The strength of the Mikic approach is a self-contained initialization procedure to the tracking process, which the proposed kc-gmm method in its current state does not have. Mikic's approach however lacks generality to extend tracking articulated objects of an arbitrary skeletal structure. Furthermore, Mikic's approach can be described as top-down in nature and the proposed approach bottom-up in nature. The result is limbs at the end of the hierarchy contribute to the estimate of the whole body as much as other parts that are higher up in the hierarchy, ultimately converging at a compromise among all components.

The research most closely related to this paper is that of the constrained mixture model work by Hunter *et al*., [10]. They too utilize the concept of constraining the configuration of Gaussian components in a mixture model but in segmented 2-D images. We extended the Hunter model to describe volumes in [3]. In both these works, the model parameters were learned utilizing Expectation-Constrained-Maximization. This estimation procedure involves injecting a constraining step following the E- and M-steps to project the parameter estimates onto the kinematically feasible manifold. It is conceivable that the M-step may compete with the so-called C-step, thereby causing instability in the optimization process. Our primary contribution in this paper is incorporating these kinematic constraints (confining pairs of components to have non-zero rotation along only the specified degrees-of-freedom) into the probability model in the form of a parameter constraining prior probability. This allows us to remove the C-step completely, stay within the EM algorithm framework, and enjoy all the proven convergence properties as a result.

Two other noteworthy approaches in the volume-based pose estimation area are by Ueda *et al*. [20] and Ogawara *et al*. [15]. Their techniques are based on the iterative-

closest-point (ICP) algorithm. The differences between our approach and theirs in this case is subtle. Each algorithm arrives at the pose estimate result with roughly the same accuracy. Their approach utilizes the actual volume reconstruction itself as part of their model in which they position the joints and divide the volume into segments. In contrast, our approach requires knowing only the dimensions of the individual segments, and does not require a representative volume reconstruction for the algorithm to operate in subsequent frames. Thereby, in cases where volume data of the articulated object we wish to track is unknown before hand, e.g. partially visible driver in a car, tracking can potentially still take place. The kc-gmm approach does not currently address the issue of adaptive dimensions adjustment according to the volume data; however, the primary motivation of this approach is that body structure can also be recovered using the same paradigm of probabilistic clustering.

The most important motivation for using a probabilistic mixture model to describe volumetric reconstructions of bodies is that conceivably it can allow us to estimate body structure, which is thus-far the most elusive articulated body attribute to learn from image data. Humans can discern one rigid body part from another quite easily by examining a sequence of voxel images of a moving articulated body and determining which voxels move together with respect to others. The moving volume cue alone should be adequate to determine joint locations, joint type, and body part dimensions for many applications. Some deterministic but automatic ways have been presented illustrating this concept of grouping rigid parts from 3-D visual-hull [19, 2], 3-D color surface points [5], and 2-D image point data [21]. There is promise that a mixture model approach will serve as a basis to learn body structure. This paper solves the first problem of estimating pose using this probabilistic model, leaving structure learning for future work.

2. Kinematically Constrained Gaussian Mixture Model

The kinematically constrained Gaussian mixture model consists of the usual mixture of Gaussians model [7] with a prior probability on the constraints which in turn influences the mixture parameters. Fig. 1 shows a graphical representation of the model. If we let y_n be distributed by a mixture of K Gaussians representing K rigid body parts, z_n be the hidden membership variable, and Θ be the embodiment of the kinematic constraints and all means and covariance matrices of every Gaussian density, the density function of $Y = \{\mathbf{y}_n\}_{n=1}^N$ has the form

$$\begin{aligned} P(Y, c|\Theta) &= \prod_n P(\mathbf{y}_n, c|\Theta) \\ &= P(c|\Theta) \prod_n P(\mathbf{y}_n|c, \Theta) \\ &= P(c|\Theta) \prod_n \left[\sum_{z_n} P(\mathbf{y}_n|z_n, \Theta) P(z_n) \right] \\ &= P(c|\Theta) \prod_n \left[\sum_{z_n} \mathcal{N}(\mathbf{y}_n|\mu_{z_n}, \Sigma_{z_n}) \pi_{z_n} \right] \end{aligned} \quad (1)$$

The expression in square-brackets is the familiar mixture model. We introduce a zero-mean normally distributed random variable c which constrain components pairwise. There are altogether three forms of these constraints: spherical (3-DOF) constraint, hardy-spicer (2-DOF) constraint, and revolute (1-DOF) constraint.

Any two components connected by a joint is constrained using the spherical constraint given by

$$c_s(\Theta) = \mu_i + \mathbf{R}_{0i}\mathbf{a}_{ij} - (\mu_j + \mathbf{R}_{0j}\mathbf{a}_{ji}) \quad (2)$$

where $\mu_i, \mu_j \in \mathbb{R}^3$ are the means of components i and j, $\mathbf{R}_{0i}, \mathbf{R}_{0j} \in \mathrm{SO}(3)$ are the rotation of the components relative to the world coordinate frame, and $\mathbf{a}_{ij}, \mathbf{a}_{ji}$ point to the joint location from the component centers in component coordinate frame. This constraint represents a path from origin, to the center of one component (μ_i), to the joint shared between the two components ($\mathbf{a}_{ij}, \mathbf{a}_{ji}$ to the center of the other component (μ_j), and back to the origin. $C_s(\Theta)$ equals zero if the two components meet at the joint. Likewise, the other two constraints operate in the same manner; when the constraint given the component means and orientations equals zero, the DOF constraint is satisfied.

The hardy-spicer constraint is given by

$$c_h(\Theta) = \mathbf{R}_{0i}\mathbf{q}_{ij} \cdot \mathbf{R}_{0j}\mathbf{q}_{ji} \quad (3)$$

where $\mathbf{q}_{ij}, \hat{\mathbf{q}}_{ji}$ are the rotational axes of each component in either component coordinate frame. In this case, they each equal one of the two rotational axes. For example, if $\mathbf{q}_{ij} = (1, 0, 0)$ and $\mathbf{q}_{ji} = (0, 1, 0)$, the joint between the two components i and j is a 2-DOF joint that can rotate along the x- and y-axes with respect to either component coordinate frame.

The revolute constraint is given by

$$c_r(\Theta) = \mathbf{R}_{0i}\mathbf{q}_{ij} - \mathbf{R}_{0j}\mathbf{q}_{ji} \quad (4)$$

Again, $\mathbf{q}_{ij}, \mathbf{q}_{ji}$ represent the rotational axes. When this constraint is satisfied, the two rotational axes align resulting in a rotation along only the single DOF. Usually, $\mathbf{q}_{ij} = \mathbf{q}_{ji}$, although this need not be the case.

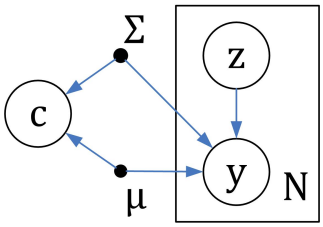

Figure 1. Graphical representation of the kinematically constrained mixture model.

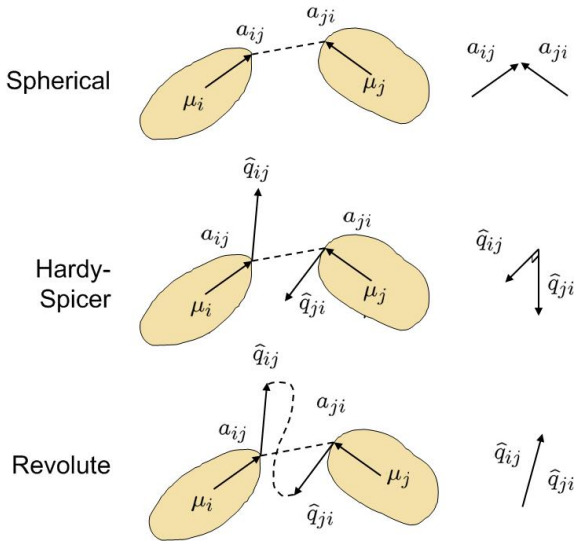

Figure 2. Illustration of the joint constraints. The right column illustrates the configuration of $\mathbf{a}$ joint location vectors and $\mathbf{q}$ rotational axes when constraints are satisfied.

Fig. 2 illustrates the mechanics of the constraints. To describe what occurs in each constraint physically, for every joint in the articulated body, c_s pulls the pair of components together to make contact at the specified joint, then c_h and c_r orients the components, possibly even translating the components such that the relative rotation between the two components are non-zero along the 2 and 1 degrees-of-freedom, respectively.

Finally, $\mathbf{R}_{0i}$ and $\mathbf{R}_{0j}$ are extracted from mixture components by parameterizing the covariance matrix of the Gaussian densities in the following way:

$$\begin{aligned}
\mathbf{y}_n &\sim \sum_{i=1}^{K} \mathcal{N}(\mathbf{y}_n|\mu_i, \Sigma_i)\pi_i \\
\Sigma_i &= \mathbf{R}_i \Lambda_i \mathbf{R}_i^\top \\
&= \mathbf{R}_{zi}\mathbf{R}_{yi}\mathbf{R}_{xi}\Lambda_i\mathbf{R}_{xi}^\top\mathbf{R}_{yi}^\top\mathbf{R}_{zi}^\top \\
&= e^{\hat{\mathbf{z}}\theta_{zi}}e^{\hat{\mathbf{y}}\theta_{yi}}e^{\hat{\mathbf{x}}\theta_{xi}}\Lambda_i e^{-\hat{\mathbf{x}}\theta_{xi}}e^{-\hat{\mathbf{y}}\theta_{yi}}e^{-\hat{\mathbf{z}}\theta_{zi}} \\
&= \Sigma(\theta_{xi}, \theta_{yi}, \theta_{zi}) \qquad (5)
\end{aligned}$$

The matrices $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, and $\hat{\mathbf{z}}$ are skew-symmetric matrices of axes of rotation, which are along the x-, y- and z-axes in world coordinate frame.

3. Learning Pose using EM

The maximum likelihood estimate of the pose is found using the EM algorithm. The E-step remains the same as the E-step for the standard Gaussian mixture model. The M-step becomes an optimization over the log-likelihood of the mixture model summed with the log-likelihood of c, the constraint on the component position (means) and orientation (covariance matrices). A closed-form expression for the mean can be found, but the orientation of each component is estimated using gradient ascent.

Using equ. 1, the problem of finding the ML estimate of Θ can be stated as maximizing the following log-likelihood equation:

$$\hat{\Theta}_{ML} = \arg\max_{\Theta} [\ln P(Y|\Theta) + \ln P(c|\Theta)] \qquad (6)$$

By introducing the 1) hidden membership variable z_n, 2) its distribution function $q(z_n)$ as of yet unknown, and 3) the posterior of z_n, and then rearranging the terms, we reveal the expression equivalent to the log-likelihood which can then be more readily maximized.

$$\begin{aligned}
&\ln P(Y|\Theta) \\
&= \sum_n \ln P(\mathbf{y}_n|\Theta) \sum_{z_n} q(z_n) - KL(q\|p) + KL(q\|p) \\
&= \sum_n \sum_{z_n} \left[q(z_n) \ln \frac{P(\mathbf{y}_n, z_n|\Theta)}{q(z_n)} - q(z_n) \ln \frac{P(z_n|\mathbf{y}_n, \Theta)}{q(z_n)} \right] \\
&= \mathcal{L}(q, \Theta) + \mathrm{KL}(q\|p) \qquad (7)
\end{aligned}$$

$\mathrm{KL}(q\|p)$ is the Kullback-Liebler divergence, $\sum_{z_n} q(z_n) = 1$, and $\mathcal{L}$ is the so-called lower-bound of the incomplete log-likelihood. Finally substituting equ. 7 back into equ. 6, we arrive at the desired expression.

$$\begin{aligned}
&\ln P(Y|\Theta) + \ln P(c|\Theta) \\
&\quad = \mathcal{L}(q, \Theta) + \mathrm{KL}(q\|p) + \ln P(c|\Theta) \\
&\quad \geq \mathcal{L}(q, \Theta) + \ln P(c|\Theta) \qquad (8)
\end{aligned}$$

Maximizing the log-likelihood lower-bound $\mathcal{L}(q, \Theta)$ is equivalent to maximizing the log-likelihood. To find the $\hat{\Theta}_{\mathrm{ML}}$, we iteratively hold the parameters Θ fixed and find distribution function q that maximizes the equation (E-step), then hold q fixed and find Θ that maximizes the log-likelihood(M-step). Details of the EM algorithm and its derivation can be found in [1].

3.1. E-Step: Solving for $q(z_n)$

The E-Step consists of evaluating the posterior probability of the hidden variable z_n while holding the parameters

fixed.

$$
\begin{aligned}
q(z_n) &= p(z_n \,|\mathbf{y}_n, \Theta^{\text{old}}) \\
&= \frac{\mathcal{N}(\mathbf{y}_n|\mu_{z_n}, \Sigma(\theta_{z_n}))\pi_{z_n}}{\sum_{z_n} \mathcal{N}(\mathbf{y}_n|\mu_{z_n}, \Sigma(\theta_{z_n}))\pi_{z_n}} \\
&= \alpha_{z_n,i} \qquad (9)
\end{aligned}
$$

When $q(z_n)$ equals the posterior of z_n, the KL divergence $KL(q\|p)$ equals zero while maintaining the same values for the incomplete log-likelihood.

3.2. M-Step: Solving for π_i and μ_i

The M-step consists of maximizing $\mathcal{L}(q, \Theta)$ over Θ.

$$
\begin{aligned}
\hat{\Theta}_{\text{ML}} &= \arg\max_{\Theta} \mathcal{L}(q^{\text{old}}, \Theta) \qquad (10) \\
&= \arg\max_{\Theta} \sum_n \sum_{z_n} q^{\text{old}} \ln P(\mathbf{y}_n, z_n|\Theta) + \ln P(c|\Theta) \qquad (11)
\end{aligned}
$$

The parameters Θ consist of the means of each component μ_i, the orientation of each component in Euler angles θ_i, and the class prior probability π_i for all components i.

Because $P(c|\Theta)$ does not depend on the class prior probabilities π_i, they can be found the same way as learning them for the standard mixture model by evaluating

$$
\hat{\pi}_i = \frac{1}{N} \sum_i \alpha_{z_n,i} \qquad (12)
$$

where N is the number of voxels.

Only the incomplete log-likelihood and spherical constraint probability $P(c_s|\Theta)$ depend on the component mean μ_i. A single component may have one or several joints constrained by the spherical joint constraint probability. Solving for μ_i involves setting the gradient of $\mathcal{L}(q, \Theta)$ with respect to all μ_i to equal zero, and solving for μ_i for all i simultaneously using Least Squares. The gradient of the log-likelihood and spherical constraint probability is given by

$$
\begin{aligned}
&\nabla_{\mu_i} \sum_n \ln P(\mathbf{y}_n|z_n = i, \Theta) \\
&\quad = \sum_n \alpha_{i,n} \Sigma(\theta_i)^{-1}(\mathbf{y}_n - \mu_i) \qquad (13) \\
&\nabla_{\mu_i} P(c_s|\Theta) \\
&\quad = -\nabla_{\mu_i} c_s^\top \Sigma_{c_s}^{-1} c_s \\
&\quad = -\Sigma_{c_s}^{-1}(\mu_i + \mathbf{R}_i \mathbf{a} ij - (\mu_j + \mathbf{R}_j \mathbf{a}_{ji})) \qquad (14) \\
&\nabla_{\mu_j} P(c_s|\Theta) \\
&\quad = +\nabla_{\mu_j} c_s^\top \Sigma_{c_s}^{-1} c_s \\
&\quad = +\Sigma_{c_s}^{-1}(\mu_i + \mathbf{R}_i \mathbf{a} ij - (\mu_j + \mathbf{R}_j \mathbf{a}_{ji})) \qquad (15)
\end{aligned}
$$

Through judicious rearrangement of terms, one can construct a system of equations for this component mean μ_i and all other component means.

$$
\begin{aligned}
&\left[-\Sigma_{c_s}^{-1} \cdots \sum_n \alpha_{i,n} \Sigma(\theta_i)^{-1} + \gamma \Sigma_{c_s}^{-1} \cdots - \Sigma_{c_s}^{-1} \right] \begin{bmatrix} \vdots \\ \mu_j \\ \vdots \\ \mu_i \\ \vdots \\ \mu_k \\ \vdots \end{bmatrix} \\
&= \sum_n \alpha_{i,n} \Sigma(\theta_i)^{-1} \mathbf{y}_n \\
&\quad + \Sigma_{c_s}^{-1}(\mathbf{R}_j \mathbf{a}_{ji} - \mathbf{R}_i \mathbf{a}_{ij}) + \Sigma_{c_s}^{-1}(\mathbf{R}_i \mathbf{a}_{ik} - \mathbf{R}_k \mathbf{a}_{ik}) \qquad (16)
\end{aligned}
$$

The component means that maximizes the log-likelihood $\hat{\mu}_i \forall i$ is the least squares solution of the system of equations shown in equ. 16.

3.3. M-Step: Solving for θ_i

Finally, to solve for the orientation θ_i of each component, we need to consider the incomplete log-likelihood and all relevant constraint probabilities $P(c_s|\Theta)$, $P(c_h|\Theta)$, and $P(c_e|\Theta)$). Gradient ascent is employed for this task. The gradient of $\mathcal{L}(q^{\text{old}}, \Theta)$ with respect to θ_i is used to iteratively step toward the solution until convergence using the update equation

$$
\theta_i^{[n+1]} = \theta_i^{[n]} + \alpha_n \nabla_{\theta_i} \mathcal{L} \qquad (17)
$$

The gradient of $\mathcal{L}(q^{\text{old}}, \Theta)$ with respect to θ_i is given by

$$
\begin{aligned}
&\nabla_{\theta_i} \mathcal{L} = \nabla_{\theta_i} \ln P(\mathbf{y}_n|\Theta) \\
&\quad + \nabla_{\theta_i} \left[\ln P(c_s|\Theta) + \ln P(c_h|\Theta) + \ln P(c_e|\Theta) \right] \qquad (18)
\end{aligned}
$$

All constraints are included in equ. 18, but each joint will utilize at most two of the above constraints. The gradient of the other constraints will equate to zero in those cases. All constraint probabilities are shown here to illustrate the positioning of the gradients.

The gradient of the incomplete likelihood is given by

$$
\nabla_{\theta_i} \ln P(\mathbf{y}_n|\Theta) = -\sum_n \frac{\alpha_{i,n}}{2} \nabla_{\theta_i} m_i(\theta_i) \qquad (19)
$$

where

$$
\begin{aligned}
&\nabla_{\theta_i} m_i(\theta_i) = \\
&\begin{bmatrix} 2(\mathbf{y}_n - \mu_i)^\top e^{\hat{\mathbf{z}}\theta_z} e^{\hat{\mathbf{y}}\theta_y} \hat{\mathbf{x}} e^{\hat{\mathbf{x}}\theta_x} \Lambda^{-1} \mathbf{R}^\top (\mathbf{y}_n - \mu_i) \\ 2(\mathbf{y}_n - \mu_i)^\top e^{\hat{\mathbf{z}}\theta_z} \hat{\mathbf{y}} e^{\hat{\mathbf{y}}\theta_y} e^{\hat{\mathbf{x}}\theta_x} \Lambda^{-1} \mathbf{R}^\top (\mathbf{y}_n - \mu_i) \\ 2(\mathbf{y}_n - \mu_i)^\top \hat{\mathbf{z}} e^{\hat{\mathbf{z}}\theta_z} e^{\hat{\mathbf{y}}\theta_y} e^{\hat{\mathbf{x}}\theta_x} \Lambda^{-1} \mathbf{R}^\top (\mathbf{y}_n - \mu_i) \end{bmatrix} \qquad (20)
\end{aligned}
$$

The gradient for the constraint probabilities all follow the form

$$\begin{aligned}\nabla_{\theta_i} \ln P(c|\Theta) &= \nabla_{\theta_i} \ln \mathcal{N}(c|\mathbf{0}, \Sigma) \\ &= \nabla_{\theta_i} \left[-\frac{1}{2} c^\top \Sigma^{-1} c \right] \\ &= -(\nabla_{\theta_i} c)\Sigma^{-1} c \end{aligned} \tag{21}$$

Depending on the the ordering of pairs of components, the gradient is of a particular form. In other words, for a particular constraint equation, if the first component is i or the "head", and the second component is j or the "tail", the second component must always be the j to the first component's i. This head-tail relationship must remain consistent throughout the calculation of the constraint probabilities and its gradients.

For the spherical constraint probability (which is used by every joint), the gradient is found by

$$\begin{aligned}&\nabla_{\theta_i} \ln P(c_s|\Theta) \\ &\quad = -\nabla_{\theta_i}(\mathbf{R}_i \mathbf{a}_{ij}) \Sigma_s^{-1} (\mu_i + \mathbf{R}_i \mathbf{a}_{ij} - (\mu_j + \mathbf{R}_j \mathbf{a}_{ji})) \\ &\quad = -\begin{bmatrix} (e^{\hat{\mathbf{z}}\theta_{zi}} e^{\hat{\mathbf{y}}\theta_{yi}} \hat{\mathbf{x}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{a}_{ij})^\top \\ (e^{\hat{\mathbf{z}}\theta_{zi}} \hat{\mathbf{y}} e^{\hat{\mathbf{y}}\theta_{yi}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{a}_{ij})^\top \\ (\hat{\mathbf{z}} e^{\hat{\mathbf{z}}\theta_{zi}} e^{\hat{\mathbf{y}}\theta_{yi}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{a}_{ij})^\top \end{bmatrix} \Sigma_{c_s}^{-1} c_s \end{aligned} \tag{22}$$

$$\begin{aligned}&\nabla_{\theta_j} \ln P(c_s|\Theta) \\ &\quad = -\nabla_{\theta_j}(-\mathbf{R}_j \mathbf{a}_{ji}) \Sigma_s^{-1} (\mu_i + \mathbf{R}_i \mathbf{a}_{ij} - (\mu_j + \mathbf{R}_j \mathbf{a}_{ji})) \\ &\quad = +\begin{bmatrix} (e^{\hat{\mathbf{z}}\theta_{zj}} e^{\hat{\mathbf{y}}\theta_{yj}} \hat{\mathbf{x}} e^{\hat{\mathbf{x}}\theta_{xj}} \mathbf{a}_{ji})^\top \\ (e^{\hat{\mathbf{z}}\theta_{zj}} \hat{\mathbf{y}} e^{\hat{\mathbf{y}}\theta_{yj}} e^{\hat{\mathbf{x}}\theta_{xj}} \mathbf{a}_{ji})^\top \\ (\hat{\mathbf{z}} e^{\hat{\mathbf{z}}\theta_{zj}} e^{\hat{\mathbf{y}}\theta_{yj}} e^{\hat{\mathbf{x}}\theta_{xj}} \mathbf{a}_{ji})^\top \end{bmatrix} \Sigma_{c_s}^{-1} c_s \end{aligned} \tag{23}$$

The gradient for the hardy-spicer joint is given by

$$\begin{aligned}&\nabla_{\theta_i} \ln P(c_h|\Theta) \\ &\quad = -\nabla_{\theta_i}(\mathbf{q}_{ij}^\top \mathbf{R}_i^\top \mathbf{R}_j \mathbf{q}_{ji}) \Sigma_h^{-1} (\mathbf{q}_{ij}^\top \mathbf{R}_i^\top \mathbf{R}_j \mathbf{q}_{ji}) \\ &\quad = -\begin{bmatrix} \mathbf{q}_{ji}^\top \mathbf{R}_j^\top e^{\hat{\mathbf{z}}\theta_{zi}} e^{\hat{\mathbf{y}}\theta_{yi}} \hat{\mathbf{x}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{q}_{ij} \\ \mathbf{q}_{ji}^\top \mathbf{R}_j^\top e^{\hat{\mathbf{z}}\theta_{zi}} \hat{\mathbf{y}} e^{\hat{\mathbf{y}}\theta_{yi}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{q}_{ij} \\ \mathbf{q}_{ji}^\top \mathbf{R}_j^\top \hat{\mathbf{z}} e^{\hat{\mathbf{z}}\theta_{zi}} e^{\hat{\mathbf{y}}\theta_{yi}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{q}_{ij} \end{bmatrix} \Sigma_{c_h}^{-1} c_h \\ &\nabla_{\theta_j} \ln P(c_h|\Theta) \\ &\quad = -\nabla_{\theta_j}(\mathbf{q}_{ij}^\top \mathbf{R}_i^\top \mathbf{R}_j \mathbf{q}_{ji}) \Sigma_h^{-1} (\mathbf{q}_{ij}^\top \mathbf{R}_i^\top \mathbf{R}_j \mathbf{q}_{ji}) \\ &\quad = -\begin{bmatrix} \mathbf{q}_{ji}^\top (-\hat{\mathbf{x}}) e^{-\hat{\mathbf{x}}\theta_{xj}} e^{-\hat{\mathbf{y}}\theta_{yj}} e^{-\hat{\mathbf{z}}\theta_{zj}} \mathbf{R}_i \mathbf{q}_{ij} \\ \mathbf{q}_{ji}^\top e^{-\hat{\mathbf{x}}\theta_{xj}} (-\hat{\mathbf{y}}) e^{-\hat{\mathbf{y}}\theta_{yj}} e^{-\hat{\mathbf{z}}\theta_{zj}} \mathbf{R}_i \mathbf{q}_{ij} \\ \mathbf{q}_{ji}^\top e^{-\hat{\mathbf{x}}\theta_{xj}} e^{-\hat{\mathbf{y}}\theta_{yj}} (-\hat{\mathbf{z}}) e^{-\hat{\mathbf{z}}\theta_{zj}} \mathbf{R}_i \mathbf{q}_{ij} \end{bmatrix} \Sigma_{c_h}^{-1} c_h \end{aligned} \tag{24}$$

The gradient for the elbow joint is given by

$$\begin{aligned}&\nabla_{\theta_i} \ln P(c_e|\Theta) \\ &\quad = -\nabla_{\theta_i}(\mathbf{R}_i \mathbf{q}_{ij}) \Sigma_e^{-1} (\mathbf{R}_i \mathbf{q}_{ij} - \mathbf{R}_j \mathbf{q}_{ji}) \\ &\quad = -\begin{bmatrix} (e^{\hat{\mathbf{z}}\theta_{zi}} e^{\hat{\mathbf{y}}\theta_{yi}} \hat{\mathbf{x}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{q}_{ij})^\top \\ (e^{\hat{\mathbf{z}}\theta_{zi}} \hat{\mathbf{y}} e^{\hat{\mathbf{y}}\theta_{yi}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{q}_{ij})^\top \\ (\hat{\mathbf{z}} e^{\hat{\mathbf{z}}\theta_{zi}} e^{\hat{\mathbf{y}}\theta_{yi}} e^{\hat{\mathbf{x}}\theta_{xi}} \mathbf{q}_{ij})^\top \end{bmatrix} \Sigma_{c_e}^{-1} c_e \end{aligned} \tag{25}$$

$$\begin{aligned}&\nabla_{\theta_j} \ln P(c_e|\Theta) \\ &\quad = -\nabla_{\theta_j}(-\mathbf{R}_j \mathbf{q}_{ji}) \Sigma_e^{-1} (\mathbf{R}_i \mathbf{q}_{ij} - \mathbf{R}_j \mathbf{q}_{ji}) \\ &\quad = +\begin{bmatrix} (e^{\hat{\mathbf{z}}\theta_{zj}} e^{\hat{\mathbf{y}}\theta_{yj}} \hat{\mathbf{x}} e^{\hat{\mathbf{x}}\theta_{xj}} \mathbf{q}_{ji})^\top \\ (e^{\hat{\mathbf{z}}\theta_{zj}} \hat{\mathbf{y}} e^{\hat{\mathbf{y}}\theta_{yj}} e^{\hat{\mathbf{x}}\theta_{xj}} \mathbf{q}_{ji})^\top \\ (\hat{\mathbf{z}} e^{\hat{\mathbf{z}}\theta_{zj}} e^{\hat{\mathbf{y}}\theta_{yj}} e^{\hat{\mathbf{x}}\theta_{xj}} \mathbf{q}_{ji})^\top \end{bmatrix} \Sigma_{c_e}^{-1} c_e \end{aligned} \tag{26}$$

(27)

4. Evaluation

To demonstrate the validity and generality of this approach, the proposed model is constructed for 2 types of articulated bodies: a 16 component, 15 joint hand, and a 10 component, 9 joint human body. The volumetric reconstructions were synthetically generated for the case of the hand, and generated using shape-from-silhouette using HumanEva II [17] image data in the case of the human body.

In both cases, ground-truth information about the position of the articulated body is available for comparison with the estimated results. The measures of accuracy used for both hand and human body test sequences is joint position error, proposed to be the standard measure of error [17]. Component (or segment) position and orientation error is also used for the hand case, for comparison.

For each test, the body model is manually sized and positioned near the actual voxel reconstruction of the body, and the parameters of the model mixtures is initialized accordingly. Fig. 3 shows the model configured and result of the pose estimate in the first frame following initial model placement.

To generate voxels for the hand, a cylinder of voxels is positioned in the space described by the body model. A sequence of 430 "voxel images" were generated of fingers bending from 0 to 90 degrees in a wave-like pattern while the palm also rotated. A few frames from the result of pose learning on this sequence are shown in fig. 9. Note that the hand closes to a fist twice.

To measure its accuracy, the orientation and position of the individual segments of the estimated and ground-truth values are compared. Various statistics of the error were calculated, including mean absolute, root mean square, median, mode, and 95-th percentile. The histogram of component center position and angular orientation error over all components are illustrated in fig. 4. The same statistics by component is illustrated in 5. With a voxel resolution of 0.5 cm to the side of each voxel, component center posi-

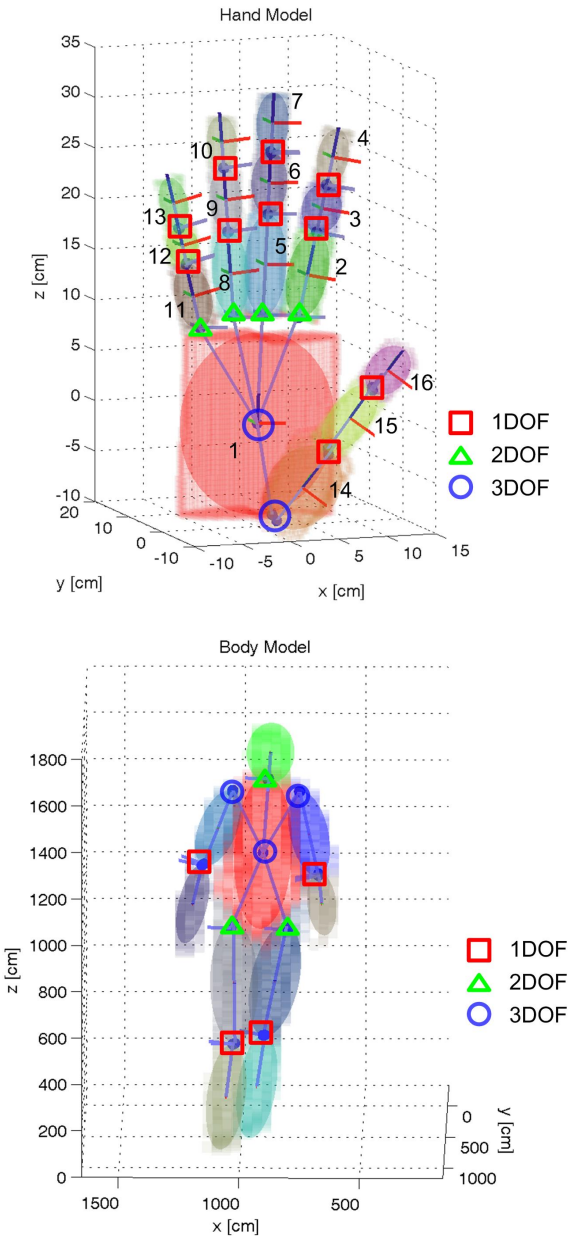

Figure 3. Articulated body models in evaluation. Joint and joint types are annotated according to the number of degrees of freedom. The center of the palm and torso are used as the center of the body and carry an additional 3 degrees of freedom for translational movement. Dimensions are based on actual bodies.

tional accuracy of 0.33 cm mean absolute error could be achieved with the hand sequence. Mean angular error measured 8.51°. Using the joint-position error metric, mean absolute error yields 0.5 cm joint position error. Fig. 6 shows the statistics of this error overall and by component. The dimensions are based on an actual hand, so the results from this test are representative of the accuracy of learning hand pose with ideal reconstructed volumes.

The next dataset used to test the algorithm is the HumanEva II dataset [17]. Subject 2 and 4 were both used. Shape-from-silhouette is used to generate the voxel reconstruction of the human subject. Silhouettes are generated by background subtraction in the HSI color space, followed by connected component analysis, retaining only the largest components.

The body model is sized and positioned over the voxel reconstruction for the first frame, and then the algorithm is allowed to process the remaining frames. Three segments of S2 sequence were processed separately: 1–190, 250–500, and 700–1220. The entire S4 sequence was processed in a single run. Several frames from the pose learning results in S2 are illustrated in fig. 10.

The performance measures consist of absolute and relative 3D spatial error of joint locations using mean Euclidean distance. Relative 3D joint location error is calculated relative to the torso point. All joints as prescribed in [17] were utilized in the error measurement. A summary of the results are tabulated in tab. 1. The mean joint position error over time and histogram are shown in fig. 7 and 8.

The S2 sequence was processed in 3 segments because of loss of tracking during frames 190–250 and 500–700. Within these ranges, the subject was turning at a particular position in the space such that the voxel reconstruction produced a diamond shaped reconstruction as observed from the top. This particular camera configuration is unable to carve away the erroneous volume in this situation. This causes the algorithm to fall into an erroneous local maxima of a pose that is turned 90 degrees from the correct pose along the length of the body. Just as the EM-algorithm is subject to convergence in local-maxima, this algorithm is no different and algorithm does not recover from this. One should keep in mind that these results are from a generic pose learning algorithm that uses only volume information, albeit the silhouettes are derived from imagery. This problem is left to be investigated in the future.

In sequence S4, this phenomenon can be seen between frames 350–600 when the error jumps from 17 cm to 35 cm. During this period, the algorithm is tracking the body with the model pose reversed (left is right and vice versa). As subject 4 walked around the second time, the model reversed a second time and the remained in the correct direction for processing for the remaining frames. We suspect the sudden spike that extends to 1.2 meters starting at around frame 300 comes from the ground-truth.

The primary reasons for the large differences in accuracy between the hand and human body test sequences is due to the discrepencies between the body part dimensions in the model and reality, and due to the quality of the voxel resconstruction. The body model used for this test were visually approximated by placing the model components in the voxel

reconstruction, part by part. This implies that the joint locations were also approximated in the same way. This results in the consistently greater than 8cm error in both subjects. As compared to the hand model, the model itself was used to generate the voxels and little descrepency resulted.

The loss of track as well as diminished of accuracy in the human body test sequence can also be attributed to the quality of the voxel reconstruction. While the hand sequence can be considered ideal voxel reconstructions, the human body voxel reconstructions are limited by several factors. Finite number of cameras and the given camera configuration results in poorer voxel reconstruction in some regions of scene compared to others, as described above with the diamond shape reconstruction. This is a limitation in shape-from-silhouette [12, 11]. Another source of error is in segmentation. Although shadows were mostly eliminated by using the HSI color space, dark areas of the subject in the scene looked very similar to shadow and was sometimes excluded in the silhouette, carving out valid areas in the voxel reconstruction.

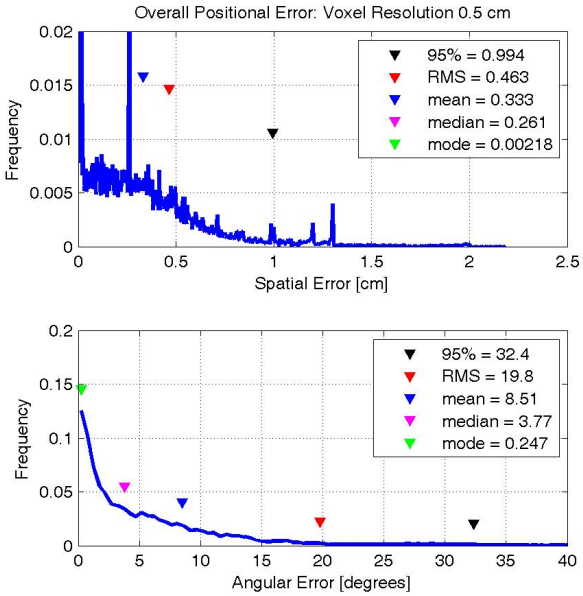

Figure 4. KC-GMM Estimation Error: Histogram of component center and angular orientation estimation error with respect to ground-truth for synthetic hand data.

5. Discussion and Concluding Remarks

Because the model represents an articulated body model that only encourages configurations of Gaussian components where the joints will have the specified 1, 2, and 3 DOF, limbs bending in infeasible directions are possible solutions in the learning process. Additional constraints are required to limit the range of motion in the joints. This will also improve the tracking performance from one frame to the next by eliminating some erroneous local optima in the pose space.

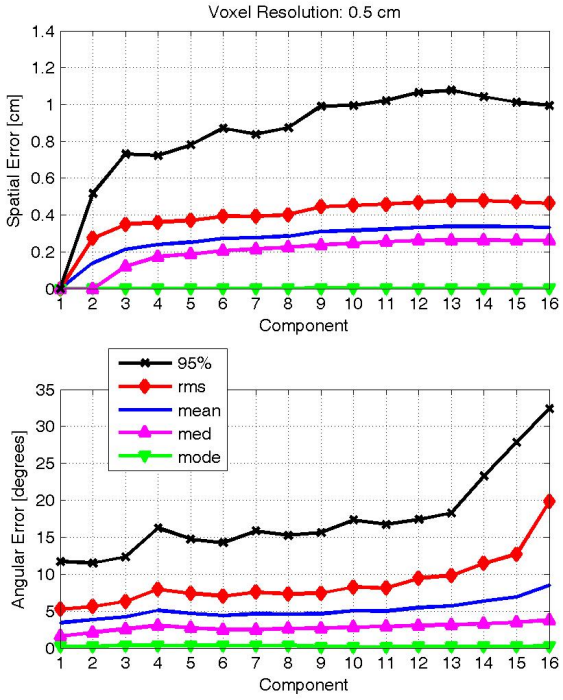

Figure 5. KC-GMM Estimation Error: Mode, median, mean, RMS, and 95th-percentile statistics of component center and angular orientation error with respect to ground-truth from synthetic hand data, by component.

The model currently does not contain a mechanism to utilize temporal information from the last processed frame, such as velocity, with which this system will benefit. The results shown merely utilize the last estimate as a starting point for learning the pose.

Despite both these shortcomings, it is clear that volumetric data alone is sufficient to recover the pose of a hand even through a nearly closed fist. The model is general enough to be easily extended for other articulated bodies. The primary contribution of this paper is a kinematically constrained Gaussian mixture model that relates volume data and the pose of the articulated body pose and the means to learn the pose using the EM algorithm; no additional constraint optimization steps external to the EM algorithm are required. The algorithm was validated on two types of articulated bodies.

We believe that the constraining of components to be one module in a complete articulated body *structure* and pose learning algorithm. Using Gaussian mixtures as the basis of representing volumes, it is conceivable several smaller components can represent a single rigid body, akin to sev-

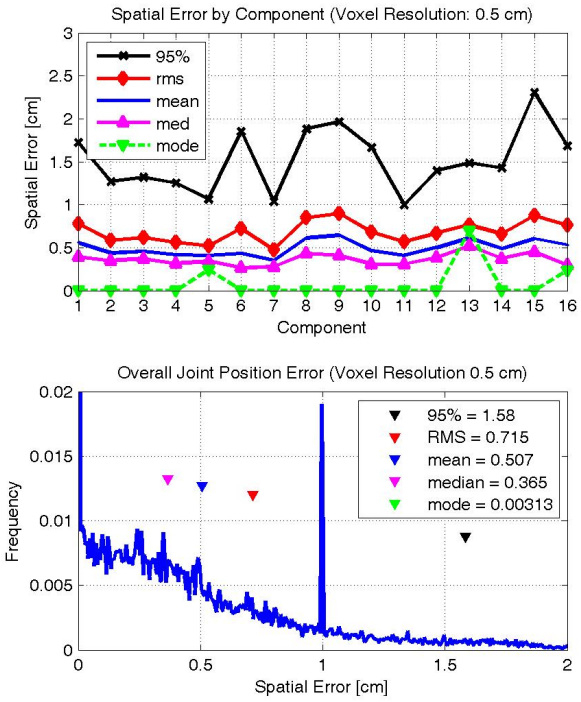

Figure 6. KC-GMM Estimation Error: Overall and by-component *joint* position error with respect to ground-truth on synthetic hand data. 3-D joint position error, as opposed to component center and orientation error, is the proposed standard estimation error measures for the HumanEvaII data set.

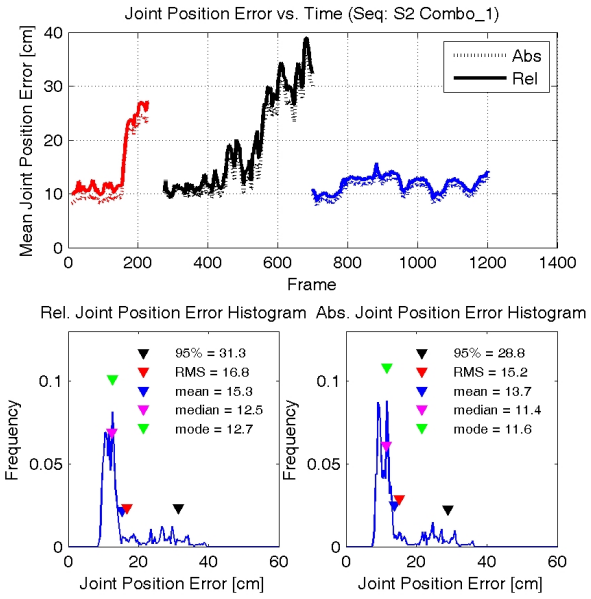

Figure 7. Overall joint position error from ground-truth of articulated body pose learning on HumanEvaII human body data set (Subject 2 Combo Trial 1).

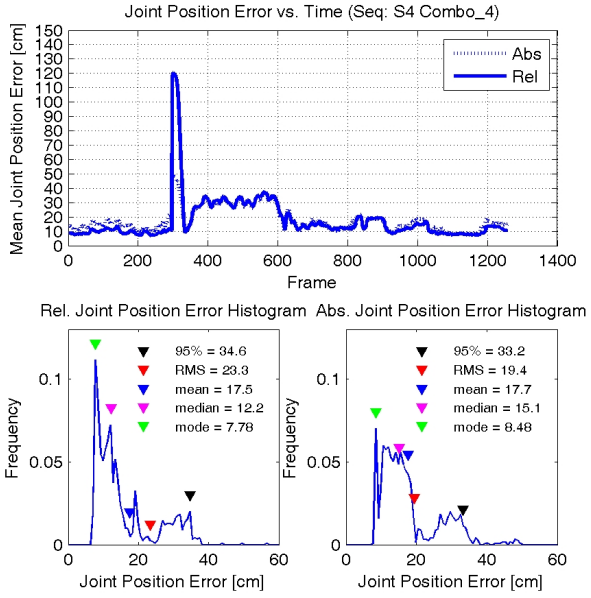

Figure 8. Overall joint position error from ground-truth of articulated body pose learning on HumanEvaII human body data set (Subject 4 Combo Trial 4).

eral atoms making up the larger whole. A path for further investigation can be to augment the model to constrain the range of motion, incorporate temporal cues. Eventually, the investigation can lead to a structure learning computational framework that begins with composing a volume with several small Gaussian components, and components that move together over several frames can meld together into larger components or components connected by joints constrained by the 1, 2, and 3 DOF joint constraints as described here.

Acknowledgements

We thank the Digital Media Innovations Porgram (DiMI) of UC Discovery Grants and Volkswagen-Audi for their sponsorship. We also thank our fellow researchers at the CVRR for their cooperation.

References

[1] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer Science+Business Media, LLC, 2006. 4

[2] G. J. Browstow, I. Essa, D. Steedly, and V. Kwatra. Novel skeletal representation for articulated creatures. In *European Conference on Computer Vision Conference*, volume 3, pages 66–78, May. 2004. 3

[3] S. Y. Cheng and M. M. Trivedi. Multimodal voxelization and kinematically constrained gaussian mixture model for full hand pose estimation: An integrated systems approach. In *IEEE Proc. Int. Conference on Computer Vision Systems*,

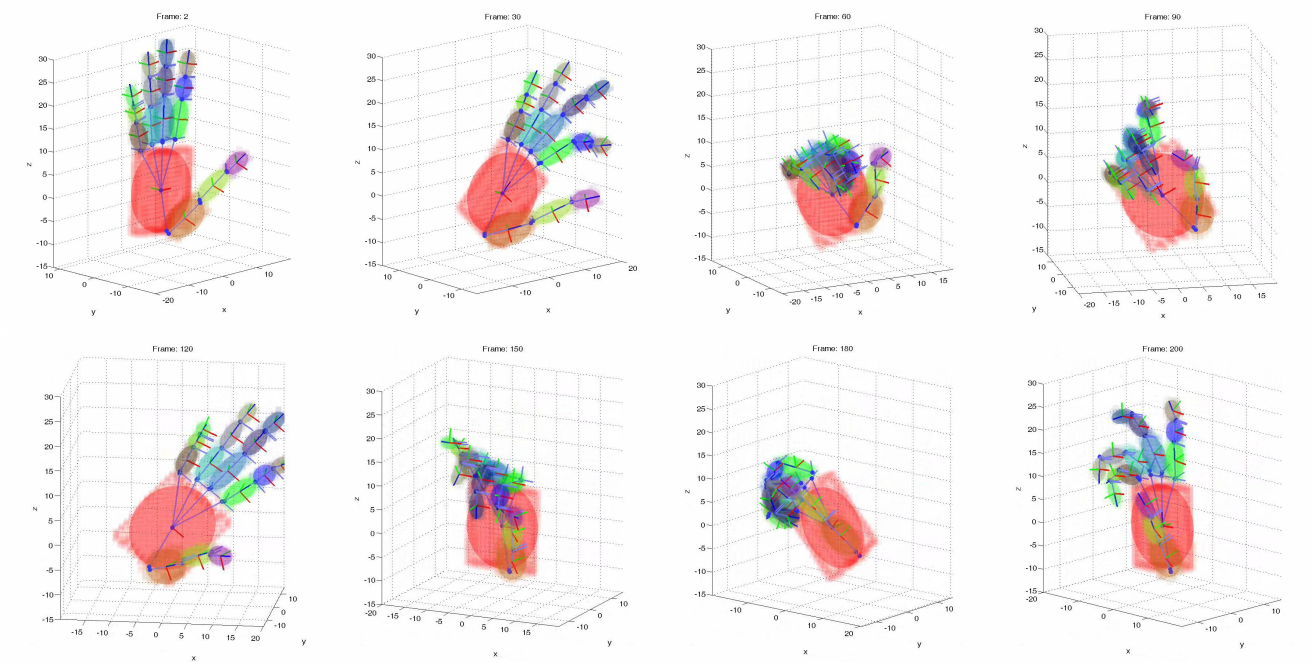

Figure 9. Hand pose learning results on synthesized hand volume reconstructions of a hand moving its fingers in a wave-like pattern while rotating at the palm.

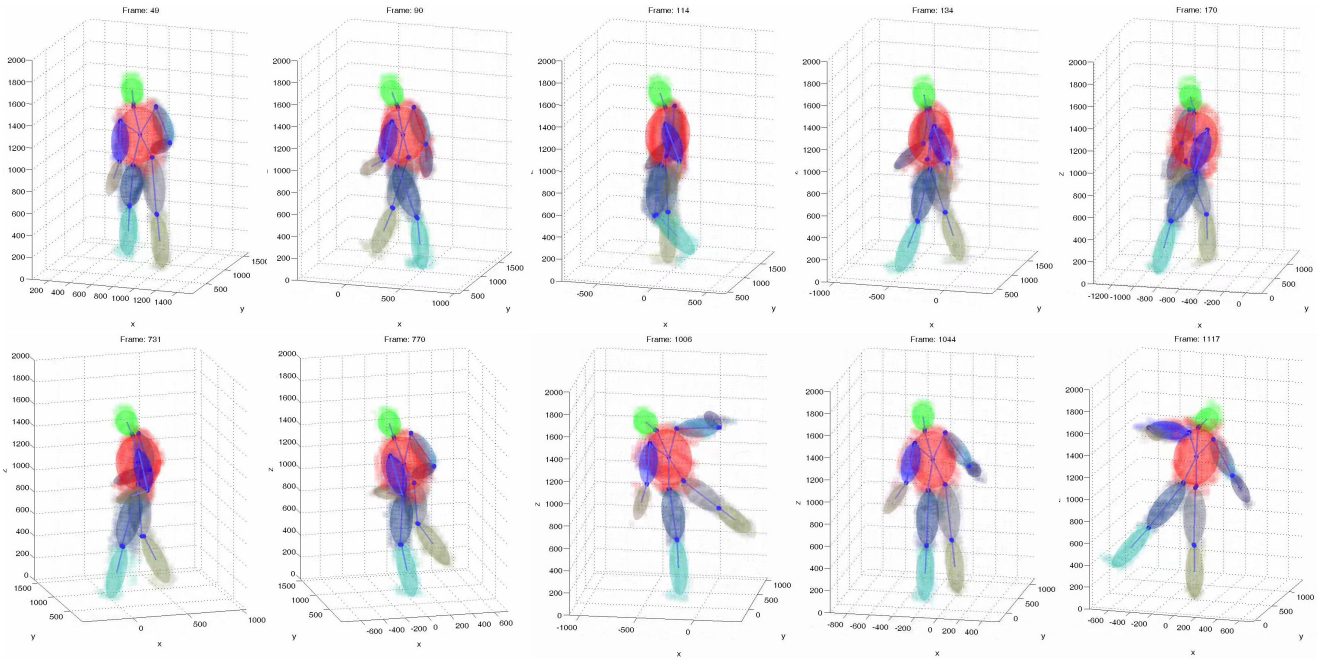

Figure 10. Body pose learning results on actual image data of a human subject walking, running and balancing. Frames 49, 90, 114, 134, 170, 731, 770, 1006, 1044, and 1117 are shown.

pages 34–42, Jan. 2

[4] G. Cheung, S. Baker, and T. Kanade. Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In *IEEE Proc. Computer Vision and Pattern Recognition Conference*, volume 1, pages 77–84. 1

[5] G. K. M. Cheung. *Visual Hull Construction, Alignment and Refinement for Human Kinematic Modeling, Motion Tracking and Rendering*. PhD thesis, Robotics Institute, Carnegie Mellon University, Oct. 2003. 3

[6] G. K. M. Cheung and T. Kanade. A real-time system for robust 3D voxel reconstruction of human motions. In *IEEE*

Table 1. Joint position error summary for kc-gmm pose learning on the HumanEva II dataset. First group follows the prescribed standard evaluation. Second group includes only successfully tracked frames.

			Rel. 3D Error			Abs. 3D Error		
		Frames	Mean	RMS	95-th	Mean	RMS	95-th
Prescribed Sub-sequences	S2 Combo_1: W	1–350	14.1	15.2	26.6	12.5	13.7	24.2
	S2 Combo_1: WR	1–700	17.9	19.7	33.6	16.0	17.8	30.7
	S2 Combo_1: WRB	1-1202	15.3	16.8	31.5	13.7	15.2	28.7
	S4 Combo_4: W	1–350	17.1	30.8	102	16.1	18.1	42.7
	S4 Combo_4: WR	1–700	21.8	29.2	36.8	21.0	22.9	34.7
	S4 Combo_4: WRB	1-1220	17.7	23.6	34.5	17.7	19.6	33.3
Successfully Tracked Sub-sequences	S2 Combo_1:W	1–160	11.0	11.1	12.3	9.2	9.3	10.3
	S2 Combo_1:W	275–350	10.8	10.8	11.8	10.2	10.3	11.2
	S2 Combo_1:R	350–550	13.5	13.8	19.0	11.8	12.0	16.5
	S2 Combo_1:B	700–1202	12.4	12.0	13.8	10.9	11.0	12.8
	S4 Combo_4:W	1–350	14.1	15.2	26.6	12.5	13.7	24.2
	S4 Combo_4:R	700–1258	12.1	12.6	19.2	13.4	13.9	18.8
	Average		**15.9**	**17.9**	**27.7**	**11.9**	**12.3**	**16.9**

Sequences: W-Walking, R-Running, B-Balancing
All units are in cm.

Proc. Computer Vision and Pattern Recognition Conference, pages 714–720, 2000. 2

[7] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1), 1977. 3

[8] A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly. A review on vision-based full DOF hand motion estimation. In *IEEE Proc. Computer Vision and Pattern Recognition Conference*, pages 75–82, 2005. 2

[9] D. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(82-98), 1999. 2

[10] E. A. Hunter, P. H. Kelly, and R. C. Jain. Estimation of articulated motion using kinematically constrained mixture densities. In *IEEE Proc. Nonrigid and Articulated Motion Workshop*, pages 10–17, 1997. 2

[11] K. N. Kutulakos and S. M. Seitz. A theory of shape by space carving. *Int. J. in Computer Vision*, 38(3):199–218, 2002. 1, 8

[12] A. Laurentini. How many 2D silhouettes does it take to reconstruct a 3D object? *Computer Vision and Image Understanding*, 67(1):81–89, July 1997. 1, 8

[13] I. Mikic, M. M. Trivedi, E. Hunter, and P. Cosman. Human body model acquisition and tracking using voxel data. *Int. J. in Computer Vision*, 53(3):199–223, 2003. 2

[14] T. B. Moeslund, A. Hilton, and V. Kruger. A survey on advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding, Special Issue on Modeling People: Vision-based understanding of a persons shape, appearance, movement and behaviour*, 104(2–3):90–126, Nov.-Dec. 2006. doi:10.1016/j.cviu.2006.08.002. 2

[15] K. Ogawara, K. Hashimoto, J. Takamatsu, and K. Ikeuchi. Grasp recognition using a 3d articulated model and infrared images. In *IEEE/RSJ Proceedings of Conference on Intelligent Robots and Systems*, volume 2, pages 27–31, 2003. 2

[16] V. I. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):677–695, Jul. 1997. 2

[17] L. Sigal and M. J. Black. HumanEva: Synchronized video and motion capture dataset for evaluation of articulated human motion. Technical Report CS-06-08, Department of Computer Science, Brown University, Providence, Rhode Island 02912, Sep. 2006. 6, 7

[18] G. Slabaugh, B. Culbertson, and T. Malzbender. A survey of methods for volumetric scene reconstruction for photographs. In *International Workshop on Volume Graphics*, pages 81–100, 2001. 1

[19] C. Theobalt, E. de Aguiar, M. Magnor, H. Theisel, and H.-P. Seidel. Marker-free kinematic skeleton estimation from sequences of volume data. In *ACM Symposium on Virtual Reality Software and Technology*, pages 57–64, 2004. 3

[20] E. Ueda, Y. Matsumoto, M. Imai, and T. Ogasawara. A hand-pose estimation for vision-based human interfaces. *IEEE Transactions on Industrial Electronics*, 50(4):676–684, August 2003. 2

[21] J. Yan and M. Pollefeys. Automatic kinematic chain building from feature trajectories of articulated objects. In *IEEE Proc. Computer Vision and Pattern Recognition Conference*, pages 712–719, 2006. 3