Unsupervised learning of skeletons from motion

David A. Ross, Daniel Tarlow, and Richard S. Zemel

University of Toronto,
10 King's College Road,
Toronto, ON, M5S 3G4,
Canada
{dross, dtarlow, zemel}@cs.toronto.edu

Abstract. Humans demonstrate a remarkable ability to parse complicated motion sequences into their constituent structures and motions. We investigate this problem, attempting to learn the structure of one or more articulated objects, given a time-series of two-dimensional feature positions. We model the observed sequence in terms of "stick figure" objects, under the assumption that the relative joint angles between sticks can change over time, but their lengths and connectivities are fixed. We formulate the problem in a single probabilistic model that includes multiple sub-components: associating the features with particular sticks, determining the proper number of sticks, and finding which sticks are physically joined. We test the algorithm on challenging datasets of 2D projections of optical human motion capture and feature trajectories from real videos.

1 Introduction

An important aspect of analyzing dynamic scenes involves segmenting the scene into separate moving objects and constructing detailed models of each object's motion. For scenes represented by trajectories of features on the objects, structure-from-motion methods are capable of grouping the features and inferring the object poses when the features belong to multiple independently-moving rigid objects. Recently, however, research has been increasingly devoted to more complicated versions of this problem, when the moving objects are articulated and non-rigid.

In this paper, we investigate this problem, attempting to learn the structure of an articulated object while simultaneously inferring its pose at each frame of the sequence, given a time-series of feature positions. We propose a single probabilistic model for describing the observed sequence in terms of one or more "stick figure" objects. We define a "stick figure" as a collection of line segments (bones or sticks) joined at their endpoints. The structure of a stick figure—the number and lengths of the component sticks, the association of each feature point with exactly one stick, and the connectivity of the sticks—is assumed to be temporally invariant, while the angles (at joints) between the sticks are allowed to change over time. We begin with no information about the figures in a sequence, as the model parameters and structure are all learned. An example of a stick figure learned by applying our model to 2D feature observations from a video of a walking giraffe is shown in Figure 1.

Learned models of skeletal structure have many possible uses. For example, detailed, manually-constructed skeletal models are often a key component in full-body

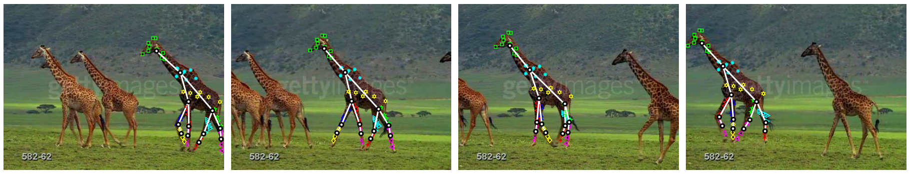

Fig. 1. Four frames from a video of a walking giraffe, augmented with a learned skeleton. Each white line represents a separate stick, the black circles are joints, and the colored markers are tracked features.

tracking algorithms. The ability to learn skeletal structure could help to automate the process, potentially producing models more flexible and accurate that those constructed manually. Additionally, skeletons are necessary for converting feature point positions into joint angles, a standard way to encode motion for animation. Furthermore, knowledge of the skeleton can be used to improve the reliability of optical motion capture, permitting disambiguation of marker correspondence and occlusion [1]. Finally, a learned skeleton might be used as a rough prior on shape to help guide image segmentation [2].

In the following section we discuss other recent approaches to modelling articulated figures from tracked feature points. In Section 3 we formulate the problem as a probabilistic model and describe the optimization of this model, which proceeds in a stage-wise fashion, building up the structure incrementally to maximize the joint probability of the model variables. In Section 5 we test the algorithm on a range of datasets. In the final section we describe assumptions and limitations of the approach, and discuss future work.

2 Related Work

The task of learning stick figures from a set of 2D feature point trajectories can be thought of as a variant of the *structure from motion* (SFM) problem. When the trajectories all arise from the motion of one rigid object, Tomasi and Kanade [3] have shown that the matrix of point locations, $\mathbf{W}$, is a linear product of a time-invariant structure matrix, $\mathbf{S}$, and a time-varying matrix of motion parameters, $\mathbf{M}$. $\mathbf{M}$ and $\mathbf{S}$ can be recovered by singular value decomposition. SFM can also be extended to handle multiple rigid objects moving independently. Costeira and Kanade [4] have shown that this problem, known as multibody SFM, can be solved by grouping the point trajectories according to the object they arise from, then solving SFM independently for each object. Grouping is accomplished by forming a shape-shape interaction or *affinity* matrix, indicating the potential for each pair of points of belonging to the same object, and using this matrix to cluster the trajectories.

Several authors have demonstrated that SFM can be interpreted as a probabilistic generative model, *e.g.* [5–7]. This view permits the inclusion of priors on the motion sequence, thereby leveraging temporal coherence. Furthermore, in the multibody case, Gruber and Weiss have presented a single probabilistic model that describes both the grouping problem and the per-object SFM problems [7]. This produces a single objective function that can be jointly optimized, leading to more robust solutions.

Unfortunately, multibody SFM cannot reliably be used to obtain the structure and motion of an articulated figure's parts since, as shown by Yan and Pollefeys [8], the motions of connected parts are linearly dependent. However, this dependence can be used to form an alternative affinity matrix for clustering the trajectories. Yan and Pollefeys use this as the basis for a stage-wise procedure for recovering articulated SFM [9]: (1) cluster point trajectories into body parts; (2) independently run SFM on each part; (3) determine connectivity between parts by running (a variant of) minimum spanning tree, where edge weights are the minimum principle angle between two parts' motion matrices (for connected, dependent parts, this should be zero); (4) finally, solve for the joint locations between connected parts. A disadvantage of this method is its lack of an overall objective function that can be optimized globally, and used to compare the quality of alternative models.

A number of authors have inferred articulated structures from three-dimensional observations, leveraging the fact that the distance between two points attached to the same rigid body part is constant, *e.g.* [10, 11]. These methods can produce detailed structures from motion capture data; however, although simple to apply in 3D, they have not been extended to 2D observations.

Others have inferred two-dimensional structures from 2D data [12–14]. Many of these methods focus on a different problem, inferring the correspondence of observations to features in each frame. With the exception of [12] (which is concerned only with the final stage of processing, after the motions of individual parts have been obtained), all of these methods build two-dimensional models directly in image coordinates. Thus, unlike SFM approaches, they are unable to deal with out-of-plane motion; a model trained on side views of a person walking would be inapplicable to a sequence of frontal views.

Learning articulated figures can also be interpreted as structure learning in probabilistic graphical models, with nodes representing the positions of parts and edges their connectivity. Learning structure is a hard problem that is usually solved approximately, using greedy methods or by restricting the class of possible structures. Song *et al.* [13] note that the optimal structure (in terms of maximum likelihood) of a graphical model is the one that minimizes the entropy of each node given its parents. Restricting their attention to graphs in which nodes each have two parents, they propose to learn the structure greedily, iteratively connecting to the graph the node with the smallest conditional entropy given its parents.

3 Model Formulation

Here we formulate a probabilistic graphical model for sequences generated from articulated skeletons. By fitting this model to a set of feature point trajectories (the observed locations of a set of features across time), we are able to parse the sequence into one or more articulated skeletons and recover the corresponding motion parameters for each frame. The observations are assumed to be 2D, whether tracked from video or projected from 3D motion capture, and the goal is to learn skeletons that capture the full 3D structure. Fitting the model is performed entirely via unsupervised learning; the only inputs are the observed trajectories, with manually-tuned parameters restricted to a small set of thresholds on Gaussian variances.

The observations for this model are the locations $\mathbf{w}_p^f$ of feature points p in frames f. A discrete latent variable $\mathbf{R}$ assigns each point to one of S sticks. Each stick s consists of a set of time-invariant 3D local coordinates $\mathbf{L}_s$, describing the relative positions of all points belonging to the stick. $\mathbf{L}_s$ is mapped to the observed world coordinate system by a different motion matrix $\mathbf{M}_s^f$ at every frame f (see Figure 2). For example, in a noiseless system, where $r_{p,1} = 1$, indicating that point p has been assigned to stick 1, $\mathbf{M}_1^f \mathbf{l}_{1,p} = \mathbf{w}_p^f$.

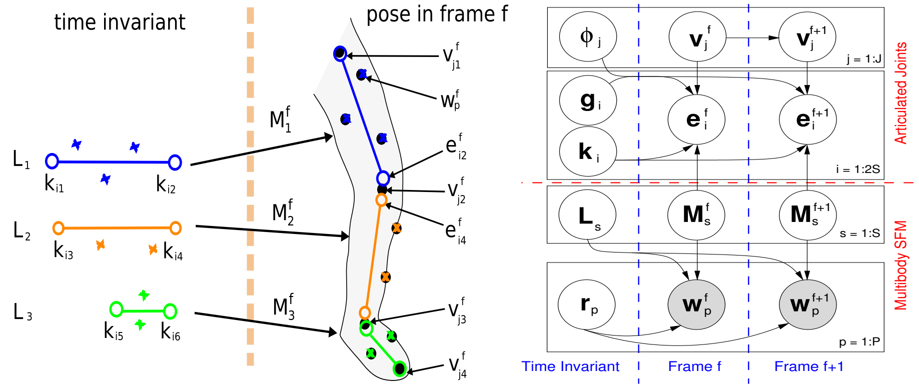

Fig. 2. (Left) The generative process for the observed feature positions, and the imputed positions of the stick endpoints. For each stick, the relative positions of its feature points and endpoints are represented in a time-invariant local coordinate system (left). For each frame in the sequence (right), motion variables attempt to fit the observed feature positions (*e.g.* $\mathbf{w}_P^f$) by mapping local coordinates to world coordinates, while maintaining structural cohesion by mapping stick endpoints to inferred vertex (joint) locations. (Right) The graphical model. The bottom half shows the model for independent multibody SFM; the top half describes the vertices and endpoints, which account for motion dependencies introduced by the articulated joints.

If all of the sticks are unconnected and move independently, then this model essentially describes multibody SFM [4, 7]. However, for an articulated structure, with connections between sticks, the stick motion variables are not independent [8]. Allowing connectivity between sticks makes the problems of describing the constraints between motions and inferring motions from the observations considerably more difficult.

To deal with this complexity, we introduce variables to model the connectivity between sticks, and the (unobserved) locations of stick endpoints and joints in each frame. Every stick has two endpoints, each of which is assigned to exactly one *vertex*. Each vertex can correspond to one or more stick endpoints (vertices assigned two or more endpoints are joints). We will let $\mathbf{k}_i$ specify the coordinates of endpoint i relative to the local coordinate system of its stick, $s(i)$, and $\mathbf{v}_j^f$ and $\mathbf{e}_i^f$ represent the world coordinate location of vertex j and endpoint i in frame f, respectively. Again, in a noiseless system, $\mathbf{e}_i^f = \mathbf{M}_{s(i)}^f \mathbf{k}_i$ for every frame f. Noting the similarity between the $\mathbf{e}_i^f$ variables and the observed feature positions $\mathbf{w}_p^f$, these endpoint locations can be interpreted as a set of pseudo-observations, inferred from the data rather than directly observed.

Vertices are used to enforce a key constraint: for all the sticks that share a given vertex, the motion matrices should map their local endpoint locations to a consistent world coordinate. This restricts the range of possible motions to only those resulting in appropriately connected figures. For example, in Figure 2(Left), endpoint 2 (of stick 1), is connected to endpoint 4 (of stick 2); both are assigned to vertex 2. Thus in every frame f both endpoints should map to the same world location, the location of the knee joint, *i.e.* $\mathbf{v}_2^f = \mathbf{e}_2^f = \mathbf{e}_4^f$.

The utility of introducing these additional variables is that, given the vertices $\mathbf{V}$ and endpoints $\mathbf{E}$, the problem of estimating the motions and local geometries ($\mathbf{M}$ and $\mathbf{L}$) factorizes into S independent structure-from-motion problems, one for each stick. Latent variable $\mathbf{g}_{i,j} = 1$ indicates that endpoint i is assigned to vertex j; hence $\mathbf{G}$ indirectly describes the connectivity between sticks. The assumed generative process for the feature observations and the vertex and endpoint pseudo-observations is shown in Figure 2(Left), and the corresponding probabilistic model in Figure 2(Right).

The complete joint probability of the model can be decomposed into a product of two likelihood terms, one for the true feature observations and the second for the endpoint pseudo-observations, and priors over the remaining variables in the model:

$$\begin{aligned}\mathbb{P} = \mathrm{P}(\mathbf{W}|\mathbf{M},\mathbf{L},\mathbf{R})\,\mathrm{P}(\mathbf{E}|\mathbf{M},\mathbf{K},\mathbf{V},\boldsymbol{\phi},\mathbf{G}) \\ \mathrm{P}(\mathbf{V})\,\mathrm{P}(\boldsymbol{\phi})\,\mathrm{P}(\mathbf{M})\,\mathrm{P}(\mathbf{L})\,\mathrm{P}(\mathbf{K})\,\mathrm{P}(\mathbf{R})\,\mathrm{P}(\mathbf{G})\end{aligned} \tag{1}$$

Assuming isotropic Gaussian noise with precision (inverse variance) τ_w, the likelihood function is

$$\mathrm{P}(\mathbf{W}|\mathbf{M},\mathbf{L},\mathbf{R}) = \prod_{f,p,s} \mathcal{N}(\mathbf{w}_p^f | \mathbf{M}_s^f \mathbf{l}_{s,p}, \tau_w^{-1}\mathbf{I})^{r_{p,s}} \tag{2}$$

where $r_{p,s}$ is a binary variable equal to 1 if and only if point p has been assigned to stick s. This distribution captures the constraint that for feature point p, its predicted world location, based on the motion matrix and its location in the local coordinate system for the stick to which it belongs ($r_{p,s} = 1$), should match its observed world location. Note that dealing with missing observations is simply a matter of removing the corresponding factors from this likelihood expression.

Each motion variable consists of a 2×3 rotation matrix $\mathbf{R}_s^f$ and a 2×1 translation vector $\mathbf{t}_s^f$: $\mathbf{M}_s^f \equiv [\mathbf{R}_s^f \quad \mathbf{t}_s^f]$. The motion prior $\mathrm{P}(\mathbf{M})$ is uniform, with the stipulation that all rotations be orthogonal: $\mathbf{R}_s^f {\mathbf{R}_s^f}^T = \mathbf{I}$.

We define the missing-data likelihood of an endpoint location as the product of two Gaussians, based on the predictions of the appropriate vertex and stick:

$$\begin{aligned}&\mathrm{P}(\mathbf{E}|\mathbf{M},\mathbf{K},\mathbf{V},\boldsymbol{\phi},\mathbf{G}) \propto \\ &\prod_{f,i} \mathcal{N}(\mathbf{e}_i^f | \mathbf{M}_{s(i)}^f \mathbf{k}_i, \tau_m^{-1}\mathbf{I}) \prod_{f,i,j} \mathcal{N}(\mathbf{e}_i^f | \mathbf{v}_j^f, \phi_j^{-1}\mathbf{I})^{g_{i,j}}\end{aligned} \tag{3}$$

Here τ_m is the precision of the isotropic Gaussian noise on the endpoint locations with respect to the stick, and $g_{i,j}$ is a binary variable equal to 1 if and only if endpoint i has been assigned to vertex j. The second Gaussian in this product captures the requirement that endpoints belonging to the same vertex should be coincident. Instead of making this a hard constraint, connectivity is softly enforced, allowing the model to accommodate a certain degree of non-rigidity in the underlying structure, as illustrated by the mismatch between endpoint and vertex positions in Figure 2(Left).

The vertex precision variables ϕ_j capture the degree of "play" in the joints, and are assigned Gamma prior distributions; the prior on the vertex locations incorporates a temporal smoothness constraint, with precision τ_t. The priors for feature and endpoint locations in the local coordinate frames, $\mathbf{L}$ and $\mathbf{K}$, are zero-mean Gaussians, with isotropic precision τ_p. Finally, the priors for the variables defining the structure of the skeleton, $\mathbf{R}$ and $\mathbf{G}$, are multinomial. Each point p selects exactly one stick s ($\sum_s r_{p,s} = 1$) with probability c_s, and each endpoint i selects one vertex j ($\sum_j g_{i,j} = 1$) with probability d_j.

$$\mathrm{P}(\boldsymbol{\phi}) = \prod_j \mathrm{Gamma}(\phi_j|\alpha_j, \beta_j) \quad \mathrm{P}(\mathbf{L}) = \prod_{s,p} \mathcal{N}(\mathbf{l}_{s,p}|0, \tau_p^{-1}\mathbf{I}) \quad \mathrm{P}(\mathbf{R}) = \prod_{p,s} (c_s)^{r_{p,s}}$$

$$\mathrm{P}(\mathbf{V}) = \prod_{f,j} \mathcal{N}(\mathbf{v}_j^f|\mathbf{v}_j^{f-1}, \tau_t^{-1}\mathbf{I}) \quad \mathrm{P}(\mathbf{K}) = \prod_i \mathcal{N}(\mathbf{k}_i|0, \tau_p^{-1}\mathbf{I}) \quad \mathrm{P}(\mathbf{G}) = \prod_{i,j} (d_j)^{g_{i,j}}$$

4 Learning

Given a set of observed feature point trajectories, we propose to fit this model in an entirely unsupervised fashion, by maximum likelihood learning. Conceptually, we divide learning into two challenges: recovering the skeletal structure of the model, and given a structure, fitting the model's remaining parameters. Structure learning involves grouping the observed trajectories into a number of rigid sticks, including determining the number of sticks, as well as determining the connectivity between them. Parameter learning involves determining the local geometries and motions of each stick, as well as imputing the locations of the stick endpoints and joints, all while respecting the connectivity constraints imposed by the structure.

Both learning tasks seek to optimize the same objective function—the expected complete log-likelihood of the data given the model—using different, albeit related, approaches. Given a structure, parameters are learned using the standard variational expectation-maximization algortihm. Structure learning is formulated as an "outer-loop" of learning: beginning with a fully disjoint multibody SFM solution, we incrementally merge stick endpoints, at each step greedily choosing the merge that maximizes the objective. Finally the expected complete log-likelihood can be used for model comparison and selection. A summary of the proposed learning algorithm is provided in Figure 4.

4.1 Learning the model parameters

Given a particular model structure, indicated by a specific setting of $\mathbf{R}$ and $\mathbf{G}$, the remaining model parameters are fit using the variational expectation-maximization (EM)

```
1. Obtain an initial grouping $\mathbf{R}$ by clustering the observed trajectories using Affinity
   Propagation. Initialize $\mathbf{G}$ to a fully-disconnected structure.
2. Optimize the parameters $\mathbf{M}$, $\mathbf{L}$, $\mathbf{K}$, $\mathbf{V}$, $\boldsymbol{\phi}$, $\mathbf{E}$, using 200 iterations of the variational
   EM updates, resampling $\mathbf{R}$ every 10 iterations.
3. Loop until no more valid merges, or maximum number of merges reached:
   (a) For all vertex-pair merges, estimate the merge cost of the proposed structure
       by updating the parameters with 20 EM iterations and noting the change in
       expected log-probability.
   (b) Choose the merge with the lowest cost, modifying $\mathbf{G}$ accordingly. Reoptimize
       all parameters using 200 EM iterations, resampling $\mathbf{R}$ every $10^{th}$ iteration.
```

Fig. 3. A summary of the learning algorithm.

algorithm. This well-known algorithm takes an iterative approach to learning: beginning with an initial setting of the parameters, each parameter is updated in turn, by choosing the value that maximizes the expected complete log-likelihood objective function, given the values (or expectations) of the other parameters.

The objective function—also known as the negative *Free Energy*—is formed by assuming a fully-factorized *variational posterior* distribution $\mathbb{Q}$ over a subset of the model parameters, then computing the expectation of the model's log probability (1) with respect to $\mathbb{Q}$, plus an entropy term:

$$\mathcal{L} = \mathrm{E}_{\mathbb{Q}}[\log \mathbb{P}] - \mathrm{E}_{\mathbb{Q}}[\log \mathbb{Q}]. \tag{4}$$

For this model, we define $\mathbb{Q}$ over the variables $\mathbf{V}$, $\mathbf{E}$, and $\boldsymbol{\phi}$, involved in the world-coordinate locations of the joints. The variational posterior for $\mathbf{v}_j^f$ is a multivariate Gaussian with mean and precision parameters $\mu(\mathbf{v}_j^f)$ and $\tau(\mathbf{v}_j^f)$; for $\mathbf{e}_i^f$ is also a Gaussian with mean $\mu(\mathbf{e}_i^f)$ and precision $\tau(\mathbf{e}_i^f)$; and for $\boldsymbol{\phi}$ is a Gamma distribution with parameters $\alpha(\phi_j)$ and $\beta(\phi_j)$:

$$\begin{aligned} \mathbb{Q} &= Q(\mathbf{V})\,Q(\mathbf{E})\,Q(\phi) & Q(\mathbf{V}) &= \prod_{f,j} \mathcal{N}(\mathbf{v}_j^f | \mu(\mathbf{v}_j^f), \tau(\mathbf{v}_j^f)^{-1}) \\ Q(\mathbf{E}) &= \prod_{f,i} \mathcal{N}(\mathbf{e}_i^f | \mu(\mathbf{e}_i^f), \tau(\mathbf{e}_i^f)^{-1}) & Q(\boldsymbol{\phi}) &= \prod_{j} \mathrm{Gamma}(\phi_j | \alpha(\phi_j), \beta(\phi_j)). \end{aligned}$$

The EM update equations are obtained by differentiating the objective function $\mathcal{L}$, with respect to each parameter, and solving for the maximum given the other parameters. We now present the parameter updates; see [15] for the derivation of $\mathcal{L}$ and the updates. The constants appearing in these equations denote the number of: observation frames F, vertices J, data points P, and sticks S; $h(f) = 1$ if $1 < f < F$ and 0 otherwise; and $s(i)$ is the index of the stick to which endpoint i belongs.

$$\tau_w^{-1} = \frac{\sum_{f,p,s} r_{p,s} \|\mathbf{w}_p^f - \mathbf{M}_s^f \mathbf{l}_{s,p}\|^2}{2FP} \quad \tau_m^{-1} = \frac{\sum_{f,i} \|\mu(\mathbf{e}_i^f) - \mathbf{M}_{s(i)}^f \mathbf{k}_i\|^2}{4FS} + \frac{\sum_{f,i} \tau(\mathbf{e}_i^f)^{-1}}{2FS}$$

$$\tau_t^{-1} = \frac{\sum_{f=2}^{F} \sum_j \|\mu(\mathbf{v}_j^f) - \mu(\mathbf{v}_j^{f-1})\|^2}{2(F-1)J} + \frac{\sum_{f,j} \tau(\mathbf{v}_j^f)^{-1}}{(F-1)J} 2^{h(f)}$$

$$\mu(\mathbf{e}_i^f) = \frac{\tau_m \mathbf{M}_{s(i)}^f \mathbf{k}_i + \sum_j g_{i,j} \frac{\alpha(\phi_j)}{\beta(\phi_j)} \mu(\mathbf{v}_j^f)}{\tau_m + \sum_j g_{i,j} \frac{\alpha(\phi_j)}{\beta(\phi_j)}} \quad \tau(\mathbf{e}_i^f) = \sum_j g_{i,j} \frac{\alpha(\phi_j)}{\beta(\phi_j)} + \tau_m$$

$$\mu(\mathbf{v}_j^f) = \frac{\frac{\alpha(\phi_j)}{\beta(\phi_j)} \sum_i g_{i,j} \mu(\mathbf{e}_i^f) + [f > 1] \tau_t \mu(\mathbf{v}_j^{f-1}) + [f < F] \tau_t \mu(\mathbf{v}_j^{f+1})}{\frac{\alpha(\phi_j)}{\beta(\phi_j)} \sum_i g_{i,j} + \tau_t 2^{h(f)}}$$

$$\tau(\mathbf{v}_j^f) = \frac{\alpha(\phi_j)}{\beta(\phi_j)} \sum_i g_{i,j} + \tau_t 2^{h(f)} \quad \alpha_j = \alpha(\phi_j) \quad \beta_j = \beta(\phi_j) \quad \alpha(\phi_j) = \alpha_j + F \sum_i g_{i,j}$$

$$\beta(\phi_j) = \beta_j + \frac{1}{2} \sum_{f,i} g_{i,j} \|\mu(\mathbf{e}_i^f) - \mu(\mathbf{v}_j^f)\|^2 + \sum_{f,i} g_{i,j} [(\tau(\mathbf{e}_i^f))^{-1} + (\tau(\mathbf{v}_j^f))^{-1}]$$

The update for the motion matrices is slightly more challenging due to the orthogonality constraint on the rotations. A straightforward approach is to separate the rotation and translation components of the motion and to solve for each individually: $\mathbf{M}_s^f = \left[\mathbf{R}_s^f \ \mathbf{t}_s^f\right]$. The update for translation is obtained simply via differentiation:

$$\mathbf{t}_{s,f} = \Big(\tau_w \sum_p r_{p,s} (\mathbf{w}_p^f - \mathbf{R}_s^f \mathbf{l}_{s,p}) + \tau_m \sum_{\{i|s(i)=s\}} (\mu(\mathbf{e}_i^f) - \mathbf{M}_s^f \mathbf{k}_{s,i})\Big) / \Big(\tau_w \sum_p r_{p,s} + 2\tau_m\Big)$$

To deal with the orthogonality constraint on $\mathbf{R}_s^f$, its update can be posed as an *orthogonal Procrustes problem* [16, 17]. Given matrices $\mathbf{A}$ and $\mathbf{B}$, the goal of orthogonal Procrustes is to obtain the matrix $\mathbf{R}$ that minimizes $\|\mathbf{A} - \mathbf{RB}\|^2$, subject to the constraint that the rows of $\mathbf{R}$ form an orthonormal basis. Computing the most likely rotation involves maximizing the likelihood of the observations (2) and of the endpoints (3), which can be written as the minimization of $\sum_p \|(\mathbf{w}_p^f - \mathbf{t}_{s,f}) - \mathbf{R}_s^f \mathbf{l}_{s,p}\|^2$ and $\sum_{\{i|s(i)=s\}} \|(\mu(\mathbf{e}_i^f) - \mathbf{t}_{s,f}) - \mathbf{R}_s^f \mathbf{k}_{s,i}\|^2$ respectively. Concatenating the two problems together, weighted by their respective precisions, allows the update of $\mathbf{R}_s^f$ to be written as a single orthogonal Procrustes problem: $\mathrm{argmin}_{\mathbf{R}_s^f} \|\mathbf{A} - \mathbf{R}_s^f \mathbf{B}\|^2$, where

$$\mathbf{A} = \Big[\left[\sqrt{\tau_w}\, r_{p,s} (\mathbf{w}_p^f - \mathbf{t}_{s,f})\right]_{p=1..P} \left[\sqrt{\tau_m}\, (\mu(\mathbf{e}_i^f) - \mathbf{t}_{s,f})\right]_{\{i|s(i)=s\}}\Big]$$

$$\mathbf{B} = \Big[\left[\sqrt{\tau_w}\, r_{p,s} \mathbf{l}_{s,p}\right]_{p=1..P} \left[\sqrt{\tau_m}\, \mathbf{k}_i\right]_{\{i|s(i)=s\}}\Big] .$$

The solution is to compute the singular value decomposition of $\mathbf{BA}^T \overset{SVD}{=} \mathbf{U\Sigma V}^T$, and let $\mathbf{R} = \mathbf{V}\, \mathbf{I}_{m \times n} \mathbf{U}^T$, where m and n are the numbers of rows in $\mathbf{A}$ and $\mathbf{B}$ respectively.

Given $\mathbf{R}_s^f$ and $\mathbf{t}_s^f$, the updates for the local coordinates are:

$$\mathbf{l}_{s,p} = \big(\sum_f [\mathbf{R}_s^f]^T \mathbf{R}_s^f + \frac{\tau_p}{\tau_w}\mathbf{I}\big)^{-1} \sum_f [\mathbf{R}_s^f]^T (\mathbf{w}_p^f - \mathbf{t}_{s,f})$$

$$\mathbf{k}_i = \big(\sum_f [\mathbf{R}_{s(i)}^f]^T \mathbf{R}_{s(i)}^f + \frac{\tau_p}{\tau_m}\mathbf{I}\big)^{-1} \sum_f [\mathbf{R}_{s(i)}^f]^T (\mu(\mathbf{e}_i^f) - \mathbf{t}_{s(i)}^f)$$

The final issue to address for EM learning is initialization. Many ways to initialize the parameters are possible; here we settle on one simple method that produces satisfactory results. The motions and local coordinates, $\mathbf{M}$ and $\mathbf{L}$, are initialized by solving SFM independently for each stick [3]. The vertex locations are initialized by averaging the observations of all sticks participating in the joint: $\mu(\mathbf{v}_j^f) = (\sum_{i,p} g_{i,j}\, r_{p,s(i)}\, \mathbf{w}_p^f)/(\sum_{i,p} g_{i,j}\, r_{p,s(i)})$. The endpoints are initially coincident with their corresponding vertices, $\mu(\mathbf{e}_i^f) = \sum_j g_{i,j}\, \mu(\mathbf{v}_j^f)$, and each $\mathbf{k}_i$ initialized by averaging the backprojected endpoint locations: $\mathbf{k}_i = \frac{1}{F}\sum_f [\mathbf{R}_{s(i)}^f]^T(\mu(\mathbf{e}_i^f) - \mathbf{t}_{s(i)}^f)$. All precision parameters are initialized to constant values, as discussed in [15].

4.2 Learning the skeletal structure

Structure learning in this model entails estimating the assignments of feature points to sticks (including the number of sticks), and the connectivity of sticks, expressed via the assignments of stick endpoints to vertices. The space of possible structures is enormous. We therefore adopt an incremental approach to structure learning: beginning with a fully-disconnected multibody-SFM model, we greedily add joints between sticks by merging vertices. Each merge forms a new model, and its parameters are updated via EM and the assignments of observations to sticks are resampled. At any step, the optimal model can be determined by simply comparing the expected complete log-likelihood of each model.

The first step in structure learning involves hypothesizing an assignment of each observed feature trajectory to a stick. This is accomplished by clustering the trajectories using the *Affinity Propagation* algorithm [18]. Affinity Propagation takes as input an affinity matrix, for which we supply the affinity measure from [8, 9] as presented in Section 2. During EM parameter learning, the stick assignments $\mathbf{R}$ are resampled every 10 iterations using the posterior probability distribution $\mathrm{P}(r_{p,s}) \propto c_s \exp(-\frac{\alpha_w}{2}\sum_f \|\mathbf{w}_p^f - \mathbf{M}_s^f \mathbf{l}_{s,p}\|^2)\quad \text{s.t.} \sum_{s'} r_{p,s'} = 1$. Instead of relying only on information available before model fitting begins, *c.f.* [4, 11, 9]), resampling of stick assignments allows model probability to be improved by leveraging current best estimates of the model parameters. This is a key advantage of our approach, employing a single model for the entire process.

The second step of structure learning involves determining which sticks' endpoints are joined together. As discussed earlier, connectivity is captured by assigning stick endpoints to vertices; each endpoint must be associated to one vertex, and vertices with two or more endpoints act as articulated joints. (Valid configurations include only cases in which endpoints of a given stick are assigned to different vertices.) We employ an incremental greedy scheme for inferring this graphical structure $\mathbf{G}$, beginning from an initial structure that contains no joints between sticks. Thus, in terms of the model, we start with $J = 2S$ vertices, one per stick-endpoint, so $g_{i,j} = 1$ if and only if $j = i$. Given this initial structure, parameters are fit using variational EM.

A joint between sticks is introduced by merging together a pair of vertices. The choice of vertices to merge is guided by our objective function $\mathcal{L}$. At each stage of merging we consider all valid pairs of vertices, putatively joining them and estimating (via 20 iterations of EM) the change in log-likelihood if this merge were accepted. The merge with the highest log-likelihood is performed, by modifying $\mathbf{G}$ accordingly, and the model parameters are re-optimized with 200 additional iterations of EM, including resampling of the stick assignments $\mathbf{R}$. This process is repeated until no valid merges remain, or the desired maximum number of merges has been reached.

As can be seen from the EM updates, each iteration of parameter learning scales linearly with F, J, P, and S. At each stage of structure learning, determining the locally-optimal merge scales with $O(J^2)$.

5 Experimental Results and Analysis

We now present experimental results on three feature point trajectory datasets—videos of an excavator and a walking giraffe, and 2D projections of human motion capture—as well as a brief comparison with a related method [9]. Further results are included in [15]. In each experiment a model was learned on the first 70% of the sequence frames, with the remaining 30% held out as a test set used to measure the model's performance. Learning was performed using the algorithm summarized in Figure 4, with greedy merging continuing (generally) until no valid merges remained. After each stage of merging, we saved the learned model and corresponding expected complete log-likelihood—the objective function learning maximizes. The likelihoods were plotted for comparison and used to select the optimal model.

The learned model's performance was evaluated based on its ability to impute (reconstruct) the locations of missing observations. For each test sequence we generated a set of missing observations by simulating an occluder that sweeps across the scene, obscuring points as it passes. We augmented this set with an additional 5% of the observations chosen to be "missing at random", to simulate drop-outs and measurement errors, resulting in a overall occlusion rate of 10-15%. The learned model was fit to the un-occluded points of the test sequence, and used to predict the location of the missing points. Performance was measured by computing the root-mean-squared error between the predictions and the locations of the heldout points. We compared the performance of our model against similar prediction errors made by single-body and multibody structure from motion models.

Our first dataset consisted of a brief video clip of an excavator. We used a KLT tracker [19] with manual clean-up to obtain 35 feature trajectories across 176 frames. Our algorithm processed the data in 4 minutes on a 2.8 gHz processor. The learned model at each stage of greedy merging is depicted in Figure 4 (Top). The optimal structure was chosen by comparing the log-likelihood at each state, as plotted in Figure 4 (Bottom,left). Using the excavator data, we also examined the model's robustness to learning with occlusions in the training data. The algorithm was able to correctly recover the structure using the occlusion scheme described above, as well as when up to 75% of the training observations were randomly withheld during training. Figure 4 (Bottom,right) shows that the system's predictions for occluded data was significantly better than either multibody or single-body SFM.

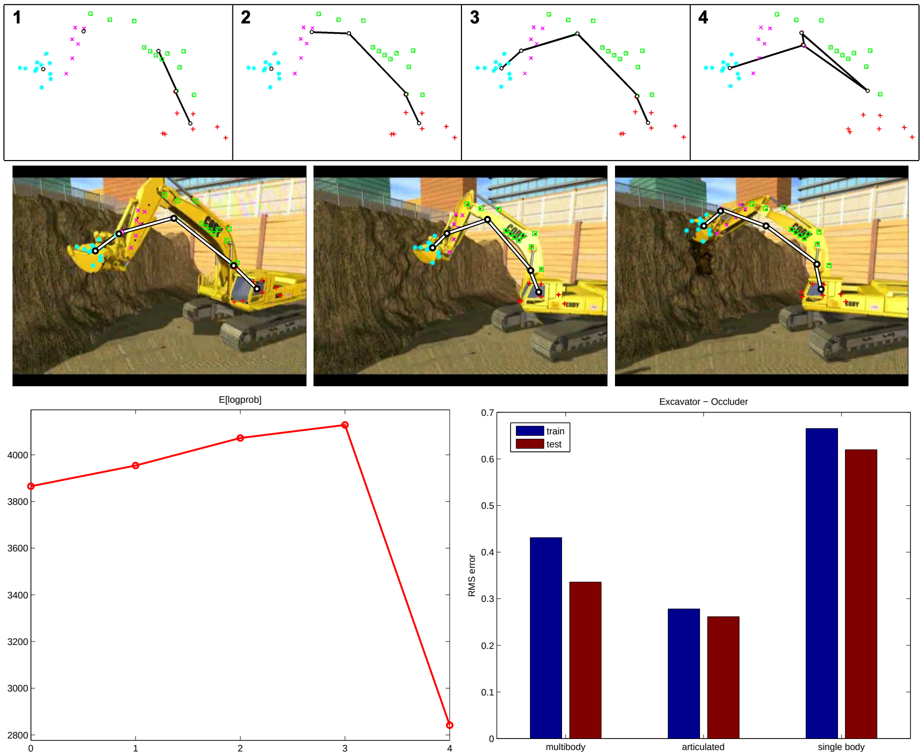

Fig. 4. Top: Learned structures during greedy merging from the Excavator dataset. Middle: superposition of the structure onto video frames. Bottom: Log-probability scores after each stage of endpoint merging, and prediction errors of occluded feature data for multibody SFM, our articulated model, and single-body SFM.

Our second dataset consisted of a video of a walking giraffe. As before, 60 features were tracked in 128 frames. Merging results are depicted in Figure 5. Using the objective function to guide model selection, the best structure corresponded to state 10, and this model is shown superimposed over the original video in Figure 1.

Our third dataset consisted of optical human motion capture data (courtesy of the Biomotion Lab, Queen's University, Canada), which we projected from 3D to 2D using an isometric projection. The data contained 53 features, tracked across a 1018-frame range-of-motion exercise (training data), and 318 frames of running on an inclined

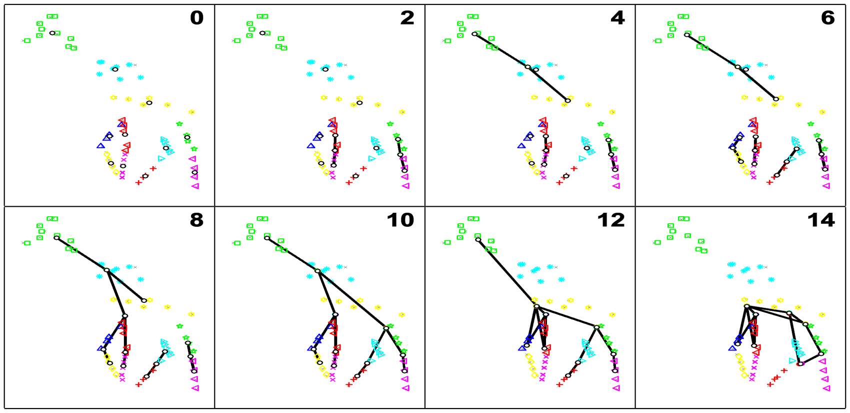

Fig. 5. Learned structures during greedy merging from the Giraffe dataset.

plane (test data). Again the objective function peaks at what is intuitively the best-looking structure (stage 11).

For comparison, we ran a re-implementation of the algorithm of Yan et al [9] on the Giraffe and 2D-Human datasets. (We note that these results are sensitive to parameter settings that are used to estimate the effective rank of motions; we manually explored a small range of parameter settings and chose the skeleton that was most visually appealing.) The criteria used by Yan and Pollefeys to determine stick connectivity relies on the dependencies between motions. Though two sticks sharing a joint will have intersecting motion subspaces and this method will correctly find these instances, there are other situations where it will choose to join two sticks that have dependent motions but that are not actually connected parts. This is clearly shown in the Giraffe result in Figure 7(Left), where front and back legs that move in phase are found to be connected. In this case, the more natural representation of our algorithm, where we are hypothesizing a joint location and seeing how well it fits the data, proves beneficial. In the 2D-Human result in Fig. 7(Right), we can see that the effects of these incorrect dependencies are not restricted to be local when the structure learning is based upon a spanning tree. In this case, the spanning tree algorithm chooses to join the two feet together because there is a strong dependence in their motions for this dataset. This decision then causes another error, where the shoulder is connected to the thigh, because connecting each to the torso would no longer produce a tree given the connection between the feet.

6 Discussion

We have demonstrated a single coherent model that can describe the structures and motion of articulated skeletons. This model can be applied to a variety of structures, requiring no input beyond the observed feature trajectories and a minimum of manually-adjusted parameters. The method extends the state-of-the-art in a number of ways. It iterates between updates of the structure and the parameters, allowing information ob-

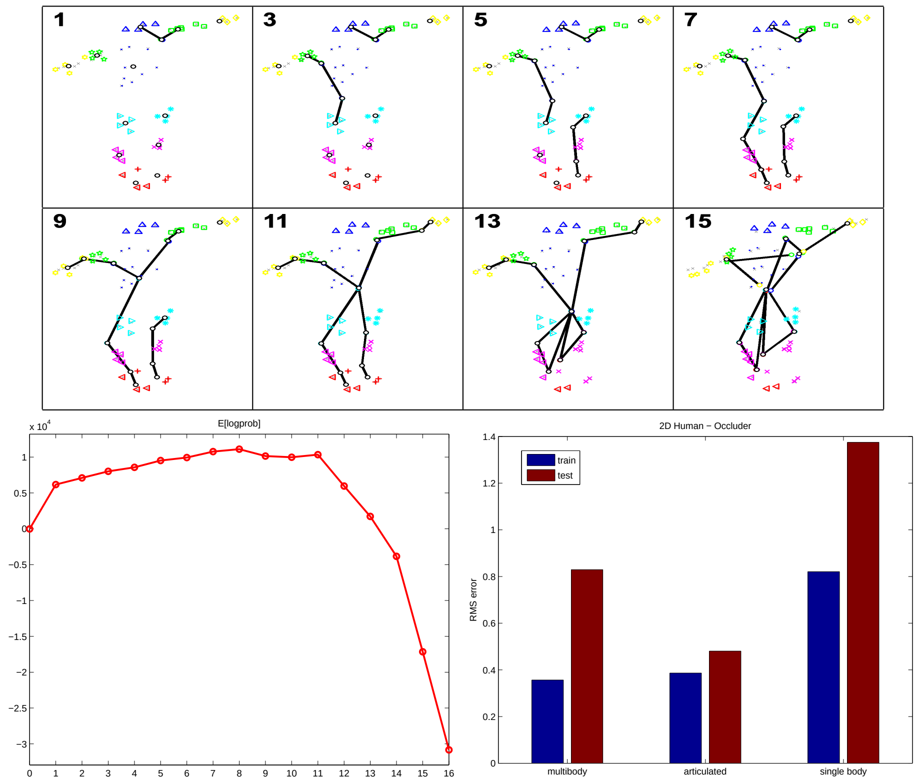

Fig. 6. Top: Learned structures during greedy merging from the 2D-Human dataset. Bottom: Log-probability scores after each stage of endpoint merging, and prediction errors of occluded feature data for multibody SFM, our articulated model, and single-body SFM.

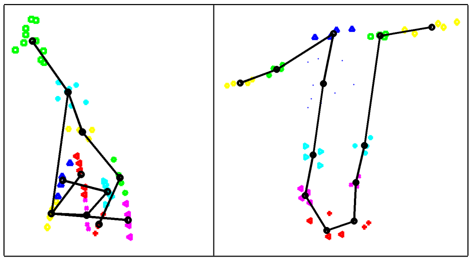

Fig. 7. Optimal structures found by the algorithm of Yan et al [9] on Giraffe (Left) and 2D-Human (Right) data.

tained from one stage to assist learning in the other. It is not limited to a single structure (additional results on feature trajectories of two separate objects were omitted due to space restrictions). Also, the noise in our generative model allows a degree of non-rigidity in the motion with respect to the learned skeleton. This not only allows a feature point to move in relation to its associated stick, but also permits complexity in the joints, as the stick endpoints joined at a vertex need not coincide exactly.

To obtain good results, our model requires a certain density of features, in particular because the 2D affinity matrix [8] requires at least 4 points per stick. The flexibility of learned models are limited to the degrees of freedom visible in the training data; if a joint is not exercised, then the body parts it connects cannot be distinguished. Finally, our model requires that the observations arise from a scene containing roughly-articulated figures; it would be a poor model of an octopus, for example.

An important extension of the current work is to apply the learned skeleton to feature trajectories from new instances of the same type of articulated structure, allowing for recognition and tracking of a novel moving object.

Acknowledgements

We thank Niko Troje, at the Biomotion Lab, Queen's University, for generously sharing his human motion capture data, and for useful discussions.

References

1. Herda, L., Fua, P., Plankers, R., Boulic, R., Thalmann, D.: Using skeleton-based tracking to increase the reliability of optical motion capture. Human Movement Science Journal **20**(3) (2001) 313–341
2. Bray, M., Kohli, P., Torr, P.: Posecut: Simultaneous segmentation and 3d pose estimation of humans using dynamic graph-cuts. In: Proceedings of the Tenth European Conference on Computer Vision. (2006) 642–655
3. Tomasi, C., Kanade, T.: Shape and motion from image streams under orthography: a factorization method. International Journal of Computer Vision **9** (1992) 137–154
4. Costeira, J.P., Kanade, T.: A multibody factorization method for independently moving-objects. International Journal of Computer Vision **29**(3) (September 1998) 159–179
5. Dellaert, F., Seitz, S.M., Thorpe, C.E., Thrun, S.: EM, MCMC, and chain flipping for structure from motion with unknown correspondence. Machine Learning **50**(1-2) (2003) 45–71
6. Torresani, L., Hertzmann, A., Bregler, C.: Learning non-rigid 3d shape from 2d motion. In: Advances in Neural Information Processing Systems (NIPS). (2003)
7. Gruber, A., Weiss, Y.: Multibody factorization with uncertainty and missing data using the EM algorithm. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2004)
8. Yan, J., Pollefeys, M.: A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. In: 9th European Conference on Computer Vision. (2006)
9. Yan, J., Pollefeys, M.: Automatic kinematic chain building from feature trajectories of articulated objects. In: Conference on Computer Vision and Pattern Recognition (CVPR). (2006)
10. Liebowitz, D., Carlsson, S.: Uncalibrated motion capture exploiting articulated structure constraints. In: International Conference on Computer Vision (ICCV). (2001)

11. Kirk, A.G., O'Brien, J.F., Forsyth, D.A.: Skeletal parameter estimation from optical motion capture data. In: Conference on Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society (2005)
12. Taycher, L., III, J.W.F., Darrell, T.: Recovering articulated model topology from observed rigid motion. In Becker, S., Thrun, S., Obermayer, K., eds.: Advances in Neural Information Processing Systems (NIPS), MIT Press (2002) 1311–1318
13. Song, Y., Goncalves, L., Perona, P.: Unsupervised learning of human motion. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(7) (July 2003) 814–827
14. Ramanan, D., Forsyth, D.A., Barnard, K.: Building models of animals from video. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(8) (2006) 1319–1334
15. Ross, D.A.: Learning Probabilistic Models for Visual Motion. PhD thesis, University of Toronto, Ontario, Canada (2008)
16. Golub, G.H., Van Loan, C.F.: Matrix Computations. The Johns Hopkins University Press (1996)
17. Viklands, T.: Algorithms for the Weighted Orthogonal Procrustes Problem and Other Least Squares Problems. PhD thesis, Ume University, Ume, Sweden (2006)
18. Frey, B., Dueck, D.: Clustering by passing messages between data points. Science **315** (February 2007) 972–976
19. Shi, J., Tomasi, C.: Good features to track. In: Conference on Computer Vision and Pattern Recognition (CVPR). (1994) 593–600