Unsupervised Hierarchical Modeling of Locomotion Styles

Wei Pan PWAY@CS.DARTMOUTH.EDU
Lorenzo Torresani LORENZO@CS.DARTMOUTH.EDU
Computer Science Department, Dartmouth College, 6211 Sudikoff Lab, Hanover, NH 03755 USA

Abstract

This paper describes an unsupervised learning technique for modeling human locomotion styles, such as distinct related activities (e.g. running and striding) or variations of the same motion performed by different subjects. Modeling motion styles requires identifying the common structure in the motions and detecting style-specific characteristics. We propose an algorithm that learns a hierarchical model of styles from unlabeled motion capture data by exploiting the cyclic property of human locomotion. We assume that sequences with the same style contain locomotion cycles generated by noisy, temporally warped versions of a single latent cycle. We model these style-specific latent cycles as random variables drawn from a common "parent" cycle distribution, representing the structure shared by all motions. Given these hierarchical priors, the algorithm learns, in a completely unsupervised fashion, temporally aligned latent cycle distributions, each modeling a specific locomotion style, and computes for each example the style label posterior distribution, the segmentation into cycles, and the temporal warping with respect to the latent cycles. We demonstrate the flexibility of the model on several application problems such as style clustering, animation, style blending, and filling in of missing data.

1. Introduction

Modeling human locomotion[1] is of fundamental importance for a wide range of applications including gait recognition, diagnosis of movement disorders, analysis of running efficiency, human tracking in video sequences, and computer animation. In this paper we focus on the specific problem of deriving computational models capable of capturing and representing distinct locomotion styles, corresponding for example to distinct related activities (e.g. walking and running) or variations of the same activity performed by different subjects. Hand-constructing style models is generally not possible due to the subtlety of the style variations and the complexity of the human dynamics. Because of such challenges, several researchers have proposed automatically learning motion style models from human motion examples. Most previously proposed approaches treat motion style modeling as a generic data fitting problem by employing general-purpose learning models. Examples of such models include Restricted Boltzmann Machines (Taylor et al., 2007), Gaussian Processes (Wang et al., 2007), linear dynamical systems (Brand & Hertzmann, 2000; Li et al., 2002; Chiappa et al., 2009), and nonlinear manifolds (Elgammal & Lee, 2004). General-purpose models fail to exploit relevant prior information, such as the cyclic property of locomotion or the knowledge that different styles of an activity must correspond to subtle variations of a common motion. Recent work (Liu et al., 2005; Urtasun et al., 2008) has shown that incorporating domain-specific prior information in the model yields motion representations that are more accurate and intuitive, and helps reducing the risk of overfitting. This is particularly important in the motion domain, where the data is high-dimensional and training examples are scarce. There are many possible ways to encode prior knowledge in the model. Liu et al. (2005) use body physics constraints. Urtasun et al. (2008) force the learned model to satisfy a specific topological structure. In this paper we propose to encode domain knowledge via hierarchical priors and probability distributions specifically suited to the properties of human locomotion. We employ hierarchical priors to encode the knowledge that distinct locomotion styles must share a common structure. We view each motion style as a random variable drawn from an unknown distribution common to all styles. This common distribution assumption constrains the styles to represent subtle variations around an average motion. Furthermore, we exploit the cyclic nature of the data and learn mod-

[1] In this paper the term locomotion is used in a wide sense to indicate any cyclic limb motion, such as walking, running, crawling, swimming.

Appearing in *Proceedings of the 26^{th} International Conference on Machine Learning*, Montreal, Canada, 2009.

els representing single cycles of locomotion instead of describing entire periodic sequences. By doing so we make more efficient use of the available data, reduce the number of unknowns in the model, and obtain models that are easy to interpret. We achieve this by inferring cyclic alignment distributions which temporally synchronize the observations and describe each sequence as a concatenation of motions generated by single-cycle models. A distinctive feature of our approach is that the learning is completely unsupervised: while the methods in (Liu et al., 2005; Urtasun et al., 2008) require training examples of a single style or data with user-provided style labels, our algorithm can automatically learn distinct style models from a large pool of unlabeled motion sequences.

Our model learns for each style a prototypical, high-resolution fusion of the cycles belonging to sequences assigned to that style (to be more precise, each learned style is a PDF describing also how cycle samples can deviate from the prototypical style cycle). We demonstrate that these cycle prototypes can be used to generate vivid animations, visually undistinguishable from real motion. Furthermore, since our algorithm yields motion style prototypes that are time-synchronized, the output of our system is directly usable by motion style blending algorithms (Rose et al., 1998; Kovar et al., 2002) to generate realistic novel motion. We demonstrate this application in our experiments.

Our work builds on the Hierarchical Bayesian Continuous Profile Model (HB-CPM) proposed by Listgarten et al. (2007) for detection of differences in time series classes. We generalize this method to the fully unsupervised case, where class labels are not available as input. Thus, our algorithm simultaneously performs clustering, difference detection, and alignment of time-series. HB-CPM was originally proposed to model time-series with class-specific differences corresponding to impulses at rare but systematic locations. However, style differences in human motion are typically extended in time (see Figure 1) and thus an impulse-based model is not appropriate for our purposes. We describe a different set of hierarchical priors, specifically suited to the case of human locomotion. Inference in (Listgarten et al., 2007) is handled by means of a Markov Chain Monte Carlo (MCMC) method. However, in our case a stochastic approach is not computationally practical due to the high-dimensionality of the data. Instead we propose an efficient variational method, which is able to learn motion style models in just a couple of minutes from datasets including up to 39 distinct 62-dimensional time series (compare this performance to the several hours needed by the MCMC algorithm in (Listgarten et al., 2007) to process 21 one-dimensional time series).

2. A Hierarchical Locomotion Model

Our approach to style modeling is inspired by the observation that motions with the same style are characterized by similar body pose sequences albeit with possibly different timing (e.g. walking performed by the same individual at different paces). On the other hand, motions with distinct styles contain significant differences in body poses under any temporal warping. Therefore, we propose a model which groups within the same cluster motions that can be temporally aligned to have similar sequences of body poses. Note that in our approach the clustering, the temporal alignment, and the learning of the underlying pose trajectories are performed simultaneously.

We now describe formally our Hierarchical Locomotion Model (HLM). Let $\mathbf{X}^k = (\mathbf{x}_1^k, \mathbf{x}_2^k, ..., \mathbf{x}_{N_k}^k)$ denote the k-th sequence in a dataset of K locomotion examples sharing a common structure but containing some stylistic variations. N_k indicates the length of the sequence and $\mathbf{x}_i^k$ is a F-dimensional vector encoding the 3D configuration of the body at time i, for example in the form of kinematic joint angles. We assume that the dataset comprises C distinct locomotion styles. We indicate with $l^k \in \{1, ..., C\}$ the unknown style label of sequence k, which we assume to be drawn from a Multinomial distribution with parameters $\tau = \{\tau_1, ..., \tau_C\}$. We model each style c by means of a hidden variable $\mathbf{Z}^c = (\mathbf{z}_1^c, ..., \mathbf{z}_M^c)$, which we will refer to as the latent cycle of style c. M denotes the length of the latent cycle and $\mathbf{z}_m^c$ is an F-dimensional vector encoding the 3D body configuration at time frame m in the cycle. We assume that the cycles of an observed sequence are generated from temporally subsampled versions of a latent cycle. Consequently, we would like M to be much larger than the typical length of a cycle in the observed motions, as this would yield higher-resolution representations of the motions. However, care must be taken to avoid overfitting. Inspired by the choice of this parameter in (Listgarten et al., 2007), we select $M = 2N$ where N is a value provided as input to the system and representing the expected length of a cycle in the training set (note that our system can handle cycle lengths in the observed motions differing considerably from N). We assume that a sequence $\mathbf{X}^k$ with motion style c consists of a concatenation of cycles generated by an HMM which moves cyclically and in left-to-right order through time samples of latent cycle $\mathbf{Z}^c$, and emits noise-corrupted versions of 3D configurations $\mathbf{z}_m^c$. In other words, we assume $\mathbf{x}_i^k \sim \mathcal{N}(\mathbf{z}_{\pi_i^k}^{l^k}, {\Lambda^{l_k}}^{-1})$, where $\pi_i^k \in \{1, ..., M\}$ indicates the HMM state and Λ^{l_k} is a diagonal, style-specific, covariance matrix. We denote with $p(\pi_i^k | \pi_{i-1}^k; \mathbf{d}^k)$ the cyclic, left-to-right transition distribution governing the HMM of sequence k, implemented as follows:

$$p(\pi_i^k = m|\pi_{i-1}^k = n; \mathbf{d}^k) = \begin{cases} d_1^k & \text{if } m-n=1 \text{ or } M+m-n=1 \\ \dots & \\ d_{J_\pi}^k & \text{if } m-n=J_\pi \text{ or } M+m-n=J_\pi \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where $1 \leq m, n \leq M$, and J_π is the maximum transition length expressed in number of frames. The d_j^k denote probabilities satisfying the condition $\sum_{j=1}^{J_\pi} d_j^k = 1$. Note that since we use $J_\pi << \frac{M}{2}$, the HMM valid transitions are only either left-to-right or from the tail section to the head section of the latent trace (corresponding to moves from state n to state m such that $(M+m)-n=j$, with $j \leq J_\pi$). This latter type of transition is used to model the periodic property of locomotion. Finally, we force the latent cycles to be aligned to one another, and to share a common structure by assuming that each style-specific $\mathbf{Z}^c$ is a random variable drawn from a distribution encouraging the latent cycle to be temporally smooth and "similar" to a parent cycle $\bar{\mathbf{Z}} = (\bar{\mathbf{z}}_1, ..., \bar{\mathbf{z}}_M)$ common to all styles. In summary, we assume the following generative process for a dataset $\mathbf{X} = \{\mathbf{X}^1, ..., \mathbf{X}^K\}$:

1. $\bar{\mathbf{Z}} \sim \mathcal{N}(\bar{\mathbf{z}}_1; \bar{\mathbf{z}}_M, \eta^{\bar{z}}\mathbf{I}) \prod_{m=2}^{M} \mathcal{N}(\bar{\mathbf{z}}_m; \bar{\mathbf{z}}_{m-1}, \eta^{\bar{z}}\mathbf{I})$
2. For each style $c \in \{1, ..., C\}$:
$$\mathbf{Z}^c \sim \mathcal{N}(\mathbf{z}_1^c; \mathbf{z}_M^c, \lambda_s^{-1}\mathbf{I}) \prod_{m=2}^{M} \mathcal{N}(\mathbf{z}_m^c; \mathbf{z}_{m-1}^c, \lambda_s^{-1}\mathbf{I}) \times \prod_{m=1}^{M} \mathcal{N}(\mathbf{z}_m^c; \bar{\mathbf{z}}_m, \lambda_{\bar{z}}^{-1}\mathbf{I})$$
3. $\tau \sim \mathcal{D}(\eta^\tau)$
4. For each sequence $k \in \{1, ..., K\}$:
   (a) $l^k \sim \text{Mult}(\tau)$
   (b) $\mathbf{d}^k \sim \mathcal{D}(\eta^d)$
   (c) $\pi^k \sim \text{Mult}(\pi_1^k; \mathbf{1}/N_k) \prod_{i=2}^{N_k} p(\pi_i^k|\pi_{i-1}^k; \mathbf{d}^k)$
   (d) $\mathbf{X}^k \sim \prod_{i=1}^{N_k} \mathcal{N}(\mathbf{x}_i^k; \mathbf{z}_{\pi_i^k}^{l^k}, {\Lambda^{l_k}}^{-1})$

where $\mathcal{D}()$ indicates a Dirichlet distribution, and Mult() a Multinomial distribution. The improper prior on $\bar{\mathbf{Z}}$ (defined via hyperparameter $\eta^{\bar{z}}$) is used to enforce temporal smoothness of the parent cycle. Note that this encourages also the last frame in the cycle to be similar to the first. The PDF of the latent cycle $\mathbf{Z}^c$ is the product between a smoothing distribution correlating configurations of consecutive frames and a Gaussian distribution which encourages each latent cycle frame $\mathbf{z}_m^c$ to be close to the corresponding parent cycle frame $\bar{\mathbf{z}}_m$. The resulting PDF is a multivariate Gaussian (Listgarten et al., 2007). λ_s and $\lambda_{\bar{z}}$ are precision parameters controlling the temporal smoothing and the distribution relating the latent cycle to the parent cycle, respectively. We assume uninformative priors for parameters $\lambda_{\bar{z}}$, λ_s and λ_f^c, where $\Lambda^c = \text{diag}([\lambda_1^c, ..., \lambda_F^c])$. We regularize parameters τ and $\mathbf{d}^k$ via hyperparameters η^τ and η^d.

Exact inference in our model is analytically intractable, and thus approximation methods need to be employed. Listgarten et al. (Listgarten et al., 2007) applied stochastic approximation (Markov Chain Monte Carlo) to learn a fully-Bayesian Hierarchical Continuous Profile Model in the simpler *supervised* setting (i.e. when class labels are provided), and for the case of one-dimensional time-series. However, a stochastic approach in our case is not computationally practical due to the high-dimensionality of our data[2] and the more complex setting deriving from the use of unlabeled data. We make the problem tractable by modeling some of the unknowns as parameters and by adopting a variational approach to estimate the distributions of the other unobservables. Specifically, we model the HMM states $\{\pi_i^k\}$, the style labels $\{l^k\}$, and the latent style cycles $\{\mathbf{Z}^c\}$ as hidden variables, which are fully marginalized out during learning. All remaining unobservable $\theta \equiv \{\bar{\mathbf{Z}}, \lambda_{\bar{z}}, \lambda_s, \Lambda^1, ..., \Lambda^C, \tau, \mathbf{d}^1, ..., \mathbf{d}^K\}$ are treated as parameters estimated using a penalized maximum likelihood framework, with penalties defined via fixed hyperparameters $\eta \equiv \{\eta^{\bar{z}}, \eta^\tau, \eta^d\}$. Thus we solve for θ to maximize

$$\begin{aligned} & p(\theta|\eta)p(\mathbf{X}|\theta) = p(\theta|\eta) \prod_{k=1}^{K} p(\mathbf{X}^k|\theta) \\ & = p(\theta|\eta) \\ & \times \prod_{k=1}^{K} \int \sum_{l^k=1}^{C} \sum_{\pi^k \in \Pi^k} p(\mathbf{X}^k, \pi^k, l^k, \mathbf{Z}^1, .., \mathbf{Z}^C|\theta) d\mathbf{Z}^1 ... d\mathbf{Z}^C \end{aligned} \quad (2)$$

where Π^k denotes the set of all possible HMM paths for sequence k. In the next section we describe the variational method for maximizing this objective.

3. Inference and learning

Using Jensen's inequality we obtain the following lower bound $\mathcal{L}_Q$ on the penalized log likelihood (Jordan et al., 1999):

$$\begin{aligned} & \log p(\theta)p(\mathbf{X}|\theta) \geq \\ & \log p(\theta) + \sum_{k=1}^{K} \int_{\mathbf{Z}^1 ... \mathbf{Z}^C} \sum_{l^k=1}^{C} \sum_{\pi^k \in \Pi^k} Q(\pi^k, l^k, \mathbf{Z}^1, ..., \mathbf{Z}^C) \\ & \times \log \frac{p(\mathbf{X}^k, \pi^k, l^k, \mathbf{Z}^1, ..., \mathbf{Z}^C|\theta)}{Q(\pi^k, l^k, \mathbf{Z}^1, ..., \mathbf{Z}^C)} d\mathbf{Z}^1 ... d\mathbf{Z}^C \\ & \equiv \mathcal{L}_Q \end{aligned} \quad (3)$$

[2] In our experiments the dimensionality of the observed configuration at each frame is 50 or higher.

where $Q(\pi^k, l^k, \mathbf{Z}^1, ..., \mathbf{Z}^C)$ is an arbitrary distribution. We now assume that this distribution factorizes as follows:

$$Q(\pi^k, l^k, \mathbf{Z}^1, ..., \mathbf{Z}^C) = Q(\pi^k)Q(l^k)\prod_{c=1}^{C} Q(\mathbf{Z}^c) \quad (4)$$

We maximize the variational bound $\mathcal{L}_Q$, subject to the mean field assumption in eq. 4, using the EM algorithm. In the E-step we keep θ fixed and estimate the factor distributions maximizing the variational bound. This is done via variational inference as described in the following subsection. In the M-step we maximize the expected complete penalized log likelihood given the hidden variable distributions. This yields closed-form updates for each of the parameters in θ. The complete penalized log-likelihood for our model is given by $\mathcal{L}^p = \mathcal{L} + \mathcal{P}$, where $\mathcal{L}$ is the log-likelihod term:

$$\begin{aligned}
\mathcal{L} &= \sum_{k=1}^{K}\Big[\log \tau^{l^k} + \log \text{Mult}(\pi_1^k) \\
&\quad + \sum_{i=2}^{N^k} \log p(\pi_i^k|\pi_{i-1}^k; \mathbf{d}^k) \\
&\quad + \sum_{i=1}^{N^k} \log \mathcal{N}(\mathbf{x}_i^k; \mathbf{z}_{\pi_i^k}^{l^k}, \Lambda^{l^k-1})\Big] \\
&\quad + \sum_{c=1}^{C}\sum_{m=1}^{M} \log \mathcal{N}(\mathbf{z}_m^c; \bar{\mathbf{z}}_m, \lambda_{\bar{z}}^{-1}\mathbf{I}) \\
&\quad + \sum_{c=1}^{C} \log \mathcal{N}(\mathbf{z}_1^c; \mathbf{z}_M^c, \lambda_{\bar{s}}^{-1}\mathbf{I}) \\
&\quad + \sum_{c=1}^{C}\sum_{m=2}^{M} \log \mathcal{N}(\mathbf{z}_m^c; \mathbf{z}_{m-1}^c, \lambda_{\bar{s}}^{-1}\mathbf{I}) \quad (5)
\end{aligned}$$

and $\mathcal{P}$ is the penalty term:

$$\begin{aligned}
\mathcal{P} &= \log p(\theta|\eta) \\
&= -\eta^{\bar{z}}||\bar{\mathbf{z}}_1 - \bar{\mathbf{z}}_M||^2 - \eta^{\bar{z}}\sum_{m=2}^{M}||\bar{\mathbf{z}}_m - \bar{\mathbf{z}}_{m-1}||^2 \\
&\quad + \log \mathcal{D}(\tau; \eta^\tau) + \sum_{k=1}^{K} \log \mathcal{D}(\mathbf{d}^k; \eta^d) \quad (6)
\end{aligned}$$

3.1. Variational Inference

The factor distributions $\{Q^*(\pi^k)\}_{k=1,..,K}$, $\{Q^*(l^k)\}_{k=1,..,K}$, $\{Q^*(\mathbf{Z}^c)\}_{c=1,..,C}$ maximizing the lower bound $\mathcal{L}_Q$ must satisfy the following equations (Jordan et al., 1999):

$$\begin{aligned}
\log Q^*(\pi^k) &= <\mathcal{L}>_{\sim\pi^k} + \text{const} \quad (7)\\
\log Q^*(l^k) &= <\mathcal{L}>_{\sim l^k} + \text{const} \quad (8)\\
\log Q^*(\mathbf{Z}^c) &= <\mathcal{L}>_{\sim\mathbf{Z}^c} + \text{const} \quad (9)
\end{aligned}$$

where $<>_{\sim h}$ denotes expectation with respect to all hidden variables except h. Note that equations (7, 8, 9) are coupled. Thus, a closed-form optimal solution is not possible. However, convergence to the optimal distributions is guaranteed if we iteratively update the distributions by solving each equation using the current estimates of the other factor distributions. The variational update steps are obtained by expanding the expectations on the right-hand side of equations (7, 8, 9). For brevity, we write $\psi^k(c) \equiv Q^*(l^k = c)$, $\gamma_i^k(m) \equiv Q^*(\pi_i^k = m)$, and $\xi_i^k(m,n) \equiv Q^*(\pi_i^k = m|\pi_{i-1}^k = n)$.

Variational update for $Q^*(\pi^k)$

$Q^*(\pi^k)$ is updated by applying the forward-backward algorithm (Rabiner, 1989) to an HMM with transition probabilities given by eq. 1 and unnormalized observation log-likelihoods given by:

$$\begin{aligned}
&\log \tilde{p}(\mathbf{x}_i^k|\pi_i^k = m) \\
&= -\frac{1}{2}\sum_{c=1}^{C}\psi^k(c)\left\langle(\mathbf{x}_i^k - \mathbf{z}_m^c)^T\Lambda^c(\mathbf{x}_i^k - \mathbf{z}_m^c)\right\rangle_{\mathbf{z}_m^c} \\
&\quad - \sum_{f=1}^{F}\log\lambda_f^c + \text{const} \quad (10)
\end{aligned}$$

Variational update for $Q^*(\mathbf{Z}^c)$

Let z_{mf}^c be the f-th entry in vector $\mathbf{z}_m^c$ with $f \in \{1, ..., F\}$ where F is the dimensionality of the configuration vector at each frame. It is easy to verify that, according to our probabilistic model, the random variables z_{mf}^c and $z_{mf'}^c$ for distinct features $f, f' \in \{1..., F\}$ are independent. Thus, we can write $Q^*(\mathbf{Z}^c) = \prod_{f=1}^{F} Q^*(\mathbf{z}^{c,f})$ where $\mathbf{z}^{c,f} = \left[z_{1f}^c, ..., z_{Mf}^c\right]^T$. $Q^*(\mathbf{z}^{c,f})$ is a multivariate Gaussian distribution with precision $S^{c,f}$ and mean $\mu^{c,f}$. The nonzero entries of $S^{c,f}$ are the diagonal entries $S_{m,m}^{c,f} = 2\lambda_s + \lambda_z + \lambda_f^c/\sum_{k=1}^{K}\sum_{i=1}^{N^k}\psi^k(c)\gamma_i^k(m)$ for $m = 1, ..., M$, the off-diagonal entries $S_{m,m+1}^{c,f} = S_{m+1,m}^{c,f} = -\lambda_s$ for $m = 1, ..., M-1$, and $S_{1,M}^{c,f} = S_{M,1}^{c,f} = -\lambda_s$. The mean $\mu^{c,f}$ is given by $\mu^{c,f} = S^{c,f}(\lambda_{\bar{z}}\bar{\mathbf{z}}^f + \mathbf{v}^{c,f})$ where $\bar{\mathbf{z}}^f = [\bar{z}_{1f}, ..., \bar{z}_{Mf}]^T$ and $\mathbf{v}^{c,f} = \lambda_f^c\frac{\sum_{k=1}^{K}\psi^k(c)\sum_{i=1}^{N^k}\gamma_i^k(m)\mathbf{x}_{if}^k}{\sum_{k=1}^{K}\psi^k(c)\sum_{i=1}^{N^k}\gamma_i^k(m)}$.

Variational update for $Q^*(l^k)$

The class label distribution $Q^*(l^k = c)$ is updated as $Q^*(l^k = c) = \rho_c^k/\sum_{c=1}^{C}\rho_c^k$ where

$$\begin{aligned}
\rho_c^k = \tau^c\prod_{f=1}^{F}(\lambda_f^c)^{\frac{N^k}{2}}\exp\Bigg\{&-\frac{1}{2}\sum_{m=1}^{M}\sum_{i=1}^{N^k}\gamma_i^k(m) \\
&\times\left\langle(\mathbf{x}_i^k - \mathbf{z}_m^c)^T\Lambda^c(\mathbf{x}_i^k - \mathbf{z}_m^c)\right\rangle_{\mathbf{z}_m^c}\Bigg\} \quad (11)
\end{aligned}$$

3.2. Parameter updates

The parameters θ are updated in the M-step. The update for each parameter is computed by setting the corresponding partial derivative of the expected penalized log-likelihood to zero. The update rules are:

$$\lambda_{\bar{z}} \leftarrow CMF/\left(\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{f=1}^{F}\left\langle (z_{m,f}^{c}-\bar{z}_{m,f})^2\right\rangle_{z_{m,f}^{c}}\right) \tag{12}$$

$$\lambda_{s} \leftarrow CMF/\left(\sum_{c=1}^{C}\sum_{m=1}^{M}\sum_{f=1}^{F}\left\langle (z_{m,f}^{c}-z_{m\vdash 1,f}^{c})^2\right\rangle_{\substack{z_{m,f}^{c}\\ z_{m\vdash 1,f}^{c}}}\right) \tag{13}$$

$$\lambda_{f}^{c} \leftarrow \sum_{k=1}^{K}\psi^{k}(c)N^{k}/\left(\sum_{k=1}^{K}\psi^{k}(c) \times\sum_{i=1}^{N^{k}}\sum_{m=1}^{M}\gamma_{i}^{k}(m)\left\langle (x_{i,f}^{k}-z_{m,f}^{c})^2\right\rangle_{z_{m,f}^{c}}\right) \tag{14}$$

$$\tau^{c} \leftarrow \left(\eta_{c}^{\tau}+\sum_{k=1}^{K}\psi^{k}(c)\right)/\left(\sum_{c'=1}^{C}(\eta_{c'}^{\tau}+\sum_{k=1}^{K}\psi^{k}(c'))\right) \tag{15}$$

$$\mathbf{d}_{j}^{k} \leftarrow \frac{1}{q}\left(\eta_{j}^{d}+\sum_{i=2}^{N^{k}}\sum_{m=1}^{M}\sum_{n\in T_{j}(m)}\xi_{i}^{k}(m,n)\right) \tag{16}$$

where $T_j(m) = \{n \in \{1,...,M\}$ s.t. $m-n=j$ or $M+m-n=j\}$, q is a constant enforcing the constraint $\sum_{j=1}^{J_\pi}\mathbf{d}_j^k = 1$, and $m \vdash 1 = (m-1)$ if $m > 1$, $m \vdash 1 = M$ otherwise.

In order to update the parent cycle we solve the linear system of M equations given by:

$$\frac{\partial\langle\mathcal{L}\rangle}{\partial\bar{\mathbf{z}}_m} = \lambda_{\bar{z}}\sum_{c=1}^{C}(<\mathbf{z}_m^c> -\bar{\mathbf{z}}_m) +\eta^{\bar{z}}(2\bar{\mathbf{z}}_{m-1}-4\bar{\mathbf{z}}_m+2\bar{\mathbf{z}}_{m+1})=0 \tag{17}$$

for $m = 2,...,M-1$, and by the two analogous equations corresponding to cases $m = 1, m = M$.

Estimating missing data If some sequences contain missing entries, we fill in the unobservable data during the M-step. The idea is to optimize the expected log-likelihood with respect to the missing entries. Let $()^\dagger$ denote the rows of the missing entries in frame $\mathbf{x}_i^k$. The update rule is:

$$(\mathbf{x}_i^k)^\dagger \leftarrow \left(\sum_{c=1}^{C}\psi^k(c)(\Lambda^c)^\dagger\right)^{-1} \times\sum_{c=1}^{C}\psi^k(c)\sum_{m=1}^{M}\gamma_i^k(m)(\Lambda^c)^\dagger(\mu^{c,m})^\dagger \tag{18}$$

where $(\Lambda^c)^\dagger$ is the square matrix sub-block corresponding to the missing entries.

4. Experiments

Data preprocessing We evaluated our method on several sets of motion capture sequences from the CMU Graphics Lab Motion Capture Database[3]. The data is represented in the form of Euler joint angles parameterized so as to avoid discontinuities. The configuration at each frame is a 62-dimensional vector. When generating animations, in addition to the joint angles, we used the 3D global translations of the body in the form of frame-to-frame 3D displacements of a root marker. In our experiments we have investigated the usefulness of PCA as a preprocessing step to reduce the dimensionality of the data. We have found that, while eliminating the last few principal components is generally beneficial, using fewer than 50 PCA dimensions results consistently in lower performance for all methods. Thus, here we report results obtained by applying PCA to each dataset and using only the first 50 dimensions. Furthermore, we show that HLM works equally well without this preprocessing by including also the results obtained by our method without the use of PCA.

Comparison We consider the following algorithms in our comparison:

- **HLM**: this is the novel hierarchical model described in this paper. We initialize the vectors $\bar{\mathbf{z}}_m$ by linearly interpolating the first $M/2$ frames of a sequence randomly chosen from the training set. For all styles c, λ_f^c was initially set equal to the sample precision of the f-th coordinate of the data. $\lambda_{\bar{z}}$ was initially set to 0.01 times the sample precision of all the data coordinates $x_{i,f}^k$, and λ_s was set to 0.1. The parameters d_j^k were initialized to $1/J_\pi$, with $J_\pi = 3$. The hyperparameters $\eta^{\bar{z}}, \eta^{\tau}, \eta^{d}$ were all set equal to 0.1. We initialized $Q^*(l_k)$ by adding small random noise to an equal-probability distribution over the labels. Finally, we initialized $Q^*(\mathbf{z}_m^c)$ by setting $\mu^{c,m}$ equal to $\bar{\mathbf{z}}_m$ and $s_f^{c,m}$ equal to λ_f^c. $Q^*(\pi^k)$ was then estimated from these initializations. In our experiments we kept $\lambda_{\bar{z}}$ fixed to its initial value, since doing so

[3] Available at http://mocap.cs.cmu.edu/

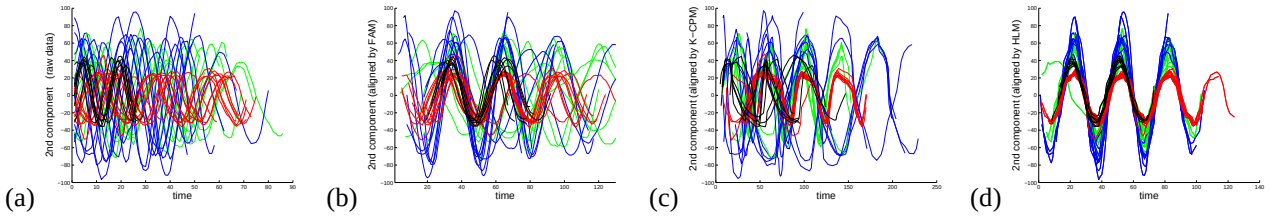

Figure 1. (a) Time series corresponding to the second PCA components of the 39 motions. Sequences having the same label in the CMU database are drawn with the same color. (b) The time series aligned by FAM. (c) Viterbi alignment computed using the CPM model. (d) Time-series warped according to the Viterbi alignment derived from our HLM model. Our algorithm successfully aligns all time series.

produced better results. We believe that our datasets are too small to be able to estimate this parameter reliably. We found that our variational algorithm generally converges very rapidly to a minimum. In our experiments we used 15 or fewer EM iterations, with 3 variational inference updates in each E-step.

- **CPM**: we modified the original Continuous Profile Model (Listgarten et al., 2005) to handle the cyclic nature of our data by using the transition distribution given by eq. 1. This algorithm aligns all sequences with respect to a single learned latent cycle. Finally, for each sequence it produces a prototypical cycle of length M obtained by averaging the motion warped according to the Viterbi alignment over the multiple observed cycles. Note that HLM differs from CPM in several ways. CPM aligns the data without performing clustering. In HLM each example is aligned with respect to style-specific latent cycles, which in turn are forced to be aligned with respect to the parent cycle. As we show in our experiments, this hierarchy leads to more accurate alignment than when warping all examples with respect to a single, generic cycle. Furthermore, our approach learns for each style a full PDF rather than a single point estimate.
- **FAM**: this algorithm uses the Functional Analysis Model described in (Ormoneit et al., 2005) to warp and segment the observed sequences into aligned cycles. As HLM, this model has the ability to fill-in missing data.
- **LGSSM**: this implements the algorithm described in (Chiappa et al., 2009). This method performs clustering of motions using a Bayesian Mixture of Linear Gaussian State-Space Models. As in HLM, the clustering and the model learning are done at the same time.

CPM and FAM produce for each sequence a prototypical cycle summarizing the motion. Thus, such methods can be naturally extended to model styles by applying a clustering algorithm to the aligned prototypical cycles. Here we evaluate these algorithmic extensions by running K-means on the prototypical cycles produced by CPM and FAM. The cluster centroids are finally used as style-prototypes. We denote with K-CPM and K-FAM the algorithms obtained by combining K-means with CPM and FAM, respectively.

4.1. Learning motion styles

In this section we show that HLM can be used to discover styles from a pool of motions, and to generate novel animations for each learned style. For this experiment we used a dataset of 39 locomotion sequences, taken from CMU subject categories 07, 08, 09, and 35. This set contains regular walking sequences performed by different subjects, as well as examples of striding and running. Figure 1(a) shows the second PCA component of each sequence plotted as a function of time. Sequences having the same motion label in the CMU database are plotted with the same color (there are four distinct CMU labels in this set). Note, however, that these labels are not provided to the algorithm, and that the motion styles are learned in a fully unsupervised way. We trained HLM and our "cyclic" version of CPM on this dataset. After training, we aligned the time series using the maximum likelihood HMM state path computed by applying the Viterbi algorithm (Rabiner, 1989) to both learned models. The results are shown in Figure 1(c) and (d) for CPM and HLM, respectively. Figure 1(b) shows the sequences aligned by FAM. There are significant alignment errors with the FAM and CPM models, while HLM synchronizes the time-series successfully, even sequences having noticeably different characteristics.

We also evaluated the quality of the clusterings by using the CMU style labels as ground truth data. The plot in Figure 2 reports the average cluster purity[4] obtained for different values of C. The purity values are computed by averaging the results over 50 runs for each algorithm. HLM yields consistently the best clustering results. Note that our algorithm performs roughly the same when applied to raw joint angles as opposed to data obtained from PCA. This indicates robustness to noise and high-dimensionality. Here K-FAM performs slightly better than K-CPM, but worse than HLM. As illustrated in this Figure, LGSSM produces very poor clustering results.

[4]The average purity is the weighted sum of the individual cluster purities, with weights proportional to the cluster sizes and normalized to sum to 1. The purity of a cluster is the fraction of motions in the majority ground truth class assigned to that cluster.

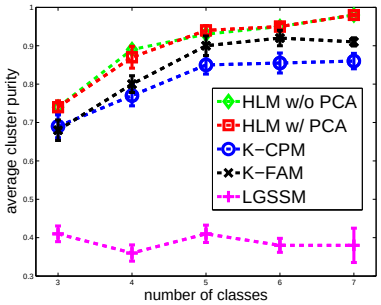

Figure 2. Average cluster purity obtained with LGSSM, K-FAM, K-CPM, and HLM (with both raw data and data processed by PCA) for different values of C. The figure includes error bars.

Examples of animations generated with the different models can be viewed at `http://www.cs.dartmouth.edu/hlm`. The animations show the means of the style-specific latent cycles obtained when setting $C = 4$. The motions learned by HLM are very realistic and stylistically distinct. One of the four learned styles clearly corresponds to running, and one to striding. The remaining two motions are different styles of walking. In contrast, the motions produced with K-CPM and K-FAM are noisy and jittery, possibly due to the inaccurate alignment. Furthermore, they appear to mix together different styles. We found that motions generated with the LGSSM model deviated considerably from the original motions, particularly as time progressed. This problem occurs even when the method clusters the sequences correctly, as this model generates the configuration at each time by using the estimate at the previous time step, and thus propagates errors over time. It typically generates body poses inconsistent with the training set within 20 frames.

4.2. Style blending

Several authors (Rose et al., 1998; Kovar et al., 2002; Grochow et al., 2004; Torresani et al., 2007) have proposed methods that generate novel styles by interpolating (or blending) *corresponding* body-poses taken from stylistically different sequences. Our algorithm can be used to establish these pose correspondences. Our approach aligns all sequences together and thus it can even support multi-way (as opposed to pairwise) interpolation. Furthermore, our approach learns synchronized latent cycle distributions. Consequently, novel styles can be directly generated from them, for example by interpolating the latent cycle expectations μ^c of different styles. Here we demonstrate this application using a dataset of six sequences, comprising three distinct styles: taichi walking, striding, and regular walking (note that as usual the style labels have not been provided to the algorithm). Figure 3 shows the result of blending together two of the learned latent cycles. The blending is obtained by computing $(\alpha(m)\mu_m^{c1} + (1 - \alpha(m))\mu_m^{c2})$, where $\alpha(m)$ is a value varying smoothly in $[0, 1]$ over the course of the sequence. The result is a realistic animation containing style transitions (see video at `http://www.cs.dartmouth.edu/hlm`).

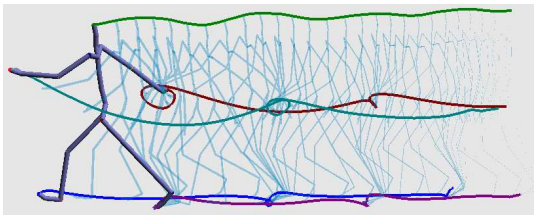

Figure 3. An illustration of the animation generated by blending learned latent cycles corresponding to "taichi" and "striding". Notice the varying walking style.

4.3. Filling in missing data

Despite the recent advances in technology, motion capture systems today are still prone to the problem of marker dropouts, i.e. markers that are lost by the tracker due to noise or occlusion. This problem is typically addressed in a post-processing stage via interactive software using interpolation methods. In contrast, our model can exploit the correlation among the joint angles and reconstruct missing data in a fully automatic way from the available data. For this experiment, we used the 39-sequence dataset previously introduced and trained the models on the raw joint angles. We selected a new walking sequence, not included in our original training set, from CMU subject category 7. We left the first half of this sequence unchanged. However, we eliminated 48 joint angles, corresponding to all the degrees of freedom of the upper body, including arms, the head, and the hips (note that this affects also the leg configuration) from the entire second half of the example. We use the update in eq. 18 to predict the missing entries. During inference we used the previously learned model to estimate the style label and the HMM state distributions for the new sequence. Note that the M-step update for missing data can also be used to fill-in unobservable entries in the training data during learning, although here we do not test such case. We compare our approach for handling missing entries with FAM, and a simple solution based on nearest neighbor (NN), as in (Taylor et al., 2007): for each frame containing missing entries, we find the most similar body configuration in the training set (in terms of Euclidean distance) and copy from it the data corresponding to the unobserved angles. We show reconstruction of missing joint angles using HLM and NN in Figure 4. The mean squared error per joint is 23.9 when using NN, and 9.3 with HLM. The motion filled in with HLM appears real, while the NN and FAM reconstructions look unnatural (see Figure 5). We found that the FAM model can effectively handle only cases where the number of missing entries is very small. On our challenging experiment FAM yields a reconstruction error greater than 200.

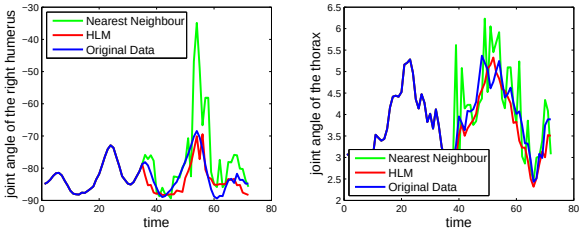

Figure 4. Filling in of right humerus (left) and thorax (right) joint angles. The original data is shown together with reconstructions computed with HLM and NN.

5. Conclusions

We have described a novel unsupervised method for learning locomotion styles using hierarchical priors and shown its versatility with a variety of applications. We have demonstrated that our algorithm outperforms state-of-the-art methods on the tasks of animation, style clustering, and filling in missing data. Our system uses a new efficient variational method which can infer the distributions of the model in a few minutes even when applied to large datasets of motions. Currently, our algorithm requires the number of classes as input. Future research will investigate Bayesian approaches for model selection. We are also interested in extending our model to represent motions with non-cyclic properties, such as turning or bending. More complex hierarchies and distributions may be needed to model these combinations of styles. Furthermore, we would like to study how our model can be adapted to create animations that simultaneously satisfy user-specified constraints and exhibit the styles learned during training. Although in this paper we have focused on the problem of modeling locomotion styles, we believe that our hierarchical approach can be applied effectively to model time-series in many other domains.

References

Brand, M., & Hertzmann, A. (2000). Style machines. *Proc. of SIGGRAPH* (pp. 183–192).

Chiappa, S., Kober, J., & Peters, J. (2009). Using bayesian dynamical systems for motion template libraries. In *Adv. in Neural Inform. Proc. Systems 21*, 297–304.

Elgammal, A. M., & Lee, C.-S. (2004). Separating style and content on a nonlinear manifold. *Proc. of Comp. Vision Pattern Recogn.* (pp. 478–485).

Grochow, K., Martin, S. L., Hertzmann, A., & Popović, Z. (2004). Style-based inverse kinematics. *ACM Trans. on Graphics*, *23*, 522–531.

Jordan, M. I., Ghahramani, Z., Jaakkola, T., & Saul, L. K. (1999). An introduction to variational methods for graphical models. *Machine Learning*, *37*, 183–233.

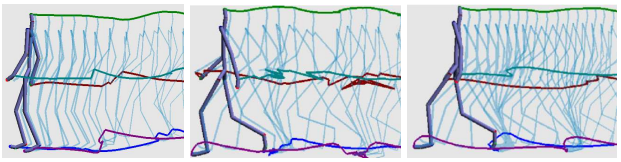

Figure 5. Reconstruction of motion with missing entries using FAM (left), NN (center), and HLM (right) to fill in the unobserved data. Note the jittery motion traces created by FAM and NN.

Kovar, L., Gleicher, M., & Pighin, F. (2002). Motion graphs. *ACM Trans. on Graphics*, *21*, 473–482.

Li, Y., Wang, T., & Shum, H.-Y. (2002). Motion texture: A two-level statistical model for character motion synthesis. *ACM Trans. on Graphics*, *21*, 465–472.

Listgarten, J., Neal, R. M., Roweis, S. T., & Emili, A. (2005). Multiple alignment of continuous time series. In *Adv. in Neural Inform. Proc. Systems 17*, 817–824.

Listgarten, J., Neal, R. M., Roweis, S. T., Puckrin, R., & Cutler, S. (2007). Bayesian detection of infrequent differences in sets of time series with shared structure. In *Adv. in Neural Inform. Proc. Systems 19*, 905–912.

Liu, K., Hertzmann, A., & Popovic, Z. (2005). Learning physics-based motion style with nonlinear inverse optimization. *ACM Trans. on Graphics*, *24*, 1071–1081.

Ormoneit, D., Black, M., Hastie, T., & Kjellström, H. (2005). Representing cyclic human motion using functional analysis. *Image and Vision Comp.*, 1264–1276.

Rabiner, L. R. (1989). A tutorial on HMMs and selected applications in speech recognition. *Proc. IEEE*, *77*.

Rose, C., Cohen, M., & Bodenheimer, B. (1998). Verbs and adverbs: multidimensional motion interpolation. *IEEE Computer Graphics and Application*, *18*, 32–40.

Taylor, G. W., Hinton, G. E., & Roweis, S. T. (2007). Modeling human motion using binary latent variables. In *Adv. in Neural Inform. Proc. Systems 19*, 1345–1352.

Torresani, L., Hackney, P., & Bregler, C. (2007). Learning motion style synthesis from perceptual observations. In *Adv. in Neural Inform. Proc. Systems 19*, 1393–1400.

Urtasun, R., Fleet, D. J., Geiger, A., Popovic, J., Darrell, T., & Lawrence, N. D. (2008). Topologically-constrained latent variable models. *Proc. Int. Conf. Machine Learning* (pp. 1080–1087).

Wang, J. M., Fleet, D. J., & Hertzmann, A. (2007). Multifactor gaussian process models for style-content separation. *Proc. Int. Conf. Machine Learning* (pp. 975–982).