Neural Networks 25 (2012) 191–199

Contents lists available at SciVerse ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Real-time stylistic prediction for whole-body human motions

Takamitsu Matsubara [a,b,*], Sang-Ho Hyon [b,c], Jun Morimoto [b]

[a] Graduate School of Information Science, NAIST, 8916-5, Takayama-cho, Ikoma, Nara, 630-0101, Japan
[b] Department of Brain Robot Interface, ATR-CNS, 2-2-2, Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0288, Japan
[c] Department of Robotics, Ritsumeikan University, 1-1-1, Nojihigashi, Kusatsu, Shiga, 525-8577, Japan

ARTICLE INFO

Article history:
Received 13 April 2011
Received in revised form 7 July 2011
Accepted 29 August 2011

Keywords:
Human motion prediction
Multifactor state-space models
Style-content separation
Stylistic prediction

ABSTRACT

The ability to predict human motion is crucial in several contexts such as human tracking by computer vision and the synthesis of human-like computer graphics. Previous work has focused on off-line processes with well-segmented data; however, many applications such as robotics require real-time control with efficient computation. In this paper, we propose a novel approach called *real-time stylistic prediction for whole-body human motions* to satisfy these requirements. This approach uses a novel generative model to represent a whole-body human motion including rhythmic motion (e.g., walking) and discrete motion (e.g., jumping). The generative model is composed of a low-dimensional state (phase) dynamics and a two-factor observation model, allowing it to capture the diversity of motion styles in humans. A real-time adaptation algorithm was derived to estimate both state variables and style parameter of the model from non-stationary unlabeled sequential observations. Moreover, with a simple modification, the algorithm allows real-time adaptation even from incomplete (partial) observations. Based on the estimated state and style, a future motion sequence can be accurately predicted. In our implementation, it takes less than 15ms for both adaptation and prediction at each observation. Our real-time stylistic prediction was evaluated for human walking, running, and jumping behaviors.

1. Introduction

Over the last decade, a considerable number of studies have been conducted on learning the generative models of human motion for modeling, prediction, and recognition (Howe, Leventon, & Freeman, 2000; Li, Wang, & Shum, 2002; Ormoneit, Sidenbladh, Blank & Hastie, 2001; Pavlovic, Rehg & MacCormick, 2000; Sidenbladh, Black & Fleet, 2000; Urtasun, Fleet, & Fua, 2006; Urtasun, Fleet, Hertzmann & Fua, 2005; Wang, Fleet, & Hertzmann, 2008). A significant limitation of these methodologies is that they cannot explicitly consider the natural variations of human motions in the generative model, widely referred to as *style* (Brand & Hertzmann, 2000; Grochow, Martin, Hertzmann & Popovic, 2004; Hsu, Pulli, & Popovic, 2005; Shapiro, Cao, & Faloutsos, 2006; Taylor & Hinton, 2009; Torresani, Hackney & Bregler, 2006; Wang, Fleet, & Hertzmann, 2007). For example, as illustrated in Fig. 1, even for an individual, each walking motion sequence has a distinct walking style. These differences can be much larger between different individuals. Therefore, to achieve highly accurate prediction for a newly observed motion sequence, adaptation of the generative model to the motion sequence by capturing the style of the sequence is necessary.

While most previous studies have focused on off-line processes with well-segmented data, many robotics applications (e.g., human–robot interaction (Onishi, Luo, Odashima, Hirano, Tahara & Mukai, 2007), imitation learning by humanoids (Ijspeert, Nakanishi, & Schaal, 2002; Inamura, Toshima, & Nakamura, 2002; Riley, Ude, Wada, & Atkeson, 2003) and powered suits (Fukuda, Tsuji, Kaneko, & Otsuka, 2003; Kawamoto, Kanbe, & Sankai, 2003)) require real-time control with high accuracy and efficient computation in the prediction procedure.

In this paper, we propose a novel approach called *real-time stylistic prediction for whole-body human motions*. Unlike previous studies (Brand & Hertzmann, 2000; Taylor & Hinton, 2009; Wang et al., 2007), as illustrated in Fig. 2, in our approach the generative model adapts to a newly observed motion sequence by estimating its style by a real-time process. Being able to perform this process in real-time is based on (1) the simple structure of the generative model and (2) the adaptation algorithm which requires small computational effort. We propose a generative model for whole-body human motion that is composed of a low-dimensional state (phase) dynamics and a two-factor (phase dependent observation bases and style parameter) observation model to capture the diversity of motion styles in humans. We also present a learning procedure to acquire the model from a variety of motion sequences

* Corresponding author at: Graduate School of Information Science, NAIST, 8916-5, Takayama-cho, Ikoma, Nara, 630-0101, Japan.
E-mail address: takam-m@is.naist.jp (T. Matsubara).

0893-6080/$ – see front matter © 2011 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2011.08.008

Fig. 1. Illustration of *style* in human motion sequences. Ten walking phase-aligned sequences by two individuals are overwritten in order of phase. The style in walking behavior is considered as a control variable for the spatial variations.

including a diversity of motion styles. A real-time adaptation algorithm was derived using an on-line Expectation-Maximization (EM) algorithm for computationally efficient inference of both the corresponding state variables and the style parameter from non-stationary unlabeled sequential observations. Moreover, with a simple modification, the algorithm allows real-time adaptation even from incomplete (partial) observations. Such applicability of the adaptation algorithm for partial observations is very important in a practical sense because we often meet situations where some elements of the observations are missing due to the limited number of sensors available or occlusions (Chai & Hodgins, 2005). Based on the estimated state and style, the generative model can accurately predict a future motion sequence.

On the other hand, most of the existing models that explicitly estimate the style of motion can achieve neither real-time adaptation nor non-stationary motion estimation since the inference algorithm requires large computational effort (Brand & Hertzmann, 2000; Ormoneit et al., 2001; Sidenbladh et al., 2000; Taylor & Hinton, 2009; Urtasun & Fua, 2004; Wang et al., 2007).

The organization of this paper is as follows. In Section 2, we present a novel generative model to represent a whole-body human motion including rhythmic motion (e.g., walking) and discrete motion (e.g., jumping). We also present a learning procedure to acquire the model from a variety of motion sequences including a diversity of motion styles. In Section 3, a real-time adaptation algorithm is derived by applying an approximated EM algorithm to the generative model. Moreover, we present a simple modification in the adaptation algorithm, which allows real-time adaptation even from incomplete (partial) observations. A real-time prediction method of future motion sequence based on the estimated state and style is also presented. In Section 4, the effectiveness of our real-time stylistic prediction is validated for human walking, running, and jumping behaviors with motion capture data. Section 5 concludes this paper.

2. Learning generative model

This section describes the proposed generative model and a learning procedure for the model with a stylistic data set.

2.1. Generative model for whole-body human motions

We first define the notation of the proposed generative model. $\mathbf{x} \in \mathbb{R}^d$ is the state variable, $\mathbf{y} \in \mathbb{R}^D$ is the observation and the probability distribution $p(\mathbf{y}_t|\mathbf{x}_t; \mathbf{w})$ is the observation model. $p(\mathbf{x}_{t+1}|\mathbf{x}_t)$ is the state-transition probability distribution. The parameter vector $\mathbf{w} \in \mathbb{R}^J$ is an additional latent variable that controls the spatial variation of observations. We call this the style parameter. Its graphical model is depicted in Fig. 3. For periodic and discrete motions, we explicitly define the state variable $\mathbf{x}$ as

$$\mathbf{x}_t = [\phi_t \ \omega_t]^T = \begin{cases} [\psi_t \ \dot{\psi}_t]^T & \text{(Rhythmic)} \\ [p_t \ \dot{p}_t]^T & \text{(Discrete).} \end{cases} \quad (1)$$

That is, we define the state variable $\mathbf{x}$ by phase ϕ as a point on a one dimensional sphere in two dimensional Euclidean space $\phi \equiv \psi \in \mathbb{S} \subset \mathbb{R}^2$ and its velocity $\omega \equiv \dot{\psi}$ to represent its periodicity of rhythmic motions, similar to Ormoneit et al. (2001) and Urtasun and Fua (2004). We also define the phase ϕ as a point on a one dimensional closed line segment $\phi \equiv p \in \mathbb{L}$ for discrete motions to represent its non-periodicity (discreteness). The explicit use of these assumptions in the generative model yields the low-dimensional state variable $\mathbf{x}$. Moreover, as presented in the next section, it allows a simple learning algorithm for the generative model from data.

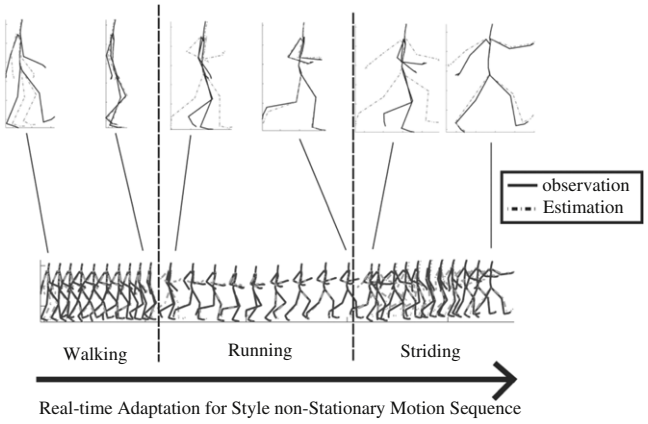

Fig. 2. Illustration of the real-time adaptation and prediction of the generative model for a non-stationary motion sequence with styles (walking behaviors). The test sequence consists of three motions, walking, running and striding generated by different individuals. The solid human figure is as observed and the dashed one is the predicted motion as a result of adaptation of the generative model to the observation sequence. The adaptation is achieved by on-line EM incrementally with little computation at each observation. For all motions, the model is rapidly adapted to the style of the recent test sequence since the time-forgetting factor effectively forgets past observations.

Based on the above assumptions, we conclude that the state-transition model and the observation model are modeled by Gaussian distributions as:

$$p(\mathbf{x}_{t+1}|\mathbf{x}_t) = \mathcal{N}(\mu_x(\mathbf{x}_t), \Sigma_x(\mathbf{x}_t)), \quad (2)$$

$$p(\mathbf{y}_t|\mathbf{z}_t; \mathbf{w}) = \mathcal{N}(\mu_y(\mathbf{z}_t; \mathbf{w}), \Sigma_y(\mathbf{z}_t; \mathbf{w})), \quad (3)$$

where

$$\mathbf{z}_t = g(\mathbf{x}_t) = \begin{cases} [\cos(\phi_t) \ \sin(\phi_t)]^T & \text{(Rhythmic)} \\ \phi_t & \text{(Discrete)} \end{cases} \quad (4)$$

and the observation model is defined as a probabilistic mapping from a phase ϕ_t (as $\mathbf{z}_t$) to an observation $\mathbf{y}_t$. The velocity of phase ω_t governs the temporal variation of the time-series, that is, it controls the velocity of human motions generated by the model. For rhythmic motions, $\mathbf{z}_t = g(\phi_t) \in \mathbb{R}^2$ represents a point on a manifold $\mathbb{S}$ in $\mathbb{R}^2$ where the radius is $r = 1$ and the angle is ϕ_t. This state representation allows us to approximately measure the geodesic distance between points on $\mathbb{S}$ as the Euclidean distance in $\mathbb{R}^2$. For discrete motions, $\mathbf{z}_t$ is the equivalent of ϕ_t.[1]

2.2. Learning procedure with a stylistic data set

The learning procedure assumes we have multiple human motion sequences including a diversity of motion styles. Let $\mathbf{Y}^s = [\mathbf{y}^s_1 \cdots \mathbf{y}^s_{C(s)}]^T \in \mathbb{R}^{C(s)\times D}$ denote a time-invariant motion sequence with a distinct style, where $s \in \{1, 2, \ldots, S\}$ is the style index in which each value indicates a corresponding distinct style, $c \in \{1, 2, \ldots, C(s)\}$ is the content index that corresponds to the phase

[1] For the case of a discrete motion, $\mathbf{z}_t$ is a scalar; however, it is kept as a vector notation for simplicity of the overall description.

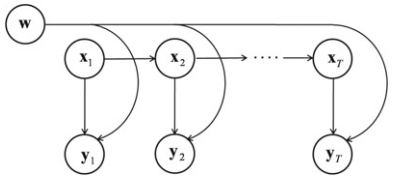

Fig. 3. A generative model representing time series data for a human motion. The state variable is defined as $\mathbf{x}_t = [\phi_t\ \omega_t]$ where the phase ϕ of the state variable is defined as a point on one dimensional sphere in two dimensional Euclidean space $\phi \equiv \psi \in \mathbb{S} \subset \mathbb{R}^2$ and as a point on one dimensional closed line segment $\phi \equiv p \in \mathbb{L}$ for rhythmic and discrete motions, respectively. The observation $\mathbf{y}_t$ is conditionally independent from all other variables given the state $\mathbf{x}_t$. $\mathbf{x}_{t+1}$ is conditionally independent given the state $\mathbf{x}_t$. $\mathbf{w}$ is the style parameter vector invariant for the time.

and $\mathbf{y}_c^s \in \mathbb{R}^D$ is an observation with the style indexed by s and content c. In the following, let us assume that a set of training sequences $\mathcal{D} = \{\mathbf{Y}^1, \ldots, \mathbf{Y}^S\}$ is given for learning a generative model as a stylistic data set. The learning procedure is composed of the following three steps: (1) Data alignment by phase information, (2) Extraction of observation bases, (3) Learning generative model from the bases. These three steps achieve learning of a compact generative model of a human motion that can represent large stylistic variations, where both the state variables and the style parameters of the generative model can be low-dimensional.

2.2.1. Data alignment by phase information

For the subsequent steps, we align all motion sequences in a stylistic data set by phase. We introduce data alignment strategies for rhythmic and discrete motions.

Rhythmic motions: We utilize auto-correlative and cross-correlative coefficients for the alignment process. First, we maximize the auto-correlative coefficient for identifying the period T of each sequence as: $T^s \leftarrow \arg\max_j A^s(j)$, where $A^s(j) = \sum_n^N \mathbf{y}_n^{s\,T} \mathbf{y}_{n+j}^s$ is the auto-correlative coefficient with the self-index shift j in phase with a style indexed by s. Next, we maximize the cross-correlative coefficient to find the optimal cross-index shift h in phase as: $h^s \leftarrow \arg\max_j C^s(j)$, where $C^s(j) = \sum_n^N \mathbf{y}_n^{b\,T} \mathbf{y}^s_{j+\mathrm{rd}(\frac{nT^s}{T^b})}$, and index b is the style index corresponding to the sequence that has the shortest period. T^b is the period of the shortest period indexed by b. The function $\mathrm{rd}(\cdot)$ is a round-off function. The above procedures yield an aligned data matrix:

$$\mathbf{Y}_a^{all} = \begin{bmatrix} \mathbf{y}^1_{\mathrm{rd}(h^1+\frac{T^1}{T^b})} & \cdots & \mathbf{y}^1_{\mathrm{rd}(h^1+T^1)} \\ \vdots & \ddots & \vdots \\ \mathbf{y}^S_{\mathrm{rd}(h^S+\frac{T^S}{T^b})} & \cdots & \mathbf{y}^S_{\mathrm{rd}(h^S+T^S)} \end{bmatrix} \tag{5}$$

where $\mathbf{Y}_a^{all} = [\mathbf{Y}_a^1 \cdots \mathbf{Y}_a^S]^T \in \mathbb{R}^{DS \times C}$ and $\mathbf{Y}_a^s \in \mathbb{R}^{C \times D}$. Note that each row of the matrix $\mathbf{Y}_a^{all}$ are the observations corresponding to the same value of the phase ϕ. Each column indicates a corresponding motion sequence indexed by s.

Discrete motions: We align each motion sequence $\mathbf{Y}^s$ from the same starting point $\phi = 0$ to the goal point $\phi = 1$. The aligned data matrix $\mathbf{Y}_a^{all}$ can be obtained by simply setting T^s as the duration of motion and $h^s = 0$ in Eq. (5) for all s.

2.2.2. Extraction of observation bases

Since the aligned data matrix $\mathbf{Y}_a^{all}$ is a rectangular matrix, Singular Value Decomposition (SVD) based matrix factorization can be applied to extract *observation bases*. By applying the factorization, we can form a style-content factorial model (referred to as the asymmetric bilinear model in Tenenbaum and Freeman (2000)).

Let $\mathbf{Y}_a^{all\,\mathrm{VT}}$ be an $S \times DC$ matrix stacked from $DS \times C$ Matrix $\mathbf{Y}_a^{all}$. Then, SVD for this matrix leads to the following factorial representation as

$$\mathbf{Y}_a^{all\,\mathrm{VT}} = \mathbf{U}\mathbf{S}\mathbf{V}^T \approx \mathbf{W}\tilde{\mathbf{Y}}. \tag{6}$$

We define the style parameter matrix $\mathbf{W} = [\mathbf{w}^1 \cdots \mathbf{w}^S]^T \in \mathbb{R}^{S \times J}$ to be the first J ($\leq S$) rows of $\mathbf{U}$, and the content parameter matrix $\tilde{\mathbf{Y}} = ([\bar{\mathbf{Y}}^1 \cdots \bar{\mathbf{Y}}^J]^T)^{\mathrm{VT}} \in \mathbb{R}^{J \times DC}$ to be the first J columns of $\mathbf{S}\mathbf{V}^T$. As a result, we can obtain an approximated form as $\mathbf{Y}_a^s \approx \sum_{j=1}^J w_j^s \bar{\mathbf{Y}}^j$. The obtained matrix $\bar{\mathbf{Y}}^j \in \mathbb{R}^{C \times D}$ is named the j-th observation basis, and the vector $\mathbf{w}^s \in \mathbb{R}^J$ is the s-th style parameter vector.

2.2.3. Learning generative model

With the extracted observation bases $\bar{\mathbf{Y}}^j$ for all j, we can learn the generative model. Since each point of an observation basis corresponds to a value of phase on $\mathbb{S} \subset \mathbb{R}^2$ (or $\mathbb{L}$), we learn a mapping between $\bar{\mathbf{y}}^j$ and $\mathbf{z}$ using all data of each basis. Here we utilize Gaussian process regression (Rasmussen & Williams, 2006) because it allows us to derive an analytically tractable predictive distribution and to learn hyperparameters by maximization of the marginalized likelihood.

Each basis $\bar{\mathbf{Y}}^j$ is independently modeled as a Gaussian process as

$$p(\bar{\mathbf{Y}}^j | \mathbf{Z}, \boldsymbol{\beta}^j) \propto \exp\left(-\frac{1}{2}\mathrm{Tr}((\mathbf{K}_y^j)^{-1}\bar{\mathbf{Y}}^j(\bar{\mathbf{Y}}^j)^T)\right) \tag{7}$$

where $\mathbf{Z}$ is the phase-aligned matrix corresponding to $\bar{\mathbf{Y}}^j$ and $\mathbf{K}_y^j \in \mathbb{R}^{C \times C}$ is the gram matrix in which (p, q) entry is $k_y^j(\mathbf{z}_p, \mathbf{z}_q) = \beta_1^j \exp(-\frac{\beta_2^j}{2}\|\mathbf{z}_p - \mathbf{z}_q\|^2) + \delta_{\mathbf{z}_p,\mathbf{z}_q}/\beta_3^j$, and the hyperparameter is represented as $\boldsymbol{\beta}^j = \{\beta_1^j, \beta_2^j, \beta_3^j\}$. With the above settings, the predictive distribution of the j-th observation basis $\bar{\mathbf{y}}^{j*}$ given a novel input $\mathbf{z}^*$ can be easily derived as

$$p(\bar{\mathbf{y}}^{j*} | \mathbf{z}^*, \bar{\mathbf{Y}}^j, \mathbf{Z}) = \mathcal{N}(\bar{\mu}^j(\mathbf{z}^*), \bar{\Sigma}^j(\mathbf{z}^*)) \tag{8}$$

where

$$\bar{\mu}^j(\mathbf{z}^*) = (\bar{\mathbf{Y}}^j)^T(\mathbf{K}_y^j)^{-1}\mathbf{k}_y^j(\mathbf{z}^*), \tag{9}$$

$$\bar{\Sigma}^j(\mathbf{z}^*) = (k_y^j(\mathbf{z}^*, \mathbf{z}^*) - \mathbf{k}_y^j(\mathbf{z}^*)^T(\mathbf{K}_y^j)^{-1}\mathbf{k}_y^j(\mathbf{z}^*))\mathbf{I}, \tag{10}$$

and $\mathbf{k}_y^j(\mathbf{z}^*) = [k_y^j(\mathbf{z}^*, \mathbf{z}_1) \cdots k_y^j(\mathbf{z}^*, \mathbf{z}_C)]^T$ (Rasmussen & Williams, 2006). The predictive distribution for observation $\mathbf{y}^*$ given a novel input $\mathbf{z}^*$ with a style parameter vector $\mathbf{w}^s$ can be written with the standard result of a Gaussian distribution and invariant linear transformation as

$$p(\mathbf{y}^* | \mathbf{z}^*, \bar{\mathbf{Y}}^{1:J}, \mathbf{Z}; \mathbf{w}^s) = \mathcal{N}(\mu_y(\mathbf{z}^*; \mathbf{w}^s), \Sigma_y(\mathbf{z}^*; \mathbf{w}^s)) \tag{11}$$

where

$$\mu_y(\mathbf{z}^*; \mathbf{w}^s) = \sum_{j=1}^J w_j^s \bar{\mu}^j(\mathbf{z}^*), \tag{12}$$

$$\Sigma_y(\mathbf{z}^*; \mathbf{w}^s) = \sum_{j=1}^J (w_j^s)^2 \bar{\Sigma}^j(\mathbf{z}^*). \tag{13}$$

This predictive distribution is set as the observation model $p(\mathbf{y}_t | \mathbf{x}_t; \mathbf{w})$ of the generative model in Eq. (3).

The state transition model $p(\mathbf{x}_{t+1} | \mathbf{x}_t)$ of the generative model in Eq. (2) is simply modeled under the assumption that all data are observed with a fixed sampling rate as $p(\mathbf{x}_{t+1} | \mathbf{x}_t) = \mathcal{N}(\mathbf{A}\mathbf{x}_t, \mathbf{Q})$ where the state transition matrix is $\mathbf{A} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$ and the covariance matrix of process noise $\mathbf{Q}$ should be set appropriately for the data by hand. The mean and variance of the state transition model are written as $\mu_x(\mathbf{x}_t) = \mathbf{A}\mathbf{x}_t$, $\Sigma_x(\mathbf{x}_t) = \mathbf{Q}$, respectively.

3. Real-time stylistic prediction

In this section, we derive a real-time adaptation algorithm for the generative model to a newly observed test sequence by using an approximated EM algorithm, i.e., the algorithm that estimates a time-series of state variables $\hat{\mathbf{X}} = [\hat{\mathbf{x}}_1 \cdots \hat{\mathbf{x}}_T]^T$ and the style parameter vector $\hat{\mathbf{w}}$ from a portion of the test sequence $\hat{\mathbf{Y}} = [\hat{\mathbf{y}}_1 \cdots \hat{\mathbf{y}}_T]^T$ in an on-line manner. In Section 3.1, we start from the derivation of off-line (batch) computations of E-step and M-step by introducing simple approximations. This gives us an efficient computation for the adaptation of the generative model for a test sequence. In Section 3.2, the derived algorithm is further extended to a fully recursive, time-forgetting and computationally efficient on-line algorithm by a modification of the likelihood as suggested by Sato and Ishii (2000). Section 3.3 describes how we can achieve real-time stylistic prediction of a test sequence with the generative model and the adaptation algorithm. Section 3.4 presents a modification of the adaptation algorithm to allow real-time adaptation even from incomplete (partial) observations.

3.1. Estimation of state variables and style parameters by using an approximated EM algorithm

To derive an efficient adaptation algorithm for the generative model to a test sequence, we fit the EM algorithm to the problem with simple approximations. The EM alternates between optimizing a distribution over the state variables (the E-step) and optimizing the style parameters given the posterior distribution obtained by the E-step (the M-step) (Dempster, Laird, & Rubin, 1977; Ghahramani & Hinton, 1996; Shumway & Stoffer, 1982). Introducing a distribution over state variables as $\mathcal{Q}(\hat{\mathbf{X}})$, a lower bound of the log-likelihood $\mathcal{F}(\mathcal{Q}, \hat{\mathbf{w}})$ given observations $\hat{\mathbf{Y}}$ can be derived from the likelihood $\mathcal{L}(\hat{\mathbf{Y}}|\hat{\mathbf{w}})$. The E-step holds the style parameters fixed and sets $\mathcal{Q}$ to be the posterior distribution over the state variables:

$$\mathcal{Q}_{k+1}(\hat{\mathbf{X}}) \leftarrow \arg\max_{\mathcal{Q}} \mathcal{F}(\mathcal{Q}, \hat{\mathbf{w}}_k) = p(\hat{\mathbf{X}}|\hat{\mathbf{Y}}, \hat{\mathbf{w}}_k). \tag{14}$$

This maximizes $\mathcal{F}$ w.r.t. $\mathcal{Q}$ turning the lower bound of the log-likelihood into the likelihood as $\mathcal{F}(\mathcal{Q}, \hat{\mathbf{w}}) = \mathcal{L}(\hat{\mathbf{Y}}|\hat{\mathbf{w}})$. The M-step holds the distribution fixed and computes the style parameters that maximizes $\mathcal{F}$ as $\hat{\mathbf{w}}_{k+1} \leftarrow \arg\max_{\hat{\mathbf{w}}} \mathcal{F}(\mathcal{Q}_{k+1}, \hat{\mathbf{w}}) = \arg\max_{\hat{\mathbf{w}}} \int_{\hat{\mathbf{X}}} \mathcal{Q}_{k+1}(\hat{\mathbf{X}}) \log p(\hat{\mathbf{X}}, \hat{\mathbf{Y}}|\hat{\mathbf{w}}) d\hat{\mathbf{X}}$. Therefore, the E-step in k-th iteration calculates the posterior of the state variable $\mathbf{X}$ as $p(\hat{\mathbf{X}}|\hat{\mathbf{Y}}, \hat{\mathbf{w}}_k)$, and the M-step updates the new style parameter $\hat{\mathbf{w}}_{k+1}$ with the posterior. This alternative procedure typically converges to a locally optimal point through a small number of iterations.

Analytical computations of the E-step and M-step for our generative model, however, are intractable due to the nonlinearity in observation bases against phase. Here, we introduce simple approximations to create analytical solutions for both steps. Under the assumption that all observation bases are smooth functions over the phase variable ϕ, the E-step can be approximately calculated in the same way as the Extended Kalman Filter (EKF) (Ko & Fox, 2008). Thus, we approximate $\mathcal{Q} \approx \mathcal{N}(\hat{\mathbf{X}})$ to obtain an analytical computation of the E-step. Subsequently, the analytical computation of the M-step is derived with the following approximation as

$$\begin{aligned}\hat{\mathbf{w}}_{k+1} &\leftarrow \arg\max_{\hat{\mathbf{w}}} \int_{\hat{\mathbf{X}}} p(\hat{\mathbf{X}}|\hat{\mathbf{Y}}; \hat{\mathbf{w}}_k) \log p(\hat{\mathbf{X}}, \hat{\mathbf{Y}}; \hat{\mathbf{w}}) d\hat{\mathbf{X}} \\ &\approx \arg\max_{\hat{\mathbf{w}}} \sum_{t=2}^{T} \log p(\hat{\mathbf{y}}_t|\hat{\mathbf{x}}_{\mathrm{pm},t}; \hat{\mathbf{w}})\end{aligned} \tag{15}$$

where $\hat{\mathbf{X}}_{\mathrm{pm}}$ is the mean of the posterior distribution as $\hat{\mathbf{X}}_{\mathrm{pm}} = \int_{\hat{\mathbf{X}}} \hat{\mathbf{X}} p(\hat{\mathbf{X}}|\hat{\mathbf{Y}}; \hat{\mathbf{w}}_k) d\hat{\mathbf{X}}$ which is obtained by the E-step. The details of the derived adaptation algorithm are presented in Appendix.

3.2. On-line implementation of the proposed estimation method

Here we present an on-line implementation of the proposed estimation method. As suggested by Sato and Ishii (2000), we introduce a time-forgetting factor in our likelihood in Eq. (15). This yields a recursive, time-forgetting on-line EM algorithm, in which both the E-step and M-step can be calculated at each new observation with little computation.

In the on-line EM algorithm, the weighted mean is defined as

$$\langle\langle f(x)\rangle\rangle_T \equiv \eta_T \sum_{t=1}^{T} \left\{ \prod_{s=t+1}^{T} \lambda_s \right\} f(x_t) \tag{16}$$

where $\eta_T \equiv \{\sum_{t=1}^{T}\{\prod_{s=t+1}^{T} \lambda_s\}\}^{-1}$ and the parameter λ_s ($0 \leq \lambda_s \leq 1$) is a time-dependent forgetting factor which is introduced for forgetting the effect of the past observations. η_T is a normalized coefficient and plays a role similar to a learning rate. This weighted mean $\langle\langle\cdot\rangle\rangle_i$ has a step-wise equation

$$\langle\langle f(x)\rangle\rangle_T = (1 - \eta_T)\langle\langle f(x)\rangle\rangle_{T-1} + \eta_T f(x_T) \tag{17}$$

where $\eta_T = \{1 + \lambda_T/\eta_{T-1}\}^{-1}$. Introducing the above weighted mean in the computation of the EM algorithm yields an on-line EM algorithm. i.e., a recursive, time-forgetting on-line adaptation algorithm. If we set $\lambda_s = \lambda < 1.0$, this can be applied for non-stationary motion sequences with styles. The details of the derived on-line adaptation algorithm are presented in Appendix.

3.3. Stylistic prediction of future observations

With the generative model in which both the state variables and the style parameter are estimated from a given test sequence, we could predict its future states and observations by the following simple algorithm. Given a portion of a test sequence until time t, we can estimate the expectation of the corresponding state variable $\hat{\mathbf{x}}_t$ and the estimated value of the style parameter $\hat{\mathbf{w}}_t$. One-step ahead prediction of the expectation of state variable $\hat{\mathbf{x}}_{t+1|t}$ can be obtained by $\hat{\mathbf{x}}_{t+1|t} = \int p(\mathbf{x}_{t+1}|\hat{\mathbf{x}}_t)\mathbf{x}_{t+1} d\mathbf{x}_{t+1} = \mathbf{A}\hat{\mathbf{x}}_t$. Corresponding estimation of observation $\hat{\mathbf{y}}_{t+1|t}$ is then obtained from $\hat{\mathbf{z}}_{t+1|t} = g(\hat{\phi}_{t+1|t})$ and Eq. (3) as $\hat{\mathbf{y}}_{t+1|t} = \int p(\mathbf{y}_{t+1}|\hat{\mathbf{z}}_{t+1|t}; \hat{\mathbf{w}}_t)\mathbf{y}_{t+1} d\mathbf{y}_{t+1} = \mu(\hat{\mathbf{z}}_{t+1|t}; \hat{\mathbf{w}}_t)$. With these calculations, stylistic prediction by identifying the style of the test sequence can be achieved. Executing the prediction recursively, it is possible to achieve multiple-step ahead predictions of future states and observations.

3.4. Adaptation from incomplete (partial) observations

In this section, we present a modification in the adaptation algorithm so that it allows real-time adaptation even from incomplete (partial) observations. In a practical sense, such applicability of the adaptation algorithm to partial observations is very important because we often meet situations where some elements of the observations are missing due to the limited number of sensors available or occlusions. Our approach is inspired by a model for dealing with packet losses in a wireless network with the Kalman filter (Liu & Goldsmith, 2004).

For simplicity (but without loss of generality), we divide an observation into two parts as $\mathbf{y} = [\breve{\mathbf{y}}^T, \grave{\mathbf{y}}^T]^T$ where $\breve{\mathbf{y}} \in \mathbb{R}^{D-D_m}$ are observable elements and $\grave{\mathbf{y}} \in \mathbb{R}^{D_m}$ are missing observations, that is, we assume that the index of missing elements in the observation is known. To manage such missing observations for the adaptation algorithm, we modify the observation model in Eq. (3) as

$$\begin{aligned} &p\left(\left[\begin{matrix}\breve{\mathbf{y}} \\ \grave{\mathbf{y}}\end{matrix}\right]_t \middle| \mathbf{z}_t; \mathbf{w}\right) \\ &= \mathcal{N}\left(\left[\begin{matrix}\breve{\boldsymbol{\mu}}_y(\mathbf{z}_t; \mathbf{w}) \\ \grave{\boldsymbol{\mu}}_y(\mathbf{z}_t; \mathbf{w})\end{matrix}\right], \left[\begin{matrix}\mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{21} & \mathbf{R}_{22} + \sigma_p \mathbf{I}_m\end{matrix}\right]\right)\end{aligned} \tag{18}$$

where $\boldsymbol{\mu}_y = [\check{\boldsymbol{\mu}}_y^T, \grave{\boldsymbol{\mu}}_y^T]^T$ is the mean, $\mathbf{R} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{bmatrix}$ is the covariance matrix, $\mathbf{R}_{11}$ is $\mathbb{R}^{(D-D_m)\times(D-D_m)}$, $\mathbf{R}_{12} = \mathbf{R}_{21}^T$ is $\mathbb{R}^{(D-D_m)\times D_m}$ and $\mathbf{R}_{22}$ is $\mathbb{R}^{D_m\times D_m}$ matrices. As indicated in Liu and Goldsmith (2004), in the above model, the missing observations $\grave{\mathbf{y}}$ can be represented by setting $\sigma_p \to \infty$ in Eq. (18). With this setting, an adaptation algorithm for the partial observation can be derived by introducing the following modifications in the fully observable case (see Appendix) as

$$\mathbf{y} \to \check{\mathbf{y}}, \qquad \boldsymbol{\mu}_y \to \check{\boldsymbol{\mu}}_y, \tag{19}$$

$$\mathbf{H} \to \check{\mathbf{H}}, \qquad \mathbf{R} \to \mathbf{R}_{11} \tag{20}$$

where $\mathbf{H} = \begin{bmatrix} \check{\mathbf{H}} \\ \grave{\mathbf{H}} \end{bmatrix}$ and $\check{\mathbf{H}}$ is $\mathbb{R}^{(D-D_m)\times D_m}$, $\grave{\mathbf{H}}$ is $\mathbb{R}^{D_m\times D_m}$ matrices. The derivation is based on a formula of the inverse of a partitioned matrix (Liu & Goldsmith, 2004).

The derived adaptation algorithm allows us to adapt the generative model of whole-body human motions to a test sequence from partially observed information.

4. Experimental results

In this section, we validate the effectiveness of our real-time stylistic prediction method for human motions using data captured by a motion capture system. Section 4.1 first presents the effectiveness for rhythmic motions such as walking and running behaviors in comparison with several on-line prediction methods with respect to computational costs and prediction accuracy. Its real-time calculability and applicability for partial observations are also validated through experiments. In Section 4.2, the proposed method is applied to discrete motions such as forward and vertical jumping behaviors to demonstrate the effectiveness of the proposed method for several kinds of behaviors.

4.1. Stylistic prediction for walking motions

We first learned a generative model from a collection of motion sequences. The data were taken from the CMU Graphics Lab's motion capture database (http://mocap.cs.cmu.edu). Each observation $\mathbf{y}_t = [\mathbf{q}_t^T\ \mathbf{r}_t^T\ \mathbf{v}_t^T]^T \in \mathbb{R}^{62}$ was down-sampled to 60 Hz, and focused on full-body joint Euler angles $\mathbf{q}_t \in \mathbb{R}^{56}$. To obtain a diversity of walking styles for this experiment, we selected several motion sequences of walking, running and striding motions captured from four subjects (**subject-A, B, C, D**).[2] More specifically, we selected eight walking sequences and one striding sequence of **subject-A**, four running sequences of **subject-B**, two running sequences and one walking sequence of **subject-C**, and one running sequence of **subject-D**.[3] The training sequences for model learning were chosen as six walking sequences and one striding sequence of **subject-A**, three running sequences of **subject-B** and two running sequences of **subject-C**. The dimension of the style parameter vector was set as $J = 4$ with a guide of the spectrum of singular values obtained. The estimated observation bases with the above settings are plotted in Fig. 4(e).

4.1.1. Verification of the learned generative model

First, we verified the learned generative model for stylistic prediction using the batch EM algorithm. For several segmented test sequences, the prediction accuracy of the adapted generative model was evaluated. The adaptation was executed with the EM algorithm using a portion of the test sequence, which was set as 0.0–2.0 s (120 frames) for walking and 0.0–1.0 s (60 frames) for running, based on their motion frequencies. Another portion for 0.5 s (30 frames) was then used as the ground truth for evaluation of the prediction accuracy. Test sequences were set as a striding sequence of **subject-A**, a running sequence of **subject-B**, a walking sequence of **subject-C**, and a running sequence of **subject-D**. Note that none of the test sequences were included in the training sequences, and no motion sequence of **subject-D** was included in the training sequences, i.e., **subject-D** was a completely unknown subject for the generative model. As a criterion for evaluation of the predictive accuracy, we used the average prediction error: $E_{\mathrm{rms}}^{\mathrm{off}} = \frac{1}{GD}\sum_{t=i+1}^{G+i} \|\mathbf{y}_t - \hat{\mathbf{y}}_{t|i}\|^2$, where G is the number of predicted frames, D is the dimension of observation and i is the index of the terminal point of the test sequence used for adaptation. In all cases, the EM procedure converged within 10 iterations. Fig. 4(a)–(d) illustrates the result of the stylistic prediction for the test sequences compared with the ground truth. Each predicted pose is very similar to the ground truth for all cases including the unknown subject. The estimated style parameters are plotted in Fig. 4(f). As shown in (f), the estimated style parameters were largely different due to the distinct style of each test sequence, producing high accuracy predictions. The average prediction error for all test sequences $E_{\mathrm{rms}}^{\mathrm{off}} = 1.68$ with $J = 1$ is 58% reduced by using $J = 4$ as $E_{\mathrm{rms}}^{\mathrm{off}} = 0.71$. These results verified the learned generative model for stylistic prediction.

4.1.2. Real-time stylistic prediction

Next, we evaluated our real-time stylistic prediction method with the derived adaptation algorithm and the learned generative model. As a non-stationary motion sequence with styles, a test sequence was prepared successively uniting three different test sequences. In this experiment, the adaptation procedure and prediction at each observation approximately required less than 15 ms in our Matlab implementation on a PC (Intel(R) Core-i7 CPU 3.33 GHz), i.e., about 13 ms for the E-step, 0.2 ms for the M-step and 0.5 ms for one-step ahead prediction of observation. Compared with the average walking period $T = 1.2$ s this processing time would be relatively short and effective enough for real-time prediction.

We compared the prediction performance of our proposed method with standard on-line prediction methods. Specifically, we compared it with recursive least square linear regression (RLS) with a time-forgetting factor λ (Haykin, 2002) and with Radial Basis Function Networks (Bishop, 1995) with a time-forgetting factor as standard approaches for on-line function approximation. For the RBFs, principal component analysis (PCA) was applied separately for each test sequence and obtained a low-dimensional subspace (RBFs + PCA). The basis was located on a grid with an even interval in each dimension of 3-dimensional feature space obtained by PCA. We used 1000 ($=10\times10\times10$) basis functions in this experiment. As a nonparametric approach, we implemented a particle filter as used in Sidenbladh et al. (2000) and Ormoneit et al. (2001). In this approach, the required posterior density function is represented by a set of random samples $\{[\mathbf{x}^k\ \mathbf{w}^k]^T\}_{k=1}^K$ with associated weights m^k as $p(\mathbf{x}_t, \mathbf{w}_t|\mathbf{y}_{1:T}) \approx \sum_{k=1}^K m^k \delta([\mathbf{x}\ \mathbf{w}]^T - [\mathbf{x}_t^k\ \mathbf{w}_t^k]^T)$ where K is the number of particles. As the number of samples becomes very large, this approximation becomes an equivalent representation of the posterior in the sense of an optimal Bayesian estimate. The proposal distribution for random sampling of particles and the likelihood model were set by the state-transition model and the observation model of our proposed model as in Eq. (3). The transition model of the style parameter was newly set as $\mathbf{w}_{t+1} \sim \mathcal{N}(\mathbf{w}_t, \sigma_w \mathbf{I})$.

[2] The correspondence of the subjects to the labels in the CMU motion capture database is **subject-A**:08 **subject-B**:35, **subject-C**:02 and **subject-D**:16.

[3] {08_01–08_09, 35_20–35_23, 02_01–02_03, 16_35}.amc. Training sequences contain {08_02–08_08, 35_20–35_22, 02_02–02_02 }.amc.

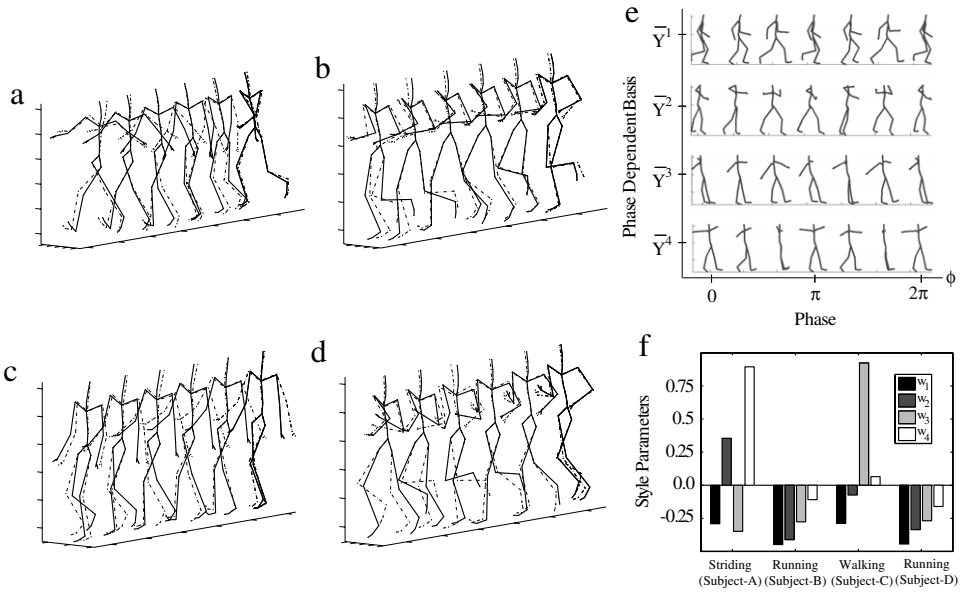

Fig. 4. Results of off-line stylistic prediction for human walking and running sequences. (a)–(b) show the prediction results where solid-lines indicate the ground truth in each frame and dash-lines show the predicted pose. The observation bases are shown in (e). The estimated style parameter **w** for test sequences is plotted in (f) where each w_j corresponds to $\bar{\mathbf{Y}}^j$ for all j. (a) Shows the prediction results for a **striding-style walking** sequence, (b) shows the results for a **running** sequence and (c) shows a **normal walking** sequence. These sequences seem to have different motion styles and such differences in motion style are captured by the estimated style parameters as shown in (f) to achieve the predictions with high accuracy. (d) Also shows the results for a **running** sequence which seems to be a slightly different motion style from (b). Such a small difference in motion style is also captured by the estimated style parameters as in shown (f).

Table 1
Prediction results for non-stationary human walking data in our Matlab implementation on a PC (Intel(R) Core-i7 CPU 3.33 GHz).

Method	Time (s)	$E^{\text{on}}_{\text{rms}}$
Proposed	0.01	0.85
RLS	2.00×10^{-4}	1.76
RBFs + PCA	0.09	1.40
PF ($K = 500$)	0.13	1.73 (0.05)
PF ($K = 10{,}000$)	105	1.23 (0.05)

The performance of the stylistic prediction was evaluated by the on-line average prediction error $E^{\text{on}}_{\text{rms}} = \frac{1}{T}\sum_{i=1}^{T}\{\frac{1}{GD}\sum_{t=i+1}^{G+i}\| \mathbf{y}_t - \hat{\mathbf{y}}_{t|i}\|^2\}$, where G is the number of predicted frames, D is the dimension of observation, T is the number of total frames of the test sequence and i is the index of the observed frame. The predicted pose $\hat{\mathbf{y}}_{i|i}$ is $\hat{\mathbf{y}}_{i|i} = \mu(\hat{\mathbf{x}}_i; \hat{\mathbf{w}}_i)$ where, $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{w}}_i$ are estimated by the adaptation algorithm using sequential observation until $\mathbf{y}_i$.

Table 1 shows the average prediction error in predictions ($G = 25$ frames). At each observation for a test sequence, adaptation and prediction were executed. Then, the prediction performance was evaluated by the average prediction error $E^{\text{on}}_{\text{rms}}$ for all cases. For the proposed method, the time-forgetting factor was set as $\lambda = 0.8$ so that the learning algorithm can quickly adapt to newly acquired data while old data is properly discarded. For the RLS and RBFs + PCA, the forgetting factor was set as $\lambda = 0.95$. Since the number of parameters required in these methods are much larger than the proposed method, the larger forgetting factor is necessary to properly evaluate the likelihood for the parameters. From Table 1, the proposed method showed the best performance among all methods in terms of both computational cost and prediction accuracy. The results of these comparisons suggest that it is impossible to represent human motion by well-known linear dynamics. Moreover, while the use of non-linear dynamics represented by RBFs potentially captures the non-linearity; the high-dimensionality resulting from the use of non-linear function approximators makes it difficult to achieve successful modeling with a modest amount of data obtained in on-line. A particle filtering approach can be competitive in terms of prediction performance with a huge number of particles; however, it requires prohibitive computational cost for real-time processes.

Table 2
Prediction from partial observations.

Missing portions	$E^{\text{on}}_{\text{rms}}$
(1) One arm	0.87
(2) One leg	0.91
(3) One leg and one arm	0.91

4.1.3. Prediction from partial observation

We also evaluated our prediction method for partial observations. For this experiment, three conditions were considered (1) missing one arm, (2) missing one leg, and (3) missing both one arm and one leg.[4] For such scenarios, the proposed method was applied for predictions and the results evaluated by the on-line prediction error using the non-stationary motion sequence with styles as a test sequence. Table 2 shows the experimental results. In all cases, the prediction performance was not much impaired by missing data. These results suggest the effectiveness of the proposed method for partial observations. Fig. 5 depicts prediction results for partial observations.

For further validation of the proposed method for partial observations, we also applied the method to a different data set from the CMU motion capture database, which has been used in Lawrence (2007) and Taylor, Hinton, and Roweis (2006). We prepared 31 motion sequences of one subject (composed of walking and running) as training data and also prepared one walking sequence as test data.[5] As in Lawrence (2007), the test sequence was modified in the two ways: (1) missing right leg and (2) missing upper body. The prediction accuracy was evaluated by

[4] One arm includes 7 DoFs: radius(1), wrist(2), hand(1), finger(1), thumb(2). One leg includes 7 DoFs: femur(3), tibia(1), foot(2), toes(1).

[5] The training data was composed of {35_1–17, 19–26 and 30–34}.amc, and the motion sequence 35_29.amc was used as test data.

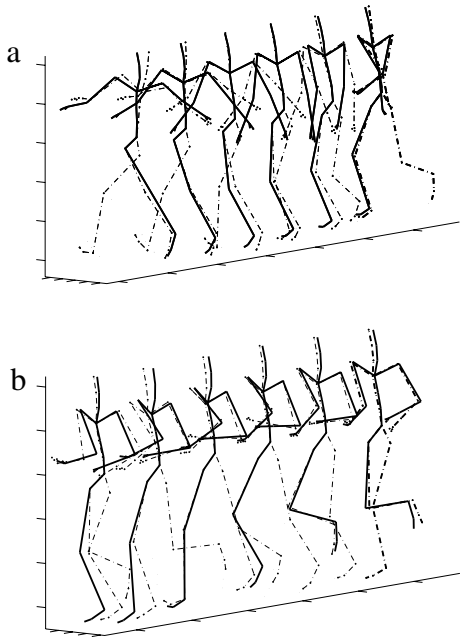

Fig. 5. Results of predictions for walking behaviors from partial observations (Left-leg information is missing). Solid-lines indicate the ground truth in each frame and dash-lines show the predicted pose. (a) Shows the result for a **striding-style** walking sequence and (b) shows the result for a **running** sequence. For both cases, even though the style of each sequence is largely different and left-leg information is missing, the predicted poses are very similar to the ground truth.

root mean square error of the missing joint angles averaged over 60 frames.

Fig. 6 shows the evaluation results of the on-line predictions. Each line was obtained by applying a zero-phase shift low-pass filter with 1.5 cut-off frequency to the result of the on-line predictions. The dimension of the style parameter vector was set as $J = 2$, and the time-forgetting factor was set as $\lambda = 0.8$. Fig. 6 indicates that, for both (1) and (2), our method rapidly converged at around 0.4 s and resulted in good prediction accuracy of missing body movements. As presented in Lawrence (2007) and Taylor et al. (2006), for this data set, the prediction of the missing upper body is easier than that of the missing lower limb. Note that only our proposed method is able to achieve the adaptation and the prediction on-line in real time.

4.2. Stylistic prediction for jumping motions

Next, we evaluated the proposed method for jumping motions. Again, the data were taken from the CMU database with the same setting of walking motions described in the previous section. To include a diversity of jumping motion styles (e.g., forward or vertical) in this experiment, we selected several motion sequences of jumping motions captured from two subjects (**subject-E, F**). More specifically, we selected seven jumping sequences of **subject-E** and two jumping sequences of **subject-F**.[6] The dimension of the style parameter vector was set manually as $J = 6$ by considering the range of singular values.

Using the learned generative model, we applied on-line prediction to two test sequences. For validation, the first 60 frames were utilized for on-line adaptation of the style and state variables,

[6] The correspondence of the subjects to the label in the CMU motion capture database is **subject-E**:13 **subject-F**:16. Training sequences contain {13_11, 13_13, 13_19, 13_39–13_42, 16_01–16_02}.amc. Test sequences contain {13_32, 13_40}.amc.

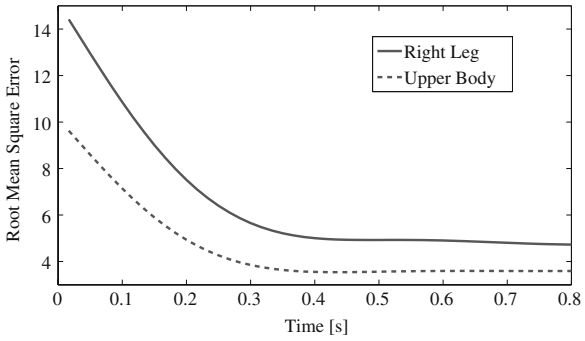

Fig. 6. Results of the on-line predictions of missing body movements. The graph indicates the root mean square error of the missing joint angles averaged over 60 future frames at each observation. "right leg" and "upper body" indicate the the cases of (1) missing right leg and (2) missing upper body in the test sequence, respectively. For the both cases, our method rapidly converged and resulted in good prediction accuracy of missing body movements.

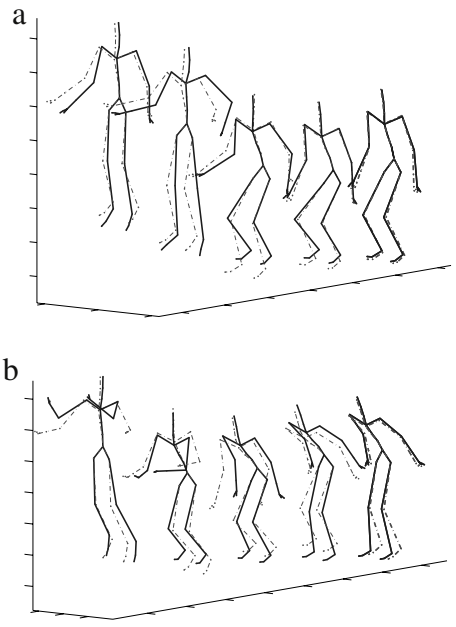

Fig. 7. Results of predictions for jumping motions. Two test sequences were used for validation. Initial motion sequences before takeoff are used for adaptation, and then future motion sequences after takeoff are predicted and compared with the ground truth to calculate prediction errors. Solid-lines indicate the ground truth in each frame and dash-lines show the predicted pose. (a) Shows the result for a **vertical jumping** motion after a deep squat motion. (b) Shows the result for a **forward jumping** motion with a large arm swing. The proposed method captured these motion-sequence specific features well in the predicted poses.

and then future observations were predicted and validated based on the average prediction errors $E^{\text{off}}_{\text{rms}}$ ($G = 30$ frames) for both cases. As a result, the errors were 0.97 and 1.05, and these values were significantly small compared with the standard deviations of both test sequences at 6.09 and 3.50. Fig. 7 depicts snapshots of both predictions and the ground truth. For both cases, each predicted pose is very similar to the ground truth. More concretely, in Fig. 7, (a) shows the result for a vertical jumping motion where the hips and knees are used for a vertical jump, while (b) shows the result for a forward jumping motion where the arms swing more than the hips and knees. The proposed method captured these motion-sequence specific features very well in the predicted poses. These results verified the effectiveness of our prediction method even for discrete motions, and suggests the applicability of this method for a wide range of human motions.

5. Conclusions

In this paper, we proposed a novel approach called real-time stylistic prediction for modeling and predicting human motions. Our algorithm can be run in real-time and manage non-stationary motion sequences with styles by capturing the style on-line with relatively little computation. Moreover, the algorithm allows real-time adaptation even from incomplete (partial) observations. Its effectiveness was demonstrated on motion capture data.

In the proposed approach, human motions are modeled by a generative model which is composed of a low-dimensional state (phase) dynamics and a two-factor (phase dependent observation bases and style parameter) observation model designed for capturing the diversity of motion styles in humans. To achieve higher prediction accuracy, a more complex model as the state dynamics (e.g., nonlinear state transitions or factorized by multiple variables) would be necessary. The proposed approach can incorporate such a complex state dynamics by extending both the adaptation and prediction algorithms. For example, a nonlinear state transition model can be incorporated in the proposed approach. In such a case, the adaptation algorithm is derived with the linearization of the state transition about the current mean of the state variable, and the prediction algorithm is executed by nonlinear regression at each time evolution. Thus, with complex state dynamics, the adaptation and prediction algorithms require additional computational effort which makes it difficult to be implemented in real time. Therefore, there is the trade-off between the prediction accuracy and the computational effort required in adaptation and prediction algorithms.

The ability to predict human motion is crucial in several contexts such as human tracking by computer vision and the synthesis of human-like computer graphics. Previous work has focused on off-line processes with well-segmented data; thus, many applications such as robotics require real-time control with efficient computation. The proposed method could be applied to these applications. Our future work will address real robotics applications such as human–robot interaction, imitation learning by humanoids and powered suits.

Our algorithm has free parameters such as time-forgetting and dimension of style parameter. Our future work will also address setting them by a relevant determination from data.

Acknowledgments

This research was supported by the Strategic Research Program for Brain Sciences "Brain Machine Interface Development" by the Ministry of Education, Culture, Sports, Science and Technology, Japan, and also partially supported by the Grant-in-Aid for Scientific Research from Japan Society for the Promotion of Science (WAKATE-B22700177).

Appendix A. Derived EM steps

E-step

$$\mathbf{x}_{t|t-1} = \mathbf{A}\hat{\mathbf{x}}_{t-1} \tag{A.1}$$

$$\Sigma_{x,t|t-1} = \mathbf{A}\hat{\Sigma}_{x,t-1}\mathbf{A}^T + \mathbf{Q} \tag{A.2}$$

$$\mathbf{H}_{t|t-1} = \left.\frac{\partial \mu(\mathbf{x};\hat{\mathbf{w}}_{t-1})}{\partial \mathbf{x}}\right|_{\mathbf{x}=\mathbf{x}_{t|t-1}} \tag{A.3}$$

$$\mathbf{K}_t = \Sigma_{x,t|t-1}\mathbf{H}_{t|t-1}^T \cdot (\mathbf{H}_{t|t-1}\Sigma_{x,t|t-1}\mathbf{H}_{t|t-1}^T + \mathbf{R}) \tag{A.4}$$

$$\hat{\mathbf{x}}_t = \mathbf{x}_{t|t-1} + \mathbf{K}_t(\mathbf{y}_t - \mu(\mathbf{x}_{t|t-1};\hat{\mathbf{w}}_{t-1})) \tag{A.5}$$

$$\hat{\Sigma}_{x,t} = (\mathbf{I} - \mathbf{K}_t\mathbf{H}_{t|t-1})\Sigma_{x,t|t-1}. \tag{A.6}$$

M-step

$$\hat{\mathbf{w}}_t = \langle \mathbf{v}^T\mathbf{v}\rangle_t^{-1}\langle \mathbf{v}^T\mathbf{y}\rangle_t, \tag{A.7}$$

where,

$$\mathbf{v}_t = [\bar{\mu}^1(\hat{\mathbf{x}}_t)\cdots\bar{\mu}^J(\hat{\mathbf{x}}_t)] \tag{A.8}$$

$$\langle\cdot\rangle_T = \frac{1}{T}\sum_{t=1}^{T}(\cdot_t). \tag{A.9}$$

On-line M-step

$$\hat{\mathbf{w}}_t = \langle\!\langle \boldsymbol{\mu}^T\boldsymbol{\mu}\rangle\!\rangle_t^{-1}\langle\!\langle \boldsymbol{\mu}^T\mathbf{y}\rangle\!\rangle_t \tag{A.10}$$

$$\langle\!\langle \boldsymbol{\mu}^T\boldsymbol{\mu}\rangle\!\rangle_t = (1-\eta_t)\langle\!\langle \boldsymbol{\mu}^T\boldsymbol{\mu}\rangle\!\rangle_{t-1} + \eta_t\boldsymbol{\mu}(\hat{\mathbf{x}}_t)^T\boldsymbol{\mu}(\hat{\mathbf{x}}_t) \tag{A.11}$$

$$\langle\!\langle \boldsymbol{\mu}^T\mathbf{y}\rangle\!\rangle_t = (1-\eta_t)\langle\!\langle \boldsymbol{\mu}^T\mathbf{y}\rangle\!\rangle_{t-1} + \eta_t\boldsymbol{\mu}(\hat{\mathbf{x}}_t)^T\mathbf{y}_t \tag{A.12}$$

$$\eta_t = \left\{1 + \frac{\lambda_t}{\eta_{t-1}}\right\}^{-1} \tag{A.13}$$

where

$$\boldsymbol{\mu}(\mathbf{x}_{t|t-1}) = [\mu^1(\mathbf{x}_{t|t-1})\cdots\mu^J(\mathbf{x}_{t|t-1})] \tag{A.14}$$

$$\boldsymbol{\mu}(\hat{\mathbf{x}}_t) = [\mu^1(\hat{\mathbf{x}}_t)\cdots\mu^J(\hat{\mathbf{x}}_t)] \tag{A.15}$$

$$\langle\cdot\rangle_T = \frac{1}{T}\sum_{t=1}^{T}(\cdot_t) \tag{A.16}$$

$$\langle\!\langle\cdot\rangle\!\rangle_T = \eta_T\sum_{t=1}^{T}\left\{\prod_{s=t+1}^{T}\lambda_s\right\}(\cdot_t) \tag{A.17}$$

$$\eta_T = \left\{\sum_{t=1}^{T}\left\{\prod_{s=t+1}^{T}\lambda_s\right\}\right\}^{-1}. \tag{A.18}$$

Appendix B. Supplementary data

Supplementary material related to this article can be found online at doi:10.1016/j.neunet.2011.08.008.

References

Bishop, C. M. (1995). *Neural networks for pattern recognition*. Oxford University Press.
Brand, M., & Hertzmann, A. (2000). Style machines. In *SIGGRAPH* (pp. 183–192).
Chai, J., & Hodgins, J. K. (2005). Performance animation from low-dimensional control signals. *ACM Transactions on Graphics*, *24*, 686–696.
Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, *39*, 1–38.
Fukuda, O., Tsuji, T., Kaneko, M., & Otsuka, A. (2003). A human-assisting manipulator teleoperated by EMG signals and arm motions. *IEEE Transaction on Robotics and Automation*, *19*, 210–222.
Ghahramani, Z., & Hinton, G. E. (1996). Parameter estimation for linear dynamical systems. *Technical report*. Dept. Computer Science, Univ. of Toronto (pp. 1–6).
Grochow, K., Martin, S. L., Hertzmann, A., & Popovic, Z. (2004). Style-based inverse kinematics. *ACM Transactions on Graphics*, *23*(1), 522–531.
Haykin, S. (2002). *Adaptive filter theory*. Prentice Hall.
Howe, N. R., Leventon, M. E., & Freeman, W. T. (2000). Bayesian reconstruction of 3D human motion from single-camera video. In *Advances in neural information processing systems: Vol. 12* (pp. 820–826).
Hsu, E., Pulli, K., & Popovic, J. (2005). Style translation for human motion. *ACM Transactions on Graphics*, *24*(3), 1082–1089.
Ijspeert, A. J., Nakanishi, J., & Schaal, S. (2002). Learning attractor landscapes for learning motor primitives. In *Advances in neural information processing systems: Vol. 15* (pp. 1523–1530).
Inamura, T., Toshima, I., & Nakamura, Y. (2002). Acquisition and embodiment of motion elements in closed mimesis loop. In *IEEE international conference on robotics and automation* (pp. 1539–1544).
Kawamoto, H., Kanbe, S., & Sankai, Y. (2003). Power assist method for HAL-3 using EMG-based feedback controller. In *IEEE international conference on systems, man and cybernetics* (pp. 1648–1653).

Ko, J., & Fox, D. (2008). GP-Bayes filters: Bayesian filtering using Gaussian process prediction and observation models. In *IEEE/RSJ international conference on intelligent robots and systems* (pp. 3471–3476).

Lawrence, N. (2007). Learning for larger datasets with the Gaussian process latent variable model. In *Proceedings of the eleventh international workshop on artificial intelligence and statistics* (pp. 243–250).

Liu, X., & Goldsmith, A. (2004). Kalman filtering with partial observation losses. In *IEEE conferences on decision and control* (pp. 4180–4186).

Li, Y., Wang, T., & Shum, H.-Y. (2002). Motion texture: a two-level statistical model for character motion synthesis. *ACM Transactions on Graphics*, *21*(3), 465–472.

Onishi, M., Luo, Z., Odashima, T., Hirano, S., Tahara, K., & Mukai, T. (2007). Generation of human care behaviors by human-interactive robot RI-MAN. In *IEEE international conference on robotics and automation* (pp. 3128–3129).

Ormoneit, D., Sidenbladh, H., Blank, M., & Hastie, T. (2001). *Advances in neural information processing systems*: *Vol. 13. Learning and tracking cyclic human motion* (pp. 894–900).

Pavlovic, V., Rehg, J. M., & MacCormick, J. (2000). *Advances in neural information processing systems*: *Vol. 12. Learning switching linear models of human motion* (pp. 981–987).

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. MIT Press.

Riley, M., Ude, A., Wada, K., & Atkeson, C.G. (2003). Enabling real-time full-body imitation: a natural way of transferring human movement to humanoids. In IEEE international conference robotics and automation (pp. 2368–2374).

Sato, M., & Ishii, S. (2000). On-line em algorithm for the normalized Gaussian network. *In Neural Computation*, *12*, 407–432.

Shapiro, A., Cao, Y., & Faloutsos, P. (2006). Style components. In *Graphics interface* (pp. 33–39).

Shumway, R. H., & Stoffer, D. S. (1982). An approach to time series smoothing and forecasting using the EM algorithm. *Journal of Time Series Analysis*, *3*, 253–264.

Sidenbladh, H., Black, M. J., & Fleet, D. J. (2000). Stochastic tracking of 3D human figures using 2D image motion. In *European conference computer vision*. *Vol. 2* (pp. 702–718).

Taylor, G. W., & Hinton, G. E. (2009). Factored conditional restricted Bolzmann machines for modeling motion style. In *International conference on machine learning* (pp. 1025–1032).

Taylor, G. W., Hinton, G. E., & Roweis, S. T. (2006). Modeling human motion using binary latent variables. In *Proceedings of advances in neural information processing systems* (pp. 1345–1352).

Tenenbaum, J. B., & Freeman, W. T. (2000). Separating style and content with bilinear models. *Neural Computation*, *12*, 1247–1283.

Torresani, L., Hackney, P., & Bregler, C. (2006). Learning motion style synthesis from perceptual observations. In *Advances in Neural Information Processing Systems*: *Vol. 19* (pp. 1393–1400).

Urtasun, R., Fleet, D.J., & Fua, P. (2006). 3D people tracking with Gaussian process dynamical models. In *IEEE Computer society conference on computer vision and pattern recognition* (pp. 238–245).

Urtasun, R., Fleet, D.J., Hertzmann, A., & Fua, P. (2005). Priors for people tracking from small training sets. In *IEEE international conference on computer vision* (pp. 403–410).

Urtasun, R., & Fua, P. (2004). 3D Human body tracking using deterministic temporal motion models. In *European Conference on Computer Vision*. *Vol. 3* (pp. 92–106).

Wang, J. M., Fleet, D. J., & Hertzmann, A. (2007). Multifactor Gaussian process models for style-content separation. In *International conference on machine learning* (pp. 975–982).

Wang, J. M., Fleet, D. J., & Hertzmann, A. (2008). Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *30*, 283–298.