

Revisiting Gaussian Process Dynamical Models

Jing Zhao, Shiliang Sun

Shanghai Key Laboratory of Multidimensional Information Processing,
Department of Computer Science and Technology, East China Normal University,
500 Dongchuan Road, Shanghai 200241, China
Email: jzhao2011@gmail.com, slsun@cs.ecnu.edu.cn

Abstract

The recently proposed Gaussian process dynamical models (GPDMs) have been successfully applied to time series modeling. There are four learning algorithms for GPDMs: maximizing a posterior (MAP), fixing the kernel hyperparameters $\bar{\alpha}$ (Fix. $\bar{\alpha}$), balanced GPDM (B-GPDM) and two-stage MAP (T.MAP), which are designed for model training with complete data. When data are incomplete, GPDMs reconstruct the missing data using a function of the latent variables before parameter updates, which, however, may cause cumulative errors. In this paper, we present four new algorithms (MAP⁺, Fix. $\bar{\alpha}$ ⁺, B-GPDM⁺ and T.MAP⁺) for learning GPDMs with incomplete training data and a new conditional model (CM⁺) for recovering incomplete test data. Our methods adopt the Bayesian framework and can fully and properly use the partially observed data. We conduct experiments on incomplete motion capture data (walk, run, swing and multiple-walker) and make comparisons with the existing four algorithms as well as k -NN, spline interpolation and VGPDs. Our methods perform much better on both training with incomplete data and recovering incomplete test data.

1 Introduction

The Gaussian process dynamical model (GPDM) was recently proposed by augmenting the Gaussian process latent variable model with a GP prior to model sequential motion data and predict the latent positions [Wang *et al.*, 2006]. It provides a nonlinear probabilistic mapping from latent positions to human poses and a nonlinear dynamical mapping on the latent space. Wang *et al.* [2008b] described and compared four algorithms for learning GPDMs: maximizing a posterior (MAP), fixing the kernel hyperparameters $\bar{\alpha}$ (Fix. $\bar{\alpha}$), balanced GPDM (B-GPDM) and two-stage MAP (T.MAP).

GPDMs are widely applicable to sequential data analysis such as people tracking, motion data recognition and synthesis and computer animation. For example, Urtasun *et al.* [2006] introduced the balanced GPDM method for learning smooth prior models of human poses and motions for 3D people tracking. Park and Yoo [2011] used the GPDM for

phoneme classification. Gamage *et al.* [2011] employed the GPDM as an alternative to hidden Markov models and artificial neural networks for hand gesture recognition in the context of sign language translation. Henter *et al.* [2012] introduced the GPDM for speech representation and synthesis. An *et al.* [2012] presented an online method for grasping motion learning using the GPDM.

Some variants and extensions of GPDMs have also been developed to adapt to specific applications. For human tracking, the particle filter GPDM was proposed, which can improve the stability and robustness of tracking [Raskin *et al.*, 2008] and deal with multi-target tracking [Wang *et al.*, 2008a]. For trajectory prediction, GPDMs were adapted to learn effective representations of the environment dynamics in continuous partially observable Markov decision processes [Dallaire *et al.*, 2009]. For modeling multiple activities, back constraints and topological constraints were incorporated within the local linear GPDM [Urtasun *et al.*, 2007; 2008]. In order to account for multiple types of dynamics, Chen *et al.* [2009] proposed to combine switching models with GPDMs to produce a switching GPDM, which has a switching layer on top of the latent variables. Similarly, the switching shared GPDM (SSGPDM) which is a nonparametric switching state-space model was proposed by Chen *et al.* [2009]. It is an extension of the shared GPDM [Deena and Galata, 2009], where multiple shared GPDMs are indexed by switching states. Later, Deena *et al.* [2013] used SSGPDM with a variable-order Markov model on phonemes for visual speech synthesis. In order to learn the interactions between pairs of actors, a hierarchical model based on GPDMs referred as HGPDM was devised by Taubert *et al.* [2012]. Similarly, Wang [2013] incorporated the exogenous variables in GPDMs, resulting in an HGPDM which achieves improved interpretation, analysis, and prediction of human movements. Recently, Velychko *et al.* [2014] presented an approach to coupling GPDMs based on a product-of-experts, which is capable to learn different motion styles of body parts for movement design and recombine previously learned component dynamics for complex coordinated movements.

As introduced above, GPDMs and their variants are widely used in practical applications. But unfortunately, data obtained from the real world are often incomplete. For example, in medical problems, not all patients have the needed measurements. In the optical motion capture problem, parts of

data can get lost as a result of some factors such as occlusions, limited field of view, errors in the capturing process or faults in the capturing equipment. Therefore, addressing missing data in GPDMs has great significance and practical value. The existing methods for learning GPDMs were almost all designed for complete training data. When the training data are incomplete, a simple reconstruction was usually adopted before parameter updates [Wang *et al.*, 2006], which is likely to bring cumulative errors. What’s more, most literatures about GPDMs [Wang *et al.*, 2006; 2008b; Li *et al.*, 2013] only dealt with the situation in which some data subsequences are totally missing. This is just one specific situation of data incompleteness. In practice, there are different situations of data incompleteness. For example, if we use a matrix to represent multiple-output sequential data, the situations include row missing, column missing and block missing. Our work is to address missing data in GPDMs where the data incompleteness can be in any situation and the missing data can occur in the training or (and) test set. In this paper, the Bayesian framework is adopted. No matter which situation of data incompleteness, for training, Bayesian methods can be used to maximize the posterior of unknown variables conditional on the observed data. Similarly, if the test data are incomplete, the lost data can be recovered by maximizing the posterior distribution of the conditional model given the observed test data and the learned model. This idea was employed before, e.g., in reconstructing the parts of motion capture data with the variational Gaussian process dynamical system (VGPDS) [Damianou *et al.*, 2011].

In this paper, we revisit GPDMs in different situations of data incompleteness and develop four algorithms (MAP⁺, Fix. $\bar{\alpha}$ ⁺, B-GPDM⁺ and T.MAP⁺) for training GPDMs with incomplete data and a conditional model (CM⁺) for recovering the incomplete test data. The highlights of our work are summarized as follows. Firstly, the proposed approaches can fully and properly use the partially observed data while the existing work reconstructs the missing data using a function of the latent variables, which may cause cumulative errors. Therefore, the proposed approaches are promising to provide significant improvements. In our experiments on the human motion data, the lost parts of the body in the sequences are recovered more accurately, as expected. Secondly, the proposed approaches for GPDMs can handle various complex situations of data incompleteness while most exiting work on GPDMs can only deal with very simple situations. In view of all the different situations of data incompleteness, our approaches show advantages for recovering missing data.

2 Review of GPDMs

GPDMs were proposed to analyze sequential data. Let $Y = [\mathbf{y}_1, \dots, \mathbf{y}_N]^\top$ be the data in the observation space and $X = [\mathbf{x}_1, \dots, \mathbf{x}_N]^\top$ be the variables in the latent space where $\mathbf{y}_i \in \mathbb{R}^D$ and $\mathbf{x}_i \in \mathbb{R}^d$. The likelihood of Y given X is expressed as a product of GPs (one for each of the D data dimensions)

$$p(Y|X, \bar{\beta}, W) = \frac{|W|^N}{\sqrt{(2\pi)^{ND} |K_Y|^D}} \exp\left(-\frac{1}{2} \text{tr}(\mathbf{K}_Y^{-1} Y W^2 Y^\top)\right), \quad (1)$$

where W is a scaling diagonal matrix and \mathbf{K}_Y is an $N \times N$ kernel matrix constructed by a kernel function κ_Y with parameters $\bar{\beta} = \{\beta_i\}_{i=1}^3$: $\kappa_Y(\mathbf{x}, \mathbf{x}') = \beta_1 \exp(-\frac{\beta_2}{2} \|\mathbf{x} - \mathbf{x}'\|^2) + \beta_3^{-1} \delta_{\mathbf{x}, \mathbf{x}'}$. The distribution of X is given by a first-order Markov Gaussian process

$$p(X|\bar{\alpha}) = \frac{p(\mathbf{x}_1)}{\sqrt{(2\pi)^{(N-1)d} |\mathbf{K}_X|^d}} \exp\left(-\frac{1}{2} \text{tr}(\mathbf{K}_X^{-1} X_{2:N} X_{2:N}^\top)\right), \quad (2)$$

where $X_{2:N} = [\mathbf{x}_2, \dots, \mathbf{x}_N]^\top$, \mathbf{K}_X is a kernel matrix constructed from $[\mathbf{x}_1, \dots, \mathbf{x}_{N-1}]^\top$, and \mathbf{x}_1 has an isotropic Gaussian prior. The GPDM uses a ‘‘linear+RBF’’ kernel for \mathbf{K}_X with parameters $\bar{\alpha} = \{\alpha_i\}_{i=1}^4$: $\kappa_X(\mathbf{x}, \mathbf{x}') = \alpha_1 \exp(-\frac{\alpha_2}{2} \|\mathbf{x} - \mathbf{x}'\|^2) + \alpha_3 \mathbf{x}^\top \mathbf{x}' + \alpha_4^{-1} \delta_{\mathbf{x}, \mathbf{x}'}$. The priors of the kernel hyperparameters are placed with $p(\bar{\alpha}) \propto \prod_i \alpha_i^{-1}$ and $p(\bar{\beta}) \propto \prod_i \beta_i^{-1}$. Parameter W has a broad half-normal prior, $p(W) = \prod_{m=1}^D \frac{2}{\sigma \sqrt{2\pi}} \exp(-\frac{w_m^2}{2\sigma^2})$, where $w_m > 0$ corresponds to the diagonal elements of W and σ is often fixed¹.

Then, the joint probability distribution of latent variables, observations, and parameters is given by

$$p(X, Y, \bar{\alpha}, \bar{\beta}, W) = p(Y|X, \bar{\beta}, W) p(X|\bar{\alpha}) p(\bar{\alpha}) p(\bar{\beta}) p(W).$$

Note that, Y can represent multiple sequences $Y^{(1)}, \dots, Y^{(P)}$, with lengths N_1, \dots, N_P . Then $X_{2:N}$ is composed of the associated latent variables $X^{(1)}, \dots, X^{(P)}$ as $X_{2:N} = [X_{2:N_1}^{(1)\top}, \dots, X_{2:N_P}^{(P)\top}]^\top$ and $X_{1:N-1}$ is given by $X_{1:N-1} = [X_{1:N_1-1}^{(1)\top}, \dots, X_{1:N_P-1}^{(P)\top}]^\top$.

2.1 GPDM Learning

The existing learning algorithms for GPDMs include MAP, Fix. $\bar{\alpha}$, B-GPDM and T.MAP [Wang *et al.*, 2008b]. The first three algorithms are based on MAP and the fourth uses Monte Carlo EM (MCEM). MAP based methods require minimizing the joint negative log-posterior of the unknowns $-\ln p(X, \bar{\alpha}, \bar{\beta}, W|Y)$ expressed as $\mathcal{L} + \text{const}$, where const represents a constant and

$$\mathcal{L} = \mathcal{L}_Y + \mathcal{L}_X + \sum_j \ln \beta_j + \sum_j \ln \alpha_j + \frac{\text{tr}(W^2)}{2\sigma^2}, \quad (3)$$

with

$$\mathcal{L}_Y = \frac{D}{2} \ln |\mathbf{K}_Y| + \frac{1}{2} \text{tr}(\mathbf{K}_Y^{-1} Y W^2 Y^\top) - N \ln |W|, \quad (4)$$

$$\mathcal{L}_X = \frac{d}{2} \ln |\mathbf{K}_X| + \frac{1}{2} \text{tr}(\mathbf{K}_X^{-1} X_{2:N} X_{2:N}^\top) + \frac{1}{2} \mathbf{x}_1^\top \mathbf{x}_1. \quad (5)$$

MAP Parameters and latent variables are optimized through minimizing (3) with respect to W in a closed form and with respect to $\{X, \bar{\alpha}, \bar{\beta}\}$ alternately using the scaled conjugate gradient method (SCG).

Fix. $\bar{\alpha}$ Hyperparameters $\bar{\alpha}$ are fixed as $[0.009, 0.2, 0.001, 10^6]^\top$ instead of being optimized to ensure that $p(X|\bar{\alpha})$ represents a strong preference for smooth observation sequences.

B-GPDM B-GPDM was introduced by multiplying \mathcal{L}_X in

¹It is set to 10^3 in our experiments as in Wang *et al.* [2008b].

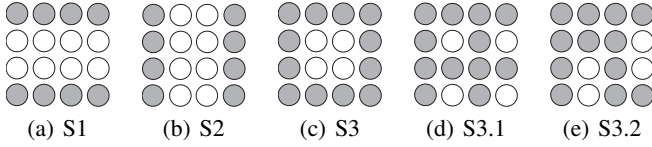


Figure 1: Illustration of data incompleteness.

Algorithm 1 MAP⁺ estimation of $\{X, \bar{\alpha}, \bar{\beta}, W\}$.

Require: Data $\{Y_{:,cd}, Y_{cn,md}\}$, integers $\{d, I, J\}$.
Initialize X_{cn} with PCA on $Y_{cn,:}$ with d dimensions.
Initialize X_{mn} using the NN² or cubic spline³ method.
Initialize $\bar{\alpha} \leftarrow (0.9, 1, 0.1, e)$, $\bar{\beta} \leftarrow (1, 1, e)$, $\{w_k\} \leftarrow 1$.

- 1: **for** $i=1$ to I **do**
- 2: **for** j in **cd** **do**
- 3: $w_j^2 \leftarrow N(Y_{:,j}^\top \mathbf{K}_Y^{-1} Y_{:,j} + \frac{1}{\sigma^2})^{-1}$
- 4: **end for**
- 5: **for** j in **md** **do**
- 6: $w_j^2 \leftarrow N_c(Y_{cn,j}^\top \mathbf{K}_{Y_{cn}}^{-1} Y_{cn,j} + \frac{1}{\sigma^2})^{-1}$
- 7: **end for**
- 8: $\{X, \bar{\alpha}, \bar{\beta}\} \leftarrow$ optimize (7) using SCG for J iterations.
- 9: **end for**

(3) by a coefficient $\frac{D}{d}$ to balance the influences of the observation reconstruction error in the high dimensional space and the prediction error in the low dimensional latent space before optimization. As a result, B-GPDM adapts the objective function (3) to favor smooth latent variable sequences.

T.MAP Unlike the previous learning algorithms, T.MAP uses MCEM to optimize $\Theta = \{\bar{\alpha}, \bar{\beta}, W\}$ and MAP to optimize X . In the E-step of MCEM, the expected complete negative log likelihood $-\ln p(Y, X|\Theta)$ under $p(X|Y, \Theta^i)$ is approximately computed by $\mathcal{L}_E(\Theta) \approx -\frac{1}{R} \sum_{r=1}^R \ln p(Y, X^r|\Theta)$, where $\{X^r\}_{r=1}^R \sim p(X|Y, \Theta^i)$ are sampled using hybrid Monte Carlo (HMC). In the M-step, the hyperparameters Θ^{i+1} are optimized by minimizing $\mathcal{L}_E(\Theta)$. As the second stage, given the optimized Θ^{i+1} , X is optimized through maximizing $\ln p(X, \Theta^{i+1}|Y)$ with SCG.

2.2 Conditional GPDM

Given the learned GPDM $\Gamma = \{Y, X, \bar{\alpha}, \bar{\beta}, W\}$, which includes the observations, learned latent variables and parameters, one can optimize the latent variables corresponding to a new observation sequence with a conditional GPDM (CM). In addition, one can predict or generate a new sequence only with an initial value \mathbf{x}_1 .

In the CM, the distribution over a new sequence $Y^* \in \mathbb{R}^{M \times D}$ and its associated latent variable $X^* \in \mathbb{R}^{M \times d}$ conditional on Γ is given by

$$p(Y^*, X^*|\Gamma) = p(Y^*|X^*, \Gamma)p(X^*|\Gamma). \quad (6)$$

²NN is short for 1-nearest neighbor where for each frame the values of known dimensions are compared to each example in the present data to find the closest match, which is used in S2 and S3.

³Cubic spline [Wang *et al.*, 2008b] is a kind of interpolation method relying on the data before and after the missing data, which is used in S1.

Algorithm 2 T.MAP⁺ estimation of $\{X, \bar{\alpha}, \bar{\beta}, W\}$.

Require: Data $\{Y_{:,cd}, Y_{cn,md}\}$. Integers $\{d, R, I, J, K\}$.
Initialize X_{cn} with PCA on $Y_{cn,:}$ with d dimensions.
Initialize X_{mn} using the NN or cubic spline method.
Initialize $\bar{\alpha} \leftarrow (0.9, 1, 0.1, e)$, $\bar{\beta} \leftarrow (1, 1, e)$, $\{w_k\} \leftarrow 1$.

- 1: **for** $i=1$ to I **do**
- 2: Generate $\{X^r\}_{r=1}^R \sim p(X|Y_{:,cd}, Y_{cn,md}, \bar{\alpha}, \bar{\beta}, W)$ using HMC sampling.
- 3: Construct $\{\mathbf{K}_Y^r, \mathbf{K}_{Y_{cn}}^r, \mathbf{K}_X^r\}$ from $\{X^r\}_{r=1}^R$.
- 4: **for** $j = 1$ to J **do**
- 5: **for** k in **cd** **do**
- 6: $w_j^2 \leftarrow N(Y_{:,k}^\top \frac{1}{R} \sum_{r=1}^R (\mathbf{K}_Y^r)^{-1} Y_{:,k} + \frac{1}{\sigma^2})^{-1}$
- 7: **end for**
- 8: **for** k in **md** **do**
- 9: $w_j^2 \leftarrow N_c(Y_{cn,k}^\top \frac{1}{R} \sum_{r=1}^R (\mathbf{K}_{Y_{cn}}^r)^{-1} Y_{cn,k} + \frac{1}{\sigma^2})^{-1}$
- 10: **end for**
- 11: $\{\bar{\alpha}, \bar{\beta}\} \leftarrow$ optimize $\tilde{\mathcal{L}}_E(\Theta)$ using SCG for K iterations.
- 12: **end for**
- 13: $X \leftarrow$ maximize $\ln p(X, \bar{\alpha}, \bar{\beta}, W|Y_{:,cd}, Y_{cn,md})$.
- 14: **end for**

Here, $p(X^*|\Gamma)$ is calculated as

$$p(X^*|\Gamma) = \frac{p(\mathbf{x}_1^*)}{\sqrt{(2\pi)^{(M-1)d} |\mathbf{K}_{X^*}|}} \exp\left(-\frac{1}{2} \text{tr}(\mathbf{K}_{X^*} Z_X Z_X^\top)\right),$$

where $Z_X = X_{2:N}^* - C^\top \mathbf{K}_X^{-1} X_{2:N}$ and $\mathbf{K}_{X^*} = D - C^\top \mathbf{K}_X^{-1} C$. $(C)_{ij} = \kappa_X(\mathbf{x}_i, \mathbf{x}_j^*)$ and $(D)_{ij} = \kappa_X(\mathbf{x}_i^*, \mathbf{x}_j^*)$ are the elements of the $(N-P) \times (M-1)$ and $(M-1) \times (M-1)$ kernel matrices, respectively. $p(Y^*|X^*, \Gamma)$ is calculated as

$$\begin{aligned} p(Y^*|X^*, \Gamma) &= \frac{|W|^M}{\sqrt{(2\pi)^{MD} |\mathbf{K}_{Y^*}|}} \exp\left(-\frac{1}{2} \text{tr}(\mathbf{K}_{Y^*}^{-1} Z_Y W^2 Z_Y^\top)\right), \end{aligned}$$

where $Z_Y = Y^* - A^\top \mathbf{K}_Y^{-1} Y$ and $\mathbf{K}_{Y^*} = B - A^\top \mathbf{K}_Y^{-1} A$. $(A)_{ij} = \kappa_Y(\mathbf{x}_i, \mathbf{x}_j^*)$ and $(B)_{ij} = \kappa_Y(\mathbf{x}_i^*, \mathbf{x}_j^*)$ are the elements of the $N \times M$ and $M \times M$ kernel matrices, respectively.

To optimize the latent variables associated with new sequences, one can obtain X_* by maximizing (6). To predict or generate a new sequence given \mathbf{x}_1 , one can first set the latent variable at each time step to be the most likely (mean) value given the previous set as $\mu_X(\mathbf{x}_{t-1}) = \mathbf{k}_X(\mathbf{x}_{t-1})^\top \mathbf{K}_X^{-1} X_{2:N}$. Then reconstruct the new data in the observation space by the mean function $\mu_Y(\mathbf{x}^*) = \mathbf{k}_Y(\mathbf{x}^*)^\top \mathbf{K}_Y^{-1} Y$.

2.3 GPDM with Incomplete Data

It is indeed possible for GPDMs to deal with missing data [Wang *et al.*, 2008b]. However, no matter for incomplete training or test data, these methods use the mean function $\mu_Y(X)$ introduced above to reconstruct the missing data. This reconstruction procedure would be performed before optimizing (3) or (6), respectively. Moreover, the reconstruction uses the latent variables obtained by some initialization methods. Thus the performance of these methods is quite limited and large cumulative errors may be incurred.

3 Our Methods

We summarize data incompleteness into three situations, i.e., row missing (S1), column missing (S2) and block missing

	07.01	35.01	35.18	64.01	four-walker
NN	15.72 / .	7.24 / .	40.70 / .	24.06 / .	14.00 / .
MAP	15.72 / 59.89	6.57 / 2.50	33.87 / 3.00	20.76 / 0.66	17.34 / 9.59
Fix. $\bar{\alpha}$	15.34 / 88.70	6.42 / 2.00	35.17 / 3.32	19.48 / 0.48	17.81 / 6.13
B-GPDM	14.45 / 5.42	5.97 / 0.57	27.01 / 0.31	17.07 / 0.22	13.47 / 0.24
T.MAP	14.91 / 19.79	6.89 / 2.57	23.15 / 1.74	14.90 / 1.74	12.15 / 10.19
MAP ⁺	14.33 / 3.71	6.26 / 0.07	24.99 / 0.27	12.80 / 0.14	13.91 / 2.72
Fix. $\bar{\alpha}$ ⁺	15.34 / 2.49	6.08 / 0.10	28.92 / 0.42	15.27 / 0.16	16.69 / 2.07
B-GPDM ⁺	14.38 / 1.47	6.51 / 0.27	26.68 / 0.26	16.06 / 0.17	15.43 / 0.19
T.MAP ⁺	15.37 / 4.38	6.62 / 0.37	18.00 / 0.20	14.68 / 0.34	10.77 / 2.44

Table 1: Averaged RMSE / $-LP$ of recovering missing training data (in S3).

(S3) in Figures 1(a), 1(b) and 1(c), where the gray nodes are observed and the white are unobserved. We first give the definitions of some symbols: \mathbf{mn} with length N_m represents the time indices of missing data, and \mathbf{md} with length D_m represents the dimension indices of missing data. Correspondingly, \mathbf{cn} and \mathbf{cd} with lengths N_c and D_c are the complements of \mathbf{mn} and \mathbf{md} in the whole indices $[1 : N]$ and $[1 : D]$, respectively. Thus, the missing data can be expressed as $Y_{\mathbf{mn},\mathbf{md}}$ and the present data can be expressed as $Y_{\mathbf{cn},\mathbf{md}}, Y_{:, \mathbf{cd}}$. In particular, Figure 1(a) can be expressed as row missing; Figure 1(b) can be expressed as column missing; Figure 1(c) can be expressed as block missing in which $\mathbf{mn} = [2, 3]$, $\mathbf{md} = [2, 3]$, $\mathbf{cn} = [1, 4]$ and $\mathbf{cd} = [1, 4]$.

It is worth noting that Figures 1(a) and 1(b) are two specific cases of Figure 1(c), and Figure 1(c) can be extended to any situations of data incompleteness. A complex situation is that the missing data are non-contiguous over time or dimensions. We divide this case of data incompleteness into two situations as in Figures 1(d) and 1(e), and note that a) if the missing data can be formed into a matrix, the methods for reconstruction are the same as the methods for addressing missing data that are contiguous over time and dimensions; b) otherwise, the missing data should be divided into blocks. We first give methods for both training and testing with incomplete data in the case of S3. After that, we will explain how to adapt our methods to the situation of non-contiguously missing.

3.1 GPDM with Incomplete Training Data

The GPDMs are defined as before, and now suppose the training data are incomplete as in Figure 1(c). According to the Bayesian framework, the joint distribution of the latent variables, the observed data and the parameters are given by

$$p(X, Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}}, \bar{\alpha}, \bar{\beta}, W) = p(Y_{:, \mathbf{cd}} | X, \bar{\beta}, W_{\mathbf{cd}}) p(Y_{\mathbf{cn}, \mathbf{md}} | X_{\mathbf{cn}}, \bar{\beta}, W_{\mathbf{md}}) p(X | \bar{\alpha}) p(\bar{\alpha}) p(\bar{\beta}) p(W),$$

where $X_{\mathbf{cn}}$ means the \mathbf{cn} th rows of X and $W_{\mathbf{md}}$ represents a diagonal matrix whose diagonal elements are the \mathbf{md} th diagonal elements of W . In this case we propose four new learning methods for GPDM training, which are denoted MAP⁺, Fix. $\bar{\alpha}$ ⁺, B-GPDM⁺ and T.MAP⁺. In MAP based methods, the GPDM with incomplete data is learned through minimizing the joint negative log-posterior of the unknowns $-\ln p(X, \bar{\alpha}, \bar{\beta}, W | Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}})$ which is given, up to an additive constant, by

$$\begin{aligned} \tilde{\mathcal{L}} = & \mathcal{L}_{Y_{:, \mathbf{cd}}} + \mathcal{L}_{Y_{\mathbf{cn}, \mathbf{md}}} + \mathcal{L}_X + \sum_j \ln \beta_j \\ & + \frac{\text{tr}(W^2)}{2\sigma^2} + \sum_j \ln \alpha_j, \end{aligned} \quad (7)$$

with

$$\mathcal{L}_{Y_{:, \mathbf{cd}}} = \frac{D_c}{2} \ln |\mathbf{K}_Y| + \frac{1}{2} \text{tr}(\mathbf{K}_Y^{-1} Y_{:, \mathbf{cd}} W_{\mathbf{cd}}^2 Y_{:, \mathbf{cd}}^\top) - N \ln |W_{\mathbf{cd}}|, \quad (8)$$

$$\begin{aligned} \mathcal{L}_{Y_{\mathbf{cn}, \mathbf{md}}} = & \frac{1}{2} \text{tr}(\mathbf{K}_{Y_{\mathbf{cn}}}^{-1} Y_{\mathbf{cn}, \mathbf{md}} W_{\mathbf{md}}^2 Y_{\mathbf{cn}, \mathbf{md}}^\top) \\ & + \frac{D_m}{2} \ln |\mathbf{K}_{Y_{\mathbf{cn}}}| - N_c \ln |W_{\mathbf{md}}|. \end{aligned} \quad (9)$$

Here, \mathcal{L}_X has the same formulation as (5). We give the details for MAP⁺, Fix. $\bar{\alpha}$ ⁺, B-GPDM⁺ and T.MAP⁺ below.

MAP⁺ Parameters are optimized through minimizing (7) with respect to W in a closed form and with respect to $\{X, \bar{\alpha}, \bar{\beta}\}$ alternately using SCG. The detailed procedure is described in Algorithm 1. We set $d = 3$, $I = 100$ and $J = 10$ in our experiments.

Fix. $\bar{\alpha}$ ⁺ Hyperparameters $\bar{\alpha}$ in (7) are fixed as $[0.009, 0.2, 0.001, 10^6]^\top$ instead of being optimized.

B-GPDM⁺ Similar to B-GPDM, \mathcal{L}_X in (7) is multiplied by a coefficient $\frac{D}{d}$ to balance the objective terms in the GPDM.

T.MAP⁺ T.MAP⁺ has the same principle as T.MAP, which uses MCEM to optimize $\Theta = \{\bar{\alpha}, \bar{\beta}, W\}$ and MAP to optimize X . Since the data are now incomplete, some calculations are different. In the E-step of MCEM, the expected joint negative log-likelihood $-\ln p(Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}}, X | \Theta)$ under $p(X | Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}}, \Theta^i)$ is approximately computed by $\tilde{\mathcal{L}}_{\mathcal{E}}(\Theta) \approx -\frac{1}{R} \sum_{r=1}^R \ln p(Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}}, X^r | \Theta)$, where $\{X^r\}_{r=1}^R \sim p(X | Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}}, \Theta^i)$ are sampled using HMC. In the M-step, the hyperparameters Θ^{i+1} are optimized by minimizing $\tilde{\mathcal{L}}_{\mathcal{E}}(\Theta)$. As the second stage, given the optimized Θ^{i+1} , X is optimized through maximizing $\ln p(X, \Theta^{i+1} | Y_{:, \mathbf{cd}}, Y_{\mathbf{cn}, \mathbf{md}})$ by SCG. The detailed procedure is given in Algorithm 2. We set $d = 3$, $R = 50$, $I = 10$, $J = 10$ and $K = 10$ in our experiments.

Comparing Algorithms 1 and 2 with the existing algorithms [Wang *et al.*, 2008b], we find that the proposed methods only double the computational complexities in the worst case. Moreover, reconstructions for missing data before parameter updates disappear in the proposed methods, which will reduce the time cost. Further, when data are incomplete as in S1, the proposed methods would be more efficient.

3.2 GPDM with Incomplete Test Data

We present a new conditional model (CM⁺) to recover the missing test data $Y_{\mathbf{mn}, \mathbf{md}}^* \in \mathbb{R}^{Mm \times Dm}$. In the CM⁺, defining $\tilde{Y} = [Y^\top, Y_{\mathbf{cn}, :}^*]^\top$, $\tilde{Y}^* = Y_{\mathbf{mn}, :}^*$, $\tilde{X} = [X^\top, X_{\mathbf{cn}}^*]^\top$ and

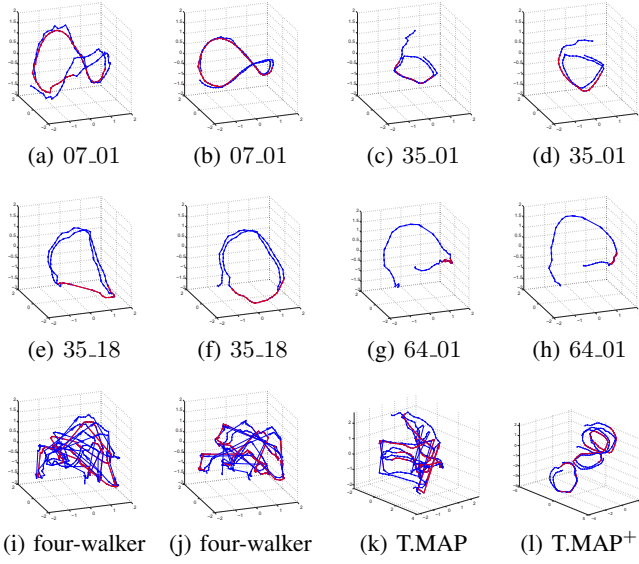


Figure 2: 3D trajectories in the latent space by training GPDMs with incomplete data. (Best viewed in color).

$\tilde{X}^* = X_{mn}^*$, the posterior density of $Y_{mn,md}^*$ is

$$p(Y_{mn,md}^* | X^*, Y_{:,cd}^*, Y_{cn,md}^*, \Gamma) = \frac{|W|^{M_m}}{\sqrt{(2\pi)^{M_m D_m} |\mathbf{K}_{\tilde{Y}^*}|^{D_m}}} \exp\left(-\frac{1}{2} \text{tr}(\mathbf{K}_{\tilde{Y}^*}^{-1} Z_{\tilde{Y}} W^2 Z_{\tilde{Y}}^\top)\right),$$

where $\tilde{Z}_{\tilde{Y}} = Y_{mn,md}^* - \tilde{A}^\top \mathbf{K}_{\tilde{Y}^*}^{-1} \tilde{Y}_{:,md}$ and $\mathbf{K}_{\tilde{Y}^*} = \tilde{B} - \tilde{A}^\top \mathbf{K}_{\tilde{Y}^*}^{-1} \tilde{A}$. $(\tilde{A})_{ij} = \kappa_Y(\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j^*)$ and $(\tilde{B})_{ij} = \kappa_Y(\tilde{\mathbf{x}}_i^*, \tilde{\mathbf{x}}_j^*)$ are the elements of the $(N + M_c) \times M_m$ and $M_m \times M_m$ kernel matrices, respectively. Different from the previous CM, the optimal value of X^* is obtained by maximizing the joint conditional distribution of observed test data $Y_{:,cd}^*, Y_{cn,md}^*$ and the associated latent variable X^* given the learned model Γ

$$p(Y_{:,cd}^*, Y_{cn,md}^*, X^* | \Gamma) = p(Y_{:,cd}^*, X^* | Y_{:,cd}, X, \bar{\beta}, W) p(Y_{cn,md}^*, X_{cn}^* | Y_{:,md}, X, \bar{\beta}, W) p(X^* | \bar{\alpha}). \quad (10)$$

With X^* optimized, the mean function $\mu_{\tilde{Y}}(\tilde{X}^*) = \mathbf{k}_{\tilde{Y}}(\tilde{X}^*)^\top \mathbf{K}_{\tilde{Y}^*}^{-1} \tilde{Y}$ is used for recovering the lost test data.

3.3 Non-contiguously Missing

Recall that Figure 1(c) can be extended to more complex situations in which the missing data are non-contiguous over time or dimensions as exhibited in Figures 1(d) and 1(e). Our approaches can be applied to these situations with some reformulations or minor modifications. Specifically, for S3.1 the approaches presented above do not require modifications but with $\mathbf{md} = [2, 4]$, $\mathbf{mn} = [2, 4]$, $\mathbf{cd} = [1, 3]$ and $\mathbf{cn} = [1, 3]$. For S3.2, the missing data can be expressed as $\mathbf{md}_1 = [2]$, $\mathbf{mn}_1 = [3, 4]$, $\mathbf{md}_2 = [4]$ and $\mathbf{mn}_2 = [2, 3]$. The complement indices are: $\mathbf{cn}_1 = [1, 2]$, $\mathbf{cn}_2 = [1, 4]$ and $\mathbf{cd} = [1, 3]$. Note that \mathbf{cd} includes the complements of the union set of \mathbf{md}_1 and \mathbf{md}_2 in the whole indices $[1 : D]$. Since data on

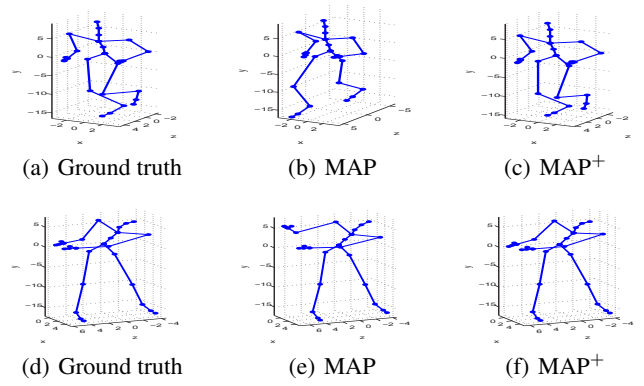


Figure 3: Recovered skeletons of run and swing with MAP and MAP+.

different dimensions are conditionally independent, the corresponding likelihood of $Y_{cn,md}$ is replaced by the product of the likelihoods of Y_{cn_1,md_1} and Y_{cn_2,md_2} . We then only need to adjust (7) and (10) accordingly.

4 Experiments

The benchmark data used for experiments are human motion capture data from the Carnegie Mellon University motion capture database. As in Wang *et al.* [2008b], we use a specific version of the default skeleton in the database for which each pose is 50-dimensional. To evaluate the effectiveness of the proposed methods, two different types of experiments have been performed: training with incomplete data and recovering incomplete test data.

4.1 Training with Incomplete Data

We evaluate the proposed four learning algorithms for handling incomplete training data. The data used include two single-walker sequences 07_01.amc (1-2-260/50-100/1-38)⁴ and 35_01.amc (55-4-338/41-60/7-31), a single-runner sequence 35_18.amc (1-2-160/21-50/23-48), a single-swinger sequence 64_01.amc (120-4-400/16-40/7-31) and a four-walker sequence spliced by 35_02.amc (55-4-338/16-40/1-38), 10_04.amc (222-4-499/16-40/1-38), 12_01.amc (22-4-328/16-40/1-38) and 16_15.amc (62-4-342/16-40/1-38). In a word, these incomplete data are all in S3. Note that, the missing data in the four-walker sequence are non-contiguous over time. For each kind of motion data, we perform the four proposed learning algorithms MAP+, Fix. $\bar{\alpha}$ +, B-GPDM+ and T.MAP+ compared with MAP, Fix. $\bar{\alpha}$, B-GPDM and T.MAP, as well as NN. Moreover, we repeat all the experiments on each motion data for nine times with different missing data. That is, \mathbf{mn} and \mathbf{md} slide on three different windows, respectively. For example, for 07_01, $\mathbf{mn} = [50 : 100]$, $[51 : 101]$ or $[52 : 102]$ and $\mathbf{md} = [1 : 38]$, $[2 : 39]$ or $[3 : 40]$.

Table 1 shows the averaged root mean square error (RMSE) per missing frame and the averaged negative log-

⁴“1-2-260/50-100/1-38” means that the data are downsampled from frames 1 to 260 by a factor of 2, with $\mathbf{mn} = [50 : 100]$ and $\mathbf{md} = [1 : 38]$.

	35_03	12_02	16_21	12_03	07_01	07_02	08_01	08_02	AVG.
k -NN	40.64 / -	40.36 / -	49.91 / -	50.45 / -	78.81 / -	64.40 / -	76.48 / -	77.06 / -	59.76 / -
VGPDS	49.92 / 12.29	45.72 / 10.84	36.00 / 7.47	43.20 / 10.32	75.74 / 24.24	73.23 / 23.37	88.78 / 14.75	71.92 / 12.29	60.56 / 14.45
MAP+CM	37.26 / 3.06	43.42 / 78.44	36.18 / 6.11	46.55 / 3.46	74.33 / 5.41	68.05 / 2.87	85.61 / 4.95	64.77 / 17.88	57.02 / 15.27
Fix. $\bar{\alpha}$ +CM	51.58 / 4.98	50.24 / 5.31	39.96 / 2.77	56.69 / 7.24	82.55 / 7.75	80.32 / 7.01	96.20 / 10.79	65.65 / 8.12	65.40 / 6.75
B-GPDM+CM	46.72 / 4.71	50.15 / 7.74	42.33 / 3.83	53.57 / 8.14	83.02 / 13.40	76.08 / 12.02	90.04 / 14.66	65.40 / 7.45	63.41 / 8.99
T.MAP+CM	57.04 / 2.26	59.60 / 2.62	46.96 / 8.13	58.59 / 3.19	86.81 / 6.97	88.56 / 6.44	93.93 / 8.25	66.20 / 19.59	69.71 / 7.18
MAP+CM ⁺	34.56 / 1.10	43.34 / 1.45	36.32 / 1.85	42.92 / 1.65	71.51 / 2.50	66.53 / 1.16	81.18 / 2.19	67.75 / 6.12	55.52 / 2.25
Fix. $\bar{\alpha}$ +CM ⁺	50.08 / 5.02	49.99 / 5.35	40.17 / 2.79	56.82 / 7.22	82.66 / 7.58	79.49 / 7.15	95.63 / 11.05	65.77 / 8.16	65.08 / 6.79
B-GPDM+CM ⁺	46.72 / 4.71	50.15 / 7.74	42.33 / 3.83	53.57 / 8.14	83.02 / 13.40	76.08 / 12.02	90.04 / 14.66	65.40 / 7.45	63.41 / 8.99
T.MAP+CM ⁺	56.86 / 1.06	57.88 / 1.60	48.77 / 2.90	60.24 / 1.92	85.03 / 1.87	83.29 / 2.00	94.88 / 4.74	70.05 / 7.45	69.63 / 2.94

Table 2: Averaged RMSE / $-LP$ of recovering missing test data (in S2).

posterior ($-LP$) over the missing data. All the shown $-LP$ s in this paper are the actual $-LP$ s divided by 100. The best results are marked in bold for each motion sequence. Through the table, we see that the lowest RMSE and $-LP$ are almost all obtained by the proposed algorithms. Further, the four proposed algorithms outperform MAP, Fix. $\bar{\alpha}$, B-GPDM and T.MAP on most sequences, respectively. Note that, since the data in some frames are not completely lost, the present data on some known dimensions will help to learn the model. In this case, the dynamics of the model are not the unique factor to influence the performance of the model. This explains why Fix. $\bar{\alpha}$ ⁺, B-GPDM⁺ and T.MAP⁺ which favor smooth latent trajectories do not always perform best.

Figure 2 shows the 3D latent trajectories learned from incomplete data, in which from (a) to (j), the couples of figures with the same name are the 3D trajectories of the same data learned by MAP on the left and MAP⁺ on the right, respectively. Figures 2(k) and 2(l) are the 3D trajectories of the four-walker motion data learned by T.MAP and T.MAP⁺, respectively. The red line corresponds to the missing frames and the blue line corresponds to the present frames. From the figure we can find that the latent trajectories learned through MAP⁺ are much smoother than those through MAP. This is because MAP⁺ makes full use of the data on the known dimensions while learning the dynamics of the latent trajectories. However, the simple reconstruction approach in MAP makes it rely on the old latent variables too much, which leads to cumulative errors. In Figures 2(i) and 2(j), the motion data combining four walkers are not well learned by MAP or MAP⁺, which is consistent to the conclusions in Wang *et al.* [2008b]. In addition, Wang *et al.* [2008b] showed that for complete data the walk cycles from the four subjects learned with T.MAP are smooth and separated in the latent space. However, when the data are incomplete (as shown in Figures 2(k) and 2(l)), T.MAP cannot achieve the same effects while T.MAP⁺ can.

In order to show the correctness of the recovery more intuitively, we exhibit the recovered skeletons of run and swing on a certain frame with MAP and MAP⁺ in Figure 3. Obviously, the legs of the runner and the right arm of the swinger are not recovered well by MAP. But, the skeletons recovered by MAP⁺ and the ground truth are almost the same.

4.2 Recovering Incomplete Test Data

Now we consider the tasks of filling in lost parts of new data in two situations (S1 and S2) of data incompleteness. We perform the proposed conditional model CM⁺ and the existing

Complete training data		Incomplete training data	
Methods	AVG.	Methods	AVG.
k -NN	50.46 / -	k -NN	70.01 / -
spline	122.24 / -	spline	122.24 / -
MAP+CM	53.69 / 230.73	MAP+CM	57.13 / 847.46
Fix. $\bar{\alpha}$ +CM	46.55 / 3.54	Fix. $\bar{\alpha}$ +CM	56.72 / 3.43
B-GPDM+CM	51.52 / 4.50	B-GPDM+CM	52.63 / 6.42
T.MAP+CM	50.90 / 158.17	T.MAP+CM	51.26 / 505.02
MAP+CM ⁺	51.73 / 186.19	MAP ⁺ +CM ⁺	51.83 / 229.88
Fix. $\bar{\alpha}$ +CM ⁺	46.54 / 3.53	Fix. $\bar{\alpha}$ ⁺ +CM ⁺	50.87 / 2.66
B-GPDM+CM ⁺	51.42 / 4.41	B-GPDM ⁺ +CM ⁺	52.91 / 3.69
T.MAP+CM ⁺	51.04 / 70.56	T.MAP ⁺ +CM ⁺	49.05 / 107.96

Table 3: Averaged RMSE / $-LP$ of recovering missing test data (in S1).

conditional model CM to recover the missing test data. The learned GPDMs on the four-walker motion data using different learning algorithms are employed. For clarity, we define the approaches as “learning algorithm + CM⁺” and “learning algorithm + CM”.

First we consider the column missing case. Eight 50-frame sequences with 40 dimensions removed are taken to be recovered. The training data used here are complete so that the learning algorithms for GPDMs combined with CM⁺ are the same to those for GPDMs combined with CM. The experiments are performed for three times with the same setting but with $\mathbf{md} = [1 : 40]$, $[2 : 41]$ and $[3 : 42]$ for each time, respectively. Table 2 gives the averaged results in terms of the RMSE of recovering and $-LP$ of missing data over three experiments. Besides GPDM based methods, VGPDS and k -NN are also employed for comparisons. Here, k is set to 15 as in Wang *et al.* [2008b] where they tried $k = [3, 6, 9, 15, 20]$, with 15 giving the lowest average error. We do not include spline which is only suitable in S1. The VGPDS which assumes that sequences are independent cannot get as good performance as GPDMs on the current data. The proposed CM⁺ recovers the missing parts more accurately than the existing CM. Moreover, MAP+CM⁺ performs best because the current situation of data incompleteness is that data on some dimensions are missing from the start to the end of a sequence: Other algorithms relying too much on the dynamics of the model cannot help to recover missing data well in this situation.

In order to verify the versatility of our methods, we also conduct experiments on recovering test data with some interval frames missing. In this situation, we consider two cases in which the training data are complete and incomplete, respectively. For incomplete training data, GPDMs are learned by the existing and proposed algorithms, respectively. VGPDS

which doesn't handle this situation is not compared. Experiments on the same sequences as Table 2 are performed, in which missing frames are [5 : 35]. Due to the limited space, the averaged RMSE and $-LP$ over all the test sequences are presented in Table 3. $\text{Fix.}\bar{\alpha}+\text{CM}^+$ and $\text{T.MAP}^++\text{CM}^+$ perform best with complete and incomplete training data, respectively. The methods which favor the dynamics perform better than $\text{MAP}+\text{CM}^+$ because data in all dimensions before and after the missing frames are given in these experiments.

5 Conclusions

In this paper, we have proposed four learning algorithms (MAP^+ , $\text{Fix.}\bar{\alpha}^+$, B-GPDM^+ and T.MAP^+) for training GPDMs with incomplete training data and a condition model (CM^+) for recovering incomplete test data. The approaches were developed under the Bayesian framework [Sun, 2013]. The advantages of the proposed approaches have been demonstrated for different situations of data incompleteness.

Acknowledgments

The corresponding author Shiliang Sun would like to thank supports from the National Natural Science Foundation of China under Project 61370175, and Shanghai Knowledge Service Platform Project (No. ZF1213).

References

- [An *et al.*, 2012] B. An, H. K. Park, and F. C. Park. Grasp motion learning with Gaussian process dynamic models. In *Proceedings of IEEE International Conference on Automation Science and Engineering*, pages 1114–1119, 2012.
- [Chen *et al.*, 2009] J. Chen, M. Kim, Y. Wang, and Q. Ji. Switching Gaussian process dynamic models for simultaneous composite motion tracking and recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 2655–2662, 2009.
- [Dallaire *et al.*, 2009] P. Dallaire, C. Besse, S. Ross, and B. Chaib-draa. GP-POMDP: Bayesian reinforcement learning in continuous POMDPs with Gaussian processes. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2604–2609, 2009.
- [Damianou *et al.*, 2011] A. C. Damianou, M. K. Titsias, and N. D. Lawrence. Variational Gaussian process dynamical systems. *Advances in Neural Information Processing Systems*, 24:2510–2518, 2011.
- [Deena and Galata, 2009] S. Deena and A. Galata. Speech-driven facial animation using a shared Gaussian process latent variable model. In *Proceedings of the 5th International Symposium on Advances in Visual Computing*, pages 89–100, 2009.
- [Deena *et al.*, 2013] S. Deena, S. Hou, and A. Galata. Visual speech synthesis using a variable-order switching shared Gaussian process dynamical model. *IEEE Transactions on Multimedia*, 15:1755–1768, 2013.
- [Gamage *et al.*, 2011] N. Gamage, T. C. Kuang, R. Akmeliawati, and S. Demidenko. Gaussian process dynamical models for hand gesture interpretation in sign language. *Pattern Recognition Letters*, 32:2009–2014, 2011.
- [Henter *et al.*, 2012] G. E. Henter, M. R. Frean, and W. B. Kleijn. Gaussian process dynamical models for nonparametric speech representation and synthesis. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4505–4508, 2012.
- [Li *et al.*, 2013] W. Li, J. Sun, X. Zhang, and Y. Wu. Spatial constraints-based maximum likelihood estimation for human motions. In *Proceedings of IEEE International Conference on Signal Processing, Communication and Computing*, pages 1–6, 2013.
- [Park and Yoo, 2011] H. Park and C. D. Yoo. Gaussian process dynamical models for phoneme classification. In *Proceedings of Neural Information Processing System Workshop on Bayesian Nonparametrics: Hope or Hype*, pages 1–2, 2011.
- [Raskin *et al.*, 2008] L. Raskin, E. Rivlin, and M. Rudzsky. Using Gaussian process annealing particle filter for 3D human tracking. *EURASIP Journal on Advances in Signal Processing*, 2008:1–13, 2008.
- [Sun, 2013] S. Sun. A review of deterministic approximate inference techniques for Bayesian machine learning. *Neural Computing and Applications*, 23:2039–2050, 2013.
- [Taubert *et al.*, 2012] N. Taubert, A. Christensen, D. Endres, and M. A. Giese. Online simulation of emotional interactive behaviors with hierarchical Gaussian process dynamical models. In *Proceedings of the ACM Symposium on Applied Perception*, pages 25–32, 2012.
- [Urtasun *et al.*, 2006] R. Urtasun, D. J. Fleet, and P. Fua. 3D people tracking with Gaussian process dynamic models. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 238–245, 2006.
- [Urtasun *et al.*, 2007] R. Urtasun, D. J. Fleet, and N. D. Lawrence. Modeling human locomotion with topologically constrained latent variable models. In *Proceedings of IEEE International Conference on Computer Vision Workshop on Human Motion: Understanding, Modeling, Capture and Animation*, pages 104–118, 2007.
- [Urtasun *et al.*, 2008] R. Urtasun, D. J. Fleet, A. Geiger, J. Popović, T. J. Darrell, and N. D. Lawrence. Topologically-constrained latent variable models. In *Proceedings of the 25th International Conference on Machine Learning*, pages 1080–1087, 2008.
- [Velychko *et al.*, 2014] D. Velychko, D. Endres, N. Taubert, and M. A. Giese. Coupling Gaussian process dynamical models with product-of-experts kernels. In *Proceedings of the 24th International Conference on Artificial Neural Networks*, pages 1–9, 2014.
- [Wang *et al.*, 2006] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models. *Advances in Neural Information Processing Systems*, 19:1441–1448, 2006.
- [Wang *et al.*, 2008a] J. Wang, Y. Yin, and H. Man. Multiple human tracking using particle filter with Gaussian process dynamical model. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008.
- [Wang *et al.*, 2008b] J. M. Wang, D. J. Fleet, and A. Hertzmann. Gaussian process dynamical models for human motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30:283–398, 2008.
- [Wang, 2013] Z. Wang. *Intention Inference and Decision Making with Hierarchical Gaussian Process Dynamics Models*. PhD thesis, Darmstadt University of Technology, 2013.