Gaussian Process Motion Graph Models for Smooth Transitions among Multiple Actions

Norimichi Ukita†[1] and Takeo Kanade‡

†Graduate School of Information Science, Nara Institute of Science and Technology
‡The Robotics Institute, Carnegie Mellon University

Abstract

We propose a unified model for human motion prior with multiple actions. Our model is generated from sample pose sequences of the multiple actions, each of which is recorded from real human motion. The sample sequences are connected to each other by synthesizing a variety of possible transitions among the different actions. For kinematically-realistic transitions, our model integrates nonlinear probabilistic latent modeling of the samples and interpolation-based synthesis of the transition paths. While naive interpolation makes unexpected poses, our model rejects them 1) by searching for smooth and short transition paths by employing the good properties of the observation and latent spaces and 2) by avoiding using samples that unexpectedly synthesize the non-smooth interpolation. The effectiveness of the model is demonstrated with real data and its application to human pose tracking.

Keywords:
Human pose, Multiple actions, Motion prior, Gaussian process, Motion graphs

1. Introduction

Motion prior is widely used in human pose tracking, simulation, and synthesis for accuracy and robustness. Motion prior is obtained from samples of real human motion. Motion datasets including the sample sequences have been distributed for motion modeling and evaluation in Computer Vision[1] and Graphics[2, 3] communities. Different kinds of actions (e.g. walking, jogging, dance) were recorded independently in these datasets. The motion model of *each action* can be leveraged for analyzing that action.

For efficiently and adaptively using motion prior of *multiple actions*, a unified motion model of these actions is useful. Different actions are smoothly transited from one to another (e.g. from walking to jogging) in a natural scenario, while they are recorded independently in motion datasets. Since it is not practical to record a huge variety of possible transitions among all of the different actions, modeling the smooth transition is important. Smooth transition paths between the different actions enable successful pose modeling over the different actions.

This paper proposes a human motion model for smooth transitions among elemental actions in dataset and its application to pose tracking. After introducing related work (Sec. 2) and existing models for motion modeling and interpolation (Secs. 3 and 4), Sec. 5 reveals the problems of naive integration of the existing models. Sec. 6 describes the proposed model. Experimental results of pose tracking with the proposed model are presented in Sec. 7, and we conclude the paper in Sec. 8.

2. Related Work

The pose of a human body is modeled by a set of joint positions/angles, which can be measured by a motion capture system. The motion has been modeled by various ways: interpolation[6], Gaussian mixture models[7], HMM[8], Variable Length Markov Model[9], exemplar (retrieval) model[4], autoregressive model[11], the mixtures of autoregressive models[5], and manifold[10]. The motion models are useful for solving the short-lasting ambiguities between a body shape and its pose due to occlusions.

High dimensionality of joint angles (30-60 dimensions) and their erratic motions make it difficult to represent various motions efficiently and correctly. Such complex motions are usually modeled in a lower dimensional space (e.g. by using PCA, LLE[12], or Isomap[13]). Recently, nonlinear probabilistic embedding (e.g. latent modeling with Gaussian Process, which is called GPLVM[14]) and its extensions are widely used for motion modeling: for example, GPLVM with scaling dimensions for coping with their different variances[15], dynamics representation[16] bidirectional smooth mapping between latent and observation spaces[17], hierarchical representation[18], and a shared latent structure that connects multiple observation spaces[19]. Above all, Gaussian Process Dynamical Models (**GPDM**)[16], dynamic extension of GPLVM, is useful for modeling temporal data such as human motion.

These latent models with Gaussian Process (GP) allow us also to model multiple kinds of actions; a mixture of independent models[20, 22] and a model with multiple actions trained together[20, 21, 23]. While these models represent transition among actions by a kind of interpolation, naive interpolation

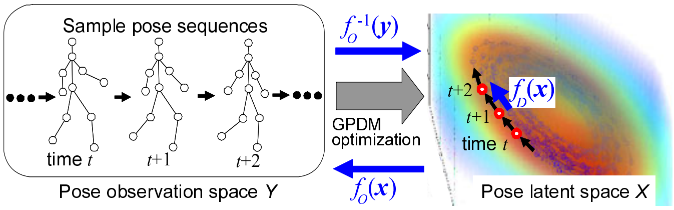


Figure 1: Pose space Y, its latent space X, and mapping functions between and within them, which are acquired by GPDM[16]. Circles and arrows in X depict latent variables and temporal mapping, $f_D(\boldsymbol{x})$, respectively. The background color in X denotes the variance at each point; lower (red) to higher (blue).

might synthesize unrealistic poses in terms of human body kinematics. Indeed, nonlinear GP makes unexpected interpolation results by the weighted sum of many kinds of samples, which might be included in different actions, in the latent space; described in detail in Sec. 5. This fact requires us to check whether or not interpolation along each potential transition path synthesizes realistic human poses.

Transitions among sample sequences of any actions are explicitly modeled by motion graphs[24, 25, 26]. A transition path is synthesized by connecting (i.e. interpolating) different sequences via similar poses. Similarity in motion graphs is evaluated in a high-dimensional pose or shape spaces, so that the sample/interpolated sequences are used as long/short as possible; see [25, 26], for example. The goal is to synthesize new paths as visually natural as possible, while even visually natural poses might be kinematically unrealistic.

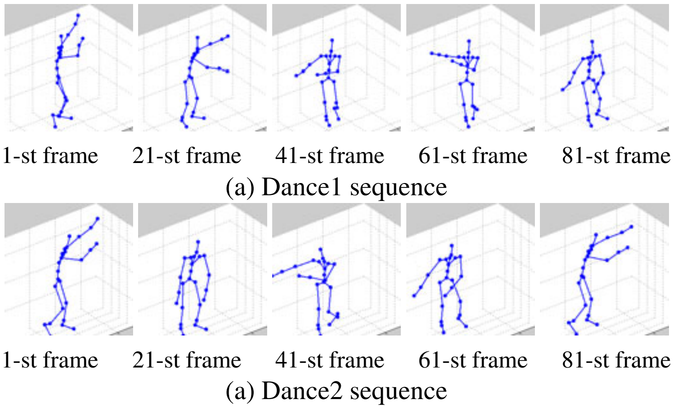
This paper proposes how to integrate the advantages of GP latent models and motion graphs to learn a variety of smooth transitions among actions. A new contribution is kinematically-realistic path synthesis by evaluating motion smoothness and distortion of pose interpolation both in the observation and latent spaces. Our model synthesizes possible transitions (i.e. not only the shortest transition) based on probabilistically-reasonable pose trajectories in a low-dimensional space.

3. Gaussian Process Dynamical Models

3.1. Overview

Gaussian Process Dynamical Models (**GPDM**)[16] (Fig. 1) provide us dimensionality reduction and temporally smooth transition in the low-dimensional latent space. Inherence of the GP allows us to optimize the latent space increasing its generalization and conformity with human body structure and kinematics. GPDM with a D-dimensional observation space Y (i.e. Pose observation space in Fig. 1), which is inherently nonlinear, and its d-dimensional latent space X (i.e. Pose latent space in Fig. 1) is defined by two mappings; 1) from a point at t to a point at $t+1$ in the latent space, $f_D(\boldsymbol{x})$ where $\boldsymbol{x} \in X$, and 2) from the latent space to the observation space, $f_O(\boldsymbol{x})$. The former mapping gives us the capability of prediction and is useful for human motion tracking.

Given a training pose sequence with N frames $\boldsymbol{Y} = [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N]$, the mapping functions are acquired by maximizing the joint likelihood of $\boldsymbol{Y}$ and $\boldsymbol{X}_{t+1}$ with respect to $\boldsymbol{X} = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N]$ and $\boldsymbol{X}_t$, respectively, where $\boldsymbol{X}_{t+1} = [\boldsymbol{x}_2, \cdots, \boldsymbol{x}_N]$ and $\boldsymbol{X}_t = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_{N-1}]$. In this optimization, similarity between variables in $\boldsymbol{X}$ is evaluated by a kernel function and compared with that in $\boldsymbol{Y}$. The kernel function in our experiments was the nonlinear Gaussian radial basis function.



(a) Dance1 sequence

(a) Dance2 sequence

Figure 2: Sample motion frames of two actions. For example, "1st frame in (a) and 1st frame in (b)" and "81-st frame in (a) and 61-st frame (b)" are almost regarded as pairs of potential transition frames between two actions.

3.2. Problems of Inter-action Transitions by GPDM

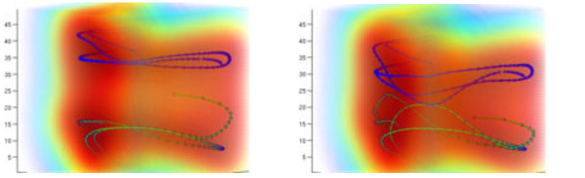
Transition between two actions, *dance1* and *dance2*, was predicted by GPDM. GPDM learned their sample sequences of joint angles [2], which were obtained by a motion capture system (Fig. 2), in a single 3D latent space. In dance1 and dance2 sequences, a subject moved the arms between "right-upper and left-upper" and "right-lower and left-upper", respectively, where transition might occur when he raised the arms. Figure 3 (a) shows the obtained latent model of two actions. In this model, transition between two actions is apparently difficult because their trajectories are distant. This result shows that a standard latent model is superior to classification of different actions but inferior to tracking between them.

To encourage the transition, topologically-constrained modeling[20] is useful. This modeling allows us to arrange the latent variables of poses where the transition may occur close to each other as shown in Fig. 3 (b). Such poses were given manually in this experiment.

To evaluate the possibility of the transition between the sample sequences, particles were distributed around the 1st frame of the dance1 sequence and then moved by temporal mapping, $f_D(\boldsymbol{x})$, of GPDM. While no transition occurred in the standard GPDM, about 10 % of particles reached the dance2 sequence in the topologically-constrained GPDM. The transitions were not sufficient yet because the chance level of the transition should be 50 % if two actions almost overlapped.

Most particles did not transit between the actions because the flows of the temporal mapping around the potential transition frames (Fig. 4) moved in the same directions as those of the sample frames. To encourage the transition, sample sequences between different actions should be prepared for modeling the inter-action transition explicitly.

[2] In all experiments, joint angles were expressed by the exponential map[28].



(a) Standard GPDM (b) Topologically-constrained GPDM

Figure 3: Latent models obtained by GPDM. The blue and green arrows show dance1 and dance2 sequences, respectively.

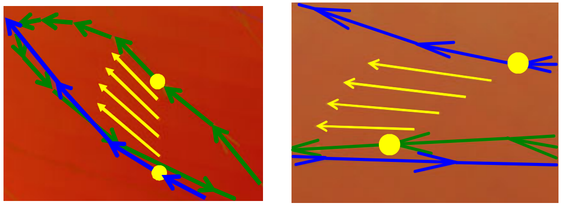
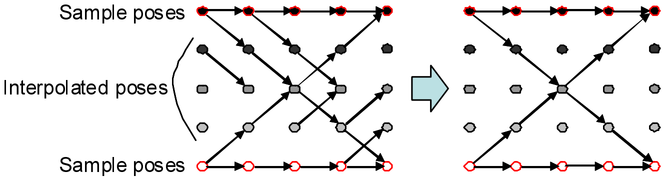


Figure 4: Temporal mapping of GPDM with no samples between different actions. Yellow circles and arrows depict the frames where the transition may occur and the temporal mapping directions from the points between those frames, respectively.



(a) Edge construction (b) Edge pruning

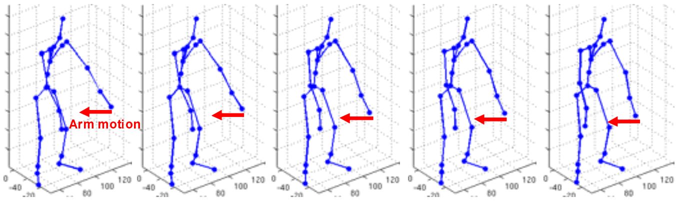
Figure 5: Constructing motion graphs with good connectivity[26]. Arrows depict edges, which connect similar poses.

4. Motion Graphs

4.1. Overview

Motion graphs[24] provide new transition paths between sample sequences with good connectivity and motion quality, which conflict with each other; good connectivity means that transitions are synthesized as many as possible, while good motion quality is achieved by transitions only between similar poses that can make smooth paths.

Motion graphs consist of pose data (i.e. nodes) and possible transitions between them (i.e. directed edges). The state-of-the-art motion graphs proposed in [26] makes a set of interpolations (depicted by "Interpolated poses" in Fig. 5) between sample sequences. This interpolation is achieved only around sample poses each of which has the local minimum of the similarity function, $\|\boldsymbol{y}_i - \boldsymbol{y}_j\|$, where $\boldsymbol{y}_i$ and $\boldsymbol{y}_j$ denote the poses of two sequences. $\boldsymbol{y}_i$ consists of the 3D positions of all joints at i-th frame. From all of the samples and interpolations, motion graphs are constructed by connecting similar poses, as illustrated in Fig. 5 (a). The graphs are then reduced by pruning edges that do not improve connectivity by the Dijkstra's shortest path search as illustrated in Fig. 5 (b); if two poses are connected via multiple paths, only the edges that compose the shortest path are left. The remaining transitions among the samples via the interpolations are more smooth and increased (i.e. better motion quality and connectivity) than those obtained with direct interpolation between the sample poses[24]. This is because pose similarity is densely evaluated between smaller differences (i.e. between more similarly-interpolated poses).



(i) Sample1 (ii) (iii) (iv) (v) Sample2

Figure 6: A pair of matched poses in sample sequences, (i) from dance1 and (v) from dance2, and three smooth interpolations between them, (ii), (iii) and (iv).

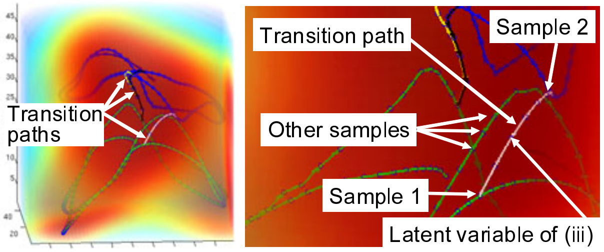
4.2. Problems of Inter-action Transitions with Motion Graphs in the Observation Space

Unlike GPDM, motion graphs are not possessed of capability of motion prediction; they just have possible transition paths. For motion prediction, one natural way is to learn new transition paths synthesized by motion graphs with original sample sequences by employing GPDM. Since GPDM relies on given sample sequences, the synthesized motions must be similar to real motions that are consistent with the human-body kinematics. Note that motion graphs synthesize visually-reasonable poses but do not necessarily take into account the human-body kinematics because motion graphs have been developed for Computer Graphics applications. The kinematic consistency of the poses synthesized by motion graphs is evaluated in this section.

The consistency was evaluated with dance1 and dance2 sequences. The poses of the transition points and their interpolations constructed by [26] are shown in Fig. 6. The interpolated poses are visually reasonable. To quantitatively evaluate their kinematic consistency, they were compared with a range of motions that were captured while a subject moved all joints as whole as possible. All frames of this sequence were modeled by GP. The likelihood of each interpolation, $\boldsymbol{x}$, was expressed by $\exp(-\sigma_{\boldsymbol{x}}^2)$, where $\sigma_{\boldsymbol{x}}^2$ denotes the variance of the distribution of the sample poses, which was obtained by GP.

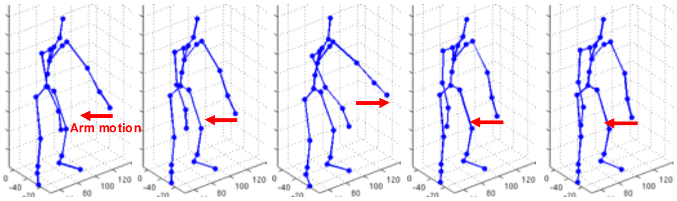
The likelihoods of many interpolations were sufficiently high. In particular, simple hinge joints (e.g. elbows and knees) could be interpolated correctly; the mean of their likelihoods was around 90 % of that of the sample frames. Complex joints (e.g. shoulders and inguinal joints) were, on the other hand, incorrectly interpolated in several frames. In these paths, their likelihoods were less than 60 % of the mean of those in the sample frames.

These kinematically-inconsistent joint angles were synthesized due to linear interpolation in the observation space; even if joint angles $\boldsymbol{a}$ and $\boldsymbol{b}$ are observed in a joint, its angle possibly cannot have several interpolated values between $\boldsymbol{a}$ and $\boldsymbol{b}$. This problem is not critical in CG applications but should be avoided in modeling human motion prior.



Whole image Magnified image

Figure 7: Samples of two actions (blue and green lines) and interpolated variables (other colors) in the latent space obtained by topologically-constrained GPDM.



(i) Sample1 (ii) (iii) (iv) (v) Sample2

Figure 8: Temporal history of sample and interpolated poses. Interpolation (iii) violated a smooth motion; while the arms were moving right to left, they moved to right at (iii).

5. Strengths and Limitations of Interpolation for New Transitions in the Latent Space

Sections 3 and 4 show "action transition with no transition samples in the latent space" and "pose interpolation among different actions in the observation space", respectively. In this section, interpolation in the latent space by motion graphs is verified. In constructing motion graphs, 1) similarity between two latent variables is measured by the Euclidean distance and 2) interpolation between sample poses is computed linearly, in the latent space. The interpolated latent variables and the respective 3D poses obtained by the mapping function $f_O(\boldsymbol{x})$ are shown in Figs. 7 and 8, respectively.

In terms of kinematic consistency, the likelihoods of the poses obtained by motion graphs in the latent space ((ii), (iii), and (iv) in Fig. 8) were higher than those in the observation space; the likelihoods, which were computed by the same way as those in Sec. 4.2, were above 70 % of the mean of those in sample frames. This result says that interpolation in the latent space gives us reasonable poses, as demonstrated in robotic control[27].

However, the human poses synthesized from the interpolated variables made an unexpected motion. The arms of interpolated pose (iii) in Fig. 8 deviated from a smooth motion. This problem occurred due to GP regression $f_O(\boldsymbol{x})$, where a distance-weighted sum of ALL samples is computed, from the latent space to the observation space. In Fig. 7, the latent variables not only of the transition points ("Sample1" and "Sample2" in Fig. 7) but also of other sample frames ("Other samples" in Fig. 7) were close to the interpolated latent variable of the pose (iii). The other sample frames then strongly and incorrectly affected the synthesized pose (iii). Since the human poses of the other sample frames are not similar to those of the transition points, the unexpected poses were synthesized.

The incorrect effect on pose synthesis is caused because interpolated poses are not used in GPDM optimization. That is, if the interpolated poses are used in optimization, GPDM models the latent space so that they are located away from the poses that are not similar to them.

Although the likelihood of a pose can be evaluated by the variance in the latent space, it is difficult to discriminate whether an interpolated pose is unexpected for the smooth path only by evaluating the variance. This is because the interpolated pose might have a low variance due to the proximity between the pose and many samples even if the pose violates the smoothness of the synthesized path.

The incorrect effect on pose synthesis from interpolated latent variables would increase as sample actions increase and/or become more complex.

From the discussions in Sec. 3, 4, and 5, we obtain the following insights:

- Interpolation in the observation space produces smooth but possible kinematically-unreasonable motion.
- Interpolation between different actions, where no samples are given, in the latent space produces kinematically-reasonable but possibly non-smooth motion due to GP regression using ALL samples.

6. Gaussian Process Motion Graph Models

6.1. New Transitions in the GP Latent Models

The goal of this work is to integrate the advantages of GPDM and motion graphs while avoiding unreasonable poses. We call the resulting latent variable models *Gaussian Process Motion Graph Models*, **GPMGM**. GPMGM evaluates pose similarity, smoothness, and distribution both in the observation and latent spaces in order to avoid unreasonable poses. The evaluation is integrated into the shortest path search algorithm for establishing new transition paths.

The steps of GPMGM optimization are described below, each of which is illustrated one by one in Fig. 9:

Step 1: GPDM is applied to a set of samples of all actions. The end points of potential transition paths are extracted from latent variables of GPDM by finding local minima of the following function between all possible pairs of the latent variables in actions i and j:

$$(1/\mu_x)||\boldsymbol{x}_i - \boldsymbol{x}_j|| + (1/\mu_v)||\boldsymbol{v}_i - \boldsymbol{v}_j||, \tag{1}$$

where $\boldsymbol{x}$ and $\boldsymbol{v}$ denote the coordinates and velocity of each latent variable, respectively; the velocity is expressed by temporal mapping of GPDM, $f_D(\boldsymbol{x})$. μ_x and μ_v are the mean values of $||\boldsymbol{x}||$ and $||\boldsymbol{v}||$ in the samples, respectively. Extracting these points in the latent space is robust to noise and nonlinearity in a high-dimensional observation space.

Efficient dimensionality reduction using GPDM allows us to extract the end points of transition paths robustly against noisy and sparse samples in the high-dimensional observation space.

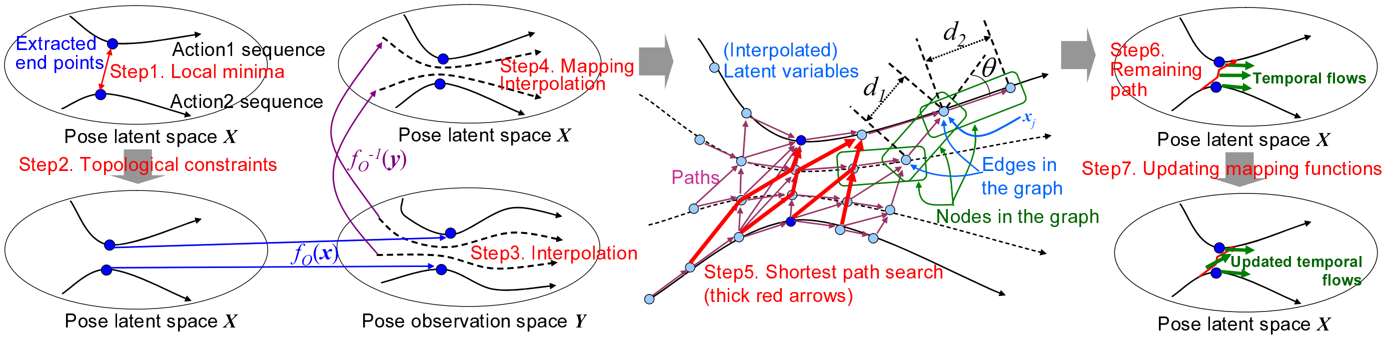


Figure 9: Overview of GPMGM construction. Red notes (from 1. to 7.) explain steps 1 to 7 described in the body text.

Step 2: The latent space is reconstructed by topologically-constrained GPDM[20] so that the extracted end points are close together (as described in Sec. 3.1). This step allows us to acquire smooth trajectories between different actions (i.e. actions i and j), which are crucial for smooth motion prediction. In addition, distance equilibrium between temporally-neighboring points given by this step is required in the following process; described in Step 5.

Step 3: Sample poses around the extracted end points are interpolated in the observation space for synthesizing intermediate poses, which are depicted by dashed arrows in Fig. 9 (as described in Sec. 4.1). Note that the sample poses corresponding to the extracted end points, which are used for interpolation in the observation space, are known. This is because the point correspondence of all training data between the observation and latent spaces is known.

Step 4: The interpolated poses are mapped to the latent space[3]. Notice again that while the poses are interpolated smoothly in the observation space, their respective latent variables might produce kinematically-unreasonable poses.

Step 5: This step 5 finds kinematically-reasonable paths from a set of all the interpolated poses, while keeping their smoothness. To this end, the variance and smoothness of the interpolated poses in the latent space are evaluated.

With the latent variables of the interpolated and sample poses, motion graphs are constructed so that 1) a path that connects two latent variables is regarded as a **node** and 2) a latent variable that connects two paths is regarded as an **edge**. As illustrated in Step5 of Fig. 9, the nodes can be established by any pairs of latent variables.

Each edge (e.g. $\boldsymbol{x}_j$ in Step5 of Fig. 9) has its length that is the weighted sum of three components: i) the lengths of its two nodes (i.e. two paths linked to $\boldsymbol{x}_j$), d_1 and d_2, ii) the angle between them, θ, and iii) the likelihood of $\boldsymbol{x}_j$, which is computed from the variance, $\hat{\sigma}_j^2$, obtained by GP. With these components, the edge length is expressed as follows:

$$(w_d/\mu_d)(d_1 + d_2) + (w_a/\mu_a)\theta + (w_l/\mu_l)\exp(-\hat{\sigma}_j^2), \tag{2}$$

where μ_d, μ_a, and μ_l are the mean values of their respective terms. Weight variables w_d, w_a, and w_l are determined empirically. $\hat{\sigma}_j^2$ is the variance of the distribution only of samples, $\boldsymbol{S}$, that are temporally-connected to the extracted end points[4]:

$$\hat{\sigma}_j^2 \quad = \quad k(\boldsymbol{x}_j, \boldsymbol{x}_j) - \boldsymbol{k}_j^T \boldsymbol{K}^{-1} \boldsymbol{k}_j, \tag{3}$$

where $k(\cdot,\cdot)$, $\boldsymbol{K}$, and $\boldsymbol{k}_j$ denote a kernel function (i.e. RBF kernel, in our experiments), the kernel matrix developed from $\boldsymbol{S}$, and a column vector whose i-th element is $k(\boldsymbol{x}_j, \boldsymbol{x}_i)$; $\boldsymbol{x}_i$ is the i-th point in $\boldsymbol{S}$. Refer to [14] for details. Using only $\boldsymbol{S}$ for computing $\hat{\sigma}_j^2$ can suppress the negative impacts on synthesizing paths that are smooth between the extracted end points.

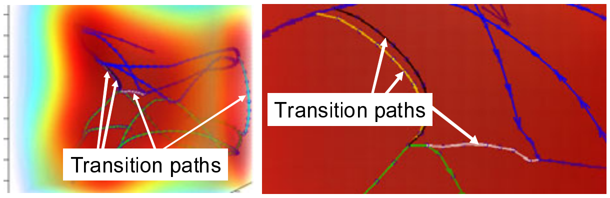
The edge length defined by Eq 2 encourages making kinematically-smooth concatenations of the interpolated poses as well as making shorter paths between different actions. The Dijkstra's algorithm is applied to every possible pair of sample poses around the extracted end points between actions i and j. Only the shortest path is then left in each pair, as depicted by thick red arrows in Step5 of Fig. 9.

New transition synthesis is designed as described above because of the following reasons:

- While shorter paths make smooth and natural transitions, longer paths might make unreasonable poses. This is because 1) as the longer path gets far from the extracted end points, which should be mainly employed for pose interpolation, the pose on the long path might be affected by other poses in GP regression and 2) in mapping the interpolated poses from the observation space, unreasonable poses are mapped to distant regions with large variance. By avoiding the longer paths, therefore, reasonable poses can be selected for path synthesis.

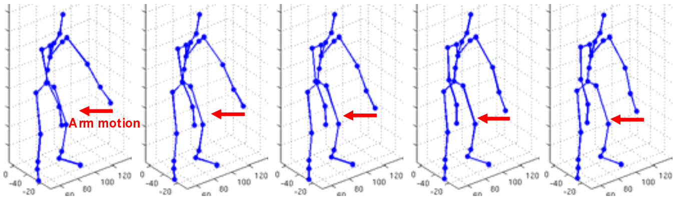
[3] Since $f_O^{-1}(\boldsymbol{y})$ is not obtained by GPDM, it is estimated by general GP regression[29] from sample poses to their respective latent variables after the optimization of GPDM ends. Another alternative is using the back-constrained GPLVM[17], which provides $f_O^{-1}(\boldsymbol{y})$ as well as $f_O(\boldsymbol{y})$. While the back-constraints have a good property for obtaining bidirectional mapping functions, $f_O^{-1}(\boldsymbol{y})$ and $f_O(\boldsymbol{y})$, integration of the back-constraints and the smoothness-constraints with GPDM increases a computational cost, and the optimization results tend to be a local minimum. In our experiments, therefore, GPDM without the back-constraints was used for emphasizing the temporal mapping function. Achieving a good balance between the back-constraints and the smoothness-constraints is included in the future work.

[4] $\boldsymbol{S}$ consisted of 30 samples before and after each end point in our experiments



Whole image Magnified image

Figure 10: Sample data of two actions (blue and green) and interpolated variables (other colors) in the latent space obtained by GPMGM.



(i) Sample1 (ii) (iii) (iv) (v) Sample2

Figure 11: Temporal history of sample and interpolated poses obtained by GPMGM.

- While only pose similarity between nodes is evaluated in motion graphs[26] because they deal with graphical models with no metric on edges, the three metrics (i.e. length, angle, and variance in Eq. (2)) in our models provide good connectivity and motion quality.

Step 6: Depending on the number of the synthesized paths given by the Dijkstra's algorithm, motion prior is determined (i.e. how often the action transition happens). Given the number (denoted by N^p), the top N^p shortest paths remain. The number should be determined in accordance with a task[5] (e.g. subjects, environments, actions, and scenarios).

Step 7: With the remaining interpolated latent variables and directed edges between them, two mapping functions, $f_O(\boldsymbol{x})$ and $f_D(\boldsymbol{x})$, are recomputed by GP regression. Note that only the mapping functions are recomputed, while the latent variables of training samples remain in where they are located by topologically-constrained GPDM[20] in Step 2.

6.2. Improved Inter-action Transitions by GPMGM

The reasonability of new transitions synthesized by GPMGM is validated. Here again, the pose sequences of two dance actions were used.

Synthesized transitions are shown in Fig. 10. Compared with Fig. 7, the number of synthesized transition paths increased, while the one that produced unreasonable poses, which was detected in Fig. 7, was not synthesized. It can be seen that their respective 3D poses were smoothly synthesized as shown in Fig. 11. From a quantitative point of view, the mean of the likelihoods of the interpolated poses was around 95 % of that of the sample poses.

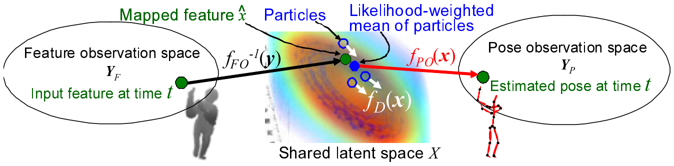


Figure 12: Feature particle filter and its regression to the pose space via the latent space.

7. Human Pose Tracking with GPMGM

7.1. Algorithm

GPMGM was applied to motion prior in human pose tracking. Pose tracking was achieved by image-to-pose regression with particle filtering, whose overview is shown in Fig. 12.

In the learning process, pose data (i.e. joint angles) at each frame is captured with its respective image features (3D volume features[30] and 2D shape contexts[31] in our experiments). for learning an image-to-pose regression function. Motion prior is obtained from the temporal pose data.

In the tracking process, the latent variable of a current pose is estimated by particle filtering with motion prior in the latent space. The current pose is then inferred by pose regression from the estimated latent variable.

The following are more specific descriptions of the tracking algorithm:

- Pose regression was achieved via X as with [32]. An input feature $\boldsymbol{y}$ is mapped to X, which is shared by the feature and pose observation spaces as proposed in [19], by mapping $f_{FO}^{-1}(\boldsymbol{y})$. The distance between the mapped feature, $\hat{\boldsymbol{x}}$, and every particle is computed to obtain their likelihood-weighted mean that is mapped to the pose observation space by mapping $f_{PO}(\hat{\boldsymbol{x}})$. The likelihood of each particle is computed by $\exp(-vl)$, where l denotes the length between $\hat{\boldsymbol{x}}$ and the particle whose variance in X is v.

- Motion prior in X was modeled by GPMGM. Particles are temporally shifted by motion prior ($f_D(\boldsymbol{x})$ in Fig. 12) and then compared with $\hat{\boldsymbol{x}}$.

The difference from the previous methods[32, 30] was that X was optimized by GPMGM.

7.2. Experiments

Pose tracking was performed by the method mentioned in Sec. 7.1. Three kinds of datasets below were used for evaluation:

Set1 A mixture of dance1 and dance2 sequences.

Set2 A mixture of walking and jogging.

Set3 A mixture of six kinds of gait actions: 1) walking, 2) walking slowly, 3) walking fast, 4) striding, 5) jogging, and 6) stopping and walking.

[5] In all experiments in this paper, only one path was left.

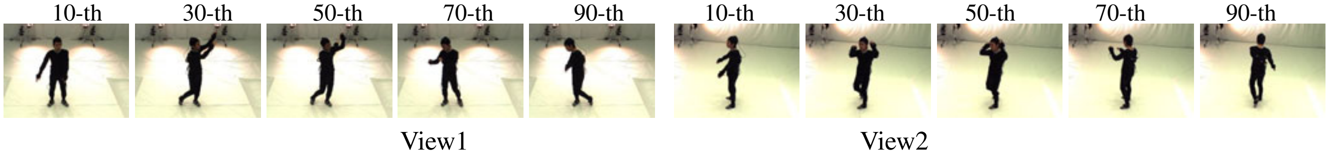
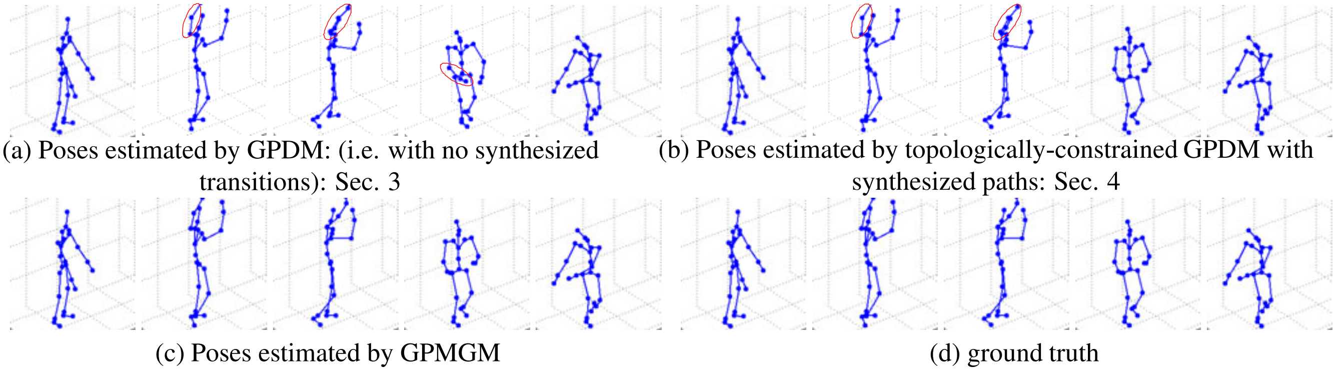


Figure 13: Temporal images of dance sequences: two different views.


(a) Poses estimated by GPDM: (i.e. with no synthesized transitions): Sec. 3

(b) Poses estimated by topologically-constrained GPDM with synthesized paths: Sec. 4

(c) Poses estimated by GPMGM

(d) ground truth

Figure 14: Results of pose tracking in dance sequences. Red circles indicate joints that were misaligned from the respective ground truth.

In all sets, five subjects were tested while only one subject was captured for sample sequences for training. While each sample sequence used for training includes only one kind of action, a test sequence consisted of a number of action transitions. Image sequences were captured at 30fps and 1024x768 pixels. Mocap data was captured also in the test sequences for evaluation. For evaluating the proposed models, these sets are more suitable than existing video and mocap datasets (e.g. HumanEva[1]) in terms of including more action transitions.

For comparison, three kinds of motion prior were tested:

(a) GPDM provided motion prior with no inter-action transitions.

(b) Topologically-constrained GPDM provided motion prior with inter-action transitions synthesized in the observation space by motion graphs.

(c) GPMGM provided motion prior with smooth transition paths. $w_d = 1$, $w_a = 3$, and $w_l = 1$ were determined empirically.

In all models, the sample sequences of different actions was used together for training (i.e. for obtaining a unified model of multiple actions).

All experiments were performed with the same parameters: 256 particles distributed in a 3D latent space.

Pose tracking was achieved by two kinds of features, 3D volume descriptors[30] using multiviews and 2D shape contexts[31] using a single view. Tables 1 and 2 show the RMS errors of all joint positions in the experiments using 3D volume descriptors and 2D shape contexts, respectively. In the tables, the RMS errors throughout all frames and around action transitions are shown. GPMGM could obtain better results than other models, in particular during the action transitions.

Figure 13 shows a test sequence of Set1. In this example, motion prior should be switched from dance1 to dance2 during 20-th and 80-th frames. The latent models by (a), (b), and (c) have been shown in Fig. 3 (a), (b), and Fig. 10. Figure 14 shows pose tracking results estimated using the volume descriptors. Red circles indicate joints that were misaligned from the respective ground truth. Several results in (a) and (b) were misaligned. In particular, the large error of 70-th frame in (a) was caused because particles moved to the dance2 samples slowly due to lack of transition paths.

Table 1: RMS errors of estimated joint positions using 3D shape contexts: (all frames)/(around transition frames).

(mm)	(a) GPDM	(b) T-GPDM	(c) GPMGM
Set1	23/46	18/29	15/26
Set2	30/44	26/34	19/23
Set3	37/52	29/45	25/34

Table 2: RMS errors of estimated joint positions using 2D shape contexts: (all frames)/(around transition frames).

(mm)	(a) GPDM	(b) T-GPDM	(c) GPMGM
Set1	31/38	25/29	23/28
Set2	34/49	28/46	20/29
Set3	48/67	39/56	31/42

Figure 15 shows a test sequence of Set2. Sample sequences of synchronized videos and pose data for learning consisted of two separate action sequences (i.e. walking and jogging sequences) of only one subject. In this example, a subject started jogging during 45-th and 65-th frames. Figure 16 shows pose tracking results obtained using the volume descriptors. Large errors in (a) show that motion prior with no transition paths had difficulty in following inter-action transitions. Relatively large errors in (b) show that tracking accuracy decreased during inter-action transitions with motion prior in the observations space.

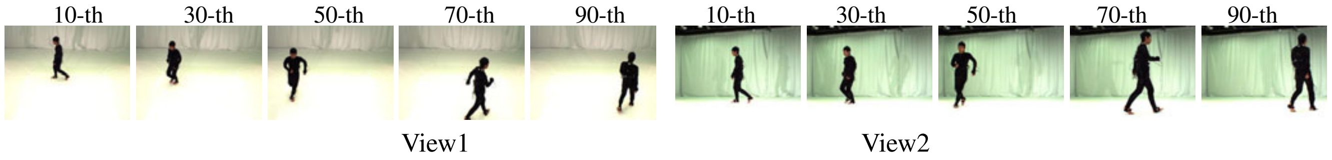
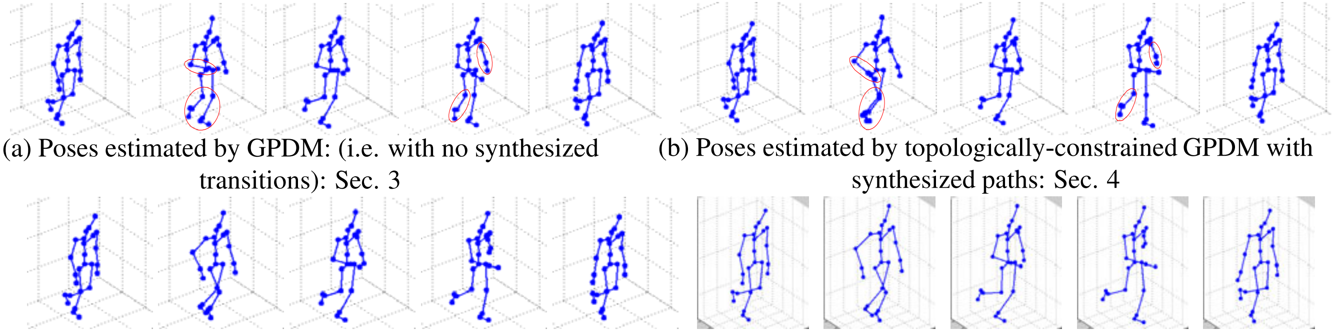


Figure 15: Temporal images of walking and jogging sequences: two different views.



(a) Poses estimated by GPDM: (i.e. with no synthesized transitions): Sec. 3

(b) Poses estimated by topologically-constrained GPDM with synthesized paths: Sec. 4

(c) Poses estimated by GPMGM

(d) ground truth

Figure 16: Results of pose tracking in walking and jogging sequences. Red circles indicate misaligned joints.

Finally, Figure 17 shows a test sequence of Set3. Figure 18 shows the results of pose tracking with 2D shape contexts and GPMGM. Although the errors were higher than those in the above two experiments due to complexity of the combination of the six actions, GPMGM could get better results than other models. It can be seen that pose tracking was achieved well even during repetitive action transitions.

In all the experiments, GPMGM could obtain better results throughout the sequences as well as during transitions among actions. Improvement during all frames other than transition frames might be happened because GPMGM was generalized so that similar motions in different action sequences were modeled closely.

8. Concluding Remarks

We proposed the motion models of multiple actions, GPMGM. GPMGM is learned from independently captured action sequences so that potential transition paths between them are synthesized. Since the transition paths are synthesized in the motion-specific latent space, they reflect the human-body kinematics of the target actions. GPMGM is applicable to any motions because transition paths can be established among any motion trajectories.

In this work, GPMGM is constructed by relying on reasonable latent space modeling. Kinematic reasonability is crucial for meaningful motion synthesis. For improving the reasonability, future work includes employing physical constraints for improving robustness and accuracy of detecting transition points[33] and pruning unrealistic motions[34]. To model a number of different actions, they should be efficiently classified and modeled separately[20] for improving accuracy and scalability of modeling, while all actions were unified in this paper.

The GPDM codes were provided by courtesy of Neil Lawrence and Jack Wang.

[1] L. Sigal and M. J. Black, "HumanEva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion," Technical Report CS-06-08, Brown University, 2006. http://vision.cs.brown.edu/humaneva/

[2] CMU Graphics Lab Motion Capture Database: http://mocap.cs.cmu.edu/

[3] D. Vlasic, I. Baran, W. Matusik, and J. Popovic, "Articulated Mesh Animation from Multi-view Silhouettes," *ACM Transactions on Graphics*, Vol.27, No.3, 2008. http://people.csail.mit.edu/drdaniel/

[4] H. Sidenbladh, M. J. Black, and L. Sigal, "Implicit Probabilistic Models of Human Motion for Synthesis and Tracking," *ECCV*, 2002.

[5] A. Agarwal and B. Triggs, "Tracking Articulated Motion using a Mixture of Autoregressive Models," *ECCV*, 2004.

[6] R. Urtasun, D. Fleet, and P. Fua, "Temporal motion models for monocular and multiview 3D human body tracking," *CVIU*, Vol.104, No.2, pp.157–177, 2006.

[7] N. Huazhong, T. Tan, L. Wang, and W. Hu. "People tracking based on motion model and motion constraints with automatic initialization," *Pattern Recognition*, Vol.37, No.7, pp.1423–1440, 2004.

[8] M. Brand, "Shadow Puppetry," *ICCV*, 1999.

[9] S. Hou, A. Galata, F. Caillette, N. Thacker, and P. Bromiley, "Real-time Body Tracking Using a Gaussian Process Latent Variable Model," *ICCV*, 2007.

[10] A. Elgammal and C.-S. Lee, "Inferring 3D Body Pose from Silhouettes using Activity Manifold Learning," *CVPR*, 2004.

[11] X. Zhao and Y. Liu, "Tracking 3D Human Motion in Compact Base Space," *IEEE Workshop on Applications of Computer Vision*, 2007.

[12] S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, Vol.290, No.5500, pp.2323–2326, 2000.

[13] J. Tenenbaum, V. de Silva, and J. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," Vol.290, No.5500, pp.2319–2323, 2000.

[14] N. D. Lawrence, "Probabilistic non-linear principal component analysis with Gaussian process latent variable models," *Journal of Machine Learning Research*, Vol.6, pp.1783–1816, 2005.

[15] K. Grochow, S. L. Martin, A. Hertzmann, Z. Popovic, "Style-based inverse kinematics," *SIGGRAPH*, 2004.

[16] J. M. Wang, D. J. Fleet, A. Hertzmann, "Gaussian Process Dynamical Models for Human Motion," *PAMI*, Vol.30, No.2, pp.283–298, 2008.

[17] N. D. Lawrence, "Local distance preservation in the gp-lvm through back constraints," *ICML*, 2006.

[18] N. D. Lawrence and A. J. Moore, "Hierarchical Gaussian process latent variable models," *International Conference in Machine Learning*, 2007.

[19] A. P. Shon, K. Grochow, A. Hertzmann, and R. P. N. Rao, "Learning Shared Latent Structure for Image Synthesis and Robotic Imitation," *NIPS*, 2005.

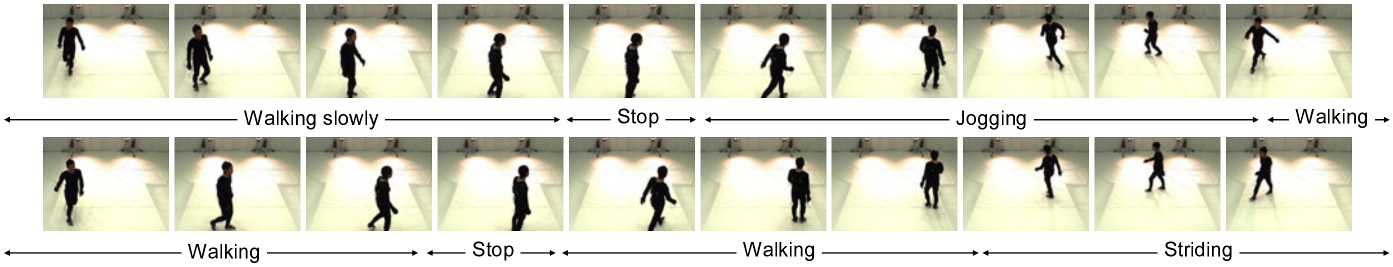


Figure 17: Temporal images (20 frames interval) of a mixture of six kinds of actions.

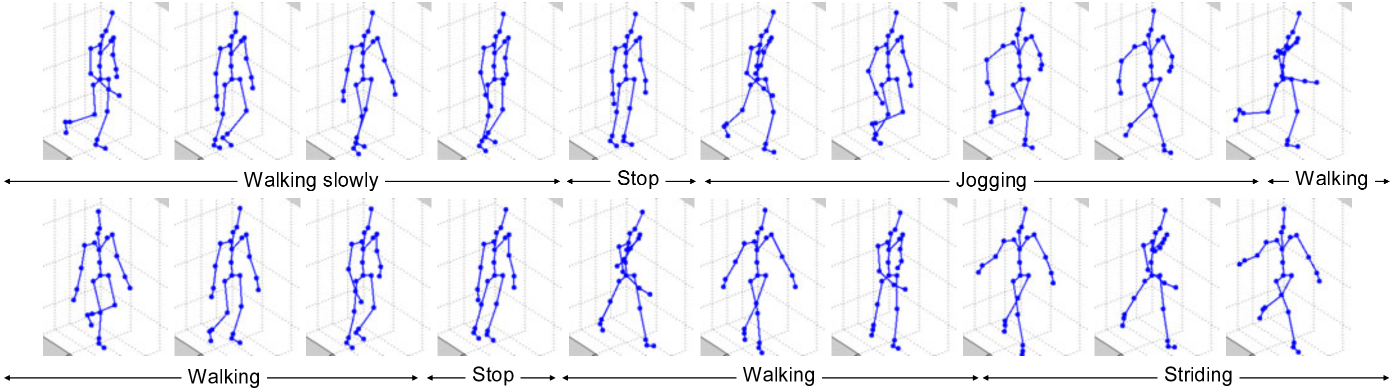


Figure 18: Results of pose tracking with GPMGM in a mixture of six action sequences.

[20] R. Urtasun, D. J. Fleet, A. Geiger, J. Popovic, T. Darrell, and N. D. Lawrence, "Topologically-Constrained Latent Variable Models," *ICML*, 2008.

[21] A. Geiger, R. Urtasun, and T. Darrell, "Rank Priors for Continuous Non-Linear Dimensionality Reduction,"*CVPR*, 2009.

[22] J. Chen, M. Kim, Y. Wang, and Q. Ji, "Switching Gaussian Process Dynamic Models for Simultaneous Composite Motion Tracking and Recognition," *CVPR*, 2009.

[23] J. M. Wang, D. J. Fleet, A. Hertzmann, "Multifactor Gaussian Process Models for Style-Content Separation," *ICML*, pp.975–982, 2007.

[24] L. Kovar, M. Gleicher, and F. H. Pighin, "Motion graphs," *SIGGRAPH*, 2002.

[25] J. McCann and N. S. Pollard, "Responsive characters from motion fragments," *ACM Trans. Graph.*, Vol.26, No.3, 2007.

[26] L. Zhao and A. Safonova, "Achieving good connectivity in motion graphs," *Graphical Models Journal*, Vol.71, No.4, pp.139–152, 2009.

[27] S. Bitzer and S. Vijayakumar, "Latent Spaces for Dynamic Movement Primitives," *IEEE RAS Humanoids*, 2009.

[28] A. K. Peters, "Practical parameterization of rotations using the exponential map," *Journal of Graphics Tools*, Vol.3, No.3, pp.29–48, 1998.

[29] C. E. Rasmussen and C. K. I. Williams, "Gaussian Processes for Machine Learning," the MIT Press, 2006.

[30] N. Ukita, M. Hirai, and M. Kidode, "Complex Volume and Pose Tracking with Probabilistic Dynamical Models and Visual Hull Constraints," *ICCV*, 2009.

[31] S. Belongie, J. Malik, and J. Puzicha, "Shape matching and object recognition using shape contexts," *PAMI*, Vol.24, No.4, pp.509–522, 2002.

[32] C. H. Ek, P. H. S. Torr, and N. D. Lawrence, "Gaussian Process Latent Variable Models for Human Pose Estimation," *4th International Workshop on Machine Learning for Multimodal Interaction*, 2007.

[33] A. Safonova, J. K. Hodgins, N. S. Pollard, "Synthesizing Physically Realistic Human Motion in Low-Dimensional, Behavior-Specific Spaces," *SIGGRAPH*, 2004.

[34] M. Vondrak, L. Sigal, and O. C. Jenkins, "Physical Simulation for Probabilistic Motion Tracking," *CVPR*, 2008.