Neural Networks 24 (2011) 493–500

Contents lists available at ScienceDirect

Neural Networks

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

Learning parametric dynamic movement primitives from multiple demonstrations

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 9 September 2010
Received in revised form 10 December 2010
Accepted 9 February 2011

Keywords:
Imitation learning
Movement primitives
Motion styles

ABSTRACT

Learning from demonstration has shown to be a suitable approach for learning control policies (CPs). However, most previous studies learn CPs from a *single* demonstration, which results in limited scalability and insufficient generalization toward a wide range of applications in real environments. This paper proposes a novel approach to learn highly scalable CPs of basis movement skills from *multiple* demonstrations. In contrast to conventional studies with a single demonstration, i.e., dynamic movement primitives (DMPs), our approach efficiently encodes multiple demonstrations by shaping a parametric-attractor landscape in a set of differential equations. Assuming a certain similarity among multiple demonstrations, our approach learns the parametric-attractor landscape by extracting a small number of common factors in multiple demonstrations. The learned CPs allow the synthesis of novel movements with novel motion styles by specifying the linear coefficients of the bases as parameter vectors without losing useful properties of the DMPs, such as stability and robustness against perturbations. For both discrete and rhythmic movement skills, we present a unified learning procedure for learning a parametric-attractor landscape from multiple demonstrations. The feasibility and highly extended scalability of DMPs are demonstrated on an actual dual-arm robot.

1. Introduction

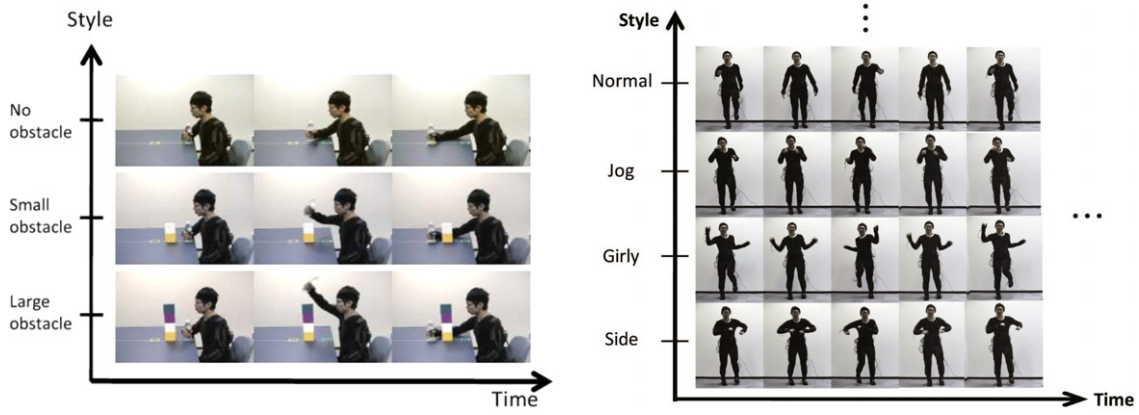
Learning from demonstration has shown to be a suitable approach for learning control policies (CPs) for humanoid robots (Schaal, 1999). As in Ijspeert, Nakanishi, and Schaal (2002), Inamura, Toshima, and Nakamura (2002), Pavlovic, Rehg, and MacCormick (2000), Schaal (1999), Ude, Gams, Asfour, and Morimoto (2010), and Wang, Fleet, and Hertzmann (2005), using a dynamical system to learn the demonstrated movement is becoming popular. One reason is that a trajectory generated by a CP can be easily modulated by changing the parameters of the dynamical system. However, previous studies have focused on learning only from a *single* demonstration. Therefore, the scalability and generalization performance of CPs have been limited. As shown in Ude et al. (2010), a human demonstration can vary greatly, depending on the environmental conditions for a particular task. Therefore, some demonstrations on specific situations are not useful for learning CPs to solve the task in other situations.

An illustrative example is a point-to-point (PTP) reaching task with an obstacle (Fig. 1(a)). While this task requires a discrete movement skill (moving from the start to the goal position), humans demonstrate largely different joint trajectories based on the obstacle's height. For this case, simple scaling of the nominal trajectory (the no-obstacle case in Fig. 1(a)), i.e., modification of the goal, or spatial and temporal scaling, is obviously insufficient to reach the goal position with obstacle avoidance. One straightforward approach would be simply preparing several obstacle-avoidance trajectories and selecting one of them according to the environmental conditions. To extract the proper trajectories from human demonstrations, Stulp, Oztop, Pastor, Beetz, and Schaal (2009) proposed using a clustering method with principal component analysis. However, if the environment is highly complicated, a huge number of trajectories need to be considered. This issue also arises if we attempt to encode a number of demonstrations with a diversity of motion styles in a CP, such as the walking behavior in Fig. 1(b).

These problems motivate us to explore a novel approach to learn the CPs of basis movement skills from *multiple* demonstrations rather than a single demonstration. Computer graphics and animation studies of human motion have proposed several techniques for the synthesis of human-like motions (Arikan, Forsyth, & O'Brien, 2003; Brand & Hertzmann, 2000; Hsu, Pulli, &

* 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.2011.02.004



(a) PTP reaching with an obstacle. (b) Stylistic walking.

Fig. 1. Sequential snapshots of multiple human demonstrations. (a) The subject performs the same motor skill (point-to-point reaching) in many trials with different obstacles. Such differences affect the joint trajectories and resulting styles in each motion sequence while preserving its contents. (b) The subject performs the same content of motion (waking behavior) with different styles.

Popovic, 2005; Shapiro, Cao, & Faloutsos, 2006; Taylor & Hinton, 2009; Torresani, Hackney, & Bregler, 2006; Wang, Fleet, & Hertzmann, 2007). We adopt such style-variable modeling approaches to extract canonical attractor dynamics in a framework of dynamic movement primitives (DMPs) (Bitzer & Vijayakumar, 2009; Ijspeert et al., 2002; Schaal, Mohajerian, & Ijspeert, 2007; Stulp et al., 2009; Ude et al., 2010) by shaping the parametric-attractor landscapes. That is, we find the styles in the shapes of the desired attractor landscapes in a set of differential equations unlike previously proposed style-variable models that directly model joint trajectories, as in Brand and Hertzmann (2000), Herzog, Ude, and Kruger (2008), Taylor and Hinton (2009) and Wang et al. (2007). A novel learning procedure with matrix factorization for both discrete and rhythmic movements is proposed. Assuming a certain similarity among multiple demonstrations, our approach learns the parametric-attractor landscape by extracting a small number of common factors in multiple demonstrations. The learned CPs allow the synthesis of novel movements with novel motion styles by specifying the linear coefficients of the bases as parameter vectors without losing useful properties of the DMPs, such as stability and robustness against perturbations.

The organization of this paper is as follows. In Section 2, we briefly introduce DMPs and the learning procedure for both discrete and rhythmic movements (Ijspeert et al., 2002; Schaal et al., 2007). Then, in Section 3, we show that parametric-attractor landscapes can be shaped in a set of differential equations from multiple demonstrations by combining the idea of the style-variable modeling and the DMP framework. Section 4 describes the experimental settings and the results obtained. We present two applications of our approach: (1) PTP reaching with obstacle avoidance and (2) stylistic walking behavior on a dual-arm robot. Sections 5 and 6 present a discussion and the conclusion of this paper, respectively.

2. Shaping the attractor landscape from a "single" demonstration

This section describes previous work on DMPs and the learning procedure for shaping an attractor landscape from a single demonstration.

2.1. Dynamic movement primitives for discrete movement

We first explain the definition of DMPs on discrete movement, along with Ijspeert et al. (2002) and Schaal et al. (2007). Assume that we have a point attractive system as a basic CP of a one degree of freedom (one-DoF) motor system as

$$\tau \dot{z} = \alpha_z \left(\beta_z \left(g - y\right) - z\right), \tag{1}$$

$$\tau \dot{y} = z + xf, \tag{2}$$

where z and y are state variables of the system, g denotes the goal position, x and f are external variables, α_z and β_z are constants, and τ is a time constant. y and $\dot{y}$ correspond to the desired position and velocity generated by the control policy.[1] For appropriate settings of parameters α_z and β_z with the constraint $f = 0$, these equations have a global stability with a unique point attractor g; i.e., y converges to g after a transient from any initial conditions.

The above dynamics can be applied for learning from the demonstration scenario by introducing an additional dynamic system of x as

$$\tau \dot{x} = -\alpha_x x \tag{3}$$

and the setting of f with x and parameter $\mathbf{w}$ as $f(x; \mathbf{w})$, where α_x is a constant. We call the system in Eq. (3) a *canonical system*, as the most basic dynamic system available to create a point attractor. x is referred to as the state variable of the canonical system in the following. Corresponding to this, the dynamical system in Eq. (2) is called an *output system*, and the system including both is called the dynamic movement primitives (DMPs). y and z are referred to as the state variables of the output system, and $f(x; \mathbf{w})$ is called the attractor function in the following. If the initial condition of x is 1 and α_x is properly set for the system to be stable, $x(t) \in [0, 1]$ is considered as a phase variable for $f(x; \mathbf{w})$ because $f(x; \mathbf{w})$ is a function of the space of phase variable x, and x also acts as a gating term of $f(x; \mathbf{w})$ for $xf(x; \mathbf{w})$ in Eq. (2). With all the above settings and assumption of the boundedness of $f(x; \mathbf{w})$ for all x, y asymptotically converges to the unique point g because the term $xf(x; \mathbf{w})$ vanishes with the convergence of phase x to 0 through time evolution of the canonical system.

Learning parameter $\mathbf{w}$ for shaping the attractor landscape suitable for imitation of a given trajectory $\{y_{\text{demo}}(t_c)\}$, $t_c = c\Delta t$, $c = 1, \ldots, C$ with its duration $T = C\Delta t$ can be accomplished by a supervised learning algorithm. The target trajectory for shaping the attractor landscape is given as $f_{\text{target}}(t_c) = \tau \dot{y}_{\text{demo}}(t_c) - z_{\text{demo}}(t_c)$ in Eq. (2), where $z_{\text{demo}}(t_c)$ is obtained by integrating Eq. (2) with $y_{\text{demo}}(t_c)$ instead of $y(t_c)$. Its input corresponds to the phase value

[1] For simplicity, we insert f in Eq. (2) rather than in Eq. (1) so that imitation is focusing on positions and velocities as in Ijspeert et al. (2002).

$x(t_c)$ at the same time t_c. We use a vector representation of each trajectory as $\mathbf{y}_{\text{demo}} = [y_{\text{demo}}(t_1), \ldots, y_{\text{demo}}(t_C)]^T$ for short. It is also applied for $\mathbf{f}_{\text{target}} = [f_{\text{target}}(t_1), \ldots, f_{\text{target}}(t_C)]^T$ and $\mathbf{x} = [x(t_1), \ldots, x(t_C)]^T$.

The learned DMP has an attractor landscape to generate similar trajectories to the demonstration by time evolutions. Temporal scaling and modification of goal position can be easily accomplished by manipulating g and τ. Note that for a multi-DoF motor system, the output system Eq. (2) must be set for every DoF independently. The canonical system can be shared across all DoFs if they are coordinated.

2.2. Dynamic movement primitives for rhythmic movement

For rhythmic movements, the limit cycle dynamics is modeled by replacing the canonical system of x in Eq. (3) with the following system, which has a stable limit cycle in polar coordinates (ϕ, r):

$$\tau\dot{\phi} = 1, \qquad \tau\dot{r} = \mu\,(r - r_0)\,, \tag{4}$$

where ϕ and r are state variables of the system, μ is a constant, and r_0 is the target amplitude parameter.

Similar to the discrete system, the output system is defined with a phase variable $\tilde{\phi} = \text{mod}(\phi, 2\pi)$ and the amplitude signal r as follows:

$$\tau\dot{z} = \alpha_z\,(\beta_z\,(y_a - y) - z)\,, \qquad \tau\dot{y} = z + rf(x; \mathbf{w}), \tag{5}$$

where $x = \tilde{\phi}$, y and z are state variables, y_a denotes an anchor point for the oscillatory trajectory, and $f(x; \mathbf{w})$ is the attractor function. α_z and β_z are constants, and τ is a time constant. For appropriate settings of parameters α_z and β_z, the system converges to a limit cycle trajectory. Like the discrete system, supervised learning can be applied for shaping an attractor landscape from a demonstrated trajectory by adjusting $\mathbf{w}$ of $f(x; \mathbf{w})$.

3. Shaping the parametric-attractor landscape from "multiple" demonstrations

In this section, we propose a novel approach to efficiently encode multiple demonstrations in a set of dynamical systems as a CP by combining the idea of the style-variable modeling and the framework of DMPs.

3.1. Parametric dynamic movement primitives

To encode multiple demonstrations efficiently while preserving useful properties of the DMPs, we propose to model the attractor function $f(x; \mathbf{w})$ in Eqs. (2) and (4) by a multi-factor function. We introduce an additional control variable $\mathbf{s}$ as $\tilde{f}(x, \mathbf{s}; \mathbf{W})$, so that $\tilde{f}(x, \mathbf{s}; \mathbf{W})$ with a specific value of $\mathbf{s}$ represents the attractor landscape encoding a specific demonstration as

$$\tilde{f}(x, \mathbf{s}; \mathbf{W}) = \sum_{j=1}^{J} s_j b_j(x; \mathbf{w}_j) = \mathbf{s}^T \mathbf{b}(x; \mathbf{W}), \tag{6}$$

where $\mathbf{s} = [s_1, \ldots, s_J]^T$ is a *style parameter*. $\mathbf{b}(x; \mathbf{W}) = [b_1(x; \mathbf{w}_1), \ldots, b_J(x; \mathbf{w}_J)]^T$ can be considered as basis functions (typically modeled by a normalized Gaussian network (NGnet) or other nonlinear function approximators) which span a subspace including a variety of attractor landscapes for multiple demonstrations, and $\mathbf{W} = [\mathbf{w}_1, \ldots, \mathbf{w}_J]^T$ is a parameter matrix. J is the dimension of the style parameter. With this multi-factor modeling, we can efficiently encode a number of demonstrations with a relatively low-dimensional subspace. We call this proposed framework parametric dynamic movement primitives (PDMPs).

3.2. Algorithm for learning parametric attractor landscapes

The learning algorithm of PDMPs from multiple demonstrations has the following four steps.

1. Alignment of demonstrations for subsequent steps. Assume that we have M sets of trajectories as multiple demonstrations $\left\{y^m_{\text{demo}}(t^m_c)\right\}, m = 1, \ldots, M, t^m_c = c\Delta t, c = 1, \ldots, C^m$, where the duration of each demonstration is given as $T^m = C^m \Delta t$. After selecting a nominal trajectory indexed by $n \in \{1, \ldots, M\}$, other trajectories are time-scaled by the ratio $\frac{T^n}{T^m}$ so that all trajectories are represented as the same size of vector as $\mathbf{y}^m_{\text{demo}} \in \mathbb{R}^{C^n \times 1}$ for all m; that is, $\mathbf{y}^{1:M}_{\text{demo}} = \{\mathbf{y}^1_{\text{demo}}, \ldots, \mathbf{y}^M_{\text{demo}}\}$.
2. Calculation of target $\mathbf{f}^m_{\text{target}}$ from demonstration $\mathbf{y}^m_{\text{demo}}$ through numerical integrations with attractor dynamics separately for all m along with the same process of DMPs. By applying this process for all demonstrations, we obtain $\mathbf{f}^{1:M}_{\text{target}} = \{\mathbf{f}^1_{\text{target}}, \ldots, \mathbf{f}^M_{\text{target}}\}$. Note that, for discrete movements, a demonstration $\mathbf{y}^m_{\text{demo}}$ is translated to start at $y^m_{\text{demo}}(t^m_c) = 0$ for all m, while y_a is set by the average of $\mathbf{y}^m_{\text{demo}}$ for all m for rhythmic movements.
3. Extraction of *basis targets* $\mathbf{f}^{1:J}_{\text{basis}} = \{\mathbf{f}^1_{\text{basis}}, \ldots, \mathbf{f}^J_{\text{basis}}\}$ from all targets $\mathbf{f}^{1:M}_{\text{target}}$ so that target $\mathbf{f}^m_{\text{target}}$ is approximately represented as $\mathbf{f}^m_{\text{target}} \approx \sum_{j=1}^{J} s^m_j \mathbf{f}^j_{\text{basis}}$, where $J \leq M$. The basis targets and corresponding style parameters for all demonstrations can be extracted by a matrix factorization with singular-value decomposition (SVD). Let $\mathbf{F}^{\text{all}}_{\text{target}} = [\mathbf{f}^1_{\text{target}} \cdots \mathbf{f}^M_{\text{target}}]^T$ be an $M \times C$ matrix. Then, SVD for this matrix leads to the following factorial representation:

$$\mathbf{F}^{\text{all}}_{\text{target}} = \mathbf{U}\Sigma\mathbf{V}^T \approx \mathbf{S}\mathbf{F}_{\text{basis}}. \tag{7}$$

We define the style parameter matrix $\mathbf{S} = [\mathbf{s}^1 \cdots \mathbf{s}^M]^T \in \mathbb{R}^{M \times J}$ to be the first $J(\leq M)$ rows of $\mathbf{U}$, and the basis target matrix $\mathbf{F}_{\text{basis}} = [\mathbf{f}^1_{\text{basis}} \cdots \mathbf{f}^J_{\text{basis}}]^T \in \mathbb{R}^{J \times C}$ to be the first J columns of $\Sigma\mathbf{V}^T$. The dimension J can be determined with the singular-value spectrum (e.g., select J so that $\sum_{j=1}^{J} \sigma_j / \sum_{m=1}^{M} \sigma_m > 0.9$, where σ_j is the jth diagonal element of Σ ordered in decreasing order).
4. Learning $\mathbf{W}$ (or basis functions $\mathbf{b}(x; \mathbf{W})$) is achieved by supervised learning with $\mathbf{f}^j_{\text{basis}}$ and the corresponding phase vector $\mathbf{x}$ separately for each $j \in \{1, \ldots, J\}$ as

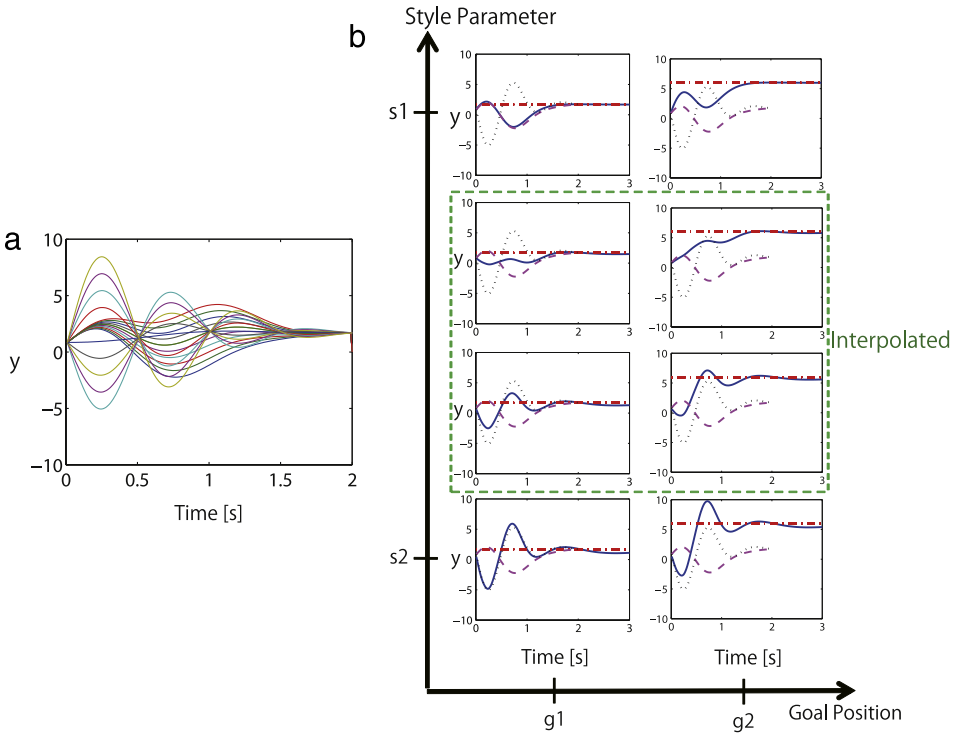
$$\mathbf{w}^*_j \leftarrow \arg\min_{\mathbf{w}_j} \sum_{c=1}^{C^n} \|f^j_{\text{basis}}(t_c) - b_j(x_c; \mathbf{w}_j)\|^2. \tag{8}$$

The optimal parameter $\mathbf{w}^*_j$ for all j with a proper supervised learning algorithm leads to a parametric-attractor landscape $\tilde{f}(x, \mathbf{s}; \mathbf{W}^*)$ parameterized by style parameter $\mathbf{s}$.

The learning $\tilde{f}(x, \mathbf{s}; \mathbf{W})$ is based on the extraction of bases $\mathbf{f}^{1:J}_{\text{basis}}$ which span the J-dimensional subspace encoding all attractor landscapes for M demonstrations. Typically, $J \ll M$ if the target trajectories are correlated. This leads to a compact representation of the PDMPs from multiple demonstrations and novel scalability by interpolating and extrapolating the style parameter $\mathbf{s}$. While the supervised learning problem in step (4) is often solved by standard least-square techniques such as LWL, in this study we utilize the Gaussian process regression (Rasmussen & Williams, 2006) because of its algorithmic simplicity and generalization performance.

3.3. Properties of CPs based on PDMPs

Since the PDMPs are based on the same attractor dynamics to DMPs except the attractor function $\tilde{f}(x, \mathbf{s}; \mathbf{W})$, it preserves several useful properties. Thus, scaling of the goal g for the discrete CP and of the amplitude r_0 for the rhythmic CP do not affect the shape of the attractor landscape. The period (duration) is also manipulated by τ. This property is compatible with the linear time-scaling



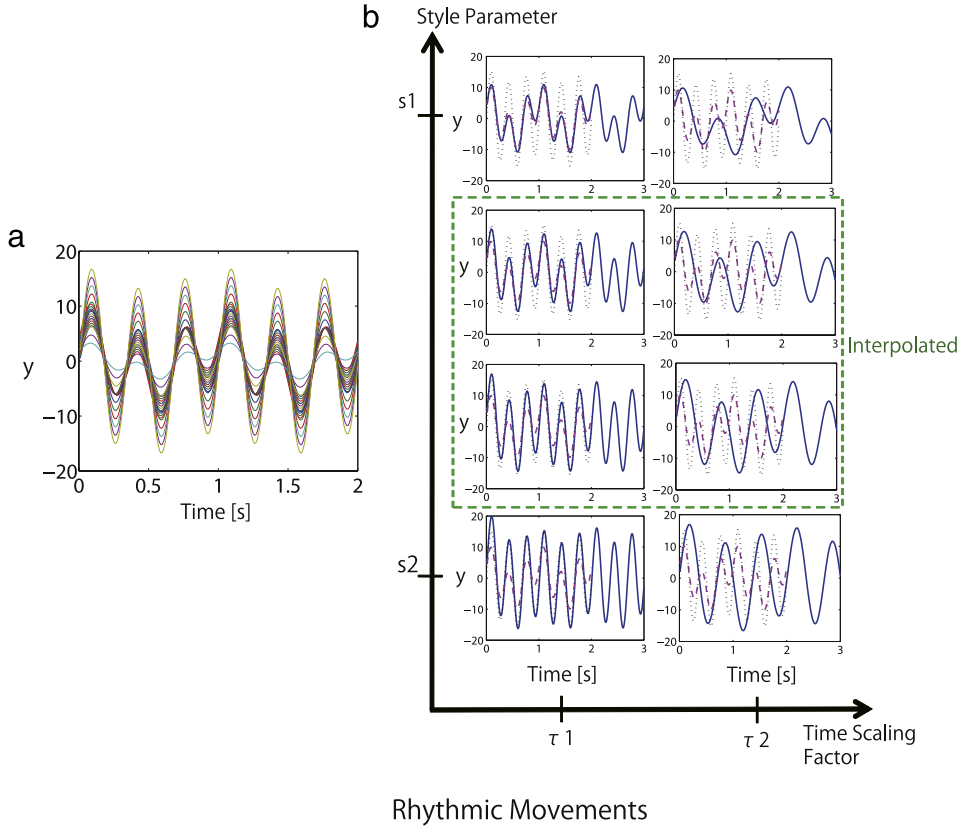


Fig. 2. Examples of time evolution of the PDMPs for both discrete (top) and rhythmic movements (bottom). (a) shows multiple demonstrations, and (b) lines generated trajectories from learned PDMPs. The dashed line shows the demonstration corresponding to style parameter **s**1 and goal position $g1$ (time constant $\tau 1$) and the dotted line indicates the demonstration corresponding to style parameter **s**2. The solid line shows the generated trajectories with several settings of g and τ. The dashed square surrounds style-synthesized (interpolated) trajectories between **s**1 and **s**2.

procedure as a preprocessing of basis extraction in the learning algorithm, i.e., the original trajectories of demonstrations may be reconstructed with a proper setting of τ. Even if these parameters are modified on-line (during movement), the CPs generate a smooth trajectory converging to an attractor with a transient. This property is significant for on-line control of robots, as explored in Park, Hoffmann, Pastor, and Schaal (2008) and Pastor, Hoffmann, Asfour, and Schaal (2009). The PDMPs additionally supply highly extended scalability resulting from the multiple demonstrations, such as with respect to motion styles and variations. These factors can also be modified on-line and are independent of other factors. Some of usefulness will be illustrated through our experimental results in Section 4.

3.4. Illustrative examples with a one-DoF motor system

Here, we illustrate essential properties of the proposed technique for both discrete and rhythmic movements by using a one-DoF motor system. As multiple demonstrations with a certain variation of a movement skill, we sampled 20 trajectories for the discrete and rhythmic movements, respectively.

Fig. 2(a) shows multiple (20) demonstrations for imitation. The corresponding targets $\mathbf{f}_{\text{target}}^{1:M}$ were calculated and the basis targets $\mathbf{f}_{\text{basis}}^{1:J}$ and style parameter $\mathbf{S}$ were extracted. In this case, the dimension of the style parameter was set as $J = 3$ for both discrete and rhythmic movements since our approach can extract a subspace of intrinsic dimension of variation in multiple demonstrations, which is much smaller than the number of demonstrations. The PDMPs were learned with the extracted basis targets. As shown in Fig. 2(b), with specific style parameter $\mathbf{s}$, goal position g (y_a) and time constant τ, the time evolution of the PDMP generates a trajectory imitating the demonstration corresponding to the style parameter. The dashed line shows the demonstration corresponding to style parameter $\mathbf{s}1$ and goal position $g1$ (time constant $\tau 1$) and the dotted line indicates the demonstration corresponding to style parameter $\mathbf{s}2$. The solid line shows the generated trajectories with several settings of g and τ. The generated trajectories were imitating demonstrations very well in both cases ($\mathbf{s}1$ and $\mathbf{s}2$). Moreover, by interpolating the style parameter $\mathbf{s}$ from $\mathbf{s}1$ to $\mathbf{s}2$, the style of the trajectory was smoothly interpolated. The goal position and time-scaling factor were also controlled by changing g and τ, respectively, independently of $\mathbf{s}$. The dashed square surrounds style-synthesized trajectories.

4. Experimental evaluations

This section demonstrates the benefits of the PDMPs against original DMPs and other style-variable models for PTP reaching with obstacle avoidance and stylistic walking behavior with implementation on our dual-arm robot.

4.1. PTP reaching movement with obstacle avoidance

In the first experiment, we applied our technique to learn a motor skill of PTP reaching movements with a human-like obstacle-avoidance strategy as a discrete PDMP from multiple demonstrations, and it was implemented on and demonstrated by a dual-arm robot, as shown in Fig. 3.

4.1.1. Learning a PDMP from multiple demonstrations

As the multiple demonstrations for the task, the subject performed the PTP reaching movement on a table with different heights of obstacles, avoiding them from above, as shown in Fig. 1(a). In all cases, the start and goal positions were fixed. The differences in the height of the obstacle largely affected the resulting joint trajectories while preserving the content as the PTP reaching skill. Fifteen motion sequences were observed by a gyro-type motion capture system as in Fig. 1(a). Four boxes (the same box) were used as obstacles to create five different environmental conditions including the no-obstacle case, and motion sequences were measured in three trials for each case. The height of each box is 0.15 m. In order to adapt human's joint trajectories to the robot, the joint trajectories in the right arm of a human were translated in the robot's right arm four-DoF joint space (three at the shoulder, one at the elbow) through the inverse kinematics technique. The objective function of the inverse kinematics was coordinated by the position and orientations of the end-effector and elbow.

Assuming that the number of joints was D, we aligned the targets for all joints $\mathbf{F}_{\text{target},d}^{\text{all}} \in \mathbb{R}^{M\times C}$, $d = 1 \sim D$ to a unified matrix $\bar{\mathbf{F}}_{\text{target}}^{\text{all}} \in \mathbb{R}^{M\times CD}$ and executed subsequent learning procedures with it for extracting a common style parameter for all joints. As a result, we extracted three basis targets and 15 sets of style parameters, each of which represents the style of each demonstration in the attractor landscape. For learning the PDMP from the extracted basis targets, the parameters were set as $\alpha_z = 1.5$, $\beta_z = 1.5$, $\alpha_x = 0.5$, and $\tau = 0.3$, respectively. The goal for each joint were set as $\mathbf{g} = [-0.55, -0.97, 0.54, -1.42]^T$.

Since the variation of motion style of the multiple demonstrations corresponded to the variation of height of the obstacles in this case, the learned PDMP obtained an additional scalability, i.e., a human-like obstacle-avoidance strategy. By manipulating the style parameter $\mathbf{s} \in \mathbb{R}^3$, the PDMP can represent a variety of CPs suitable for avoiding several heights of obstacles. Of course, other scalabilities, such as the goal position g and duration τ, can be also controlled by each parameter.

The learned PDMP was successfully implemented on the robot. The demonstrations of reaching with several obstacles are presented in Fig. 3. The style parameter was manually selected from $\mathbf{S}$ by choosing one column for each, i.e., $\mathbf{s} = [-0.06, -0.12, -0.38]^T$ (top), $[0.12, -0.28, 0.00]^T$ (middle) and $[0.27, -0.23, 0.50]^T$ (bottom). In the experiment, several obstacles were successfully avoided, as presented in Fig. 3(a).

Fig. 3(b) and (c) show the generated trajectories with different styles and goals. It is important to note that all trajectories converge to the goal positions regardless of the style parameters because the effect of the goal parameter g on the generated trajectories is independent of the effect of the style parameter $\mathbf{s}$. This independence of parameters of the CP is one of the most important advantages of PDMPs.

4.1.2. Prediction of obstacle-avoidance strategies for novel heights of obstacles

In this section, we evaluate the generalization performance of the learned PDMP for novel heights of obstacles. In order to adapt the avoidance strategy for a novel height of obstacle, we derive a mapping between the height of the obstacle h and the corresponding style parameter $\mathbf{s}$ as $U : h \mapsto \mathbf{s}$. Since such a mapping is unknown in general, it needs to be empirically learned from data. Once we learn the mapping, we can predict the style parameter $\hat{\mathbf{s}}$ even for a novel height of obstacle $\hat{h}$, and can predict a PTP reaching motion sequence with a human-like obstacle-avoidance strategy from the PDMP.

All 15 motion sequences measured with the five conditions created by using the four boxes, the same data set used in the previous section, were divided into two groups: the training and the test data sets. The training data set included the nine motion sequences of zero ($h = 0.00$), two ($h = 0.30$) and three ($h = 0.45$) boxes from the 15 motion sequences, where the height of each box is 0.15 m. The test data set contained the six motion sequences of one ($h = 0.15$) and four ($h = 0.60$) boxes. Thus, for the PDMP learned from the training data, we can evaluate an "interpolation" performance with the one-box condition and an "extrapolation" performance with the four-box condition using the test data.

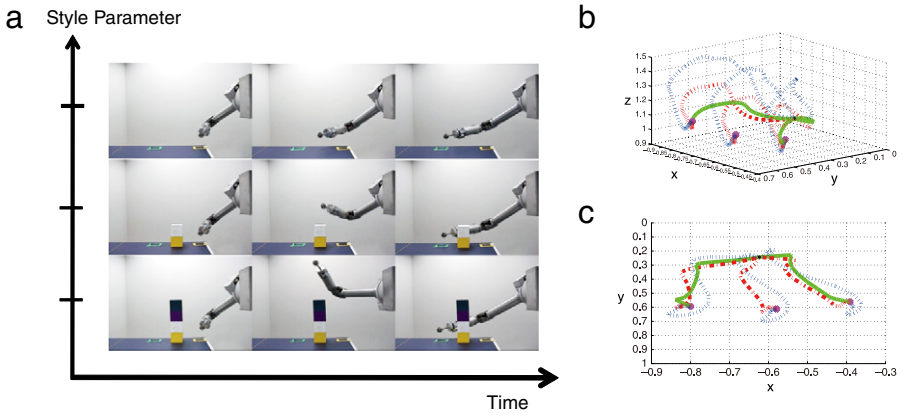


Fig. 3. PTP reaching with obstacle-avoidance strategy on the dual-arm robot (using the right arm). With the setting $\mathbf{g} = [-0.55, -0.97, 0.54, -1.42]^T$ in four-DoF joint space and $\tau = 0.6$, the style parameters were set as $\mathbf{s} = [-0.06, -0.12, -0.38]^T$ (top), $[0.12, -0.28, 0.00]^T$ (middle) and $[0.27, -0.23, 0.50]^T$ (bottom). All motions were the reaching movement with the same initial posture and goal position; however, the styles (trajectories to the goal) were largely different. In the experiment, several obstacles were successfully avoided by properly selecting the style parameters. (a) shows sequential snapshots of the behavior. (b) and (c) plot the hand trajectories with several styles and goals.

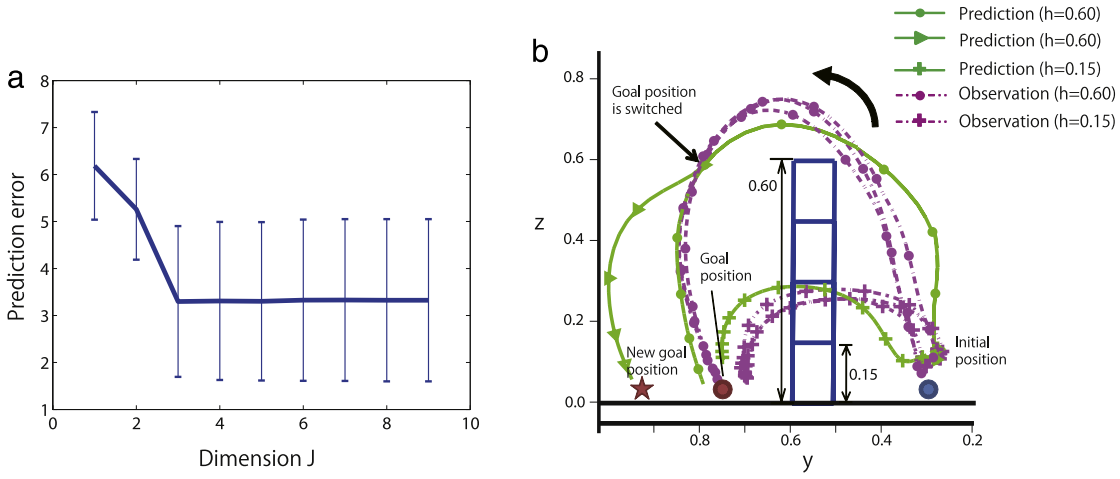


Fig. 4. (a) Prediction errors of motion sequences generated by the PDMPs with different number of style parameters and (b) the predicted trajectories (with $J = 3$) and observed human motion sequences of the tip of the right hand for test conditions. In (a), the solid line indicates the mean of the prediction error E for all test data. The error bar indicates its standard deviation. In (b), the dotted lines indicate the observed human motion sequences for test conditions, and the solid lines show the predicted motion sequences from the height of obstacle h as input. For the test conditions of both one box ($h = 0.15$) and four boxes ($h = 0.60$), the robot successfully avoided the obstacle and reached the goal position in simulations. The goal position can also be independently controlled by the parameter g in an on-line manner during movement. The solid line with a triangle marker presents the predicted trajectory when the goal position was changed at the instance indicated by the arrow. The new goal position is indicated by the star.

We modeled the mapping by a linear function as $U(h) = \boldsymbol{\beta}\bar{\mathbf{h}}$, where $\boldsymbol{\beta} \in \mathbb{R}^{J\times 2}$ is the parameter matrix and $\bar{\mathbf{h}} = [h, 1]^T$ is the query vector. As the result of learning the PDMP from the training data, a set of data was obtained as $\{h^m, \mathbf{s}^m | m = 1 \sim M\}$, where $M = 9$. With the data set, the optimal parameter $\boldsymbol{\beta}^*$ was learned using a standard least-square method.

The generalization performance of the PDMP was then evaluated with the learned PDMP and the learned mapping U. By using the height of the obstacles in the test data, a motion sequence can be predicted. The performance of the generalization was measured by the error between the predicted motion sequences $\hat{\mathbf{y}}(t)$ and measured motion sequences in the test data $\mathbf{y}_{\text{test}}(t)$ as $E = \frac{1}{T}\sum_{i=1}^{T} \|\hat{\mathbf{y}}(t) - \mathbf{y}_{\text{test}}(t)\|^2$. Fig. 4(a) shows the relationship between the prediction errors and the dimension of basis functions J in the learned PDMP. The prediction errors for test conditions are almost saturated at $J = 3$, which is consistent with the result of our suggested strategy to select the dimension from training data (e.g., to determine by the singular-value spectrum as presented in the learning algorithm in Section 3.2).

Note also that if we set a smaller number of J, such as $J = 1$ or $J = 2$, the predicted trajectories could not avoid the obstacles successfully for the test conditions. Thus, with this experimental setting, a human obstacle-avoidance strategy was able to be compactly represented by a three-dimensional style parameter vector in the PDMP. The predicted trajectories with $J = 3$ and observed human motion sequences for the test conditions are plotted in Fig. 4(b). As shown in Fig. 4(b), the goal position can also be independently controlled by the parameter g in an on-line manner during movement. Although we changed the goal position, a smooth transition of the generated motion sequence was achieved. Such a smooth motion transition is an advantage of adopting a DMP framework in the CP.

4.2. Stylistic arm movements of walking behavior

In the second experiment, we applied our technique to learn a motor skill of walking behavior with several motion styles as a rhythmic PDMP from multiple demonstrations.

As shown in Fig. 1(b), the subject performed walking behavior with different motion styles. Twelve motion sequences including a diversity of walking styles (such as normal walking, jogging, girly style, striding, and so on) were observed by the motion

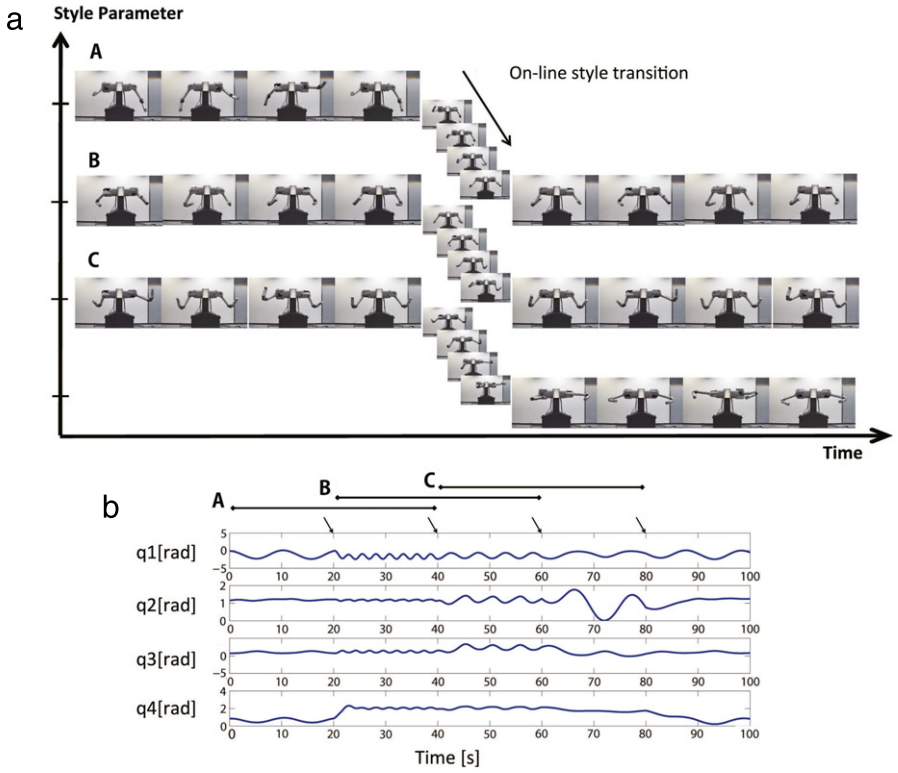


Fig. 5. Stylistic arm swing movements as a part of walking behavior by the dual-arm robot. Several scaling parameters including the style parameter were changed during movement in an on-line manner at 20 s intervals. (a) Sequential snapshots of the behavior. (b) Time series of all joint trajectories. q1–q4 indicate the 1st–4th joint angles of the right arm, respectively.

capture system. As a result, we extracted four basis targets and 12 style parameters, each of which represents the style of each demonstration as similar to the discrete movement case. For learning the PDMP from extracted basis targets, the parameters were set as $\alpha_z = 0.5\beta_z = 0.5$, and $\mu = 0.1$, and y_a was set as the mean of each demonstration and each joint, respectively. τ was set from the walking period of nominal demonstration.

Since the variation of motion style of the multiple demonstrations corresponded to the variation of walking styles in this case, the learned PDMP had an additional scalability, i.e., the style synthesis of walking motion. By manipulating the style parameter $\mathbf{s} \in \mathbb{R}^4$, the PDMP can represent CPs of walking behavior with a variety of motion styles.

The learned PDMP was successfully implemented on the robot. The demonstrations of the walking behavior are presented in Fig. 5. The style parameter **s**, frequency τ, and amplitude r_0 were switched during movement in an on-line manner at 20 s intervals, and the style parameters were selected from **S**, as in the discrete movement case. y_a was also properly set with **s**.

In the experiment, the walking styles, frequency, and amplitude smoothly converged to a specific motion style, as in Fig. 5. Such smooth transitions of the generated motion sequences even for on-line switching of the parameters is also an advantage of adopting a DMP framework in the CP. Note again that with a DMP learned from the normal walking demonstration, even if we modify its frequency and amplitude parameters consistently, it is still impossible to generate jogging-style walking behavior. On the other hand, our approach allows such style synthesis by manipulating a small number of style parameters.

5. Discussion

In contrast to conventional studies with a single demonstration, our approach for learning a PDMP as a CP efficiently encodes multiple demonstrations in a set of differential equations by shaping a parametric-attractor landscape. The learned CPs allow the synthesis of novel movements with novel motion styles by specifying the linear coefficients of the bases as parameter vectors without losing useful properties of DMPs, such as stability and robustness against perturbations. A motion library based approach, proposed by Gams and Ude (2009), could also allow us to synthesize novel motions in a DMP framework; however, it requires executing the learning process for DMPs at every motion synthesis trial because the processes of motion synthesis and learning DMPs are separated. Calinon, D'halluin, Caldwell, and Billard (2009) proposed learning motor primitives from multiple demonstrations; however, the learned CP does not have parameters for scalability. In our approach, once a PDMP is learned as a CP from multiple demonstrations, the synthesis of motion style can be achieved by changing the style parameter, even in motion. This characteristic is highly suitable for on-line motion planning and modifications. Also, note that this study aims to develop a framework for learning such scalable movement primitives from multiple demonstrations rather than trying to solve specific tasks. Recent progresses in the obstacle-avoidance task by DMPs can be found in Park et al. (2008) and Pastor et al. (2009).

The PDMP can be viewed as a combination of a DMP framework and a linear dimensionality-reduction method. Here, we discuss the possibility of using a nonlinear dimensionality-reduction method in the PDMP framework. Nonlinear dimensionality-reduction methods such as kernel PCA (Schölkopf, Smola, & Müller, 1999) or Gaussian process latent variable models (GPLVMs) (Lawrence, 2005) require setting a proper feature space that captures the nonlinearity of data as an assumption; however, it is not easy to find such a feature space in general. Moreover, much higher computational costs for both learning and prediction are needed than for a linear method. From the viewpoint of on-line motion planning and generation for robots, such high computational costs may be unsuitable. In our experiments, the linear method found

proper basis functions for learning a PDMP, and its generalization performance was also validated, as shown in Section 4.1.2.

While the PDMPs presented here are based on the DMPs in Ijspeert et al. (2002), the technique is easily combined with several extensions and modifications of DMPs as in Park et al. (2008) and Pastor et al. (2009). In the experiments presented in the previous section, the style parameters were manually set by hand to be suitable for environmental situations. By combining the PDMPs with a mapping between the style parameter and perceptual feedback from environments using supervised learning techniques as shown in Section 4.1.2, automatic adjustment of the style parameters for environmental conditions can be easily achieved. Learning such a mapping by reinforcement learning, as in Kober, Oztop, and Peters (2010), would be another possibility.

6. Conclusion

This paper has presented a novel approach to learn highly scalable CPs of basis movement skills from multiple demonstrations. In contrast to conventional studies with a single demonstration, i.e., dynamic movement primitives (DMPs), our approach efficiently encodes multiple demonstrations by shaping a parametric-attractor landscape in a set of differential equations. The feasibility and highly extended scalability of the DMPs were demonstrated on an actual dual-arm robot for PTP reaching with obstacle avoidance and stylistic walking behavior. For applications of PDMPs to realistic situations, several concerns such as joint limits and self-collision should be managed in our approach; this will be addressed as part of our future work.

Acknowledgements

We would like to acknowledge Prof. Masatsugu Kidode for his encouragement and support on this collaborative research activity. We would also like to thank Sho Nagashima, Daisuke Shinohara, and Tomohiro Kinoshita for their assistance with all experiments.

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 of Japan, and also partially supported by the Grant-in-Aid for Scientific Research from the Japan Society for the Promotion of Science (WAKATE20800030, WAKATE-B22700177).

References

Arikan, O., Forsyth, D. A., & O'Brien, J. F. (2003). Motion synthesis from annotations. *ACM Transactions on Graphics*, *22*, 402–408.

Bitzer, S., & Vijayakumar, S. (2009). Latent spaces for dynamic movement primitives. In *IEEE-RAS humanoid robotics*.

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

Calinon, S., D'halluin, F., Caldwell, D. G., & Billard, A. G. (2009). Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework. In *IEEE-RAS international conference on humanoid robots* (pp. 582–588).

Gams, A., & Ude, A. (2009). Generalization of example movements with dynamic systems. In *IEEE-RAS international conference on humanoid robots* (pp. 28–33).

Herzog, D., Ude, A., & Kruger, V. (2008). Motion imitation and recognition using parametric hidden Markov models. In *IEEE-RAS international conference on humanoid robots* (pp. 339–346).

Hsu, E., Pulli, K., & Popovic, J. (2005). Style translation for human motion. *ACM Transactions on Graphics*, *24*, 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).

Kober, J., Oztop, E., & Peters, J. (2010). Reinforcement learning to adjust robot movements to new situations. In *Proceedings of robotics: science and systems* (pp. 1–8).

Lawrence, N. (2005). Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, *6*, 1783–1816.

Park, D. -H., Hoffmann, H., Pastor, P., & Schaal, S. (2008). Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields. In *IEEE-RAS international conference on humanoid robotics*.

Pastor, P., Hoffmann, H., Asfour, T., & Schaal, S. (2009). Learning and generalization of motor skills by learning from demonstration. In *IEEE international conference on robotics and automation* (pp. 763–768).

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

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

Schaal, S. (1999). Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences*, *3*, 233–242.

Schaal, S., Mohajerian, P., & Ijspeert, A. (2007). Dynamics systems vs. optimal control—a unifying view. *Progress in Brain Research*, *165*, 425–445.

Schölkopf, B., Smola, A., & Müller, K.-R. (1999). Kernel principal component analysis. In B. Schölkopf, C. J. C. Burges, & A. J. Smola (Eds.), *Advances in kernel methods—SV learning* (pp. 327–352). Cambridge, MA: MIT Press.

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

Stulp, F., Oztop, E., Pastor, P., Beetz, M., & Schaal, S. (2009). Compact models of motor primitive variations for predictable reaching and obstacle avoidance. In *IEEE-RAS humanoid robotics*.

Taylor, G. W., & Hinton, G. E. (2009). Factored conditional restricted Bolzmann machines for modeling motion style. In *Proceedings of the 26th annual international conference on machine learning* (pp. 1025–1032).

Torresani, L., Hackney, P., & Bregler, C. (2006). Learning motion style synthesis from perceptual observations. In *Proceedings of the advances in neural information processing systems* (pp. 1393–1400).

Ude, A., Gams, A., Asfour, T., & Morimoto, J. (2010). Task-specific generalization of discrete and periodic dynamic movement primitives. *IEEE Transactions on Robotics*, *26*, 800–815.

Wang, J., Fleet, D., & Hertzmann, A. (2005). Gaussian process dynamical models. In *Proceedings of advances in neural information processing systems* (pp. 1441–1448).

Wang, J. M., Fleet, D. J., & Hertzmann, A. (2007). Multifactor Gaussian process models for style–content separation. In *Proceedings of the 24th annual international conference on machine learning* (pp. 975–982).