Handbook of Robotics

Chapter 59: Robot Programming by Demonstration

Aude Billard and Sylvain Calinon
LASA Laboratory, School of Engineering,
Ecole Polytechnique Fédérale de Lausanne - EPFL,
Station 9, 1015-Lausanne, Switzerland
{aude.billard,sylvain.calinon}@epfl.ch

Ruediger Dillmann
Forschungszentrum Informatik,
IDS, University of Karlsruhe,
Haid-und-Neu-Str. 10-14, 76131 Karlsruhe, Germany

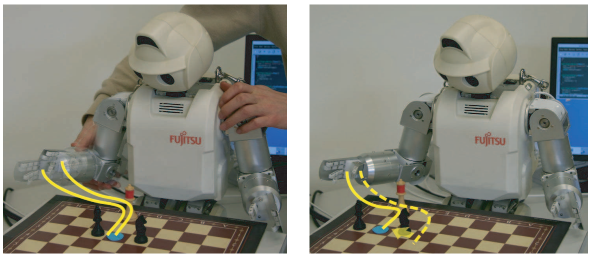
Stefan Schaal
Computer Science & Neuroscience, University of Southern California,
Ronald Tutor Hall, RTH-401 3710 McClintock Avenue
Los Angeles, CA 90089, USA

January 3, 2008

Contents

Chapter 59

Robot Programming by Demonstration



Observation of multiple demonstrations — Reproduction of a generalized motion in a different situation

Figure 59.1: *Left:* A robot learns how to make a chess move (namely moving the queen forward) by generalizing across different demonstrations of the task performed in slightly different situations (different starting positions of the hand). The robot records its joints' trajectories and learns to extract *what-to-imitate*, i.e. that the task constraints are reduced to a subpart of the motion located in a plane defined by the three chess pieces. *Right:* The robot reproduces the skill in a new context (for different initial position of the chess piece) by finding an appropriate controller that satisfies both the task constraints and constraints relative to its body limitation (*how-to-imitate* problem), adapted from [1].

59.1 Introduction

Robot Programming by demonstration (PbD) has become a central topic of robotics that spans across general research areas such as human-robot interaction, machine learning, machine vision and motor control.

Robot PbD started about 30 years ago, and has grown importantly during the past decade. The rationale for moving from purely preprogrammed robots to very flexible user-based interfaces for training robots to perform a task is three-fold.

First and foremost, PbD, also referred to as *imitation learning*, is a powerful mechanism for reducing the complexity of search spaces for learning. When observing either good or bad examples, one can reduce the search for a possible solution, by either starting the search from the observed good solution (local optima), or conversely, by eliminating from the search space what is known as a bad solution. Imitation learning is, thus, a powerful tool for enhancing and accelerating learning in both animals and artifacts.

Second, imitation learning offers an implicit means of training a machine, such that explicit and tedious programming of a task by a human user can be minimized or eliminated (Figure 59.1). Imitation learning is thus a "natural" means of interacting with a machine that would be accessible to lay people.

Third, studying and modeling the coupling of perception and action, which is at the core of imitation learning, helps us to understand the mechanisms by which the self-organization of perception and action could arise during development. The reciprocal interaction of perception and action could explain how competence in motor control can be grounded in rich structure of perceptual variables, and vice versa, how the processes of perception can develop as means to create successful actions.

PbD promises were thus multiple. On the one hand, one hoped that it would make learning faster, in contrast to tedious reinforcement learning methods or trials-and-error learning. On the other hand, one expected that the methods, being user-friendly, would enhance the application of robots in human daily environments. Recent progresses in the field, which we review in this chapter, show that the field has made a leap forward during the past decade toward these goals. In addition, we anticipate that these promises may be fulfilled very soon.

59.1.1 Chapter Content

The remaining of this chapter is divided as follows. Section 59.2 presents a brief historical overview of robot Programming by Demonstration (PbD), introducing several issues that will be discussed later in this chapter. Section 59.3 reviews engineering approaches to robot PbD

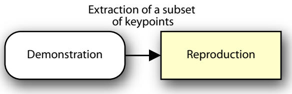


Figure 59.2: Exact copy of a skill by interpolating between a set of pre-defined keypoints, see e.g., [2].

with an emphasis on machine learning approaches that provide the robot with the ability to *adapt* the learned skill to different situations (Section 59.3.1). This Section discusses also the different types of representation that one may use to encode a skill and presents incremental learning techniques to refine the skill progressively (Section 59.3.2). Section 59.3.3 emphasizes the importance to give the teacher an active role during learning and presents different ways in which the user can convey cues to the robot to help it to improve its learning. Section 59.3.4 discusses how PbD can be jointly used with other learning strategies to overcome some limitations of PbD. Section 59.4 reviews works that take a more biological approach to robot PbD and develops models of either the cognitive or neural processes of imitation learning in primates. Finally, Section 59.5 lists various open issues in robot PbD that have yet been little explored by the field.

59.2 History

At the beginning of the 1980s, PbD started attracting attention in the field of manufacturing robotics. PbD appeared as a promising route to automate the tedious manual programming of robots and as way to reduce the costs involved in the development and maintenance of robots in a factory.

As a first approach to PbD, symbolic reasoning was commonly adopted in robotics [2, 3, 4, 5, 6], with processes referred to as *teach-in*, *guiding* or *play-back* methods. In these works, PbD was performed through manual (teleoperated) control. The position of the end-effector and the forces applied on the object manipulated were stored throughout the demonstrations together with the positions and orientations of the obstacles and of the target. This sensorimotor information was then segmented into discrete subgoals (keypoints along the trajectory) and into appropriate primitive actions to attain these subgoals (Figure 59.2). Primitive actions were commonly chosen to be simple point-to-point movements that industrial robots employed at this time. Examples of subgoals would be, e.g., the robot's gripper orientation and

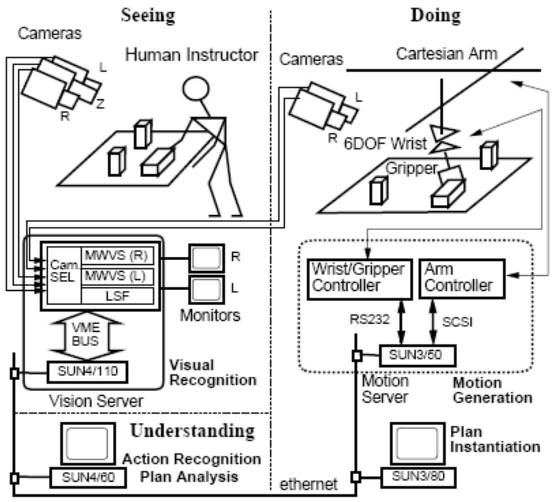


Figure 59.3: Early approaches to Robot Programming by Demonstration decomposed a task into functional and symbolic units. Temporal dependencies across these units were used to build a hierarchical task plan that drove the robot's reproduction of the task [7].

position in relation to the goal [4]. Consequently, the demonstrated task was segmented into a sequence of state-action-state transitions.

To take into account the variability of human motion and the noise inherent to the sensors capturing the movements, it appeared necessary to develop a method that would consolidate all demonstrated movements. For this purpose, the state-action-state sequence was converted into symbolic "if-then" rules, describing the states and the actions according to symbolic relationships, such as "in contact", "close-to", "move-to", "grasp-object", "move-above", etc. Appropriate numerical definitions of these symbols (i.e., when would an object be considered as "close-to" or "far-from") were given as prior knowledge to the system. A complete demonstration was thus encoded in a graph-based representation, where each state constituted a graph node and each action a directed link between two nodes. Symbolic reasoning could then unify different graphical representations for the same task by merging and deleting nodes [3].

Muench et al [8] then suggested the use of *Machine Learning* (ML) techniques to recognize *Elementary Operators* (EOs), thus defining a discrete set of basic motor skills, with industrial robotics applications in mind. In this early work, the authors already established several key-issues of PbD in robotics. These include questions such as how to generalize a task, how to reproduce a skill in a completely novel situation, how to evaluate a reproduction attempt, and how to better define the role of the user during learning. Muench et al [8] admitted that generalizing over a sequence of discrete actions was only one part of the problem since the controller of the robot also required the learning of continuous trajectories to control the actuators. They proposed that the missing parts of the learning process could be overcome by adapting them to the user who had taken an active role in the teaching process.

These early works highlighted the importance of providing a set of examples that the robot can use: (1) by constraining the demonstrations to modalities that the robot can understand; and (2) by providing a sufficient number of examples to achieve a desired generality. They noted the importance of providing an adaptive controller to reproduce the task in new situations, that is, how to adjust an already acquired program. The evaluation of a reproduction attempt was also leveraged to the user by letting him/her provide additional examples of the skill in the regions of the learning space that had not been covered yet. Thus, the teacher/expert could control the generalization capabilities of the robot.

In essence, much current works in PbD follows a conceptual approach very similar to older work. Recent progresses affected mostly the interfaces at the basis of teaching. Traditional ways of guiding/teleoperating the robot were progressively replaced by more user-friendly interfaces, such as vision [9, 10, 11], data gloves [12], laser range finder [13] or kinesthetic teaching (i.e., by manually guiding the robot's arms through the motion) [1, 14, 15].

The field progressively moved from simply copying the demonstrated movements to generalizing across sets of demonstrations. Early work adopted a user-guided generalization strategy, in which the robot may ask the user for additional sources of information, when needed. As Machine Learning progressed, PbD started incorporating more of those tools to tackle both the perception issue, i.e., how to generalize across demonstrations, and the production issue, i.e., how to generalize the movement to new situations. These tools include *Artificial Neural Networks* (ANNs) [16, 17], *Radial-Basis Function Networks* (RBFs) [18], *Fuzzy Logic* [19], or *Hidden Markov Models* (HMMs) [20, 21, 22, 23, 24].

As the development of mobile and humanoid robots more animal-like in their behaviors increased, the field went towards adopting an interdisciplinary approach. It took into account evidence of specific neural mechanisms for visuo-motor imitation in primates [25, 26] as well as evidence of developmental stages of imitation capacities in children [27, 28]. Eventually, the notion of *"Robot Programming by Demonstration"* was replaced by the more biological labelling of *"Imitation Learning"*.

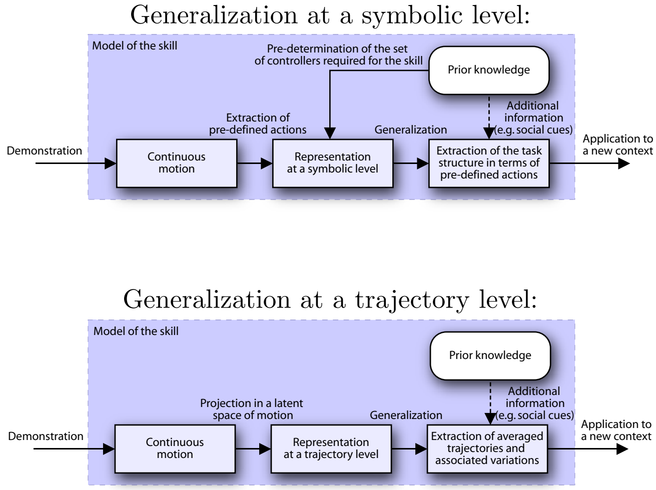


Figure 59.4: Illustration of the different levels of representation for describing the skill.

New learning challenges were, thus, set forth. Robots were expected to show a high degree of flexibility and versatility both in their learning system and in their control system in order to be able to interact naturally with human users and demonstrate similar skills (e.g., by moving in the same rooms and manipulating the same tools as humans). Robots were expected more and more to act "human-like" so that their behavior would be more predictable and acceptable.

Thanks to this swift bioinspiration, Robot PbD became once again a core topic of research in robotics [29, 30, 31, 32]. This after the original wave of robotic imitation based on symbolic artificial intelligence methods lost its thrust in the late 1980s. Robot PbD is now a regular topic at the two major conferences on robotics (IROS and ICRA), as well as at conferences on related fields, such as Human-Robot Interaction (HRI, AISB) and Biomimetic Robotics (BIOROB, HUMANOIDS).

59.3 Engineering-oriented Approaches

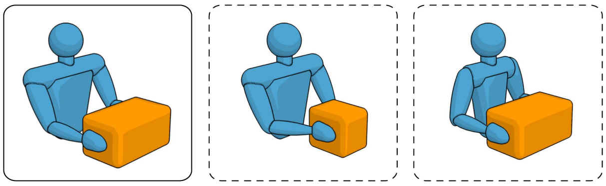
Engineering-oriented and Machine Learning approaches to robot PbD focus on developing algorithms that are generic in their representation of the skills and in the way they are generated.

Table 59.1: Advantages and drawbacks of representing a skill at a symbolic/trajectory level.

	Span of the generalization process	Advantages	Drawbacks
Symbolic level	Sequential organization of pre-defined motion elements	Allows to learn hierarchy, rules and loops	Requires to pre-define a set of basic controllers for reproduction
Trajectory level	Generalization of movements	Generic representation of motion which allows encoding of very different types of signals/gestures	Does not allow to reproduce complicated high-level skills

Current approaches to represent a skill can be broadly divided between two trends: a low-level representation of the skill, taking the form of a non-linear mapping between sensory and motor information, which we will later refer to as "trajectories encoding", and, a high-level representation of the skill that decomposes the skill in a sequence of action-perception units, which we will refer to as "symbolic encoding".

The field has identified a number of key problems that need to be solved to ensure such a generic approach for transferring skills across various agents and situations [33, 34]. These have been formulated as a set of generic questions, namely *what to imitate*, *how to imitate*, *when to imitate* and *who to imitate*. These questions were formulated in response to the large body of diverse work in Robot PbD that could not easily be unified under a small number of coherent operating principles [35, 36, 18, 37, 38, 39]. The above four questions and their solutions aim at being generic in the sense of making no assumptions on the type of skills that may be transmitted. Who and When to imitate have been largely unexplored sofar. Here we essentially review approaches to tackling What and How to imitate, which we refer to as "learning a skill" (what to imitate) and "encoding a skill" (how to imitate). See Figure 59.5 for an illustration.



Demonstration — Reproduction in a different situation (use of a different object) — Reproduction using a different embodiment (use of robots with different limbs size)

Figure 59.5: Illustration of the correspondence problems.

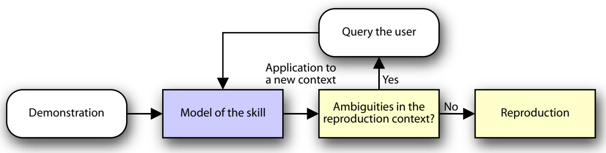


Figure 59.6: Learning of a skill through a query-based approach, see e.g., [8].

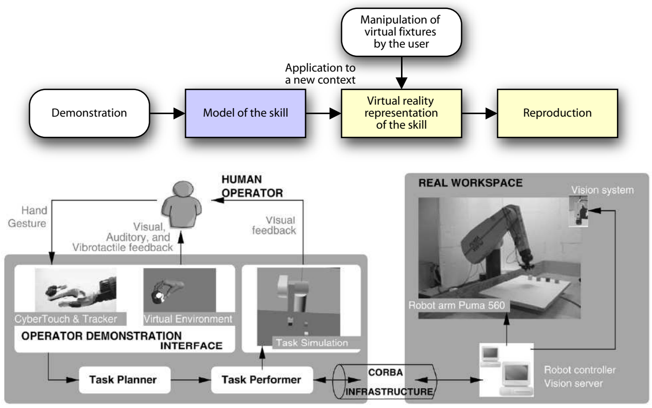


Figure 59.7: PbD in a Virtual Reality (VR) setup, providing the robot with virtual fixtures. VR acts as an intermediate layer of interaction to complement real-world demonstration and reproduction, adapted from [40].

59.3.1 Learning a Skill

As mentioned in Section 59.2, early approaches to solving the problem of how to generalize a given skill to a new/unseen context consisted in explicitly asking the teacher for further information (Figure 59.6).

Another way of providing further information to the robot without relying on symbolic/verbal cues consists in doing part of the training in *Virtual Reality* (VR) or *Augmented Reality* (AR) and providing the robot with *virtual fixtures*, see Figure 59.7 and [43, 44, 45, 46].

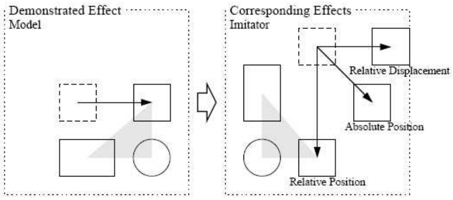


Figure 59.8: Use of a metric of imitation performance to evaluate a reproduction attempt and find an optimal controller for the reproduction of a task (here, to displace the square in a 2D world). The figure is reproduced from [41].

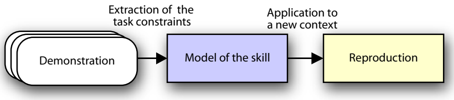


Figure 59.9: Generalization of a skill by extracting the statistical regularities across multiple observations, see e.g., [42].

Additionally, one sees further methods of learning a skill by allowing a robot to automatically extract the important features characterizing the skill and to look for a controller that optimizes the reproduction of these characteristic features. Determining a *metric of imitation performance* is a key concept at the bottom of these approaches. One must first determine the metric, i.e. determine the weights one must attach to reproducing each of the components of the skill. Once the metric is determined, one can find an optimal controller to imitate by trying to minimize this metric (e.g., by evaluating several reproduction attempts or by deriving the metric to find an optimum). The metric acts as a cost function for the reproduction of the skill [33]. In other terms, a metric of imitation provides a way of expressing quantitatively the user's intentions during the demonstrations and to evaluate the robot's faithfulness at reproducing those. Figure 59.8 shows an illustration of the concept of a metric of imitation performance and its use to drive the robot's reproduction.

To learn the metric (i.e., to infer the task constraints), one common approach consists in creating a model of the skill based on several demonstrations of the same skill performed in slightly different conditions (Figure 59.9). This generalization process consists of exploiting the variability inherent to the various demonstrations and to extract the essential components of the task. These essential components should be those that remain unchanged across the various demonstrations [42, 47, 48, 49, 1, 50, 51, 52].

Figure 59.4 presents a schematic of the learning process by considering either a representation of the skill at a symbolic level or at a trajectory level (these two schemas are detailed versions of the tinted boxes depicted in Figure 59.9). Table 59.1 summarizes the advantages and drawbacks of the different approaches.

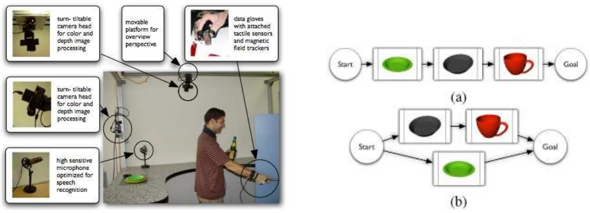


Figure 59.10: *Left:* Training center with dedicated sensors. *Right:* Precedence graphs learned by the system for the 'setting the table' task. (a) Initial task precedence graph for the first three demonstrations. (b) Final task precedence graph after observing additional examples. Adapted from [50].

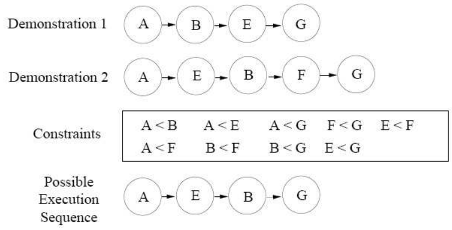


Figure 59.11: Extraction of the task constraints by considering a symbolic representation of the skill. The figure is reproduced from [52].

Next, we review a number of specific approaches to learning a skill at the symbolic and trajectory levels.

Symbolic Learning and Encoding of Skills

A large body of work uses a symbolic representation of both the learning and the encoding of skills and tasks [8, 53, 22, 49, 50, 52, 54, 55]. This symbolic way of encoding skills may take several forms. One common way is to segment and encode the task according to sequences of *predefined* actions, described symbolically. Encoding and regenerating the sequences of these actions can, however, be done using classical machine learning techniques, such as HMM, see [22].

Often, these actions are encoded in a hierarchical manner. In [49], a graph-based approach is used to generalize an object's transporting skill by using a wheeled mobile robot. In the model, each node in the graph represents a complete behaviour and generalization takes place at the level of the topological representation of the graph. The latter is updated incrementally.

[50] and [52] follow a similar hierarchical and incremental approach to encode various household tasks (such as

setting the table and putting dishes in a dishwasher), see Figure 59.10. There, learning consists in extracting symbolic rules that manages the way each object must be handled, see Figure 59.11.

[54] also exploits a hierarchical approach to encoding a skill in terms of pre-defined behaviours. The skill consists in moving through a maze where a wheeled robot must avoid several kinds of obstacles and reach a set of specific subgoals. The novelty of the approach is that it uses a symbolic representation of the skill to explore the teacher's role in guiding the incremental learning of the robot.

Finally, [55] takes a symbolic approach to encoding human motions as sets of pre-defined postures, positions or configuration and considers different levels of granularity for the symbolic representation of the motion. This *a priori* knowledge is then used to explore the correspondence problem through several simulated setups including motion in joint space of arm links and displacements of objects on a 2D plane (Figure 59.8).

The main advantage of these symbolic approaches is that high-level skills (consisting of sequences of symbolic cues) can be learned efficiently through an interactive process. However, because of the symbolic nature of their encoding, the methods rely on a large amount of prior knowledge to predefine the important cues and to segment those efficiently (Table 59.1).

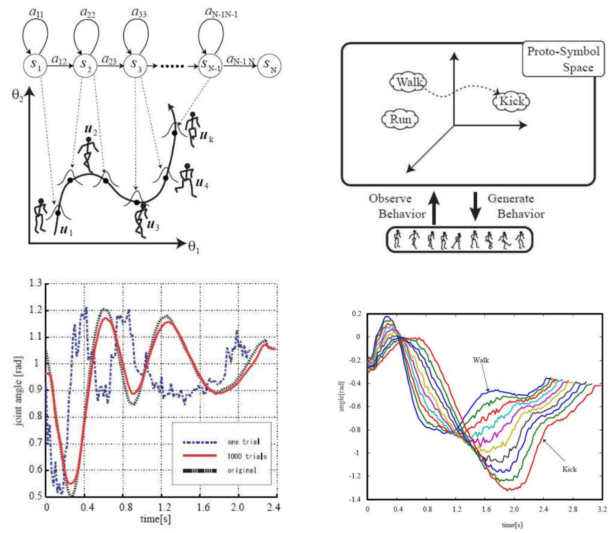


Figure 59.12: Learning of a gesture through the *Mimesis Model* by using *Hidden Markov Model* (HMM) to encode, recognize and retrieve a generalized version of the motion [70]. *Top-left:* Encoding of a full-body motion in a HMM. *Top-right:* Representation of different gestures in a proto-symbol space where the different models are positioned according to the distance between their associated HMM representations. *Bottom-left:* Retrieval of a gesture by using a stochastic generation process based on the HMM representation. *Bottom-right:* Combination of different HMMs to retrieve a gesture combining different motion models.

Learning and Encoding a Skill at Trajectory-Level

Choosing the variables well to encode a particular movement is crucial, as it already gives part of the solution to the problem of defining what is important to imitate. Work in PbD encodes human movements in either joint space, task space or torque space [56, 57, 58]. The encoding may be specific to a cyclic motion [14], a discrete motion [1], or to a combination of both [59].

Encoding often encompasses the use of dimensionality reduction techniques that project the recorded signals into a latent space of motion of reduced dimensionality. These techniques may either perform locally linear transformations [60, 61, 62] or exploit global non-linear methods [63, 64, 65], see Figure 59.14.

The most promising approaches to encoding human movements are those that encapsulate the dynamics of the movement into the encoding itself [66, 67, 68, 69, 59]. Several of these methods are highlighted below.

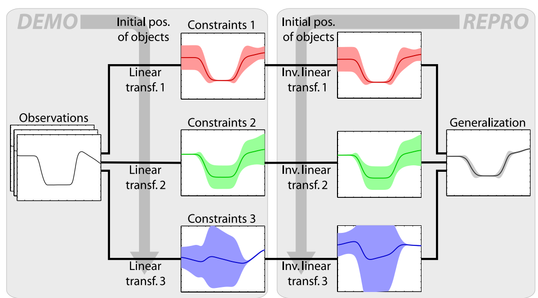


Figure 59.13: Schematic illustration showing continuous constraints extracted from a set of demonstrations performed in different contexts (namely, different initial positions of objects). Each set of signals recorded during the demonstration is first projected into different latent spaces (through an automatic process of reduction of the dimensionality, such as *Principal Component Analysis* (PCA) or *Independent Component Analysis* (ICA)). Each constraint in this constricted space is then represented probabilistically through *Gaussian Mixture Regression* (GMR) (see Table 59.2). In order to reproduce the task, each constraint is first re-projected in the original data space and a trajectory satisfying optimally all constraints is then computed, adapted from [71].

Skill encoding based on statistical modeling

Table 59.2: Probabilistic encoding and reproduction of a skill through *Gaussian Mixture Regression* (GMR).

A dataset $\xi = \{\xi_j\}_{j=1}^N$ is defined by N observations $\xi_j \in \mathbb{R}^D$ of sensory data changing through time (e.g., joint angle trajectories, hand paths), where each datapoint $\xi_j = \{\xi_t, \xi_s\}$ consists of a temporal value $\xi_t \in \mathbb{R}$ and a spatial vector $\xi_s \in \mathbb{R}^{(D-1)}$. The dataset ξ is modelled by a Gaussian Mixture Model (GMM) of K components, defined by the probability density function

$$p(\xi_j) \quad = \quad \sum_{k=1}^{K} \pi_k \mathcal{N}(\xi_j; \mu_k, \Sigma_k),$$

where π_k are prior probabilities and $\mathcal{N}(\xi_j; \mu_k, \Sigma_k)$ are Gaussian distributions defined by *mean* vectors μ_k and *covariance* matrices Σ_k, whose temporal and spatial components can be represented separately as

$$\mu_k = \{\mu_{t,k}, \mu_{s,k}\} \quad , \quad \Sigma_k = \begin{pmatrix} \Sigma_{tt,k} & \Sigma_{ts,k} \\ \Sigma_{st,k} & \Sigma_{ss,k} \end{pmatrix}.$$

For each component k, the expected distribution of ξ_s given the temporal value ξ_t is defined by

$$\begin{aligned} p(\xi_s|\xi_t, k) &= \mathcal{N}(\xi_s; \hat{\xi}_{s,k}, \hat{\Sigma}_{ss,k}), \\ \hat{\xi}_{s,k} &= \mu_{s,k} + \Sigma_{st,k}(\Sigma_{tt,k})^{-1}(\xi_t - \mu_{t,k}), \\ \hat{\Sigma}_{ss,k} &= \Sigma_{ss,k} - \Sigma_{st,k}(\Sigma_{tt,k})^{-1}\Sigma_{ts,k}. \end{aligned}$$

By considering the complete GMM, the expected distribution is defined by

$$p(\xi_s|\xi_t) \quad = \quad \sum_{k=1}^{K} \beta_k \; \mathcal{N}(\xi_s; \hat{\xi}_{s,k}, \hat{\Sigma}_{ss,k}),$$

where $\beta_k = p(k|\xi_t)$ is the probability of the component k to be responsible for ξ_t, i.e.,

$$\beta_k = \frac{p(k)p(\xi_t|k)}{\sum_{i=1}^{K} p(i)p(\xi_t|i)} = \frac{\pi_k \mathcal{N}(\xi_t; \mu_{t,k}, \Sigma_{tt,k})}{\sum_{i=1}^{K} \pi_i \mathcal{N}(\xi_t; \mu_{t,i}, \Sigma_{tt,i})}.$$

By using the linear transformation properties of Gaussian distributions, an estimation of the conditional expectation of ξ_s given ξ_t is thus defined by $p(\xi_s|\xi_t) \sim \mathcal{N}(\hat{\xi}_s, \hat{\Sigma}_{ss})$, where the parameters of the Gaussian distribution are defined by

$$\hat{\xi}_s = \sum_{k=1}^{K} \beta_k \; \hat{\xi}_{s,k} \quad , \quad \hat{\Sigma}_{ss} = \sum_{k=1}^{K} \beta_k^2 \; \hat{\Sigma}_{ss,k}.$$

By evaluating $\{\hat{\xi}_s, \hat{\Sigma}_{ss}\}$ at different time steps ξ_t, a generalized form of the motions $\hat{\xi} = \{\xi_t, \hat{\xi}_s\}$ and associated covariance matrices $\hat{\Sigma}_{ss}$ describing the constraints are computed. If multiple constraints are considered (e.g., considering actions $\xi^{(1)}$ and $\xi^{(2)}$ on two different objects), the resulting constraints are computed by first estimating $p(\xi_s|\xi_t) = p(\xi_s^{(1)}|\xi_t) \cdot p(\xi_s^{(2)}|\xi_t)$ and then computing $\mathbb{E}[p(\xi_s|\xi_t)]$ to reproduce the skill. See Figure 59.13 for an illustration of this method to learning continuous constraints in a set of trajectories. Adapted from [71].

One trend of work investigates how statistical learning techniques deal with the high variability inherent to the demonstrations.

For instance, Ude et al [56] use spline smoothing techniques to deal with the uncertainty contained in several motion demonstrations performed in a *joint space* or in a *task space*.

In [42], using different demonstrators ensures variability across the demonstrations and quantifies the accuracy required to achieve a *Pick & Place* task. The different trajectories form a boundary region that is then used to define a range of acceptable trajectories.

In [48], the robot acquires a set of sensory variables while demonstrating a manipulation task consisting of arranging different objects. At each time step, the robot stores and computers the mean and variance of the collected variables. The sequence of means and associated variance is then used as a simple generalization process, providing respectively a generalized trajectory and associated constraints.

A number of authors following such statistically-based learning methods exploited the robustness of *Hidden Markov Models* (HMMs) in order to encode the temporal and spatial variations of complex signals, and to model, recognize and reproduce various types of motions. For instance, Tso et al [23] use HMM to encode and retrieve Cartesian trajectories, where one of the trajectories contained in the training set is also used to reproduce the skill (by keeping the trajectory of the dataset with the highest likelihood, i.e., the one that generalizes the most compared to the others). Yang et al [57] use HMMs to encode the motion of a robot's gripper either in the *joint space* or in the *task space* by considering either the positions or the velocities of the gripper.

The *Mimesis Model* [72, 73, 74, 75] follows an approach in which the HMM encodes a set of trajectories, and where multiple HMMs can be used to retrieve new generalized motions through a stochastic process (see Figure 59.12). A drawback of such an approach is that it generates discontinuities in the trajectories regenerated by the system. Interpolation techniques have been proposed to deal with this issue [76, 77, 78]. Another approach consists of pre-decomposing the trajectories into a set of relevant keypoints and to retrieve a generalized version of the trajectories through spline fitting techniques [79, 80, 81].

As an alternative to HMM and interpolation techniques, Calinon et al [1] used *Gaussian Mixture Model* (GMM) to encode a set of trajectories, and *Gaussian Mixture Regression* (GMR) to retrieve a smooth generalized version of these trajectories and associated variabilities (Figure 59.13 and Table 59.2).

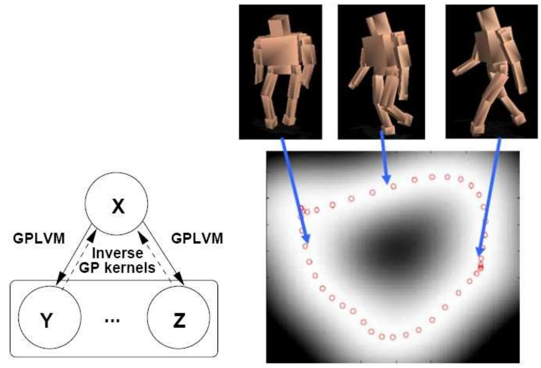


Figure 59.14: Motion learning in a subspace of lower dimensionality by using a non-linear process based on *Gaussian Processes* (GP) [63]. *Left:* Graphical model of imitation consisting of two pairs of Gaussian Process regression models. In the forward direction, latent variable models map from a low-dimensional latent space X to the human joint space Y and robot joint space Z. In the inverse direction, a regression model maps human motion data to points in the latent space, where two similar human postures in the joint angle space produce two points in the latent space that are close one to the other. Thus, the use of generative regression models allows one to interpolate between known postures in the latent space to create reasonable postures during the reproduction. *Right:* The model provides a smooth certainty estimate in the posture's latent space it infers (shaded map from black to white), where training data are represented with circles (here, a walking motion is depicted). Once a latent variable model has been learned, a new motion can be quickly generated from the learned kernels.

Skill encoding based on dynamical systems

Dynamical systems offer a particularly interesting solution to an imitation process aimed at being robust to perturbations which is robust to dynamical changes in the environment.

The first work to emphasize this approach was that of Ijspeert et al [59], who designed a motor representation based on dynamical systems for encoding movements and for replaying them in various conditions, see Figure 59.15. The approach combines two ingredients: nonlinear dynamical systems for robustly encoding the trajectories, and techniques from non-parametric regression for shaping the attractor landscapes according to the demonstrated trajectories. The essence of the approach is to start with a simple dynamical system, e.g., a set of linear differential equations, and to transform it into a nonlinear system with prescribed attractor dynamics by means of a learnable autonomous forcing term. One can generate both point attractors and limit cycle attractors of almost arbitrary complexity. The point attractors and limit cycle attractors are used to respectively encode discrete (e.g. reaching) and rhythmic movements (e.g.

drumming).

Locally Weighted regression (LWR) was initially proposed to learn the above system's parameters [82, 83, 84]. It can be viewed as a memory-based method combining the simplicity of linear least squares regression and the flexibility of nonlinear regression. Further work mainly concentrated on moving on from a memory-based approach to a model-based approach, and moving on from a batch learning process to an incremental learning strategy [85, 86, 61]. Schaal et al [86] used *Receptive Field Weighted Regression* (RFWR) as a non-parametric approach to incrementally learn the fitting function with no need to store the whole training data in memory. Vijayakumar et al [61] then suggested that one uses *Locally Weighted Projection Regression* (LWPR) to improve the way one approaches operating efficiently in high dimensional space. Hersch et al [87] extended the above dynamical approach to learning combinations of trajectories in a multidimensional space (Figure 59.16). The dynamical system is modulated by a set of learned trajectories encoded in a Gaussian Mixture Model, see Section 59.3.1.

The approach offers four interesting properties. First, the learning algorithm, Locally Weighted Regression, is very fast. It does one-shot learning and therefore avoids the slow convergence exhibited by many neural network algorithms, for instance. Second, the dynamical systems are designed so that, from a single demonstration, they can replay similar trajectories (e.g. tennis swings) with the online modification of a few parameters (e.g. the attractor coordinates). This is of great importance for reusing the dynamical systems in new tasks, i.e. the notion of generalization. Third, dynamical systems are designed to be intrinsically robust in the face of perturbations. Small random noise will not affect the attractor dynamics. In the event of a large constraint (e.g. someone blocking the arm of the robot), feedback terms can be added to the dynamical systems and accordingly modify the trajectories online. Finally, the approach can also be used for movement classification. Given the temporal and spatial invariance of the representation, similar parameters tend to adjust to trajectories that are topologically similar. This means that the same representation used for encoding trajectories can also help classifying them, e.g. it provides a tool for measuring the similarities and dissimilarities of trajectories.

Ito et al [14] proposed another way to encode implicitly the dynamics of bimanual and unimanual tasks, using a recurrent neural networks. The model allows online imitation. Of interest is that fact that the network

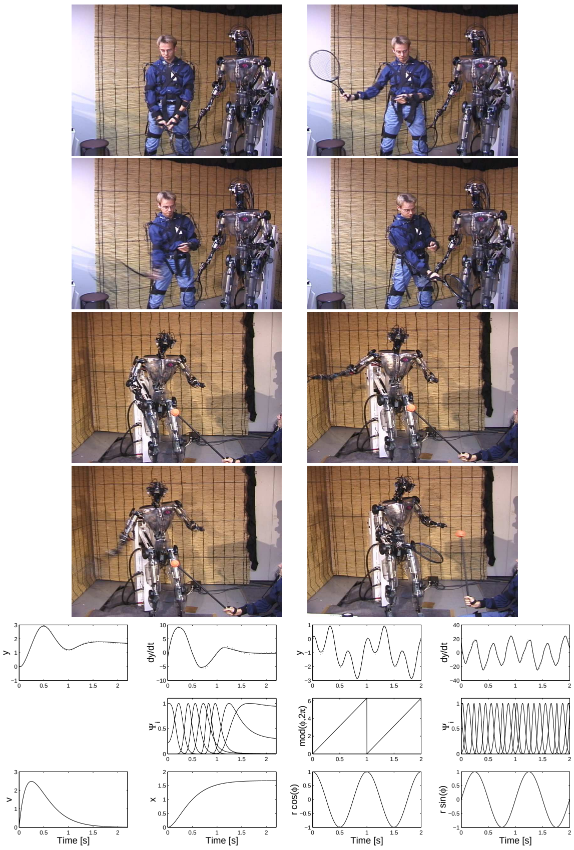


Figure 59.15: *Top:* Humanoid robot learning a forehand swing from a human demonstration. *Bottom:* Examples of time evolution of the discrete (*left*) and rhythmic dynamical movement primitives (*right*). Adapted from [88, 59].

can switch across those motions in a smooth manner when trained separately to encode two different sensorimotor loops (Figure 59.17). The approach was validated for modelling dancing motions during which the user first initiated imitation and where the roles of the imitator and demonstrator could then be dynamically interchanged. They also considered cyclic manipulation tasks during which the robot continuously moved a ball from one hand to the other and was able to switch dynamically to another cyclic motion that consisted of lifting and releasing the ball.

59.3.2 Incremental Teaching Methods

The statistical approach described previously, see Section 59.3.1, has its limitations though it is an interesting way to autonomously extract the important features of the task. In addition, it avoids putting too much prior

Table 59.3: Imitation process using a dynamical system.

A control policy is defined by the following (z, y) dynamics which specify the attractor landscape of the policy for a trajectory y towards a goal g

$$\dot{z} = \alpha_z(\beta_z(g - y) - z), \tag{59.1}$$

$$\dot{y} = z + \frac{\sum_{i=1}^{N} \Psi_i w_i}{\sum_{i=1}^{N} \Psi_i} v. \tag{59.2}$$

This is essentially a simple second-order system with the exception that its velocity is modified by a nonlinear term (the second term in (59.2)) which depends on internal states. These two internal states, (v, x) have the following second-order linear dynamics

$$\dot{v} = \alpha_v(\beta_v(g - x) - v), \tag{59.3}$$

$$\dot{x} = v. \tag{59.4}$$

The system is further determined by the positive constants $\alpha_v, \alpha_z, \beta_v$, and β_z, and by a set of N Gaussian kernel functions Ψ_i

$$\Psi_i = \exp\left(-\frac{1}{2\sigma_i^2}(\tilde{x} - c_i)^2\right), \tag{59.5}$$

where $\tilde{x} = (x - x_0)/(g - x_0)$ and x_0 is the value of x at the beginning of the trajectory. The value x_0 is set each time a new goal is fed into the system, and $g \neq x_0$ is assumed, i.e. the total displacement between the beginning and the end of a movement is never exactly zero. The attractor landscape of the policy can be adjusted by learning the parameters w_i using locally weighted regression [86].
The approach was validated with a 35-degrees-of-freedom humanoid robot, see Figure 59.15. Adapted from [88, 59].

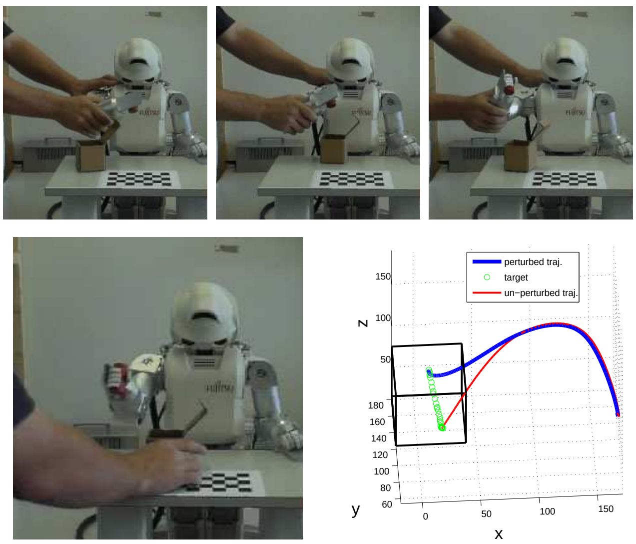


Figure 59.16: Dynamical systems provide a robust robot controller in the face of perturbations during the reproduction of a learned skill. *First row:* The robot is trained through kinesthetic demonstrations by a human trainer (here, the skill consists of putting an object into a box). During these different demonstrations (starting from different positions for both hand and target), the robot records the various velocity profiles its right arm follows. *Second row:* Handling of dynamic perturbations during the reproduction. Here, the perturbations are produced by the user who displaces the box during the reproduction attempts. We see that the robot smoothly adapts its original generalized trajectory (*thin line*) to this perturbation (*thick line*). Adapted from [87].

knowledge in the system. For instance, one can expect that requesting multiple demonstrations of a single task would annoy the user. Therefore, PbD systems should be capable of learning a task from as few demonstrations as possible. Thus, the robot could start performing its tasks right away and gradually improve its performance while, at the same time, being monitored by the user. Incremental learning approaches that gradually refine the task knowledge as more examples become available pave the way towards PbD systems suitable for real-time human-robot interactions.

Figure 59.19 shows an example of such incremental teaching of a simple skill, namely grasping and placing an object on top of another object, see [71] for details.

These incremental learning methods use various forms of deixis, verbal and non-verbal, to guide the robot's attention to the important parts of the demonstration or to particular mistakes it produces during the reproduction of the task. Such incremental and guided learning is often referred to as *scaffolding* or *moulding* of the robot's knowledge. It was deemed most important to allow the robot to learn tasks of increasing complexity [89, 54].

Research on the use of incremental learning techniques for robot PbD has contributed to the development of methods for learning complex tasks within the household domain from as few demonstrations as possible. Moreover, it contributed to the development and appli-

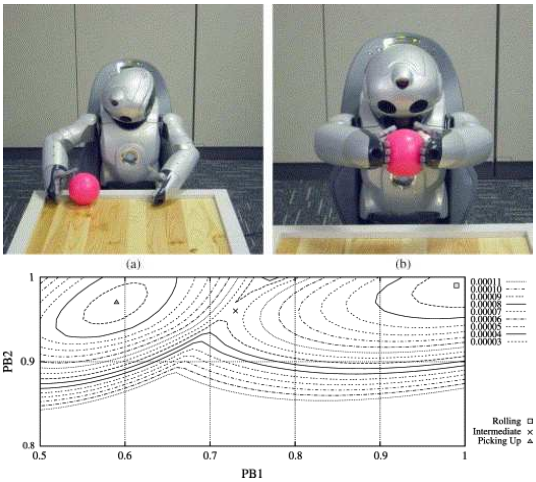


Figure 59.17: Dynamical encoding of the imitation process by using recurrent neural networks. *First row:* Online switching between two interactive behaviours, namely rolling a ball from one hand to the other and lifting the ball. *Second row:* Prediction error distribution in the parameters space representing the two behaviours. One can see how the robot can switch smoothly across the behaviors by moving continuously in the parameter space. Adapted from [14].

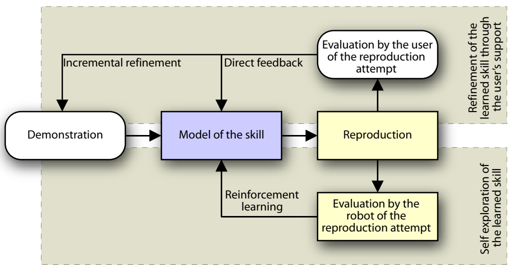


Figure 59.18: Iterative refinement of the learned skill through teacher's support or through self-exploration by the robot.

cation of machine learning that allow continuous and incremental refinement of the task model. Such systems have sometimes been referred to as *background knowledge based* or *EM deductive PbD-systems*, as presented in [90, 91]. They usually require very few or even only a single user demonstration to generate executable task descriptions.

The main objective of this type of work is to build a meta-representation of the knowledge the robot has acquired on the task and to apply reasoning methods to this knowledge database (Figure 59.10). This reasoning involves recognizing, learning and representing repetitive tasks.

Pardowitz et al [92] discuss how different forms of knowledge can be balanced in an incrementally learning system. The system relies on building *task precedence*

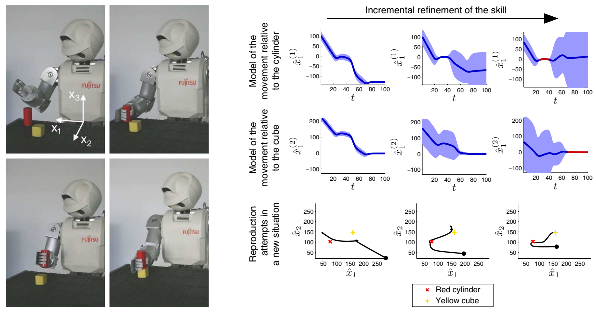


Figure 59.19: Incremental refinement of movements coded in a frame of reference located on the objects that are manipulated. The task constraints are encoded at the trajectory-level. *Left:* Example of a task consisting of grasping a cylinder and placing it on top of a cube. *Right:* Refinement of the *Gaussian Mixture Regression* (GMR) models representing the constraints all along the movement (see Table 59.2). After a few demonstrations, we see that the trajectories relative to the two objects are highly constrained for particular subparts of the task, namely when reaching for the cylinder (thin envelope around time step 30) and when placing it on top of the cube (thin envelope around time step 100). Adapted from [71].

graphs. Task precedence graphs encode hypotheses that the system makes on the sequential structure of a task. Learning the task precedence graphs allows the system to schedule its operations most flexibly while still meeting the goals of the task (see [93] for details). Task precedence graphs are directed acyclic graphs that contain a temporal precedence relation that can be learned incrementally. Incremental learning of task precedence graphs leads to a more general and flexible representation of the task knowledge, see Figure 59.10.

59.3.3 Human-Robot Interaction in PbD

Another perspective adopted by PbD to make the transfer of skill more efficient is to focus on the interaction aspect of the transfer process. As this transfer problem is complex and involves a combination of social mechanisms, several insights from Human-Robot Interaction (HRI) were explored to make efficient use of the teaching capabilities of the human user.

The development of algorithms for detecting "social cues" (given implicitly or explicitly by the teacher during training) and their integration as part of other generic mechanisms for PbD has become the focus of a large body of work in PbD. Such social cues can be viewed as a way to introduce priors in a statistical learning system, and, by so doing, speed up learning. Indeed, several hints can be used to transfer a skill not only by demonstrating the task multiple times but also by highlighting the important components of the skill. This can be achieved

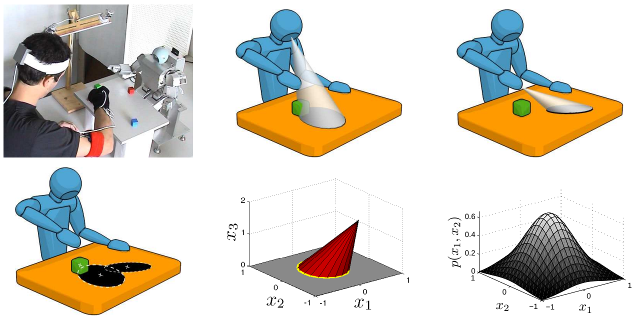


Figure 59.20: Illustration of the use of social cues to speed up the imitation learning process. Here, gazing and pointing towards information are used to select probabilistically the objects relevant for different subparts of a manipulation skill. *First row:* Illustration of the setup proposed in [94] to probabilistically highlight the importance of a set of objects through the use of motion sensors recording gazing and pointing directions. *Second row:* Probabilistic representation of the intersection between the gazing/pointing cone and the table to estimate where the attention of the user is focused on.

by various means and by using different modalities.

A large body of work explored the use of pointing and gazing (Figure 59.20) as a way to convey the intention of the user [95, 96, 97, 98, 99, 100, 44, 101]. Vocal deixis, using a standard speech recognition engine, has also been explored widely [102, 44]. In [50], the user makes vocal comments to highlight the steps of the teaching that are deemed most important. In [103, 104], sole the prosody of the speech pattern is looked at, rather than the exact content of the speech, as a way to infer some information on the user's communicative intent.

In [94], these social cues are learned through an imitative game, whereby the user imitates the robot. This allows the robot to build a user-specific model of these social pointers, and to become more robust at detecting them.

Finally, a core idea of HRI approach to PbD is that imitation is goal-directed, that is, actions are meant to fulfill a specific purpose and convey the intention of the actor [105]. While a longstanding trend in PbD approached the problem from the standpoint of trajectory following [106, 107, 108] and joint motion replication, see [81, 109, 110, 88] and Section 59.3.1, recent works, inspired by the above rationale, start from the assumption that imitation is not just about observing and replicating the motion, but rather about *understanding* the goals of a given action. Learning to imitate relies importantly on the imitator's capacity to infer the demonstrator's intentions [111, 89]. However, demonstrations may be ambiguous and extracting the intention of the demonstrator requires building a cognitive model of the demonstrator [51, 112], as well as exploiting other social cues to provide complementary knowledge [89, 54, 113].

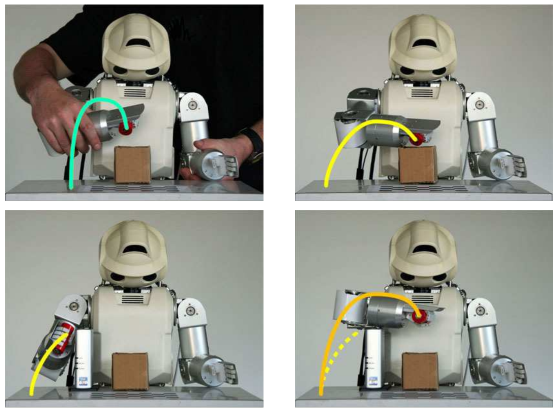


Figure 59.21: Illustration of the use of reinforcement learning to complement PbD. *Top-left:* The robot is trained through kinesthetic demonstration on a task that consists of placing a cylinder in a box. *Top-right:* The robot reproduces successfully the skill when the new situation is only slightly different from that of the demonstration, using the dynamical system described in Figure 59.16. *Bottom-left:* The robot fails at reproducing the skill when the context has changed importantly (a large obstacle has been placed in the way). *Bottom-right:* The robot relearns a new trajectory that reproduces the essential aspect of the demonstrated skill, i.e. putting the cylinder in the box, but avoiding the obstacle. Adapted from [117].

Understanding the way humans learn to both extract the goals of a set of observed actions and to give these goals a hierarchy of preference is fundamental to our understanding of the underlying decisional process of imitation. Recent work tackling these issues has followed a probabilistic approach to explain both a goal's derivation and sequential application. The explanation in turn makes it possible to learn manipulatory tasks that require the sequencing of a goal's subsets [114, 92, 106, 115].

Understanding the goal of the task is still only half of the picture, as there may be several ways of achieving the goal. Moreover, what is good for the demonstrator may not necessarily be good for the imitator [33]. Thus, different models may be allowed to compete to find a solution that is optimal both from the point of view of the imitator and that of the demonstrator [116, 77].

59.3.4 Joint Use of Robot PbD with Other Learning Techniques

To recall, a main argument for the development of PbD methods was that it would speed up learning by providing an example of "good solution". This, however, is true in sofar that the context for the reproduction is sufficiently similar to that of the demonstration. We have seen in Section 59.3.1 that the use of dynamical systems allows the robot to depart to some extent from a learned trajectory to reach for the target, even when both the object and the hand of the robot have moved from the location shown during the demonstration. This approach would not work in some situations, for example, when placing a large obstacle in the robot's path, see Figure 59.21. Besides, robots and humans may differ significantly in their kinematics and dynamics of motion and, although there are varieties of ways to bypass the so-called correspondence problem (Figure 59.5), relearning a new model may still be required in special cases.

To allow the robot to learn how to perform a task again in any new situation, it appeared important to combine PbD methods with other motor learning techniques. *Reinforcement learning* (RL) appeared particularly indicated for this type of problem, see Figures 59.18 and 59.21.

Early work on PbD using RL began in the 1990s and featured learning how to control an inverse pendulum and make it swing up [83]. More recent efforts [118, 119, 120] have focused on the robust control of the upper body of humanoid robots while performing various manipulation tasks.

One can also create a population of agents that copy (mimic) each other so that robots can learn a control strategy by making experiments themselves and by watching others. Such an evolutionary approach, using for example genetic algorithms, has been investigated by a number of authors, e.g. for learning manipulation skills [121], navigation strategies [51] or sharing a common vocabulary to name sensoriperception and actions [122].

59.4 Biologically-Oriented Learning Approaches

Another important trend in robot PbD takes a more biological stance and develops computational models of imitation learning in animals. We here briefly review recent progresses in this area.

59.4.1 Conceptual Models of Imitation Learning

Bioinspiration is first revealed in the conceptual schematic of the sensorimotor flow which is at the basis of imitation learning that some authors in PbD have followed over the years.

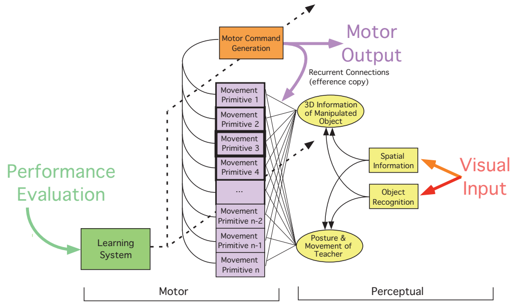


Figure 59.22: Conceptual sketch of an imitation learning system. The right side of the figure contains primarily perceptual elements and indicates how visual information is transformed into spatial and object information. The left side focuses on motor elements, illustrating how a set of movement primitives competes for a demonstrated behavior. Motor commands are generated from input of the most appropriate primitive. Learning can adjust both movement primitives and the motor command generator. Adapted from [123].

Figure 59.22 sketches the major ingredients of such a conceptual imitation learning system based on sensorimotor representation [123]. Visual sensory information needs to be parsed into information about objects and their spatial location in an internal or external coordinate system; the depicted organization is largely inspired by the dorsal (what) and ventral (where) stream as discovered in neuroscientific research [124]. As a result, the posture of the teacher and/or the position of the object while moving (if one is involved) should become available. Subsequently, one of the major questions revolves around how such information can be converted into action. For this purpose, Figure 59.22 alludes to the concept of movement primitives, also called "movement schemas", "basis behaviors", "units of action", or "macro actions" [125, 34, 126, 127]. Movement primitives are sequences of action that accomplish a complete goal-directed behavior. They could be as simple as an elementary action of an actuator (e.g., "go forward", "go backward", etc.), but, as discussed in [123], such low-level representations do not scale well to learning in systems with many degrees-of-freedom. Thus, it is useful for a movement primitive to code complete temporal behaviors, like "grasping a cup", "walking", "a tennis serve", etc. Figure 59.22 assumes that the perceived action of the teacher is mapped onto a set of existing primitives in an assimilation phase, which is also suggested in [128, 129]. This mapping process also needs to resolve the correspondence problem concerning a mismatch between the teacher's body and the student's body [130]. Subsequently, one can adjust the most appropriate primitives by learning to improve the performance in an accommodation phase. Figure 59.22 indicates such a

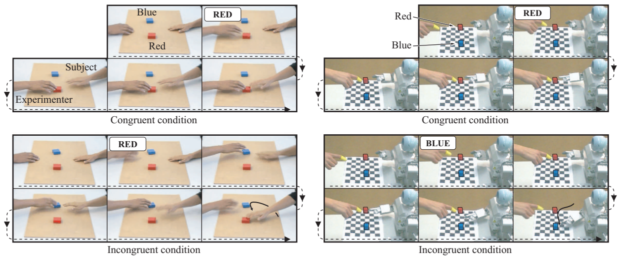


Figure 59.23: A human experimenter, a human imitator and a robot imitator play a simple imitation game, in which the imitator must point to the object named by the experimenter and not to the object which the experimenter points at. The robot's decision process is controlled by a neural model similar to that found in humans. As a result, it experiences the same associated deficit, known as the principle of *ideomotor compatibility*, stating that *observing the movements of others influences the quality of one's own performance*. When presented with conflicting cues (incongruent condition), e.g. when the experimenter points at a different object than the one named, the robot, like the human subject, either fails to reach for the correct object, or hesitates and later corrects the movement. Adapted from [132].

process by highlighting the better-matching primitives with increasing line widths. If no existing primitive is a good match for the observed behavior, a new primitive must be generated. After an initial imitation phase, self-improvement, e.g., with the help of a reinforcement-based performance evaluation criterion [131], can refine both movement primitives and an assumed stage of motor command generation (see below) until a desired level of motor performance is achieved.

59.4.2 Neural Models of Imitation Learning

Bioinspiration is also revealed in the development of neural network models of the mechanisms at the basis of imitation learning. Current models of Imitation Learning all ground their approach on the idea that imitation is at core driven by a *mirror neuron* system. The mirror neuron system refers to a network of brain areas in premotor and parietal cortices that is activated by both the recognition and the production of the same kind of object oriented movements performed by oneself and by others. See [133, 26, 134] for recent reports on this system in monkeys and humans, and its link to imitation.

Models of the mirror neuron system assume that somewhere, sensory information about the motion of others and about self-generated motion is coded using the same representation and in a common brain area. The models, however, differ in the way they represent this common center of information. Besides, while all models draw from the evidence of the existence of a mirror neuron circuit and of its application to explain multimodal sensory-motor processing, they, however, go further and tackle the issue of how the brain manages to process the flow of sensorimotor information that is at the basis of how one observes and produces actions. For a comprehensive and critical review of computational models of the mirror neuron system, the reader may refer to [135]. Next, we briefly review the most recent works among those.

One of the first approaches to explaining how the brain processes visuomotor control and imitation was based on the idea that the brain relies on Forward Models to compare, predict and generate motions [129]. These early works set the ground for some of the current neural models of the visuomotor pathway underlying imitation. For instance, recent work by Demiris and colleagues [128, 116] combines the evidence there is a mirror neuron system which is the basis of recognition and production of basic grasping motions and which proves the existence of forward models for guiding these motions. These models contribute both to the understanding of the mirror neuron system (MNS) in animals as well as its use in controlling robots. For instance, the models successfully reproduce the timing of neural activity in animals while observing various grasping motions and reproducing the kinematics of arm motions during these movements. The models were implemented to control reaching and grasping motions in humanoid and non-humanoid robots.

In contrast, Arbib and colleagues' models take a strong biological stance and follow an evolutionary approach to modeling the Mirror Neuron System (MNS), emphasizing its application to robotics in a second stage only. They hypothesize that the ability to imitate we find in humans has evolved from Monkeys' ability to reach and grasp and from Chimpanzees' ability to perform simple imitations. They develop models of each of these evolutionary stages, starting from a detailed model of the neural substrate underlying Monkeys' ability to reach and grasp [136], extending this model to include the Monkey MNS [137] and finally moving to models of the neural circuits underlying a human's ability to imitate [138]. The models replicate findings of brain imaging and cell recording studies, as well as make predictions on the time course of the neural activity for motion prediction. As such, the models reconciliate a view of forward models of action with an immediate MNS representation of these same actions. They were used to control reaching and grasping motions in humanoid robotic arms and hands [139].

Expanding on the concept of a MNS for sensorimotor coupling, Sauser & Billard [140, 141] have explored the use of competitive neural fields to explain the dynamics underlying multimodal representation of sensory information and the way the brain may disambiguate and select across competitive sensory stimuli to proceed to a given motor program. This work explains the principle of *ideomotor compatibility*, by which "observing the movements of others influences the quality of one's own performance", and develops neural models which account for a set of related behavioral studies [142], see Figure 59.23. The model expands the basic mirror neuron circuit to explain the consecutive stages of sensory-sensory and sensory-motor processing at the basis of this phenomenon. Sauser et al [132] discuss how the capacity for ideomotor facilitation can provide a robot with human-like behavior at the expense of several disadvantages such as hesitation and even mistakes, see Figure 59.23.

59.5 Conclusions and Open Issues in Robot PbD

This chapter aimed at assessing recent progress in modelling the cognitive or neural mechanisms underlying imitation learning in animals and the application of these models to controlling robots, on the one hand. On the other hand, it summarized various machine learning and computational approaches to providing the necessary algorithms for robot programming by demonstration.

Key questions remaining to be assessed by the field are:

- Can imitation use known motor learning techniques or does it require the development of new learning and control policies?
- How does imitation contribute and complement motor learning?
- Does imitation speed up skill learning?
- What are the costs of imitation learning?
- Do models of human kinematics used in gesture recognition drive the reproduction of the task?
- Can one find a level of representation of movement common to both gesture recognition and motor control?
- How could a model extract the intent of the user's actions from watching a demonstration?
- How can we create efficient combinations of imitation learning and reinforcement learning, such that systems can learn in rather few trials?
- What is the role of imitation in human-robot interactions?
- How to find a good balance between providing enough prior knowledge for learning to be fast and incremental, as in task/symbolic learning, and avoid restricting too much the span of learning?

In conclusion, robot PbD contributes to major advances in robot learning and paves the way to the development of robust controllers for both service and personal robots.

Bibliography

[1] S. Calinon, F. Guenter, and A. Billard. On learning, representing and generalizing a task in a humanoid robot. *IEEE Transactions on Systems, Man and Cybernetics, Part B. Special issue on robot learning by observation, demonstration and imitation*, 37(2):286–298, 2007.

[2] T. Lozano-Perez. Robot programming. *Proceedings of the IEEE*, 71(7):821–841, 1983.

[3] B. Dufay and J.-C. Latombe. An approach to automatic robot programming based on inductive learning. *The International Journal of Robotics Research*, 3(4):3–20, 1984.

[4] A. Levas and M. Selfridge. A user-friendly high-level robot teaching system. In *Proceeding of the IEEE International Conference on Robotics*, pages 413–416, March 1984.

[5] A.B. Segre and G. DeJong. Explanation-based manipulator learning: Acquisition of planning ability through observation. In *IEEE Conference on Robotics and Automation (ICRA)*, pages 555–560, March 1985.

[6] A.M Segre. *Machine learning of robot assembly plans*. Kluwer Academic Publishers, Boston, MA, USA, 1988.

[7] Y. Kuniyoshi, Y. Ohmura, K. Terada, A. Nagakubo, S. Eitoku, and T. Yamamoto. Embodied basis of invariant features in execution and perception of whole-body dynamic actionsknacks and focuses of roll-and-rise motion. *Robotics and Autonomous Systems*, 48(4):189–201, 2004.

[8] S. Muench, J. Kreuziger, M. Kaiser, and R. Dillmann. Robot programming by demonstration (RPD) - Using machine learning and user interaction methods for the development of easy and comfortable robot programming systems. In *Proceedings of the International Symposium on Industrial Robots (ISIR)*, pages 685–693, 1994.

[9] Y. Kuniyoshi, M. Inaba, and H. Inoue. Teaching by showing: Generating robot programs by visual observation of human performance. In *Proceedings of the International Symposium of Industrial Robots*, pages 119–126, October 1989.

[10] Y. Kuniyoshi, M. Inaba, and H. Inoue. Learning by watching: Extracting reusable task knowledge from visual observation of human performance. *IEEE Transactions on Robotics and Automation*, 10(6):799–822, 1994.

[11] S.B. Kang and K. Ikeuchi. A robot system that observes and replicates grasping tasks. In *Proceedings of the International Conference on Computer Vision (ICCV)*, pages 1093–1099, June 1995.

[12] C.P. Tung and A.C. Kak. Automatic learning of assembly task using a DataGlove system. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 1–8, 1995.

[13] K. Ikeuchi and T. Suchiro. Towards an assembly plan from observation, part I: Assembly task recognition using face-contact relations (polyhedral objects). In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 3, pages 2171–2177, May 1992.

[14] M. Ito, K. Noda, Y. Hoshino, and J. Tani. Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. *Neural Networks*, 19(3):323–337, 2006.

[15] T. Inamura, N. Kojo, and M. Inaba. Situation recognition and behavior induction based on geometric symbol representation of multimodal sensorimotor patterns. In *Proceedings of the IEEE/RSJ international Conference on Intelligent Robots and Systems (IROS)*, pages 5147–5152, October 2006.

[16] S. Liu and H. Asada. Teaching and learning of deburring robots using neural networks. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 3, pages 339–345, May 1993.

[17] A. Billard and G. Hayes. Drama, a connectionist architecture for control and learning in autonomous robots. *Adaptive Behavior*, 7(1):35–64, 1999.

[18] M. Kaiser and R. Dillmann. Building elementary robot skills from human demonstration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 3, pages 2700–2705, April 1996.

[19] R. Dillmann, M. Kaiser, and A. Ude. Acquisition of elementary robot skills from human demonstration. In *Proceedings of the International Symposium on Intelligent Robotic Systems (SIRS)*, pages 1–38, July 1995.

[20] J. Yang, Y. Xu, and C.S. Chen. Hidden Markov model approach to skill learning and its application in telerobotics. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 396–402, May 1993.

[21] P.K. Pook and D.H. Ballard. Recognizing teleoperated manipulations. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 578–585, May 1993.

[22] G.E. Hovland, P. Sikka, and B.J. McCarragher. Skill acquisition from human demonstration using a hidden Markov model. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2706–2711, 1996.

[23] S.K. Tso and K.P. Liu. Hidden Markov model for intelligent extraction of robot trajectory command from demonstrated trajectories. In *Proceedings of the IEEE International Conference on Industrial Technology (ICIT)*, pages 294–298, December 1996.

[24] C. Lee and Y. Xu. Online, interactive learning of gestures for human/robot interfaces. In *Proceedings of the IEEE international Conference on Robotics and Automation (ICRA)*, volume 4, pages 2982–2987, April 1996.

[25] G. Rizzolatti, L. Fadiga, L. Fogassi, and V. Gallese. Resonance behaviors and mirror neurons. *Archives Italiennes de Biologie*, 137(2-3):85–100, May 1999.

[26] J. Decety, T. Chaminade, J. Grezes, and A.N. Meltzoff. A PET exploration of the neural mechanisms involved in reciprocal imitation. *Neuroimage*, 15(1):265–272, 2002.

[27] J. Piaget. *Play, Dreams and Imitation in Childhood*. Norton, New York, USA, 1962.

[28] J. Nadel, C. Guerini, A. Peze, and C. Rivet. The evolving nature of imitation as a format for communication. In *Imitation in Infancy*, pages 209–234. Cambridge University Press, 1999.

[29] M. J Matarić. Sensory-motor primitives as a basis for imitation: Linking perception to action and biology to robotics. In Chrystopher Nehaniv and Kerstin Dautenhahn, editors, *Imitation in Animals and Artifacts*. The MIT Press, 2002.

[30] S. Schaal. Nonparametric regression for learning nonlinear transformations. In H. Ritter and O. Holland, editors, *Prerational Intelligence in Strategies, High-Level Processes and Collective Behavior*. Kluwer Academic Press, 1999.

[31] A. Billard. Imitation: a means to enhance learning of a synthetic proto-language in an autonomous robot. In K. Dautenhahn and C. Nehaniv, editors, *Imitation in Animals and Artifacs*, pages 281–311. MIT Press, 2002.

[32] K. Dautenhahn. Getting to know each other - Artificial social intelligence for autonomous robots. *Robotics and Autonomous Systems*, 16(2-4):333–356, 1995.

[33] C. Nehaniv and K. Dautenhahn. Of hummingbirds and helicopters: An algebraic framework for interdisciplinary studies of imitation and its applications. In J. Demiris and A. Birk, editors, *Interdisciplinary Approaches to Robot Learning*, volume 24, pages 136–161. World Scientific Press, 2000.

[34] C.L. Nehaniv. Nine billion correspondence problems and some methods for solving them. In *Proceedings of the International Symposium on Imitation in Animals and Artifacts (AISB)*, pages 93–95, 2003.

[35] P. Bakker and Y. Kuniyoshi. Robot see, robot do : An overview of robot imitation. In *Proceedings of the workshop on Learning in Robots and Animals (AISB)*, pages 3–11, April 1996.

[36] M. Ehrenmann, O. Rogalla, R. Zoellner, and R. Dillmann. Teaching service robots complex tasks: Programming by demonstation for workshop and household environments. In *Proceedings of the IEEE International Conference on Field and Service Robotics (FRS)*, 2001.

[37] M. Skubic and R.A. Volz. Acquiring robust, force-based assembly skills from human demonstration. In *IEEE Transactions on Robotics and Automation*, volume 16, pages 772–781, 2000.

[38] M. Yeasin and S. Chaudhuri. Toward automatic robot programming: learning human skill from visual data. *IEEE Transactions on Systems, Man and Cybernetics, Part B*, 30(1):180–185, 2000.

[39] J. Zhang and B. Rössler. Self-valuing learning and generalization with application in visually guided grasping of complex objects. *Robotics and Autonomous Systems*, 47(2-3):117–127, 2004.

[40] J. Aleotti and S. Caselli. Trajectory clustering and stochastic approximation for robot programming by demonstration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1029–1034, August 2005.

[41] A. Alissandrakis, C.L. Nehaniv, K. Dautenhahn, and J. Saunders. Evaluation of robot imitation attempts: Comparison of the system's and the human's perspectives. In *Proceedings of the ACM SIGCHI/SIGART conference on Human-robot interaction (HRI)*, pages 134–141, March 2006.

[42] N. Delson and H. West. Robot programming by human demonstration: Adaptation and inconsistency in constrained motion. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 30–36, 1996.

[43] A. Kheddar. Teleoperation based on the hidden robot concept. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 31(1):1–13, 2001.

[44] R. Dillmann. Teaching and learning of robot tasks via observation of human performance. *Robotics and Autonomous Systems*, 47(2-3):109–116, 2004.

[45] J. Aleotti, S. Caselli, and M. Reggiani. Leveraging on a virtual environment for robot programming by demonstration. *Robotics and Autonomous Systems*, 47(2-3):153–161, 2004.

[46] S. Ekvall and D. Kragic. Grasp recognition for programming by demonstration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 748–753, April 2005.

[47] T. Sato, Y. Genda, H. Kubotera, T. Mori, and T. Harada. Robot imitation of human motion based on qualitative description from multiple measurement of human and environmental data. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 2377–2384, 2003.

[48] K. Ogawara, J. Takamatsu, H. Kimura, and K. Ikeuchi. Extraction of essential interactions through multiple observations of human demonstrations. *IEEE Transactions on Industrial Electronics*, 50(4):667–675, 2003.

[49] M.N. Nicolescu and M.J. Mataric. Natural methods for robot task learning: Instructive demonstrations, generalization and practice. In *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 241–248, 2003.

[50] M. Pardowitz, R. Zoellner, S. Knoop, and R. Dillmann. Incremental learning of tasks from user demonstrations, past experiences and vocal comments. *IEEE Transactions on Systems, Man and Cybernetics, Part B. Special issue on robot learning by observation, demonstration and imitation*, 37(2):322–332, 2007.

[51] B. Jansen and T. Belpaeme. A computational model of intention reading in imitation. *Robotics and Autonomous Systems*, 54(5):394–402, 2006.

[52] S. Ekvall and D. Kragic. Learning task models from multiple human demonstrations. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 358–363, September 2006.

[53] H. Friedrich, S. Muench, R. Dillmann, S. Bocionek, and M. Sassin. Robot programming by demonstration (RPD): Supporting the induction by human interaction. *Machine Learning*, 23(2):163–189, 1996.

[54] J. Saunders, C.L. Nehaniv, and K. Dautenhahn. Teaching robots by moulding behavior and scaffolding the environment. In *Proceedings of the ACM SIGCHI/SIGART conference on Human-Robot Interaction (HRI)*, pages 118–125, March 2006.

[55] A. Alissandrakis, C.L. Nehaniv, and K. Dautenhahn. Correspondence mapping induced state and action metrics for robotic imitation. *IEEE Transactions on Systems, Man and Cybernetics, Part B. Special issue on robot learning by observation, demonstration and imitation*, 37(2):299–307, 2007.

[56] A. Ude. Trajectory generation from noisy positions of object features for teaching robot paths. *Robotics and Autonomous Systems*, 11(2):113–127, 1993.

[57] J. Yang, Y. Xu, and C.S. Chen. Human action learning via hidden Markov model. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 27(1):34–44, 1997.

[58] K. Yamane and Y. Nakamura. Dynamics filter - Concept and implementation of online motion generator for human figures. *IEEE Transactions on Robotics and Automation*, 19(3):421–432, 2003.

[59] A.J. Ijspeert, J. Nakanishi, and S. Schaal. Learning control policies for movement imitation and movement recognition. In *Neural Information Processing System (NIPS)*, volume 15, pages 1547–1554, 2003.

[60] S. Vijayakumar and S. Schaal. Locally weighted projection regression: An O(n) algorithm for incremental real time learning in high dimensional spaces. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 288–293, 2000.

[61] S. Vijayakumar, A. D'souza, and S. Schaal. Incremental online learning in high dimensions. *Neural Computation*, 17(12):2602–2634, 2005.

[62] N. Kambhatla. *Local Models and Gaussian Mixture Models for Statistical Data Processing*. PhD thesis, Oregon Graduate Institute of Science and Technology, Portland, OR, 1996.

[63] A. Shon, K. Grochow, and R. Rao. Robotic imitation from human motion capture using Gaussian processes. In *Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids)*, 2005.

[64] K. Grochow, S.L. Martin, A. Hertzmann, and Z. Popovic. Style-based inverse kinematics. In *Proceedings of the ACM International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, pages 522–531, 2004.

[65] K.F. MacDorman, R. Chalodhorn, and M. Asada. Periodic nonlinear principal component neural networks for humanoid motion segmentation, generalization, and generation. In *Proceedings of the International Conference on Pattern Recognition (ICPR)*, volume 4, pages 537–540, August 2004.

[66] D. Bullock and S. Grossberg. VITE and FLETE: neural modules for trajectory formation and postural control. In W.A. Hersberger, editor, *Volitional control*, pages 253–297. Elsevier, 1989.

[67] F.A. Mussa-Ivaldi. Nonlinear force fields: a distributed system of control primitives for representing and learning movements. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 84–90. 1997.

[68] P. Li and R. Horowitz. Passive velocity field control of mechanical manipulators. *IEEE Transactions on Robotics and Automation*, 15(4):751–763, 1999.

[69] G. Schoener and C. Santos. Control of movement time and sequential action through attractor dynamics: a simulation study demonstrating object interception and coordination. In *Proceedings of the International Symposium on Intelligent Robotic Systems (SIRS)*, 2001.

[70] T. Inamura, H. Tanie, and Y. Nakamura. Keyframe compression and decompression for time series data based on continuous hidden Markov models. In *Proceedings of the IEEE/RSJ international Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 1487–1492, 2003.

[71] S. Calinon and A. Billard. What is the teacher's role in robot programming by demonstration? - Toward benchmarks for improved learning. *Interaction Studies. Special Issue on Psychological Benchmarks in Human-Robot Interaction*, 8(3):441–464, 2007.

[72] T. Inamura, I. Toshima, and Y. Nakamura. Acquiring motion elements for bidirectional computation of motion recognition and generation. In B. Siciliano and P. Dario, editors, *Experimental Robotics VIII*, volume 5, pages 372–381. Springer-Verlag, 2003.

[73] T. Inamura, N. Kojo, T. Sonoda, K. Sakamoto, K. Okada, and M. Inaba. Intent imitation using wearable motion capturing system with online teaching of task attention. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 469–474, 2005.

[74] D. Lee and Y. Nakamura. Stochastic model of imitating a new observed motion based on the acquired motion primitives. In *Proceedings of the IEEE/RSJ international Conference on Intelligent Robots and Systems (IROS)*, pages 4994–5000, October 2006.

[75] D. Lee and Y. Nakamura. Mimesis scheme using a monocular vision system on a humanoid robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2162–2168, April 2007.

[76] S. Calinon and A. Billard. Learning of gestures by imitation in a humanoid robot. In K. Dautenhahn and C.L. Nehaniv, editors, *Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communicative Dimensions*, pages 153–177. Cambridge University Press, 2007.

[77] A. Billard, S. Calinon, and F. Guenter. Discriminative and adaptive imitation in uni-manual and bi-manual tasks. *Robotics and Autonomous Systems*, 54(5):370–384, 2006.

[78] S. Calinon and A. Billard. Recognition and reproduction of gestures using a probabilistic framework combining PCA, ICA and HMM. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 105–112, August 2005.

[79] S. Calinon, F. Guenter, and A. Billard. Goal-directed imitation in a humanoid robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 299–304, April 2005.

[80] T. Asfour, F. Gyarfas, P. Azad, and R. Dillmann. Imitation learning of dual-arm manipulation tasks in humanoid robots. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 40–47, December 2006.

[81] J. Aleotti and S. Caselli. Robust trajectory learning and approximation for robot programming by demonstration. *Robotics and Autonomous Systems*, 54(5):409–413, 2006.

[82] C.G. Atkeson. Using local models to control movement. In *Advances in Neural Information Processing Systems (NIPS)*, volume 2, pages 316–323, 1990.

[83] C.G. Atkeson, A.W. Moore, and S. Schaal. Locally weighted learning for control. *Artificial Intelligence Review*, 11(1-5):75–113, 1997.

[84] A.W. Moore. Fast, robust adaptive control by learning only forward models. In *Advances in Neural Information Processing Systems (NIPS)*, volume 4, San Francisco, CA, USA, 1992. Morgan Kaufmann.

[85] S. Schaal and C.G. Atkeson. From isolation to cooperation: An alternative view of a system of experts. In *Advances in Neural Information Processing Systems (NIPS)*, volume 8, pages 605–611, San Francisco, CA, USA, 1996. Morgan Kaufmann Publishers Inc.

[86] S. Schaal and C.G. Atkeson. Constructive incremental learning from only local information. *Neural Computation*, 10(8):2047–2084, 1998.

[87] M. Hersch, F. Guenter, S. Calinon, and A.G. Billard. Learning dynamical system modulation for constrained reaching tasks. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 444–449, December 2006.

[88] A.J. Ijspeert, J. Nakanishi, and S. Schaal. Movement imitation with nonlinear dynamical systems in humanoid robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1398–1403. 2002.

[89] C. Breazeal, M. Berlin, A. Brooks, J. Gray, and A.L. Thomaz. Using perspective taking to learn from ambiguous demonstrations. *Robotics and Autonomous Systems*, 54(5):385–393, 2006.

[90] Y. Sato, K. Bernardin, H. Kimura, and K. Ikeuchi. Task analysis based on observing hands and objects by vision. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1208–1213, 2002.

[91] R. Zoellner, M. Pardowitz, S. Knoop, and R. Dillmann. Towards cognitive robots: Building hierarchical task representations of manipulations from human demonstration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1535–1540, 2005.

[92] M. Pardowitz, R. Zoellner, and R. Dillmann. Incremental learning of task sequences with information-theoretic metrics. In *Proceedings of the European Robotics Symposium (EUROS)*, December 2005.

[93] M. Pardowitz, R. Zoellner, and R. Dillmann. Learning sequential constraints of tasks from user demonstrations. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 424–429, December 2005.

[94] S. Calinon and A. Billard. Teaching a humanoid robot to recognize and reproduce social cues. In *Proceedings of the IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 346–351, September 2006.

[95] Brian Scassellati. Imitation and mechanisms of joint attention: A developmental structure for building social skills on a humanoid robot. *Lecture Notes in Computer Science*, 1562:176–195, 1999.

[96] H. Kozima and H. Yano. A robot that learns to communicate with human caregivers. In *Proceedings of the International Workshop on Epigenetic Robotics*, September 2001.

[97] H. Ishiguro, T. Ono, M. Imai, and T. Kanda. Development of an interactive humanoid robot robovie - An interdisciplinary approach. *Robotics Research*, 6:179–191, 2003.

[98] K. Nickel and R. Stiefelhagen. Pointing gesture recognition based on 3D-tracking of face, hands and head orientation. In *Proceedings of the international conference on Multimodal interfaces (ICMI)*, pages 140–146, 2003.

[99] M. Ito and J. Tani. Joint attention between a humanoid robot and users in imitation game. In *Proceedings of the International Conference on Development and Learning (ICDL)*, 2004.

[100] V.V. Hafner and F. Kaplan. Learning to interpret pointing gestures: experiments with four-legged autonomous robots. In S. Wermter, G. Palm, and M. Elshaw, editors, *Biomimetic Neural Learning for Intelligent Robots. Intelligent Systems, Cognitive Robotics, and Neuroscience*, pages 225–234. Springer Verlag, 2005.

[101] C. Breazeal, D. Buchsbaum, J. Gray, D. Gatenby, and B. Blumberg. Learning from and about others: Towards using imitation to bootstrap the social understanding of others by robots. *Artificial Life*, 11(1-2), 2005.

[102] P.F. Dominey, M. Alvarez, B. Gao, M. Jeambrun, A. Cheylus, A. Weitzenfeld, A. Martinez, and A. Medrano. Robot command, interrogation and teaching via social interaction. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 475–480, December 2005.

[103] A.L. Thomaz, M. Berlin, and C. Breazeal. Robot science meets social science: An embodied computational model of social referencing. In *Workshop Toward Social Mechanisms of Android Science (CogSci)*, pages 7–17, July 2005.

[104] C. Breazeal and L. Aryananda. Recognition of affective communicative intent in robot-directed speech. *Autonomous Robots*, 12(1):83–104, 2002.

[105] H. Bekkering, A. Wohlschlaeger, and M. Gattis. Imitation of gestures in children is goal-directed. *Quarterly Journal of Experimental Psychology*, 53A(1):153–164, 2000.

[106] M. Nicolescu and M.J. Mataric. Task learning through imitation and human-robot interaction. In K. Dautenhahn and C.L. Nehaniv, editors, *in Models and Mechanisms of Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communicative Dimensions*, pages 407–424. Cambridge University Press, 2007.

[107] J. Demiris and G. Hayes. Imitative learning mechanisms in robots and humans. In V. Klingspor, editor, *Proceedings of the European Workshop on Learning Robots*, pages 9–16, 1996.

[108] P. Gaussier, S. Moga, J.P. Banquet, and M. Quoy. From perception-action loop to imitation processes: a bottom-up approach of learning by imitation. *Applied Artificial Intelligence*, 7(1):701–729, 1998.

[109] M. Ogino, H. Toichi, Y. Yoshikawa, and M. Asada. Interaction rule learning with a human partner based on an imitation faculty with a simple visuo-motor mapping. *Robotics and Autonomous Systems*, 54(5):414–418, 2006.

[110] A. Billard and M. Matarić. Learning human arm movements by imitation: Evaluation of a biologically-inspired connectionist architecture. *Robotics and Autonomous Systems*, 37(2):145–160, 2001.

[111] W. Erlhagen, A. Mukovskiy, E. Bicho, G. Panin, C. Kiss, A. Knoll, H. van Schie, and H. Bekkering. Goal-directed imitation for robots: A bio-inspired approach to action understanding and skill learning. *Robotics and Autonomous Systems*, 54(5):353–360, 2006.

[112] A. Chella, H. Dindo, and I. Infantino. A cognitive framework for imitation learning. *Robotics and Autonomous Systems*, 54(5):403–408, 2006.

[113] S. Calinon and A. Billard. Incremental learning of gestures by imitation in a humanoid robot. In *Proceedings of the ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 255–262, March 2007.

[114] R.H. Cuijpers, H.T. van Schie, M. Koppen, W. Erlhagen, and H. Bekkering. Goals and means in action observation: A computational approach. *Neural Networks*, 19(3):311–322, 2006.

[115] M.W. Hoffman, D.B. Grimes, A.P. Shon, and R.P.N. Rao. A probabilistic model of gaze imitation and shared attention. *Neural Networks*, 19(3):299–310, 2006.

[116] Y. Demiris and B. Khadhouri. Hierarchical attentive multiple models for execution and recognition of actions. *Robotics and Autonomous Systems*, 54(5):361–369, 2006.

[117] F. Guenter, M. Hersch, S. Calinon, and A. Billard. Reinforcement learning for imitating constrained reaching movements. *Advanced Robotics, Special Issue on Imitative Robots*, 2007. In press.

[118] J. Peters, S. Vijayakumar, and S. Schaal. Reinforcement learning for humanoid robotics. In *Proceedings of the IEEE International Conference on Humanoid Robots (Humanoids)*, October 2003.

[119] T. Yoshikai, N. Otake, I. Mizuuchi, M. Inaba, and H. Inoue. Development of an imitation behavior in humanoid Kenta with reinforcement learning algorithm based on the attention during imitation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1192–1197, September-October 2004.

[120] D.C. Bentivegna, C.G. Atkeson, and G. Cheng. Learning tasks from observation and practice. *Robotics and Autonomous Systems*, 47(2-3):163–169, 2004.

[121] Y.K. Hwang, K.J. Choi, and D.S. Hong. Self-learning control of cooperative motion for a humanoid robot. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 475–480, May 2006.

[122] A. Billard and K. Dautenhahn. Grounding communication in autonomous robots: an experimental study. *Robotics and Autonomous Systems, Special Issue on Scientific methods in mobile robotics*, 24(1-2):71–79, 1998.

[123] S. Schaal. Is imitation learning the route to humanoid robots? *Trends in Cognitive Sciences*, 3(6):233–242, 1999.

[124] J.H. Maunsell and D.C. Van Essen. Functional properties of neurons in middle temporal visual area of the macaque monkey. II. Binocular interactions and sensitivity to binocular disparity. *Neurophysiology*, 49(5):1148–1167, 1983.

[125] M. Arbib, T. Iberall, and D. Lyons. Coordinated control program for movements of the hand. *Experimental brain research Suppl.*, 10:111–129, 1985.

[126] D. Sternad and S. Schaal. Segmentation of endpoint trajectories does not imply segmented control. *Experimental Brain Research*, 124/1:118–136, 1999.

[127] R.S. Sutton, S. Singh, D. Precup, and B. Ravindran. Improved switching among temporally abstract actions. In *Advances in Neural Information Processing Systems (NIPS)*, volume 11, pages 1066–1072, 1999.

[128] Y. Demiris and G. Hayes. Imitation as a dual route process featuring predictive and learning components: a biologically-plausible computational model. In K. Dautenhahn and C. Nehaniv, editors, *Imitation in Animals and Artifacs*, pages 327–361. MIT Press, 2002.

[129] D.M. Wolpert and M. Kawato. Multiple paired forward and inverse models for motor control. *Neural Networks*, 11(7-8):1317–1329, 1998.

[130] C. Nehaniv and K. Dautenhahn. *Imitation in Animals and Artifacs*. MIT Press, Boston, USA, 2002.

[131] R.S. Sutton and A.G. Barto. *Reinforcement learning : an introduction*. Adaptive computation and machine learning. MIT Press, Cambridge, MA, USA, 1998.

[132] E. Sauser and A. Billard. Biologically inspired multimodal integration: Interferences in a human-robot interaction game. In *Proceedings of the IEEE/RSJ international Conference on Intelligent Robots and Systems (IROS)*, pages 5619–5624, October 2006.

[133] G. Rizzolatti, L. Fogassi, and V. Gallese. Neurophysiological mechanisms underlying the understanding and imitation of action. *Nature Reviews Neuroscience*, 2:661–670, 2001.

[134] M. Iacoboni, R.P. Woods, M. Brass, H. Bekkering, J.C. Mazziotta, and G Rizzolatti. Cortical mechanisms of human imitation. *Science*, 286:2526–2528, 1999.

[135] E. Oztop, M. Kawato, and M.A. Arbib. Mirror neurons and imitation: A computationally guided review. *Neural Networks*, 19(3):254–21, 2006.

[136] A.H. Fagg and M.A. Arbib. Modeling parietal-premotor interactions in primate control of grasping. *Neural Networks*, 11(7):1277–1303, 1998.

[137] E. Oztop and M.A. Arbib. Schema design and implementation of the grasp-related mirror neuron system. *Biological Cybernetics*, 87(2):116140, 2002.

[138] M. Arbib, A. Billard, M. Iacoboni, and E. Oztop. Mirror neurons, imitation and (synthetic) brain imaging. *Neural Networks*, 13(8-9):953–973, 2000.

[139] E. Oztop, M. Lin, M. Kawato, and G. Cheng. Dexterous skills transfer by extending human body schema to a robotic hand. In *Proceedings of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 82–87, December 2006.

[140] E. Sauser and A. Billard. Parallel and distributed neural models of the ideomotor principle: An investigation of imitative cortical pathways. *Neural Networks*, 19(3):285–298, 2006.

[141] E.L. Sauser and A.G. Billard. Dynamic updating of distributed neural representations using forward models. *Biological Cybernetics*, 95(6):567–588, 2006.

[142] M. Brass, H. Bekkering, A. Wohlschlaeger, and W. Prinz. Compatibility between observed and executed movements: Comparing symbolic, spatial and imitative cues. *Brain and Cognition*, 44(2):124–143, 2001.