

Task learning through teleoperation

Victor Barberte^{1,2}

Fumio Kanehiro²

Abstract—We provide a complete framework for learning, reproducing and generalizing tasks from human demonstrations. This method adapts recent developments in automatic, unsupervised segmentation of time-series to humanoid robotics by preprocessing the data obtained from a broad range of the robot’s sensors.

In more detail, we reproduce and extend the acquired multi-step task using Dynamic Movement Primitives in simulation for the JVR1 Robot, and further validate it in the real world with the Kaleido Robot, thus showcasing the capacity of our approach to create an extensive library of reusable skills for complex humanoids.

Index Terms—learning from demonstration, generalization, teleoperation, probabilistic inference, feature selection, dynamic movement primitives, sensory skill integration

I. INTRODUCTION

A. Learning from Demonstration

Demonstrations provide an intuitive way for non-professional users to specify tasks to the robot [1]. Most of robot’s controllers to date, albeit successful to achieve complex tasks, often require theoretical knowledge and is time-consuming. To address these issues, research about robot learning from demonstration (LfD) has become increasingly important over the past two decades with the surge of Machine Learning (ML) algorithms [2] [3]. In this framework, users can demonstrate a task that is then learned and generalized by the robot. Generalization is defined as the capacity to perform a learned task in a similar environment as the one of the demonstration, but under different initial or goal poses. LfD has been widely used and proved successful on robotic arms or in simulation [4], but are more seldom used on real humanoid robots [5] due to their high number of degrees of freedom (DoF) and unlinear behavior. A core idea of this work is to capitalize on the physiologic similarities between us, humans, and humanoid robots to come up with an intuitive LfD technique.

Note: There is little difference between LfD and imitation learning, mainly imitation learning requires embodiment mapping [2], that is to say a treatment of the data to transform it from high-level information (i.e, a video of the executed task) to low-level data (joint positions, torques) easily understandable for the robot. LfD provides directly this low-level data to the robot. In this article, we will employ both terms indifferently, as our main focus is how to deal with the low-level data in input of the robot.

Several techniques are found in the literature to provide demonstrations to the robot. This can be done through simulation [6], kinesthetic teaching [7], motion capture [8] or with teleoperation devices. These teleoperations devices appear under various shapes, from joysticks [9] to exoskeletons where the user remotely controls the robot [10]. In all the above cited works, some prefer 1-shot imitation learning, where the demonstration can be a seed for an initial policy that is then derived and learned through Reinforcement Learning (RL) [11] [12], Inverse RL [13] or Hierarchical RL [14] that can possibly be corrected with online coaching [15]. Another recent approach is to create more demonstrations with the initial one, thus enriching the dataset thanks to meta-learning processes [16]. On the contrary, other methods rely on a set of demonstrations to perform probabilistic inference based on Hidden Markov Models [17], Neural Networks [18] or using Granger Causality [19]. The former often derives only one policy that is locally optima, and thus the learning process is repeated for each task, whereas the latter methods prefer to decompose the task in elementary *skills* with segmentation methods, thus enhancing the reusability of the stored skills. Our proposed method is of the latter category, that is further detailed in I-B

Whatever the method used, once the task learned, the great majority of the references cited above use Dynamic Movement Primitives (DMP), that were first introduced by Ijspeert and Schaal in [20] and further explored in [21]. It provides a suitable framework to reproduce trajectories and to smoothly adapt to new conditions.

B. Related work

Splitting a complex task such as grasping an object or pouring water has the advantage of making the result more generalizable. Indeed, every part of the trajectory (every *skill*) has its own level of generalizability thanks to DMP, thus resulting in an intra-task generalization, better than a single end-to-end DMP. Nonetheless, a complex task is almost all the time demonstrated without specifying the number of segments: the demonstrator naturally achieves the task. Consequently, one challenge of this approach is to automatically segment the trajectory in human-interpretable skills that, once stored in a library, can be reused to learn new skills [22].

Recent methods provide plans to the robot about the different phases of the demonstration using natural language or computer vision [23] [24]. Besides, probabilistic approaches are employed to segment a demonstration in an unsupervised fashion. The number of phases involved as well as the temporal

¹Ecole Polytechnique, Palaiseau, France

²CNRS-AIST JRL (Joint Robotics Laboratory), IRL, Japan

position of changepoint (CP) are inferred automatically. These techniques include Hidden Markov Models (HMM) [25], Gaussian Mixture Regression (GMR) [26] [5] and more seldom reinforcement learning techniques [27]. Throughout the upcited references, the features that are selected to learn the task are manually selected by the user. Working with such specially chosen features often produces satisfactory results inasmuch as the skilled demonstrator knows which feature (positions, orientations, state of the gripper, ...) is relevant to study. However, the feature selection has to be rethought from scratch for each new task.

Contributions:

Our method addresses all the upcited issues in a comprehensive framework that combines state-of-the-art LfD techniques with unsupervised automatic segmentation to achieve robust task learning and generalization. We primarily fetch almost all the data available, whether it is kinetic or sensory, from a small number of demonstrations and automatically selects the most relevant features to study with Dynamic Time Warping (DTW) and correlation analysis. Segmentation is achieved with Bayesian Online Changepoint Detection with Online Detection (BOCPDMS) (or the one we will choose) and task reproduction and generalization is performed with DMP [20] [21]. This framework is applied in simulation on the JVRC1 and in real world on the Kaleido humanoid robot.

C. Outline

Section II details the theoretical aspects of the different steps of the proposed framework. Following that, section III delves into how the previously described concepts are applied to our learning and generalizing a task to humanoid robots. Section IV puts on display the results obtained and then compared to other existing methods. Limits and future work possibilities are finally explored in section V.

II. BACKGROUND

A. Bayesian Changepoint detection

Theory on Bayesian Online Changepoint Detection

B. Dynamic Movement Primitives

The concept of Discrete DMP was first introduced by Ispjeert and al. in 2002 [20], and more accurately developed in 2013 [21]. The discrete DMP is utilized to encode a point-to-point motion into a stable dynamical system (e.g, a damped oscillator).

A DMP for 1-DoF trajectory y of a discrete movement (point-to-point) is defined by the following set of nonlinear differential equations [20]:

$$\tau \dot{z} = \alpha_z (\beta_z (g - y) - z) + f(x), \quad (1)$$

$$\tau \dot{y} = z, \quad (2)$$

$$\tau \dot{x} = \alpha_x x, \quad (3)$$

where x is the phase variable and z is an auxiliary variable. Parameters α_z and β_z define the behavior of the second-order system described by Equations (1) and (2). Ensuring

conditions $\tau > 0$, $\alpha_z = 4\beta_z$, and $\alpha_x > 0$ leads the dynamical system to converge towards a unique attractor point at $y = g$, $z = 0$ [21]. Alternatively, the gains α_z and β_z can be learned from training data while preserving the convergence of the system (Tan et al. 2016). In the DMP literature, Equations (1)-(2) are often referred to as the transformation system, while Equation (3) represents the canonical system. $f(x)$ is defined as a linear combination of N nonlinear Radial Basis Functions (RBFs), which allows the robot to follow any smooth trajectory from y_0 to the goal g .

$$f(x) = \sum_{i=1}^N P_i \Psi_i(x) x, \quad (4)$$

$$\Psi_i(x) = \exp(-h_i(x - c_i)^2), \quad (5)$$

In Equation (4), P_i represents the weight associated with the i -th term, $\Psi_i(x)$ is defined as the i -th Radial Basis Function (RBF), and x is the phase variable. Each RBF, given by Equation (5), is characterized by a center c_i and a bandwidth parameter h_i . For a given N and setting τ equal to the total duration of the desired movement, we have $c_i = \exp\left(-\frac{\alpha_x(i-1)}{N-1}\right)$ and $h = \frac{1}{N(N-1)} \sum_{i=1}^{N-1} (c_{i+1} - c_i)^2$. Each degree of freedom (DoF) requires adjusting the weights w_i based on the measured data to achieve the desired behavior. The number of weights should be chosen based on the desired resolution of the trajectory. When controlling a robotic system with multiple DoFs, we represent the movement of each DoF using its own equation system (1)-(2), but with a common phase (3) to synchronize them.

For a discrete motion, given a demonstrated trajectory $y_d(t_k)$, where $t_k = 1, \dots, T$, and its time derivatives $\dot{y}_d(t_k)$ and $\ddot{y}_d(t_k)$, it is possible to invert Equation (1) and approximate the desired shape f_d as follows:

$$f_d(t_k) = \frac{\tau^2 \ddot{y}_d(t_k) - \alpha_z (\beta_z (g - y_d(t_k)) - \tau \dot{y}_d(t_k))}{\alpha_z}. \quad (6)$$

Equation 6 provides an approximation of the desired shape $f_d(t_k)$ based on the demonstrated trajectory and its time derivatives.

By stacking each $f_d(t_k)$ and w_i into the column vectors $F = \begin{pmatrix} f_d(t_1) \\ \vdots \\ f_d(t_T) \end{pmatrix}$ and $w = \begin{pmatrix} w_1 \\ \vdots \\ w_N \end{pmatrix}$, we obtain the following linear system:

$$\Phi w = F, \quad (7)$$

where

$$\Phi = \begin{bmatrix} \frac{\Psi_1(x_1)}{\sum_{i=1}^N \Psi_i(x_1)x_1} & \cdots & \frac{\Psi_N(x_1)}{\sum_{i=1}^N \Psi_i(x_1)x_1} \\ \vdots & \ddots & \vdots \\ \frac{\Psi_1(x_T)}{\sum_{i=1}^N \Psi_i(x_T)x_1} & \cdots & \frac{\Psi_N(x_T)}{\sum_{i=1}^N \Psi_i(x_T)x_1} \end{bmatrix} \quad (8)$$

The phase variable x in Equation (3) enables the user to slow or accelerate the motion: this is the phase-stopping mechanism [20]:

$$\frac{\tau_x \dot{x}}{\alpha_x} = -1 + \alpha_y x |\tilde{y} - y|, \quad (9)$$

Additionally, one useful feature of the DMP is the possibility to adapt the trajectory online if the goal changes [21]:

$$\tau_g \dot{g} = \alpha_g (g_0 - g). \quad (10)$$

However, the standard formulation of DMPs is not suitable for directly encoding skills with specific geometric constraints, such as orientation profiles represented by unit quaternions. For instance, direct integration of unit quaternions does not ensure the unity of the quaternion's norm. Thus, the equations of the classical discrete DMP have to be adapted in order to be generalizable to quaternions. The equations system can be found in this tutorial survey about DMP [28]. Even though the core ideas are the same, one difference is the introduction of the *quaternion logarithm* $\text{Log}^q(gq^*\bar{q})$ that has an impact when it comes to computation. This expression defines the angular velocity ω that rotates quaternion q into gq within a unit sampling time.

The function $\text{Log}_q(\cdot) : S^3 \rightarrow \mathbb{R}^3$ is defined as:

$$\text{Log}_q(q) = \begin{cases} \arccos(\nu) \frac{q}{\|q\|}, & \text{if } q \neq 0 \\ [0 \ 0 \ 0]^T, & \text{otherwise} \end{cases}$$

This definition maps a quaternion q from the unit sphere S^3 to a vector in \mathbb{R}^3 using the arccosine function and normalization.

III. LEARNING FROM DEMONSTRATIONS

Scheme of all the framework

A. Gathering demonstrations

Building a dataset that covers a wide range of configurations is essential to reach an efficient generalization of the task, hence providing multiple demonstrations to the robot. To do so, we demonstrated the same task in a similar environment (in the same room, with similar objects on the table, ...) but with different conditions. These include different position parameters (poses of the object, goal poses, grasping poses) as well as variate dynamic variables (speed of the movement). Having such set of demonstrations limits the *covariate shift*, that's to say the difference of distribution between the training dataset and the real-world cases. Nonetheless, we can not get rid of this covariate shift in the extent that the action space of a given task in the real world is most often far bigger than what can represent a few demonstrations. In addition to that, our approach is to have a dataset corresponding to what might come natural to a non-professional user.

Once the demonstrations gathered, the choice of the features to study is another core step that often requires the experience of a skilled user. Whereas a great majority of LfD techniques selects manually the relevant features to study, we chose to

take the measurements of all the sensors available as well as the proprioceptive information, namely 6D Cartesian position and orientation and angular and linear velocities obtained from the robot's solver. Mainly two reasons justify that:

- A non-skilled user has to be able to demonstrate a new task only by achieving it in a natural way while wearing the teleoperation device
- Capitalize on our similarity with humanoid robots. Indeed, integrating sensory parameters amounts to add a kind of *sensory integration* to the robot. *Sensory integration* was first theorized by Dr A. Jeans Ayres in 1972 in the field of neuroscience. It states that when thinking about achieving some known task, our brain processes not only how to reach the goal (the different steps), but also the sensations we had when priorly achieving this task. This idea was first exploited by P. Pastor in [29]. A telling example of sensory memory is that when we grab an object, the brain area corresponding to our sense of touch activates, and we expect to have some sensation when grabbing an object. In [29], a robot arm corrects his trajectory during grabbing by adding a term that corresponds to his expected force feedback on the gripper. Thus, taking sensor into account while building the dataset can help to detect new phases of the movement [30] and correcting the movement online (after the learning phase).

We first design a controller to build a dataset in simulation on the JVRC1 virtual humanoid. To that end, we designed a finite state machine (FSM) controller with B-splines with mc-rtc and mc-mujoco [31] to simulate a one-handed grasping task of a stick in 9 different configurations (figure with mujoco interface and 2D schema). Then, we build a real-world dataset, with the JRL's teleoperation device with the Kaleido robot (figure of experimental setup). The user has a the visual of the humanoid's front camera, and achieves the task naturally, except that there is no force feedback on the grippers, meaning that the demonstrator is not constrained by the physical shape of the object during the demonstrations.

In both cases, mc-rtc logs all the sensor and proprioceptive values that are then used to learn and generalize the task. The task can be represented by all its normalized feature values in skill maps (figure skill map). Using this map, we have to extract appropriate features and segment it into interpretable phases.

B. Feature Selection

Having gathered a set of demonstrations containing numerous features, we now want to automatically extract the most relevant features to segment the task in skills reproducible with DMP.

A first step is to temporally align the signals in order to compare the features across the dataset. Indeed, our goal is to find the most correlated features. Intuitively, if one feature has the same shape in all the trials, it is likely that it will be useful for segmentation, and that it is critical to define the task. For

example, in a grasping task, the opening of the gripper clearly identifies the different phases of the movement and has the same shape across all the trials. On the contrary, we expect that other parameters like the position along the z-axis have no importance in the grasping task.

To align the signals, we use Dynamical Time Warping (DTW). The main idea behind DTW is to find the optimal alignment between the two sequences by warping their time axes. This warping process allows for the matching of similar patterns, despite variations in timing and duration. It works by finding the minimum distance path through a grid or matrix that represents the pairwise distances between elements of the two sequences.

We can then compute a *correlation matrix* for each feature across all trials (figure of a correlation matrix). The mean of the coefficients of these matrices can therefore be interpreted as the similarity across the datasets. This first result, albeit convincing, include a bias due to the temporal alignment of the signals. Indeed, it is possible to end up with a highly correlated feature when the signals are aligned, but in reality very distant. In fact, the distance computed with DTW can also be used as a tool to measure similarity. Taking again the statistical mean of the two by two distances for every feature and normalizing the values between 0 and 1, we can penalize the correlation coefficients computed afterwards. We employ the normalized DTW distances as weights for the correlation coefficients. This allows to take into account the distance between the trials, eventually leading to a more objective measure of the relevance of the features. We see in (figure hist weighted) that the most correlated features in (hist normal) are even more detached from all the other features.

Finally, we select the features to pass in argument for the segmentation phase. There is a trade-off between selecting the most correlated features and discarding others - thus possibly missing important information - and choosing too much, - thus thwarting the segmentation with uncorrelated features -. In addition to the most correlated feature F , we select the features that are above $CorrCoef_F \times T$, with T a threshold that we set at 0.9.

Note: This correlation analysis is only relevant for demonstrations that are long enough (more than a few seconds). Otherwise, the p-value of the Pearson coefficient [32] is above 0.05 and the correlation is very likely to be irrelevant.

C. Segmentation

Parameters used, which library If we clean the data or not Once we have possible breakpoints, we filter them with sensory information (cite detection with sensory information for insertion task

D. Task reproduction

Explain how to recognize a task (shapelets, signature), prendre les points de demos les plus proches et extrapoler les poids

IV. RESULTS

A. Segmentation on multivariate raw data

TODO

B. Segmentation on multivariate preprocessed data

TODO

C. Task recognition and reproduction

How to recognize a task that has already been shown

V. DISCUSSION

A. Comparative study and limits

TODO take results of other articles 6d pose estimation statistical approach to select relevant features library of skills

B. Conclusion and future work

Throughout this work, we presented an end-to-end framework to learn and generalize a new task for humanoid robot by automatically segment demonstrations provided in a natural way through teleoperation into reproducible skills. In addition as being a novelty for complex humanoid robots, the proposed approach can be extended to every humanoid robot, and theoretically to every combination of end-effectors (one hand, two hands, legs). On top of that, automatically extracting the relevant features to learn the task with a correlation analysis corroborated with sensory information enables non-skilled users to learn tasks to the robots, therefore fostering humanoids integration in real-world work sites. [Whether validated by the simulation and real trial.]

This work mainly paves the for two potential prolongations.

- *Increase the efficiency and the automation.* This would include compute 6d-pose estimation of objects based on visual input, or refine our DMP implementation to make it more efficient, add goal-switching and obstacle avoidance as stated in [28].
- *Increase the generalizability and the objectivity.* One could think about replacing the arbitrary thresholds in the feature selection by statistical methods, or trying to avoid covariate shift by generating demonstrations through meta-learning [16].

REFERENCES

- [1] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in Cognitive Sciences*, vol. 3, no. 6, pp. 233–242, Jun. 1999. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S1364661399013273>
- [2] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469–483, May 2009. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/S0921889008001772>
- [3] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent Advances in Robot Learning from Demonstration," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 297–330, May 2020. [Online]. Available: <https://www.annualreviews.org/doi/10.1146/annurev-control-100819-063206>
- [4] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *2009 IEEE International Conference on Robotics and Automation*, May 2009, pp. 763–768, ISSN: 1050-4729.

- [5] 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 (Cybernetics)*, vol. 37, no. 2, pp. 286–298, Apr. 2007. [Online]. Available: <http://ieeexplore.ieee.org/document/4126276/>
- [6] C. Sammut, S. Hurst, D. Kedzier, and D. Michie, "Learning to Fly," in *Machine Learning Proceedings 1992*, D. Sleeman and P. Edwards, Eds. San Francisco (CA): Morgan Kaufmann, Jan. 1992, pp. 385–393. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9781558602472500553>
- [7] Z. Wu, W. Lian, C. Wang, M. Li, S. Schaal, and M. Tomizuka, "Prim-LAID: A Framework to Learn and Adapt Primitive-Based Skills from Demonstrations for Insertion Tasks," Dec. 2022, arXiv:2212.00955 [cs]. [Online]. Available: <http://arxiv.org/abs/2212.00955>
- [8] K. Ramirez-Amaro, M. Beetz, and G. Cheng, "Understanding the intention of human activities through semantic perception: observation, understanding and execution on a humanoid robot," *Advanced Robotics*, vol. 29, no. 5, pp. 345–362, Mar. 2015. [Online]. Available: <http://www.tandfonline.com/doi/abs/10.1080/01691864.2014.1003096>
- [9] T. Ando, T. Watari, and R. Kikuuwe, "Master-Slave Bipedal Walking and Semi-Automatic Standing Up of Humanoid Robots," in *2020 IEEE/SICE International Symposium on System Integration (SII)*. Honolulu, HI, USA: IEEE, Jan. 2020, pp. 360–365. [Online]. Available: <https://ieeexplore.ieee.org/document/9025820/>
- [10] B. Fang, X. Wei, F. Sun, H. Huang, Y. Yu, and H. Liu, "Skill learning for human-robot interaction using wearable device," *Tsinghua Science and Technology*, vol. 24, no. 6, pp. 654–662, Dec. 2019. [Online]. Available: <https://ieeexplore.ieee.org/document/8727760/>
- [11] M. Vecerik, O. Sushkov, D. Barker, T. Rothorl, T. Hester, and J. Scholz, "A Practical Approach to Insertion with Variable Socket Position Using Deep Reinforcement Learning," in *2019 International Conference on Robotics and Automation (ICRA)*. Montreal, QC, Canada: IEEE, May 2019, pp. 754–760. [Online]. Available: <https://ieeexplore.ieee.org/document/8794074/>
- [12] F. Stulp, E. A. Theodorou, and S. Schaal, "Reinforcement Learning With Sequences of Motion Primitives for Robust Manipulation," *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1360–1370, Dec. 2012.
- [13] J. Rouot and J.-B. Lasserre, "On inverse optimal control via polynomial optimization," in *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*. Melbourne, Australia: IEEE, Dec. 2017, pp. 721–726. [Online]. Available: <http://ieeexplore.ieee.org/document/8263745/>
- [14] Y. Zhao, P. Liu, W. Zhao, and X. Tang, "Variational Diversity Maximization for Hierarchical Skill Discovery," *Neural Processing Letters*, vol. 55, no. 1, pp. 839–855, Feb. 2023. [Online]. Available: <https://link.springer.com/10.1007/s11063-022-10912-8>
- [15] B. D. Argall, B. Browning, and M. Veloso, "Learning robot motion control with demonstration and advice-operators," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 399–404.
- [16] T. Yu, C. Finn, A. Xie, S. Dasari, T. Zhang, P. Abbeel, and S. Levine, "One-Shot Imitation from Observing Humans via Domain-Adaptive Meta-Learning," Feb. 2018, arXiv:1802.01557 [cs]. [Online]. Available: <http://arxiv.org/abs/1802.01557>
- [17] M. A. Rana, M. Mukadam, S. R. Ahmadzadeh, S. Chernova, and B. Boots, "Towards Robust Skill Generalization: Unifying Learning from Demonstration and Motion Planning," in *Proceedings of the 1st Annual Conference on Robot Learning*. PMLR, Oct. 2017, pp. 109–118. [Online]. Available: <https://proceedings.mlr.press/v78/rana17a.html>
- [18] T. Zhang, Z. McCarthy, O. Jow, D. Lee, X. Chen, K. Goldberg, and P. Abbeel, "Deep Imitation Learning for Complex Manipulation Tasks from Virtual Reality Teleoperation," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. Brisbane, QLD: IEEE, May 2018, pp. 5628–5635. [Online]. Available: <https://ieeexplore.ieee.org/document/8461249/>
- [19] C. Chuck, K. Black, A. Arjun, Y. Zhu, and S. Niekum, "Granger-causal hierarchical skill discovery," 2023.
- [20] A. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, vol. 2. Washington, DC, USA: IEEE, 2002, pp. 1398–1403. [Online]. Available: <http://ieeexplore.ieee.org/document/1014739/>
- [21] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical Movement Primitives: Learning Attractor Models for Motor Behaviors," *Neural Computation*, vol. 25, no. 2, pp. 328–373, Feb. 2013. [Online]. Available: <https://direct.mit.edu/neco/article/25/2/328-373/7850>
- [22] F. Meier, E. Theodorou, F. Stulp, and S. Schaal, "Movement segmentation using a primitive library," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2011, pp. 3407–3412, iSSN: 2153-0866.
- [23] R. Caccavale, M. Saveriano, A. Finzi, and D. Lee, "Kinesthetic teaching and attentional supervision of structured tasks in human-robot interaction," *Autonomous Robots*, vol. 43, no. 6, pp. 1291–1307, Aug. 2019. [Online]. Available: <http://link.springer.com/10.1007/s10514-018-9706-9>
- [24] A. Saran, E. S. Short, A. Thomaz, and S. Niekum, "Enhancing Robot Learning with Human Social Cues," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Mar. 2019, pp. 745–747, iSSN: 2167-2148.
- [25] S. Niekum, S. Osentoski, G. Konidaris, S. Chitta, B. Marthi, and A. G. Barto, "Learning grounded finite-state representations from unstructured demonstrations," *The International Journal of Robotics Research*, vol. 34, no. 2, pp. 131–157, Feb. 2015. [Online]. Available: <http://journals.sagepub.com/doi/10.1177/0278364914554471>
- [26] S. Calinon, F. D'halluin, E. L. Sauser, D. G. Caldwell, and A. G. Billard, "Learning and Reproduction of Gestures by Imitation," *IEEE Robotics & Automation Magazine*, vol. 17, no. 2, pp. 44–54, Jun. 2010.
- [27] O. Kroemer, C. Daniel, G. Neumann, H. Van Hoof, and J. Peters, "Towards learning hierarchical skills for multi-phase manipulation tasks," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. Seattle, WA, USA: IEEE, May 2015, pp. 1503–1510. [Online]. Available: <http://ieeexplore.ieee.org/document/7139389/>
- [28] M. Saveriano, F. J. Abu-Dakka, A. Kramberger, and L. Peternel, "Dynamic Movement Primitives in Robotics: A Tutorial Survey," Feb. 2021, arXiv:2102.03861 [cs]. [Online]. Available: <http://arxiv.org/abs/2102.03861>
- [29] P. Pastor, M. Kalakrishnan, L. Righetti, and S. Schaal, "Towards associative skill memories," in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, 2012, pp. 309–315.
- [30] Z. Su, O. Kroemer, G. E. Loeb, G. S. Sukhatme, and S. Schaal, "Learning manipulation graphs from demonstrations using multimodal sensory signals," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2758–2765.
- [31] R. P. Singh, P. Gergondet, and F. Kanehiro, "mc-mujoco: Simulating articulated robots with fsm controllers in mujoco," in *2023 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2023, pp. 1–5.
- [32] J. Benesty, J. Chen, Y. Huang, and I. Cohen, *Pearson Correlation Coefficient*, 04 2009, vol. 2, pp. 1–4.