Enhancing Probabilistic Imitation Learning with Robotic Perception for Self-organising Robot Workstation

Daniel Barros [1,2], Abhishek Padalkar[1], Christoph Willibald[1], João Silvério [1]
[1] German Aerospace Center (DLR), Robotics and Mechatronics Center (RMC),
Münchener Str. 20, 82234 Weßling, Germany.
[2] Technical University of Munich (TUM)

Abstract: The scalability of robotic systems is constrained by traditional programming methods that require specialized expertise. While learning from human demonstrations presents an intuitive programming solution, generalizing learned behaviors to changing surroundings remains a key challenge. In this work, we tackle these issues by integrating Kernelized Movement Primitives (KMP) with computer vision, enabling robots to adapt object-centric tasks learned from demonstrations to varying object configurations. YOLO-based object detection and 3D pose estimation allow the system to dynamically capture variations in object placement and adapt learned trajectories accordingly, leading to precise interaction with objects regardless of their placement. We developed a scalable framework to collect data on the robot and used BlenderProc to automatically generate extensive image datasets for training. We demonstrate this approach on a self-organizing workstation task, where a 7-DOF robot autonomously and effectively cleans up scattered objects.

Keywords: Imitation Learning, Computer Vision, Kernelized Movement Primitives

1 Introduction

Imitation learning (IL) allows robots to learn tasks by observing and replicating human demonstrations [1]. This is particularly useful in areas requiring natural human-robot interaction, such as assistive robotics [2], collaborative manufacturing [3], and service robotics [4]. In this context, a well-designed demonstration interface bridges the gap between human intention and robotic execution, enabling the transfer of nuanced skills with minimal effort [5].

Traditional IL methods like Hidden Markov Models and Gaussian Mixture Models (GMR) generate trajectories from demonstrations but struggle to generalize to unseen situations [1]. Probabilistic Motion Primitives (ProMP) [6] improve adaptability using via points but at the cost of added complexity. Kernelized Movement Primitives (KMP) [7] overcome these limitations by using a kernel-based, non-parametric method that efficiently adapts trajectories to dynamic environments. KMP has been applied to tasks such as soldering, painting, and rehabilitation [7, 8, 9]. It can be combined with computer vision [10, 11] and implemented on Riemannian manifolds to handle orientation data.

In this work, we present Vision-enhanced Kernelized Movement Primitives as an end-to-end imitation learning framework, implemented on the DLR Safe Autonomous Robotic Assistant (SARA) [12]. The system, depicted in Figure 1, uses a scalable, robot-agnostic, and input-device-agnostic data collection pipeline for robot demonstrations and leverages BlenderProc [13] for automatic generation of images to train a YOLOv7 model [14]. This model enables automated recording of object-centric demonstrations as shown in Figure 2 and provides start and end poses for smooth trajectory adaptation during deployment.

8th Conference on Robot Learning (CoRL 2024), Munich, Germany.

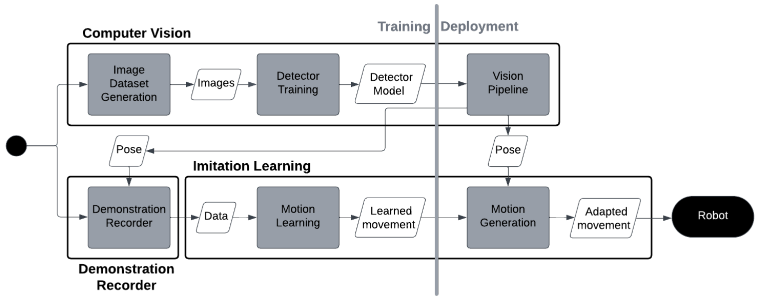


Figure 1: System overview - 3 subsystems: Demonstration Recorder, Imitation Learning and Computer Vision; and 2 phases: training (data collection + model learning) and deployment (model inference).

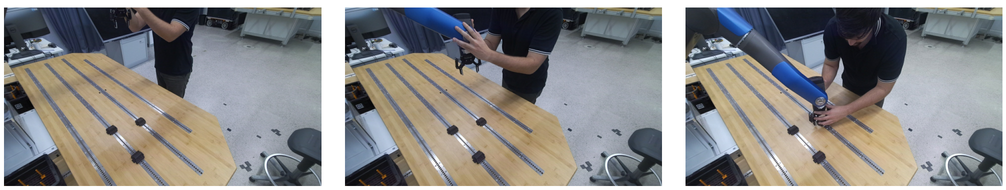
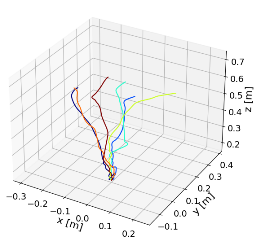


Figure 2: Recording kinesthetic demonstrations of human-like object pick up on DLR-SARA robot

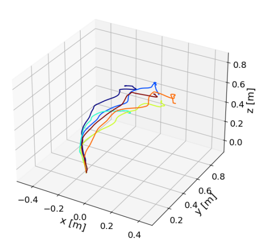
2 Methodology

We developed an end-to-end pipeline that enables robots to learn tasks such as self-organizing its workstation. This pipeline consists of three subsystems as shown in Figure 1: Demonstration Recording, Imitation Learning, and Computer Vision. In this section we will discuss each subsystem in detail.

Demonstration Recorder The purpose of this subsystem is to provide a user-friendly demonstration experience. It is composed of several abstract components: the *Data Logger* records time series data of the robot's poses; the *Robot Event Handler* manages robot control actions such as switching to position control or gravity compensation; the *User Input Handler* processes inputs from a user interface. The *Information Display* logs and displays system actions. Finally, the *Demo Recorder* coordinates the overall process, handling user inputs and sending events to the relevant components. This structure can be used with any robotic platform to record any type of data. In our case, the recorded pose data is then transformed from the robot's tool center point to the object's coordinate frame, as well as aligned over time using Dynamic Time Warping [15]. The result is a structured, temporally aligned object-centric pose dataset that serves as input for imitation learning models.

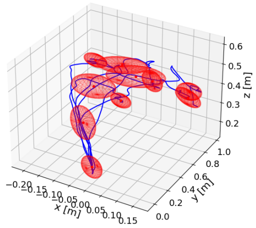


(a) Pick grid clamp phase

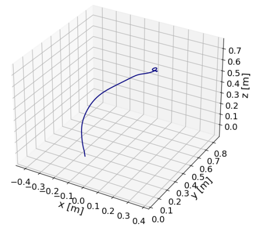


(b) Place in bin phase

Figure 3: 3D position plots of 6 kinesthetic demonstrations of self-organising workstation task in the object frame, described in 3



(a) Learned GMM Model (means and covariances in red) from object-centric demonstrations (blue)



(b) Object-centric trajectory with vanilla KMP (not adapted)

Figure 4: Movement for self-organising workstation task is learned by fitting a GMM to the demonstration data, running GMR to get a trajectory, use that to initialize KMP and later run with via point adaptation

Imitation Learning In this subsystem, let $\{\{t_{n,h}, \boldsymbol{\xi}_{n,h}\}_{n=1}^{N}\}_{h=1}^{H}$ denote the set of demonstration data. $t_{n,h} \in \mathbb{R}$ is the time input and $\boldsymbol{\xi}_{n,h} \in (\mathbb{R}^3 \times \mathrm{SO}(3))$ the pose output. H represents the number of demonstrations and N the trajectory length. . To retrieve a reference trajectory, the pose data is first fitted with a Gaussian Mixture Model (GMM), represented as:

$$\begin{bmatrix} t \\ \boldsymbol{\xi} \end{bmatrix} \sim \sum_{c=1}^{C} \pi_c \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c). \tag{1}$$

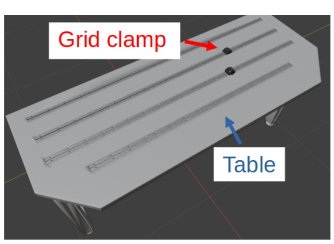
where π_c, $\boldsymbol{\mu}_c$, and $\boldsymbol{\Sigma}_c$ are the prior probability, mean, and covariance of the c-th component. After fitting the GMM, Gaussian Mixture Regression (GMR) is used to condition the trajectory on the time variable, yielding the reference trajectory $\{\hat{\boldsymbol{\xi}}_n\}_{n=1}^{N}$.

The reference trajectory is then refined using KMP [7], which computes a prediction of the pose for each time step based on a Gaussian kernel $k(t_i, t_j) = \exp(-\ell(t_i - t_j)^2)$:

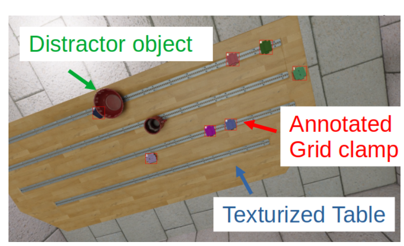
$$\mathbb{E}(\boldsymbol{\xi}(t^*)) = \boldsymbol{k}^*(\boldsymbol{K} + \lambda \boldsymbol{\Sigma}^{-1})\boldsymbol{\mu}, \tag{2}$$

see [7] for the derivation and definitions of $\boldsymbol{K}$ and $\boldsymbol{k}^*$. Via points can be introduced to adapt the trajectory in dynamic environments, with the kernel's length scale ℓ governing the smoothness of this adaptation.

Computer Vision This subsystem is responsible for detecting and estimating the pose of objects. The BlenderProc framework was employed to create 1,600 scenes, each with varying grid clamp placements, resulting in a dataset of 40,000 images with varying camera angles. These images were annotated with bounding boxes, and distractor objects were included to improve robustness (Figure 5a. The YOLO model was trained on this dataset to detect grid clamps, outputting bounding boxes and confidence scores. During deployment, the bounding boxes are combined with depth data to estimate the pose of each grid clamp in the robot's base frame.



(a) Plain STL model in Blender



(b) Example of an annotated scene

Figure 5: For training YOLO, an extensive image dataset was generated in an automated way with BlenderProc

3 Experimental Setup

The task of the 7-DOF SARA robot is to learn to autonomously remove grid clamps from its workstation and place them in a bin. Grid clamps are small black objects that are placed in varying number and pose on or off the metallic rails of the table. An Azure Kinect camera is placed at a 30-degree angle on the left side. Human demonstrations of the necessary movements are recorded and subsequently used with imitation learning and computer vision.

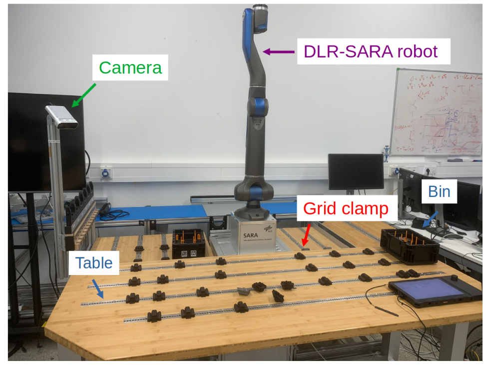
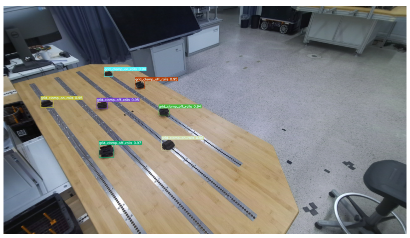
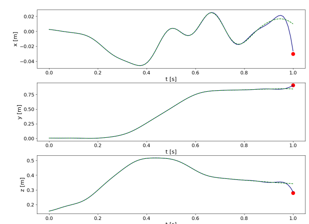


Figure 6: DLR-SARA self-organising robot workstation setup with camera

4 Results and Discussion



(a) YOLO grid clamp detection and on/off-rail classification



(b) KMP (blue line) adapts GMR trajectory (dotted green line) to new bin location (red dot) in object frame

Figure 7: The vision system accurately detects and classifies grid clamps on the table. The imitation learning model handles smooth trajectory adaptation.

The system is able to generalize the smooth pick-and-place to randomly distributed grid clamp poses on the workstation with 100% success rate on 60 placements from 6 demonstrations. Without KMP adaptation, the robot encounters singularities at the workspace edges. The precise choice of length scale ℓ proved crucial, with smaller ℓ causing abrupt changes and larger ℓ affecting accuracy at via points [7]. There was some bias towards right-side bin placements from the demonstrations.

The YOLOv7 vision model reliably detected the grid clamps, maintaining high confidence (0.91-0.97) across various distances and orientations. Pose estimation errors were acceptable, with an x-error of 7mm and y-error of 3mm, allowing for consistent grasping. However, detection failed for grid clamps placed on objects with different textures, highlighting the need for more diverse training data.

Future work to expand the system could explore uncertainty-influenced impedance control [16], sensor fusion with the Azure Kinect and TCP camera for better object pose estimation, and reinforcement learning for optimizing force interactions [17, 18]. Further improvements include expanding detection to more or even unkown objects, employing constrained imitation learning [19], and incorporating more modalities like LLMs/VLMs for an even more intuitive user interface.

5 Conclusion

This paper presents an integrated system combining Kernelized Movement Primitives with computer vision, enabling a robot to learn and adapt manipulation tasks from human demonstrations. Object-centric data collection led to a rich image dataset and a concise pose dataset, facilitating learning from few demonstrations. This holistic approach, demonstrated on a self-organizing workstation task, highlights the potential of blending KMP and computer vision to advance robotic learning.

Acknowledgments

This work was partially funded by the DLR project “Factory of the Future Extended” and the European Union’s Horizon Research and Innovation Programme under Grant 101136067 (INVERSE).

References

[1] A. Billard, S. Calinon, R. Dillmann, and S. Schaal. *Survey: Robot Programming by Demonstration*, page 1371–1394. Springrer, 2008. ISBN 978-3-540-23957-4. doi:10.1007/978-3-540-30301-5_60. URL https://infoscience.epfl.ch/handle/20.500.14299/14639.

[2] C. Lauretti, F. Cordella, E. Guglielmelli, and L. Zollo. Learning by demonstration for planning activities of daily living in rehabilitation and assistive robotics. *IEEE Robotics and Automation Letters*, 2(3):1375–1382, 2017.

[3] L. Rozo, S. Calinon, D. G. Caldwell, P. Jimenez, and C. Torras. Learning physical collaborative robot behaviors from human demonstrations. *IEEE Transactions on Robotics*, 32(3):513–527, 2016.

[4] H. Urbanek, A. Albu-Schaffer, and P. van der Smagt. Learning from demonstration: repetitive movements for autonomous service robotics. In *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, volume 4, pages 3495–3500. IEEE, 2004.

[5] 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*, 3 (1):297–330, 2020.

[6] A. Paraschos, C. Daniel, J. R. Peters, and G. Neumann. Probabilistic movement primitives. *Advances in neural information processing systems*, 26, 2013.

[7] Y. Huang, L. Rozo, J. a. Silvério, and D. G. Caldwell. Kernelized movement primitives. *The International Journal of Robotics Research*, 38(7):833–852, 2019.

[8] Y. Huang, F. J. Abu-Dakka, J. a. Silvério, and D. G. Caldwell. Toward orientation learning and adaptation in cartesian space. *IEEE Transactions on Robotics*, 37(1):82–98, 2020.

[9] Z. Liu, Q. Ai, H. Liu, W. Meng, and Q. Liu. Human-like trajectory planning based on postural synergistic kernelized movement primitives for robot-assisted rehabilitation. *IEEE Transactions on Human-Machine Systems*, 2024.

[10] N. Manakitsa, G. S. Maraslidis, L. Moysis, and G. F. Fragulis. A review of machine learning and deep learning for object detection, semantic segmentation, and human action recognition in machine and robotic vision. *Technologies*, 12(2):15, 2024.

[11] C. Che, H. Zheng, Z. Huang, W. Jiang, and B. Liu. Intelligent robotic control system based on computer vision technology. *arXiv preprint arXiv:2404.01116*, 2024.

[12] M. Iskandar, C. Ott, O. Eiberger, M. Keppler, A. Albu-Schäffer, and A. Dietrich. Joint-level control of the dlr lightweight robot sara. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8903–8910. IEEE, 2020.

[13] M. Denninger, M. Sundermeyer, D. Winkelbauer, Y. Zidan, D. Olefir, M. Elbadrawy, A. Lodhi, and H. Katam. Blenderproc. *arXiv preprint arXiv:1911.01911*, 2019.

[14] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475, 2023.

[15] M. Müller. Dynamic time warping. *Information retrieval for music and motion*, pages 69–84, 2007.

[16] J. a. Silverio, Y. Huang, F. J. Abu-Dakka, L. Rozo, and D. G. Caldwell. Uncertainty-aware imitation learning using kernelized movement primitives. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, Nov. 2019. doi:10.1109/iros40897.2019.8967996. URL http://dx.doi.org/10.1109/IROS40897.2019.8967996.

[17] J. Kober, J. A. Bagnell, and J. Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

[18] A. Padalkar, G. Quere, A. Raffin, et al. A guided reinforcement learning approach using shared control templates for learning manipulation skills in the real world. PREPRINT (Version 1) available at Research Square, August 2023. URL https://doi.org/10.21203/rs.3.rs-3289569/v1.

[19] Y. Huang and D. G. Caldwell. A linearly constrained nonparametric framework for imitation learning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4400–4406. IEEE, 2020.