

# Modifying RL Policies with Imagined Actions: How Predictable Policies can Enable Users to Perform Novel Tasks

Isaac Sheidlower<sup>1</sup>, Reuben Aronson<sup>1</sup>, Elaine Short<sup>1</sup>

Tufts University, Department of Computer Science  
{isaac.sheidlower, reuben.aronson, elaine.short}@tufts.edu<sup>1</sup>

## Abstract

It is crucial that users are empowered to use the functionalities of a robot to creatively solve problems on the fly. A user who has access to a Reinforcement Learning (RL) based robot may want to use the robot’s autonomy and their knowledge of its behavior to complete new tasks. One way is for the user to take control of some of the robot’s action space through teleoperation while the RL policy simultaneously controls the rest. However, an out-of-the-box RL policy may not readily facilitate this. For example, a user’s control may bring the robot into a failure state from the policy’s perspective, causing it to act in a way the user is not familiar with, hindering the success of the user’s desired task. In this work, we formalize this problem and present Imaginary Out-of-Distribution Actions, IODA, an initial algorithm for addressing that problem and empowering user’s to leverage their expectation of a robot’s behavior to accomplish new tasks.

## Introduction

As robots are increasingly in the hands of users, it is necessary to ensure that these users are empowered to use the robot and its functionality to accomplish the tasks they want. Ideally, one can deploy a robot with an exhaustive library of Reinforcement Learning (RL) policies that can autonomously perform any task the user wishes. However, this approach is impractical as users may wish to seamlessly use the robot for a task or in an environment that was not foreseen by the designers of the robot. In reality, the user will be provided with a limited set of policies and a means of controlling the robot, such as teleoperation. Given this, there is a need for methods that enable users to take advantage of these two types of control to perform new and emergent tasks.

Consider the following illustrative example. Sally frequently uses her assistive robot arm to perform tasks around the house. Sally has used the robot arm’s pick-and-place RL policy to move cups full of liquid many times. She knows how the robot will pick up the cup and steadily move it to a specified location. Now Sally wants to use the same functionality to water her flowers. To do this, she starts the autonomous policy and during execution rotates the robot’s wrist to pour the water over her basin and water her flowers as the robot moves along its path. Although in this example

the interaction led to the desired result, an RL policy “as-is” may not facilitate this. For instance, if the RL policy was trained on a reward function such that if it spills liquid, it should stop moving along its path until the cup is upright. In this case, Sally may accidentally pour all her water onto one flower, since the robot did not continue along its path as she had expected. Similar problems exist in the autonomous driving domain, such as driving intentionally on the sidewalk or hitting an obstacle.

These examples demonstrate potential problems of naively using an RL policy while the user controls part of the robot’s behavior. Since the user may have only seen successful examples of task execution, they may not know that their control signal can cause failure with respect to the robot’s reward function, causing it to behave unexpectedly. Furthermore, that unexpected behavior may impede the user’s ability to perform novel tasks or result in task failure from the user’s perspective. This problem setting also allows the user to bring the policy “out-of-distribution,” which may result in undesirable behavior. There is a need to both better understand this type of interaction from a user’s point of view and develop algorithms that act on RL policies to better facilitate the user completing their task. Our key insight is that whilst the user has partial control, the robot should act in a way similar to the behavior the user has seen before and is familiar with, allowing the user to leverage their prior knowledge about the robot’s behavior to adapt it to new tasks.

In this paper, we formalize the aforementioned problem setting and present Imaginary Out-of-Distribution Actions (IODA), an algorithm that modifies the state passed into an RL policy by projecting its current state to a previous state the user has seen: despite the robot’s current real state, it chooses actions based on an *imagined* state. IODA only modifies the state when the user brings the state out-of-distribution (OOD) with respect to states the user is familiar with/has seen before. IODA uses an OOD detector to determine when to modify the state, and then a projection function to determine the nearest state the user is familiar with to project on to. We study IODA in simulation and then discuss our future work studying how IODA can facilitate real users to accomplish new tasks.

## Related Work

Reinforcement learning (RL) is a common and effective way for robots to learn new tasks (Kaelbling, Littman, and Moore 1996; Kober, Bagnell, and Peters; Ibarz et al. 2021). Much of the success of RL for robotics has come from Deep RL (Arulkumaran et al. 2017; Nguyen and La 2019) and human-in-the-loop learning (Christiano et al. 2017; Arzate Cruz and Igarashi 2020; Schrum et al. 2022). However, these methods either assume that the task the user wishes to accomplish is known in advance or that a user is willing to spend time teaching a robot. These assumptions may limit the scope or applicability of any RL-based robot to one or a few tasks. While methods such as multitask reinforcement learning (Xie and Finn 2022; Shridhar, Manuelli, and Fox 2023; Walke et al. 2023), large-language model skill grounding (Ichter et al. 2023; Driess et al. 2023; Liang et al. 2023), and behavioral-diversity learning (Eysenbach et al. 2018; Osa, Tangkaratt, and Sugiyama 2022) have all partially addressed this problem by making robot’s task repertoire larger and more diverse, they do not explicitly empower users with a method of controlling or collaborating with the robot other than through instruction.

In contrast to RL, teleoperation gives users complete control over a robot’s behavior (Lichiardopol 2007). However, the complete teleoperation of a complex robotic system can be challenging for users as there can be many degrees of freedom to control at once with limited feedback from the robot (Darvish et al. 2023). A common approach to alleviate some of the burdens of teleoperation while maintaining a user’s control-based empowerment is Shared Control (SC) (Li et al. 2015; Matsumoto and Riek 2022; Bustamante et al. 2021; Dragan and Srinivasa 2013b). SC often manifests itself as a blending of the user’s and robot’s control. We examine a case where the user has complete/teleoperation control over some parts of the robot while the others are fully autonomous. This is not unlike Shared Autonomy (SA) (Javdani, Srinivasa, and Bagnell 2015; Gopinath, Jain, and Argall 2017; Selvaggio et al. 2021; Fontaine and Nikolaidis 2021), where a user’s control signal is interpreted as an indication of their desired goal. In SA, a typical assumption is that a user’s goal is known in advance and can be represented in clear contrast to other goals in the environment. Although this assumption has been relaxed in some previous work by updating the potential set of goals (Zurek et al. 2021; Fontaine and Nikolaidis 2021; Yoneda et al.), these approaches require a user to first demonstrate the task or a similar task without assistance. Instead, we focus on a user leveraging their creative problem-solving skills to partially control an otherwise fully autonomous robot to complete a novel task. In other words, the robot is not assisting or augmenting the user’s control signal; rather, the user wants to make use of the robot’s behavior as they understand it. Because of this, a robot’s behavior should remain predictable under the partial control of the user.

Legible robot motion can be defined as robot behavior that is easy for a human to predict and understand. A common way to generate legible motion in a goal-based robotic tasks is to model the user as having an internal cost function that is minimized when the robot’s behavior saliently

moves towards a given goal (Dragan and Srinivasa 2013a; Dragan, Lee, and Srinivasa 2013; Faria, Melo, and Paiva 2021). Another approach is to learn from humans through demonstrations or through feedback (Busch et al. 2017; Bied and Chetouani 2020). An important part of each of these works is that legibility and predictable robot behavior are in the context of the robot *completing the task* and are often in real-time as opposed to pre-hoc or post-hoc explanations (Das, Banerjee, and Chernova 2021; Cruz et al. 2021; Sakai and Nagai 2022; Paleja et al. 2021). Predictability is also an important part of our work. We make and leverage the assumption that a robot’s behavior is predictable if the user has seen similar behavior before. This assumption is not entirely dissimilar from robot-centric notions of out-of-distribution (OOD) states and behavior.

OOD detection is an important problem in many robotic and machine learning tasks (Yang et al. 2022; Sun et al. 2022). Detecting when a scenario is OOD with respect to a robot’s training data or what it has previously encountered may have implications about the robot’s environment or performance, such as when an RL agent may behave sporadically or in an unexpected way when in a state it has never been in before (Lan, Zhang, and Hsieh 2022). It has also been used to detect when a robot may require feedback from a person to help complete a novel task (Dass et al. 2023). OOD in RL has recently been studied to detect when an agent is acting in a new MDP (Haider et al. 2023). We apply similar techniques to detect when the robot is in part of the environment that they would not otherwise act in. This can happen when a user is partially controlling a robot to perform a new task.

## Problem Setting

Here we describe a problem setting in which a user is familiar with the autonomous execution of a task and wishes to partially control a robot during that execution to accomplish another task. In short, the user creates a plan to partially control the robot based on how they expect the robot to behave to accomplish a novel task. Thus, it may be critical that the robot behaves in a way that is predictable to the user, no matter where the robot may be in the state space. Given that the robot behaves in a user-predictable way, as opposed to sporadically or in an unfamiliar way, a user can perform various new tasks with relative ease and few surprises.

In this setting, the user has seen the robot complete its task many times. We refer to this as a history of task “roll-outs.” Based on this, we assume the following: when the robot is in a state the user has never seen before, the user expects that the robot will do the same thing as it would do in the “closest” state to its current state. Where “closest” is both problem and user specific, however, the intuition is that the robot will behave in a similar way in similar circumstances and that in novel circumstances, a user will project on to what they have seen before. This assumption temporarily constrains the problem space; however, it is a reasonable assumption in many robotic tasks. Thus, the problem can be defined as, for any given state unseen to the user, the robot should find a state that the user has seen before and act as if it were in that state.

We will now define the original task the robot can complete autonomously and how this task is used to build up a user’s expectation of the robot’s behavior. Let task *orig* be defined as an MDP with states  $S \subset \mathbb{R}^n$ , actions  $A$ , reward function  $r : S \times A \rightarrow \mathbb{R}$ , and transition function  $T : S \times A \rightarrow S$ , and discount factor  $\gamma$ . And there is a robot that has learned an optimal policy for the task denoted  $\pi^*$ . Let  $D$  be defined as a history of rollouts under  $\pi_{orig}^*$  that the user has seen. Then, let the user expectation of the robot’s behavior given  $D$  be:

$$W_D : S \rightarrow S$$

Here,  $W$  is a function that maps from the robot’s current state to their anticipation of what the next state will be.

To this setting, we introduce partitioned control, where the user teleoperates one or more parts of the robot’s actions. We separate the action space into two separate sets  $A_U, A_R \subset A$ ;  $A_U$  denotes the actions that the user can take and  $A_R$  denotes the actions that the robot can take. We further assume that the user and robot action spaces are *disjoint*; that is,  $A_U \cap A_R = \{\mathbf{0}\}$ . In other words, the user and the robot control different parts/different axes of the action space. For example, if the robot is acting in Cartesian space, the user may take control over x-axis actions, or take control over the rotation of a specific joint. We denote the user’s expectation of how the robot will act with their partial control signal as:

$$W_{D,U} : S \rightarrow S$$

where  $U$  is the user control. For brevity, hereafter we refer to this only as  $W$ .

In order to make the robot behavior more predictable for the user, we want to adjust the behavior of the robot policy when it is outside the user’s observation set  $D$ . The goal is to identify when the robot is in a novel state  $s$  where there exists a state  $s' \in D$  that leads to more predictable behavior. Formally, identify when  $\exists s' \in D$  s.t.:

$$d(W(s), T(s, u \circ \pi^*(s))) \geq d(W(s), T(s, u \circ \pi^*(s'))) \quad (1)$$

where  $d$  is a task-dependent distance metric between states, and  $u \circ \pi^*(s)$  denotes the disjoint combination of the autonomous action of the robot and the user’s teleoperation. When such a  $s'$  is identified, the robot should act as if it were in  $s'$ . Specifically, we want to select a new proxy state  $s'$  such that the user’s *predicted* state  $W(s)$  is closer to the actual resultant state when simulating the policy in  $s'$ ,  $T(s, u \circ \pi^*(s'))$  than to the resultant state of running the policy directly  $T(s, u \circ \pi^*(s))$ . Lastly, in this setting, the true  $W$  and the nature of the new task that the user wishes to accomplish are unknown. However, formalizing  $W$  as such can be useful for modeling and/or simulating, creating a learning objective, or creating metrics to measure the success of algorithms applied to this problem.

## Imaginary Out-of-Distribution Actions (IODA)

In this section, we present Imaginary Out-of-Distribution Actions (IODA), to facilitate a user to accomplish new tasks given a policy and a means of teleoperating an axis of robot

behavior (the problem setting described in the previous section). Our key insight is that when the robot is acting in a region that greatly differs from what the user has seen before, the policy should act with imagined states that are as similar to the real state as possible while being “in-distribution” of what the user is familiar with and anticipates. For the rest of this section, unless otherwise specified, we refer to “in/out of distribution” states with respect to  $D$ .

---

### Algorithm 1: Imaginary Out-of-Distribution Actions (IODA)

---

**Initialize:** Rollout history  $D$   
**Initialize:** OOD-state detector  
**Initialize:** State  $s$

- 1: **while** not done **do**
- 2:   **if**  $s$  is OOD **then**
- 3:      $s' \rightarrow \arg \min_{s' \in D} d(s, s')$
- 4:      $a \rightarrow \pi^*(s')$
- 5:   **else**
- 6:      $a \rightarrow \pi^*(s)$
- 7:   **end if**
- 8:    $u \rightarrow$  user’s control signal
- 9:    $s_{t+1} \rightarrow T(s, u \circ a)$
- 10: **end while**

---

The complete IODA algorithm is presented in Algorithm 1. Here, we require that an OOD detector be trained on  $D$ . This is then used to detect “novel” states. While this technique is not new from a robot-centered perspective, it is also a human-modeling choice that draws an analogy between when an OOD detector outputs a state that is OOD and when a human may be projecting to a state they have seen in the past. It is also being used to determine when to search for a state that the robot policy should “imagine” it is in.

## Simulation Example

In this section we demonstrate in a 2d continuous navigation task both the problem setting and the application of the IODA algorithm. In the original task (see Figure 1), the robot learned to navigate to any specified goal point from within its workspace (highlighted in gray in Figure 1). The reward function for the task is defined as  $-|p_{agent} - p_{goal}|$ , or the negative distance of the agent’s position to the goal. The user then wishes to leverage the optimal behavior they have observed to control the x-axis actions to first guide the robot to an intermediate goal outside of the workspace (shown in Figure 1 as blue or green dots) and then to the primary goal: a novel task not represented by the robot’s policy.

We train two RL agents to optimally solve two slightly different versions of the original task using the off-policy RL algorithm SAC (Haarnoja et al. 2019). We use SAC as it has been shown to be relatively robust when out-of-distribution (Lan, Zhang, and Hsieh 2022). In one version of the task, Figure 1, b, the RL algorithm was trained constrained to the gray workspace, simply being penalized for leaving it. In the other version, Figure 1, c, the agent is penalized if it is out of the workspace, and is further penalized by moving in the y-axis whilst it is out of the workspace. This encourages the agent to return to its workspace as quickly as possible before



## References

- Arulkumaran, K.; Deisenroth, M. P.; Brundage, M.; and Bharath, A. A. 2017. A Brief Survey of Deep Reinforcement Learning. *IEEE Signal Processing Magazine*, 34(6): 26–38. ArXiv:1708.05866 [cs, stat].
- Arzate Cruz, C.; and Igarashi, T. 2020. A Survey on Interactive Reinforcement Learning: Design Principles and Open Challenges. In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*, 1195–1209. Eindhoven Netherlands: ACM. ISBN 978-1-4503-6974-9.
- Bied, M.; and Chetouani, M. 2020. Integrating an Observer in Interactive Reinforcement Learning to Learn Legible Trajectories. In *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 760–767. ISSN: 1944-9437.
- Busch, B.; Grizou, J.; Lopes, M.; and Stulp, F. 2017. Learning Legible Motion from Human–Robot Interactions. *International Journal of Social Robotics*, 9(5): 765–779.
- Bustamante, S.; Quere, G.; Hagmann, K.; Wu, X.; Schmaus, P.; Vogel, J.; Stulp, F.; and Leidner, D. 2021. Toward Seamless Transitions Between Shared Control and Supervised Autonomy in Robotic Assistance. *IEEE Robotics and Automation Letters*, 6(2): 3833–3840. Conference Name: IEEE Robotics and Automation Letters.
- Christiano, P.; Leike, J.; Brown, T. B.; Martic, M.; Legg, S.; and Amodei, D. 2017. Deep reinforcement learning from human preferences. Number: arXiv:1706.03741 arXiv:1706.03741 [cs, stat].
- Cruz, F.; Dazeley, R.; Vamplew, P.; and Moreira, I. 2021. Explainable robotic systems: understanding goal-driven actions in a reinforcement learning scenario. *Neural Computing and Applications*.
- Darvish, K.; Penco, L.; Ramos, J.; Cisneros, R.; Pratt, J.; Yoshida, E.; Ivaldi, S.; and Pucci, D. 2023. Teleoperation of Humanoid Robots: A Survey. *IEEE Transactions on Robotics*, 39(3): 1706–1727. Conference Name: IEEE Transactions on Robotics.
- Das, D.; Banerjee, S.; and Chernova, S. 2021. Explainable AI for Robot Failures: Generating Explanations that Improve User Assistance in Fault Recovery. In *Proceedings of the 2021 ACM/IEEE International Conference on Human-Robot Interaction, HRI '21*, 351–360. New York, NY, USA: Association for Computing Machinery. ISBN 978-1-4503-8289-2.
- Dass, S.; Pertsch, K.; Zhang, H.; Lee, Y.; Lim, J.; and Nikolaidis, S. 2023. PATO: Policy Assisted TeleOperation for Scalable Robot Data Collection. In *Robotics: Science and Systems XIX*. Robotics: Science and Systems Foundation. ISBN 978-0-9923747-9-2.
- Dragan, A.; and Srinivasa, S. 2013a. Generating Legible Motion. In *Robotics: Science and Systems IX*. Robotics: Science and Systems Foundation. ISBN 978-981-07-3937-9.
- Dragan, A. D.; Lee, K. C.; and Srinivasa, S. S. 2013. Legibility and predictability of robot motion. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 301–308. Tokyo, Japan: IEEE. ISBN 978-1-4673-3101-2 978-1-4673-3099-2 978-1-4673-3100-5.
- Dragan, A. D.; and Srinivasa, S. S. 2013b. A policy-blending formalism for shared control. *The International Journal of Robotics Research*, 32(7): 790–805. Publisher: SAGE Publications Ltd STM.
- Driess, D.; Xia, F.; Sajjadi, M. S. M.; Lynch, C.; Chowdhery, A.; Ichter, B.; Wahid, A.; Tompson, J.; Vuong, Q.; Yu, T.; Huang, W.; Chebotar, Y.; Sermanet, P.; Duckworth, D.; Levine, S.; Vanhoucke, V.; Hausman, K.; Toussaint, M.; Greff, K.; Zeng, A.; Mordatch, I.; and Florence, P. 2023. PaLM-E: An Embodied Multimodal Language Model. ArXiv:2303.03378 [cs].
- Eysenbach, B.; Gupta, A.; Ibarz, J.; and Levine, S. 2018. Diversity is All You Need: Learning Skills without a Reward Function. Number: arXiv:1802.06070 arXiv:1802.06070 [cs].
- Faria, M.; Melo, F. S.; and Paiva, A. 2021. Understanding Robots: Making Robots More Legible in Multi-Party Interactions. In *2021 30th IEEE International Conference on Robot & Human Interactive Communication (RO-MAN)*, 1031–1036. ISSN: 1944-9437.
- Fontaine, M. C.; and Nikolaidis, S. 2021. Differentiable Quality Diversity. Number: arXiv:2106.03894 arXiv:2106.03894 [cs].
- Gopinath, D.; Jain, S.; and Argall, B. D. 2017. Human-in-the-Loop Optimization of Shared Autonomy in Assistive Robotics. *IEEE Robotics and Automation Letters*, 2(1): 247–254. Conference Name: IEEE Robotics and Automation Letters.
- Haarnoja, T.; Zhou, A.; Hartikainen, K.; Tucker, G.; Ha, S.; Tan, J.; Kumar, V.; Zhu, H.; Gupta, A.; Abbeel, P.; and Levine, S. 2019. Soft Actor-Critic Algorithms and Applications. *arXiv:1812.05905 [cs, stat]*.
- Haider, T.; Roscher, K.; Schmoeller da Roza, F.; and Günemann, S. 2023. Out-of-Distribution Detection for Reinforcement Learning Agents with Probabilistic Dynamics Models. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems, AAMAS '23*, 851–859. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. ISBN 978-1-4503-9432-1.
- Ibarz, J.; Tan, J.; Finn, C.; Kalakrishnan, M.; Pastor, P.; and Levine, S. 2021. How to Train Your Robot with Deep Reinforcement Learning; Lessons We’ve Learned. *The International Journal of Robotics Research*, 40(4-5): 698–721. ArXiv:2102.02915 [cs].
- Ichter, B.; Brohan, A.; Chebotar, Y.; Finn, C.; Hausman, K.; Herzog, A.; Ho, D.; Ibarz, J.; Irpan, A.; Jang, E.; Julian, R.; Kalashnikov, D.; Levine, S.; Lu, Y.; Parada, C.; Rao, K.; Sermanet, P.; Toshev, A. T.; Vanhoucke, V.; Xia, F.; Xiao, T.; Xu, P.; Yan, M.; Brown, N.; Ahn, M.; Cortes, O.; Sievers, N.; Tan, C.; Xu, S.; Reyes, D.; Rettinghouse, J.; Quiambao, J.; Pastor, P.; Luu, L.; Lee, K.-H.; Kuang, Y.; Jesmonth, S.; Joshi, N. J.; Jeffrey, K.; Ruano, R. J.; Hsu, J.; Gopalakrishnan, K.; David, B.; Zeng, A.; and Fu, C. K. 2023. Do As I Can, Not As I Say: Grounding Language in Robotic Affordances. In *Proceedings of The 6th Conference on Robot Learning*, 287–318. PMLR. ISSN: 2640-3498.

- Javdani, S.; Srinivasa, S.; and Bagnell, A. 2015. Shared Autonomy via Hindsight Optimization. In *Robotics: Science and Systems XI*. Robotics: Science and Systems Foundation. ISBN 978-0-9923747-1-6.
- Kaelbling, L. P.; Littman, M. L.; and Moore, A. W. 1996. Reinforcement Learning: A Survey. ArXiv:cs/9605103.
- Kober, J.; Bagnell, J. A.; and Peters, J. 2018. Reinforcement Learning in Robotics: A Survey.
- Lan, L.-C.; Zhang, H.; and Hsieh, C.-J. 2022. Can Agents Run Relay Race with Strangers? Generalization of RL to Out-of-Distribution Trajectories.
- Li, Y.; Tee, K. P.; Chan, W. L.; Yan, R.; Chua, Y.; and Limbu, D. K. 2015. Continuous Role Adaptation for Human-Robot Shared Control. *IEEE Transactions on Robotics*, 31(3): 672–681. Conference Name: IEEE Transactions on Robotics.
- Liang, J.; Huang, W.; Xia, F.; Xu, P.; Hausman, K.; Ichter, B.; Florence, P.; and Zeng, A. 2023. Code as Policies: Language Model Programs for Embodied Control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 9493–9500.
- Lichiardopol, S. 2007. Contents Contents 2.
- Matsumoto, S.; and Riek, L. D. 2022. Shared Control in Human Robot Teaming: Toward Context-Aware Communication. ArXiv:2203.10218 [cs].
- Nguyen, H.; and La, H. 2019. Review of Deep Reinforcement Learning for Robot Manipulation. In *2019 Third IEEE International Conference on Robotic Computing (IRC)*, 590–595. Naples, Italy: IEEE. ISBN 978-1-5386-9245-5.
- Osa, T.; Tangkaratt, V.; and Sugiyama, M. 2022. Discovering diverse solutions in deep reinforcement learning by maximizing state-action-based mutual information. *Neural Networks*, 152: 90–104.
- Paleja, R.; Ghuy, M.; Ranawaka Arachchige, N.; Jensen, R.; and Gombolay, M. 2021. The Utility of Explainable AI in Ad Hoc Human-Machine Teaming. In *Advances in Neural Information Processing Systems*, volume 34, 610–623. Curran Associates, Inc.
- Ruff, L.; Vandermeulen, R.; Goernitz, N.; Deecke, L.; Sidiqi, S. A.; Binder, A.; Müller, E.; and Kloft, M. 2018. Deep One-Class Classification. In *Proceedings of the 35th International Conference on Machine Learning*, 4393–4402. PMLR. ISSN: 2640-3498.
- Sakai, T.; and Nagai, T. 2022. Explainable autonomous robots: a survey and perspective. *Advanced Robotics*, 36(5-6): 219–238. Publisher: Taylor & Francis. eprint: <https://doi.org/10.1080/01691864.2022.2029720>.
- Schrum, M. L.; Hedlund-Botti, E.; Moorman, N.; and Gombolay, M. C. 2022. MIND MELD: Personalized Meta-Learning for Robot-Centric Imitation Learning. In *Proceedings of the 2022 ACM/IEEE International Conference on Human-Robot Interaction, HRI '22*, 157–165. Sapporo, Hokkaido, Japan: IEEE Press.
- Selvaggio, M.; Cognetti, M.; Nikolaidis, S.; Ivaldi, S.; and Siciliano, B. 2021. Autonomy in Physical Human-Robot Interaction: A Brief Survey. *IEEE Robotics and Automation Letters*, 6(4): 7989–7996. Conference Name: IEEE Robotics and Automation Letters.
- Shridhar, M.; Manuelli, L.; and Fox, D. 2023. Perceiver-Actor: A Multi-Task Transformer for Robotic Manipulation. In *Proceedings of The 6th Conference on Robot Learning*, 785–799. PMLR. ISSN: 2640-3498.
- Sun, Y.; Ming, Y.; Zhu, X.; and Li, Y. 2022. Out-of-Distribution Detection with Deep Nearest Neighbors. In *Proceedings of the 39th International Conference on Machine Learning*, 20827–20840. PMLR. ISSN: 2640-3498.
- Walke, H. R.; Yang, J. H.; Yu, A.; Kumar, A.; Orbik, J.; Singh, A.; and Levine, S. 2023. Don't Start From Scratch: Leveraging Prior Data to Automate Robotic Reinforcement Learning. In *Proceedings of The 6th Conference on Robot Learning*, 1652–1662. PMLR. ISSN: 2640-3498.
- Xie, A.; and Finn, C. 2022. Lifelong Robotic Reinforcement Learning by Retaining Experiences. In *Proceedings of The 1st Conference on Lifelong Learning Agents*, 838–855. PMLR. ISSN: 2640-3498.
- Yang, J.; Zhou, K.; Li, Y.; and Liu, Z. 2022. Generalized Out-of-Distribution Detection: A Survey. ArXiv:2110.11334 [cs].
- Yoneda, T.; Sun, L.; Yang, G.; Stadie, B.; and Walter, M. R. 2022. To the Noise and Back: Diffusion for Shared Autonomy.
- Zurek, M.; Bobu, A.; Brown, D. S.; and Dragan, A. D. 2021. Situational Confidence Assistance for Lifelong Shared Autonomy. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2783–2789. Xi'an, China: IEEE Press.