Simposio CEA de Robótica, Bioingeniería, Visión por Computador y Automática Marina 2026

# Single-Demonstration Imitation with Residual Reinforcement Learning for Dual-Arm Robotic Bottle Opening

Naranjo-Campos, Francisco J.*, Victores, Juan G., Balaguer, Carlos, Jardón Huete, Alberto.

*RoboticsLab, Systems and Automation Engineering Department, University Carlos III of Madrid, Leganés, 28911, Madrid, Spain*

**Abstract**

Learning manipulation from extremely limited data remains challenging for robotic systems. We present a framework that combines Behavioural Cloning (BC) from a single kinesthetic demonstration with residual Reinforcement Learning (RL) to solve a long-horizon dual-arm bottle unscrewing task. A base policy is first trained via supervised imitation, capturing nominal behaviour. A residual policy is then learned in simulation using PPO to produce bounded corrective actions, improving robustness and generalization to variations in bottle geometry. Results show that while single-demonstration BC performs reliably under nominal conditions, it degrades under distribution shifts. The residual formulation preserves nominal performance and significantly improves robustness. The final controller is deployed in a one-shot sim-to-real transfer, achieving successful execution on different bottle types.

*Keywords:* Imitation learning, behavioural cloning, one-shot demonstration, residual reinforcement learning, sim-to-real transfer, sample efficiency.

**Imitación con una sola demostración con aprendizaje por refuerzo residual para la apertura de botellas mediante un robot bi-manipulador**

**Resumen**

El aprendizaje de manipulaciones a partir de datos extremadamente limitados sigue siendo un reto para los sistemas robóticos. Presentamos un marco que combina la clonación conductual (BC) a partir de una única demostración cinestésica con el aprendizaje por refuerzo residual (RL) para resolver una tarea de desenroscar botellas con dos brazos en un horizonte lejano. En primer lugar, se entrena una política básica mediante imitación supervisada, capturando el comportamiento nominal. A continuación, se aprende una política residual en simulación utilizando PPO para producir acciones correctivas limitadas, mejorando la robustez y la generalización a las variaciones en la geometría de la botella. Los resultados muestran que, si bien la BC de demostración única funciona de manera fiable en condiciones nominales, se degrada bajo cambios de distribución. La formulación residual preserva el rendimiento nominal y mejora significativamente la robustez. El controlador final se implementa en una transferencia única de simulación a realidad, logrando una ejecución exitosa en distintos tipos de botellas.

*Palabras clave:* Aprendizaje por imitación, clonación conductual, demostración única, aprendizaje por refuerzo residual, transferencia de simulación a realidad, uso eficiente de las muestras.

## 1. Introduction

Robotic manipulation remains difficult to implement in unstructured and variable tasks. Unlike industrial environments, where every element is precisely measured and controlled, performing everyday tasks such as opening a bottle becomes significantly more challenging due to friction, compliance, and complex contact geometry. These tasks are also typically long-horizon, requiring sustained interaction over time. Moreover, when dual arm manipulation is considered, the coordination of motion and force regulation further increases the difficulty (Lin et al., 2024).

*Autor para correspondencia: frnaranj@ing.uc3m.es

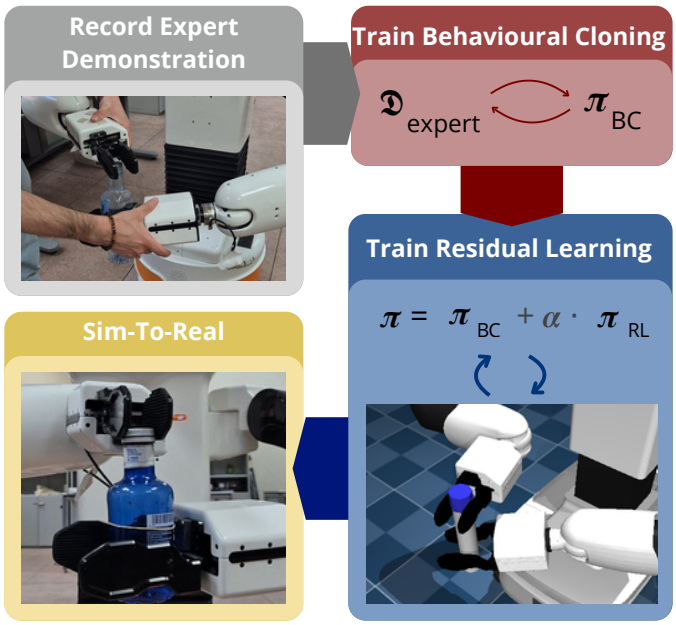


Figure 1: Pipeline of the proposed single-demonstration residual learning approach. Behavioural Cloning provides a nominal policy from one expert trajectory, while PPO learns additive residual corrections in simulation. The final policy is evaluated through one-shot sim-to-real deployment.

In this context, Learning from Demonstration (LfD) serves as a practical solution for acquiring skills from human examples, avoiding hand-engineered planning and enabling rapid skill acquisition (Bain and Sammut, 2001).

However, LfD often suffers from poor performance and error accumulation during execution, especially when encountering unexplored scenarios or when adaptation capacity is limited. Additionally, LfD typically requires collecting many demonstrations to properly optimize the learning algorithms. When trained on a single trajectory or limited datasets, standard approaches often lack convergence and are more likely to fail when facing unseen situations (Ross et al., 2011; Foster et al., 2024).

Reinforcement Learning (RL), on the other hand, has shown excellent results in solving complex tasks, including long-horizon, dual-arm, and whole-body manipulation problems. It also has the capacity to explore a wide range of scenarios, as it generates its own experience through interaction. However, end-to-end RL is often sample-inefficient and, as a result, difficult to deploy safely on physical robots due to unstable exploration and sim-to-real gaps (Brunke et al., 2021).

Therefore, residual RL offers an attractive bridge: it restricts learning to bounded corrections around a safe and previously tested policy or controller. This improves sample efficiency and practical safety while preserving reliable behaviour (Johannink et al., 2019; Shi et al., 2021).

In this work, we address the long-horizon task of unscrewing a bottle cap by combining LfD with residual reinforcement learning on a dual-arm robot. Our objective is not only successful execution, but to achieve an efficient and safe implementation starting from a single expert demonstration.

We first record a trajectory of the robot opening a bottle through direct human guidance. A base policy is then trained using Behavioural Cloning (BC) (Bain and Sammut, 2001). Building on this policy, we train a residual policy in simulation using PPO with a shaped reward (Schulman et al., 2017), enabling adaptation and generalization to variations in bottle size. The resulting controller is finally evaluated in a one-shot sim-to-real transfer. An overview of the proposed framework is shown in Figure 1.

The main contributions of this work are:

- A practical implementation of a BC + residual RL framework for a long-horizon, contact-rich dual-arm manipulation task.
- Demonstration that a single expert trajectory can be leveraged to bootstrap learning and enable generalization through residual reinforcement learning.
- An efficient and safety-oriented training strategy that constrains exploration around a demonstrated behaviour.
- A one-shot sim-to-real transfer of the learned policy to the physical robot.

## 2. Related Work

This work builds on established research in BC and residual RL for contact-rich manipulation. We review the most relevant literature and position our contribution.

### *2.1. Imitation Learning and Single-shot*

BC learns a policy through supervised prediction of expert actions and is attractive due to its simplicity and data efficiency (Bain and Sammut, 2001). However, in sequential decision-making, small prediction errors can lead to covariate shift and compounding errors, especially when only few demonstrations are available (Ross et al., 2011). DAgger mitigates this issue by collecting data under the learner's policy and querying the expert for corrective labels, but this requires interactive expert access, which contradicts strict one-shot constraints (Ross et al., 2011). Recent analyses revisit the horizon dependence of BC and studies when it remain stable, depending on task structure and realizability assumptions (Foster et al., 2024).

Other works attempt to infer rewards from a single demonstration, often relying on final-state matching (Naranjo-Campos et al., 2024a). Our method instead learns bounded residual corrections on top of a single-demo BC policy.

### *2.2. Residual Reinforcement Learning*

Previous work on bottle opening has explored hybrid approaches that combine reinforcement learning with a stable baseline controller (Naranjo-Campos et al., 2024b), assigning RL to specific sub-tasks such as the twisting motion while maintaining predefined control for the remaining phases. In contrast, Residual Policy Learning formalizes the idea of learning an additive correction over an existing controller, improving sample efficiency by restricting optimization to local refinements (Johannink et al., 2019). In robotics, residual RL has demonstrated robust performance in contact-rich scenarios by superimposing learned corrections onto conventional feedback controllers (Shi et al., 2021). Our work instantiates this paradigm by deriving the baseline from a single real dual-arm demonstration and training a PPO-based residual policy in simulation to enhance robustness and generalization.

## 3. Methodology

We formulate the unscrewing task as a finite-horizon Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, $P$ the transition dynamics induced by the simulator, $r$ the reward function, and $\gamma \in (0, 1]$ the discount factor. The objective is to learn a policy $\pi(a|s)$ that maximizes the reward for a long-horizon, contact-rich dual-arm manipulation task.

### 3.1. State and Action Representation

Rather than learning in joint space, we define actions as incremental Cartesian pose commands applied to the right and left end-effector. The action vector is

$$a_t = [\Delta x_t, \Delta y_t, \Delta z_t, \Delta\psi_t, g_t] \in [-1, 1]^5, \tag{1}$$

where $(\Delta x, \Delta y, \Delta z)$ are scaled Cartesian increments applied to the right end-effector, while only planar increments $(\Delta x, \Delta y)$ are applied to the left end-effector (its vertical position remains fixed). $\Delta\psi$ denotes a yaw increment around the vertical axis of the right end-effector, and $g_t \in \{-1, 1\}$ encodes the binary gripper command (open/close).

Pose increments are scaled and mapped to joint velocities through inverse kinematics before being applied to the simulator.

The observation vector is defined as

$$s_t = \left[p_t^r, \psi_t^r, p_t^l, g_t, f_t, \Delta z\right], \tag{2}$$

where $p_t^r$ and $p_t^l$ are the Cartesian positions of the right and left end-effectors, $\psi_t^r$ is the right yaw angle, $g_t$ is the binary gripper state, and $f_t$ is a binary force proxy. $\Delta z$ is the cumulation of yaw roll with the gripper closed, as a proxy of cap screw.

The force proxy is defined as:

$$f_t = \begin{cases} 1 & \text{if gripper closed and } \|p_t^{cap} - p_t^r\| < \epsilon \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

This signal approximates whether a valid contact-consistent grasp is established, reducing reliance on precise force modelling and improving sim-to-real robustness. The threshold $\epsilon$ is empirically defined, with different values used in simulation and real-world execution to account for discrepancies in sensing and contact modelling.

To promote generalization, we introduce variability at episode reset by perturbing the initial bottle-cap position within $\pm 0.20$, m of the recorded demonstration and varying the bottle size within $\pm 0.05$, m. The residual policy is trained across these variations in simulation.

### 3.2. Threaded Cap Model

The cap motion is modelled through an accumulated roll variable $\theta_t$:

$$\theta_t = \theta_{t-1} + \Delta\psi_t. \tag{4}$$

Vertical displacement is derived from a fixed thread pitch $k$:

$$z_t = z_{t-1} + k\Delta\psi_t. \tag{5}$$

The episode terminates when $\theta_t \geq \theta_{target}$.

### 3.3. Behavioural Cloning Initialization

Given a single expert trajectory $\tau_E = \{(s_t^E, a_t^E)\}_{t=0}^T$, collected via kinaesthetic guidance, we first learn a base policy $\pi_{BC}$ by minimizing the supervised objective:

$$\mathcal{L}_{BC} = \sum_{t=0}^{T} \left\|\pi_{BC}(\tilde{s}_t^E) - a_t^E\right\|^2. \tag{6}$$

Instead of directly using the raw state representation $s_t$, we define a normalized observation $\tilde{s}_t$ based on relative quantities. In particular, positional components are expressed with respect to their values at the initial time-step of the demonstration:

$$\tilde{p}_t = p_t - p_0. \tag{7}$$

This transformation reduces the dependence of the behavioural policy on absolute spatial configurations, which is critical given that $\pi_{BC}$ is trained from a single trajectory.

Without this modification, changes in the initial configuration when including the residual policy with change in environment would result in observations outside the training distribution, causing the policy to fail. By encoding the task in relative terms, the learned policy better captures the underlying structure of the manipulation behaviour and improves robustness to variations in the initial state.

Observations are further normalized using dataset statistics before training. The resulting policy reproduces the demonstrated nominal behaviour under these relative representations.

### 3.4. Residual Reinforcement Learning

To improve robustness and enable adaptation to variations in bottle dimensions and initial position, we learn a residual policy $\pi_{RL}$ trained with PPO.

The executed action is defined as:

$$a_t^{final} = \pi_{BC}(s_t) + \alpha\, \pi_{RL}(s_t), \tag{8}$$

where $\alpha$ bounds the magnitude of corrections.

The reward is composed of four terms that encourage progress while maintaining stable contact and smooth control:

$$r_t = r_\theta + r_z + r_a + r_c. \tag{9}$$

The individual components are defined as:

$$r_\theta = w_1 \max(0, \Delta\theta_t), \tag{10}$$

$$r_z = w_2 \max(0, \Delta z_t), \tag{11}$$

$$r_a = -w_3 \|a_t\|^2, \tag{12}$$

$$r_c = -w_4 \|p_t^{cap} - p_t^r\|. \tag{13}$$

The term $r_\theta$ rewards positive rotational progress of the cap, ensuring that only forward unscrewing motion is reinforced. The term $r_z$ rewards vertical displacement induced by the threaded model, encouraging consistent upward motion as rotation accumulates. The regularization term $r_a$ penalizes large control inputs, promoting smooth and stable behaviour. Finally, $r_c$ penalizes excessive distance between the cap and the right end-effector, encouraging contact consistency and reducing slip.

Together, these terms shape the learning signal toward progressive unscrewing while constraining unstable or non-contact behaviours.

### *3.5. Curriculum Learning for Residual Training*

To facilitate stable learning of the residual policy, we adopt a curriculum learning strategy in which the level of environment variation is progressively increased during training.

Initially, the residual policy is trained under conditions matching the expert demonstration, where the behavioural policy provides a reliable baseline. As training progresses, perturbations are increased in a staged manner, gradually expanding the range of initial positional deviations of the right and left arms, as well as variations in the bottle size. This requires the residual policy to compensate for increasingly significant deviations from the demonstrated trajectory.

This curriculum mitigates the difficulty of learning under large distribution shifts from the outset, allowing the residual policy to first learn local corrections before generalizing to more challenging scenarios.

This strategy is particularly important in our setting, as the effectiveness of residual learning depends on the base policy operating within a region where corrective actions remain meaningful.

### *3.6. One-Shot Sim-to-Real Transfer*

After training, the residual policy is deployed on the physical robot without further fine-tuning. The proposed formulation, based on incremental Cartesian control, a simplified force proxy, and bounded residual corrections, enables direct sim-to-real transfer.

## 4. Experiment and Results

We evaluate the proposed approach in both simulation and real-world execution. The goal of the experiments is threefold: (i) to assess learning behaviour during residual PPO training, (ii) to compare success and final performance across methods with and without bottle variation, (iii) to analyse the effect of increasing environment variation on residual policy performance, and (iv) to validate one-shot sim-to-real transfer.

### *4.1. Experimental setup*

Experiments are conducted using a dual-arm TIAGo robot platform equipped with two 7-DoF manipulators and parallel grippers. The robot provides torque-controlled arms and Cartesian end-effector control through inverse kinematics. As previously described,, manipulation is performed using incremental Cartesian commands applied to both end-effectors. A simplified binary force proxy is used instead of precise force sensing to detect grasping conditions, improving robustness and reducing dependence on accurate contact modelling.

The simulation and learning framework is implemented using the MuJoCo physics engine integrated with Gymnasium environments. Reinforcement learning is trained using Stable-Baselines3, specifically the PPO algorithm. The system is implemented in Python, and experiments are conducted using parallel simulation environments to improve sample efficiency.

### *4.2. Training Behaviour in Simulation*

We first analyze the learning dynamics of the residual policy trained with PPO. Figure 2 shows the evolution of the episodic return across training episodes. The curve corresponds to the mean reward, accompanied by the standard deviation and a smoothed trend computed using a moving average with a window size of 50 episodes for visualization purposes. Additionally, vertical markers indicate curriculum learning stages, where the level of perturbation in the initial conditions is progressively increased. Training starts with no deviation from the expert demonstration, and gradually incorporates increasing positional perturbations as well as variations in bottle size. Figure 3 further reports the mean episode length during training, providing additional insight into policy efficiency and task completion behaviour.

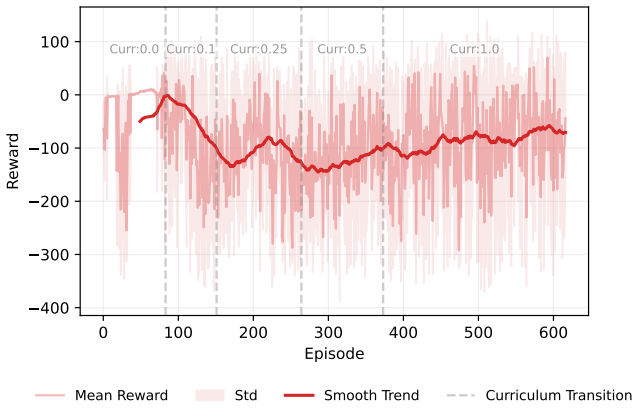


Figure 2: Mean episodic return with standard deviation and smoothing (window = 50) though training over $4 \times 10^6$ steps and 4 environments. Vertical lines indicate curriculum changes.

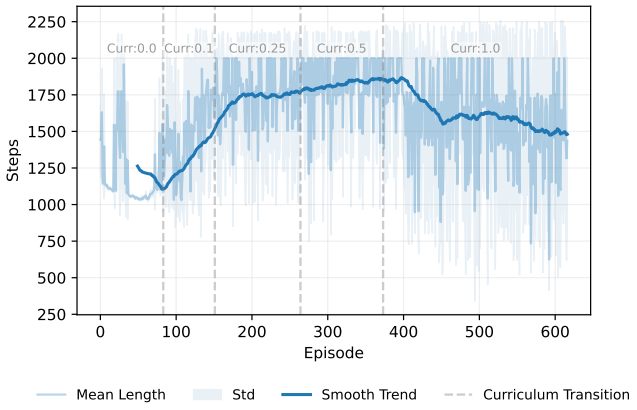


Figure 3: Mean episodic length with standard deviation and smoothing (window = 50) though training over $4 \times 10^6$ steps and 4 environments. Vertical lines indicate curriculum changes.

Regarding the reward evolution, performance is initially higher under zero deviation conditions and improves over training episodes. However, temporary drops are observed at each curriculum transition, after which the policy recovers and continues improving.

A similar trend is observed for episode length: the task is completed in fewer steps at the beginning, while increased perturbations lead to longer episodes. As training progresses, the episode length decreases again, indicating improved efficiency.

### *4.3. Simulation Performance*

We evaluate four simulation configurations: BC under the same conditions as the single demonstration, BC with deviations, BC + residual RL under the same conditions, and BC + residual RL under deviations. Results are summarized in Table 1, and representative execution picture is shown in Figure 4.

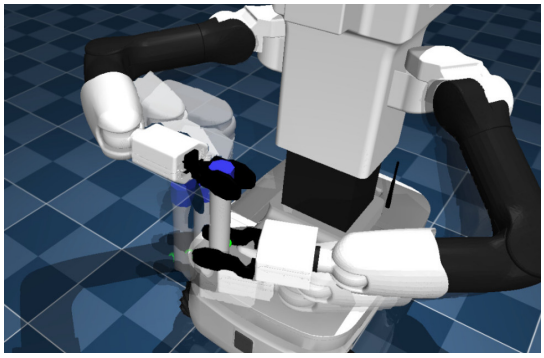

Figure 4: Simulation setup and representative execution of the unscrewing task.

Table 1: Simulation performance comparison across configurations. Mean and standard deviation over 100 executions.

| **Method** | **Reward** | **Episode Length** |
|---|---|---|
| BC | 21.41 ± 0.00 | 880.0 ± 0.0 |
| BC with Dev | −3.41 ± 24.63 | 1719.6 ± 303.6 |
| BC + RL | 21.42 ± 0.00 | 878.0 ± 0.0 |
| BC + RL with Dev | 28.25 ± 19.48 | 857.7 ± 124.3 |

The comparison highlights two main observations. First, BC achieves strong performance under the same conditions as the demonstration but degrades significantly when deviations are introduced, as reflected by the drop in reward and the increase in episode length. Second, BC + residual RL maintains similar performance in the nominal setting while substantially improving performance under deviations, achieving higher rewards and shorter episodes compared to BC alone, indicating improved robustness to variations in initial position and bottle size.

### *4.4. Effect of Environment Variation on Residual Performance*

We analyse the relationship between the magnitude of the initial perturbation and the response of the residual policy.

Figure 5 shows the relationship between the initial deviation and both the residual action magnitude and the obtained reward. As the deviation increases, the residual policy produces larger corrective actions. In contrast, the reward varies smoothly and does not show a strong dependence on the deviation, indicating consistent performance across different levels of perturbation.

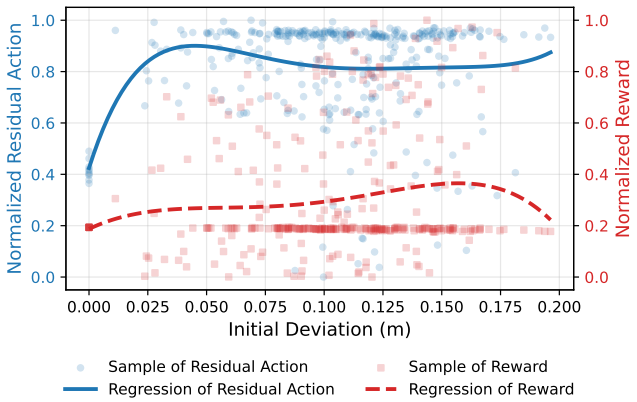


Figure 5: Normalized residual action and reward as a function of initial deviation (absolute value). Points represent sampled values over $5 \times 10^5$ executions, while lines show regression trends.

### *4.5. Real-World Evaluation*

To validate sim-to-real transfer, we deploy the learned residual policy directly on the physical robot without additional fine-tuning. Figure 6 shows the real robot setup and representative execution motion.

The sim-to-real transfer is facilitated by several design choices aimed at reducing the discrepancy between simulation and real-world execution. Actions are defined as incremental Cartesian end-effector commands, which are less sensitive to modelling errors than joint-level control. Additionally, contact interaction is approximated using a binary force proxy with an empirically defined threshold, avoiding reliance on accurate force modelling. Finally, the residual reinforcement learning formulation constrains the learned policy to bounded corrective actions around a stable behavioural cloning policy, improving robustness under sim-to-real mismatch.

Figure 6: Real-world experimental setup and representative execution.

We evaluate the system on two different bottle types: the nominal bottle used for demonstration and a geometrically distinct bottle. For each bottle type, we perform 10 independent trials. Table 2 reports the success rate for both bottle types.

Table 2: Real-world success rate per bottle type, mean and standard deviation over 10 trials.

| Bottle Type | Success Rate (%) |
|---|---|
| Nominal bottle | 78 ± 10% |
| Alternative bottle | 63 ± 12% |

The results indicate that the learned residual corrections transfer to the real robot, although with a decrease in success rate compared to simulation and under increased variability across bottle types. Despite this performance gap, the policy maintains stable grasping and consistent unscrewing behaviour, suggesting that the learned residual corrections remain effective under real-world conditions.

### *4.6. Discussion*

The results provide several insights into the behaviour and limitations of the proposed approach. First, the training dynamics show that the residual policy progressively adapts to increasing levels of perturbation introduced through curriculum learning. The drops in reward observed at each curriculum transition, followed by recovery, indicate that the policy incrementally acquires the ability to compensate for deviations while maintaining stability.

In simulation, the comparison between imitation alone and the addition of a residual policy confirms that behavioural cloning is highly sensitive to deviations from the demonstration conditions. In contrast, the residual policy effectively compensates for such deviations, achieving higher rewards and shorter episode lengths. This suggests that the residual component can correct errors that arise when operating outside the distribution of the expert trajectory. Moreover, the analysis of residual action magnitude shows an increase with environment variation compared to the no-deviation case, indicating active correction. At the same time, the reward remains relatively stable, reflecting consistent performance across different levels of perturbation.

Finally, the real-world results demonstrate that the learned residual policy transfers to the physical robot without additional fine-tuning, although with a reduction in success rate compared to simulation. Despite discrepancies between simulation and reality, including a simplified contact representation based on a binary force signal to approximate grasping, the policy maintains stable execution across different bottle types. This suggests that the residual learning framework, together with the simplified environment design, can partially bridge the sim-to-real gap by focusing on corrective adjustments, even if performance is not fully preserved in real-world conditions.

## 5. Conclusions

This work presented a residual learning approach built on top of behavioural cloning to improve robustness from a single demonstration. The proposed method combines a nominal policy obtained through imitation learning with a residual policy trained via reinforcement learning to compensate for deviations in initial conditions and object properties.

Experimental results in simulation show that the residual policy preserves performance under demonstration conditions while significantly improving robustness under variations. The curriculum learning strategy enables stable training under progressively increasing perturbations, and the analysis of residual behaviour highlights its role in adaptive correction.

Real-world experiments provide initial validation of the approach, showing that the learned policy can be transferred in a one-shot sim-to-real setting across different bottle types without additional training. While performance is reduced compared to simulation, the results suggest that residual reinforcement learning can contribute to improving generalization and robustness in manipulation tasks derived from limited demonstrations.


## Acknowledgements

This research has been funded through the R&D activity program with reference TEC-2024/TEC-62 and acronym iRoboCity2030-CM, granted by the Regional Government of Madrid through the Directorate-General for Research and Technological Innovation via Order 5696/2024; "iREHAB: AI-powered Robotic Personalized Rehabilitation" (ISCIII-AES-2022/003041, financed by Instituto de Salud Carlos III (ISCIII) and EU) and funded by the "ASEPEYO-UC3M: FASE IV" serious game-based rehabilitation program; and from the Carlos III University of Madrid.


## References


Bain, M., Sammut, C., 2001. A framework for behavioural cloning.

Brunke, L., Greeff, M., Hall, A. W., Yuan, Z., Zhou, S., Panerati, J., Schoellig, A. P., 12 2021. Safe learning in robotics: From learning-based control to safe reinforcement learning. Annual Review of Control, Robotics, and Autonomous Systems 5, 411–444.
DOI: 10.48550/arXiv.2108.06266

Foster, D. J., Block, A., Misra, D., 11 2024. Is behavior cloning all you need? understanding horizon in imitation learning. Advances in Neural Information Processing Systems 37.
DOI: 10.48550/arXiv.2407.15007

Johannink, T., Bahl, S., Nair, A., Luo, J., Kumar, A., Loskyll, M., Ojea, J. A., Solowjow, E., Levine, S., 5 2019. Residual reinforcement learning for robot control. Proceedings - IEEE International Conference on Robotics and Automation, 6023–6029.
DOI: 10.1109/ICRA.2019.8794127

Lin, T., Yin, Z.-H., Qi, H., Abbeel, P., Malik, J., 10 2024. Twisting lids off with two hands. Proceedings of Machine Learning Research 270, 5220–5235.
DOI: 10.48550/arXiv.2403.02338

Naranjo-Campos, F. J., Victores, J. G., Balaguer, C., 11 2024a. Expert-trajectory-based features for apprenticeship learning via inverse reinforcement learning for robotic manipulation. Applied Sciences 2024, Vol. 14, 14.
DOI: 10.3390/app142311131

Naranjo-Campos, F. J., Victores, J. G., Balaguer, C., 9 2024b. Method for bottle opening with a dual-arm robot. Biomimetics 2024, Vol. 9, 9.
DOI: 10.3390/biomimetics9090577

Ross, S., Gordon, G. J., Bagnell, J. A., Learning, M., 6 2011. A reduction of imitation learning and structured prediction to no-regret online learning.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Openai, O. K., 7 2017. Proximal policy optimization algorithms.
DOI: 10.48550/arXiv.1707.06347

Shi, Y., Chen, Z., Liu, H., Riedel, S., Gao, C., Feng, Q., Deng, J., Zhang, J., 2021. Proactive action visual residual reinforcement learning for contact-rich tasks using a torque-controlled robot. Proceedings - IEEE International Conference on Robotics and Automation 2021-May, 765–771.
DOI: 10.1109/ICRA48506.2021.9561162