2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Unknown Object Retrieval in Confined Space through Reinforcement Learning with Tactile Exploration

Xinyuan Zhao[1], Wenyu Liang[1], Xiaoshi Zhang[1,2], Chee Meng Chew[2] and Yan Wu[1]

***Abstract*— The potential of tactile sensing for dexterous robotic manipulation has been demonstrated by its ability to enable nuanced real-world interactions. In this study, the retrieval of unknown objects from confined spaces, which is unsuitable for conventional visual perception and gripper-based manipulation, is identified and addressed. Specifically, a tactile-sensorized tool stick that well fits in the narrow space is utilized to provide multi-point contact sensing for object manipulation. A reinforcement learning (RL) agent with a hybrid action space is then proposed to acquire the optimal policy for manipulating the objects without prior knowledge of their physical properties. To accelerate on-hardware training, a focused training strategy is adopted with the hypothesis that an agent trained on a small set of representative shapes can be generalized to a wide range of everyday objects. Additionally, a curriculum on terminal goals is designed to further accelerate the hardware-based training process. Comparative experiments and ablation studies have been conducted to evaluate the effectiveness and robustness of the proposed approach, which highlights the high success rate of our solution for retrieving everyday objects.**

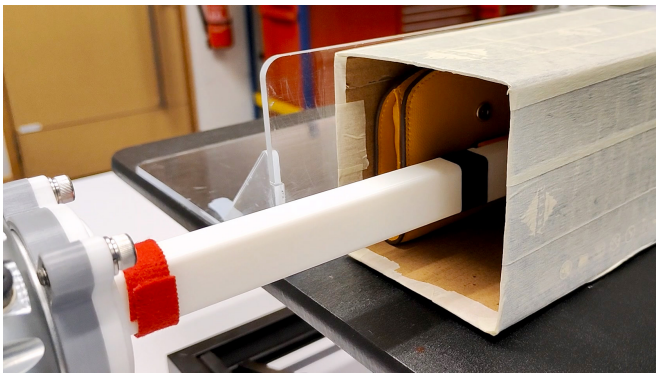

Fig. 1: Illustration of a robot retrieving a wallet in confined space. The robot is equipped with a tool stick where a magnetic-based tactile sensor is placed at the tip to measure contact information.

I. Introduction

The sense of touch is the first sensory modality that humans develop and plays an essential role in our interactions with the world. Illustrative examples are the human capabilities of getting things out of constrained spaces such as pockets or hard-to-reach areas under furniture solely through the sense of touch, where visual cues are not necessary or hard to access.

Though robots have demonstrated successful applications in various dexterous manipulation tasks nowadays, challenges exist if they are expected to be deployed for object retrieval from constrained spaces, primarily due to the compromised visual perception caused by unavoidable occlusion. Tactile sensing, which provides local, multi-point contact information about its environment and thus endows robots with a similar capability of touch as humans do, becomes crucial for such applications. Tactile sensors have been increasingly used in robotic manipulation tasks and achieved good results in object/material classification [1]–[3], grasping [4]–[6], and non-prehensile manipulation [7]–[10]. Compared to conventional force/torque sensors that typically capture single-point measurements, tactile sensors measure over an array of contacts, thereby yielding richer data for inferring geometry, texture, and other interactive features [11].

In previous studies on tactile-based object manipulation, it is a common assumption that complete knowledge of the manipulated objects (e.g. their geometries, physical properties, etc.) is known to the robots beforehand [10], [12]. Such assumption simplifies the problem by limiting the number of cases to search through, but on the other hand, it prevents the approaches from adapting to a wider range of objects like those commonly encountered in daily life. To mitigate the restrictions, learning-based approaches are promising to implicitly or explicitly infer the required properties through experience of interactions, without relying on prior knowledge or models [13].

This work aims to address the problem of retrieving diverse daily objects in confined spaces (e.g. retrieving a wallet fallen into a small gap) without prior knowledge of their geometrical and physical properties. Methods relying on visual perception like RGB-D cameras or motion capture systems [14]–[16] are not practical for such tasks because of occlusion caused by the confined space. Furthermore, spatial constraints may prohibit the use of conventional grippers. As a result, similar to humans using a stick to retrieve objects from narrow spaces, a custom tool stick is deployed for a robot to perform such retrieval tasks (seen in Fig. 1). A thin tactile sensor is attached at the tip of the tool stick, enabling real-time acquisition of 3D touch information upon contact with the objects. An RL-based

This research is supported by A*STAR partially under its Career Development Fund (C210812049) and its RIE2025 Industry Alignment Fund for Pre-Positioning Program (M21K1a0104).

[1] Robotics and Autonomous Systems Department, Institute for Infocomm Research (I^{2}R), A*STAR, Singapore 138632. {zhao_xinyuan, liang_wenyu, wuy}@i2r.a-star.edu.sg

[2] Department of Mechanical Engineering, National University of Singapore, Singapore 117575. zhangxiaoshi@u.nus.edu, chewcm@nus.edu.sg

motion planner is then implemented, allowing the robot to learn the desired manipulation policies solely from tactile measurements. Notably, as each tactile sensor has complex physical properties and the objects of interest are not limited to rigid ones, realistic modeling of the sensor and objects for simulation-based training becomes challenging, which encourages direct on-hardware training for the RL agent.

However, training an RL agent directly on a real robot can be time-consuming and cause severe wear and tear to the hardware due to extensive trial-and-error actions. Therefore, several techniques are adopted to simplify the problem and accelerate the training process so that the on-hardware training can be achieved within an acceptable duration. In this work, we propose that while dynamic properties of diverse objects remain unknown for exploration, a combination of two primitive shapes makes up the bulk of daily objects. As such, the RL agent is trained only on two representative shapes, standard cuboid and cylindrical objects, with different weights and materials. This focused training strategy is hypothesized to be sufficient to yield policies generalizable to a diverse range of unseen objects with different geometries and properties. Additionally, we investigate the use of a hybrid combination of one parameterized primitive action with non-primitives to accelerate the agent's exploration of indirect reward-generating actions. Besides, a learning curriculum is also applied to the terminal goals, encouraging the agent to generate better actions gradually.

The contributions of this paper are threefold as listed below. Firstly, an RL-based manipulation planner with tactile inputs and parameterized action primitive is developed for retrieving unknown objects in confined spaces. Secondly, an efficient training process is proposed, where standard cuboids and cylinders are used as representative objects to simplify the problem but strong generalizations on unseen daily objects are demonstrated. Lastly, a curriculum on terminal goals is developed to further accelerate the on-hardware training process. Comparative experiments and ablation studies are conducted to evaluate the effectiveness and robustness of the proposed solutions.

The rest of the paper is structured as follows. Section II provides a review of related works. Section III elaborates on the methodologies and techniques for designing the RL agent. Section IV presents the evaluations of the proposed approaches through comprehensive experiments and ablation studies. Finally, Section V summarizes our solutions, and suggests topics for future studies.

II. Related Work

In [4], tactile sensors are used in cable manipulation with robotic grippers. The pose of the cable and the frictional forces are estimated from the tactile images, which then guide two independent controllers for sliding and re-grasping motions. [10] explores object pivoting using tools held by a tactile-sensorized gripper, where the tactile measurements are used to estimate the poses of both the tool and the object. This approach assumes perfect knowledge of friction coefficients across all cases, which may limit its applicability to a diverse range of everyday objects. [7] uses the high-frequency components of tactile signals to distinguish between sliding and slipping in planar object pushing. While tested on cubic objects, the approach lacks validation on daily items with other shapes. [13] proposes an adaptive control framework for stably pushing unknown objects, using tactile sensors to estimate relative poses. This approach turns out to be easier to generalize across different situations than modeling the interaction dynamics.

In [17], a bio-inspired approach is introduced for tactile-reactive manoeuvre in densely cluttered environments. A discrete action space is employed to generate fixed motion primitives in different directions. Effective though, the lack of parameterized action primitives could potentially limit adaptability to different scenarios. In [15], a Q-learning framework is introduced to synthesize pushing and grasping primitives for handling unknown, cluttered objects. The framework uses two convolutional networks to evaluate primitives parameterized by pixel and orientation. The study presented in [12] decomposes complex manipulation behaviors into mechanically simple primitives, with each primitive assuming a specific contact pattern for easier state estimation. While efficient, the need for complete knowledge of object properties could limit its general applicability. [18] uses a hierarchical policy framework that utilizes a repertoire of behavior primitives to generate both the types of behaviors and their corresponding parameters for robotic manipulation. However, allowing robots to only explore from a repertoire of action primitives which are human-engineered may yield sub-optimal learning outcomes. As such, we believe that a combination of generic continuous actions and parameterized action primitives shall minimize bias introduced by primitives and grant the agent greater freedom to explore better policies.

In [19], an RL-based approach is employed to tackle the peg-in-hole problem involving objects with unknown geometries. It is shown that the policy improves faster in complex environments with a curriculum on the difficulty of the tasks. Curriculums on tasks and terminal goals are also developed in [20], [21] to expedite the training of RL policies in simulations with sparse rewards. Similarly, we introduce a curriculum on terminal goals to accelerate hardware-based training.

III. Methodology

An overview diagram of the system is shown in Fig. 2. It mainly consists of three components, namely the robot manipulator, the tactile sensor, and the RL-based motion planner. In this section, an introduction to the robot and the tactile sensor will be provided first, followed by a detailed presentation of the proposed RL-based motion planner.

A. Robotic System

Due to spatial constraints as discussed previously, a 3D-printed tool stick in length of 20 cm is used to interact with the objects. The tool is attached to a 7-DoF KUKA

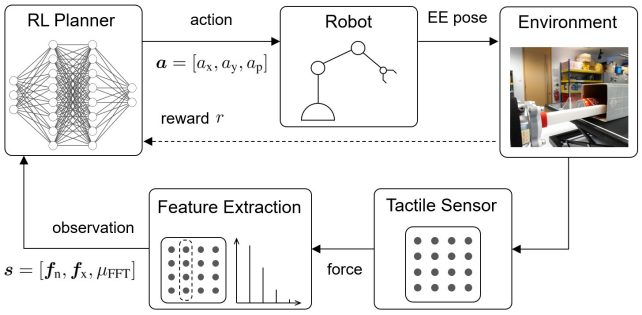

Fig. 2: An overview diagram of the system for the task of retrieving unknown objects in confined spaces. The dashed line indicates that reward signals from the environment are only used during training.

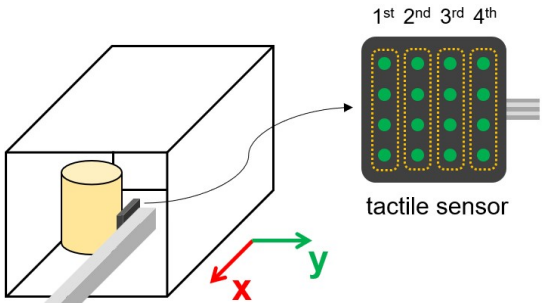

Fig. 3: Schematic of the workspace with coordinate framework and illustration of the 4x4 sensing array of the uSkin tactile sensor from XELA Robotics.

iiwa LBR 14 R820 robot, which can offer high-performance positional servo control and thus facilitate subtle tactile-reactive interactions.

To acquire real-time measurements of contact information, a flat, magnetic-based tactile sensor "uSPa44" from XELA Robotics is utilized [22]. This sensor has a 4×4 measuring array with three channels for each taxel, capturing calibrated contact forces in three orthogonal axes, namely one normal force and two shear forces. Different from vision-based tactile sensors, the XELA tactile sensor offers higher sampling and processing rates at 100 Hz, which better suits the problem of real-time feedback control. In addition, the XELA tactile sensor demonstrates strong resistance to impacts and constant wear and tear, which are frequently encountered challenges when training RL agents directly on hardware.

B. Manipulation Planning

The target of our work is to retrieve diverse daily objects without requiring their prior knowledge. As unknown physical properties will be encountered in this case, modeling the dynamics explicitly becomes challenging and unreliable. On the other hand, learning-based approaches enable the robot to learn from its experience of interactions and generalize to a wide range of unseen daily objects. Instead of modeling or estimating the interaction dynamics between the tool and the objects, we propose to develop an RL-based motion planner to learn the desired policies directly from the observed tactile data. In addition, it is assumed that the objects have been successfully located, and the implementation of a specific searching phase is out of the scope of this work.

1) Observation Space: At each timestep, the observation for the RL agent is solely derived from measured tactile data. Specifically, the observation is formally represented as $\boldsymbol{s} = [\boldsymbol{f}_n, \boldsymbol{f}_x, \mu_{FFT}] \in \mathbb{R}^9$ with $\mathbf{f}_n = [f_{n,1}, \ldots, f_{n,4}] \in \mathbb{R}^4$ and $\mathbf{f}_x = [f_{x,1}, \ldots, f_{x,4}] \in \mathbb{R}^4$, respectively. $f_{n,i}$ denotes the maximum normal force along the i-th column of the measured tactile data, as depicted in Fig. 3. Similarly, $f_{x,i}$ denotes the shear force in the x-axis along the i-th column that has the maximum absolute value. Inspired by tactile-based slippage detection during object manipulation [7], high-frequency components of contact forces are also incorporated in the observation to automatically capture potential slippage events. Here, μ_{FFT} denotes the mean of frequency components exceeding 30 Hz, calculated within a 32-sample window of the shear forces in the x-axis.

2) Action Space: The action space for the robot is a 3-dimensional vector denoted as $\boldsymbol{a} = [a_x, a_y, a_p] \in \mathbb{R}^3$. The first two components, a_x and a_y, correspond to the EE's displacements in the horizontal plane, along the intended direction of object retrieval and perpendicular to it, respectively. The third component, a_p, serves as an action primitive for backward adjustment of the EE, which becomes particularly beneficial for re-establishing contact when the manipulated object rolls. If $a_p \leq 0$, the robot follows the displacements dictated by a_x and a_y. Conversely, for $a_p > 0$, the robot disregards a_x and a_y and executes a predefined backward adjustment, the distance for which is determined by a_p. An illustration of the parameterized backward adjustment is shown in Fig. 4. The effectiveness and necessity of this parameterized action primitive are validated later in Section IV through an ablation study.

3) Reward Function Design: The reward function for the RL agent comprises both ongoing (dense) and terminal (sparse) components and is expressed as

$$r = r_t + r_o + r_f + r_g + r_p. \quad (1)$$

The term r_t serves as a time penalty at each timestep,

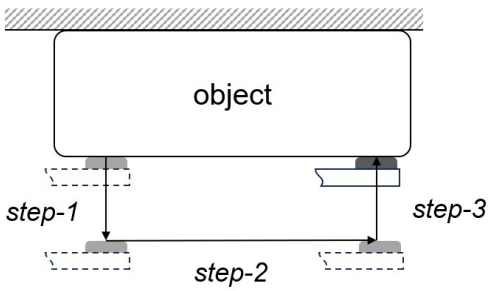

Fig. 4: Top-view illustration of the parameterized action primitive: the distance of backward adjustment, i.e. of *step-2*, is determined by the value of a_p.

aiming to speed up task completion. It is worth noting that when the action primitive a_{p} is utilized, its associated time penalty is calculated based on the equivalent number of timesteps it would take to achieve the same adjustment using only a_{x} and a_{y}.

The term r_{o} is defined by

$$r_{\mathrm{o}} = \alpha_{\mathrm{r,o}} d_{\mathrm{o}}, \tag{2}$$

and aims to reward the agent for moving the object in the intended direction, where d_{o} quantifies the forward displacement of the manipulated object within the current timestep. It is captured by OptiTrack, a vision-based motion tracking system during training.

The term r_{f} is designed to regulate the forces exerted on the object and is given by

$$r_{\mathrm{f}} = \begin{cases} 0 & \text{if } f_{\mathrm{n}}^{\max} < f_{\mathrm{n}}^{\mathrm{l}} \\ r_{\mathrm{f}}^{\mathrm{h}} & \text{if } f_{\mathrm{n}}^{\max} > f_{\mathrm{n}}^{\mathrm{h}} \\ (f_{\mathrm{n}}^{\max} - f_{\mathrm{n}}^{\mathrm{l}})^2 & \text{otherwise} \end{cases}. \tag{3}$$

Here, $f_{\mathrm{n}}^{\max}$ represents the maximum normal force measured by the tactile sensor. The threshold $f_{\mathrm{n}}^{\mathrm{l}} > 0$ is included to avoid the agent getting stuck at local minima and giving up interacting with the object. If $f_{\mathrm{n}}^{\max} > f_{\mathrm{n}}^{\mathrm{h}}$, additionally, the agent will release the press, reducing $f_{\mathrm{n}}^{\max}$ to $f_{\mathrm{n}}^{\mathrm{l}}$, while incurring a modest penalty for exceeding the force threshold.

The term r_{g} is calculated by

$$r_{\mathrm{g}} = \alpha_{\mathrm{r,g}} d_{\mathrm{g}}, \tag{4}$$

and it serves as a substantial positive reward issued only when the agent successfully accomplishes the predefined goal within a single episode, where $d_{\mathrm{g}} > 0$ specifies the distance that the manipulated object must advance from its original position to qualify for this terminal reward.

To encourage the agent to take bolder steps when it is familiar with smaller ones (similar to a toddler learning how to walk), d_{g} is dynamically adjusted according to a curriculum designed to gradually increase the complexity. Initially set at $d_{\mathrm{g}}^{\mathrm{l}}$, d_{g} is increased by one (in mm) each time the agent accomplishes the goal until it reaches $d_{\mathrm{g}}^{\mathrm{h}}$. This adaptation of terminal goals provides strong incentives for goal-oriented behaviors and encourages the agent to tackle more complex tasks as it evolves. This will allow the agent to derive a more efficient action set to accomplish the task with a shorter duration. At the same time, it also reduces the training requirements. The effectiveness of this curriculum will be validated in Section IV-D through an ablation study.

Lastly, the term r_{p} punishes the action primitive in cases where it fails to re-establish contact with the object. Specifically, a backward adjustment is deemed failed if the robot is unable to regain contact with the object within a certain period after initiating the primitive.

C. Implementation Details

The motion planner is realized through the Soft-Actor-Critic (SAC) algorithm [23], which is provided by *Stable-Baselines3* [24], a reliably-implemented library for state-of-the-art RL algorithms. SAC is a prominent off-policy algorithm known for its advanced exploration capabilities and high sampling efficiency, offering the potential for accelerated training on hardware compared to on-policy alternatives [25].

The RL motion planner is trained directly on the real iiwa robot. The *environment* for the RL agent is implemented in MATLAB while the RL agent itself is implemented in Python. The communications between each block shown in Fig. 2 are realized through topics and services provided by the Robot Operating System (ROS).

The action components a_{x}, a_{y} and a_{p} produced by the agent were all normalized to the range $[-1, 1]$. Before being executed by the robot, the action is scaled by a factor $\boldsymbol{\alpha}_{\mathrm{a}} = [\alpha_{\mathrm{a,x}}, \alpha_{\mathrm{a,y}}, \alpha_{\mathrm{a,p}}]$, where $\alpha_{\mathrm{a,x}} = 0.5$, $\alpha_{\mathrm{a,y}} = 0.2$ and $\alpha_{\mathrm{a,p}} = 30$ (all in mm), respectively.

For the SAC algorithm used, both the actor and critic networks consist of fully connected layers with two hidden layers, each containing 256 units, and Rectified Linear Units (ReLUs) are applied as the activation functions. Detailed specifications of other relevant parameters can be found in Table I.

TABLE I: Parameter Specification

variable	value	variable	value	variable	value
r_{t}	-0.1	$r_{\mathrm{f}}^{\mathrm{h}}$	-10.0	r_{p}	-10.0
$f_{\mathrm{n}}^{\mathrm{l}}$	0.2 N	$f_{\mathrm{n}}^{\mathrm{h}}$	1.5 N	$d_{\mathrm{g}}^{\mathrm{l}}$	20.0 mm
$d_{\mathrm{g}}^{\mathrm{h}}$	50.0 mm	$\alpha_{\mathrm{r,o}}$	5.0	$\alpha_{\mathrm{r,g}}$	2.0

IV. Experiments and Results

This section begins by presenting the RL training process, followed by a detailed comparison between our proposed solution and a heuristic-based baseline method. Additionally, the results of two ablation studies are also presented to further validate the effectiveness of the parameterized action primitive and the curriculum on terminal goals, respectively.

A. RL Planner Training

Fig. 5(a) shows all the objects used for training the RL planner. Objects labeled *Obj-1* to *Obj-4* are representatives of standard geometric cuboids, varying in weight and material. Similarly, objects labeled *Obj-5* to *Obj-8* represent standard geometric cylinders with different physical properties as well. Please refer to the supplementary video for detailed information on the geometries, weights, and materials of these objects.

To accelerate the training process, we adopt a focused training strategy to first train two separate policies on hardware and then merge them offline. Specifically, a model is first trained for retrieving cuboid objects using *Obj-1* to *Obj-4* and reaches convergence after about 125 episodes. Then, another model is trained from scratch to handle cylindrical objects using *Obj-5* to *Obj-8*, the convergence of which is achieved after about 300 episodes. With the replay buffers from both the two models collected separately,

the last step is to merge the two replay buffers to train an integrated model offline. This focused training strategy limits the total time spent on hardware training to approximately five hours and meanwhile achieves good generalization to other objects except for the ones used for training.

The integrated model derived from the last step is directly used in subsequent comparative evaluations, requiring no additional online refinements or pre-training on new objects. Importantly, it is worth noting that the motion capture system OptiTrack is only used in the training stage to help obtain the rewards while for evaluations, the agent has to conduct the retrieval tasks relying on learned policies with tactile inputs alone.

(a) Eight objects used in the training set, where *Obj-5* is empty and light while *Obj-8* is stuffed and much heavier.

(b) Twelve objects used in the test set, where *Obj-20* will be manipulated on a piece of corrugated paper, different from the other cases.

Fig. 5: Illustration of all the objects used for training and testing the RL agent and the heuristic-based method in our experiments.

B. Comparative Evaluation

To evaluate the performance of the proposed RL planner, a heuristic-based approach, referred to as *heuristic planner* hereinafter, is used as the baseline. The heuristic planner is programmed to maintain the maximum normal force exerted on the object, i.e., $f_{\mathrm{n}}^{\max}$, within a range of $[f_{\mathrm{n}}^{\mathrm{l}}, f_{\mathrm{n}}^{\mathrm{h}}]$ N. Once the normal force is established, the tool stick will move in the intended direction of retrieval at a constant speed $0.8\alpha_{\mathrm{a,x}}$, which is 80 % of the maximum speed for the RL planner. When the condition $f_{\mathrm{n},1} > f_{\mathrm{n}}^{\mathrm{l}}$ *AND* $f_{\mathrm{n},i} \leq f_{\mathrm{n}}^{\mathrm{l}}$, $i = 2, 3, 4$ is met, the heuristic planner will generate a backward adjustment similar to what the parameterized action primitive would do, as illustrated in Fig. 4. The only difference here is that the backward distance of *step-2* is fixed at 10.0 mm according to our experience in training the RL planner.

Fig. 6 shows the experimental setup for the comparative evaluation of the two planners. Both planners are required to manipulate objects of varying shapes and sizes, and a task is considered successfully completed if the object is advanced by 100.0 mm within 60 seconds. A total of twenty distinct objects/scenarios are included for the evaluation, as shown in Fig. 5. While our proposed RL planner was initially trained on *Obj-1* to *Obj-8*, the test set comprises entirely new objects, labeled as *Obj-9* to *Obj-20*. Both methods undergo five trials for each object to ensure reliable assessment. Notably, the physical attributes of the objects such as weight, material, geometry, or friction coefficient are unknown to both methods. Please refer to the supplementary video for more information.

The results of the comparative evaluation are tabulated in Table II. It highlights that the proposed planner achieved a success rate of up to 90 % which is much higher than the baseline method on average, particularly when handling objects with complex geometries or textures (e.g. *Obj-15* and *Obj-18*). For instance, in the trails with *Obj-18* shown in Fig. 7, the mug exhibits two movement modalities: it rotates when only its body contacts the "wall" but slides forward when the handle also engages the wall, generally incurring higher friction. The RL planner has implicitly learned to correlate contact force patterns with rewards and to adjust accordingly, which is lacking in the heuristic planner. Consequently, the heuristic planner struggles when the mug switches movement modalities, while the RL planner successfully adapts and completes all five trials.

C. Effectiveness of Parameterized Action Primitive

An ablation study is conducted to assess the significance of the incorporated parameterized action primitive. To this end, the primitive is removed from the proposed RL model, and

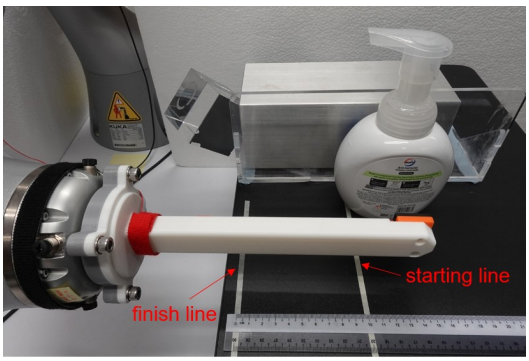

Fig. 6: Illustration of the experimental setup for the comparative evaluation.

TABLE II: Comparison of Success Rates: Our Proposed Method vs. Heuristic-based Method

	Obj-1	Obj-2	Obj-3	Obj-4	Obj-5	Obj-6	Obj-7	Obj-8	Obj-9	Obj-10	Obj-11	Obj-12
Ours	5/5	5/5	5/5	5/5	5/5	5/5	4/5	5/5	5/5	5/5	5/5	5/5
Heuristic	5/5	5/5	0/5	0/5	5/5	5/5	5/5	0/5	5/5	0/5	5/5	5/5
	Obj-13	Obj-14	Obj-15	Obj-16	Obj-17	Obj-18	Obj-19	Obj-20	**Test Set Only**		**Overall**	
Ours	4/5	5/5	4/5	5/5	3/5	5/5	3/5	5/5	**90 %**		**93 %**	
Heuristic	5/5	5/5	1/5	2/5	2/5	0/5	1/5	2/5	**55 %**		**58 %**	

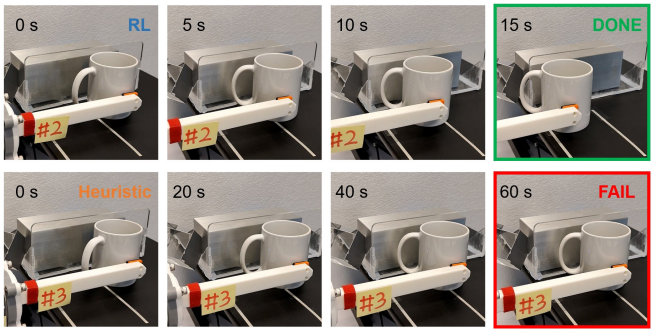

Fig. 7: Snapshots from the comparative experiments with *Obj-18*. Top: The RL planner successfully completed the task in 15 s; Bottom: The heuristic planner failed because of timeout.

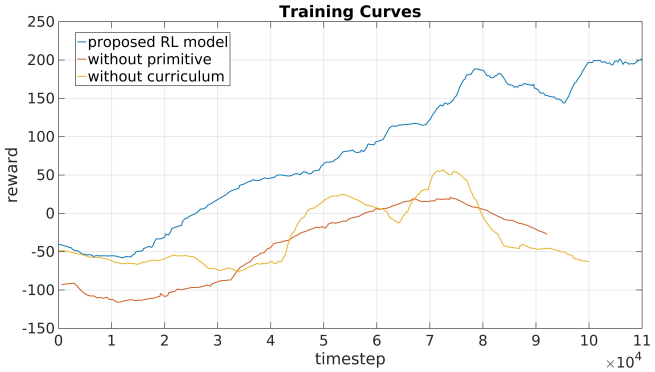

Fig. 8: Comparison of training curves for three different models learning policies for cylindrical objects manipulation.

subsequently, the remaining model is trained to manipulate cylindrical objects *Obj-5* to *Obj-8*, replicating the approach previously mentioned in Section IV-A. The training curves for both models are presented in Fig. 8.

Upon comparison of the training curves, it is evident that the model lacking the action primitive significantly underperforms the proposed RL model. The capability to adjust backwards for reconfiguring contact locations is not acquired by this model. Consequently, the resulting policy tried to minimize interactions with objects and failed all evaluation trials involving the cylindrical objects in the training set.

D. Effectiveness of Curriculum on Terminal Goal

Another ablation study is carried out to assess the effectiveness of the designed curriculum on terminal goals in the training process. In this experiment, the proposed RL model is retrained from scratch on cylindrical objects, with the initial terminal-goal distance d_g set to d_g^h, thus bypassing the curriculum. The training curves for both models are depicted in Fig. 8.

In the ablation study, it was observed that the model without curriculum took approximately 50,000 timesteps to reach the terminal goal for the first time. In contrast, the proposed RL model achieved its first goal after only about 17,000 timesteps, leading to a more rapid improvement in the learned policy, as revealed by the training curves. Additionally, the policy acquired by the ablated model demonstrated poor generalization across the four training objects, resulting in a decline in the learned policy within the second half of the training process.

V. Conclusion

In this study, a tactile-guided RL-based planner for manipulating unknown objects in confined spaces was introduced. Tactile information is leveraged to enable interactions with objects when visual perception is compromised by occlusion. A relationship between the pattern of interaction forces and task completion is implicitly built by the RL-based planner, eliminating the need for prior knowledge of the physical attributes of the objects. Various strategies and techniques, including a focused training process, a design of hybrid action space and a learning curriculum on goals, are employed to simplify the problem and accelerate the on-hardware training process.

The obtained RL planner demonstrated remarkable performance compared to a heuristic-based baseline method and generalizes to a broad range of everyday objects with various physical properties. The achieved generalization is attributed to the higher level of abstraction employed in the methodology, which focuses on learning policies to manipulate objects directly from tactile measurements rather than modeling the underlying dynamics. Future research will explore the manipulation of unknown objects in more complex environments, such as sloped or vertical spaces, where gravitational forces will contribute additional complexity.

REFERENCES

[1] D. Driess, D. Hennes, and M. Toussaint, "Active multi-contact continuous tactile exploration with gaussian process differential entropy," in *International Conference on Robotics and Automation (ICRA)*, 2019, pp. 7844–7850.

[2] M. Kaboli, K. Yao, D. Feng, and G. Cheng, "Tactile-based active object discrimination and target object search in an unknown workspace," *Autonomous Robots*, vol. 43, pp. 123–152, 2019.

[3] R. Gao, T. Taunyazov, Z. Lin, and Y. Wu, "Supervised autoencoder joint learning on heterogeneous tactile sensory data: Improving material classification performance," in *International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 10 907–10 913.

[4] Y. She, S. Wang, S. Dong, N. Sunil, A. Rodriguez, and E. Adelson, "Cable manipulation with a tactile-reactive gripper," *The International Journal of Robotics Research*, vol. 40, pp. 1385–1401, 2021.

[5] S. Kim and A. Rodriguez, "Active extrinsic contact sensing: Application to general peg-in-hole insertion," in *International Conference on Robotics and Automation (ICRA)*, 2022, pp. 10 241–10 247.

[6] W. Liang, F. Fang, C. Acar, W. Q. Toh, Y. Sun, Q. Xu, and Y. Wu, "Visuo-tactile feedback-based robot manipulation for object packing," *IEEE Robotics and Automation Letters*, vol. 8, pp. 1151–1158, 2023.

[7] M. Meier, G. Walck, R. Haschke, and H. J. Ritter, "Distinguishing sliding from slipping during object pushing," in *International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5579–5584.

[8] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," in *International Conference on Robotics and Automation (ICRA)*, 2019, pp. 818–824.

[9] N. Doshi, O. Taylor, and A. Rodriguez, "Manipulation of unknown objects via contact configuration regulation," in *International Conference on Robotics and Automation (ICRA)*, 2022, pp. 2693–2699.

[10] Y. Shirai, D. K. Jha, A. U. Raghunathan, and D. Hong, "Tactile tool manipulation," in *International Conference on Robotics and Automation (ICRA)*, 2023, pp. 12 597–12 603.

[11] Q. Li, O. Kroemer, Z. Su, F. F. Veiga, M. Kaboli, and H. J. Ritter, "A review of tactile information: Perception and action through touch," *IEEE Transactions on Robotics*, vol. 36, pp. 1619–1634, 2020.

[12] F. R. Hogan, J. Ballester, S. Dong, and A. Rodriguez, "Tactile dexterity: Manipulation primitives with tactile feedback," in *International Conference on Robotics and Automation (ICRA)*, 2020, pp. 8863–8869.

[13] J. Lloyd and N. F. Lepora, "Goal-driven robotic pushing using tactile and proprioceptive feedback," *IEEE Transactions on Robotics*, vol. 38, pp. 1201–1212, 2021.

[14] F. R. Hogan, E. R. Grau, and A. Rodriguez, "Reactive planar manipulation with convex hybrid mpc," in *International Conference on Robotics and Automation (ICRA)*, 2018, pp. 247–253.

[15] A. Zeng, S. Song, S. Welker, J. Lee, A. Rodriguez, and T. Funkhouser, "Learning synergies between pushing and grasping with self-supervised deep reinforcement learning," in *International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4238–4245.

[16] C. Song and A. Boularias, "A probabilistic model for planar sliding of objects with unknown material properties: Identification and robust planning," in *International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 5311–5318.

[17] W. Liang, Q. Ren, X. Chen, J. Gao, and Y. Wu, "Dexterous manoeuvre through touch in a cluttered scene," in *International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6308–6314.

[18] S. Nasiriany, H. Liu, and Y. Zhu, "Augmenting reinforcement learning with behavior primitives for diverse manipulation tasks," in *International Conference on Robotics and Automation (ICRA)*, 2022, pp. 7477–7484.

[19] S. Dong, D. K. Jha, D. Romeres, S. Kim, D. Nikovski, and A. Rodriguez, "Tactile-rl for insertion: Generalization to objects of unknown geometry," in *International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6437–6443.

[20] C. Florensa, D. Held, X. Geng, and P. Abbeel, "Automatic goal generation for reinforcement learning agents," in *International Conference on Machine Learning (ICML)*, 2018, pp. 1515–1528.

[21] A. Campero, R. Raileanu, H. Küttler, J. B. Tenenbaum, T. Rocktäschel, and E. Grefenstette, "Learning with amigo: Adversarially motivated intrinsic goals," in *International Conference on Learning Representations (ICLR)*, 2021.

[22] T. P. Tomo, S. Somlor, A. Schmitz, S. Hashimoto, S. Sugano, and L. Jamone, "Development of a hall-effect based skin sensor," in *Sensors*, 2015, pp. 1–4.

[23] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning (ICML)*, 2018, pp. 1861–1870.

[24] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, pp. 1–8, 2021. [Online]. Available: http://jmlr.org/papers/v22/20-1364.html

[25] J. Ibarz, J. Tan, C. Finn, M. Kalakrishnan, P. Pastor, and S. Levine, "How to train your robot with deep reinforcement learning: lessons we have learned," *The International Journal of Robotics Research*, vol. 40, pp. 698–721, 2021.