DyWA: Dynamics-adaptive World Action Model for Generalizable Non-prehensile Manipulation

Jiangran Lyu[1,2], Ziming Li[1,2], Xuesong Shi[2], Chaoyi Xu[2], Yizhou Wang[1,†], He Wang[1,2,†]
[1]CFCS, School of Computer Science, Peking University [2]Galbot
https://pku-epic.github.io/DyWA/

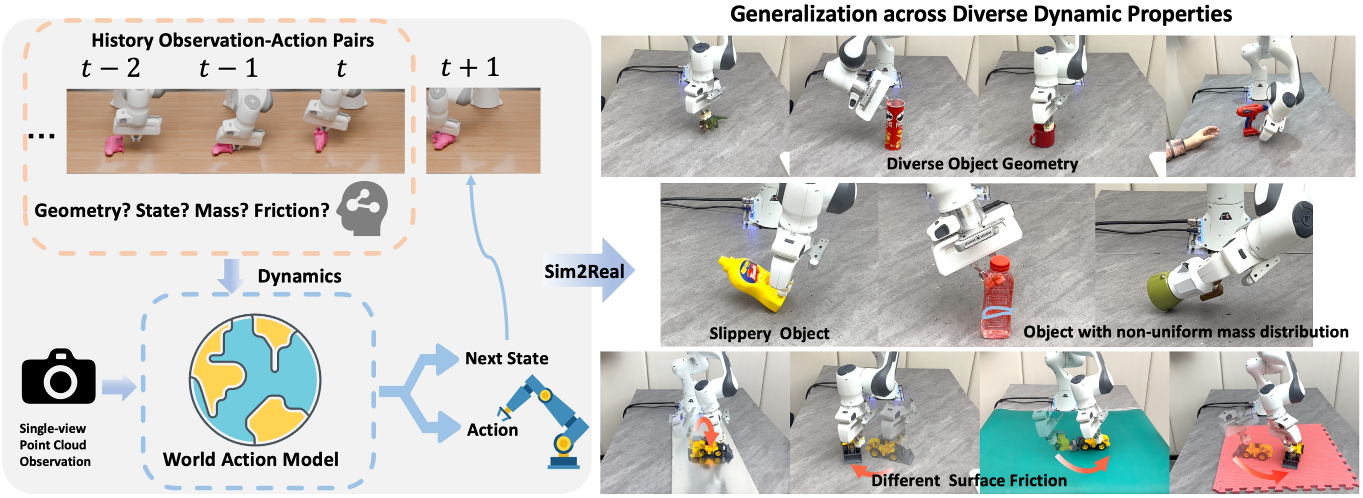

Fig. 1: **Illustration of the high-level idea and generalization ability of DyWA.** Given a target object's 6D pose and *single-view* object point cloud, our non-prehensile manipulation policy aims to rearrange the object without grasping. **Left**: Our key insight is to enhance action learning by jointly predicting future states while adapting to dynamics from historical trajectories. (For clarity, rendered images are used for visualization, while the actual visual input consists of partial point clouds.) **Right**: After being trained in simulation, our policy achieves zero-shot sim-to-real transfer and generalizes across diverse dynamic properties, including variations in object geometry, object physical property (e.g., slipperiness and non-uniform mass distribution), and surface friction.

Abstract— **Nonprehensile manipulation is crucial for handling objects that are too thin, large, or otherwise ungraspable in unstructured environments. While conventional planning-based approaches struggle with complex contact modeling, learning-based methods have recently emerged as a promising alternative. However, existing learning-based approaches face two major limitations: they heavily rely on multi-view cameras and precise pose tracking, and they fail to generalize across varying physical conditions, such as changes in object mass and table friction. To address these challenges, we propose the Dynamics-Adaptive World Action Model (DyWA), a novel framework that enhances action learning by jointly predicting future states while adapting to dynamics variations based on historical trajectories. By unifying the modeling of geometry, state, physics, and robot actions, DyWA enables more robust policy learning under partial observability. Compared to baselines, our method improves the success rate by 31.5% using only single-view point cloud observations in the simulation. Furthermore, DyWA achieves an average success rate of 68% in real-world experiments, demonstrating its ability to generalize across diverse object geometries, adapt to varying table friction, and robustness in challenging scenarios such as half-filled water bottles and slippery surfaces.**

I. Introduction

Non-prehensile manipulation—such as pushing, sliding, toppling, and flipping—greatly extends the capabilities of robotic manipulators beyond traditional pick-and-place operations. These dexterous actions enable robots to handle tasks where grasping is infeasible or inefficient due to object geometry, clutter, or workspace constraints. Over the years, significant progress has been made in this area, particularly through planning-based approaches [10, 13, 11, 16]. While effective, these methods typically rely on prior knowledge of object properties, such as mass, friction coefficients, or even complete CAD models, which limits their practicality in real-world applications. Recently, learning-based methods [17] have emerged as a promising alternative, improving generalization across diverse unseen objects. In this paradigm, policies are trained in simulation and then deployed zero-shot in the real world. For instance, HACMan [18] leverages vision-based reinforcement learning (RL) on object surface point clouds to determine contact locations and motion directions for executing action primitives. Similarly, CORN [4] employs a teacher-student distillation framework, where a teacher policy is first trained using RL with privileged state knowledge and then distilled into a vision-based student policy.

However, these methods face two key limitations that hinder robust real-world deployment. First, as noted by [5], they rely heavily on multi-view cameras for accurate object geometry

†: Corresponding authors

and on precise pose tracking modules for state estimation. In practical settings, nevertheless, multi-view setups may be unavailable, and tracking modules are often imperfect, leading to unreliable state information. Second, these approaches struggle to generalize across diverse physical conditions, such as variations in object mass and table friction, as their models primarily focus on geometry while overlooking the underlying dynamics.

In contrast, we argue that a generalizable non-prehensile manipulation policy in a realistic robotic setting should not only accommodate diverse object geometries but also adapt to varying physical properties, all while relying solely on a single-camera setup without the need for additional tracking modules.

To achieve this objective, we first experiment with the popular teacher-student policy distillation framework under this challenging setting. Our experiments reveal that while the RL teacher policy, when given oracle information, achieves high performance across diverse dynamic conditions, the distilled student policy, relying on partial observations, suffers from a significant performance drop. We then identify three key factors contributing to this issue. First, severe partial observability from single-view setting harms action learning by omitting critical geometric cues. Second, the Markovian student model inherently learns only an averaged behavior across diverse physical variations, resulting in suboptimal performance. Third, conventional distillation methods supervise only latent features and final actions, which is insufficient to capture the underlying dynamics necessary for effectively learning contact-rich action.

To address the first two issues, we introduce a Dynamics Adaptation Module, inspired by RMA [6], which encodes historical observation-action pairs to model dynamic properties, incorporating both sufficient geometric and physical knowledge. For the third issue, we extend conventional action learning by enforcing the joint prediction of actions and their corresponding future states. This reformulation transforms the conventional action model into a world action model, introducing additional supervisory signals beyond those provided by the teacher. This synergistic learning paradigm improves imitation loss optimization and significantly enhances overall success rates. Finally, to guide the world action model with the dynamics embedding adequately, we bridge the two parts using Feature-wise Linear Modulation (FiLM) conditioning. In short, we propose a novel policy learning framework that jointly predicting future states while adapting dynamics from historical trajectories. We term our approach **DyWA (Dynamics-Adaptive World Action Model)**.

We conduct extensive experiments in both simulation and the real world to evaluate the effectiveness and generalization of our policy, comparing it against baseline methods. To address the lack of a unified benchmark for non-prehensile manipulation, we build a comprehensive benchmark based on CORN, varying camera views (one or three) and the presence of a ground-truth pose tracker. Our method demonstrates the superiority of its model design across different settings, with a 31.5% improvement in success rate than baselines. Furthermore, comprehensive ablation studies validate the synergistic benefits of dynamics adaptation and world modeling when jointly learning actions. Finally, real-world experiments show that DyWA generalizes across object geometries at a 68% success rate and adapts to physical variations like table friction. It also achieves robustness in handling non-uniform mass distributions (e.g., half-filled water bottles) and slippery objects. Additionally, we showcase its applications combined with VLM, which assists human or grasping models with thin or wide objects.

In summary, this work makes the following contributions:

- We propose DyWA, a novel policy learning approach by jointly predicting future states, with adaptation of dynamics modeling from historical trajectories.
- We improve generalizable non-prehensile manipulation, reducing dependence on multi-camera setups and pose tracking modules while ensuring robustness across varying physical conditions.
- We provide a comprehensive simulation benchmark for generalizable non-prehensile manipulation. Our approach surpasses all baseline methods, and we showcase its effectiveness through several real-world applications.

II. Method

A. Task Formulation

Following HACMan and CORN, we focus on the task of 6D object rearrangement via non-prehensile manipulation. The robot's objective is to execute a sequence of non-prehensile actions (i.e., pushing, flipping) to move an object on the table to a target 6D pose. We define the goal pose G as a 6DoF transformation relative to the object's initial pose, assuming both are stable on the table. The task state S_t at timestep t is represented by the relative transformation between the object's current pose and the goal pose. The observation space includes a point cloud P_t captured by a depth sensor, robot's joint positions and velocities J_t, and the end-effector pose E_t computed via forward kinematics.

B. World Action Model

a) Observation and Goal Encoding.: Our model takes Observation and Goal Description as input, encoding different modalities using individual encoders. For the partial point cloud observation, we process it using a simplified PointNet++ [14] to obtain f_t^P, striking a balance between efficiency and capacity. The architectural details are provided in the supplementary material. For robot proprioception, we separately encode joint positions and velocities (f_t^J) and the end-effector pose (f_t^E) using shallow MLPs. For the Goal Description, instead of relying on the unknown task state S_t, we construct a visual goal representation by transforming the initial point cloud P_0 to the goal pose, yielding $P_G = GP_0$. This goal point cloud is then encoded using the same network as the observation point cloud encoder, ensuring consistency in feature extraction.

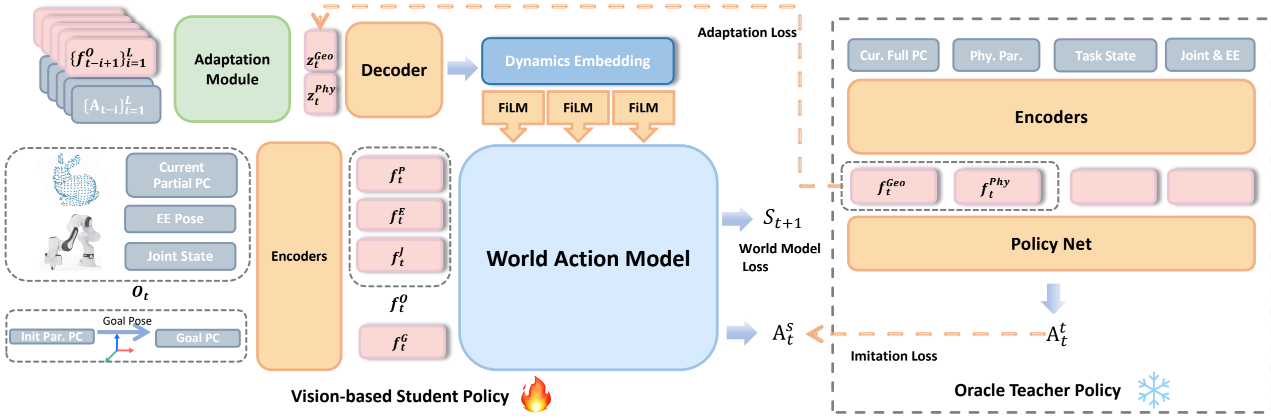

Fig. 2: Our World Action Model processes the embeddings of the current observation (partial point cloud, end-effector pose, and joint state) and the goal point cloud (transformed from the initial partial observation) to predict the robot action and next state. Additionally, an adaptation module encodes historical observations and actions, decoding them into the dynamics embedding that conditions the model via FiLM. A pre-trained RL teacher policy (right) supervises both the action and adaptation embedding using privileged full point cloud and physics parameter embeddings.

b) State-based World Modeling.: We enforce the end-to-end model that jointly makes action decisions and predicts their outcomes, creating a synergistic learning process that, in turn, improves action learning. Specifically, the observation and goal embeddings are processed through MLPs to produce both the action A_t and the next task state S_{t+1}, with supervision signals separately derived from the teacher policy and simulation outcomes. Our object-centric world model represents the environment using task state S_{t+1} instead of high-dimensional visual signals, enabling the policy to focus on task-relevant dynamics. To represent rotations, we adopt the 9D representation [7, 9], and define the world model loss as:

$$\mathcal{L}_{world} = \|T_{t+1} - \hat{T}_{t+1}\|^2 + |\mathbf{R}_{t+1} - \hat{\mathbf{R}}_{t+1}| \tag{1}$$

where $T_{t+1} \in \mathbb{R}^3$ and $\mathbf{R}_{t+1} \in SO(3)$ are the predicted translation and rotation, while $\hat{T}_{t+1} \in \mathbb{R}^3$ and $\hat{\mathbf{R}}_{t+1} \in SO(3)$ denote the ground-truth transformation obtained from simulation outcomes after action execution. Additionally, we employ an imitation loss, defined as the L2 loss between the predicted action and the teacher action:

$$\mathcal{L}_{imitation} = \|A^s_t - A^t_t\|^2 \tag{2}$$

C. Dynamics Adaptation

To enhance the world model's ability to adapt to diverse dynamics, we extract abstract representations of environmental variations from historical trajectories. Our approach distills teacher knowledge regarding full point cloud and physical parameter into an adaptation embedding, which is subsequently decoded into the dynamics embedding. This embedding then conditions the world action model through a learnable feature-wise linear modulation mechanism.

a) Adaptation Embedding.: we design an adaptation module that processes sequential observation-action pairs to compensate for missing geometry and physics knowledge in the current partial observation. Specifically, at each timestep, we concatenate the observation embeddings $f^O_t = \{f^P_t, f^J_t, f^E_t\}$ with the previous action embedding f^A_{t-1}, where the action embedding is obtained via a shallow MLP. We construct an input sequence of L past observation-action tuples which is then processed by a 1D CNN-based adaptation module, for extracting a compact adaptation embedding:

$$z_t = Embed(\{\text{concat}(f^O_{t-i-1}, f^A_{t-i-2})\}^L_{i=1}) \tag{3}$$

To ensure meaningful representation learning, we supervise the adaptation embedding using the concatenation of the full point cloud embedding and physics embedding from the teacher encoder.

$$\mathcal{L}_{adapt} = \|z^{Geo,Phy}_t - concat(f^{Geo}_t, f^{Phy}_t)\|^2 \tag{4}$$

b) Dynamics Conditioning.: Once the adaptation embedding is obtained, we decode it into the dynamics embedding, which serves as a conditioning input for the world action model via Feature-wise Linear Modulation (FiLM). FiLM [12] dynamically modulates the intermediate feature representations of the world action model by applying learned scaling and shifting transformations, allowing the model to adapt to varying dynamics. Each FiLM block consists of two shallow MLPs which take the dynamics embedding as input and output the modulation parameters γ and β for each latent feature f:

$$\text{FiLM}(f|\gamma, \beta) = \gamma f + \beta \tag{5}$$

We integrate FiLM blocks densely in the early layers of the world action model while leaving the final layers unconditioned. The technique that has proven highly effective in integrating language guidance into vision encoders [3, 1]. In our case, this mechanism allows the dynamics embedding to selectively influence feature representations, enabling adaptive adjustments to the model's behavior based on the underlying dynamics.

Methods	Action Type	Known State (3 view)		Unknown State (3 view)		Unknown State (1 view)	
		Seen	Unseen	Seen	Unseen	Seen	Unseen
HACMan [18]	Primitive	3.8(42.2)	5.7(39.4)	3.0(23.6)	4.1(26.5)	2.1(19.2)	
CORN [4]	Closed-loop	86.8	79.9	46.0	47.8	29.0	29.8
CORN (PN++)	Closed-loop	87.3	84.3	76.1	75.7	50.7	49.4
Ours	Closed-loop	**87.9**	**85.0**	**85.8**	**82.3**	**82.2**	**75.0**

TABLE I: Quantitative results measured by success rate in the simulation benchmark. For HACMan, we also reports its performance given 3 DoF planar goal(i.e.$[\Delta x, \Delta y, \Delta \theta]$) in parentheses. Note that the third track with unknown state and single view camera is the most realistic and challenging track for fully comparison of each methods.

D. Action Space with Variable Impedance

To enable adaptive force interaction between the robot and object, we employ variable impedance control as the low-level action execution mechanism. This allows the robot to dynamically regulate the interaction force based on task demands. Specifically, the action space of our policy consists of the subgoal residual of the end effector, $\Delta T_{ee} \in SE(3)$, along with joint-space impedance parameters. The joint-space impedance is parameterized by positional gains ($P \in \mathbb{R}^7$) and damping factors ($\rho \in \mathbb{R}^7$), where the velocity gains are computed as $D = \rho\sqrt{P}$. To execute the commanded end-effector motion, we first solve for the desired joint position using inverse kinematics with the damped least squares method [2]:

$$q_d = q_t + IK(\Delta T_{ee}) \tag{6}$$

Then, the desired joint position q_d and impedance parameters K, D are applied to a joint-space impedance controller to generate impedance-aware control commands for the robot. We utilize the widely adopted Polymetis API [8] for implementation.

E. Training Protocol

The overall learning objective is formulated as the sum of the imitation loss, world model loss, and adaptation loss:

$$\mathcal{L} = \mathcal{L}_{imitation} + \mathcal{L}_{world} + \mathcal{L}_{adapt} \tag{7}$$

We begin by training the teacher policy for 200K iterations in simulation using PPO. Subsequently, we employ DAgger to train the student policy under teacher supervision for 500K iterations. To enhance robustness and generalization, we introduce domain randomization during training by varying the object's mass, scale, and friction, as well as the restitution properties of the object, table, and robot gripper. The object scale is adjusted such that its largest diameter remains within a predefined range. To further improve sim-to-real transfer, we inject small perturbations into the torque commands, object point cloud, and goal pose when training the student policy.

III. Experiments

A. Benchmarking Tabletop Non-prehensile Rearrangement in Simulation

We evaluate our method alongside several baselines within a unified simulation environment to enable a fair comparison of their performance. Although prior works [4, 18] have developed their own simulation environments for training and validating non-prehensile manipulation policies, there remains a lack of a standardized benchmark for evaluating both existing and future approaches. To bridge this gap, we establish a comprehensive benchmark based on the CORN setting. Specifically, we adopt the IsaacGym simulation environment and utilize 323-object asset from DexGraspNet [15] for training. Additionally, we enrich the task setting by introducing an unseen object test set, consisting of 10 geometrically diverse objects, each scaled to five different sizes, resulting in a total of 50 evaluation objects. Furthermore, we introduce two additional perception dimensions: (i) single-view vs. multi-view (three-camera) observations and (ii) whether known object poses for constructing the task state S_t. Both the training and testing environments are fully randomized w.r.t.dynamics properties including mass, friction, and restitution.

Methods	W.M.	D.A.	FiLM	Seen	Unseen
DAgger	✘	✘	✘	59.9	57.5
World Model	✓	✘	✘	61.6	59.4
RMA [6]	✘	✓	✘	65.6	57.9
Ours w/o W.M.	✘	✓	✓	70.0	63.7
Ours w/o FiLM	✓	✓	✘	73.3	59.4
Ours	✓	✓	✓	**82.2**	**75.0**

TABLE II: Ablation study on the most challenging evaluation track, i.e., unknown state with single-view observation. W.M. means World Model and D.A. means Dynamics Adaptation.

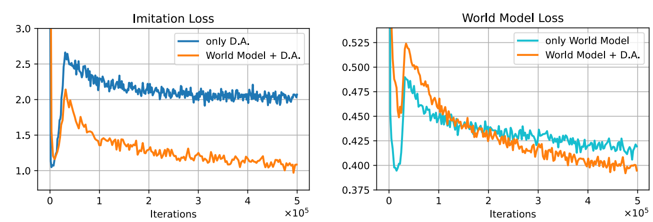

Fig. 3: **Loss curves during the distillation process.** We adopt DAgger which starts with teacher action for execution and gradually adds the weights of student action so that the initial loss declines rapidly. **Left:** Comparison of imitation loss between using only Dynamics Adaptation and incorporating the World Model. **Right:** Comparison of World Model loss between using only the World Model and integrating Dynamics Adaptation.

Methods	Normal							Slippery	Non-uniform Mass		Avg.
	Mug	Bulldozer	Card	Book	Dinosaur	Chips Can	Switch	YCB-Bottle	Half-full Bottle	Coffee jar	
CORN w tracking	1/5	3/5	**4/5**	**4/5**	2/5	0/5	2/5	0/5	0/5	2/5	18/50 (36%)
Ours	**3/5**	**4/5**	**4/5**	**4/5**	**3/5**	**2/5**	**4/5**	**3/5**	**4/5**	**3/5**	**34/50 (68%)**

TABLE III: Quantitative results in the real world.

Methods	μ_1		μ_2		μ_3		μ_4	
	S.R. ↑	Avg. Time ↓	S.R. ↑	Avg. Time ↓	S.R. ↑	Avg. Time ↓	S.R. ↑	Avg. Time ↓
Ours w/o D.A.	3/5	**65 s**	3/5	**81 s**	**4/5**	**96 s**	3/5	**124 s**
Ours	**4/5**	**45 s**	**4/5**	**50 s**	**4/5**	**49 s**	**4/5**	**51 s**

TABLE IV: Experiments on different surface frition, with progressive friction levels, $\mu_1 < \mu_2 < \mu_3 < \mu_4$.

a) Task Setup.: At the beginning of each episode, we randomly place the object in a stable pose on the table. The robot arm is then initialized at a joint configuration uniformly sampled within predefined joint bounds, positioned slightly above the workspace to prevent unintended collisions with the table or object. Next, we sample a random 6D stable goal pose on the table, ensuring it is at least $0.1\,\mathrm{m}$ away from the initial pose to prevent immediate success upon initialization. To guarantee valid initial and goal poses for each object, we precompute a set of stable poses, as detailed in the supplementary. An episode is considered successful if the object's final pose is within $0.05\,\mathrm{m}$ and 0.1 radians of the target pose.

b) Baselines.: We evaluate our approach against two state-of-the-art baselines: HACMan and CORN, which represent primitive-based and closed-loop methods, respectively. Since HACMan was originally implemented in the MuJoCo simulator, we re-implemented it within our benchmark for a fair comparison. CORN shares the same simulation environment as our method, allowing us to train and evaluate it directly with minimal modifications. To ensure a fair comparison, we further enhanced CORN by replacing its shallow MLP-based point cloud encoder with the same vision backbone as ours. Additionally, for settings where the current object pose is unknown, we provided all methods with the same goal point cloud representation to maintain consistency.

c) Results.: As shown in Table I, our method consistently outperforms all baselines across all three evaluation tracks. In particular, we achieve a significant performance gain over previous approaches, with at least a **31.5%** improvement in success rate. Notably, the performance gap is most pronounced in challenging scenarios involving unknown states and single-view observations, where our method's dynamics modeling capability plays a crucial role. Compared to HACMan, our approach benefits from its closed-loop execution and variable impedance control, enabling more robust dexterous manipulation. While HACMan relies on pre-defined motion primitives, its adaptability to complex geometries and variations in physics are limited. Moveover, our method surpasses CORN due to our adaptation mechanism refines the world model based on historical trajectories, allowing the policy to adjust effectively to variations in object properties such as mass, friction, and scale. These results highlight the effectiveness of our strong generalization capabilities in diverse rearrangement tasks.

B. Ablation Study

We conduct ablation studies on the most challenging evaluation track, i.e., unknown state with single-view observation. Our goal is to systematically analyze the contribution of each key module to the overall performance.

a) Synergy between Next State Prediction and Action Learning.: To analyze the optimization process, we visualize the loss curve during training and compare the approach that uses only dynamics adaptation (i.e., RMA) with that adding World Modeling. Our results show that during the distillation, simultaneous learning of the next state improves action coverage, confirming the synergy between world modeling and action learning. Additionally, we discuss the integration of the world model in the RL teacher policy, which is elaborated in the supplementary material.

b) Indivisibility of Dynamics Adaptation and World Modeling.: We investigate the individual and combined effects of dynamics adaptation and world modeling. Our results (Table II) show that using only the world model or dynamics adaptation, i.e.RMA, provides only marginal improvements over the naive DAgger baseline, with success rates increasing by just 1.7% and 5.7%, respectively. However, when both modules are used together, the performance jumps significantly from 59.9% to 73.3%. This improvement can be attributed to the complementary nature of these components. Without dynamics adaptation, the world model lacks sufficient information to reason about the dynamic effects of interaction. Conversely, using only dynamics adaptation also provides limited benefits due to the absence of a sufficiently structured learning target. These findings highlight the indivisibility of world modeling and dynamics adaptation, demonstrating that their combination is a non-trivial yet highly effective design choice.

C. Real-World Experiments

a) Real-World Setup: Our experimental setup is illustrated in the supplementary. We use a Franka robot arm for action execution and a RealSense D435 camera positioned

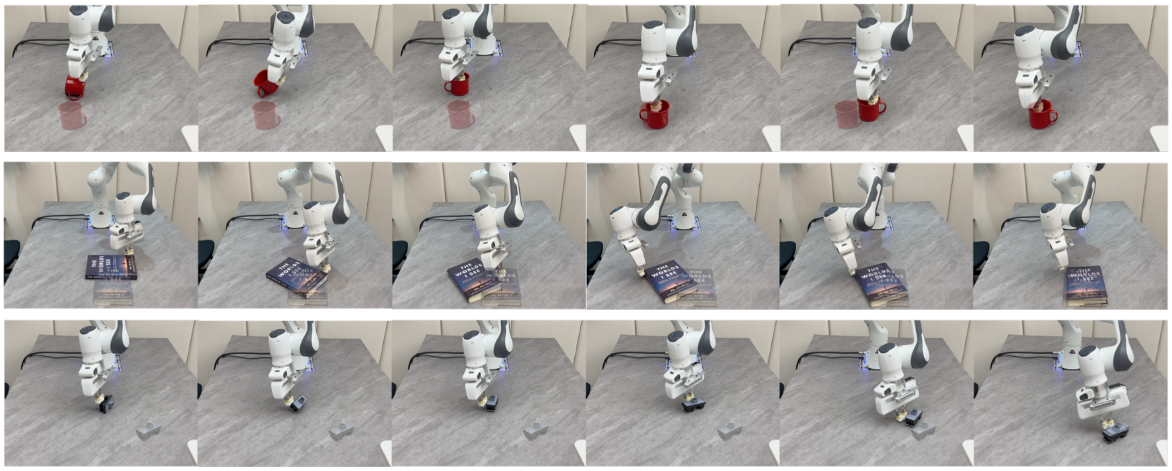

Fig. 4: Qualatative Results in the real world. The goal pose is shown transparently.

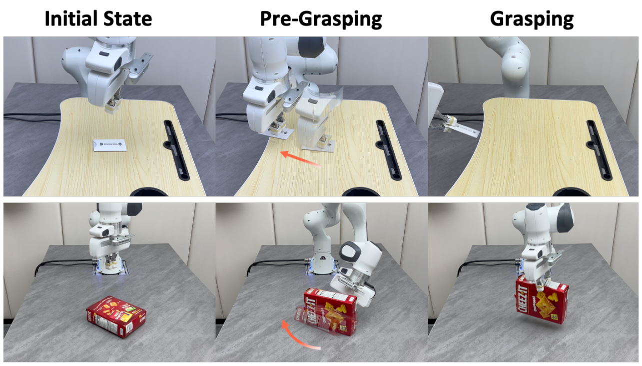

Fig. 5: Our policy helps grasping a thin card and broad cracker box.

at a side view to capture RGB-D images. We evaluate our approach on 10 unseen real-world objects, including both slippery objects and those with non-uniform mass distribution such as a half-filled bottle. Before each episode, we first place the object at the target goal pose and record its point cloud. Then, we reposition the object in a random stable pose and allow our policy to execute the manipulation task. Upon completion, we use Iterative Closest Point (ICP) to measure the pose error between the final object position and the recorded target pose. For symmetric objects where direct ICP alignment is ambiguous, we relax the success criteria along the symmetric axes and compute errors only in translation and relevant rotational components.

b) Generalization across Diverse Objects.: We evaluate our model's generalization ability by comparing it with CORN, which relies on an external tracking module for object pose estimation in real-world experiments. As shown in Figure 4 and Table III, our method achieves accurate manipulation across diverse objects without external pose tracking, significantly outperforming CORN with an average success rate of 68% versus 36%. CORN struggles with precise execution due to occlusions in single-view partial point clouds and inaccuracies in real-world pose estimation. Additionally, our model demonstrates robust performance on slippery objects and those with non-uniform mass, where CORN fails. We validate the generalization ability of our model and compare our method against CORN, which depends on an external tracking module to estimate object poses in real-world experiments.

c) Robustness to Surface Friction Variations.: To assess the effectiveness of dynamics adaptation, we conduct experiments on surfaces with varying friction coefficients. We select four tablecloths (Figure 1) with progressive friction levels, i.e.$\mu_1, \mu_2, \mu_3, \mu_4$ and use the bulldozer toy as the test object. Additionally, we report the average execution time for successful episodes. As shown in Table IV, the model without dynamics adaptation exhibits significant performance degradation when interacting with surfaces of different friction levels, leading to erratic execution times. In contrast, our policy with dynamics adaptation maintains consistent success rates while ensuring stable execution times across all surface conditions. This highlights the robustness of our approach in handling diverse real-world contact dynamics.

D. Applications

By leveraging a VLM-specified goal pose and applying our non-prehensile manipulation as a pre-grasping step, we can reorient these objects into grasp-friendly configurations, significantly improving grasp success rate.

IV. Conclusion, Limitations, and Future Works

In this work, we present a novel policy learning approach that jointly predicts future states while adapting dynamics from historical trajectories. Our model enhances generalizable non-prehensile manipulation by reducing reliance on multi-camera setups and pose tracking modules while maintaining robustness across diverse physical conditions. Extensive simulation and real-world experiments validate the effectiveness of our approach. However, our method also has certain limitations since it relies solely on point clouds as the visual input modality. It struggles with symmetric objects due to geometric ambiguity, and faces challenges with transparent and specular objects, where raw depth is imcomplete. A promising direction is to incorporate additional appearance information to provide richer visual cues.

References

[1] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[2] Samuel R Buss. Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods. *IEEE Journal of Robotics and Automation*, 17(1-19):16, 2004.

[3] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[4] Yoonyoung Cho, Junhyek Han, Yoontae Cho, and Beomjoon Kim. Corn: Contact-based object representation for nonprehensile manipulation of general unseen objects. In *12th International Conference on Learning Representations, ICLR 2024*. International Conference on Learning Representations, ICLR, 2024.

[5] Juan Del Aguila Ferrandis, Joao Pousa De Moura, and Sethu Vijayakumar. Learning visuotactile estimation and control for non-prehensile manipulation under occlusions. In *The 8th Conference on Robot Learning*, pages 1–15, 2024.

[6] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. *Robotics: Science and Systems XVII*, 2021.

[7] Jake Levinson, Carlos Esteves, Kefan Chen, Noah Snavely, Angjoo Kanazawa, Afshin Rostamizadeh, and Ameesh Makadia. An analysis of svd for deep rotation estimation. *Advances in Neural Information Processing Systems*, 33:22554–22565, 2020.

[8] Yixin Lin, Austin S. Wang, Giovanni Sutanto, Akshara Rai, and Franziska Meier. Polymetis. https://facebookresearch.github.io/fairo/polymetis/, 2021.

[9] Jiangran Lyu, Yuxing Chen, Tao Du, Feng Zhu, Huiquan Liu, Yizhou Wang, and He Wang. Scissorbot: Learning generalizable scissor skill for paper cutting via simulation, imitation, and sim2real. In *8th Annual Conference on Robot Learning*.

[10] Igor Mordatch, Zoran Popović, and Emanuel Todorov. Contact-invariant optimization for hand manipulation. In *Proceedings of the ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 137–144, 2012.

[11] João Moura, Theodoros Stouraitis, and Sethu Vijayakumar. Non-prehensile planar manipulation via trajectory optimization with complementarity constraints. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 970–976. IEEE, 2022.

[12] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[13] Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014.

[14] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[15] Ruicheng Wang, Jialiang Zhang, Jiayi Chen, Yinzhen Xu, Puhao Li, Tengyu Liu, and He Wang. Dexgraspnet: A large-scale robotic dexterous grasp dataset for general objects based on simulation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11359–11366. IEEE, 2023.

[16] William Yang and Michael Posa. Dynamic on-palm manipulation via controlled sliding. *arXiv preprint arXiv:2405.08731*, 2024.

[17] Xiang Zhang, Siddarth Jain, Baichuan Huang, Masayoshi Tomizuka, and Diego Romeres. Learning generalizable pivoting skills. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5865–5871. IEEE, 2023.

[18] Wenxuan Zhou, Bowen Jiang, Fan Yang, Chris Paxton, and David Held. Hacman: Learning hybrid actor-critic maps for 6d non-prehensile manipulation. In *Conference on Robot Learning*, pages 241–265. PMLR, 2023.