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]Center on Frontiers of Computing Studies, School of Computer Science, Peking University [2]Galbot
https://pku-epic.github.io/DyWA/

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.
Keywords: World Action Model, Non-prehensile Manipulation

1 Introduction

Non-prehensile manipulation—such as pushing, sliding, and toppling—extends robotic capabilities beyond traditional grasping, enabling task execution under geometric, clutter, or workspace constraints. While planning-based methods [1, 2, 3, 4] have shown success, they rely on precise object properties (e.g., mass, friction, CAD models), which are rarely available in the real world.

Recent learning-based approaches [5] shift toward end-to-end policy learning from visual input, demonstrating stronger generalization. For instance, HACMan [6] and CORN [7] exploit vision-based RL or distillation to acquire contact-rich skills. However, these methods remain geometry-centric and shows poor robustness under dynamic variations such as friction or mass changes.

To achieve generalization across dynamic variations, we argue that contact-rich manipulation fundamentally requires world modeling: policies must not only output actions but also internalize how interactions shape future states. Under this lens, existing teacher-student distillation frameworks fall short—while the privileged RL teacher can exploit full dynamics, the student policy trained from partial observations suffers due to (1) single-view partial point cloud observation, (2) Markovian policies collapsing over multiple dynamics, and (3) weak supervision limited to action imitation.

To address this, we introduce Dynamics-adaptive World Action Model (DyWA). DyWA explicitly integrates world modeling into action learning:(i) a dynamics adaptation module infers latent physical properties from observation-action history, (ii) action prediction is reformulated as joint

†: Corresponding authors

9th Conference on Robot Learning (CoRL 2025), Seoul, Korea.

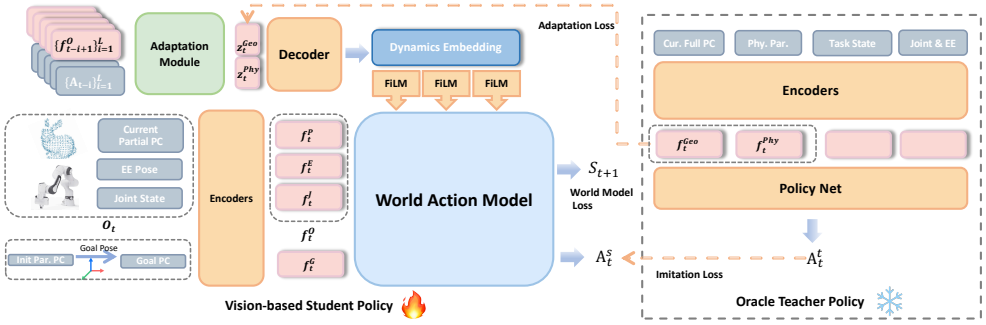

Figure 1: Our World Action Model processes the embeddings of the current observation (partial point cloud, end-effector pose, and joint state) and the goal point cloud (transformed from the initial partial observation) to predict the robot action and next state. Additionally, an adaptation module encodes 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.

prediction of actions and future states, providing richer supervision, and (iii) Feature-wise Linear Modulation (FiLM) bridges inferred dynamics with policy learning.

We benchmark DyWA against strong baselines on CORN, varying camera and tracking settings. DyWA improves success by 31.5% in simulation, and in real-world tests achieves 68% success across diverse geometries, frictional conditions, and mass distributions (e.g., half-filled bottles). We further demonstrate DyWA's compatibility with VLMs for challenging thin/wide-object scenarios.

2 Method

2.1 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 $\mathbf{G}$ 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. Observations include the partial point cloud P_t, joint states J_t, and end-effector pose E_t.

2.2 Pipeline Overview

Our training pipeline follows a standard teacher-student distillation framework. Due to the difficulty of obtaining high-quality demonstrations for our task, we first train a state-based RL policy with additional privileged information—i.e., the full object point cloud, task state, and physical parameters—as the teacher policy. For consistency, we adopt the same reward design as CORN, as elaborated in the supplementary material. To obtain a vision-based policy suitable for real-world deployment, we introduce our Dynamics-adaptive World Action Model, which serves as the student policy distilled from the teacher policy. Unlike the teacher, our student model relies solely on limited observations that are feasible to obtain in real-world settings.

In the following sections, we detail the design of the world action model (Sec. 2.3) and the dynamics adaptation mechanism (Sec. 2.4). To enable adaptive force interaction in this contact-rich manipulation, we further incorporate a variable impedance controller (Sec. 2.5). Once trained (Sec. 2.6), our model can be transferred from simulation to the real world in a zero-shot manner, without requiring real-world fine-tuning.

2.3 World Action Model

Definition. A *world action model* refers to a policy model that jointly predicts actions and forecasts the corresponding future states. Although the current action is not provided as an explicit input, the model exhibits world model characteristics by implicitly conditioning on the current policy action prediction.

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++ [8] to obtain $\mathbf{f}_t^P$. The architectural details are provided in the supplementary material. For robot proprioception, we separately encode joint positions and velocities ($\mathbf{f}_t^J$) and the end-effector pose ($\mathbf{f}_t^E$) 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 $P_G = \mathbf{G}P_0$. This goal point cloud is then encoded using the shared network with the observation point cloud encoder.

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 $\mathbf{A}_t$ 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 [9, 10], and define the world model loss as:

$$\mathcal{L}_{\text{world}} = \|\mathbf{T}_{t+1} - \hat{\mathbf{T}}_{t+1}\|_2^2 + \|\mathbf{R}_{t+1} - \hat{\mathbf{R}}_{t+1}\|_1 \tag{1}$$

where $\mathbf{T}_{t+1} \in \mathbb{R}^3$ and $\mathbf{R}_{t+1} \in SO(3)$ are the predicted translation and rotation, while $\hat{\mathbf{T}}_{t+1} \in \mathbb{R}^3$ 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:

$$\mathcal{L}_{\text{imitation}} = \|\mathbf{A}_t^s - \mathbf{A}_t^t\|^2 \tag{2}$$

2.4 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.

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 $f_t^O = \{f_t^P, f_t^J, f_t^E\}$ with the previous action embedding f_{t-1}^A, 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:

$$\mathbf{z}_t = \text{Embed}\left(\left[\text{concat}(\mathbf{f}_{t-i-1}^O, \mathbf{f}_{t-i-2}^A)\right]_{i=1}^L\right) \tag{3}$$

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.

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

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 [11] 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 [12, 13]. 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.

2.5 Action Space with Variable Impedance

To enable adaptive force interaction between the robot and object, we employ variable impedance control [7] 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 [14]:

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

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 [15] for implementation.

2.6 Training Protocol

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

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

We begin by training the teacher policy for 200K iterations in simulation using PPO. Subsequently, we employ DAgger [16] 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.

3 Experiments

3.1 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 [7, 6] 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 [18] for training. Additionally, we enrich the task setting by introducing an unseen object test set,

Methods	Action Type	Known State (3 view)		Unknown State (3 view)		Unknown State (1 view)	
		Seen	Unseen	Seen	Unseen	Seen	Unseen
HACMan [6]	Primitive	3.8(42.2)	5.7(39.4)	3.0(23.6)	4.1(26.5)	1.5(17.9)	2.9(18.3)
CORN [7]	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 1: 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.

Methods	W.M.	D.A.	FiLM	Seen	Unseen
DAgger [16]	✘	✘	✘	59.9	57.5
World Model	✓	✘	✘	61.6	59.4
RMA [17]	✘	✓	✘	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 2: 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.

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.

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.

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. However, because it requires strict per-point correspondence as input, its success rate is extremely low in the unknown state setting. 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.

Results. As shown in Table 1, 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.

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 3: Quantitative results in the real world. Each cell shows the number of successful trials out of 5 attempts. Our method consistently achieves high success rates across diverse objects.

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

Compared to HACMan, our approach benefits from its closed-loop execution and variable impedance control, enabling more robust dexterous manipulation. While HACMan relies on predefined 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.

3.2 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.

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 loss convergence, 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.

On the Complementarity of Dynamics Adaptation and World Modeling. We investigate the individual and combined effects of dynamics adaptation and world modeling. Our results (Table 2) 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 complementarity of world modeling and dynamics adaptation, demonstrating that their combination is a non-trivial yet highly effective design choice.

Effectiveness of FiLM Conditioning. We further evaluate the role of Feature-wise Linear Modulation (FiLM) in bridging adaptation embeddings and the world action model. Our results indicate that FiLM provides a more effective and structured conditioning mechanism than direct input concatenation. Specifically, incorporating FiLM into RMA boosts performance from 65.6% to 70.0%. More notably, when all three modules (world modeling, dynamics adaptation, and FiLM) are used together, the success rate reaches 82.2%, with FiLM contributing an additional 8.9% improvement.

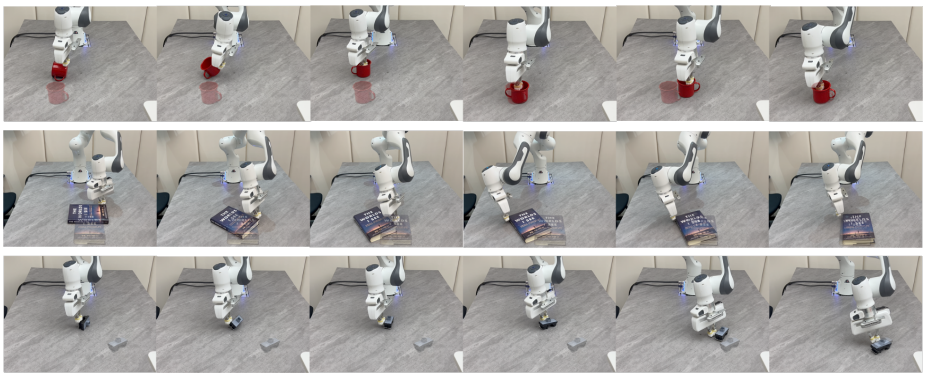

Figure 2: Qualatative Results in the real world. The goal pose is shown transparently.

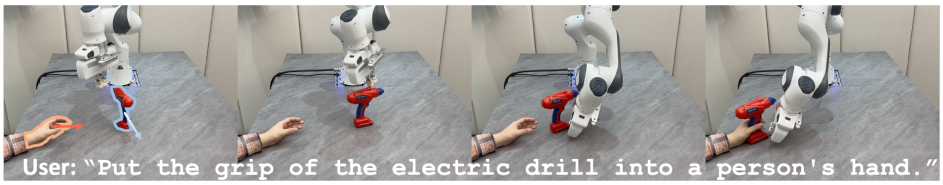

Figure 3: By integrating with Vision-Language Models (VLMs), our goal-conditioned policy can be executed based on natural language instructions.

We also discuss different methods for conditioning in the supplementary whose conclusion consists with our claims. This reinforces FiLM as a lightweight and effective choice for integrating adaptation embeddings.

3.3 Real-World Experiments

To evaluate the real-world applicability of our method, we conduct experiments on a physical robot setup. Our goal is to validate the zero-shot transferability of our policy from simulation to the real world and compare its performance against prior methods.

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 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.

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 2 and Table 3, 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.

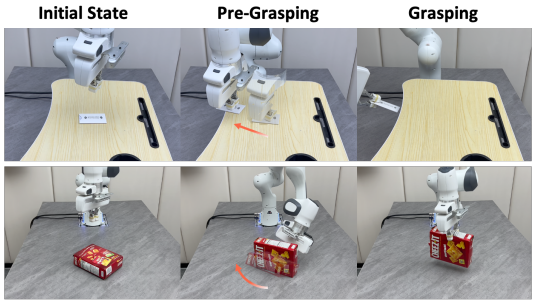

Figure 4: Our policy helps grasping a thin card and broad cracker box.

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 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 4, 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.

3.4 Applications

We present a practical manipulation system that integrates Vision-Language Models (VLMs), our non-prehensile policy, and a grasping model [19]. By leveraging VLMs, our goal-conditioned policy can be executed based on natural language instructions. Specifically, we utilize SoFar [20], a model capable of generating semantic object poses from language commands, to specify goals for our policy. As shown in Figure 3, given the command *"Put the grip of the electric drill into a person's hand"*, SoFar generates the target transformation of the drill (e.g., rotation $\Delta\theta = 122^\circ$ and translation $\Delta x, \Delta y = [0.54, 0.09]$), which is then used as the goal for our policy. This enables natural, instruction-driven object handovers, highlighting the potential of our approach in human-robot interaction.

Additionally, we showcase the system outperforms or complements traditional prehensile manipulation. As illustrated in the third row of Figure 2, a standard pick-and-place strategy struggles to flip a tiny switch due to gripper-table collisions, whereas our policy enables efficient rearrangement in a single continuous motion. Furthermore, our policy serves as an effective pre-grasping step in the system. As shown in Figure 4, certain objects are inherently difficult to grasp due to their geometry—for example, a thin card lying flat on a surface or a broad cracker box exceeding the gripper's maximum span. Our system can firstly reorient these objects into grasp-friendly configurations, significantly improving grasp success rate.

4 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 incomplete. A promising direction is to incorporate additional appearance information [21, 22, 23, 24, 25, 26, 27] to provide richer visual cues.

Acknowledgments

We thank Yixin Zheng for organizing the code release, and Junhao Yang for assistance with rendering. We also appreciate the valuable suggestions and discussions from Jiayi Chen, Jiazhao Zhang, Mi Yan and Shenyuan Gao. This work was supported in part by National Science and Technology Major Project (2022ZD0114904) & NSFC-6247070125.

References

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

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

[3] J. Moura, T. Stouraitis, and S. 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.

[4] W. Yang and M. Posa. Dynamic on-palm manipulation via controlled sliding. *arXiv preprint arXiv:2405.08731*, 2024.

[5] X. Zhang, S. Jain, B. Huang, M. Tomizuka, and D. Romeres. Learning generalizable pivoting skills. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5865–5871. IEEE, 2023.

[6] W. Zhou, B. Jiang, F. Yang, C. Paxton, and D. Held. Hacman: Learning hybrid actor-critic maps for 6d non-prehensile manipulation. In *Conference on Robot Learning*, pages 241–265. PMLR, 2023.

[7] Y. Cho, J. Han, Y. Cho, and B. 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.

[8] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[9] J. Levinson, C. Esteves, K. Chen, N. Snavely, A. Kanazawa, A. Rostamizadeh, and A. Makadia. An analysis of svd for deep rotation estimation. *Advances in Neural Information Processing Systems*, 33:22554–22565, 2020.

[10] J. Lyu, Y. Chen, T. Du, F. Zhu, H. Liu, Y. Wang, and H. Wang. Scissorbot: Learning generalizable scissor skill for paper cutting via simulation, imitation, and sim2real. In *8th Annual Conference on Robot Learning*.

[11] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

[12] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[13] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[14] S. 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.

[15] Y. Lin, A. S. Wang, G. Sutanto, A. Rai, and F. Meier. Polymetis. https://facebookresearch.github.io/fairo/polymetis/, 2021.

[16] S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

[17] A. Kumar, Z. Fu, D. Pathak, and J. Malik. Rma: Rapid motor adaptation for legged robots. *Robotics: Science and Systems XVII*, 2021.

[18] R. Wang, J. Zhang, J. Chen, Y. Xu, P. Li, T. Liu, and H. 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.

[19] J. Shi, Y. Jin, D. Li, H. Niu, Z. Jin, H. Wang, et al. Asgrasp: Generalizable transparent object reconstruction and grasping from rgb-d active stereo camera. *arXiv preprint arXiv:2405.05648*, 2024.

[20] Z. Qi, W. Zhang, Y. Ding, R. Dong, X. Yu, J. Li, L. Xu, B. Li, X. He, G. Fan, et al. So-far: Language-grounded orientation bridges spatial reasoning and object manipulation. *arXiv preprint arXiv:2502.13143*, 2025.

[21] Y. Ma, H. Liu, H. Wang, H. Pan, Y. He, J. Yuan, A. Zeng, C. Cai, H.-Y. Shum, W. Liu, et al. Follow-your-emoji: Fine-controllable and expressive freestyle portrait animation. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–12, 2024.

[22] Y. Ma, Y. He, H. Wang, A. Wang, L. Shen, C. Qi, J. Ying, C. Cai, Z. Li, H.-Y. Shum, et al. Follow-your-click: Open-domain regional image animation via motion prompts. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 6018–6026, 2025.

[23] Y. Ma, K. Feng, X. Zhang, H. Liu, D. J. Zhang, J. Xing, Y. Zhang, A. Yang, Z. Wang, and Q. Chen. Follow-your-creation: Empowering 4d creation through video inpainting. *arXiv preprint arXiv:2506.04590*, 2025.

[24] Y. Ma, Y. Liu, Q. Zhu, A. Yang, K. Feng, X. Zhang, Z. Li, S. Han, C. Qi, and Q. Chen. Follow-your-motion: Video motion transfer via efficient spatial-temporal decoupled finetuning. *arXiv preprint arXiv:2506.05207*, 2025.

[25] Y. Ma, X. Cun, Y. He, C. Qi, X. Wang, Y. Shan, X. Li, and Q. Chen. Magicstick: Controllable video editing via control handle transformations. *arXiv preprint arXiv:2312.03047*, 2023.

[26] Y. Ma, Y. He, X. Cun, X. Wang, S. Chen, X. Li, and Q. Chen. Follow your pose: Pose-guided text-to-video generation using pose-free videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 4117–4125, 2024.

[27] Y. Ma, Y. Wang, Y. Wu, Z. Lyu, S. Chen, X. Li, and Y. Qiao. Visual knowledge graph for human action reasoning in videos. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 4132–4141, 2022.