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

Anonymous Authors

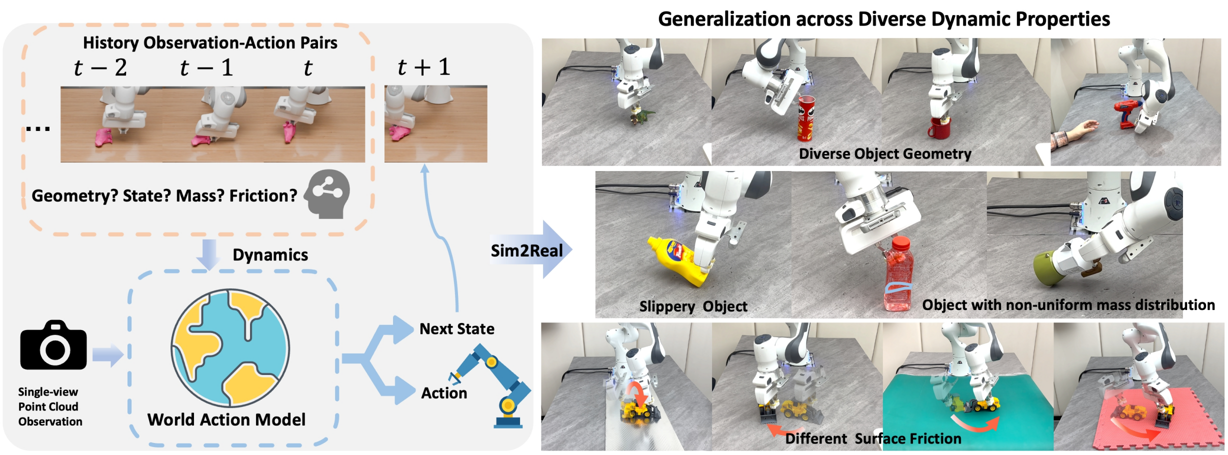

Fig. 1: **Illustration of the high-level idea and generalization ability of DyWA.** Given a target object's 6D pose and *single-view* object point cloud, our non-prehensile manipulation policy aims to rearrange the object without grasping. **Left**: Our key insight is to enhance action learning by jointly 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, and toppling—extends robotic capabilities beyond traditional grasping, enabling task execution under geometric, clutter, or workspace constraints. While planning-based methods [8, 11, 9, 14] have shown success, they require precise knowledge of object properties (e.g., mass, friction, CAD models), limiting real-world applicability.

Learning-based approaches [15] offer improved generalization by training in simulation and deploying policies zero-shot. For instance, HACMan [16] and CORN [2] use vision-based RL or distillation frameworks to learn contact-rich actions. However, these methods face two key limitations: (1) heavy reliance on multi-view cameras and accurate pose tracking [3], and (2) poor generalization across dynamic conditions due to geometry-centric modeling.

We argue that a generalizable non-prehensile manipulation policy should work with a single camera and adapt to varying physical properties without requiring tracking modules. To this end, we revisit the teacher-student framework under this constrained setting. While a privileged RL teacher achieves strong performance, the student policy, based on partial observations, suffers significantly due to: (1) partial observability omitting critical geometry, (2) Markovian policies averaging over diverse dynamics, and (3) limited supervision during distillation.

To tackle these issues, we propose **DyWA (Dynamics-Adaptive World Action Model)**. First, a dynamics adaptation module encodes historical observation-action pairs, inspired by RMA [4], to infer latent physical properties. Second, we reformulate action prediction as joint prediction of actions and

†: Corresponding authors

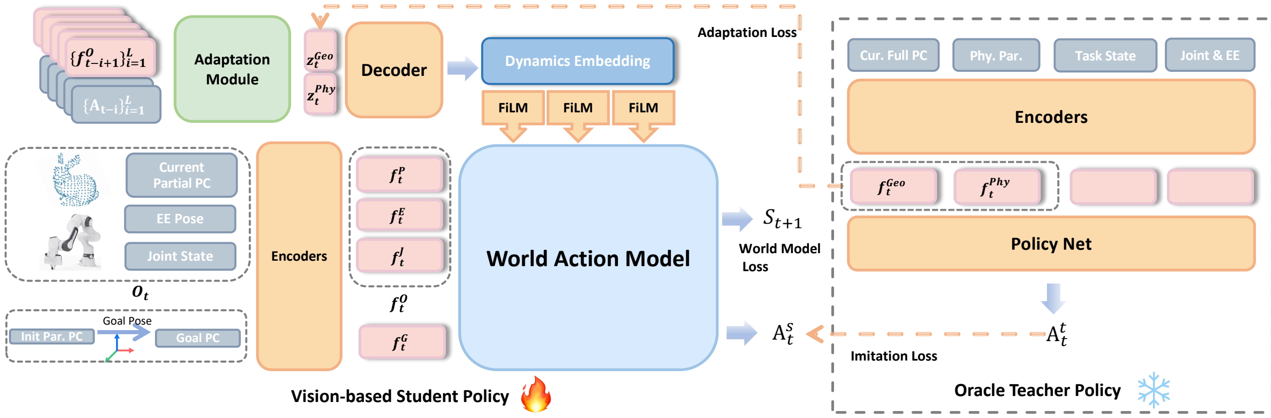

Fig. 2: Our World Action Model processes the embeddings of the current observation (partial point cloud, end-effector pose, and joint state) and the goal point cloud (transformed from the initial partial observation) to predict the robot action and next state. Additionally, an adaptation module encodes 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.

future states, introducing a world action model with richer supervision. Finally, we apply Feature-wise Linear Modulation (FiLM) to bridge dynamics and policy learning effectively.

We build a comprehensive benchmark based on CORN, varying camera views and tracking settings. DyWA outperforms baselines by 31.5% in simulation. In real-world settings, DyWA achieves 68% success across diverse geometries, varying friction, and non-uniform mass distributions (e.g., half-filled bottles). We also demonstrate applications combining DyWA with VLMs for thin/wide object manipulation.

II. Method

A. Task Formulation

Following HACMan and CORN, we consider 6D object rearrangement via non-prehensile actions (e.g., pushing, flipping), aiming to move an object from its initial pose to a goal pose G on the table. We define G as a 6DoF transformation relative to the initial stable pose. The task state S_t at time t is the relative transformation between the current and goal pose. Observations include the partial point cloud P_t, joint states J_t, and end-effector pose E_t.

B. World Action Model

a) Observation and Goal Encoding.: We use separate encoders for each modality. The point cloud is processed by a simplified PointNet++ [12], yielding f_t^P. Joint states f_t^J and end-effector pose f_t^E are encoded via MLPs. The goal point cloud P_G is constructed as GP_0 and encoded by the same network as P_t.

b) State-based World Modeling.: We jointly predict action A_t and next task state S_{t+1} via MLPs. The model learns from both the teacher policy and simulated rollouts. We represent the task state with translation $T_{t+1} \in \mathbb{R}^3$ and 9D rotation $\mathbf{R}_{t+1} \in SO(3)$ [5, 7], with world model loss:

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

The imitation loss is:

$$\mathcal{L}_{imitation} = |A_t^s - A_t^t|^2 \tag{2}$$

C. Dynamics Adaptation

We extract dynamics information from history to enhance adaptability.

a) Adaptation Embedding.: At each timestep, we concatenate observation embeddings $f_t^O = f_t^P, f_t^J, f_t^E$ with the previous action embedding f_{t-1}^A. A sequence of L such tuples is passed through a 1D CNN to obtain the adaptation embedding z_t:

$$z_t = Embed(\{\text{concat}(f_{t-i-1}^O, f_{t-i-2}^A)\}_{i=1}^L) \tag{3}$$

Supervision is provided by matching z_t to the teacher-provided full geometry and physics embedding:

$$\mathcal{L}_{adapt} = |z_t^{Geo,Phy} - \text{concat}(f_t^{Geo}, f_t^{Phy})|^2 \tag{4}$$

b) Dynamics Conditioning.: We decode z_t into a dynamics embedding and apply it via FiLM [10] blocks:

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

FiLM blocks are densely inserted in early layers of the world model, enabling adaptive modulation.

D. Action Space with Variable Impedance

We adopt variable impedance control to enable compliant interaction. The action includes a subgoal residual $\Delta T_{ee} \in SE(3)$ and joint-space impedance parameters: gains $P \in \mathbb{R}^7$, damping factors $\rho \in \mathbb{R}^7$, with $D = \rho\sqrt{P}$. Desired joint position is computed via damped least-squares inverse kinematics [1]:

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

We implement impedance control using the Polymetis API [6].

Methods	Action Type	Known State (3 view)		Unknown State (3 view)		Unknown State (1 view)	
		Seen	Unseen	Seen	Unseen	Seen	Unseen
HACMan [16]	Primitive	3.8(42.2)	5.7(39.4)	3.0(23.6)	4.1(26.5)	2.1(19.2)	
CORN [2]	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.

Methods	W.M.	D.A.	FiLM	Seen	Unseen
DAgger	✗	✗	✗	59.9	57.5
World Model	✓	✗	✗	61.6	59.4
RMA [4]	✗	✓	✗	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.

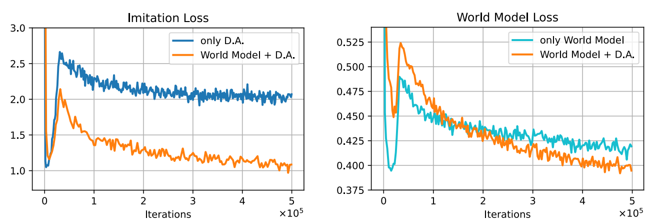

Fig. 3: **Loss curves during the distillation process.** We adopt DAgger which starts with teacher action for execution and gradually adds the weights of student action so that the initial loss declines rapidly. **Left:** Comparison of imitation loss between using only Dynamics Adaptation and incorporating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

E. Training Protocol

The overall objective is:

$$\mathcal{L} = \mathcal{L}_{imitation} + \mathcal{L}_{world} + \mathcal{L}_{adapt} \quad (7)$$

The teacher policy is trained via PPO for 200K steps. The student is trained with DAgger for 500K steps. Domain randomization is applied on mass, scale, friction, and restitution. To improve sim-to-real transfer, we inject noise into torques, point clouds, and goal poses during student training.

III. Experiments

A. Benchmarking Tabletop Non-prehensile Rearrangement in Simulation

We benchmark our method and baselines in a unified IsaacGym-based simulation to ensure fair comparison. While prior works [2, 16] developed their own simulators, a standard benchmark is still lacking. We establish one based on CORN, using 323 DexGraspNet [13] objects for training and 50 unseen test objects (10 shapes × 5 scales). We vary two perception conditions: (i) single vs. multi-view observations and (ii) known vs. unknown object poses. Dynamics properties (mass, friction, restitution) are randomized during both training and testing.

a) Task Setup.: Each episode starts with the object in a random stable pose and the robot initialized above the workspace. A goal pose at least $0.1\,\mathrm{m}$ away is sampled. Stable poses are precomputed. Success is defined as final pose within $0.05\,\mathrm{m}$ and 0.1 radians of the goal.

b) Baselines.: We compare with HACMan (primitive-based) and CORN (closed-loop). HACMan is re-implemented in our setup. CORN is adapted with our vision encoder for fairness. When object pose is unknown, all methods receive the same goal point cloud.

c) Results.: As shown in Table I, our method outperforms all baselines, with at least **31.5%** higher success rate. Gains are largest in the hardest settings (single-view, unknown state), where dynamics modeling is crucial. Compared to HACMan, our closed-loop execution and variable impedance control provide greater robustness. CORN, despite sharing architecture, lacks our adaptation mechanism that refines dynamics understanding over time.

B. Ablation Study

We conduct ablations on the hardest track (unknown state, single-view) to assess module contributions.

a) Synergy between Next State Prediction and Action Learning.: Training curves show that combining world modeling with dynamics adaptation improves learning efficiency and action coverage. This synergy is further explored in the supplementary.

b) Indivisibility of Dynamics Adaptation and World Modeling.: Table II shows that using only dynamics adaptation or only world modeling yields marginal gains (1.7%, 5.7%). Combining both boosts performance from 59.9% to 73.3%, validating their complementarity. Dynamics adaptation provides context for modeling, while world modeling gives structure for learning—together, they enable effective generalization.

C. Real-World Experiments

a) Setup.: We use a Franka arm and a side-mounted RealSense D435 camera. The policy is evaluated on 10 un-

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, $\mu_1 < \mu_2 < \mu_3 < \mu_4$.

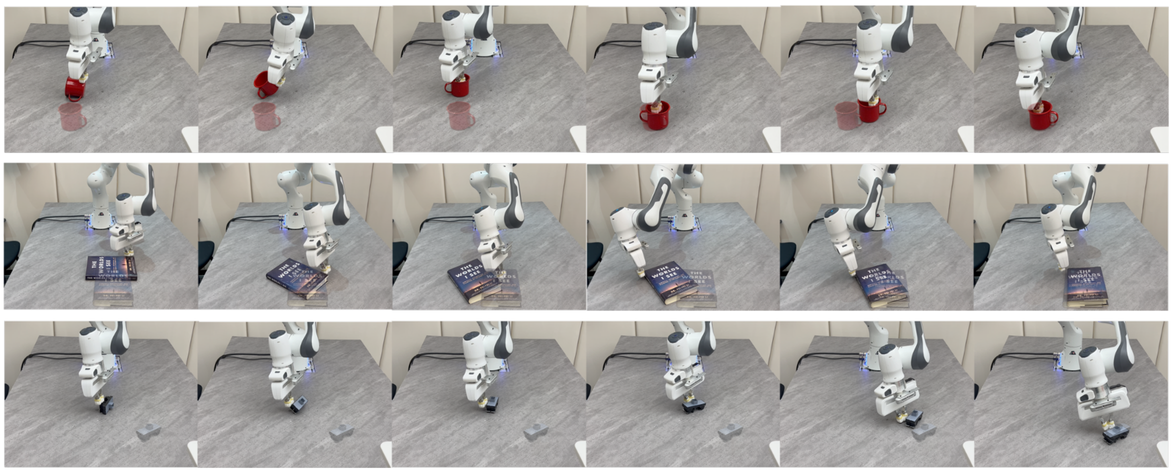

Fig. 4: Qualatative Results in the real world. The goal pose is shown transparently.

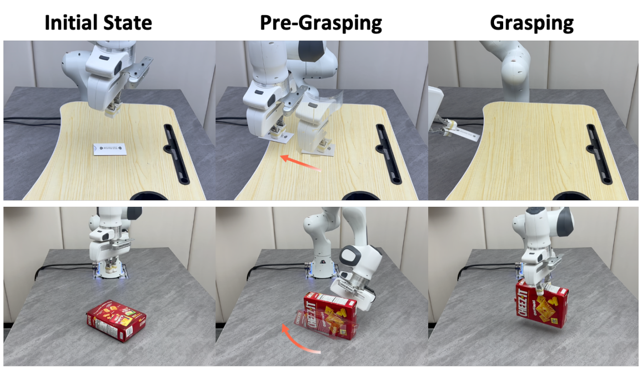

Fig. 5: Our policy helps grasping a thin card and broad cracker box.

seen real-world objects, including slippery and asymmetrically weighted ones (e.g., half-filled bottle). For each trial, we record the goal point cloud by first placing the object in the desired pose. The object is then repositioned randomly, and the policy executes the task. We use ICP to compute final pose error, relaxing alignment criteria for symmetric objects by ignoring irrelevant rotation axes.

b) Generalization.: Compared to CORN, which relies on external pose tracking, our method achieves significantly higher success (68% vs. 36%) without external perception (Table III). CORN suffers from occlusions and tracking errors, while our model handles diverse geometries and dynamics, including slippery and unbalanced objects.

c) Robustness to Friction Variations.: To test dynamics adaptation, we manipulate a bulldozer toy on four surfaces with increasing friction (μ_1–μ_4). Without adaptation, performance degrades and execution becomes unstable. With adaptation, our method maintains high success and consistent execution time across all conditions (Table IV).

D. Applications

By specifying a goal pose via a VLM and using our method as a pre-grasping step, we can reorient objects into graspable poses, boosting grasp success.

IV. Conclusion and Limitations

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. 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.

REFERENCES

[1] 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.

[2] 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.

[3] 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.

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

[5] 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.

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

[7] 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*.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] 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.

[13] 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.

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

[15] 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.

[16] 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.