Exploring the Limits of Vision-Language-Action Manipulation in Cross-task Generalization

Jiaming Zhou HKUST(GZ)
Ke Ye HKUST(GZ)
Jiayi Liu HKUST(GZ)
Teli Ma HKUST(GZ)
Zifan Wang HKUST(GZ)
Ronghe Qiu HKUST(GZ)
Kun-Yu Lin The University of Hong Kong
Zhilin Zhao Sun Yat-sen University
Junwei Liang* HKUST(GZ) and HKUST

Abstract

The generalization capabilities of vision-language-action (VLA) models to unseen tasks are crucial to achieving general-purpose robotic manipulation in open-world settings. However, the cross-task generalization capabilities of existing VLA models remain significantly underexplored. To address this gap, we introduce **AGNOSTOS**, a novel simulation benchmark designed to rigorously evaluate zero-shot cross-task generalization in manipulation. AGNOSTOS comprises 23 unseen manipulation tasks for test, which are distinct from common training task distributions, and incorporates two levels of generalization difficulty to assess robustness. Our systematic evaluation reveals that current VLA models, despite being trained on diverse datasets, struggle to generalize effectively to these unseen tasks. To overcome this limitation, we propose **Cross-Task In-Context Manipulation (X-ICM)**, a method that conditions large language models (LLMs) on in-context demonstrations from seen tasks to predict action sequences for unseen tasks. Additionally, we introduce a **dynamics-guided sample selection** strategy that identifies relevant demonstrations by capturing cross-task dynamics. On AGNOSTOS, X-ICM significantly improves zero-shot cross-task generalization performance over leading VLA models, achieving improvements of 6.0% over π_0 [1] and 7.9% over VoxPoser [2]. We believe AGNOSTOS and X-ICM will serve as valuable tools for advancing general-purpose robotic manipulation. Project page: https://jiaming-zhou.github.io/AGNOSTOS/.

1 Introduction

Vision-Language-Action (VLA) models [3, 4, 5, 6, 7, 2, 8, 9, 1, 10] have motivated a new era of robotic manipulation by integrating visual perception, language understanding, and action generation. Through large-scale pre-training on diverse data, including human videos [11, 12], real or simulated cross-embodiment robotic demonstrations [3, 13, 14], VLA models can effectively generalize across visual variations ***within known tasks*** (i.e., within-task generalization of seen tasks), such as handling objects in novel scenes or with altered properties. However, the true promise of VLA models lies in their capacity to generalize *across* tasks: to handle previously unseen combinations of objects, goals, and actions without prior exposure. This capability, **zero-shot cross-task generalization**, is essential for real-world deployment, where robots are expected to tackle novel tasks as they arise dynamically.

Despite the rapid progress in VLA research, most prior work [3, 4, 8, 9, 1] has focused on generalization testing in real-world environments. However, these evaluations are typically non-reproducible, and rarely target cross-task (i.e., ***unseen task***) zero-shot generalization. Recently, many

*Corresponding author: junweiliang@hkust-gz.edu.cn

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

 https://doi.org/10.52202/085713-4674

Figure 1: The proposed AGNOSTOS benchmark evaluates zero-shot cross-task generalization through two difficulty levels. Level-1 testing involves 13 unseen tasks sharing partial similarity (objects or motions) with seen tasks. Level-2 testing has 10 unseen tasks from entirely novel scenarios, requiring stronger generalization capabilities. We systematically assess three broad categories of vision-language-action models, revealing critical limits in their ability to adapt to unseen tasks.

works [15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26] have been devoted to developing comprehensive simulated benchmarks, which could be utilized to evaluate the generalization of VLA models. While promising, these efforts mainly assess ***within-task*** generalization, leaving zero-shot cross-task generalization largely unexplored.

To address this critical gap, we present AGNOSTOS, a novel benchmark for evaluating zero-shot cross-task generalization in robotic manipulation. Built on RLBench [18], our benchmark comprises 23 **unseen** tasks that are carefully curated to differ from 18 commonly used **seen** training tasks [27, 28]. As shown in Figure 1, to probe different aspects of generalization, these unseen tasks are categorized into two difficulty levels:

- **Level-1** (13 tasks) shares partial semantics (e.g., similar objects like "cups" or motions like "put") with seen tasks.
- **Level-2** (10 tasks) introduces entirely novel scenarios with no overlapping objects or actions.

We benchmark three broad categories of VLA models:

1. Models [1, 2, 8, 9] trained on large-scale real-world robotic demonstrations [29] or built on multimodal large language models (MLLMs);
2. Models [30, 31, 32] pre-trained on large-scale human action video datasets [11];
3. Models [33, 27, 34] trained purely on in-domain RLBench data with advanced architectures.

Our empirical findings reveal a key limitation: **none** of the existing models generalize effectively to unseen tasks, highlighting the need for approaches that directly address cross-task generalization.

https://doi.org/10.52202/085713-4674

Motivated by the success of in-context learning [35, 36, 37] in large language models (LLMs), we propose a **Cross-task In-context Manipulation (X-ICM)** method to address the challenges of zero-shot cross-task generalization. X-ICM uses demonstrations from seen tasks as in-context examples to prompt LLMs to generate action plans for unseen tasks. A key challenge under this setup is selecting relevant demonstrations; irrelevant prompts fail to activate appropriate knowledge in the LLM, leading to poor cross-task predictions. To address this, we introduce a **dynamics-guided sample selection** strategy. This approach leverages learned representations of task dynamics—captured by predicting final observations from initial states and task descriptions—to identify relevant demonstrations across tasks. Guided by the learned dynamics, the dynamics-aware prompts can be formed that enable X-ICM to elicit high-quality action predictions from LLMs, even for unfamiliar, unseen tasks.

Our contributions are threefold:

- We introduce AGNOSTOS, the first benchmark to systematically evaluate zero-shot cross-task generalization in robotic manipulation for VLA models, with 23 unseen tasks spanning two levels of generalization difficulty.
- We propose X-ICM, a novel method that combines in-context prompting and dynamics-guided sample selection to enhance zero-shot cross-task transfer in VLA models.
- We conduct extensive evaluations of diverse VLA models on AGNOSTOS, uncovering fundamental limitations in current approaches and demonstrating the superior generalization of X-ICM.

2 Related Works

Vision-Language-Action Models. Generalizable robotic manipulation models [3, 4, 5, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51], designed to enable robots to understand instructions and interact with the physical world, have largely followed two main paradigms. The first involves modular-based approaches [2, 52, 53, 54, 55, 56, 57, 58], where different MLLM components handle perception, language understanding, planning, and action execution. For example, VoxPoser [2] uses MLLMs to synthesize composable 3D value maps for manipulation. The second paradigm focuses on end-to-end approaches [3, 29, 9], training a policy model to directly map raw sensory inputs (e.g., vision, language instructions) to robot actions. The success of these models heavily depends on the scale and diversity of training data, which comes from various data sources, including large-scale human action videos [11] and large-scale cross-embodiment robotic data [29, 59]. Based on the data, OpenVLA [8] is the first fully open-sourced work that significantly promotes the development of the VLA community. π_0 [1] proposes a VLM-based flow matching architecture, which is trained on a diverse dataset from multiple dexterous robot platforms.

Benchmarks for Vison-Language-Action Models. Evaluating the generalization capabilities of VLA models requires robust and comprehensive benchmarks. While existing VLA models mainly focus on real-world testing, which suffers from reproducibility issues and rarely focuses systematically on zero-shot generalization to unseen tasks. Recently, many simulated benchmarks [15, 16, 17, 18, 19, 20, 21, 23, 24, 25, 26] have been proposed to offer a controlled environment for rigorous evaluation. Colosseum [24] enables evaluation of models across 14 axes of perturbations, including changes in color and texture, etc. GemBench [25] designs four levels of generalization, spanning novel placements, rigid and articulated objects, and complex long-horizon tasks. We find that these benchmarks have predominantly concentrated on evaluating visual variations **within tasks**, assessing how well models can generalize to new scenes or altered attributes of objects within the same task. However, these benchmarks do not focus on the more challenging aspect of **zero-shot cross-task evaluation**, where models must generalize to entirely new tasks with unseen combinations of object categories and motions. This gap in evaluation limits our understanding of the true generalization capabilities of VLA models. Motivated by this, this work proposes the first zero-shot cross-task manipulation benchmark, expanding the evaluation scope of the generalization capabilities of VLA models.

In-context Learning with Large Language Models (LLMs). LLMs [60, 35, 61, 62, 63, 64, 65, 66] have demonstrated a remarkable capability known as in-context learning (ICL) [35, 36, 37], where they can learn to perform new tasks based on a few examples provided within the prompt, without requiring updates to parameters. The potential of ICL is being explored in robotics [67, 68, 69, 70, 71, 72]. Prior works like KAT [67] and RoboPrompt [71] have shown that off-the-shelf LLMs can predict robot actions directly using within-task in-context samples. By extending their paradigms to a

 https://doi.org/10.52202/085713-4674

cross-task setting, this work specifically focuses on the zero-shot cross-task generalization problem. Our X-ICM model uses demonstrations from seen tasks as in-context examples to prompt LLMs for action generation in completely unseen tasks. Under the zero-shot cross-task setting, the selection of in-context samples [73, 74, 75, 76] is crucial for a robust generalization. Our X-ICM method addresses this challenge by introducing a dynamics-guided sample selection strategy, ensuring that the selected seen demonstrations are relevant to the unseen tasks, thereby stimulating the cross-task generalization capabilities of LLMs.

3 AGNOSTOS: Zero-shot Cross-task Generalization Benchmark

To systematically assess the **zero-shot cross-task generalization** ability of vision-language-action (VLA) models, we introduce AGNOSTOS, a reproducible benchmark built upon the RLBench simulation environment. AGNOSTOS features 18 seen tasks for training and 23 unseen tasks for rigorous generalization testing.

Training. For training, we adopt the standard set of 18 RLBench tasks that are widely used in prior work [27, 28]. Examples of these seen tasks are shown in Figure A1. We collect 200 language-conditioned demonstrations per task, resulting in 3600 demonstrations in total. These demonstrations enable VLA models to be fine-tuned to reduce the domain and embodiment gaps between pre-training data and RLBench data.

Testing. As illustrated in Figure 1, AGNOSTOS comprises 23 held-out unseen tasks with semantics that are disjoint from the seen set (videos of all tasks are available in the Supplementary Materials). We categorize the unseen tasks into two difficulty levels. **Level-1**: 13 tasks that share partial semantic similarity with seen tasks—either in object types (e.g., cups) or motion primitives (e.g., stacking). **Level-2**: 10 tasks that exhibit no overlap in either object categories or motion types, requiring broader compositional reasoning and semantic extrapolation. Details on task curation and difficulty categorization are provided in Section A1.1 of the Appendix. For each unseen task, we perform three test runs with different seeds, each consisting of 25 rollouts, and report the mean and standard deviation of success rates.

To explore generalization boundaries, AGNOSTOS evaluates three broad families of VLA[2]models:

1. **Foundation VLA models**: trained on large-scale real-world cross-embodiment robotic data [29] or built upon LLM or VLM models, including OpenVLA [8], RDT [9], π_0 [1], LLARVA [77], SAM2Act [34], 3D-LOTUS++ [25], and VoxPoser [2].

2. **Human-video VLA models**: pre-trained on large-scale human action videos [11] to capture rich human-object interactions for downstream robotic fine-tuning, including R3M [30], D4R [31], R3M-Align [32], and D4R-Align [32].

3. **In-domain VLA models**: trained from scratch on RLBench's 18 seen tasks with task-specific model architectures. These serve as strong baselines without domain mismatch, including PerAct [27], RVT [28], RVT2 [33], Sigma-Agent [78], and Instant Policy [69].

To ensure a fair comparison, we fine-tune all models on the same 18 seen tasks when the models involve the embodiment gaps, following the official protocols of each method. A detailed description of the fine-tuning process and evaluation on AGNOSTOS is provided in Section A1.2 of the Appendix. Table 1 compares AGNOSTOS with existing robotic manipulation benchmarks in terms of their support for cross-task evaluation, i.e., considering test tasks that have different object categories or motions from training tasks. While many benchmarks focus on within-task visual generalization, few explicitly support **zero-shot cross-task generalization test**, and even fewer include tasks as challenging as our Level-2 testing scenarios. Even benchmarks that include some form of cross-task evaluation[3] often test on a narrow set of tasks, and typically evaluate only in-domain models, neglecting foundation and human-video pre-trained VLA models. AGNOSTOS fills this gap by **supporting broad model coverage and introducing novel, difficult tasks in Level-2**, providing a more comprehensive and diagnostic assessment of cross-task generalization.

[2]We only consider evaluating the VLA models that are fully open-source.

[3]We carefully categorize their test tasks into our defined two-level difficulty task sets.

https://doi.org/10.52202/085713-4674

Table 1: Comparison of manipulation benchmarks that focuses on cross-task generalization testing. The comparison evaluates whether each benchmark includes cross-task testing, the types of VLA models evaluated, and the number of Level-1 and Level-2 unseen tasks included.

Benchmark	Simulator	No. of train tasks	No. of test tasks	Cross-task Zero-shot Generalization Test					
				Cross task	**Evaluated VLA models**			**Level-1 unseen tasks**	**Level-2 unseen tasks**
					In-domain	Human-video	Foundation		
RLBench-18Task [27]	RLBench	18	18	✗	-	-	-	-	-
CALVIN [17]	PyBullet [79]	34	34	✗	-	-	-	-	-
Colosseum [24]	RLBench	20	20	✗	-	-	-	-	-
VLMBench [23]	RLBench	8	8	✗	-	-	-	-	-
Ravens [80]	PyBullet [79]	10	10	✓	✓	✗	✗	3	0
VIMA-Bench [81]	Ravens	13	17	✓	✓	✗	✗	4	0
GemBench [25]	RLBench	16	44	✓	✓	✗	✗	15	0
AGNOSTOS (Ours)	RLBench	18	23	✓	✓	✓	✓	13	10

4 X-ICM: Cross-task In-context Manipulation Method

To push the boundaries of zero-shot cross-task generalization in vision-language-action (VLA) models, we propose a method called Cross-task In-context Manipulation (X-ICM). Leveraging the cross-task generalization capabilities of LLMs, X-ICM utilizes demonstrations from seen tasks as in-context examples. The dynamic characteristics of these examples are used to prompt the LLM to predict action sequences for unseen tasks. In contrast to prior works [71, 67] that apply LLMs' in-context learning to within-task generalization, our work is the first to extend this paradigm to a **zero-shot cross-task setting**. A central challenge in this setting is that the selection of in-context demonstrations significantly affects generalization performance. To address this, we design a **dynamics-guided sample selection** module that measures similarities between dynamic representations of seen and unseen tasks to guide the selection process, resulting in improved cross-task generalization.

4.1 Problem Definition and Method Overview

We tackle the problem of zero-shot cross-task robotic manipulation by exploiting the in-context learning ability of off-the-shelf LLMs. We assume access to a dataset of demonstrations from seen tasks, denoted as $\mathcal{D}^s = \{V_i^s, A_i^s, L_i^s\}_{i=1}^N$, where N is the total number of seen demonstrations, $V_i^s = \{v_{i,1}^s, \cdots, v_{i,T}^s\}$, $A_i^s = \{a_{i,1}^s, \cdots, a_{i,T}^s\}$, and L_i^s are the visual observation sequence, action sequence, and language description of the i-th seen demonstration (T is the sequence length, which varies for each demonstration). For an unseen task with the given initial visual observation v^u, and language description L^u, our zero-shot cross-task manipulation method can be formulated as follows:

$$A_{pred}^u = \{a_1^u, \cdots, a_t^u\} = LLM(\mathcal{P}(\mathcal{F}^{sel}(\mathcal{D}^s), v^u, L^u)), \tag{1}$$

where $\mathcal{F}^{sel}$ is the cross-task in-context demonstration selection process, $\mathcal{P}$ is the cross-task in-context prompt construction process, and A_{pred}^u is the predicted action sequence for the tested unseen task.

Figure 2 outlines our X-ICM framework, which comprises two core modules. The **Dynamics-guided Sample Selection module** introduces a dynamics diffusion model, and then retrieves effective demonstrations based on dynamic similarities between seen and unseen demonstrations. The **Cross-task In-context Prediction module** constructs the LLM prompt using retrieved demonstrations to predict action sequences for the unseen task. Next, we describe each module in detail below.

4.2 Dynamics-guided Sample Selection module

Selecting effective in-context examples is essential for achieving robust cross-task generalization. We propose a two-stage Dynamics-guided Sample Selection module that learns dynamic representations from demonstrations using diffusion models, and retrieves examples that are most relevant to the current unseen task based on the similarities between the dynamic representations of demonstrations.

Diffusion-based dynamics modeling stage. To effectively capture the dynamic representations within each demonstration, we train a dynamics diffusion model $\mathcal{G}$ on all seen demonstrations. For each demonstration (e.g., the i-th), the dynamics diffusion model $\mathcal{G}$ takes the initial visual observation

https://doi.org/10.52202/085713-4674

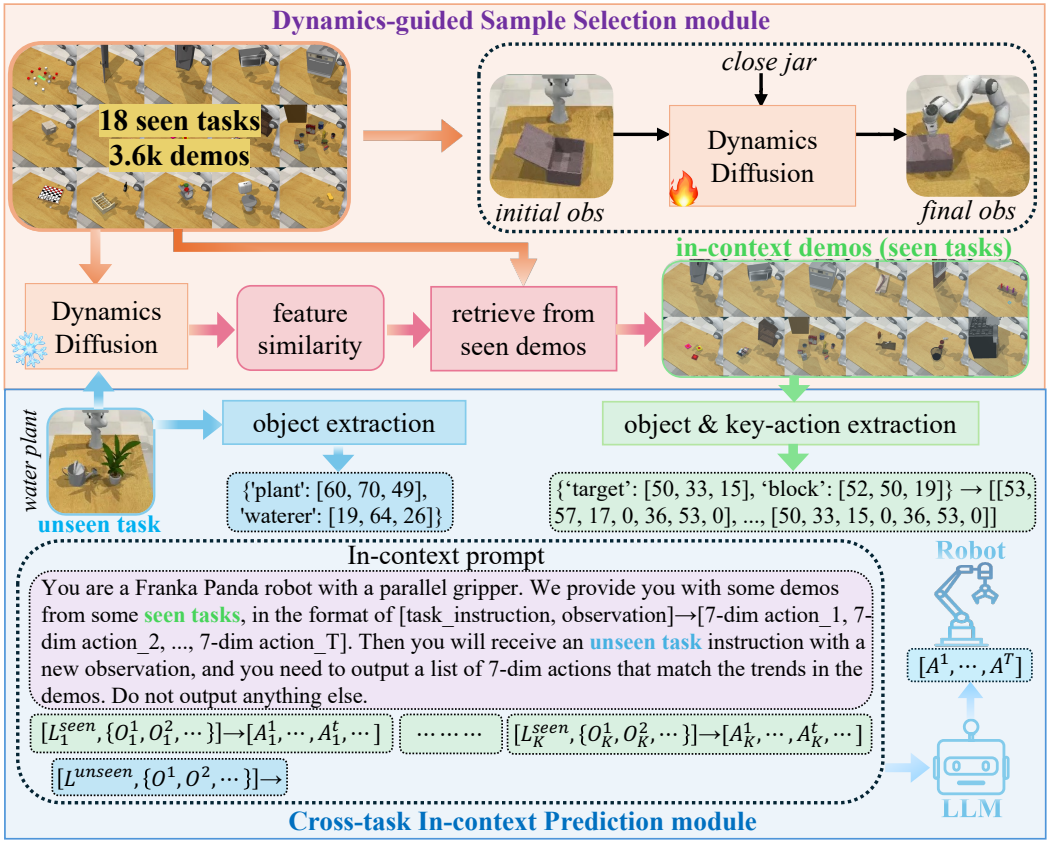


Figure 2: **X-ICM Method Overview.** X-ICM employs a dynamics-guided sample selection module to retrieve effective demonstrations from seen tasks for each tested unseen task. These demonstrations are then used by the cross-task in-context prediction module to construct the prompt that drives the LLM to predict the corresponding action sequence.

$v^s_{i,1}$ and language description L^s_i as inputs, and predicts the future observation that matches the final visual observation $v^s_{i,T}$. The model $\mathcal{G}$ is initialized from InstructPix2Pix [82] and is optimized by:

$$\min_{\mathcal{G}} \mathbb{E}_{i,z,\epsilon\sim\mathcal{N}(0,I)} \left[\left\| \epsilon - \epsilon_{\mathcal{G}}(v^s_{i,T,z}, z, v^s_{i,1}, L^s_i) \right\|_2^2 \right], \tag{2}$$

where z is the diffusion timestep, ϵ is the noise, $v^s_{i,T,z}$ is the noised final observation, and $\epsilon_{\mathcal{G}}$ is the noise predictor in model $\mathcal{G}$. By predicting the future, the inherent dynamics of each demonstration can be effectively modeled. Figure A6 shows the generation results of our dynamics diffusion model.

Dynamics-guided retrieving stage. Once trained, we use $\mathcal{G}$ to extract a dynamic feature f^s_i for the i-th seen demonstration:

$$f^s_i = [f^{s,vis}_i, f^{s,lang}_i] = \mathcal{G}(v^s_{i,1}, L^s_i) \in \mathbb{R}^{2\times 1024}, \tag{3}$$

where $f^{s,vis}_i \in \mathbb{R}^{1024}$ is the predicted latent feature of the target visual observation, $f^{s,lang}_i \in \mathbb{R}^{1024}$ is the textual feature of the language description. For the unseen task, we similarly compute its feature f^u. Then, we compute cosine similarities between features and select the top-K seen demonstrations with the highest similarity:

$$R_i = \frac{f^u \cdot f^s_i}{\|f^u\| \|f^s_i\|} \quad i \in \{1, \ldots, N\}, \tag{4}$$

$$\mathcal{I}_{idx} = \arg\operatorname{topk}^K_{i\in\{1,\ldots,N\}}(R_i), \tag{5}$$

where R_i represents the cross-task dynamics similarity score between the tested unseen task and the i-th seen demonstration. And the set $\mathcal{I}_{idx}$ contains the indices of K seen demonstrations that achieve the highest similarity scores. These K seen demonstrations are retrieved to construct the cross-task in-context prompt.

4.3 Cross-task In-context Prediction module

For a tested unseen task, we use the obtained K most-relevant seen demonstrations to construct the in-context prompt for cross-task action prediction with LLMs. Our Cross-task In-context Prediction module follows the practices in existing within-task in-context manipulation methods [71, 67], but we are the first to extend this paradigm to the zero-shot cross-task setting.

Textualizing key information. As shown at the bottom of Figure 2, for each selected seen demonstration, we extract the 3D positions of the centers of objects and the key-action sequence. For the 3D positions of objects' centers, we use GroundingDINO [83] to detect the 2D positions of objects, and then use the depth, intrinsic, and extrinsic information of the camera to obtain the 3D coordinates of objects' centers under the robot-base coordinate system. Additionally, we evenly divide the workspace into $100 \times 100 \times 100$ grids, and then normalize the 3D coordinates of the objects' centers into the grid's coordinate system. Thus, for the m-th object O_k^m in the k-th seen demonstration, we textualize it as $O_k^m =$ "$objectname$: [x, y, z]" (e.g., "block: [52, 50, 19]"). For the key-action sequence, following [84], we select the actions when the gripper state changes or the joint velocities are near zero. In this way, the execution process of a demonstration can be determined by several key-actions. We use the 7-dimensional state of the end-effector to represent the robot action, i.e., the 3D positions, the roll-pitch-yaw angles, and the gripper's open/close. The 3D positions of the end-effector are also normalized into the grid's coordinate system of the workspace. And the angles of end-effector are discretized into 72 bins, with each occupying 5 degrees. Thus, for the t-th key-action A_k^t in the k-th seen demonstration, we textualize it as $A_k^t =$ "[x, y, z, $roll$, $pitch$, yaw, $gripper$]" (e.g., "[53, 57, 17, 0, 36, 53, 0]").

Constructing in-context prompt. With the above textualization of object and key-action information, for each seen demonstration (e.g., the k-th), we can textualize it as the mapping from its language description and object context to its key-action context:

$$\text{“}[L_k^{seen}, \{O_k^1, \cdots, O_k^m, \cdots\}] \rightarrow [A_k^1, \cdots, A_k^t, \cdots]\text{”}. \tag{6}$$

We textualize all K selected seen demonstrations in this manner, and concatenate them in descending order based on their dynamics similarity scores to the tested unseen task. For the tested unseen task, we textualize it using its language description and object context, i.e., "$[L^{unseen}, \{O^1, \cdots, O^m, \cdots\}] \rightarrow$". Finally, the cross-task in-context prompt is formed by concatenating the system prompt, the textualized all seen demonstrations, and the textualized tested unseen task (see the bottom of Figure 2 and an example in Figure A7 of the Appendix). This prompt will serve as the input of LLMs, which incentivizes the cross-task generalization capability of LLMs to predict the key-action sequence for the tested unseen task.

5 Experiments

Implementation Details. For our X-ICM method on the AGNOSTOS benchmark, we use a total of $N = 3600$ seen demonstrations. During in-context prompt construction, we select $K = 18$ demonstrations. We mainly use off-the-shelf Qwen2.5-Instruct [65] models with 7B and 72B parameters, referred to as X-ICM (7B) and X-ICM (72B), respectively. These are deployed using two or eight A6000 GPUs. For a fair comparison with existing zero-shot baselines (e.g., VoxPoser [2]), in simulation we use the ground-truth positions of objects. Ablation on using different sizes of LLMs is presented in Sec A2.2 of the Appendix.

5.1 Benchmarking Vision-Language-Action Models

Table 2 presents the zero-shot cross-task performance of our X-ICM models on 23 unseen tasks from the benchmark, including 13 Level-1 and 10 Level-2 tasks. These tasks are designed to evaluate the generalization ability of VLA models under varying levels of difficulty. We compare our models against a diverse set of baselines, including models trained on in-domain RLBench data, human videos, and LLM/VLM-based foundations. Key observations include:

- X-ICM (7B) and X-ICM (72B) achieve average success rates of 23.5% and 30.1%, respectively, outperforming all existing VLA models.
- On Level-1 tasks, X-ICM (7B) surpasses the prior SoTA π_0 [1] by 6.9%. For Level-2 tasks, performance gains are more pronounced with the 72B model.

 https://doi.org/10.52202/085713-4674

Table 2: Cross-task zero-shot manipulation performance on 23 unseen tasks, where the column headers show the abbreviation of each unseen task (see full task names in Table A1). N/A indicates that the tested tasks are removed since they overlap with the training tasks of the methods. Level-1 and Level-2 represent tasks with two different difficulty levels. The prefix * indicates second best.

	methods	**Level-1 tasks**												
		Toilet	*Knife*	*Fridge*	*Microwave*	*Laptop*	*Phone*	*Seat*	*LampOff*	*LampOn*	*Book*	*Umbrella*	*Grill*	*Bin*
in-domain training	PerAct [27]	0.0	5.3	37.3	64.0	2.7	0.0	72.0	0.0	1.3	0.0	1.3	8.0	54.7
	RVT [28]	0.0	2.7	50.7	26.7	50.7	2.7	40.0	0.0	1.3	0.0	1.3	0.0	6.7
	Sigma-Agent [78]	0.0	9.3	56.0	9.3	30.7	1.3	65.3	1.3	0.0	0.0	0.0	1.3	4.0
	RVT2 [33]	0.0	1.3	0.0	17.3	42.7	1.3	62.7	2.7	1.3	0.0	1.3	5.3	34.7
	InstantPolicy [69]	0.0	1.3	13.3	4.0	4.0	18.7	24.0	0.0	0.0	0.0	0.0	0.0	0.0
human-video pretraining	D4R [31]	0.0	8.0	32.0	30.7	24.0	0.0	65.3	20.0	4.0	0.0	0.0	0.0	0.0
	R3M [30]	0.0	0.0	37.3	22.7	25.3	1.3	62.7	6.7	4.0	0.0	0.0	0.0	0.0
	D4R-Align [32]	0.0	2.7	45.3	74.7	24.0	0.0	41.3	0.0	0.0	1.3	0.0	0.0	0.0
	R3M-Align [32]	0.0	4.0	49.3	25.3	21.3	0.0	49.3	0.0	5.3	0.0	0.0	1.3	1.3
VLA foundations	OpenVLA [8]	0.0	5.3	38.7	40.0	57.3	0.0	53.3	12.0	1.3	1.3	0.0	10.7	0.0
	RDT [9]	0.0	0.0	46.7	13.3	14.7	0.0	50.7	0.0	0.0	1.3	0.0	8.0	0.0
	π_0 [1]	0.0	5.3	85.3	24.0	40.0	1.3	64.0	18.7	8.0	1.3	0.0	33.3	1.3
	LLARVA [77]	0.0	0.0	12.0	0.0	6.7	0.0	40.0	0.0	0.0	0.0	0.0	0.0	0.0
	3D-LOTUS [25]	0.0	6.7	N/A	N/A	N/A	0.0	6.7	0.0	0.0	0.0	0.0	13.3	5.3
	3D-LOTUS++ [25]	0.0	5.3	N/A	N/A	N/A	9.3	68.0	10.7	0.0	0.0	0.0	29.3	13.3
	SAM2Act [34]	0.0	0.0	36.0	40.0	6.7	6.7	62.7	6.7	0.0	1.3	1.3	9.3	0.0
	VoxPoser [2]	0.0	0.0	0.0	0.0	5.3	8.0	28.0	88.7	25.3	0.0	0.0	0.0	82.7
Ours	**X-ICM (7B)**	1.3	26.7	22.7	45.3	33.3	57.3	48.0	58.7	50.7	1.3	0.0	8.0	18.7
	X-ICM (72B)	6.7	69.3	12.7	58.7	34.0	68.0	51.3	86.7	74.7	2.0	1.3	5.3	18.7

	methods	**Level-2 tasks**										**Level-1 avg (std)**	**Level-2 avg (std)**	**All avg (std)**
		USB	*Lid*	*Plate*	*Ball*	*Scoop*	*Rope*	*Oven*	*Buzz*	*Plants*	*Charger*			
in-domain training	PerAct [27]	58.7	2.7	0.0	0.0	0.0	0.0	1.3	4.0	6.7	2.7	19.0 (1.4)	7.6 (1.1)	14.0 (0.9)
	RVT [28]	89.3	2.7	0.0	0.0	0.0	0.0	4.0	8.0	5.3	4.0	14.0 (1.4)	11.3 (1.6)	12.8 (0.2)
	Sigma-Agent [78]	88.0	0.0	0.0	0.0	0.0	0.0	4.0	8.0	5.3	1.3	13.7 (1.6)	10.7 (1.7)	12.4 (0.4)
	RVT2 [33]	22.7	40.0	0.0	0.0	0.0	0.0	0.0	1.3	1.3	1.3	13.1 (0.4)	6.7 (1.3)	10.3 (0.6)
	InstantPolicy [69]	26.7	1.3	0.0	0.0	0.0	0.0	0.0	1.3	0.0	0.0	4.3 (4.2)	2.9 (1.4)	3.7 (3.0)
human-video pretraining	D4R [31]	98.7	0.0	0.0	0.0	0.0	0.0	1.3	1.3	1.3	4.0	14.1 (0.3)	10.7 (0.2)	12.6 (0.2)
	R3M [30]	48.0	0.0	0.0	0.0	0.0	0.0	8.0	2.7	2.7	1.3	12.3 (1.4)	6.3 (0.9)	9.7 (0.6)
	D4R-Align [32]	89.3	1.3	0.0	0.0	0.0	0.0	8.0	6.7	0.0	1.3	14.5 (1.0)	10.7 (0.2)	12.8 (0.6)
	R3M-Align [32]	90.7	0.0	1.3	0.0	0.0	0.0	2.7	13.3	4.0	0.0	12.9 (0.7)	11.2 (0.7)	12.2 (0.3)
VLA foundations	OpenVLA [8]	77.3	0.0	0.0	0.0	0.0	0.0	6.7	5.3	2.7	0.0	16.9 (1.3)	9.2 (0.7)	13.6 (0.8)
	RDT [9]	100.0	29.3	4.0	0.0	0.0	0.0	8.0	2.7	0.0	0.0	10.4 (0.5)	14.4 (0.9)	12.1 (0.4)
	π_0 [1]	97.3	0.0	1.3	0.0	0.0	0.0	14.7	5.3	1.3	0.0	*21.7 (0.4)	12.0 (0.9)	*17.5 (0.4)
	LLARVA [77]	24.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.5 (0.1)	2.4 (0.0)	3.6 (0.1)
	3D-LOTUS [25]	85.3	0.0	1.3	0.0	0.0	0.0	0.0	4.0	0.0	0.0	3.2 (0.5)	9.1 (0.7)	6.2 (0.5)
	3D-LOTUS++ [25]	90.7	30.7	0.0	0.0	5.3	1.3	8.0	8.7	6.7	0.0	13.6 (1.0)	15.1 (1.1)	14.4 (1.0)
	SAM2Act [34]	92.0	49.3	0.0	0.0	0.0	0.0	1.3	6.7	4.0	5.3	13.1 (0.4)	*15.9 (1.3)	14.0 (0.7)
	VoxPoser [2]	32.0	76.0	0.0	8.0	0.0	0.0	0.0	1.3	0.0	4.0	18.1 (0.4)	12.1 (0.4)	15.6 (0.2)
Ours	**X-ICM (7B)**	98.7	20.0	6.7	9.3	0.0	6.7	16.0	2.7	5.3	4.0	**28.6 (1.9)**	**16.9 (1.3)**	**23.5 (1.6)**
	X-ICM (72B)	98.7	13.3	4.7	36.0	0.7	16.0	20.7	7.3	2.7	2.7	**37.6 (1.4)**	**20.3 (1.7)**	**30.1 (1.0)**

- While some VLA foundation models show decent performance, particularly π_0 on Level-1 tasks and SAM2Act [34]/3D-LOTUS++ [25] on Level-2 tasks, they fall short of the broad generalization capabilities demonstrated by our X-ICM models.
- All prior models completely fail on at least eight of the 23 tasks. In contrast, X-ICM (7B) fails on only two, and X-ICM (72B) succeeds on all.

5.2 Ablation Studies

Dynamics-guided sample selection. We assess the impact of the dynamics-guided sample selection module by comparing X-ICM (72B) with and without it (denoted as *w/o sel*, using random sampling) at the top-left of Table 3. Incorporating the module notably boosts performance and robustness across both task levels. This confirms its effectiveness in identifying informative demonstrations that enhance the LLM's cross-task generalization ability. Further analysis, including visualizations from the dynamics diffusion model and comparisons of dynamic features, are provided in Appendix A2.1.

Table 3: Effects of dynamics-guided sample selection module and different model sizes.

Models	Level-1	Level-2	All
X-ICM (72B) *w/o sel*	30.7 (4.7)	18.0 (2.2)	25.2 (3.2)
X-ICM (72B)	37.6 (1.4)	20.3 (1.7)	30.1 (1.0)

Table 4: Effect of using different LLM models.

LLMs	Level-1	Level-2	All
Deepseek-R1-Distill-Qwen-7B	10.7 (1.1)	7.9 (0.5)	9.5 (0.5)
Llama3.0-8B-Instruct	17.4 (0.7)	11.7 (1.6)	15.1 (0.3)
Ministral-8B-Instruct-2410	22.9 (0.7)	14.8 (0.3)	19.5 (0.5)
InternLM3-8B-Instruct	27.9 (0.7)	13.3 (0.4)	21.8 (0.5)
Qwen2.5-7B-Instruct	**28.6 (1.9)**	**16.9 (1.3)**	**23.5 (1.6)**

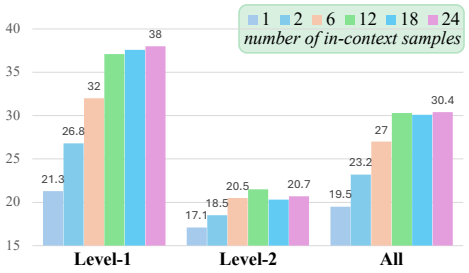


Figure 3: The effect of varying the number of in-context demonstrations.

Effect of the number of in-context demonstrations. We explore how performance varies with the number of in-context demonstrations using the Qwen2.5-72B-Instruct model. Figure 3 shows that performance improves rapidly as the number increases from 1 to 12, after which it plateaus. This suggests that a moderate number of relevant demonstrations is critical. Too many may introduce irrelevant noise and diminish gains.

Comparison of LLM backbones. Table 4 compares the performance of X-ICM (7B) when replacing Qwen2.5-7B-Instruct with alternative 7B/8B models, including Deepseek-R1-Distill-Qwen-7B [85], Llama3.0-8B-Instruct [86], Ministral-8B-Instruct-2410 [87], and InternLM3-8B-Instruct [88]. The results show significant variability in performance, underscoring the importance of choosing a powerful and well-aligned LLM backbone.

5.3 Real-world Experiments

To evaluate real-world applicability, we test X-ICM on five physical manipulation tasks using an xArm7 robot arm, DH-Robotics gripper, and a third-person Orbbec camera: *put block into bin*, *push button*, *put bottle into box*, *stack blocks*, and *stack cups*. We collect 20 demonstrations for each of the five tasks. During testing, we use the demonstrations from the other four tasks to construct the cross-task in-context prompt, enabling zero-shot generalization. The Qwen2.5-72B-Instruct LLM is used as the backbone and the number of seen demonstrations is set to 18. Each task is executed 20 times, and we report average success rates. Results shown at the bottom of Figure 4 demonstrate the strong real-world **zero-shot cross-task** performance of our approach. We provide additional details in Section A3 of the Appendix, and showcase some testing videos that include both successful and failed cases in the Supplementary Materials.

In addition, we also evaluate the proposed X-ICM model on long-horizon tasks under the zero-shot cross-task setting. We use tasks *put block into bin*, *push button*, *put bottle into box*, *stack blocks*, and *pull out the middle block* as seen tasks, where we have 10 demonstrations for each task. The long-horizon task we evaluated is *clean the table*, according the robot's visual observation, we decompose the long-horizon task into three sequential sub-tasks, ie, *stack cups*, *put the stacked cups into plate*, *place the mango on top of the stacked cups*. We evaluated this task over 20 rollouts, with sub-task success contingent on the success of all prior sub-tasks. For our X-ICM method, the success rates on sub-tasks "stack cups", "put the stacked cups into plate", "place the mango on top of the stacked cups." are 40%, 25%, and 5%, respectively. Thus, the overall success rate for the long-horizon task is 5%. These results align with expectations, as the sequential nature of long-horizon tasks leads to cumulative failure. The cross-task zero-shot setting exacerbates these difficulties. These findings highlight the need for more advanced algorithms to handle long-horizon tasks in zero-shot scenarios.

Figure 4: **Results of five real-world tasks.** The tests are conducted in a zero-shot cross-task manner.

https://doi.org/10.52202/085713-4674

6 Conclusions and Discussions

Conclusions. In this work, we have introduced AGNOSTOS, a benchmark for evaluating zero-shot cross-task generalization in robotic manipulation. With 23 unseen tasks across two difficulty levels, AGNOSTOS provides a rigorous testbed for assessing the limits of Vision-Language-Action (VLA) models. Our evaluation of diverse VLA models reveals their significant limitations in unseen task generalization. To address this, we propose X-ICM, a cross-task in-context manipulation method that leverages the cross-task generalization capabilities of LLMs. The X-ICM achieves substantial improvements over existing VLA models, demonstrating robust zero-shot cross-task generalization.

Discussions. While X-ICM significantly enhances zero-shot cross-task generalization, its performance on many unseen tasks, particularly those with both novel objects and motion primitives, remains limited due to LLMs' challenges in extrapolating beyond pre-training data and in-context demonstrations. In addition, the use of visual information is limited to textualizing object information, which may ignore important visual context in the raw data. To address these limitations, future work could integrate multi-modal reasoning, combining vision, language, and action data to improve generalization to novel semantics. Additionally, leveraging generalizable concepts like object trajectories as a bridge between MLLMs' perception and dynamic action sequence prediction could reduce cross-task generalization complexity. Scaling X-ICM to diverse robotic embodiments would further enhance its versatility across varied platforms and sensor configurations. We believe the proposed X-ICM benchmark and X-ICM method will inspire future research in generalizable robotic manipulation, paving the way for robots that can seamlessly adapt to open-world environments.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (No. 62306257), the Guangzhou Municipal Science and Technology Project (No. 2024A03J0619 and No. 2024A04J4390) and the Guangzhou-HKUST(GZ) Joint Funding Program (Grant No.2023A03J0008), Education Bureau of Guangzhou Municipality.

References

[1] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[2] Wenlong Huang, Chen Wang, Ruohan Zhang, Yunzhu Li, Jiajun Wu, and Li Fei-Fei. Voxposer: Composable 3d value maps for robotic manipulation with language models. *arXiv preprint arXiv:2307.05973*, 2023.

[3] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Xi Chen, Krzysztof Choromanski, Tianli Ding, Danny Driess, Avinava Dubey, Chelsea Finn, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

[4] Octo Model Team, Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, et al. Octo: An open-source generalist robot policy. *arXiv preprint arXiv:2405.12213*, 2024.

[5] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. *arXiv preprint arXiv:2403.09631*, 2024.

[6] Jonathan Yang, Catherine Glossop, Arjun Bhorkar, Dhruv Shah, Quan Vuong, Chelsea Finn, Dorsa Sadigh, and Sergey Levine. Pushing the limits of cross-embodiment learning for manipulation and navigation. *arXiv preprint arXiv:2402.19432*, 2024.

[7] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

[8] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, et al. Openvla: An open-source vision-language-action model. *arXiv preprint arXiv:2406.09246*, 2024.

[9] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

[10] Gemini Robotics Team, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Travis Armstrong, Ashwin Balakrishna, Robert Baruch, Maria Bauza, Michiel Blokzijl, et al. Gemini robotics: Bringing ai into the physical world. *arXiv preprint arXiv:2503.20020*, 2025.

[11] Kristen Grauman, Andrew Westbury, Eugene Byrne, Zachary Chavis, Antonino Furnari, Rohit Girdhar, Jackson Hamburger, Hao Jiang, Miao Liu, Xingyu Liu, et al. Ego4d: Around the world in 3,000 hours of egocentric video. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18995–19012, 2022.

[12] Kristen Grauman, Andrew Westbury, Lorenzo Torresani, Kris Kitani, Jitendra Malik, Triantafyllos Afouras, Kumar Ashutosh, Vijay Baiyya, Siddhant Bansal, Bikram Boote, et al. Ego-exo4d: Understanding skilled human activity from first-and third-person perspectives. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19383–19400, 2024.

[13] Frederik Ebert, Yanlai Yang, Karl Schmeckpeper, Bernadette Bucher, Georgios Georgakis, Kostas Daniilidis, Chelsea Finn, and Sergey Levine. Bridge data: Boosting generalization of robotic skills with cross-domain datasets. *arXiv preprint arXiv:2109.13396*, 2021.

[14] Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Shenyuan Gao, Xindong He, Xu Huang, Shu Jiang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. *arXiv preprint arXiv:2503.06669*, 2025.

[15] Mohit Shridhar, Jesse Thomason, Daniel Gordon, Yonatan Bisk, Winson Han, Roozbeh Mottaghi, Luke Zettlemoyer, and Dieter Fox. Alfred: A benchmark for interpreting grounded instructions for everyday tasks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10740–10749, 2020.

[16] Tongzhou Mu, Zhan Ling, Fanbo Xiang, Derek Yang, Xuanlin Li, Stone Tao, Zhiao Huang, Zhiwei Jia, and Hao Su. Maniskill: Generalizable manipulation skill benchmark with large-scale demonstrations. *arXiv preprint arXiv:2107.14483*, 2021.

[17] Oier Mees, Lukas Hermann, Erick Rosete-Beas, and Wolfram Burgard. Calvin: A benchmark for language-conditioned policy learning for long-horizon robot manipulation tasks. *IEEE Robotics and Automation Letters*, 7(3):7327–7334, 2022.

[18] Stephen James, Zicong Ma, David Rovick Arrojo, and Andrew J Davison. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020.

[19] Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *Advances in Neural Information Processing Systems*, 36:44776–44791, 2023.

[20] Soroush Nasiriany, Abhiram Maddukuri, Lance Zhang, Adeet Parikh, Aaron Lo, Abhishek Joshi, Ajay Mandlekar, and Yuke Zhu. Robocasa: Large-scale simulation of everyday tasks for generalist robots. *arXiv preprint arXiv:2406.02523*, 2024.

[21] Chengshu Li, Ruohan Zhang, Josiah Wong, Cem Gokmen, Sanjana Srivastava, Roberto Martín-Martín, Chen Wang, Gabrael Levine, Michael Lingelbach, Jiankai Sun, et al. Behavior-1k: A benchmark for embodied ai with 1,000 everyday activities and realistic simulation. In *Conference on Robot Learning*, pages 80–93. PMLR, 2023.

[22] Shiduo Zhang, Zhe Xu, Peiju Liu, Xiaopeng Yu, Yuan Li, Qinghui Gao, Zhaoye Fei, Zhangyue Yin, Zuxuan Wu, Yu-Gang Jiang, et al. Vlabench: A large-scale benchmark for language-conditioned robotics manipulation with long-horizon reasoning tasks. *arXiv preprint arXiv:2412.18194*, 2024.

[23] Kaizhi Zheng, Xiaotong Chen, Odest Chadwicke Jenkins, and Xin Wang. Vlmbench: A compositional benchmark for vision-and-language manipulation. *Advances in Neural Information Processing Systems*, 35:665–678, 2022.

[24] Wilbert Pumacay, Ishika Singh, Jiafei Duan, Ranjay Krishna, Jesse Thomason, and Dieter Fox. The colosseum: A benchmark for evaluating generalization for robotic manipulation. *arXiv preprint arXiv:2402.08191*, 2024.

[25] Ricardo Garcia, Shizhe Chen, and Cordelia Schmid. Towards generalizable vision-language robotic manipulation: A benchmark and llm-guided 3d policy. *arXiv preprint arXiv:2410.01345*, 2024.

 https://doi.org/10.52202/085713-4674

[26] Rui Yang, Hanyang Chen, Junyu Zhang, Mark Zhao, Cheng Qian, Kangrui Wang, Qineng Wang, Teja Venkat Koripella, Marziyeh Movahedi, Manling Li, et al. Embodiedbench: Comprehensive benchmarking multi-modal large language models for vision-driven embodied agents. *arXiv preprint arXiv:2502.09560*, 2025.

[27] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pages 785–799. PMLR, 2023.

[28] Ankit Goyal, Jie Xu, Yijie Guo, Valts Blukis, Yu-Wei Chao, and Dieter Fox. Rvt: Robotic view transformer for 3d object manipulation. In *Conference on Robot Learning*, pages 694–710. PMLR, 2023.

[29] Quan Vuong, Sergey Levine, Homer Rich Walke, Karl Pertsch, Anikait Singh, Ria Doshi, Charles Xu, Jianlan Luo, Liam Tan, Dhruv Shah, et al. Open x-embodiment: Robotic learning datasets and rt-x models. In *Towards Generalist Robots: Learning Paradigms for Scalable Skill Acquisition@ CoRL2023*, 2023.

[30] Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3m: A universal visual representation for robot manipulation. *arXiv preprint arXiv:2203.12601*, 2022.

[31] Sudeep Dasari, Mohan Kumar Srirama, Unnat Jain, and Abhinav Gupta. An unbiased look at datasets for visuo-motor pre-training. In *Conference on Robot Learning*, pages 1183–1198. PMLR, 2023.

[32] Jiaming Zhou, Teli Ma, Kun-Yu Lin, Zifan Wang, Ronghe Qiu, and Junwei Liang. Mitigating the human-robot domain discrepancy in visual pre-training for robotic manipulation. *arXiv preprint arXiv:2406.14235*, 2024.

[33] Ankit Goyal, Valts Blukis, Jie Xu, Yijie Guo, Yu-Wei Chao, and Dieter Fox. Rvt-2: Learning precise manipulation from few demonstrations. *arXiv preprint arXiv:2406.08545*, 2024.

[34] Haoquan Fang, Markus Grotz, Wilbert Pumacay, Yi Ru Wang, Dieter Fox, Ranjay Krishna, and Jiafei Duan. Sam2act: Integrating visual foundation model with a memory architecture for robotic manipulation. *arXiv preprint arXiv:2501.18564*, 2025.

[35] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners, 2020.

[36] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Tianyu Liu, et al. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

[37] Noam Wies, Yoav Levine, and Amnon Shashua. The learnability of in-context learning. *Advances in Neural Information Processing Systems*, 36:36637–36651, 2023.

[38] Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, et al. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation. *arXiv preprint arXiv:2411.19650*, 2024.

[39] Jiaming Liu, Mengzhen Liu, Zhenyu Wang, Pengju An, Xiaoqi Li, Kaichen Zhou, Senqiao Yang, Renrui Zhang, Yandong Guo, and Shanghang Zhang. Robomamba: Efficient vision-language-action model for robotic reasoning and manipulation. *Advances in Neural Information Processing Systems*, 37:40085–40110, 2024.

[40] Jinming Li, Yichen Zhu, Zhibin Tang, Junjie Wen, Minjie Zhu, Xiaoyu Liu, Chengmeng Li, Ran Cheng, Yaxin Peng, and Feifei Feng. Improving vision-language-action models via chain-of-affordance. *arXiv preprint arXiv:2412.20451*, 2024.

[41] Junjie Wen, Yichen Zhu, Jinming Li, Minjie Zhu, Zhibin Tang, Kun Wu, Zhiyuan Xu, Ning Liu, Ran Cheng, Chaomin Shen, et al. Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation. *IEEE Robotics and Automation Letters*, 2025.

[42] Delin Qu, Haoming Song, Qizhi Chen, Yuanqi Yao, Xinyi Ye, Yan Ding, Zhigang Wang, JiaYuan Gu, Bin Zhao, Dong Wang, et al. Spatialvla: Exploring spatial representations for visual-language-action model. *arXiv preprint arXiv:2501.15830*, 2025.

[43] Zhongyi Zhou, Yichen Zhu, Minjie Zhu, Junjie Wen, Ning Liu, Zhiyuan Xu, Weibin Meng, Ran Cheng, Yaxin Peng, Chaomin Shen, et al. Chatvla: Unified multimodal understanding and robot control with vision-language-action model. *arXiv preprint arXiv:2502.14420*, 2025.

[44] Karl Pertsch, Kyle Stachowicz, Brian Ichter, Danny Driess, Suraj Nair, Quan Vuong, Oier Mees, Chelsea Finn, and Sergey Levine. Fast: Efficient action tokenization for vision-language-action models. *arXiv preprint arXiv:2501.09747*, 2025.

[45] Yifan Zhong, Xuchuan Huang, Ruochong Li, Ceyao Zhang, Yitao Liang, Yaodong Yang, and Yuanpei Chen. Dexgraspvla: A vision-language-action framework towards general dexterous grasping. *arXiv preprint arXiv:2502.20900*, 2025.

[46] Jian-Jian Jiang, Xiao-Ming Wu, Yi-Xiang He, Ling-An Zeng, Yi-Lin Wei, Dandan Zhang, and Wei-Shi Zheng. Rethinking bimanual robotic manipulation: Learning with decoupled interaction framework. In *International Conference on Computer Vision*, 2025.

[47] Jian-Jian Jiang, Xiao-Ming Wu, Zibo Chen, Yi-Lin Wei, and Wei-Shi Zheng. igrasp: An interactive 2d-3d framework for 6-dof grasp detection. In *International Conference on Pattern Recognition*, 2024.

[48] Zhi Hou, Tianyi Zhang, Yuwen Xiong, Haonan Duan, Hengjun Pu, Ronglei Tong, Chengyang Zhao, Xizhou Zhu, Yu Qiao, Jifeng Dai, et al. Dita: Scaling diffusion transformer for generalist vision-language-action policy. *arXiv preprint arXiv:2503.19757*, 2025.

[49] Chengmeng Li, Junjie Wen, Yan Peng, Yaxin Peng, Feifei Feng, and Yichen Zhu. Pointvla: Injecting the 3d world into vision-language-action models. *arXiv preprint arXiv:2503.07511*, 2025.

[50] Yangtao Chen, Zixuan Chen, Junhui Yin, Jing Huo, Pinzhuo Tian, Jieqi Shi, and Yang Gao. Gravmad: Grounded spatial value maps guided action diffusion for generalized 3d manipulation. *arXiv preprint arXiv:2409.20154*, 2024.

[51] FIGURE. Helix: A vision-language-action model for generalist humanoid control, 2025.

[52] Fangchen Liu, Kuan Fang, Pieter Abbeel, and Sergey Levine. Moka: Open-vocabulary robotic manipulation through mark-based visual prompting. In *First Workshop on Vision-Language Models for Navigation and Manipulation at ICRA 2024*, 2024.

[53] Haoxu Huang, Fanqi Lin, Yingdong Hu, Shengjie Wang, and Yang Gao. Copa: General robotic manipulation through spatial constraints of parts with foundation models. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9488–9495. IEEE, 2024.

[54] Yuelei Li, Ge Yan, Annabella Macaluso, Mazeyu Ji, Xueyan Zou, and Xiaolong Wang. Integrating lmm planners and 3d skill policies for generalizable manipulation. *arXiv preprint arXiv:2501.18733*, 2025.

[55] Xiaoqi Li, Mingxu Zhang, Yiran Geng, Haoran Geng, Yuxing Long, Yan Shen, Renrui Zhang, Jiaming Liu, and Hao Dong. Manipllm: Embodied multimodal large language model for object-centric robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18061–18070, 2024.

[56] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. *arXiv preprint arXiv:2409.01652*, 2024.

[57] Mingjie Pan, Jiyao Zhang, Tianshu Wu, Yinghao Zhao, Wenlong Gao, and Hao Dong. Omnimanip: Towards general robotic manipulation via object-centric interaction primitives as spatial constraints. *arXiv preprint arXiv:2501.03841*, 2025.

[58] Jiafei Duan, Wentao Yuan, Wilbert Pumacay, Yi Ru Wang, Kiana Ehsani, Dieter Fox, and Ranjay Krishna. Manipulate-anything: Automating real-world robots using vision-language models. *arXiv preprint arXiv:2406.18915*, 2024.

[59] Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre He, Vivek Myers, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023.

[60] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems (NeurIPS)*, 30, 2017.

[61] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, et al. Training language models to follow instructions with human feedback, 2022.

[62] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners, 2019.

[63] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models, 2023.

 https://doi.org/10.52202/085713-4674

[64] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2022.

[65] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

[66] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[67] Norman Di Palo and Edward Johns. Keypoint action tokens enable in-context imitation learning in robotics. *arXiv preprint arXiv:2403.19578*, 2024.

[68] Jiaqiang Ye Zhu, Carla Gomez Cano, David Vazquez Bermudez, and Michal Drozdzal. Incoro: In-context learning for robotics control with feedback loops. *arXiv preprint arXiv:2402.05188*, 2024.

[69] Vitalis Vosylius and Edward Johns. Instant policy: In-context imitation learning via graph diffusion. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025.

[70] Yecheng Jason Ma, Joey Hejna, Chuyuan Fu, Dhruv Shah, Jacky Liang, Zhuo Xu, Sean Kirmani, Peng Xu, Danny Driess, Ted Xiao, et al. Vision language models are in-context value learners. In *The Thirteenth International Conference on Learning Representations*, 2024.

[71] Yida Yin, Zekai Wang, Yuvan Sharma, Dantong Niu, Trevor Darrell, and Roei Herzig. In-context learning enables robot action prediction in llms. *arXiv preprint arXiv:2410.12782*, 2024.

[72] Letian Fu, Huang Huang, Gaurav Datta, Lawrence Yunliang Chen, William Chung-Ho Panitch, Fangchen Liu, Hui Li, and Ken Goldberg. In-context imitation learning via next-token prediction. *arXiv preprint arXiv:2408.15980*, 2024.

[73] Ohad Rubin, Jonathan Herzig, and Jonathan Berant. Learning to retrieve prompts for in-context learning. *arXiv preprint arXiv:2112.08633*, 2021.

[74] Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*, 2022.

[75] Yuanhan Zhang, Kaiyang Zhou, and Ziwei Liu. What makes good examples for visual in-context learning? *Advances in Neural Information Processing Systems*, 36:17773–17794, 2023.

[76] Yucheng Zhou, Xiang Li, Qianning Wang, and Jianbing Shen. Visual in-context learning for large vision-language models. *arXiv preprint arXiv:2402.11574*, 2024.

[77] Dantong Niu, Yuvan Sharma, Giscard Biamby, Jerome Quenum, Yutong Bai, Baifeng Shi, Trevor Darrell, and Roei Herzig. Llarva: Vision-action instruction tuning enhances robot learning. *arXiv preprint arXiv:2406.11815*, 2024.

[78] Teli Ma, Jiaming Zhou, Zifan Wang, Ronghe Qiu, and Junwei Liang. Contrastive imitation learning for language-guided multi-task robotic manipulation. *arXiv preprint arXiv:2406.09738*, 2024.

[79] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for robotics, games and machine learning. `http://pybullet.org`, 2016–2021.

[80] Andy Zeng, Pete Florence, Jonathan Tompson, Stefan Welker, Jonathan Chien, Maria Attarian, Travis Armstrong, Ivan Krasin, Dan Duong, Vikas Sindhwani, et al. Transporter networks: Rearranging the visual world for robotic manipulation. In *Conference on Robot Learning*, pages 726–747. PMLR, 2021.

[81] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. In *Fortieth International Conference on Machine Learning*, 2023.

[82] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 18392–18402, 2023.

[83] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024.

[84] Stephen James and Andrew J Davison. Q-attention: Enabling efficient learning for vision-based robotic manipulation. *IEEE Robotics and Automation Letters*, 7(2):1612–1619, 2022.

[85] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[86] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[87] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.

[88] Zheng Cai, Maosong Cao, Haojiong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui Chen, Zhi Chen, Pei Chu, et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*, 2024.

[89] Matthias Minderer, Alexey Gritsenko, and Neil Houlsby. Scaling open-vocabulary object detection. *Advances in Neural Information Processing Systems*, 36:72983–73007, 2023.

[90] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026, 2023.

[91] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021.

[92] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[93] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

 https://doi.org/10.52202/085713-4674

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction accurately reflect the main contributions and scope of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: In the Discussions Section, we discuss the limitations of this work, and demonstrate some promising directions for future work.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We disclose all the information needed to reproduce the main experimental results in the Method, Experiment, and Appendix sections. We will release the code upon acceptance of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

https://doi.org/10.52202/085713-4674

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We commit to releasing the code, data, and models upon acceptance of the paper.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide all the training and test details necessary to understand the results in the Method, Experiment, and Appendix sections.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We show the standard deviation of the tasks' success rates.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The information on the computer resources is provided in Experiment section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We adhere to the NeurIPS Code of Ethics, since the paper does not include any content or practices that violate ethical guidelines

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the potential societal impacts of the paper in the Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

https://doi.org/10.52202/085713-4674

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper does not involve data or models that have a high risk for misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The paper properly credits the creators or original owners of all used assets, and properly respects the license and terms of use.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Our new benchmark is built upon the existing RLBench environment. We provide details about our benchmark in the Benchmark section and Appendix. We will release the data for the benchmark upon acceptance.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [Yes]

    Justification: The task categorization process is conducted by GPT4o. The results are verified by three human experts. The verification only takes about 10 minutes. According to our country's policy, human experts have received decent remuneration.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: There are no risks to human subjects, and no need for Institutional Review Board (IRB) approvals.

    Guidelines:

https://doi.org/10.52202/085713-4674

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

    Answer: [NA]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

Appendix

A1 AGNOSTOS Benchmark

A1.1 Task Selection and Categorization

The AGNOSTOS benchmark is built upon the RLBench simulated environment, which includes 100 pre-defined manipulation tasks. Existing robotic manipulation studies typically use a subset of 18 RLBench tasks for training and close-set testing. Figure A1 illustrates these 18 tasks, which we refer to as seen tasks. To rigorously assess the generalization capabilities of vision-language-action (VLA) models on novel tasks, we select 23 unseen tasks from RLBench with distinct semantics compared to the seen tasks. Figure 1 visualizes these 23 unseen tasks. In the Supplementary Materials, we provide video examples of all seen and unseen tasks.

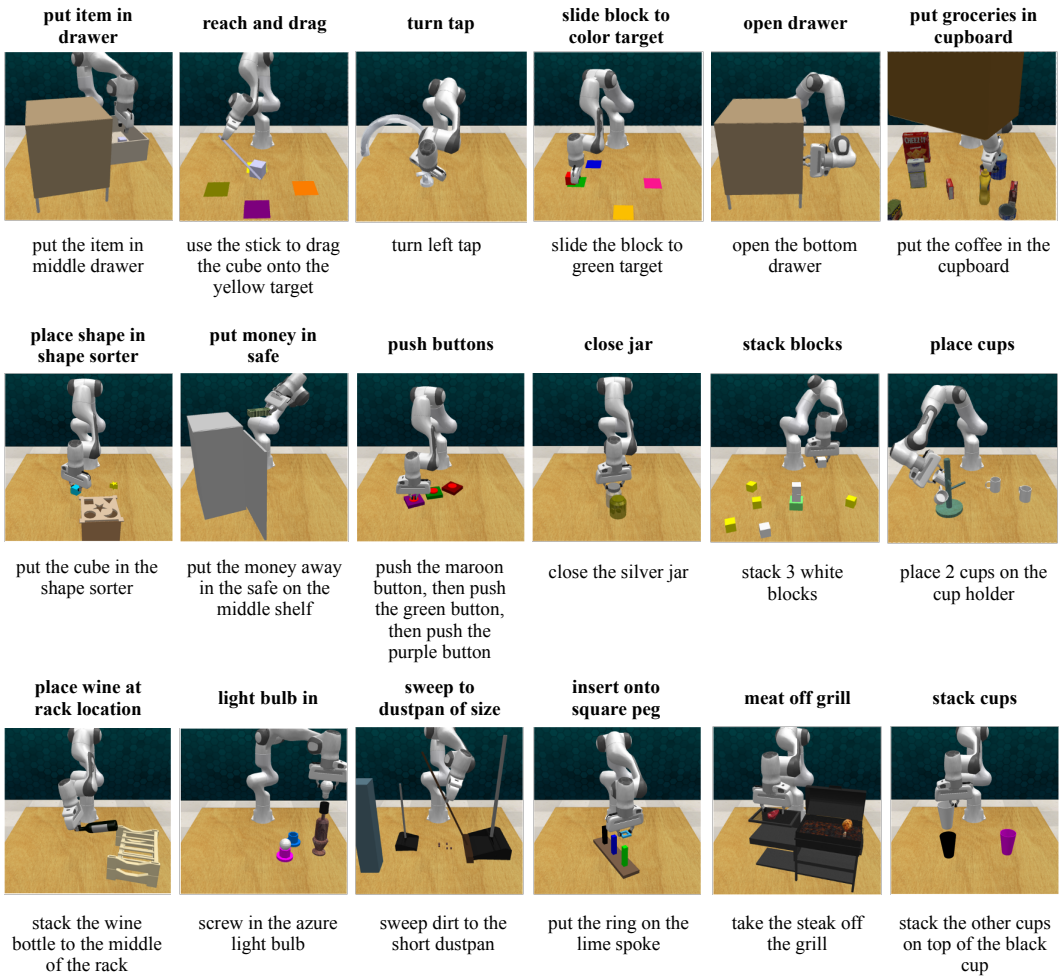


Figure A1: **Examples of 18 widely used training (seen) tasks on RLBench.**

The AGNOSTOS benchmark categorizes the 23 unseen tasks into two difficulty levels. The Level-1 testing includes 13 unseen tasks, which have similar semantics to seen tasks in either objects or motions. The Level-2 testing comprises 10 unseen tasks with entirely novel semantics, distinct from the seen tasks. The task categorization is performed by GPT-4o based on the semantics similarities between seen and unseen task descriptions. The categorization results are verified by three human experts in the field of robotics. In Table A1 we list all 23 unseen tasks, where the text in each "[]" symbol is an abbreviation for the corresponding task. Under the full task name of each unseen task, we show the seen task whose semantics are the most similar to the unseen task.

https://doi.org/10.52202/085713-4674

Table A1: 23 unseen tasks in our AGNOSTOS benchmark. The text in each "[]" symbol is an abbreviation for the corresponding task. Under the full task name of each unseen task, we show its most similar task among the 18 seen tasks.

[Toilet] put toilet roll on stand (place wine at rack location)	[Knife] put knife on chopping board (put item in drawer)	[Fridge] close fridge (close jar)
[Microwave] close microwave (close jar)	[Laptop] close laptop lid (close jar)	[Phone] phone on base (place wine at rack location)
[Seat] toilet seat down (reach and drag)	[LampOff] lamp off (push buttons)	[LampOn] lamp on (push buttons)
[Book] put books on bookshelf (put groceries in cupboard)	[Umbrella] put umbrella in umbrella stand (put item in drawer)	[Charger] unplug charger (no similar seen task)
[Grill] open grill (open drawer)	[Bin] put rubbish in bin (put item in drawer)	[USB] take usb out of computer (no similar seen task)
[Lid] take lid off saucepan (no similar seen task)	[Plate] take plate off colored dish rack (no similar seen task)	[Ball] basketball in hoop (no similar seen task)
[Scoop] scoop with spatula (no similar seen task)	[Rope] straighten rope (no similar seen task)	[Oven] turn oven on (no similar seen task)
[Buzz] beat the buzz (no similar seen task)	[Plants] water plants (no similar seen task)	

A1.2 Evaluations of existing VLA models

Due to embodiment gaps (e.g., differences in robots, action spaces, and camera configurations), existing learning-based VLA models cannot effectively generalize to tasks in unseen embodiments. To address this, we fine-tune VLA models using data from RLBench's 18 seen tasks and evaluate their generalization on the 23 unseen tasks.

A1.2.1 Human-video pre-trained VLA models

Existing VLA models pre-trained on large-scale human action videos, such as R3M [30], D4R [31], R3M-Align [32], and D4R-Align [32], claim to learn generalizable representations for robotic manipulation. HR-Align [32] adapts these models to RLBench by fine-tuning them on the 18 seen tasks, achieving strong performance on these tasks. We evaluate these adapted models provided by [32] on our 23 unseen tasks to assess their generalization.

A1.2.2 In-domain data trained VLA models

For the VLA models trained on the RLBench's 18 training (seen) tasks (i.e., in-domain training), including PerAct [27], RVT [28], RVT2 [33], Sigma-Agent [78], and Instant Policy [69], we test their officially released models on our 23 unseen tasks. These VLA models serve as strong baselines on our benchmark, since they have sophisticated model designs on RLBench data and do not involve the data domain gap between their training tasks and our unseen testing tasks. Instant Policy is a within-task in-context imitation learning method that formulates policy prediction as a graph diffusion process over structured demonstrations and observations. To evaluate its cross-task generalization capability, we evaluate the released model on our 23 unseen tasks in a zero-shot manner, where a randomly sampled seen demonstration is used for in-context prompting.

A1.2.3 Foundation VLA models

The foundation VLA models trained on large-scale robotic data or built upon LLM or VLM models, are expected to have strong generalization capabilities on new tasks. In this work, we evaluate OpenVLA [8], RDT [9], π_0 [1], LLARVA [77], SAM2Act [34], 3D-LOTUS++ [25], and VoxPoser [2]. Below, we elaborate on our fine-tuning details, which follow the official fine-tuning guidelines.

OpenVLA [8]. We fine-tune OpenVLA using 3,600 demonstrations from the 18 seen tasks, with each demonstration comprising a front RGB view (size of 256×256) and the corresponding language instruction. We use a batch size of 16 and apply LoRA fine-tuning with a rank of 32 and a learning rate of 5×10^{-4}. Figure A2 shows the training loss and action accuracy during fine-tuning, indicating rapid convergence within 2,000 steps. We evaluate the model on the 23 unseen tasks every 1,000 steps and select the model with the highest generalization performance.

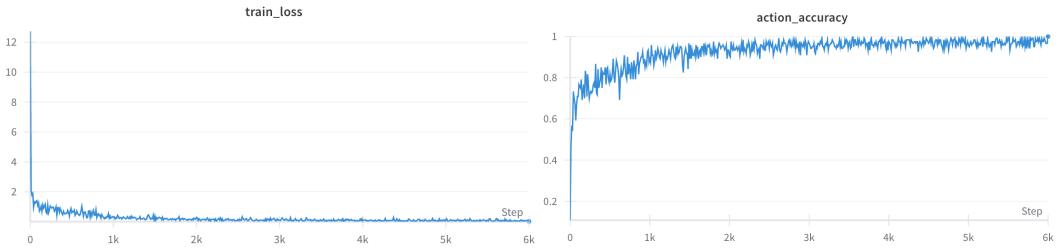


Figure A2: The statistics of the fine-tuning of OpenVLA.

RDT [9]. For the fine-tuning of RDT, we also use the 3,600 seen demonstrations. For each demonstration, RDT takes the front and wrist RGB views as well as the language instruction as inputs, where the image size is 256×256. We use a batch size of 16 and fine-tune the RDT for 400,000 steps by following their official guidance. Figure A3 shows the training loss and the overall average sample MSE loss (a good metric claimed by the authors) during the fine-tuning, which indicates that the RDT converges well. We evaluate the RDT model every 10,000 steps, and report the highest generalization performance on our 23 unseen tasks.

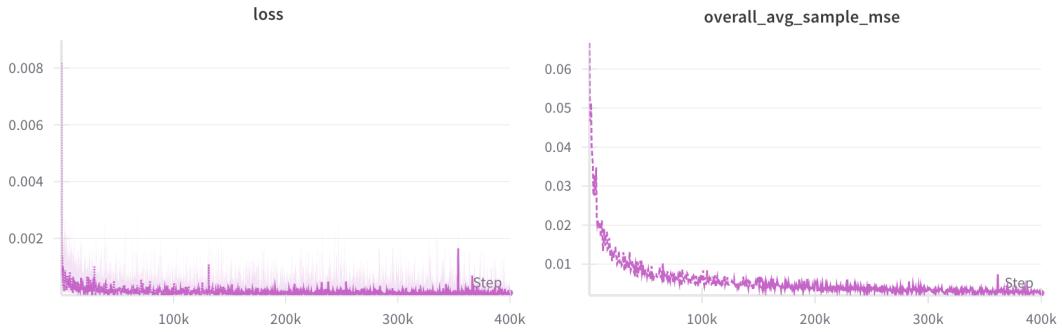


Figure A3: The statistics of the fine-tuning of RDT.

π_0 [1]. We fine-tune π_0 with LoRA using 3,600 demonstrations, each including front, wrist, and overhead RGB views (size of 256×256) and the language instruction. Following the official protocol, we set the batch size to 64 and train for 100,000 steps. Figure A4 shows the training loss during fine-tuning. We evaluate the model every 10,000 steps and report the best generalization performance on the 23 unseen tasks.

LLARVA [77]. LLARVA is a model trained with instruction tuning on the Open X-Embodiment dataset [29], which unifies various robotic tasks and environments. The authors further adapt LLARVA to RLBench to mitigate the embodiment gap. Therefore, we use the officially released model to evaluate the generalization performance on our 23 unseen tasks.

3D-LOTUS/3D-LOTUS++ [25]. We directly evaluate the generalization performance of 3D-LOTUS using the pretrained model released by the authors. For 3D-LOTUS++, we follow the recommended configuration and adopt LLaMA3-8B as the LLM, and the base models of OwlViT v2 [89] and

https://doi.org/10.52202/085713-4674

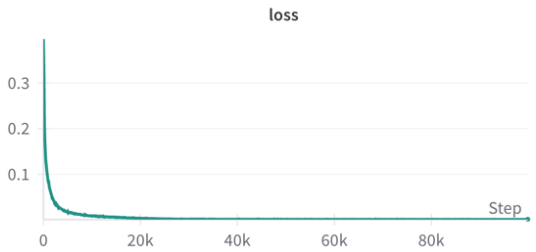


Figure A4: The statistics of the fine-tuning of π_0.

SAM [90] as the VLM components. These modules enable 3D-LOTUS++ to effectively decompose high-level instructions and ground object references in 3D space. We evaluate the model by following the official pipeline on our 23 unseen tasks.

SAM2Act [34]. SAM2Act builds upon the RVT-2 framework and introduces a novel module that integrates semantic masks from SAM [90] into the action learning pipeline. The SAM2Act model is trained end-to-end on RLBench's 18 training tasks. In our experiment, we use the checkpoint released by the authors and evaluate its performance directly on our 23 unseen tasks.

VoxPoser [2]. VoxPoser uses a large language model to generate 3D affordance maps for motion planning. We adopt the open-sourced Qwen2.5-72B-Instruct variant as the LLM and modify the original single-task pipeline to support batched evaluation on custom datasets. During evaluation, object positions are directly obtained from the simulator as ground-truth. We evaluate the model in a zero-shot manner on our 23 unseen tasks.

A1.3 Performance on Seen Tasks.

In Table A2, we evaluate the performance of RVT2, R3M-Align, OpenVLA, π_0 and our X-ICM (72B) on 18 in-domain (seen) tasks, alongside their performance on 23 unseen tasks. These results demonstrate that all methods achieve high success rates on in-domain tasks after fine-tuning on seen task demonstrations, highlighting the challenge of cross-task zero-shot generalization.

For X-ICM, we use 18 within-task demonstrations to construct in-context prompts for in-domain tasks, yielding a 60.4% average success rate, which is substantially higher than its 30.1% on unseen tasks. However, our X-ICM's performance on in-domain tasks is lower than that of learning-based VLA methods like (70.5%). This gap is expected, as X-ICM relies solely on the LLM's in-context learning capability without fine-tuning.

	18 seen/in-domain avg (std)	23 unseen avg (std)
RVT2	82.3 (0.7)	10.3 (0.6)
R3M-Align	59.2 (0.8)	12.2 (0.3)
OpenVLA	63.1 (1.4)	13.6 (0.8)
π_0	70.5 (0.9)	17.5 (0.4)
X-ICM (72B)	60.4 (0.7)	30.1 (1.0)

Table A2: Performance on 18 seen tasks and 23 unseen tasks.

A2 More Analysis

A2.1 Dynamics-guided Sample Selection module

Different In-context Sample Selection Baselines. Our Dynamics-guided Sample Selection module is designed to capture the dynamic semantics critical for cross-task generalization by modeling the transformation from an initial to a final state in manipulation tasks, as detailed in Section 4.2. Unlike standard methods like Video-CLIP [91], which rely on video sequences and are thus incompatible with our setting—where only a single initial frame is available during testing unseen tasks—our diffusion-based model predicts future observations from this single frame. This enables effective modeling of task temporal dynamics essential for generalization.

https://doi.org/10.52202/085713-4674

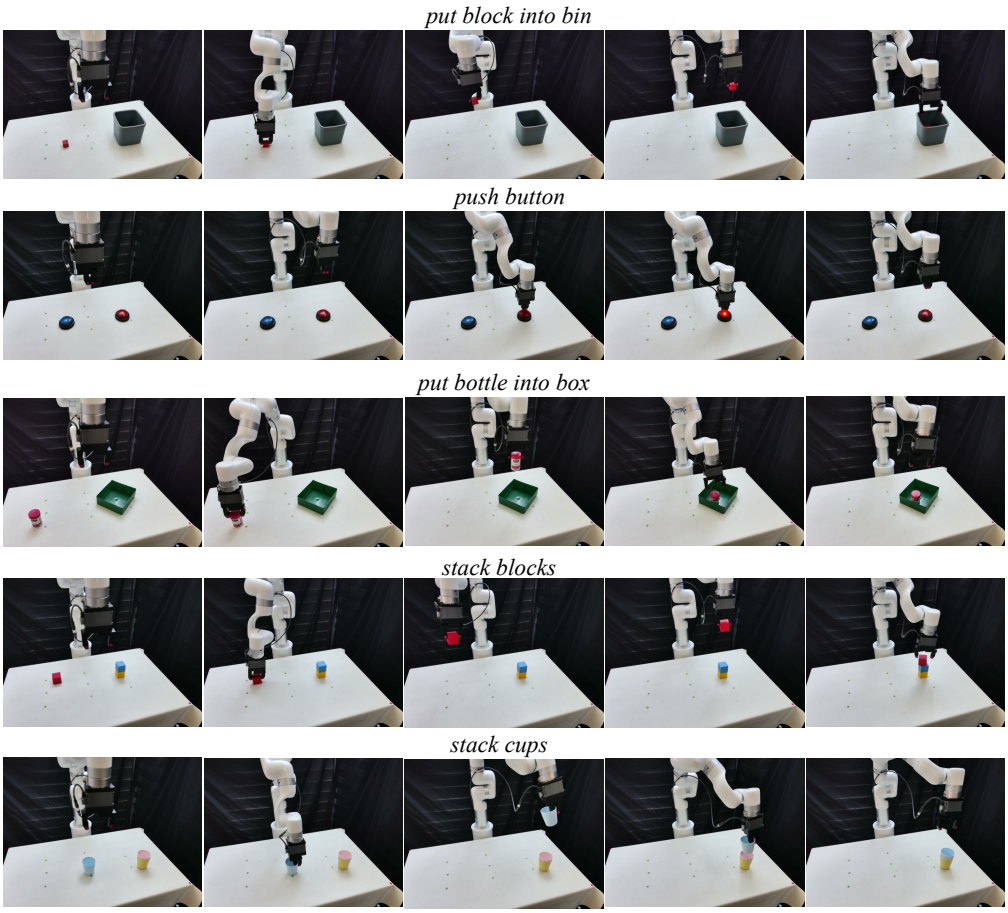


Figure A5: **Demonstration examples of five real-world tasks.** We visualize one demonstration for each task, where we show the key-observations captured by the third-view Orbbec camera.

To validate the necessity of our approach, we compared it with baselines using CLIP [92] and DINOv2 [93] features for sample selection, alongside a random selection strategy. The results in Table A3 demonstrate that our method achieves higher average success rates and lower variance compared to these general-purpose features, as it is trained specifically on manipulation dynamics, unlike the broader datasets used for CLIP and DINOv2.

	Level-1 avg (std)	Level-2 avg (std)	All avg (std)
Random	30.7 (4.7)	18.0 (2.2)	25.2 (3.2)
CLIP feature	32.1 (3.5)	18.9 (2.6)	26.4 (2.9)
DINOV2 feature	33.7 (3.1)	19.3 (2.0)	27.4 (2.5)
Dynamics Diffusion (Ours)	37.6 (1.4)	20.3 (1.7)	30.1 (1.0)

Table A3: Comparisons between different feature representations.

Qualitative results of the dynamics diffusion model. To facilitate the cross-task sample selection, we train a dynamics diffusion model to encode the dynamic representations within each robot demonstration. The similarities of the captured dynamic representations across seen and unseen demonstrations are effective in identifying relevant demonstrations, thus guiding the cross-task sample selection process. For each demonstration, the dynamics diffusion model takes the initial observation and the corresponding language description as inputs, aiming to predict the final visual observation at the completion of the demonstration. Figure A6 showcases qualitative predictions from the trained dynamics diffusion model.

https://doi.org/10.52202/085713-4674

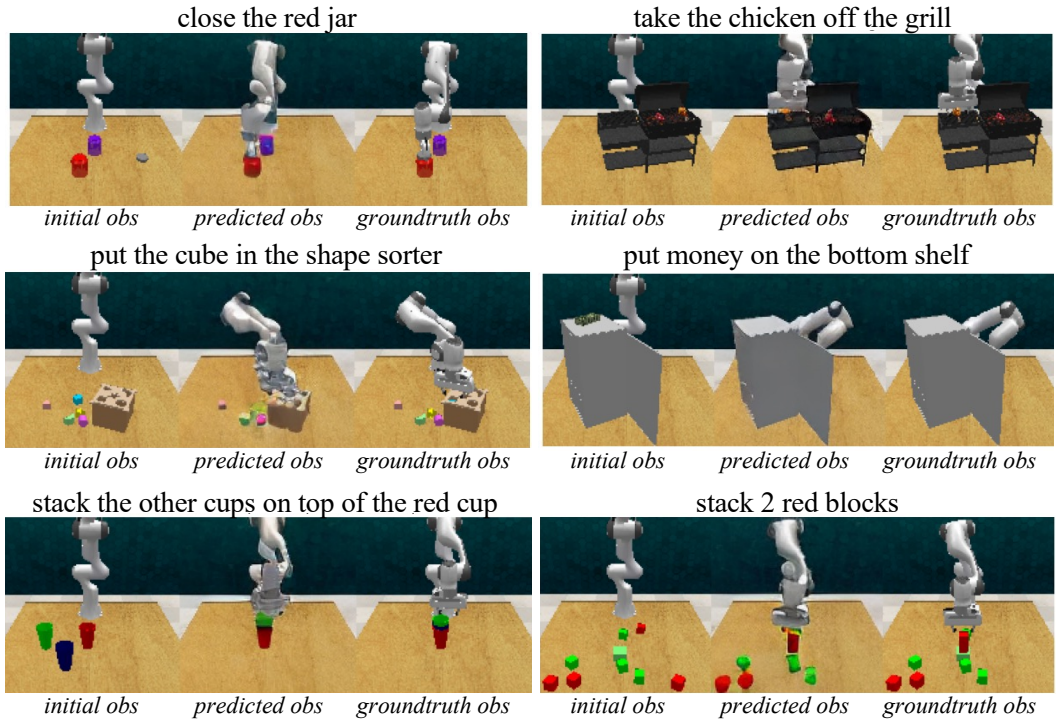


Figure A6: Qualitative results of the trained dynamics diffusion model. For each demonstration, the dynamics diffusion model uses the initial visual observation and language description as conditions and predicts the final visual observation upon task completion.

Using different dynamic features. Our dynamics diffusion model adopts the architecture of InstructPix2Pix [82]. After training, we extract multi-modal features from each demonstration to serve as dynamic features for sample selection. In this work, we use the combination of the textual feature of the language description (i.e., $feat_{lang}$) and the predicted latent feature of the target observation (i.e., $feat_{vis.out}$) as the dynamic features. Alternatively, we can use the latent feature of the initial observation (i.e., $feat_{vis.in}$), and try different combinations to form the dynamic features. Table A4 presents the generalization performance on the 23 unseen tasks for different feature combinations. The predicted latent feature of the target observation (i.e., $feat_{vis.out}$) proved most effective, validating the ability of our dynamics diffusion model to capture generalizable dynamics. Additionally, we find that the latent feature of the initial observation (i.e., $feat_{vis.in}$) is also effective, while using the textual feature of the language description (i.e., $feat_{lang}$) does not show benefit.

Table A4: Generalization performance using different dynamic feature combinations. Upon our InstructPix2Pix-based dynamics diffusion model, $feat_{lang}$ denotes the language instruction feature, $feat_{vis.in}$ represents the latent feature of the initial observation, and $feat_{vis.out}$ denotes the predicted latent feature of the target observation.

$feat_{lang}$	$feat_{vis.in}$	$feat_{vis.out}$	Level-1	Level-2	All
×	×	×	30.7 (4.7)	18.0 (2.2)	25.2 (3.2)
×	✓	×	34.9 (2.9)	17.8 (1.4)	27.5 (1.0)
×	×	✓	37.2 (2.4)	19.9 (2.1)	29.7 (2.2)
×	✓	✓	37.1 (0.8)	17.6 (2.0)	28.6 (1.3)
✓	✓	×	36.3 (0.9)	16.4 (0.4)	27.7 (0.7)
✓	×	✓	37.6 (1.4)	20.3 (1.7)	**30.1 (1.0)**
✓	✓	✓	38.6 (1.6)	16.7 (2.4)	29.0 (1.6)

A2.2 Different model sizes of LLMs.

We investigated the impact of model scale on performance by evaluating the X-ICM model at various sizes of Qwen2.5-Instruct: 7B, 14B, and 72B parameters. As shown in Table A5, larger models

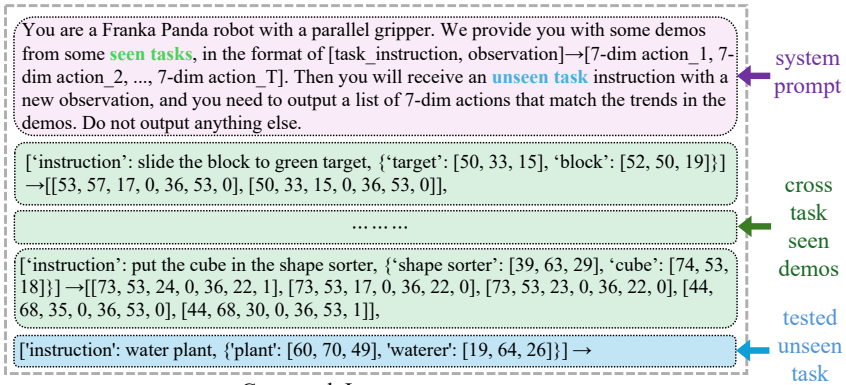
generally tend to yield better performance. Performance improved noticeably as the model size increased from 7B to 14B. The performance gain diminished when scaling models to 72B. This suggests that while increasing model size is generally beneficial, the returns may diminish.

Table A5: Effects of different model sizes.

Models	Level-1	Level-2	All
X-ICM (7B)	28.6 (1.9)	16.9 (1.3)	23.5 (1.6)
X-ICM (14B)	38.6 (0.5)	18.1 (2.3)	29.7 (0.7)
X-ICM (72B)	37.6 (1.4)	20.3 (1.7)	30.1 (1.0)

A2.3 Cross-task In-context Prompt

Figure A7 shows an example of the construction process of the cross-task in-context prompt. The prompt is concatenated by the system prompt, textualized cross-task seen demonstrations that are selected, and the textualized tested unseen task.



Cross-task In-context prompt

Figure A7: An example of the construction of our cross-task in-context prompt, which is concatenated by the system prompt, textualized cross-task seen demonstrations that are selected, and the textualized tested unseen task.

A3 Real-world Manipulation Experiments

In Figure A5, we visualize the collected demonstrations for five real-world tasks. For each task, we randomly select one demonstration for visualization. And we visualize the visual observations when the key-actions occur in each demonstration. In the Supplementary Materials, we showcase some testing videos that include both successful and failed cases.

A4 Broader Impacts Statements

There are no ethical issues involved in this paper. This work proposes a cross-task generalization benchmark to evaluate the generalization capabilities of existing vision-language-action models. The method proposed in this work is purely algorithmic. Therefore, the proposed benchmark and method does not change the societal impacts of robotic manipulation.

https://doi.org/10.52202/085713-4674