Embodiment Transfer Learning for Vision-Language-Action Models

Chengmeng Li[1] **Yaxin Peng**[1,†]
[1]Shanghai University

Abstract

Vision-language-action (VLA) models have significantly advanced robotic learning, enabling training on large-scale, cross-embodiment data and fine-tuning for specific robots. However, state-of-the-art autoregressive VLAs struggle with multi-robot collaboration. We introduce embodiment transfer learning, denoted as **ET-VLA**, a novel framework for efficient and effective transfer of pre-trained VLAs to multi-robot. ET-VLA's core is Synthetic Continued Pretraining (SCP), which uses synthetically generated data to warm up the model for the new embodiment, bypassing the need for real human demonstrations and reducing data collection costs. SCP enables the model to learn correct actions and precise action token numbers. Following SCP, the model is fine-tuned on target embodiment data. To further enhance the model performance on multi-embodiment, we present the Embodied Graph-of-Thought technique, a novel approach that formulates each sub-task as a node, that allows the VLA model to distinguish the functionalities and roles of each embodiment during task execution. Our work considers bimanual robots, a simple version of multi-robot to verify our approaches. We validate the effectiveness of our method on both simulation benchmarks and real robots covering three different bimanual embodiments. In particular, our proposed ET-VLA can outperform OpenVLA on six real-world tasks over 53.2%. We will open-source all codes to support the community in advancing VLA models for robot learning. The project is available at https://et-vla.github.io

1. Introduction

Autoregressive models, which generate sequences of discrete tokens by iteratively predicting the next element conditioned on prior outputs, serve as the foundation for modern large language models. This sequential prediction paradigm has revolutionized robotics, enabling Vision-Language-Action (VLA) models such as RT-2 (Brohan et al., 2023) and OpenVLA (Kim et al.) to map sensory inputs to robotic actions through discretized token predictions. While these models excel in unimanual manipulation tasks, their performance degrades significantly when applied to multi-robot systems. Sometimes they even fail to produce structurally valid action sequences (e.g., generating incorrect numbers of tokens). This raises two critical questions: (1) Why do current VLA models struggle to adapt to multi-robots? (2) How can we enhance their robustness and generalization for deployment on novel multi-robot robotic platforms?

Regarding the first question, we identify a dependency of VLAs on pretraining data. These models rely on vast datasets to learn precise token-to-action mappings, yet existing open-source robotics datasets (e.g., OXE (O'Neill et al., 2023b), Droid (Khazatsky et al., 2024) focus exclusively on unimanual robot systems. Consequently, VLAs encounter a distribution shift when applied to multi-robots, lacking the necessary priors to interpret action tokens for unseen embodiments. To address these limitations, we propose Synthetic Continued Pretraining (SCP). In particular, SCP generates synthetic multi-robot data, enabling cost-effective embodied transfer learning without reliance on expensive teleoperated datasets. Rather than directly learning the mapping between observations and robot actions, SCP focuses on establishing the mapping between the embodiment and action sequence. To further enhance VLA's performance on multi-robot, we propose Embodied Graph-of-Thought (EGoT). It creates an actionable graph that meticulously allocates tasks to each robot, ensuring they act in accordance with the designated task sequence. By combining these two techniques, our proposed ET-VLA can do multi-tasking on a multi-robot with quick fine-tuning.

To assess the effectiveness of our approach, we conducted both simulations and real-world experiments, using bimanual robots, a simple version of multi-robots as examples. Our experiments are conducted on three bimanual robots, including bimanual UR5e, bimanual Franka, and bimanual AgileX. These experiments clearly show that directly using OpenVLA in multi-robot scenarios leads to unacceptably low success rates. In contrast, our ET-VLA model, which leverages OpenVLA's architecture and pre-trained weights, demonstrates a substantial improvement after fine-tuning

† Corresponding author.

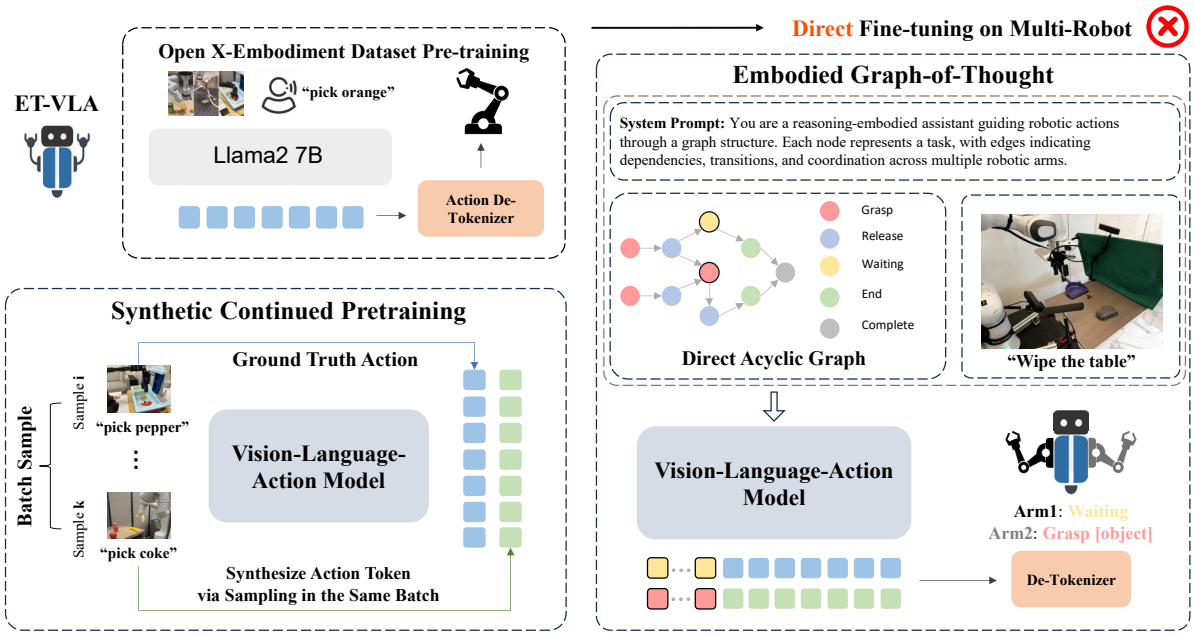

Figure 1. **Overview of ET-VLA:** we propose **Synthetic Continued Pretraining (SCP)** and **Embodied Graph-of-Thought (EGoT)** as key components: **(1) SCP** uses synthetically generated data to warm up the model for the new embodiment, bypassing the need for real human demonstrations and reducing data collection costs. SCP enables the model to learn correct actions and precise action token numbers. **(2) Embodied GoT** formulates each sub-task as a node, that allows the VLA model to distinguish the functionalities and roles of each embodiment during task execution, promoting more effective coordination.

for specific dexterous tasks. For instance, when tested on six tasks involving two UR5e robots, ET-VLA achieved success rates nine times higher than OpenVLA. These results strongly validate the advantages of our ET-VLA model for multi-robot applications.

In summary, our contributions are threefold:

- We provide an in-depth analysis of the limitations of existing autoregressive VLA models in multi-robot multi-task settings.
- We propose ET-VLA, a novel framework incorporating two advanced techniques: Synthetic Continued Pretraining and Embodied Graph-of-Thought, which significantly enhance the performance of VLAs in multi-robot manipulation.
- We extensively evaluate ET-VLA on the real robot and simulation, demonstrating its superior performance compared to state-of-the-art VLA models.

2. Related Works

Vision-language-action models. Vision-language-action models (VLAs) represent a family of work that leverages the power of pre-trained vision-language models as a backbone to understand language as well as observation. Such a method directly fine-tuned large pretrained VLMs (Liu et al., 2025; Zhu et al., 2024f;d; Lu et al., 2024; Awadalla et al., 2023; Laurençon et al., 2023; Liu et al., 2023b;a; Wang et al., 2024a; Chen et al., 2024; Zhu et al., 2021; Jiang et al., 2024; Meng et al., 2023; Zhu et al., 2022; Zhou & Zhu, 2023; Huang et al., 2022) for predicting robot actions (Brohan et al., 2023; Kim et al.; Zawalski et al., 2024; Li et al., 2023; Huang et al.; Wen et al., 2024a; Black et al., 2024; Pertsch et al., 2025; Wu et al., 2025). Existing works can be largely categorized into two different methods, the one uses next-token prediction as the training objective to generate action tokens autoregressively, similar to the ways language models generate text. The other use policy head, i.e., RNN head (Li et al., 2023) or diffusion head (Chi et al., 2023; Zhu et al., 2024c; Wang et al., 2024b; Prasad et al., 2024; Black et al., 2023a;b; Dasari et al., 2024; Lin et al., 2024; Reuss et al., 2024; Zhao et al.; Uehara et al., 2024a;b; Zhu et al., 2024b;e; Wen et al., 2024c; Li et al., 2025a; Zhu et al., 2025a; Li et al., 2025b; 2024; Zhou et al., 2025; Zhu et al., 2025b) to predict robot action. These methods have demonstrated strong performance in both simulations and real-world tasks. However, most are focused on unimanual robot settings. In this work, we show that popular VLAs can fail in multi-robot manipulation tasks. We explore ways to extend these methods to multi-robot without requiring costly re-training.

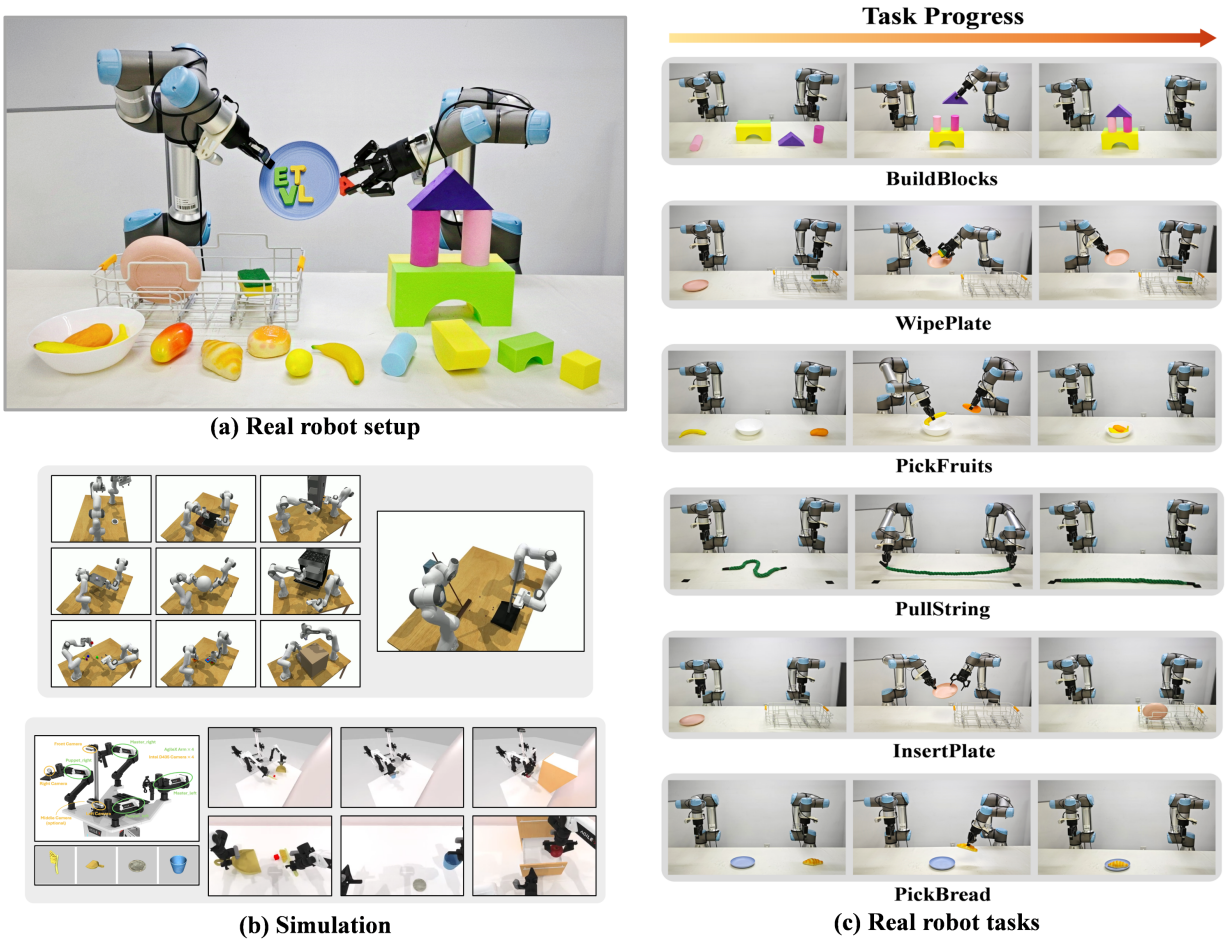

Figure 2. **(a) Real robot and all objects used in our work.** We use two UR5 robot arms equipped with Robotiq grippers and incorporate a diverse set of everyday objects in our manipulation tasks. A RealSense D457 camera is applied to capture visual observations on the top. **(b) Simulation.** We conduct experiments on two simulation benchmarks, the RLBench2 (Grotz et al., 2024) and RoboTwin (Mu et al., 2024). **(c) Real robot tasks.** We designed six collaborative multi-robot tasks for our real-world experiment.

Reasoning in robot learning. A number of works have explored the use of Large Language Models and Vision-Language Models (Liu et al., 2023a) in robotics, particularly for visual representations, object detection (Jiang et al., 2022; Wen et al., 2024b), high-level planning (Rana et al., 2023; Chowdhery et al., 2023; Zhu et al., 2024a; Niu et al., 2024), and code generation (Liang et al., 2023). ECoT proposes a method to predict an object's position, task plans, and current action to enhance action generation. DAG (Gao et al., 2024) leverage directed acyclic graph to help LLM perform better planning for multi-robot. CoT-VLA (Zhao et al., 2024) generates sub-goa for autoregressive VLA to guide action. In this work, our proposed Embodied Graph-of-Thought enables the model to differentiate tasks for two robots, facilitating collaboration between the two embodiments.

3. Methodology

3.1. Preliminary: Vision-Language-Action Models

The architecture of recent Vision-Language-Action (VLA) Models consists of two main components: 1) visual encoders, which map image inputs to feature embeddings and use a projector — typically a few layers of multi-layer perception — to align with the input space of a language model, and 2) a large language model (LLM) backbone that processes all input representations. The model undergoes two primary training phases: a pre-training stage and a fine-tuning stage. During both stages, the model is trained end-to-end using a next-token prediction objective. At inference time, the model accepts both visual observations and language instructions, producing discrete action tokens that control the robot.

This pipeline is widely adopted by various projects, including OpenVLA (Kim et al.), one of the most influential and open-source VLA frameworks. OpenVLA is pre-

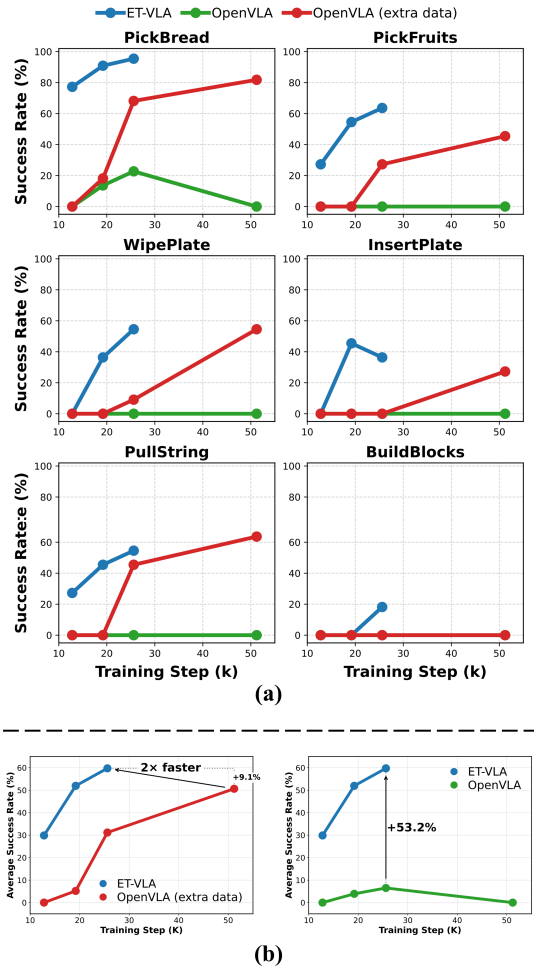

Figure 3. **(a) Learning efficiency.** We show the learning curves of ET-VLA and OpenVLA in 6 real-world tasks. ET-VLA demonstrates a rapid convergence towards high accuracy. **(b) Success rate over six tasks.** We doubled the train data and extended the training duration by a factor of two for OpenVLA, referring to this result as **OpenVLA (extra data)**. Under these conditions, ET-VLA outperforms OpenVLA (extra data) by 9.1% with 2 times less training time.

trained on the Open X-Embodiment (OXE in short) (O'Neill et al., 2023a) datasets and fine-tuned on the Bridge Data V2 (Walke et al., 2023), utilizing the Prismatic-7B VLM (Karamcheti et al., 2024) along with 600 million visual encoder parameters. During testing, OpenVLA generates seven action tokens to control the robot, demonstrating strong performance on standard tasks as well as notable generalization capabilities. This work follows the OpenVLA paradigm, addressing the problem and presenting the solution within this framework.

3.2. Motivation: How does VLA fail on multi-robot

While Vision-Language Action (VLA) models have shown impressive capabilities in learning robot actions and generalizing well in single-arm (unimanual) tasks, it's unclear whether this performance translates directly to more complex systems like multi-robot platforms. This section investigates this question by evaluating the transferability of VLAs to a bimanual UR5e setup, which serves as a simplified multi-robot system, using the state-of-the-art OpenVLA model. Our physical setup consists of two UR5 arms, each equipped with a wrist-mounted Intel RealSense 435i camera. An external Intel RealSense 457 camera is positioned in front of the robots. We collected a dataset of 458 trajectories across six different tasks, detailed in Section 4.1. Experiments denoted with "extra data" utilize an additional dataset of 980 human demonstration trajectories. Specifically, we find two major types of failure:

1) The autoregressive nature of VLAs presents a challenge for generating the full set of action tokens required for multi-robot manipulation (seven tokens per robot in our case). We observed a roughly 50% failure rate in generating the correct number of tokens, significantly impacting performance. To investigate if insufficient training was the root cause, we doubled the training duration and incorporated the extra human demonstration data. This extended training did enable the model to consistently generate the required number of tokens. We believe this issue arises from limitations in the pre-training data. Pre-training is crucial for VLAs, enhancing generative capabilities and accelerating downstream task convergence. OpenVLA, for example, is pre-trained on Open X-Embodiment, a large-scale dataset containing only unimanual robot data. Consequently, the model learns to generate a limited number of action tokens (around six or seven). This unimanual pre-training hinders the model's ability to generate the correct number of tokens when applied to bimanual robots, especially with limited fine-tuning data and iterations.

Even when OpenVLA does generate the correct number of tokens, the average task success rate plateaus on most tasks. Figure 3(a) illustrates this for the BuildBlocks task: even with extra data, OpenVLA's success rate remains at 0. Similar plateaus are visible on other tasks. Doubling the training data and duration (red line) only marginally improves performance, showing a flat learning curve compared to ET-VLA (blue line). Figure 3(b) further demonstrates that simply increasing fine-tuning steps without additional data does not improve OpenVLA's performance. Regarding the average success rate over six tasks, with the same training data and duration, ET-VLA achieves a 53.2% higher success rate than OpenVLA. While adding extra data to OpenVLA does lead to some improvement, it still underperforms significantly compared to ET-VLA. Even with double the training time and data, ET-VLA outperforms OpenVLA by approximately 9%.

2) For tasks requiring multi-robot collaboration, such as opening a bag to place a ball inside, the model frequently

Figure 4. **Example of Embodied Graph-of-Thought (EGoT)**. To facilitate better understanding for our readers, we provide only the simplified version of the prompt and tasks output.

skips intermediate actions, proceeding directly to the subsequent task. For instance, the fine-tuned OpenVLA bypasses the initial step of opening the bag and grabs the tennis ball directly. This limitation suggests that the standard autoregressive VLA lacks effective planning capabilities for varied embodiments and task allocation.

Based on these observations and analyses of autoregressive VLA's multi-robot performance, we propose new methods to enhance autoregressive VLA's effectiveness in manipulation tasks.

3.3. Synthetic Continued Pretraining

In the previous section, we identified that autoregressive VLA can fail to generate the required number of action tokens for controlling a multi-robot system due to insufficient training data. A straightforward approach to address this issue would be to train the model with multi-robot data. However, this method presents significant challenges. First, pre-training demands substantial computational resources. For example, OpenVLA was trained on 64 A100 GPUs over 14 days. Second, multi-robot data is scarce, and the size of available bimanual datasets is several orders of magnitude smaller than that of large datasets such as Open X-Embodiment, which contains over 1 million real robot trajectories, the majority of which are unimanual robotic systems. Therefore, it is crucial to explore alternative methods to enable OpenVLA for multi-robot manipulation with limited data. We propose generating synthetic multi-robot data to train a multi-robot-capable autoregressive VLA.

Our focus is to ensure that VLA models predict the correct number of action tokens. For each data sample x, where $x = \{\text{observation, instruction}\}$, the standard VLA maps the sample x to a set of 7 action tokens $A = \phi(x)$, with $\text{len}(A) = 7$. To address the limitations noted above, we create synthetic robot data. Specifically, for each sample x, we aim for the mapping function to produce 14 action tokens, representing 7 Degrees of Freedom (DoF) for each robot. To generate a valid yet distinct set of tokens for the additional 7 DoF, we introduce a cross-sampling approach.

Assuming a batch size of n samples $X_n = \{x_1, x_2, \ldots, x_n\}$, for a given sample x_i, we randomly select another sample x_j, where $i \neq j$. We then append the predicted action tokens A_j to A_i, resulting in a ground truth of 14 action tokens for a sample x_i. Through this simple operation, the model learns valid actions for robots. Although the image/instruction-to-action mapping may not always be entirely correct, this method ensures that the autoregressively trained VLA learns the token-to-robot mapping. In fact, we used this method to train the VLA model on the BridgeData V2 (Walke et al., 2023) for one epoch. As shown in Table 4 we demonstrate that this approach is essential for VLA, improving OpenVLA's success rate from 6.49% to 37.66%.

3.4. Embodied Graph-of-Thought

In Section 3.2, we discuss the limitations of existing Vision-Language Action (VLA) models in performing an example of multi-robot tasks, identifying two key challenges. While continued pretraining on synthetic data can address the first challenge, the second, more fundamental issue remains: VLAs struggle with task planning, particularly in coordinating actions between multiple robots. This is a crucial requirement for multi-robot, as parallel task execution necessitates careful synchronization, which linear models typically fail to exploit. Without the ability to properly sequence and coordinate actions, robots are unable to perform complex tasks that require simultaneous or interleaved operations.

To address this limitation, we propose Embodied Graph-of-Thought (EGoT), a novel approach that enables VLAs to understand and manage the temporal dependencies inherent in multi-robot tasks. EGoT decomposes complex tasks into an action graph that explicitly represents these dependencies, providing a structured and interpretable framework for planning. This graph allows the robot to reason about the order in which sub-tasks should be executed, considering the constraints and relationships between them.

Table 1. **Quantitative results in real-world experiments.** We report the average success rate across multiple tasks for all models.

Model \ Tasks	Real World Experiment						
	PickBread	PickFruits	WipePlate	InsertPlate	PullString	BuildBlocks	Avg.
Diffusion Policy (Chi et al., 2023)	6/22	4/11	5/11	2/11	5/11	0/11	22/77
TinyVLA (Wen et al., 2024a)	19/22	6/11	0/11	0/11	6/11	1/11	32/77
OpenVLA (Kim et al.)	5/22	0/11	0/11	0/11	0/11	0/11	5/77
ET-VLA	**21/22**	**7/11**	**6/11**	**4/11**	**6/11**	**2/11**	**46/77**

This graph better represents the complex temporal dependencies of multi-robot collaboration. We use two robots as an example here, where this framework can easily extend to more robots operating simultaneously. We define the graph as $G = (V, E, T, N)$, where V denotes the tasks, E represents the dependencies, T categorizes the task types, and N specifies the robot requirements. Each vertex $v_i \in V$ corresponds to a specific sub-task, and each directed edge $e_{ij} = (v_i, v_j) \in E$ indicates that v_i must be completed before v_j . The task types include: 1) Grasp, where tasks involve the engagement of the robot's gripper and the robot will be occupied; 2) Release, for tasks where an object is released from the gripper, often associated with placement or release into a specific location; 3) Waiting, where the robot pauses and waits for the other robot to complete its task. This is designed to facilitate coordination and synchronization between the two robots during collaborative operations; 4) End, indicates that all tasks assigned to the robot have been completed, marking the conclusion of its activity in the task sequence; and 5) Complete, serves as the terminal node in the task graph, signifying the successful execution of all tasks and the end of the workflow. This classification aids in specifying the nature of the task and the number of robots required, thereby enabling a more sophisticated and efficient planning strategy tailored for multi-robot systems.

4. Experiments

In this section, we aim to answer the following question via our experiments:

- How does ET-VLA perform on real robots compared to other state-of-the-art VLA models, such as OpenVLA?
- How does ET-VLA compare with other non-VLA methods, such as Diffusion Policy?
- How important are our proposed techniques — synthetic continued pretraining and embodied graph-of-thought — to overall performance?

4.1. Results on Real Robots

Implementation Details. The experimental setup is depicted in Fig 2. For real-world robotic tasks, we use two UR5 robot arms with Robotiq gripper. We use a single, fixed Realsense D457 camera positioned above the robot. We only use a single RGB image with a resolution of 224 $\times$ 224. Since OpenVLA employs only one extrinsic camera view, we maintained consistency for fair comparison by not using additional wrist cameras. In the synthetic continued pretraining phase, we applied a learning rate of 2e-5, aligning with the rate used in OpenVLA's pretraining phase, and trained for one epoch. During the fine-tuning stage, we used 50–100 demonstrations for each task and trained all tasks together in a mixed setting. Fine-tuning was performed with an initial learning rate of 2e-4, employing cosine learning rate decay over 20 epochs. All models were trained on 16 A100 GPUs with the same total number of training epochs. To ensure unbiased results, we report only the final checkpoint, avoiding selective reporting.

Task Description. As illustrated in Fig 2, we designed six collaborative tasks for our real-world experiment: **PickBread**, **PickFruits**, **WipePlate**, **InsertPlate**, **PullString**, and **BuildBlocks**. Objects were placed randomly within a small range, and we report the average success rate for each method.

1) **PickBread**: The model chooses which robot to pick up bread from one side of the table and place it on a central plate while keeping the other robot stationary, ensuring robot independence.

2) **PickFruits**: A banana and a mango on opposite sides of the table are picked up by different robots and placed sequentially in a bowl, testing grip adaptation and task order.

3) **WipePlate**: One robot holds a sponge, and the other remains stationary as the model cleans a plate with a complex wiping motion, requiring distinct robot coordination.

4) **InsertPlate**: A plate is passed between robots to be inserted into a rack with precise end-effector (EEF) posture control, testing fine motor skills.

5) **PullString**: The model grasps and aligns both ends of a string marked with tape, ignoring its initial shape, and pulls it straight to a marked spot.

6) **BuildBlocks**: robots construct a block house by stacking cylinders and placing a wedge as a roof, demanding

Table 2. **Performance of different 2D methods on various tasks in RLBench2 (Grotz et al., 2024) simulation benchmark.**

Method	(a) box	(b) ball	(c) buttons	(d) plate	(e) drawer	(f) fridge	(g) handover	(h) laptop	(i) rope	(j) dust	(k) tray	(l) handover easy	(m) oven	Avg.
ACT (Fu et al., 2024)	0%	36%	4%	0%	13%	0%	0%	0%	16%	0%	6%	0%	2%	5.9%
OpenVLA (Kim et al.)	0%	0%	6%	0%	10%	0%	0%	0%	0%	0%	0%	0%	0%	1.2%
ET-VLA	**8%**	**56%**	**7%**	**1%**	**19%**	0%	0%	**3%**	**21%**	0%	**11%**	0%	**6%**	**10.2%**

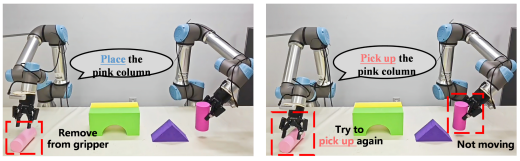

Figure 5. **EGoT is capable of handling complex task sequences.** We deliberately remove the column from the gripper and place it back on the table. we observe the robot try to pick up again. And the model's output transitions to "pick up the pink column".

precision and error-free sequencing.

These tasks evaluate the model's ability to handle independent and coordinated robot movements across varied scenarios.

Real-world experimental results. We evaluated our method against Diffusion Policy (DP) (Chi et al., 2023), TinyVLA (Wen et al., 2024a) and OpenVLA (Kim et al.). For a fair comparison, we used only a single RGB image with a resolution of 224 × 224 to train all baselines, as OpenVLA employs a single extrinsic camera view. Furthermore, our approach integrates language instructions throughout the process, aligning with the input used in all the baselines.

The experimental results are shown in Table 1. ET-VLA achieved the highest success rate across all tasks, surpassing other baselines. Notably, the InsertPlate and BuildBlocks tasks require the model to control the end-effectors to perform relatively complex rotational operations. Additionally, TinyVLA and Diffusion Policy require extra state data to predict actions, whereas ET-VLA relies solely on input images. We also observed that OpenVLA exhibits an extremely low success rate, achieving 0% on five tasks. These results reinforce our earlier observation that autoregressive VLAs, even when trained on large-scale, multi-embodied data, may not effectively transfer their capabilities to multi-robot. While our method, ET-VLA, builds upon OpenVLA, the significant performance improvement clearly demonstrates the importance of our proposed techniques.

Real-world experiment setup In our real-world experiment setup, we use two UR5 robot arms equipped with Robotiq grippers, as is shown in Figure 6. A single, fixed RealSense D457 camera is positioned overhead to capture visual observations. We rely solely on a single RGB image with a resolution of 224 × 224. Since OpenVLA utilizes only one extrinsic camera view, we ensure a fair comparison by not incorporating additional wrist cameras.

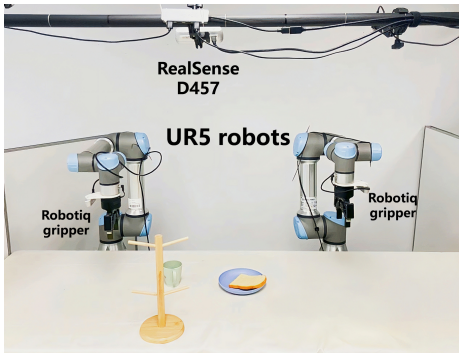

Figure 6. **Real-world experiment setup.**

Table 3. **Experiment on RoboTwin simulation (Mu et al., 2024).** We compare the Diffusion Policy using 50 demonstrations.

Task	Diffusion Policy	OpenVLA	**ET-VLA**
Apple Cabinet Storage	72%	16%	**85%**
Block Handover	28%	7%	**45%**
Blocks Stack (Easy)	2%	0%	**18%**
Container Place	0%	0%	**15%**
Dual Bottles Pick (Easy)	54%	10%	**70%**
Empty Cup Place	20%	0%	**35%**
Shoe Place	9%	0%	**24%**
Block Hammer Beat	0%	0%	**12%**
Block Sweep	95%	24%	**96%**
Blocks Stack (Hard)	0%	0%	**8%**
Diverse Bottles Pick	0%	0%	**14%**
Dual Bottles Pick (Hard)	28%	0&	**42%**
Mug Hanging	0%	0%	**10%**
Shoes Place	0%	0%	**13%**
Average	27.7%	4.1%	**40.1%**

4.2. EGoT is Real-Time Action Interpreter

Our previous section demonstrates that Embodied-GoT (EGoT) effectively improves model performance and generalization. In this section, we show that, in addition to enhancing the model's ability to learn robotic actions, EGoT also provides interpretable language, enabling users to understand the model's current "thought" process.

As shown in Fig. 5, EGoT is capable of handling complex task sequences. For example, in the BuildBlocks task, EGoT follows a task sequence where the left robot picks up the pink column while the right robot pick up another. After the pink column is picked up, the model's output transitions to "place the pink column". If we deliberately remove the column from the left robot and place it back on the table, we observe the robot hesitating and try to pick up the column again. And the model's output transitions to "pick up the pink column". For a typical observer, it may be unclear whether the model is processing this new state

Table 4. **Ablation study for two components in ET-VLA.** We report the average success rate across multiple tasks.

Model \ Tasks	Real World Experiment						
	PickBread	PickFruits	WipePlate	InsertPlate	PullString	BuildBlocks	Avg.
ET-VLA	**21/22**	**7/11**	**6/11**	**4/11**	**6/11**	**2/11**	**46/77** (59.74%)
- Embodied Graph-of-Thought	17/22	5/11	2/11	1/11	4/11	0/11	29/77 (37.66%)
- Synthetic Continued Pretraining	5/22	0/11	0/11	0/11	0/11	0/11	5/77 (6.49%)

or abandoning the task. However, with our approach, EGoT updates to reflect the new state, such as "pick up the pink column", giving observers clear insight into the model's current focus without ambiguity. This interpretability is an emergent property of EGoT and does not require specific training.

4.3. Simulation Experiments

We further conduct experiments on two simulation benchmarks, the RLBench2 (Grotz et al., 2024) and RoboTwin (Mu et al., 2024).

RLBench2. RLBench2 extends the original RLBench benchmark (James et al., 2020) from unimanual to bimanual robot manipulation. This new benchmark comprises 13 core bimanual tasks, totaling 23 unique task variations. RLBench2 retains the core functionality and key properties of RLBench. Standard baselines, including ACT (Fu et al., 2024), RVT (Goyal et al., 2023), and PerACT (Shridhar et al., 2023), are provided. However, as our approach uses only 2D input, we compare solely against ACT, since RVT and PerACT require 3D input. Additionally, we implemented OpenVLA (Kim et al.) using the publicly available model and pre-trained weights. We also initialized our model with these pre-trained OpenVLA weights.

As demonstrated in Table 2, ET-VLA significantly outperforms ACT and OpenVLA in manipulation tasks, achieving a 10.2% average success rate compared to 5.9% for ACT and 1.2% for OpenVLA. While ET-VLA excels in tasks like "ball" and "rope," performance varies considerably across tasks for all methods, with challenges remaining in areas like "fridge" and "oven." OpenVLA demonstrates consistently low performance, while ACT shows moderate success. These results highlight the need for our approach to improve the robustness and generalization of beyond unimanual manipulation methods.

RoboTwin. We assess the performance of our method on RoboTwin (Mu et al., 2024), a new simulation benchmark specifically designed for bimanual robot manipulation using AgileX robots. This benchmark includes a variety of tasks. We compare our approach to the Diffusion Policy (Chi et al., 2023), a well-established baseline for visuomotor policy learning. For all experiments, we used a standard image resolution of 320×240 for the camera input.

Table 3 presents a comparative performance analysis between Diffusion Policy and ET-VLA across 14 distinct robot manipulation tasks. The data clearly demonstrates the superior performance of ET-VLA, achieving a significantly higher average success rate of 40.1% compared to the Diffusion Policy's 27.7%. Across nearly all individual tasks, ET-VLA exhibits a notable improvement in performance. For instance, ET-VLA achieves an 85% success rate in "Apple Cabinet Storage" compared to Diffusion Policy's 72%, and a 45% success rate in "Block Handover" where Diffusion Policy only scores 28%. Even in challenging tasks like "Blocks Stack (Easy)" and "Container Place," where Diffusion Policy struggles with very low success rates, ET-VLA achieves significantly higher, albeit still moderate, performance. This consistent outperformance across the majority of tasks highlights the effectiveness of ET-VLA as a solution for robot manipulation challenges.

4.4. Ablation Study

To evaluate the contributions of the proposed SCP and EGoT components in our framework, we conducted two ablation experiments: 1) Removing EGoT while retaining SCP. 2) Removing both EGoT and SCP, leaving only the baseline OpenVLA. These experiments aim to analyze the performance degradation caused by the absence of these components, demonstrating their individual and combined impacts on the overall success rates. The experimental results are demonstrated in Table 4.

In the first ablation experiment, the EGoT module was removed, while the SCP component was retained. This setup evaluates the contribution of EGoT in enhancing task planning capabilities and its awareness of task sequencing for different robots. The model's performance dropped significantly, particularly on tasks such as WashPlate and PlacePlate, which rely heavily on the task planning capabilities provided by EGoT. In contrast, the performance on the PickBread task, which requires only one robot, showed only a slight decline.

In the second ablation experiment, both the EGoT and SCP components were removed, leaving only the baseline OpenVLA. This setup evaluates the combined impact of these two modules on task success. As expected, the performance dropped further compared to the first experiment, with success rates significantly lower across all tasks. Although the PickBread task requires only one robot, it demands the model to generate all 14 action tokens accurately. However,

the model occasionally failed to produce the full sequence of 14 tokens and often moved the arm aimlessly, resulting in task failures.

5. Conclusion

Vision-language-action (VLA) models are widely used for developing robust and generalizable robotic policies. However, current VLA models are not designed for multi-robot applications. In this work, we empirically analyze the failure modes of VLA models on multi-robot manipulation tasks. Specifically, we identify two key issues causing failure: 1) VLA models often fail to generate a sufficient number of action tokens to control multi-robot, and 2) VLA models struggle to execute correct action sequences for parallel robots. To address these challenges, we propose synthetic continued pretraining to enable VLA models to produce the correct number of action tokens, followed by an Embodied Graph-of-Thought approach to enhance the model's awareness of multi-robot task sequences. Our approach is validated through real-world robotic tasks, where we demonstrate that our methods improve the average success rate of state-of-the-art VLA models on real robots from 6.49% to 59.74%.

References

Awadalla, A., Gao, I., Gardner, J., Hessel, J., Hanafy, Y., Zhu, W., Marathe, K., Bitton, Y., Gadre, S., Sagawa, S., et al. Openflamingo: An open-source framework for training large autoregressive vision-language models. *arXiv preprint arXiv:2308.01390*, 2023.

Black, K., Janner, M., Du, Y., Kostrikov, I., and Levine, S. Training diffusion models with reinforcement learning. *arXiv preprint arXiv:2305.13301*, 2023a.

Black, K., Nakamoto, M., Atreya, P., Walke, H., Finn, C., Kumar, A., and Levine, S. Zero-shot robotic manipulation with pretrained image-editing diffusion models. *arXiv preprint arXiv:2310.10639*, 2023b.

Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., Fusai, N., Groom, L., Hausman, K., Ichter, B., Jakubczak, S., Jones, T., Ke, L., Levine, S., Li-Bell, A., Mothukuri, M., Nair, S., Pertsch, K., Shi, L. X., Tanner, J., Vuong, Q., Walling, A., Wang, H., and Zhilinsky, U. π_0: A vision-language-action flow model for general robot control, 2024. URL https://arxiv.org/abs/2410.24164.

Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Chen, X., Choromanski, K., Ding, T., Driess, D., Dubey, A., Finn, C., et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. *arXiv preprint arXiv:2307.15818*, 2023.

Chen, Z., Wu, J., Wang, W., Su, W., Chen, G., Xing, S., Zhong, M., Zhang, Q., Zhu, X., Lu, L., et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024.

Chi, C., Feng, S., Du, Y., Xu, Z., Cousineau, E., Burchfiel, B., and Song, S. Diffusion policy: Visuomotor policy learning via action diffusion. *arXiv preprint arXiv:2303.04137*, 2023.

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H. W., Sutton, C., Gehrmann, S., et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113, 2023.

Dasari, S., Mees, O., Zhao, S., Srirama, M. K., and Levine, S. The ingredients for robotic diffusion transformers. *arXiv preprint arXiv:2410.10088*, 2024.

Fu, Z., Zhao, T. Z., and Finn, C. Mobile aloha: Learning bimanual mobile manipulation with low-cost whole-body teleoperation. *arXiv preprint arXiv:2401.02117*, 2024.

Gao, Z., Mu, Y., Qu, J., Hu, M., Guo, L., Luo, P., and Lu, Y. Dag-plan: Generating directed acyclic dependency graphs for dual-arm cooperative planning. *arXiv preprint arXiv:2406.09953*, 2024.

Goyal, A., Xu, J., Guo, Y., Blukis, V., Chao, Y.-W., and Fox, D. Rvt: Robotic view transformer for 3d object manipulation. In *Conference on Robot Learning*, pp. 694–710. PMLR, 2023.

Grotz, M., Shridhar, M., Chao, Y.-W., Asfour, T., and Fox, D. Peract2: Benchmarking and learning for robotic bimanual manipulation tasks. In *CoRL 2024 Workshop on Whole-body Control and Bimanual Manipulation: Applications in Humanoids and Beyond*, 2024.

Huang, J., Yong, S., Ma, X., Linghu, X., Li, P., Wang, Y., Li, Q., Zhu, S.-C., Jia, B., and Huang, S. An embodied generalist agent in 3d world. In *ICLR 2024 Workshop: How Far Are We From AGI*.

Huang, Y., Liu, X., Zhu, Y., Xu, Z., Shen, C., Che, Z., Zhang, G., Peng, Y., Feng, F., and Tang, J. Label-guided auxiliary training improves 3d object detector. In *European Conference on Computer Vision*, pp. 684–700. Springer, 2022.

James, S., Ma, Z., Arrojo, D. R., and Davison, A. J. Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, 5(2):3019–3026, 2020.

Jiang, X., Qiu, R., Xu, Y., Zhu, Y., Zhang, R., Fang, Y., Xu, C., Zhao, J., and Wang, Y. Ragraph: A general retrieval-augmented graph learning framework. *Advances in Neural Information Processing Systems*, 37:29948–29985, 2024.

Jiang, Y., Gupta, A., Zhang, Z., Wang, G., Dou, Y., Chen, Y., Fei-Fei, L., Anandkumar, A., Zhu, Y., and Fan, L. Vima: General robot manipulation with multimodal prompts. *arXiv preprint arXiv:2210.03094*, 2(3):6, 2022.

Karamcheti, S., Nair, S., Balakrishna, A., Liang, P., Kollar, T., and Sadigh, D. Prismatic vlms: Investigating the design space of visually-conditioned language models. *arXiv preprint arXiv:2402.07865*, 2024.

Khazatsky, A., Pertsch, K., Nair, S., Balakrishna, A., Dasari, S., Karamcheti, S., Nasiriany, S., Srirama, M. K., Chen, L. Y., Ellis, K., et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

Kim, M. J., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., et al. Openvla: An open-source vision-language-action model.

Laurençon, H., Saulnier, L., Tronchon, L., Bekman, S., Singh, A., Lozhkov, A., Wang, T., Karamcheti, S., Rush, A., and Kiela, D. Obelisc: An open web-scale filtered dataset of interleaved image-text documents. *arXiv preprint arXiv:2306.16527*, 2023.

Li, C., Wen, J., Peng, Y., Peng, Y., Feng, F., and Zhu, Y. Pointvla: Injecting the 3d world into vision-language-action models. *arXiv preprint arXiv:2503.07511*, 2025a.

Li, J., Zhu, Y., Xu, Z., Gu, J., Zhu, M., Liu, X., Liu, N., Peng, Y., Feng, F., and Tang, J. Mmro: Are multimodal llms eligible as the brain for in-home robotics? *arXiv preprint arXiv:2406.19693*, 2024.

Li, J., Zhu, Y., Tang, Z., Wen, J., Zhu, M., Liu, X., Li, C., Cheng, R., Peng, Y., Peng, Y., et al. Coa-vla: Improving vision-language-action models via visual-text chain-of-affordance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9759–9769, 2025b.

Li, X., Liu, M., Zhang, H., Yu, C., Xu, J., Wu, H., Cheang, C., Jing, Y., Zhang, W., Liu, H., et al. Vision-language foundation models as effective robot imitators. *arXiv preprint arXiv:2311.01378*, 2023.

Liang, J., Huang, W., Xia, F., Xu, P., Hausman, K., Ichter, B., Florence, P., and Zeng, A. Code as policies: Language model programs for embodied control. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9493–9500. IEEE, 2023.

Lin, F., Hu, Y., Sheng, P., Wen, C., You, J., and Gao, Y. Data scaling laws in imitation learning for robotic manipulation, 2024. URL https://arxiv.org/abs/2410.18647.

Liu, H., Li, C., Li, Y., and Lee, Y. J. Improved baselines with visual instruction tuning. *arXiv preprint arXiv:2310.03744*, 2023a.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b. URL https://openreview.net/forum?id=w0H2xGHlkw.

Liu, X., Zhu, Y., Gu, J., Lan, Y., Yang, C., and Qiao, Y. Mm-safetybench: A benchmark for safety evaluation of multimodal large language models. In *European Conference on Computer Vision*, pp. 386–403. Springer, 2025.

Lu, H., Liu, W., Zhang, B., Wang, B., Dong, K., Liu, B., Sun, J., Ren, T., Li, Z., Yang, H., et al. Deepseek-vl: towards real-world vision-language understanding. *arXiv preprint arXiv:2403.05525*, 2024.

Meng, W., Zaiter, F., Zhang, Y., Liu, Y., Zhang, S., Tao, S., Zhu, Y., Han, T., Zhao, Y., Wang, E., et al. Logsummary: Unstructured log summarization for software systems. *IEEE Transactions on Network and Service Management*, 20(3):3803–3815, 2023.

Mu, Y., Chen, T., Peng, S., Chen, Z., Gao, Z., Zou, Y., Lin, L., Xie, Z., and Luo, P. Robotwin: Dual-arm robot benchmark with generative digital twins (early version). *arXiv preprint arXiv:2409.02920*, 2024.

Niu, D., Sharma, Y., Biamby, G., Quenum, J., Bai, Y., Shi, B., Darrell, T., and Herzig, R. Llarva: Vision-action instruction tuning enhances robot learning. *arXiv preprint arXiv:2406.11815*, 2024.

O'Neill, A., Rehman, A., Gupta, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023a.

O'Neill, A., Rehman, A., Gupta, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023b.

Pertsch, K., Stachowicz, K., Ichter, B., Driess, D., Nair, S., Vuong, Q., Mees, O., Finn, C., and Levine, S. Fast: Efficient action tokenization for vision-language-action models. *arXiv e-prints*, pp. arXiv–2501, 2025.

Prasad, A., Lin, K., Wu, J., Zhou, L., and Bohg, J. Consistency policy: Accelerated visuomotor policies via consistency distillation. *arXiv preprint arXiv:2405.07503*, 2024.

Rana, K., Haviland, J., Garg, S., Abou-Chakra, J., Reid, I. D., and Suenderhauf, N. Sayplan: Grounding large language models using 3d scene graphs for scalable task planning. *CoRR*, 2023.

Reuss, M., Yağmurlu, Ö. E., Wenzel, F., and Lioutikov, R. Multimodal diffusion transformer: Learning versatile behavior from multimodal goals. 2024.

Shridhar, M., Manuelli, L., and Fox, D. Perceiver-actor: A multi-task transformer for robotic manipulation. In *Conference on Robot Learning*, pp. 785–799. PMLR, 2023.

Uehara, M., Zhao, Y., Black, K., Hajiramezanali, E., Scalia, G., Diamant, N. L., Tseng, A. M., Biancalani, T., and Levine, S. Fine-tuning of continuous-time diffusion models as entropy-regularized control. *arXiv preprint arXiv:2402.15194*, 2024a.

Uehara, M., Zhao, Y., Black, K., Hajiramezanali, E., Scalia, G., Diamant, N. L., Tseng, A. M., Levine, S., and Biancalani, T. Feedback efficient online fine-tuning of diffusion models. *arXiv preprint arXiv:2402.16359*, 2024b.

Walke, H. R., Black, K., Zhao, T. Z., Vuong, Q., Zheng, C., Hansen-Estruch, P., He, A. W., Myers, V., Kim, M. J., Du, M., et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pp. 1723–1736. PMLR, 2023.

Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024a.

Wang, Y., Zhang, Y., Huo, M., Tian, R., Zhang, X., Xie, Y., Xu, C., Ji, P., Zhan, W., Ding, M., et al. Sparse diffusion policy: A sparse, reusable, and flexible policy for robot learning. *arXiv preprint arXiv:2407.01531*, 2024b.

Wen, J., Zhu, Y., Li, J., Zhu, M., Wu, K., Xu, Z., Cheng, R., Shen, C., Peng, Y., Feng, F., et al. Tinyvla: Towards fast, data-efficient vision-language-action models for robotic manipulation. *arXiv preprint arXiv:2409.12514*, 2024a.

Wen, J., Zhu, Y., Zhu, M., Li, J., Xu, Z., Che, Z., Shen, C., Peng, Y., Liu, D., Feng, F., and Tang, J. Object-centric instruction augmentation for robotic manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4318–4325, 2024b. doi: 10.1109/ICRA57147.2024.10609992.

Wen, J., Zhu, Y., Zhu, M., Li, J., Xu, Z., Che, Z., Shen, C., Peng, Y., Liu, D., Feng, F., et al. Object-centric instruction augmentation for robotic manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4318–4325. IEEE, 2024c.

Wu, K., Zhu, Y., Li, J., Wen, J., Liu, N., Xu, Z., and Tang, J. Discrete policy: Learning disentangled action space for multi-task robotic manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8811–8818. IEEE, 2025.

Zawalski, M., Chen, W., Pertsch, K., Mees, O., Finn, C., and Levine, S. Robotic control via embodied chain-of-thought reasoning. *arXiv preprint arXiv:2407.08693*, 2024.

Zhao, Q., Lu, Y., Kim, M. J., Fu, Z., Zhang, Z., Wu, Y., Ma, M. L. Q., Han, S., Finn, C., Handa, A., Liu, M.-Y., Xiang, D., Wetzstein, G., and Lin, T.-Y. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models, 2024.

Zhao, T. Z., Tompson, J., Driess, D., Florence, P., Ghasemipour, S. K. S., Finn, C., and Wahid, A. Aloha unleashed: A simple recipe for robot dexterity. In *8th Annual Conference on Robot Learning*.

Zhou, Q. and Zhu, Y. Make a long image short: Adaptive token length for vision transformers. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 69–85. Springer, 2023.

Zhou, Z., Zhu, Y., Wen, J., Shen, C., and Xu, Y. Vision-language-action model with open-world embodied reasoning from pretrained knowledge. *arXiv preprint arXiv:2505.21906*, 2025.

Zhu, M., Zhu, Y., Li, J., Wen, J., Xu, Z., Che, Z., Shen, C., Peng, Y., Liu, D., Feng, F., and Tang, J. Language-conditioned robotic manipulation with fast and slow thinking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4333–4339, 2024a. doi: 10.1109/ICRA57147.2024.10611525.

Zhu, M., Zhu, Y., Li, J., Wen, J., Xu, Z., Che, Z., Shen, C., Peng, Y., Liu, D., Feng, F., et al. Language-conditioned robotic manipulation with fast and slow thinking. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 4333–4339. IEEE, 2024b.

Zhu, M., Zhu, Y., Li, J., Wen, J., Xu, Z., Liu, N., Cheng, R., Shen, C., Peng, Y., Feng, F., et al. Scaling diffusion policy in transformer to 1 billion parameters for robotic manipulation. *arXiv preprint arXiv:2409.14411*, 2024c.

Zhu, M., Zhu, Y., Liu, X., Liu, N., Xu, Z., Shen, C., Peng, Y., Ou, Z., Feng, F., and Tang, J. A comprehensive overhaul of multimodal assistant with small language models. *arXiv preprint arXiv:2403.06199*, 2024d.

Zhu, M., Zhu, Y., Li, J., Wen, J., Xu, Z., Liu, N., Cheng, R., Shen, C., Peng, Y., Feng, F., et al. Scaling diffusion policy in transformer to 1 billion parameters for robotic manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 10838–10845. IEEE, 2025a.

Zhu, M., Zhu, Y., Li, J., Zhou, Z., Wen, J., Liu, X., Shen, C., Peng, Y., and Feng, F. Objectvla: End-to-end open-world object manipulation without demonstration. *arXiv preprint arXiv:2502.19250*, 2025b.

Zhu, Y., Meng, W., Liu, Y., Zhang, S., Han, T., Tao, S., and Pei, D. Unilog: Deploy one model and specialize it for all log analysis tasks. *arXiv preprint arXiv:2112.03159*, 2021.

Zhu, Y., Liu, N., Xu, Z., Liu, X., Meng, W., Wang, L., Ou, Z., and Tang, J. Teach less, learn more: On the undistillable classes in knowledge distillation. *Advances in Neural Information Processing Systems*, 35:32011–32024, 2022.

Zhu, Y., Ou, Z., Mou, X., and Tang, J. Retrieval-augmented embodied agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17985–17995, 2024e.

Zhu, Y., Zhu, M., Liu, N., Xu, Z., and Peng, Y. Llava-phi: Efficient multi-modal assistant with small language model. In *Proceedings of the 1st International Workshop on Efficient Multimedia Computing under Limited*, pp. 18–22, 2024f.