SkillVLA: Tackling Combinatorial Diversity in Dual-Arm Manipulation via Skill Reuse

Xuanran Zhai*[1], Zekai Huang*[1], Longyan Wu*[3,6], Qianyou Zhao[5], Qiaojun Yu[4], Jieji Ren[5], Ce Hao†[1,2], and Harold Soh[1,7]

[1]National University of Singapore, [2]Zhongguancun Academy, [3]Shanghai Innovation Institute, [4]Shanghai AI Laboratory, [5]Shanghai Jiao Tong University, [6]Fudan University, [7]Smart Systems Institute, NUS

* Equal contribution. † Corresponding author: cehao@u.nus.edu, haoce@bza.edu.cn

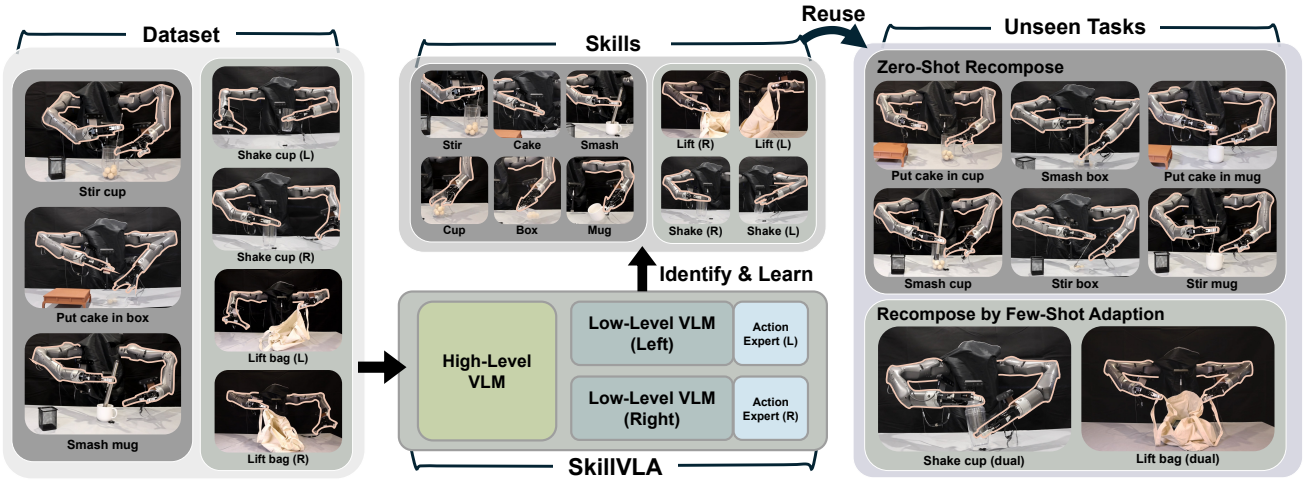

Fig. 1: SkillVLA extracts single-arm skills from training data with hierarchical reasoning and skill-adaptive learning, being able to recompose them into unseen combinations during test time.

***Abstract*—Recent progress in vision–language–action (VLA) models has demonstrated strong potential for dual-arm manipulation, enabling complex behaviors and generalization to unseen environments. However, mainstream bimanual VLA formulations largely overlook the critical challenge of *combinatorial diversity*. Different pairings of single-arm behaviors can induce qualitatively distinct task behaviors, yet existing models do not explicitly account for this structure. We argue that effective bimanual VLAs should support skill reuse—the ability to recombine previously learned single-arm skills across novel left–right pairings—thereby avoiding the need to separately learn every possible combination. Current VLA designs entangle skills across arms, preventing such recomposition and limiting scalability. To address this limitation, we propose SkillVLA, a framework explicitly designed to enable skill reuse in dual-arm manipulation. Extensive experiments demonstrate that SkillVLA substantially improves skill composition, increasing zero-shot recomposition success rate from 0% to 51%, and achieves strong performance on cooperative and long-horizon tasks.**

I. Introduction

Bimanual manipulation expands a robot's effective workspace and enables concurrent, coordinated actions, supporting a wider range of tasks than single-arm systems [44, 66, 55, 16, 13, 28]. In this context, recent vision–language–action (VLA) models [8, 27, 31, 47, 45, 52] have shown impressive performance on bimanual manipulation; by jointly predicting both arms' actions (e.g., via a concatenated action vector) with large-capacity architectures, VLA methods can represent complex coordinated behaviors. Further, by leveraging vision–language pretraining of VLM models, recent bimanual VLAs have shown the ability to generalize to novel scenes and instructions [27, 17, 26, 5, 7, 4, 8, 29].

Despite recent progress, learning generalist bimanual policies remains challenging, as dual-arm control introduces richer behavior patterns. Specifically, real-world bimanual behaviors exhibit pronounced ***combinatorial diversity***. As shown in Fig. 1, many dual-arm tasks can be viewed as compositions of single-arm behaviors, and different pairings of left- and right-arm skills induce different bimanual tasks. As the underlying skill set grows, the number of possible pairings increases quadratically, resulting in an overwhelming number of combinations corresponding to different tasks. Current VLA formulations largely overlook combinatorial diversity. Predicting both arms via action concatenation requires the model to learn the joint distribution of left- and right-arm distributions. While this design is simple and captures tight coordination, it confines outputs to action pairings observed in demonstrations. As a result, the policy has limited combinatorial generalization and cannot form new bimanual behaviors by recombining single-arm skills, which makes them unable to address combinatorial diversity. We refer to this limitation as ***skill entanglement*** and analyze it in detail in Sec. IV-C.

To address the combinatorial diversity, a model must be able to effectively reuse robot skills learned from given demonstrations in a more flexible and thorough way. We name this ability as ***skill reuse***. Concretely, instead of mechanically treating every bimanual behavior as an indivisible dual-arm primitive, an ideal model should automatically uncover the latent skill structure within demonstrations and perform reasoning at the skill level. For bimanual behaviors that are essentially compositions of single-arm skills, the model should reuse the discovered skills across different left–right combinations and compose them to form novel pairings that are not explicitly observed during training. We view such flexible reuse as a key foundation for broad generalization in bimanual settings, and provide a more formal definition and discussion in Sec. IV.

We propose **SkillVLA** to enable skill reuse and address combinatorial diversity. Unlike conventional VLAs relying on monolithic dual-arm reasoning throughout, SkillVLA automatically identifies skill structures from demonstrations and performs *skill-adaptive* action generation. Concretely, for single-arm skills, it can retrieve and invoke them as independent units for recomposition or further adaptation. For behaviors that require cooperation, the model instead captures inter-arm dependence. SkillVLA achieves this with a two-level reasoning architecture. A shared high-level VLM, equipped with a cooperation estimator, captures global task intent and makes skill decisions. At the lower level, the policy follows the high-level instruction and switches between operation modes: it performs per-arm reasoning and action generation for single-arm skills, while allowing these two streams to be entangled by communication when cooperative behavior is deemed necessary by the high-level reasoning. Through skill-adaptive generation and per-arm reasoning, SkillVLA further exploits the VLM's capability by extending reasoning to a finer semantic granularity, leading to stronger generalization.

We conduct real-robot experiments on 20 manipulation tasks and two long-horizon multi-stage tasks to validate SkillVLA's skill reuse and generalization ability. First, to evaluate if models are able to tackle combinatorial diversity, we test them on unseen tasks composed by learned skills, where strong bimanual VLA baselines achieve near-zero success due to skill entanglement. SkillVLA raises the success rate to $\mathbf{51}\%$, demonstrating a decisive improvement in combinatorial generalization. Second, to confirm that our method does not sacrifice inter-arm cooperation, we evaluate on three highly-cooperative tasks, where SkillVLA matches the performance of the baselines, showing that its adaptive communication remains expressive for close coordination. Third, we study two multi-stage long-horizon tasks that interleave collaborative and independent phases. While SkillVLA correctly identifies the cooperation level required and, further, completes the tasks faster by recomposing skills to parallelize the two arms when possible, reducing execution time by $\mathbf{21}\%$. Finally, we conduct a continual-learning study to show that skill reuse also benefits long-term skill learning: SkillVLA can effectively reuse existing behaviors for acquiring new skills, achieving significantly better performance under limited demonstrations.

In summary, this paper makes three main contributions:

- We identify the *skill entanglement* issue in current VLAs and formulate the bimanual *skill reuse* problem.
- We propose SkillVLA, which enables flexible skill reuse to tackle combinatorial diversity.
- We validate SkillVLA on a real dual-arm robot, demonstrating combinatorial generalization ability via skill reuse, while maintaining strong performance on cooperative and long-horizon tasks.

II. Related Work

Dual-arm manipulation has long been a central topic in robotics. Early control-based or rule-based methods are often developed around specific task structures, leading to task-centric designs for applications such as assembly [20, 63], cloth folding [35, 6, 54, 41, 10, 2, 15], and bagging [12, 3, 11]. For dual-arm manipulation, many traditional methods explicitly model bimanual relations, such as master–slave coordination [23, 46, 1], relative pose control [48, 39, 37], and internal force regulation [9, 42, 49]. These methods can be effective within a specific family of tasks and under controlled conditions, but their generalization to multitask settings or unfamiliar environments remains limited, as their coordination strategies are usually designed for particular objects, task procedures, or interaction patterns.

Learning-based methods [14, 62, 61, 22] form another major family of manipulation approaches. By learning from expert demonstrations, they have been applied to a broad range of settings, including dexterous end-effectors [50, 43, 18], mobile manipulation [56, 19], learning from human videos [66, 21, 38], and multitask learning [22, 51]. Owing to their flexibility, imitation learning methods have become a dominant paradigm for robot manipulation. More recently, large-scale pretraining with vision-language-action models [58, 8, 27, 59, 17, 53, 67, 64] and world-action models [65, 57, 34, 60] has made generalist robot policies increasingly feasible. Recent policies have shown promising generalization to unseen environments without task-specific finetuning.

However, unlike control-based bimanual methods, most learning-based methods still treat dual-arm robots as a monolithic system and learn a joint policy over the two arms. This formulation is simple and scalable, but it largely overlooks the structural differences between bimanual manipulation and single-arm manipulation, such as role assignment, asymmetric cooperation, cross-arm dependency, and coordinated skill reuse. Only a few recent works [25, 33, 36] have begun to decompose dual-arm learning to improve learning efficiency. To the best of our knowledge, skill recomposition in bimanual manipulation remains underexplored and has not been systematically addressed.

III. Problem Formulation

We consider a bimanual robot with left and right manipulators. At each time step, the robot receives an input $x \in \mathcal{X}$ (usually observation, instruction and robot state) and produces

actions $a_L \in \mathcal{A}_L$ and $a_R \in \mathcal{A}_R$; we write $a = (a_L, a_R)$. We are given a dataset of demonstrations $\mathcal{D} = \{(x_i, a_L^i, a_R^i)\}_{i=1}^{\mathcal{I}}$, where each tuple is a state–action sample extracted from expert trajectories. We assume these expert actions are generated by executing skills drawn from an underlying skill set

$$S = S_{\text{pair}} \cup S_D,\ S_{\text{pair}} = \{(s_L^m, s_R^m)\}_{m=1}^M,\ S_D = \{s_D^n\}_{n=1}^N$$

where S_L and S_R are the single-arm skill sets for the left and right arm (*Def.* 1), and S_D is a set of genuinely dual-arm (bimanual) skills (*Def.* 2). Here S_{pair} collects the *paired* single-arm skills that actually appear in the demonstrations.

Our goal is to learn a bimanual policy $\pi_\theta(a_L, a_R \mid x)$ that is able to effectively ***reuse*** these learned skills (*Def.* 3), which we view as a key factor for resolving combinatorial diversity in generalist bimanual manipulation policies.

IV. Skill Reuse in Dual-Arm Manipulation

In this section, we first give specific definitions of skills and skill reuse (Sec. IV-A), then we analyze what is required to solve this problem: for skill reuse to be possible, the model must (i) select suitable skills for any given scene x (Sec. IV-B), and (ii) generate correct actions for the selected skill (or skill pair). The latter motivates explicitly distinguishing between single-arm and dual-arm skills during training and execution, which current VLA-based methods do not support (Sec. IV-C).

A. Definitions

To formulate and analyze skill reuse in a general bimanual setting, we first define what we mean by a *skill*. We distinguish between *single-arm skills*, which can be executed by one arm in isolation, and *dual-arm skills*, which require coordinated behavior between the two arms.

Definition 1 (Single-Arm Skills). For $\kappa \in \{L, R\}$, S_κ is the set of all possible left/right arm skills. A single-arm skill $s_\kappa \in S_\kappa$ is a conditional distribution $\pi_{s_\kappa}(a_\kappa \mid x)$. For any pair $(s_L, s_R) \in S_L \times S_R$, their composition is defined as

$$\pi_{s_L, s_R}(a_L, a_R \mid x) = \pi_{s_L}(a_L \mid x)\, \pi_{s_R}(a_R \mid x).$$

Definition 2 (Dual-Arm Skills). A dual-arm skill $s_D \in S_D$ is a joint conditional distribution $\pi_{s_D}(a_L, a_R \mid x)$ such that the conditional mutual information $I_{\pi_{s_D}}(a_L; a_R \mid x) > 0$. Equivalently, $\pi_{s_D}(a_L, a_R \mid x) \neq \pi_{s_L}(a_L \mid x)\, \pi_{s_R}(a_R \mid x)$ for any single-arm skills s^L, s^R.

Achieving inter-arm coordination for dual-arm skills requires an information pathway that induces dependence between a_L and a_R. We denote this pathway conceptually by an inter-arm message m, yielding action generation forms such as $\pi_L(a_L \mid x, m_L)$ and $\pi_R(a_R \mid x, m_R)$. In practice, inter-arm message may be realized in various ways, such as explicit message passing, or implicitly through shared parameters as in common monolithic policies.

For any possible scene x, we assume there exists a (possibly multi-valued) set of *correct skills* $S^\star(x) \subseteq (S_L \times S_R) \cup S_D$ that are appropriate for that scene. We decompose this set as $S^\star(x) = S^\star_{\text{pair}}(x) \cup S^\star_D(x)$, where

$$S^\star_{\text{pair}}(x) = S^\star(x) \cap (S_L \times S_R), \qquad S^\star_D(x) = S^\star(x) \cap S_D,$$

corresponding to correct skills that can be expressed as a composition of single-arm skills and correct skills that are genuinely dual-arm. Note that $S^\star_{\text{pair}}(x)$ is defined over the full product $S_L \times S_R$, and is not restricted to the paired set S_{pair}; in particular, it may contain recombinations of single-arm skills that never co-occur in the demonstrations.

Definition 3 (Skill Reuse). For a bimanual policy learned from skill set $S = S_{\text{pair}} \cup S_D = \{(s_L^m, s_R^m)\}_{m=1}^M \cup \{s_D^n\}_{n=1}^N$, we define its possible behavior set as $S^p = (S_L \times S_R) \cup S_D$, where S_L, S_R are the decomposed per-arm skills with $S_L = \{s_L^m\}_{m=1}^M$ and $S_R = \{s_R^m\}_{m=1}^M$. We say that the policy exhibits ***skill reuse*** if, for every scene x (possibly unseen) with $S^\star(x) \cap S^p \neq \emptyset$, there exists a skill $s \in S^\star(x) \cap S^p$ such that $\pi_\theta(a_L, a_R \mid x) \approx \pi_s(a_L, a_R \mid x)$.

Equivalently, this requirement decomposes into two major subproblems: (i) *single-arm composition*: if the correct behavior for x is a composition of single-arm skills, i.e. $S^\star(x) \cap S^p = \{(s_L^i, s_R^j)\}$ for some i, j, the policy must be able to realize this composition, even for (s_L^i, s_R^j) that never appeared in S_{pair}; and (ii) *dual-arm reproduction*: when the correct skill is genuinely dual-arm, i.e. $S^\star(x) \cap S^p = \{s_D^i\}$ for some i, the policy should also produce action for s_D^i while maintaining its inter-arm coordination structure. For cases where multiple skill choices are valid, specifically, the model should then select and produce one of them.

Remark (Fast bimanual continual learning by skill reuse). In practice, many dual-arm skills are close to simple compositions of two single-arm skills, in the sense that each arm largely follows an independent per-arm behavior under the same context x. The challenge is the coordination between the two arms: the joint action distribution deviates from an independent product due to inter-arm coupling (cf. $I(a_L; a_R \mid x) > 0$ in Def. 2). Therefore, if a model can flexibly invoke reusable single-arm skills, acquiring a new bimanual skill can often be done with minimal finetuning by mainly learning additional information for coupling on top of the existing per-arm skills, rather than relearning both arms from scratch. Therefore, effective skill reuse can improve efficiency in continual learning or large-scale learning.

B. Skill Selector

Given a library of skills, action generation can be viewed as first selecting which skill (or skill pair) to use for a given scene x, and then sampling actions from the corresponding skill distributions. We assume a (finite) skill library consisting of indexed single-arm skills $\{s_L^k\}_{k \in K_L} \subseteq S_L$ and $\{s_R^l\}_{l \in K_R} \subseteq S_R$, and indexed dual-arm skills $\{s_D^d\}_{d \in K_D} \subseteq S_D$. We introduce a discrete *skill index* random variable

$$Y \in \mathcal{Y} := K_D \cup (K_L \times K_R),$$

where $Y = d \in K_D$ means that the dual-arm skill $\pi_{s_D}(a_L, a_R \mid x)$ should be used, and $Y = (k, l) \in K_L \times K_R$ means that the single-arm skills $\pi_{s_L^k}(a_L \mid x)$ and $\pi_{s_R^l}(a_R \mid x)$ should be used concurrently.

An ideal high-level selector defines a distribution $p^\star(Y \mid x)$, and the resulting action distribution is the mixture

$$\pi^\star(a_L, a_R \mid x) = \sum_{d \in K_D} p^\star(Y = d \mid x)\, \pi_{s_D}(a_L, a_R \mid x) + \sum_{(k,l) \in K_L \times K_R} p^\star(Y = (k,l) \mid x)\, \pi_{s_L^k}(a_L \mid x)\, \pi_{s_R^l}(a_R \mid x).$$

We use the term *skill selector* for this conceptual mechanism, without assuming a specific module or architecture. Ideally, the skill selector should select appropriate skills not only for scenes seen in the demonstrations, but also for novel inputs where the correct skill configuration has never been observed.

C. Skill Entanglement in Current VLAs

VLAs are often built upon a pretrained VLM, which provides strong generalization over visual scenes and natural-language instructions. Typically, a VLA has an additional action module (or "action expert") to generate actions. In bimanual manipulation, the action is commonly represented as a single vector by concatenating the left- and right-arm actions.

The VLM is a natural candidate to implement a *generalizable skill selector*—i.e., to map a scene x to an appropriate skill index Y (or an equivalent decision variable) that can generalize beyond the demonstrated scenes. However, even if the upstream skill decision is sufficiently able to separate scenes requiring different skills, can the downstream action generation mechanism *reuse* skills in the sense of Sec. IV-A?

We argue that common VLA designs exhibit two forms of *skill entanglement* that hinder effective skill reuse:

Action Entanglement. Many bimanual VLA policies are trained to predict a single concatenated joint action vector (a_L, a_R). This monolithic supervision couples the two arms at the output level and encourages the model to fit the empirical *joint* distribution induced by paired demonstrations. Consequently, the learned policy may internalize dataset-specific cross-arm correlations rather than isolating reusable single-arm structure. This becomes problematic for skill reuse and recomposition. Even if the upstream vision–language reasoning can distinguish scenes requiring different skills, the downstream action generator may fail to (i) disentangle single-arm skills from dual-arm coordination patterns and (ii) support recomposition of single-arm skills beyond the left–right pairings observed during training. In other words, joint-action learning can bias the model toward reproducing demonstrated bimanual patterns, limiting its ability to generalize to unseen combinations of per-arm behaviors.

Latent entanglement in action-expert VLAs. As previously mentioned, recent VLA methods augment a pretrained VLM with a dedicated action-generation module (e.g., π_0/$\pi_{0.5}$ [8, 27], RDT2 [47], DexVLA [52]). Abstractly, a VLM encodes context x into a representation z, and an action module predicts bimanual actions conditioned on z,

$$p(a_L, a_R \mid x) = \int p(a_L, a_R \mid z)\, p(z \mid x)\, dz,$$

where $p(z \mid x)$ is induced by the VLM backbone and $p(a_L, a_R \mid z)$ is implemented by the action expert.

While this architecture can be effective in practice, it introduces an additional pathway for skill entanglement. In bimanual imitation, the shared latent z learned from paired demonstrations may implicitly encode cross-arm dependencies. This *latent entanglement* can degrade skill recomposition when the policy is evaluated on unseen left–right pairings, since the action expert conditions both arms on a representation that already mixes information across arms.

V. Method: SkillVLA

In this section, we present **SkillVLA**, a method designed to enable effective skill reuse to address combinatorial diversity and accelerate the acquisition of new skills. We first describe the overall architecture, which realizes skill reuse via a two-level reasoning pipeline and a cooperation estimator for behavior-type identification. We then introduce additional training strategies that help the model better capture skill structure and infer the degree of inter-arm cooperation.

A. Method Pipeline

Our method (overview in Fig. 2) follows the common VLA paradigm, with a top-level VLM and actions generated via an iterative flow-matching process [30, 32]. In our implementation, we use the pretrained PaliGemma [7] backbone released with $\pi_{0.5}$ [27] for initiating the VLMs. Our method consists of two main functional components, described below:

Two-level reasoning (skill selection and action generation). Because an explicit skill library is typically unavailable in practice, we aim for the model to *discover* and *instantiate* skill representations that support both learning and reuse. Skills can be represented in many forms; in SkillVLA we use natural language as the skill descriptor, which aligns naturally with the VLM backbone. We implement this choice via a two-level reasoning pipeline.

As shown in Fig. 2, the high-level module explicitly produces per-arm sub-prompts that act as skill descriptors. Concretely, we prompt the VLM to generate two texts (u_L, u_R) in a fixed format:

$$(u_L, u_R) \sim p(u_L, u_R \mid x),$$

where u_L and u_R describe what the left and right arms should do, respectively (e.g., "pick the cake on the right" vs. "pick the box on the left"). This representation targets task intent and *explicitly* decouples per-arm skill selection, enabling flexible single-arm recomposition: novel combinations can be formed by pairing previously generated (or learned) u_L and u_R in new scenes. During low-level skill learning, we freeze the high-level VLM to preserve its vision–language generalization while training the action components.

At the low level, per-arm actions are generated by two separate streams. Each stream uses its own low-level VLM (fine-tuned separately, e.g., via LoRA [24]) to process the visual input and the corresponding per-arm prompt, producing a per-arm latent representation $z_i = f_i(x, u_i)$, where

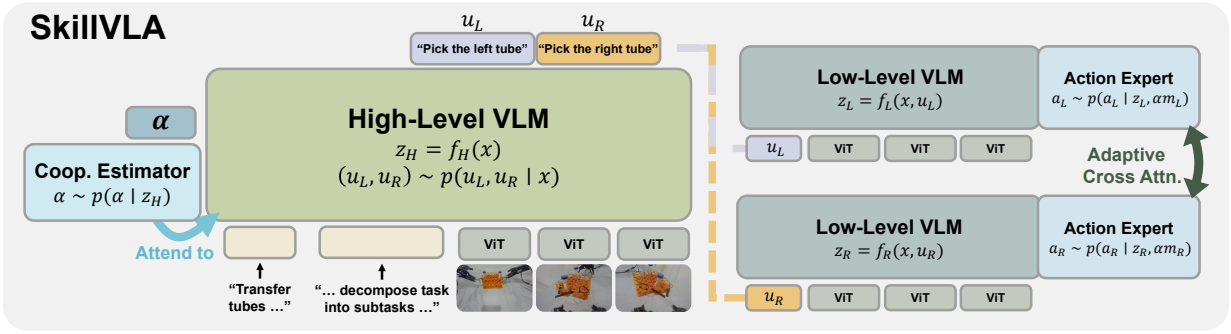

Fig. 2: **SkillVLA framework.** SkillVLA adopts a two-level reasoning pipeline, where the high-level VLM generates separate subtasks for arms and low-level VLMs further process the prompts to instruct action generation. Inter-arm cross-attention enables cooperative behavior generation, controlled by a cooperation estimator that identifies the operation mode required.

$i \in \{L, R\}$. Action experts then predict actions conditioned on the corresponding latent and the arm state. To support coordinated bimanual behaviors when needed, we introduce an adaptive cross-attention mechanism between the action experts to capture inter-arm dependence:

$$a_i \sim p(a_i \mid z_i, \alpha\, m_i)\,,$$

where m_i is the inter-arm message produced by cross-attention. The *cooperation level* signal α gates this message to enable skill-adaptive action generation and is further elaborated upon below.

Cooperation estimator (behavior mode identification). While inter-arm communication is useful for capturing low-level dependence, it should be *selectively* enabled; for single-arm skills, the two arms should remain largely disentangled during both training and evaluation. To this end, we introduce a cooperation estimator that attends to the high-level VLM representation and predicts a scalar $\alpha \in [0, 1]$, indicating the degree of inter-arm cooperation (larger α implies stronger coupling). Formally, the estimator predicts

$$\alpha \sim p(\alpha \mid z_H), \qquad z_H = f_H(x),$$

where z_H is the latent representation produced by the frozen high-level VLM. This signal serves as a mode identifier, indicating whether the current behavior is better explained as (i) a composition of single-arm skills or (ii) a cooperative dual-arm skill. We gate inter-arm message passing by α, enabling the policy to interpolate between per-arm generation and coupled dual-arm generation.

To train α_t, we use a simple *communication usefulness* objective derived from behavioral cloning (BC). At timestep t, the BC loss is

$$\mathcal{L}_{\text{BC}} = \|\hat{a}_t^L - a_t^L\|_2^2 + \|\hat{a}_t^R - a_t^R\|_2^2.$$

We compute $\mathcal{L}_{\text{BC}}^{\text{on}}$ with cross-attention enabled and $\mathcal{L}_{\text{BC}}^{\text{off}}$ with cross-attention disabled. We then define

$$\mathcal{L}_{\text{coop}} = \lambda \cdot \left(\mathcal{L}_{\text{BC}}^{\text{on}} - \mathcal{L}_{\text{BC}}^{\text{off}}\right)_{\text{sg}} \cdot \alpha_t,$$

where $(\cdot)_{\text{sg}}$ denotes stop-gradient. Minimizing $\mathcal{L}_{\text{coop}}$ increases α_t when communication reduces the BC error and decreases α_t otherwise.

B. Additional Cooperation-Level Learning

Because α directly gates cross-arm interaction, accurately inferring the cooperation level is critical. We introduce additional mechanisms to encourage reliable cooperation estimation, which we use by default in our implementation.

Priors and regularizers for cooperation learning. VLMs are pretrained on web-scale data and thus encode broad task semantics and commonsense regularities (e.g., when two arms are typically required). This makes them a natural source of priors for estimating task-dependent cooperation. To distill this information into a lightweight estimator, we use an off-the-shelf VLM to produce a prior coordination strength $\alpha^{\text{vlm}} \in [0, 1]$ for the current scene and task (or $\alpha^{\text{vlm}} \in \{0, 1\}$ when using a discrete gate).

Let $\tilde{\alpha}_t$ denote the gate value actually used for communication (i.e., the continuous α_t, or the soft discrete probability $\hat{y}_t$ defined below). We regularize $\tilde{\alpha}_t$ toward the VLM prior and further encourage (a) sticky gating for temporal smoothness and (b) conservative communication, keeping the gate small unless interaction is beneficial:

$$\mathcal{L}_{\text{prior}} = \ell(\tilde{\alpha}_t, \alpha^{\text{vlm}}),$$
$$\mathcal{L}_{\text{sticky}} = \ell(\tilde{\alpha}_t, \tilde{\alpha}_{t-1}), \qquad \mathcal{L}_{\text{sup}} = \tilde{\alpha}_t.$$

We set $\ell(u, v) = \|u - v\|_2^2$ when $\tilde{\alpha}_t$ is continuous, and use a Bernoulli cross-entropy $\ell(u, v) = -v \log u - (1-v)\log(1-u)$ when $\tilde{\alpha}_t \in (0, 1)$ is interpreted as a probability.

Cooperation level discretization. In practice, a continuous gate α_t can exhibit small but persistent fluctuations that destabilize action generation. To improve stability, we (optionally) discretize the gate by restricting $\alpha_t \in \{0, 1\}$. Specifically, the model predicts $\hat{y}_t \in (0, 1)$, the probability of enabling cross-arm communication, and we train it with a binary cross-entropy loss:

$$\mathcal{L}_{\text{disc}} = \text{BCE}(y_t, \hat{y}_t), \qquad y_t = \mathbb{I}\left[\mathcal{L}_{\text{BC}}^{\text{on}} < \mathcal{L}_{\text{BC}}^{\text{off}}\right].$$

At inference time, we binarize $\hat{y}_t$ to obtain $\hat{\alpha}_t \in \{0, 1\}$, which directly enables or disables the cross-attention message. We apply the same priors and regularizers from Sec. V-B to $\hat{y}_t$ as a soft relaxation, thereby shaping the resulting discrete gate. This tokenized formulation simplifies gate prediction and

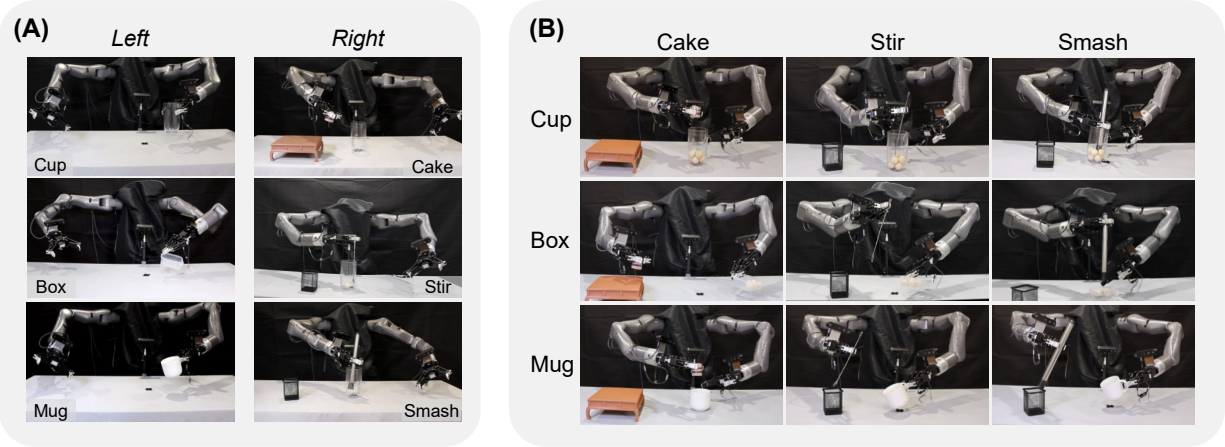

Fig. 3: **Skill recomposition tasks. (A):** The models are trained on demonstrations of three skills for each arm. **(B):** After the models have learned the skills, zero-shot tests are conducted for every possible combination of left and right-arm skills.

TABLE I: Success Rates on Skill Recomposition Tasks ↑

Methods	Recomposition Tasks									
	Cup×Cake	Cup×Stir	Cup×Smash	Box×Cake	Box×Stir	Box×Smash	Mug×Cake	Mug×Stir	Mug×Smash	Avg.
$\pi_{0.5}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
π_0-FAST	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
TwinVLA	0.1	0.0	0.0	0.2	0.0	0.1	0.0	0.0	0.0	0.04
SkillVLA	**0.7**	**0.4**	**0.5**	**0.6**	**0.4**	**0.5**	**0.6**	**0.3**	**0.6**	**0.51**

TABLE II: Success Rates on Skills Learned

Methods	Skills Learned						
	Cup	Box	Mug	Cake	Stir	Smash	Avg.
$\pi_{0.5}$	0.7	**1.0**	**0.8**	0.8	**0.6**	0.7	0.77
π_0-FAST	**0.9**	0.8	0.7	0.9	0.5	0.4	0.70
TwinVLA	0.7	0.8	0.5	0.8	0.5	0.7	0.67
SkillVLA	0.8	0.7	**0.8**	**1.0**	**0.6**	**0.8**	**0.78**

empirically improved stability in preliminary experiments. Implementation and training details are provided in the appendix.

VI. Experiments

In this section, we evaluate our model's ability to (i) freely reuse learned single-arm skills, (ii) capture the inter-arm coordination required by truly cooperative behaviors, and (iii) identify when cooperation is necessary for a given state. Our experiments are designed to answer the following questions:

1) Can SkillVLA flexibly recompose learned skills and generalize to unseen combinations of single-arm skills?
2) Is SkillVLA's communication mechanism sufficient to support tight inter-arm coordination and reproduce cooperative dual-arm skills?
3) Can SkillVLA handle multi-stage, long-horizon tasks by estimating cooperation level and switching between operating modes?
4) In continual learning, does SkillVLA better leverage previously learned skills to acquire related new skills compared to existing methods?

Baselines and evaluation metrics. We compare SkillVLA against $\pi_{0.5}$ [27], a representative state-of-the-art VLA baseline. For Q1, we additionally include π_0-FAST [40] as a representative autoregressive VLA, which represents the other major paradigm beyond diffusion-based VLAs and tests whether skill entanglement persists across architectures. In the Q1 setting, these general-purpose VLAs achieve near-zero success due to severe skill entanglement. To enable more informative comparisons, we further include TwinVLA [25] as a factorized baseline that composes two single-arm VLAs by sharing action experts while allowing *uncontrolled* cross-arm attention. For most tasks (except the long-horizon task), we report *success rate* as the primary metric. For long-horizon tasks which comprise multiple stages, we additionally report a *progress score* and the *average completion time* to quantify execution quality and efficiency. Additional experimental details are provided in the appendix.

A. Single-Arm Skill Reuse

Task Setup. Here, we evaluate whether a bimanual VLA can *reuse* learned single-arm skills under *unseen* left–right combinations. As shown in Fig. 3, our training data include three skills per arm, collected from single-arm executions while the other arm remains stationary. At test time, we ask the policy to execute a left-arm skill and a right-arm skill *simultaneously* in combinations that never appear in the training set. This setting directly probes skill reuse since the policy should invoke the correct per-arm behaviors and maintain stable concurrent execution, rather than reverting to familiar demonstration-time patterns.

Performance Results. As reported in Tab. I and Tab. II, all methods perform comparably on the *seen* single-arm skills,

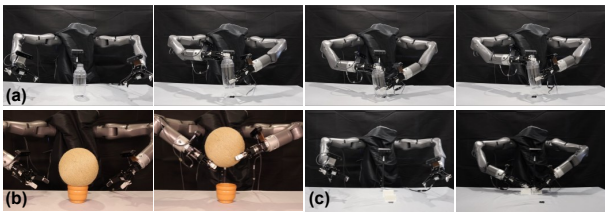

Fig. 4: **Cooperative tasks. (a)** SHAKE: Shake the cup without the cap falling off. **(b)** BALL: Lift the ball steadily. **(c)** ALIGN: Align the blocks on the table.

but their behaviors *diverge sharply* when asked to *zero-shot* recompose unseen left-right combinations.

In particular, $\pi_{0.5}$ and π_0-FAST achieve a 0% success rate. Qualitatively, these models frequently revert to the training-time regime in which one arm acts while the other remains idle, indicating that they primarily reproduce skill pairings present in the dataset rather than freely recombining per-arm behaviors. Moreover, when instructed to perform a two-arm task, $\pi_{0.5}$ often exhibits instability in that the nominally "idle" arm jitters or drifts. These results are not entirely surprising since the test settings are essentially out-of-distribution (OOD) for standard VLAs and this failure mode can be attributed to *entangled* bimanual control. Because $\pi_{0.5}$ uses a single shared representation to control both arms, each left–right pairing effectively corresponds to a distinct latent mode specifying a coupled bimanual action. Unseen pairings therefore require a latent that is not induced by any training demonstration, producing an OOD input to the downstream action expert.

TwinVLA attains non-zero success by occasionally producing the intended composition. However, although it uses separate VLM streams for the two arms, it relies on *uncontrolled* cross-attention between streams, which can introduce implicit coupling. This yields unpredictable interference under unseen pairings and keeps overall success low.

In contrast, SkillVLA achieves a substantially higher success rate of 51% on unseen combinations. By explicitly disentangling per-arm skill invocation, performing skill-adaptive action generation, and enabling *controlled* inter-arm communication only when needed, SkillVLA more reliably reuses the same single-arm behaviors under novel pairings while reducing spurious cross-arm interference.

B. Dual-Arm (Cooperative) Skill Reproduction

Task Setup. To evaluate whether SkillVLA can reproduce cooperative dual-arm skills that require close inter-arm coordination, we design three cooperative tasks (Fig. 4). These tasks fall into two categories with qualitatively different coordination demands. The first category, SHAKE and BALL, requires *tight low-level alignment* between the two arms. The second category, ALIGN, requires both low-level alignment and *high-level coordination*; two blocks are placed near the center of the table, and the two arms must simultaneously pick the objects and align them. Importantly, the arm-object assignment is randomized across trials, requiring the policy to resolve task allocation on the fly.

TABLE III: Success Rate in Cooperative Tasks ↑

Methods	Shake	Ball	Align	Avg.
$\pi_{0.5}$	**0.30**	0.45	0.65	0.47
TwinVLA	0.15	**0.55**	0.55	0.42
SkillVLA	0.25	0.50	**0.70**	**0.48**
SkillVLA w/o Attn.	0.00	0.10	0.40	0.17

Performance Results. As reported in Tab. III, SkillVLA achieves performance comparable to the strong baseline $\pi_{0.5}$ across all cooperative tasks. This suggests that SkillVLA can identify when cooperation is required and that its explicit communication mechanism is expressive enough to support tight inter-arm coordination when needed.

Ablation: Cross-Attention. We further isolate the role of action-level communication by evaluating an ablation that disables cross-attention between the two action experts. This variant performs particularly poorly on SHAKE and BALL, confirming that cross-attention-based communication is important for maintaining tight low-level alignment. On ALIGN, which places greater emphasis on high-level task assignment and is comparatively less sensitive to continuous low-level coupling, disabling communication results in a smaller performance drop.

C. Applications in Long-Horizon Settings

We next evaluate whether SkillVLA can (i) infer the required cooperation level and (ii) switch between collaborative and non-collaborative modes *within a single episode*.

Task setup. We design two long-horizon real-world tasks, TUBES and COLLECT ITEMS (Fig. 5). Each task intentionally interleaves phases that admit independent per-arm execution with phases that require explicit bimanual cooperation. As a representative example, TUBES consists of two stages with distinct coordination demands: (1) transferring tubes from one rack to another (independent per-arm collection), and (2) lifting and placing the loaded rack onto another rack (cooperative bimanual manipulation). We train the model using demonstrations of three skills, i.e., transferring a tube with the left arm, transferring a tube with the right arm, and moving the rack with both arms. We then evaluate the policy *zero-shot* on the full long-horizon task that composes these stages.

Performance and Efficiency Results. As shown in Fig. 5 (bottom right), SkillVLA achieves a progress score comparable to strong baselines, indicating that it can select an appropriate operating mode and complete the task reliably. More importantly, SkillVLA *reduces average completion time* by $\approx 21\%$ relative to the baselines. Qualitatively, monolithic baselines tend to follow demonstration-like sequential patterns (one arm acts while the other waits), whereas **SkillVLA** leverages *skill reuse* to exploit parallelism when available. In TUBES, during the tube-collection stage, it concurrently invokes the single-arm tube-transfer skills with both arms (i.e., $\alpha = 0$), enabling parallel left–right execution and substantially accelerating completion. This ability to opportunistically

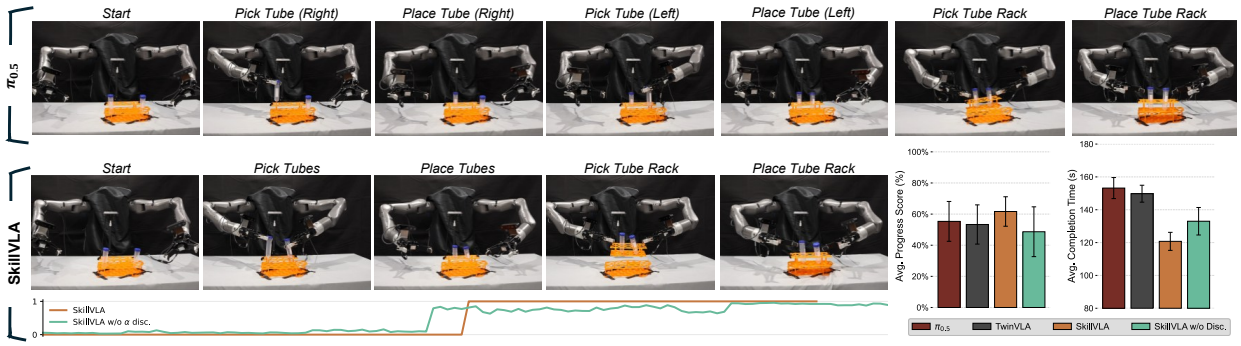

Fig. 5: **Long-horizon tasks behaviors and results. Top:** Behavior of $\pi_{0.5}$ in TUBES. **Middle left:** Behavior of SkillVLA in TUBES. **Bottom left:** Changes of α values throughout the completion, respectively from SkillVLA and an ablated version without discretization of α. **Bottom right:** Averaged progress score and completion time of methods on the long-horizon tasks.

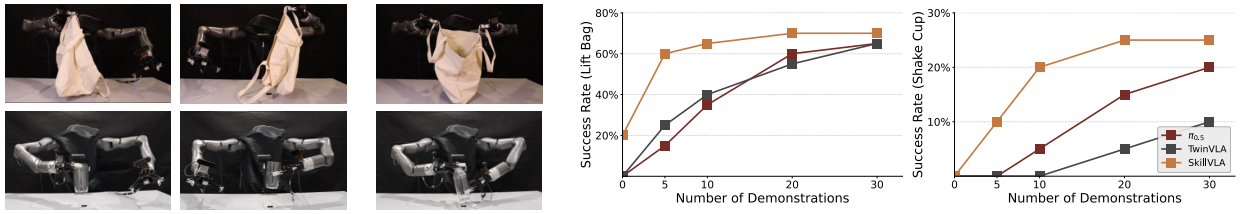

Fig. 6: **Continual learning tasks and results. Left:** SkillVLA in continual learning tasks, LIFT BAG and SHAKE CUP. **Right:** Success rates on cooperative tasks LIFT BAG and SHAKE CUP w.r.t the number of continual learning demonstrations.

parallelize subtasks better utilizes the robot's hardware and improves practical throughput.

Mode switching via the cooperation gate. Fig. 5 (bottom left) visualizes **SkillVLA**'s discrete gate output $\alpha \in \{0, 1\}$, where $\alpha = 0$ corresponds to independent per-arm execution and $\alpha = 1$ activates inter-arm cooperation. We observe that α evolves coherently with task progress; it switches to $\alpha = 1$ during inherently cooperative phases such as moving the loaded rack with both arms, while remaining at $\alpha = 0$ during collection phases where the arms can operate independently. This stage-consistent switching suggests that the gate can infer the required cooperation level from scene and instruction context, allowing SkillVLA to reuse both single-arm and dual-arm behaviors within a single rollout. In COLLECT ITEMS, the gate exhibits more complex transitions across multiple stages, further demonstrating the adaptability of our method to diverse coordination patterns.

Ablation: Continuous vs. Discretized Gating. We also evaluate an ablation that uses a continuous (non-discretized) gate. This variant is less stable over long horizons. As shown in Fig. 5 (bottom left), continuous α exhibits noticeably higher fluctuation despite following a similar overall trend. This instability propagates to action generation, leading to lower success and higher variance in performance (Fig. 5, bottom right). These results suggest that discretizing α improves robustness by enforcing simpler, clearer mode switching.

D. Reuse of Learned Skills in Continual Learning

Task setup. To answer Q4, we study whether SkillVLA can reuse learned single-arm skills to accelerate the acquisition of a new cooperative dual-arm skill via fine-tuning. As illustrated in Fig. 6 (left), we consider two task groups, LIFT BAG and SHAKE CUP. In each group, we first pretrain the model on the corresponding single-arm dataset (e.g., lifting a bag with one arm), and then fine-tune it with a small number of demonstrations to obtain the target dual-arm skill (e.g., lifting with both arms). For each task group, we construct a two-stage continual-learning protocol: (i) pretraining on single-arm executions to learn reusable per-arm behaviors, followed by (ii) fine-tuning on a limited set of dual-arm demonstrations to learn the cooperative variant.

Learning Curves and Data Efficiency. Fig. 6 (right) reports success rates as a function of the number of dual-arm demonstrations used for fine-tuning. In LIFT BAG, the target dual-arm behavior is close to a composition of the pretrained single-arm skills, and SkillVLA achieves 20% success *zero-shot*. Moreover, with only 5 fine-tuning demonstrations, it reaches near-peak performance. In contrast, $\pi_{0.5}$ and TwinVLA remain far from convergence even after training on 30 demonstrations. These results indicate that SkillVLA can effectively leverage previously learned single-arm behaviors for rapid adaptation.

Adapting Skills to New Coordination. SHAKE CUP is challenging because the required coordination deviates from the original single-arm behavior; one hand must stabilize the cap rather than shaking the cup body. Despite this mismatch, SkillVLA exhibits the same data-efficient trend, which suggests that reuse in continual learning is not limited to replaying identical action patterns. Instead, the model can transfer related single-arm behaviors as a strong initialization and quickly adapt them into a new cooperative policy. Overall, these results support skill reuse as a practical mechanism for continual learning and for scaling generalist bimanual policies with reduced demonstration burden.

VII. Conclusion, Limitations, and Future Work

In this work, we study the *combinatorial diversity* challenge in bimanual manipulation through the lens of *skill reuse*. We identify *skill entanglement* in existing VLA designs as a central obstacle to reuse—particularly for composing diverse left–right skill pairings and generalizing to unseen task combinations. To address this, we propose SkillVLA, which mitigates entanglement by explicitly separating per-arm skill invocation and enabling skill-adaptive action generation, while still leveraging the generalization of pretrained vision–language models. Across real-world evaluations, SkillVLA improves combinatorial generalization and data efficiency in continual skill acquisition, demonstrating more reliable reuse of both single-arm and cooperative dual-arm behaviors.

Limitations. SkillVLA depends on pretrained VLMs for high-level reasoning and for providing priors on cooperation, which ties performance to the capability, latency, and cost profile of the underlying foundation model. Future work will explore more efficient ways to distill or compress this reasoning into lightweight policy components, as well as richer skill representations beyond natural language. More broadly, we view disentangled skill learning/selection and controlled cross-arm coupling as key ingredients for scaling toward truly generalist bimanual agents that can recombine skills reliably on demand.

Acknowledgment

This work is supported by the Zhongguancun Academy (Grant No.c20250502). This research / project is partially supported by the National Research Foundation, Singapore, under its Thematic Competitive Research Programme 2025 (NRF-T-CRP-2025-0003).

References

[1] M Mahdi Ghazaei Ardakani, Martin Karlsson, Klas Nilsson, Anders Robertsson, and Rolf Johansson. Master-slave coordination using virtual constraints for a redundant dual-arm haptic interface. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8751–8757. IEEE, 2018.

[2] Yahav Avigal, Lars Berscheid, Tamim Asfour, Torsten Kröger, and Ken Goldberg. Speedfolding: Learning efficient bimanual folding of garments. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–8. IEEE, 2022.

[3] Arpit Bahety, Shreeya Jain, Huy Ha, Nathalie Hager, Benjamin Burchfiel, Eric Cousineau, Siyuan Feng, and Shuran Song. Bag all you need: Learning a generalizable bagging strategy for heterogeneous objects. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 960–967. IEEE, 2023.

[4] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

[5] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-vl technical report, 2025. URL https://arxiv.org/abs/2502.13923.

[6] Christian Bersch, Benjamin Pitzer, and Sören Kammel. Bimanual robotic cloth manipulation for laundry folding. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1413–1419. IEEE, 2011.

[7] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, Thomas Unterthiner, Daniel Keysers, Skanda Koppula, Fangyu Liu, Adam Grycner, Alexey Gritsenko, Neil Houlsby, Manoj Kumar, Keran Rong, Julian Eisenschlos, Rishabh Kabra, Matthias Bauer, Matko Bošnjak, Xi Chen, Matthias Minderer, Paul Voigtlaender, Ioana Bica, Ivana Balazevic, Joan Puigcerver, Pinelopi Papalampidi, Olivier Henaff, Xi Xiong, Radu Soricut, Jeremiah Harmsen, and Xiaohua Zhai. Paligemma: A versatile 3b vlm for transfer, 2024. URL https://arxiv.org/abs/2407.07726.

[8] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, Laura Smith, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A Vision-Language-Action Flow Model for General Robot Control. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.010.

[9] RC Bonitz and Tien C Hsia. Internal force-based impedance control for cooperating manipulators. *IEEE Transactions on Robotics and Automation*, 12(1):78–89, 1996.

[10] Alper Canberk, Cheng Chi, Huy Ha, Benjamin Burchfiel, Eric Cousineau, Siyuan Feng, and Shuran Song. Cloth

funnels: Canonicalized-alignment for multi-purpose garment manipulation. In *International Conference of Robotics and Automation (ICRA)*, 2022.

[11] Lawrence Yunliang Chen, Baiyu Shi, Roy Lin, Daniel Seita, Ayah Ahmad, Richard Cheng, Thomas Kollar, David Held, and Ken Goldberg. Bagging by learning to singulate layers using interactive perception. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3176–3183. IEEE, 2023.

[12] Lawrence Yunliang Chen, Baiyu Shi, Daniel Seita, Richard Cheng, Thomas Kollar, David Held, and Ken Goldberg. Autobag: Learning to open plastic bags and insert objects. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3918–3925, 2023. doi: 10.1109/ICRA48891.2023.10161402.

[13] Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Qiwei Liang, Zixuan Li, Xianliang Lin, Yiheng Ge, Zhenyu Gu, et al. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025.

[14] Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. In *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[15] Cheng Chi, Zhenjia Xu, Chuer Pan, Eric Cousineau, Benjamin Burchfiel, Siyuan Feng, Russ Tedrake, and Shuran Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. In *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[16] Rohan Chitnis, Shubham Tulsiani, Saurabh Gupta, and Abhinav Gupta. Efficient bimanual manipulation using learned task schemas. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1149–1155. IEEE, 2020.

[17] InternVLA-M1 Contributors. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy. *arXiv preprint arXiv:2510.13778*, 2025.

[18] Runyu Ding, Yuzhe Qin, Jiyue Zhu, Chengzhe Jia, Shiqi Yang, Ruihan Yang, Xiaojuan Qi, and Xiaolong Wang. Bunny-visionpro: Real-time bimanual dexterous teleoperation for imitation learning. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 12248–12255, 2025. doi: 10.1109/IROS60139.2025.11247017.

[19] Zipeng Fu, Tony Z. Zhao, and Chelsea Finn. Mobile ALOHA: Learning bimanual mobile manipulation using low-cost whole-body teleoperation. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=FO6tePGRZj.

[20] Dongsheng Ge, Huan Zhao, Dianxi Li, Dongchen Han, Xiangfei Li, Jiexin Zhang, and Han Ding. A dual-arm robotic cooperative framework for multiple peg-in-hole assembly of large objects. *Robotics and Computer-Integrated Manufacturing*, 95:102991, 2025.

[21] 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.

[22] Ce Hao, Xuanran Zhai, Yaohua Liu, and Harold Soh. Abstracting robot manipulation skills via mixture-of-experts diffusion policies. In *The Fourteenth International Conference on Learning Representations*, 2026.

[23] Samad Hayati, Kam Tso, and Thomas Lee. Dual arm coordination and control. *Robotics and Autonomous Systems*, 5(4):333–344, 1989.

[24] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[25] Hokyun Im, Euijin Jeong, Jianlong Fu, Andrey Kolobov, and Youngwoon Lee. Twinvla: Data-efficient bimanual manipulation with twin single-arm vision-language-action models, 2025. URL https://arxiv.org/abs/2511.05275.

[26] Physical Intelligence, Ali Amin, Raichelle Aniceto, Ashwin Balakrishna, Kevin Black, Ken Conley, Grace Connors, James Darpinian, Karan Dhabalia, Jared DiCarlo, Danny Driess, Michael Equi, Adnan Esmail, Yunhao Fang, Chelsea Finn, Catherine Glossop, Thomas Godden, Ivan Goryachev, Lachy Groom, Hunter Hancock, Karol Hausman, Gashon Hussein, Brian Ichter, Szymon Jakubczak, Rowan Jen, Tim Jones, Ben Katz, Liyiming Ke, Chandra Kuchi, Marinda Lamb, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Yao Lu, Vishnu Mano, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Charvi Sharma, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, Will Stoeckle, Alex Swerdlow, James Tanner, Marcel Torne, Quan Vuong, Anna Walling, Haohuan Wang, Blake Williams, Sukwon Yoo, Lili Yu, Ury Zhilinsky, and Zhiyuan Zhou. $\pi^*_{0.6}$: a vla that learns from experience, 2025. URL https://arxiv.org/abs/2511.14759.

[27] Physical Intelligence, Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization, 2025. URL https://arxiv.org/abs/2504.16054.

[28] Jaehyung Kim, Jiho Kim, Dongryung Lee, Yujin Jang,

and Beomjoon Kim. A low-cost and lightweight 6 DoF bimanual arm for dynamic and contact-rich manipulation. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.055.

[29] 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.

[30] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=PqvMRDCJT9t.

[31] 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. In *The Thirteenth International Conference on Learning Representations*, 2025.

[32] Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=XVjTT1nw5z.

[33] Guanxing Lu, Tengbo Yu, Haoyuan Deng, Season Si Chen, Yansong Tang, and Ziwei Wang. Anybimanual: Transferring unimanual policy for general bimanual manipulation, 2025. URL https://arxiv.org/abs/2412.06779.

[34] Teli Ma, Jia Zheng, Zifan Wang, Chunli Jiang, Andy Cui, Junwei Liang, and Shuo Yang. Dit4dit: Jointly modeling video dynamics and actions for generalizable robot control. *arXiv preprint arXiv:2603.10448*, 2026.

[35] Jeremy Maitin-Shepard, Marco Cusumano-Towner, Jinna Lei, and Pieter Abbeel. Cloth grasp point detection based on multiple-view geometric cues with application to robotic towel folding. In *2010 IEEE International Conference on Robotics and Automation*, pages 2308–2315. IEEE, 2010.

[36] Tomohiro Motoda, Ryo Hanai, Ryoichi Nakajo, Masaki Murooka, Floris Erich, and Yukiyasu Domae. Learning bimanual manipulation via action chunking and inter-arm coordination with transformers, 2025. URL https://arxiv.org/abs/2503.13916.

[37] Davide Ortenzi, Rajkumar Muthusamy, Alessandro Freddi, Andrea Monteriù, and Ville Kyrki. Dual-arm cooperative manipulation under joint limit constraints. *Robotics and Autonomous Systems*, 99:110–120, 2018.

[38] Georgios Papagiannis, Norman Di Palo, Pietro Vitiello, and Edward Johns. R+ x: Retrieval and execution from everyday human videos. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8284–8290. IEEE, 2025.

[39] H Andy Park and CS George Lee. Extended cooperative task space for manipulation tasks of humanoid robots. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6088–6093. IEEE, 2015.

[40] 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. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.012.

[41] Gautam Salhotra, I Liu, Gaurav Sukhatme, et al. Learning robot manipulation from cross-morphology demonstration. *arXiv preprint arXiv:2304.03833*, 2023.

[42] STANLEYA Schneider and Robert H Cannon. Object impedance control for cooperative manipulation: Theory and experimental results. In *Proceedings, 1989 international conference on robotics and automation*, pages 1076–1083. IEEE, 1989.

[43] Kenneth Shaw, Yulong Li, Jiahui Yang, Mohan Kumar Srirama, Ray Liu, Haoyu Xiong, Russell Mendonca, and Deepak Pathak. Bimanual dexterity for complex tasks. In *8th Annual Conference on Robot Learning*, 2024. URL https://openreview.net/forum?id=55tYfHvanf.

[44] Christian Smith, Yiannis Karayiannidis, Lazaros Nalpantidis, Xavi Gratal, Peng Qi, Dimos V. Dimarogonas, and Danica Kragic. Dual arm manipulation—a survey. *Robotics and Autonomous Systems*, 60(10):1340–1353, 2012. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2012.07.005. URL https://www.sciencedirect.com/science/article/pii/S092188901200108X.

[45] Wenxuan Song, Jiayi Chen, Pengxiang Ding, Han Zhao, Wei Zhao, Zhide Zhong, Zongyuan Ge, Jun Ma, and Haoang Li. Accelerating vision-language-action model integrated with action chunking via parallel decoding. *arXiv preprint arXiv:2503.02310*, 2025.

[46] Markku Suomalainen, Sylvain Calinon, Emmanuel Pignat, and Ville Kyrki. Improving dual-arm assembly by master-slave compliance. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8676–8682. IEEE, 2019.

[47] RDT Team. Rdt2: Enabling zero-shot cross-embodiment generalization by scaling up umi data, September 2025. URL https://github.com/thu-ml/RDT2.

[48] Masaru Uchiyama and Pierre Dauchez. A symmetric hybrid position/force control scheme for the coordination of two robots. In *Proceedings. 1988 IEEE international conference on robotics and automation*, pages 350–356. IEEE, 1988.

[49] Christos K Verginis, Daniel Zelazo, and Dimos V Dimarogonas. Cooperative manipulation via internal force regulation: A rigidity theory perspective. *IEEE Transactions on Control of Network Systems*, 10(3):1222–1233, 2022.

[50] Chen Wang, Haochen Shi, Weizhuo Wang, Ruohan Zhang, Li Fei-Fei, and Karen Liu. DexCap: Scalable and Portable Mocap Data Collection System for Dexterous Manipulation. In *Proceedings of Robotics: Science and Systems*, Delft, Netherlands, July 2024. doi: 10.15607/RSS.2024.XX.043.

[51] Yixiao Wang, Yifei Zhang, Mingxiao Huo, Ran Tian, Xiang Zhang, Yichen Xie, Chenfeng Xu, Pengliang Ji, Wei Zhan, Mingyu Ding, et al. Sparse diffusion policy: A sparse, reusable, and flexible policy for robot learning. *arXiv preprint arXiv:2407.01531*, 2024.

[52] Junjie Wen, Yichen Zhu, Jinming Li, Zhibin Tang, Chaomin Shen, and Feifei Feng. Dexvla: Vision-language model with plug-in diffusion expert for general robot control. *arXiv preprint arXiv:2502.05855*, 2025.

[53] 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.

[54] Thomas Weng, Sujay Man Bajracharya, Yufei Wang, Khush Agrawal, and David Held. Fabricflownet: Bimanual cloth manipulation with a flow-based policy. In *Conference on Robot Learning*, pages 192–202. PMLR, 2022.

[55] Fan Xie, Alexander Chowdhury, M De Paolis Kaluza, Linfeng Zhao, Lawson Wong, and Rose Yu. Deep imitation learning for bimanual robotic manipulation. *Advances in neural information processing systems*, 33: 2327–2337, 2020.

[56] Jingyun Yang, Ziang Cao, Congyue Deng, Rika Antonova, Shuran Song, and Jeannette Bohg. Equibot: SIM(3)-equivariant diffusion policy for generalizable and data efficient learning. In *8th Annual Conference on Robot Learning*, 2024.

[57] Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies, 2026. URL https://arxiv.org/abs/2602.15922.

[58] Jiawen Yu, Hairuo Liu, Qiaojun Yu, Jieji Ren, Ce Hao, Haitong Ding, Guangyu Huang, Guofan Huang, Yan Song, Panpan Cai, Cewu Lu, and Wenqiang Zhang. Forcevla: Enhancing vla models with a force-aware moe for contact-rich manipulation, 2025. URL https://arxiv.org/abs/2505.22159.

[59] Samson Yu, Lin Kelvin, and Harold Soh. Demonstrating the Octopi-1.5 Visual-Tactile-Language Model. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.058.

[60] Tianyuan Yuan, Zibin Dong, Yicheng Liu, and Hang Zhao. Fast-wam: Do world action models need test-time future imagination?, 2026. URL https://arxiv.org/abs/2603.16666.

[61] Xuanran Zhai, Qianyou Zhao, Qiaojun Yu, and Ce Hao. Vfp: Variational flow-matching policy for multi-modal robot manipulation, 2025. URL https://arxiv.org/abs/2508.01622.

[62] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. In *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023. doi: 10.15607/RSS.2023.XIX.016.

[63] Tony Z. Zhao, Jonathan Tompson, Danny Driess, Pete Florence, Seyed Kamyar Seyed Ghasemipour, Chelsea Finn, and Ayzaan Wahid. Aloha unleashed: A simple recipe for robot dexterity. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 1910–1924. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/zhao25b.html.

[64] 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.

[65] Ruijie Zheng, Jing Wang, Scott Reed, Johan Bjorck, Yu Fang, Fengyuan Hu, Joel Jang, Kaushil Kundalia, Zongyu Lin, Loic Magne, Avnish Narayan, You Liang Tan, Guanzhi Wang, Qi Wang, Jiannan Xiang, Yinzhen Xu, Seonghyeon Ye, Jan Kautz, Furong Huang, Yuke Zhu, and Linxi Fan. Flare: Robot learning with implicit world modeling, 2025. URL https://arxiv.org/abs/2505.15659.

[66] Huayi Zhou, Ruixiang Wang, Yunxin Tai, Yueci Deng, Guiliang Liu, and Kui Jia. You Only Teach Once: Learn One-Shot Bimanual Robotic Manipulation from Video Demonstrations. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025. doi: 10.15607/RSS.2025.XXI.149.

[67] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, Quan Vuong, Vincent Vanhoucke, Huong Tran, Radu Soricut, Anikait Singh, Jaspiar Singh, Pierre Sermanet, Pannag R. Sanketi, Grecia Salazar, Michael S. Ryoo, Krista Reymann, Kanishka Rao, Karl Pertsch, Igor Mordatch, Henryk Michalewski, Yao Lu, Sergey Levine, Lisa Lee, Tsang-Wei Edward Lee, Isabel Leal, Yuheng Kuang, Dmitry Kalashnikov, Ryan Julian, Nikhil J. Joshi, Alex Irpan, Brian Ichter, Jasmine Hsu, Alexander Herzog, Karol Hausman, Keerthana Gopalakrishnan, Chuyuan Fu, Pete Florence, Chelsea Finn, Kumar Avinava Dubey, Danny Driess, Tianli Ding, Krzysztof Marcin Choromanski, Xi Chen, Yevgen Chebotar, Justice Carbajal, Noah Brown, Anthony Brohan, Montserrat Gonzalez Arenas, and Kehang Han. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In Jie Tan, Marc Toussaint, and Kourosh Darvish, editors, *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pages 2165–2183. PMLR, 06–09 Nov 2023.