Training VLAs with Vision-Language Anchoring and Language-Action Alignment

Dwip Dalal[1] Shivansh Patel[1] Jeonghwan Kim[1] Utkarsh Mishra[2] Alex Baratian[1] Hyeonjeong Ha[1] Heng Ji[1] Svetlana Lazebnik[1] Unnat Jain[3]

[1]UIUC, [2]Texas AM University, [3]UCI

Abstract. Finetuning pretrained vision-language models (VLMs) on robot demonstrations has become the standard path to capable Vision-Language-Action (VLA) policies, but we show that this process introduces two concrete failure modes. First, action finetuning causes *representation erosion*: gradient updates progressively overwrite pretrained VLM representations, degrading vision-language reasoning across multiple axes. Second, because language and action are always supervised on separate observations, the same model can produce contradictory outputs on the same robot observation, reporting "left" while executing an upward motion. We propose Anchor-and-Align, a finetuning recipe that addresses both. *Vision-Language Anchoring* keeps a frozen copy of the pretrained VLM as an anchor and applies a layer-wise distillation loss to preserve pretrained representations throughout finetuning. *Language-Action Alignment* uses `ACL` (Action-Consistent Language), a framework that programmatically converts robot demonstrations into synchronized language-action supervision with no human annotation, and jointly trains the model on action prediction and action-grounded language targets on the same observations. Anchor-and-Align improves task success on LIBERO-Pro and LIBERO-Plus stress tests and transfers consistently to a real robot setup, while mechanistic analysis confirms that pretrained VLM representations are preserved and action decodability improves, explaining the generalization gains.

1 Introduction

Vision-language models (VLMs) are strong generalists. Models trained on internet-scale data identify objects, reason about spatial relationships, follow complex instructions, and transfer across visual domains without task-specific training. By finetuning a pretrained VLM to generate continuous control outputs (3D end-effector position, rotation, and gripper state) rather than words, one obtains a Vision-Language-Action (VLA) model that inherits this generalization capability for robot manipulation. This approach has become a large and active research area. A particularly effective recipe has emerged: directly fine-tune a pretrained VLM on downstream task demonstrations [32], bypassing large-scale cross-embodiment robot pretraining that earlier approaches required [17,40]. We study what this finetuning process does to the VLM and how to do it better.

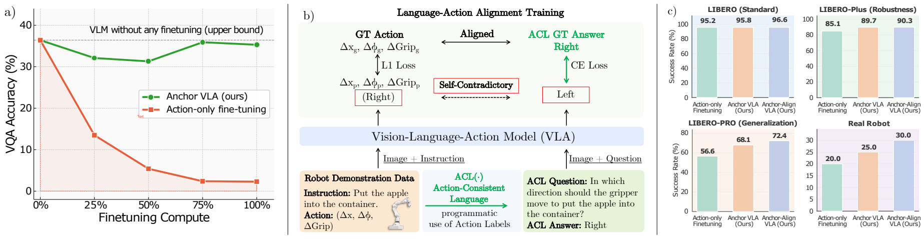


Fig. 1: Two failure modes in VLM-to-VLA finetuning addressed by Anchor-and-Align. *(a)* Action finetuning causes representation erosion. Vision-Language Anchoring in Anchor-and-Align prevents this, preserving pretrained VLM knowledge throughout training. *(b)* VLAs exhibit language-action inconsistency on the same robot observation. ACL measures this, and Language-Action Alignment resolves it. *(c)* Anchor-and-Align consistently improves task success across LIBERO, LIBERO-Pro, LIBERO-Plus, and a real xArm7 robot setup.

As shown in Fig. 1a, finetuning on robot action prediction substantially degrades vision-language reasoning capability. We probe this across multiple axes: VQA accuracy, motion semantics, spatial grounding, and orientation reasoning all decline. The cause is *representation erosion*: action finetuning shifts the model's internal representations away from those established during VLM pretraining. The VLM backbone is left unfrozen to adapt to robot observations, and it does adapt, but gradient updates focused on action prediction progressively overwrite the pretrained VLM representations that support generalization.

We address representation erosion through *Vision-Language Anchoring*: keeping a frozen copy of the original pretrained VLM (the *anchor*) and applying a layer-wise distillation loss that constrains the policy's hidden states to remain close to the anchor's. This objective requires no additional data or architectural changes, and outperforms other knowledge preservation baselines.

finetuning also introduces a second failure mode: language-action inconsistency, illustrated in Fig. 1b. On the same robot observation, the action output and language output can directly contradict: the action encodes upward motion while the model outputs "left," or the gripper remains open while the model reports task completion. Co-trained VLAs, which jointly train on generic internet VQA data and robot actions specifically to retain language capability, exhibit the same problem. Language and action are supervised on separate observations throughout training, so the two output heads have no mechanism to agree with each other. Co-training preserves language accuracy on generic benchmarks without resolving the inconsistency on robot observations. To our knowledge, no prior work measures language-action agreement on the same robot observation, making this inconsistency invisible to existing evaluation protocols.

We address this by deriving synchronized language supervision from the downstream task demonstrations already used for action training. We programmatically convert action labels into discrete language targets covering motion direction, grasp state, and task completion, each paired with the corresponding action on the same observation. We call this framework ACL (Action-Consistent Language). *Language-Action Alignment*, which trains jointly on continuous action prediction and these action-grounded language targets, resolves the inconsistency and improves downstream task performance, establishing language-action agreement as a reliable indicator of policy quality.

We evaluate on LIBERO, LIBERO-Pro, and LIBERO-Plus (recently proposed variants that stress-test generalization beyond what standard LIBERO reveals), and on a physical xArm7 robot manipulation setup (Fig. 1c). Vision-Language Anchoring and Language-Action Alignment together form **Anchor-and-Align**, improving task success by 5–15% on LIBERO-Pro and LIBERO-Plus over action-only finetuning, with consistent gains in real-world transfer.

In summary, this paper contributes:

(1) **A characterization of representation erosion in VLA finetuning.** We show that action finetuning degrades pretrained VLM representations across multiple axes, and that this degradation persists even in co-trained VLAs.

(2) **Vision-Language Anchoring**, a layer-wise distillation from a frozen VLM copy that prevents representation erosion with minimal overhead, outperforming knowledge insulation [10] and other baselines.

(3) `ACL` (Action-Consistent Language), a programmatic framework that converts any robot demonstration into synchronized language-action pairs with zero human annotation. `ACL` serves two roles: as an evaluation protocol enabling the first systematic measurement of language-action agreement in VLAs, and as a supervision source for the alignment objective in Anchor-and-Align.

(4) **Language-Action Alignment via `ACL` supervision**, which jointly trains continuous action prediction and action-grounded language prediction on the same observations, eliminating self-contradiction and improving generalization. Combined with Vision-Language Anchoring, Anchor-and-Align improves task success by 5–15% on LIBERO-Pro & Plus, quadruples performance under position-swap perturbation (24.4% vs. 6.1%), and transfers to real-world xArm7 manipulation.

2 Related Works

General-Purpose VLAs for Robot Manipulation. VLAs leverage pretrained VLMs as foundation policies for generalist manipulation, adapting to downstream tasks with minimal finetuning [4, 9, 11, 16, 19, 27, 28, 31]. RT-2 [40] co-fine-tunes a VLM on robot trajectories and shows web-scale VL pretraining yields emergent generalization to novel objects and compositional reasoning. OpenVLA [17] is trained on real-robot demos from Open X-Embodiment [24], demonstrating the effectiveness of large-scale cross-embodiment training. Architectural variants include flow-matching continuous actions (π_0, $\pi_{0.5}$) [2, 10], a frozen-VLM + diffusion-transformer dual system for humanoids [1], and a billion-parameter diffusion expert with curriculum learning [33]. VLA-Adapter [32] challenges this reliance on large-scale robotic pre-training entirely: by introducing a lightweight Bridge Attention mechanism, it turns any off-the-shelf VLM into a VLA, achieving strong performance. Unlike previous VLA works, our work investigates which finetuning recipe best preserves the VLM representations.

Co-training in VLAs. Co-training on a mixture of robot and generic vision-language data has emerged as the dominant strategy to prevent VL catastrophic forgetting while finetuning a VLM on robot action data [18, 26, 33, 35]. Magma [35] introduces the Set-of-Mark objective to better ground actions within multimodal representations. MolmoAct [18] adds depth-aware perception tokens and editable waypoint traces to support low-level action prediction. ECoT [36] trains VLAs to produce explicit chain-of-thought plans and grounding before acting. ChatVLA [38, 39] uses phased alignment and mixture-of-experts to reduce catastrophic forgetting during control training. These works share a common structure: language preservation is treated as a secondary objective, addressed by mixing in external VQA data or adding modality-specific losses drawn from sources separate from robotics. None of these prevent the drift of the source by constraining the VLM backbone, nor do they derive language supervision directly from the action trajectories themselves. Anchor-and-Align addresses both: it anchors the backbone to a frozen pretrained VLM via layer-wise distillation to prevent drift, and it constructs language targets programmatically from the same demonstrations used for action training, explicitly tying language and control to the same observations rather than to separate data streams.

Embodied Question Answering. Evaluating multimodal models in physically grounded tasks has a long history, from navigation-centric question answering [8] to open-world embodied reasoning with foundation models [23]. These works establish a principle: language understanding and physical grounding must be evaluated together. Recent efforts bring this principle to robotic manipulation, constructing question-answering benchmarks from real-world robot datasets [6, 26], probing whether VLMs maintain coherent internal world models during interaction [13]. Despite this progress, a critical gap persists specifically for VLAs. Existing co-trained VLAs [35, 39, 40] evaluate retained language capability on generic benchmarks such as TextVQA [30], which use web-crawled images far removed from robot observations. These evaluations measure whether language survives co-training in isolation, but never whether the model's language predictions are consistent with its own action predictions on the same robot observation. We introduce `ACL` to close this gap: it converts any robot demonstration into synchronized language-action evaluation pairs and measures language-action agreement as an important metric, revealing misalignment in co-trained VLAs that generic VQA metrics entirely miss.

3 Anchor-and-Align

We present Anchor-and-Align (Fig. 2), a language-action alignment finetuning recipe for adapting a pretrained VLM into a continuous action policy while preserving its pretrained VL representations. We use VLA-Adapter as our backbone (Sec. 3.1). Anchor-and-Align operates through two objectives: (i) layer-wise anchoring to the frozen VLM to retain the visual-language understanding ability of the backbone VLM (Sec. 3.2), and (ii) a language alignment objective that first derives discrete language labels from the action trajectories using `ACL` (Sec. 3.3)

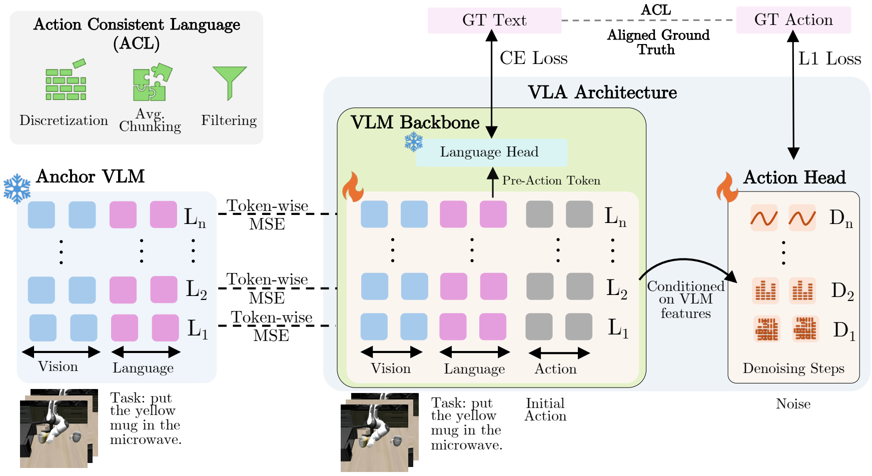


Fig. 2: Anchor-and-Align. During VLA training, we use a frozen VLM as an anchor to prevent the VLM backbone's representations from drifting. We align language and action generation using ACL-derived action-aligned language–action ground-truth targets. We apply a token-wise MSE loss across transformer layers to match the backbone's vision-language representations to the anchor VLM.

and then supervises the backbone to predict them (Sec. 3.4). Both objectives are jointly optimized alongside the primary action loss (Sec. 3.5).

3.1 Background on Base VLA Architecture

We build on the architecture of VLA-Adapter [32], a lightweight framework that turns any off-the-shelf VLM into a continuous action policy without large-scale robotic pre-training. Unlike other VLAs [1, 5, 17, 39], it does not require pre-training on massive robotics data before it can be finetuned to a downstream robotics task. Its bridge attention mechanism identifies the VL conditions most informative for control, and injects them into the action space via cross-attention rather than co-training the entire VLM on robot data. During finetuning, only LoRA parameters and the action head are updated. Despite this efficiency, action-centric LoRA finetuning induces representation erosion: the backbone's internal VL abstractions erode, degrading the pretrained knowledge that underlies generalization (empirically demonstrated in Sec. 4.5).

3.2 Anchoring VLM Representations

To prevent the representation erosion, we anchor the backbone VLM of the VLA with a frozen copy of the same VLM, initialized with pretrained weights, that processes the same input batch in parallel. This continues distilling the original pretrained VLM's knowledge into the backbone VLM of the VLA. We align hidden states between the backbone and this anchor at every transformer layer $\ell \in 1, \ldots, 24$, excluding the embedding layer. Alignment is restricted to *non-action* positions, comprising vision patches and text tokens, with all action token positions masked out. Let $\mathbf{m}$ denote the set of non-action positions. The per-layer anchoring loss is $\mathcal{L}_{\text{anchor}}^{(\ell)} = \left\|\mathbf{H}_\ell^S[\mathbf{m}] - \mathbf{H}_\ell^A[\mathbf{m}]\right\|_2^2$, where $\mathbf{H}_\ell^S \in \mathbb{R}^{N \times d}$ and $\mathbf{H}_\ell^A \in \mathbb{R}^{N \times d}$ are the backbone and anchor VLM hidden states at layer ℓ, respectively. The total anchoring loss averages over all L layers: $\mathcal{L}_{\text{anchor}} = \frac{1}{L}\sum_{\ell=1}^{L} \mathcal{L}_{\text{anchor}}^{(\ell)}$. This per-layer MSE constraint prevents the representational erosion that unconstrained LoRA action finetuning otherwise causes.

3.3 Deriving Language Labels from Robot Actions with ACL

The alignment loss uses language supervision derived directly from ground-truth action trajectories. Specifically, for each training sample, ACL automatically assigns a discrete language target derived from the action label at that training instance. First, ACL creates a set of language-based direction labels W: $\{$*up*, *down*, *left*, *right*, *front*, or *back*$\}$ that are easily interpretable by the VLM. Then, ACL maps these labels to the action ground truth. ACL does this via a 3-step process: average chunking, filtering, and discretization.

We perform average chunking to reduce noise in the action ground truth. Here, we average the translational components (x, y, z) of the action tokens across the chunk dimension K. Given chunked actions $\mathbf{A} \in \mathbb{R}^{B \times K \times 7}$, the mean translation for each sample in the batch is $\bar{\mathbf{v}} = \frac{1}{K}\sum_{k=1}^{K} \mathbf{A}_{:,k,1:3} \in \mathbb{R}^{B \times 3}$. Then, we filter out near-stationary samples, since a direction label is not meaningful when the robot barely moves. Any sample i satisfying $\|\bar{\mathbf{v}}_i\|_2 < \tau_{\text{dir}}$ is masked out, where τ_{dir} is a motion-threshold hyperparameter. Finally, we discretize the remaining samples. We assign a direction label by identifying the dominant axis, $j^\star = \arg\max_{j \in \{x,y,z\}} |\bar{v}_{i,j}|$, and using the sign of $\bar{v}_{i,j^\star}$ to select one of the six direction words. This produces a discrete language label for each valid trajectory that can be used as a language supervision signal in the alignment loss.

3.4 Aligning Language With Action

Given the images and instruction, the model should be able to predict what direction the robot is about to move, expressed as a real English word. Requiring this prediction to be natural language forces the pre-action representation to remain language-grounded while remaining informative for downstream action decoding. Supervising this position also sends gradients through both the vision and language pathways without directly manipulating vision patch embeddings.

To tie such language representations to the physical action space, we supervise a single token that sits at the exact boundary where language ends and action generation begins. The hidden state of the last text token, immediately preceding the first action token, satisfies both requirements via causal self-attention: it attends to all vision patches and all text tokens, making it a natural summary of everything the model has observed before producing an action. Let t_i^{act} be the index of the first action token in sample i within the text-and-action segment, and let V be the vision prefix length. The pre-action position and its hidden state are $t_i^{\text{pre}} = V + t_i^{\text{act}} - 1, \text{and } \mathbf{h}_i^{\text{pre}} = \mathbf{H}_L^S\left[t_i^{\text{pre}}\right] \in \mathbb{R}^d$, where $\mathbf{H}_L^S$ is the backbone VLM's last-layer hidden state. We then project $\mathbf{h}_i^{\text{pre}}$

into the language embedding space and apply the frozen pretrained language head:

$$\mathbf{z}_i = \mathbf{W}_{\text{proj}}\, \mathbf{h}_i^{\text{pre}} \in \mathbb{R}^d, \qquad \boldsymbol{\ell}_i = \mathbf{W}_{\text{lm}}\, \mathbf{z}_i \in \mathbb{R}^{|\mathcal{V}|}, \tag{1}$$

where $\mathbf{W}_{\text{proj}} \in \mathbb{R}^{d\times d}$ is a learned projection and $\mathbf{W}_{\text{lm}} \in \mathbb{R}^{|\mathcal{V}|\times d}$ is the frozen LM head weight matrix. The alignment loss is then the cross-entropy against the direction label $\mathbf{y}$ derived in Section 3.3: $\mathcal{L}_{\text{align}} = \text{CE}(\boldsymbol{\ell}, \mathbf{y})$.

3.5 Training Details

The primary action loss $\mathcal{L}_{\text{action}}$ is ℓ_1 regression between the predicted and the ground-truth robot actions across denoising steps, consistent with [32]), for a headon comparison. Anchoring loss $\mathcal{L}_{\text{anchor}}$ is defined in Sec. 3.2 and the alignment loss $\mathcal{L}_{\text{align}}$ in Sec. 3.4. The three terms are jointly optimized as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{action}} + \lambda_{\text{anchor}}\, \mathcal{L}_{\text{anchor}} + \lambda_{\text{align}}\, \mathcal{L}_{\text{align}} \tag{2}$$

We update only the backbone's LoRA parameters, the action head, and $\mathbf{W}_{\text{proj}}$. The anchor VLM and the pretrained LM head $\mathbf{W}_{\text{lm}}$ remain frozen.

Implementation Details. We use Qwen2.5-0.5B [34] and fine-tune it with LoRA with rank $r = 64$, applied on all layers. The input sequence is vision + text + action tokens. The 512 vision patches are obtained from two images. We extract DINOv2 and SigLIP features for each image and concatenate them along the feature dimension per patch, producing a single richer patch embedding. Each image yields 256 patches; with two input images, the full vision prefix is 512 patches in total. The backbone VLM forward pass runs once, and the resulting hidden states are shared across all loss terms to avoid redundant computation. Additional details about hyperparameters and other design choices have been described in detail in App. **??**.

4 Experiments

We organize our experimental evaluation as follows: Sec. 4.1 describes the benchmarks and baselines. Sec. 4.2 presents quantitative results on LIBERO, LIBERO-PLUS (robustness), and LIBERO-PRO (generalization) benchmarks. Sec. 4.3 validates that the gains transfer to a physical robot. Sec. 4.4 introduces the `ACL` diagnostic framework, which measures language–action alignment in state-of-the-art co-trained VLA models. Finally, Sec. 4.5 provides ablations and representational analyses that explain why anchoring and alignment improve performance and generalization.

4.1 Benchmarks and Baselines

Baselines. We compare against 12 baselines spanning non-VLM policies, VLM-based VLAs with standard action finetuning, and methods that explicitly address knowledge preservation during finetuning.

• **Non-VLM and standard VLM-based VLAs.** We compare against Diffusion Policy [7] (non-VLM diffusion-based control), π_0-FAST [2,25] (DCT-based action tokenization), SmolVLA [29] (flow-matching action expert), OpenVLA-OFT [16] (parallel-decoded continuous actions), VLA-0 [14] (plain-text action integers), and VLA-Adapter [32] (Qwen2.5 [34] with LoRA), which serves as our backbone.

- **Knowledge-preservation approaches.** $\pi_{0.5}$-KI [10] addresses catastrophic forgetting by training discrete FAST action tokens on the VLM backbone while stop-gradient prevents action-gradients from corrupting pretrained representations. VLA-Adapter[Frozen] [32] represents the extreme preservation baseline: the VLM backbone is entirely frozen and only the action head is trained, trivially preserving all pretrained knowledge at the cost of reduced task performance.
- **Co-training baseline.** MolmoAct [18] augments a VLM with depth-aware perception tokens and image-space waypoint planning for 3D spatial reasoning. We further implement Knowledge Insulation [10] on top of VLA-Adapter. Following Driess et al. [10], we apply a stop-gradient operation to prevent action gradients from updating the VLM backbone, and jointly train on VQA data (VSR [22], GQA [15], and COCO [20]) alongside action finetuning with LoRA, maintaining the language modeling head for VQA forward passes while using the L1 regression action head for robot data. This baseline tests whether shielding the VLM from action gradients and co-training on language data is sufficient to prevent the representational drift we observe in action-only finetuning.

Benchmarks. We evaluate on the LIBERO simulation suite [21] and its two recent stress-test extensions, LIBERO-PRO [37] and LIBERO-Plus [12]. Together, these three benchmarks span in-distribution task completion [21], semantic generalization under perturbations [37], and perceptual robustness under deployment-realistic distributional shift [12].

- **LIBERO** [21] is a tabletop manipulation benchmark with four suites of 10 tasks each—Spatial, Object, Goal, and Long—that isolate spatial reasoning, object knowledge, procedural knowledge, and long-horizon composition, respectively. Test initial states are drawn from the same distribution as training, so LIBERO measures *in-distribution generalization*: the ability to reliably reproduce learned behaviors under familiar conditions.
- **LIBERO-PRO** [37] tests *semantic generalization* via three controlled out-of-distribution perturbations: language rephrase (paraphrased instructions), object swap (visually distinct instances from the same category), and position swap (rearranged spatial layouts). Since standard LIBERO conditions are near-identical to training, policies can score above 95% by memorizing fixed motor programs; LIBERO-PRO exposes this by requiring genuine linguistic grounding, visual generalization, and compositional spatial reasoning. Position swap (Fig. 3) is the hardest axis: strong VLAs, including OpenVLA-OFT and Molmoact, score 0%.
- **LIBERO-Plus** [12] tests *perceptual and physical robustness* across seven deployment-realistic perturbation axes: camera viewpoint, lighting, background texture, object layout, robot initial state, language instruction, and sensor noise. These directly mirror real-world variability such as camera mounting imprecision, workspace lighting changes, and sensor degradation, making LIBERO-Plus the closest proxy for sim-to-real transfer.

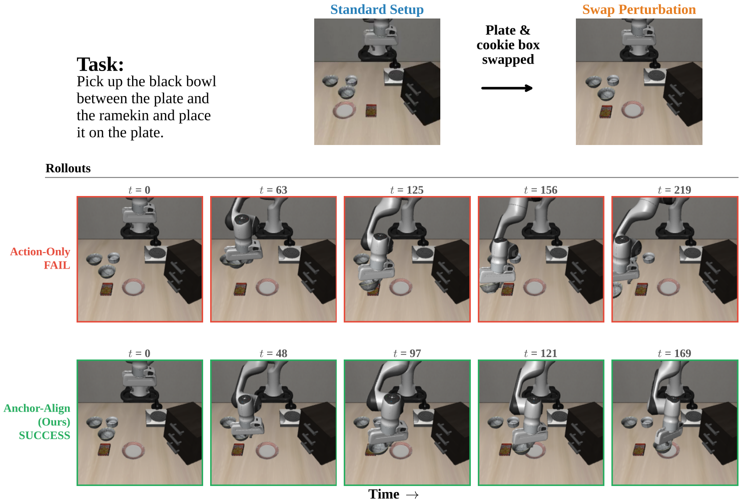

Standard Setup
Swap Perturbation
Plate & cookie box swapped
Task:
Pick up the black bowl between the plate and the ramekin and place it on the plate.
Rollouts
t = 0
t = 63
t = 125
t = 156
t = 219
Action-Only FAIL
t = 0
t = 48
t = 97
t = 121
t = 169
Anchor-Align (Ours) SUCCESS
Time →

Fig. 3: Qualitative comparison under position-swap perturbation. Top: the standard and swap configurations for the task *"pick up the black bowl between the plate and the ramekin and place it on the plate"*; the plate and cookie box are swapped relative to training. Bottom: rollout trajectories. Action-Only Finetuning (red) fails to adapt to the rearranged layout, while Anchor-Align VLA (green) correctly identifies the target bowl and completes the pick-and-place despite the unseen object positions.

4.2 Quantitative Results

In this section, we present the main empirical results of the paper. Additional results and supporting analyses are provided in the supplementary (App.**??**).

Anchor-and-Align improves generalization and robustness. Both Anchor-Align VLA and Anchor VLA demonstrate strong performance on the generalization and robustness tests of LIBERO-PRO and LIBERO-Plus. This confirms that our method successfully improves generalization by preserving VLM knowledge and aligning language with action. Anchor-Align VLA surpasses the strongest prior method (VLA-Adapter [32]) by approximately 6% in overall LIBERO-Plus success rate (90.3% vs. 85.1%). The gains are most pronounced on the hardest perturbation categories.

Anchor-and-Align significantly improves language understanding. The language-sensitive axes of both benchmarks reveal the largest gains from our method. On LIBERO-PRO language rephrase, Anchor-Align VLA achieves 96.6% and Anchor VLA achieves 95.4%, compared to 74.2% for VLA-Adapter and 74.4% for OpenVLA-OFT, an improvement of over 22% over the best prior action-only finetuned model. Even MolmoAct, a 7B model with depth-aware perception and co-training, reaches only 77.8%. The same trend holds on LIBERO-Plus language instruction, where Anchor VLA scores 90.5% and Anchor-Align VLA scores 87.2%, surpassing VLA-Adapter (85.1%), OpenVLA-OFT (81.5%), and MolmoAct (79.5%). These consistent gains across both semantic rephrasing and instruction-level perturbations demonstrate that anchoring the VLM backbone preserves the pretrained linguistic representations that action-only finetun-

	LIBERO-PRO			**LIBERO-Plus**							
Method	Lang. Reph.	Object Swap	Pos. Swap	Lang. Instr.	Bg. Text.	Robot Init	Cam. View	Obj. Layout	Light Cond.	Sensor Noise	Overall
Know. Insulation* [10]	12.2	82.8	0.0	18.2	66.3	28.0	90.2	48.8	72.3	12.0	48.0
MolmoAct [18]	77.8	82.4	0.0	79.5	84.1	47.4	10.1	76.5	77.4	53.4	60.8
VLA-A [Frozen] [32]	56.0	73.4	0.0	41.5	70.9	35.1	94.4	62.3	84.9	36.2	59.9
OpenVLA-OFT [16]	74.4	95.2	0.0	81.5	95.7	40.3	94.7	88.6	95.5	28.2	74.1
VLA-Adapter [32]	74.2	89.6	6.0	85.1	90.7	52.6	92.6	93.2	93.2	89.5	85.1
Anchor VLA	95.4	96.2	12.6	**90.5**	96.9	56.0	95.2	95.8	**100.0**	95.7	89.7
Anchor-Align VLA	**96.6**	**96.2**	**24.4**	87.2	**99.6**	**59.1**	**96.3**	**97.4**	99.0	**96.9**	**90.3**
Δ Accuracy	+22.4	+6.6	+18.4	+2.1	+8.9	+6.5	+3.7	+4.2	+5.8	+7.4	+5.2

Table 1: Robustness and generalization test. Anchor-Align VLA achieves state-of-the-art performance on LIBERO-PRO and LIBERO-Plus benchmarks (see Sec. 4.1), with significant improvements over baselines across both semantic and perceptual perturbations. Best result is **bolded**. The Δ Accuracy row reports absolute improvement over VLA-Adapter, discussed in Sec. 4.2. *Our implementation of the knowledge insulation method on VLA-Adapter, adapted to test the efficacy of knowledge insulation in the finetuning regime.

ing destroys, and that explicit language–action alignment further strengthens the model's ability to ground language instructions in correct motor behavior.

Anchor-and-Align preserves spatial understanding and improves visual grounding of the backbone VLM. LIBERO-PRO position swap (Swap) is the most challenging perturbation axis: the spatial arrangement of task-relevant objects is permuted at test time, so a policy that has memorized fixed scene-to-action mappings during training would fail. Prior VLAs, *i.e.*, Molmoact, OpenVLA-OFT, Frozen VLA-adapter, score 0% under Pro-Swap. Anchor-Align VLA achieves 24.4%, quadrupling VLA-Adapter's performance. We attribute this to two complementary mechanisms: (i) layer-wise anchoring preserves the pretrained VLM's spatial and relational representations, which encode object identity and relative position, and (ii) the alignment objective forces the model to ground its language predictions in the current observation, coupling the instruction semantics to the actual spatial layout rather than to a memorized trajectory. Fig. 3 illustrates this concretely: when the plate and cookie box are swapped relative to training, Action-Only Finetuning executes the memorized motor plan and fails, whereas Anchor-Align VLA executes a visually grounded trajectory that correctly localizes the target bowl in the rearranged scene and completes the pick-and-place.

Anchor-and-Align achieves SOTA in-distribution performance. Beyond out-of-distribution robustness, Anchor-Align VLA sets a new state of the art on standard LIBERO, the in-distribution generalization benchmark where test initial states are drawn from the same distribution as training. As shown in Tab. 2, Anchor-Align VLA achieves the highest success rate on most suites, 98.4% (Spatial), 100.0% (Object), 97.2% (Goal), surpassing prior methods including $\pi_{0.5}$-KI and OpenVLA-OFT, which use substantially larger backbones and large-scale robotic pre-training. This demonstrates that anchoring and alignment do not trade in-distribution competence for robustness; rather, by preserving and lever-

Method	Spatial	Object	Goal	Long
Diffusion Policy [7]	78.3	92.5	68.3	50.5
π_0-FAST [2, 25]	87.0	63.0	89.0	48.0
SmolVLA-0.24B [29]	87.0	93.0	88.0	63.0
SmolVLA-2.25B [29]	93.0	94.0	91.0	77.0
OpenVLA-OFT [16]	94.3	95.2	91.7	86.5
MolmoAct [18]	87.0	95.4	87.6	77.2
$\pi_{0.5}$-KI [10]	96.6	97.2	94.6	85.8
VLA-0 [14]	93.6	96.0	95.6	87.6
VLA-Adapter[Frozen] [32]	89.4	89.6	88.0	84.5
VLA-Adapter [32]	96.0	99.8	96.0	89.0
Anchor VLA	97.0	98.4	96.4	**91.4**
Anchor-Align VLA	**98.4**	**100.0**	**97.2**	90.8

Table 2: Success rates across LIBERO suites. Anchoring and Aligning achieve state-of-the-art results on unperturbed (standard) benchmarks. Best per column is **bolded**.

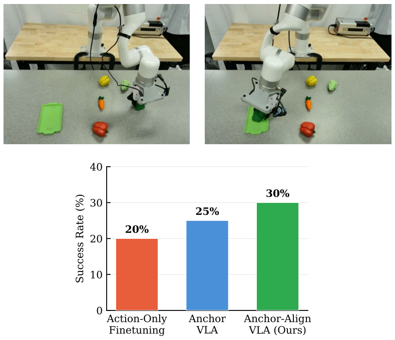


Fig. 4: Real-world pick-and-place. Top: pick (left) and place (right) phases. Bottom: Success rate on real-world trials.

aging the VLM's pretrained representations, Anchor-and-Align improves both simultaneously.

4.3 Real-World Results

To validate that Anchor-and-Align transfers beyond simulation, we conduct real robot experiments on a pick-and-place task requiring object discrimination and spatial reasoning under clutter. We use a UFactory XArm 7 equipped with a wrist-mounted RGB camera, and another camera is placed in front of the robot. The workspace contains five toy vegetables arranged in randomized configurations across episodes. The robot must identify and grasp the broccoli, which varies in position each episode, and place it on a fixed green tray. This task is purposefully designed to probe the same capabilities stressed by LIBERO-PRO: the policy cannot rely on memorized spatial locations to the current visual scene. We collect 40 demonstration episodes via teleoperation. Each model is trained for 15K steps, and the best-performing checkpoint is evaluated over 20 trials.

As shown in Fig. 4, Anchor-Align VLA achieves 30% success, outperforming both the action-only finetuning (20%) and Anchor-only VLA (25%). While absolute success rates are modest in comparison to Libero, this is consistent with the difficulty observed on the LIBERO-PRO position-swap benchmark. The relative ordering of methods, action-only < anchor-only < Anchor-Align VLA, mirrors the simulation trends, supporting the conclusion that language-action alignment finetuning provides a consistent benefit that transfers to physical robots. For real-world rollout examples, refer to App. **??**.

4.4 Diagnostic Evaluation of Misalignment in Co-trained VLAs

Diagnosing language–action misalignment. We extend `ACL`, here as a diagnostic benchmark that tests whether a VLA's language representations are internally consistent with its action outputs. Rather than evaluating language and action accuracy in isolation, `ACL` measures their joint agreement at each timestep, exposing cases where the two channels describe different next-step

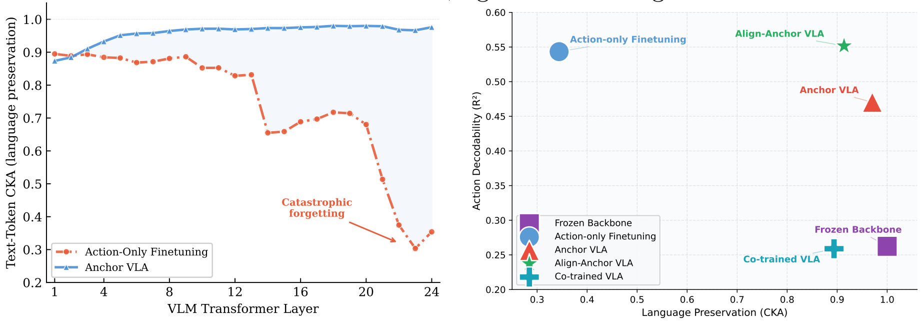
	Task Completion ↑			Grasp ↑			Orientation ↑			Direction ↑		
Model	Align.	Lang.	Action	Align.	Lang.	Action	Align.	Lang.	Action	Align.	Lang.	Action
ChatVLA	48.1	65.6	44.3	47.7	50.4	66.0	20.7	12.9	14.1	20.9	20.3	20.4
MolmoAct	24.6	44.6	65.2	37.8	51.8	17.1	7.4	15.0	28.1	15.5	22.6	48.3
Magma	39.0	56.6	65.8	10.8	49.4	49.5	12.6	13.6	13.9	17.2	27.3	28.0

Table 3: The language-action misalignment is pervasive across VLAs. Despite including co-training on internet datasets eg. GQA (in contrast to `ACL`) prior VLAs fall short on *functional* understanding when questioned "is the task completed?" (task completion), "in which direction should the end effector move?" (direction), etc. Reported numbers denote accuracy %.

behaviors. We partition each trajectory into four semantic phases based on the functional understanding required for robotic manipulations (details in App. **??**): *task completion* (near-terminal states), *direction* (early translational motion), *grasp* (contact events), and *orientation* (mid-trajectory reorientation), and define three metrics: **Language accuracy**, the fraction of timesteps where the language head predicts the correct action-derived label; **Action accuracy**, the fraction where the discretized policy action matches the ground-truth label; and **Alignment accuracy**, the fraction where the language and action predictions agree with each other, regardless of correctness. Tab. 3 evaluates three VLAs (Molmoact [18], ChatVLA [39], and Magma [35]) on `ACL`. Alignment is consistently weaker than either unimodal accuracy in isolation: even when language and action heads each achieve moderate scores on a given axis, their joint agreement drops substantially, indicating that the two channels frequently describe different next-step behaviors for the same observation. The misalignment is most severe on fine-grained control axes such as Direction and Grasp, where all three models exhibit low alignment despite non-trivial unimodal performance. These results confirm that current VLAs do not internally synchronize their language and action representations, motivating the language-action alignment training objective introduced in Sec. 3. Details of benchmark curation in App. **??** and additional analysis in App. **??**.

4.5 Ablations and Analysis

Language Preservation across layers. To understand how finetuning reshapes the internal representations of the pretrained VLM backbone, we perform a language preservation analysis. In Language Preservation (Fig 5a), we measure how much of the pretrained language model's representational geometry is retained after action finetuning using the Centered Kernel Alignment (CKA) metric [3]. CKA quantifies the structural similarity between two sets of representations. Given hidden state matrices $X \in \mathbb{R}^{n \times d}$ from the frozen VLM and $Y \in \mathbb{R}^{n \times d}$ from the VLM Backbone at the same layer, linear CKA is computed as: $\mathrm{CKA}(X, Y) = \frac{|\tilde{X}^\top \tilde{Y}|_F^2}{|\tilde{X}^\top \tilde{X}|_F \cdot |\tilde{Y}^\top \tilde{Y}|_F}$ where $\tilde{X}$ and $\tilde{Y}$ are column-centered. A CKA of 1.0 indicates identical representational structure; lower values indicate that finetuning has reshaped the layer's information. We report text token CKA because it directly reflects the degree to which the model's language understanding



(a) Language representation preservation across layers. The Action-Only Finetuning model destroys language representations in later layers (CKA drops below 0.35). Anchor VLA maintains near-perfect preservation via MSE distillation (CKA > 0.95).

(b) Language preservation vs. action decodability. Anchor-Align VLA retains the VLM's language geometry (high CKA) while producing the most action-decodable representations (linear-probe R^2 higher is better).

Fig. 5: Representation analysis. Anchoring preserves pretrained language (left), and Anchor-Align VLA achieves the best trade-off between language retention and action decodability (right).

is preserved. As shown in Fig 5a, standard finetuning catastrophically destroys text representations in the output layers (CKA drops to 0.34 at layer 24). Anchor VLA recovers near-perfect language preservation (CKA > 0.95 across all layers) through all-layer MSE distillation, confirming that the anchoring with frozen VLM directly prevents representational collapse.

VLM knowledge retention: VQA test. Adding technical rigor to our Fig. 1a. We use the GQA dataset [15], which tests compositional visual reasoning, including spatial relations, object attributes, counting, and multi-hop comparisons. In this test, we observe catastrophic forgetting without distillation during finetuning. The Action-Only Finetuning loses 63% of its GQA accuracy within the first 2,500 finetuning steps (36.4% → 13.5%) and 94% by 10,000 steps (36.4% → 2.3%). This monotonic decline confirms that unconstrained action finetuning rapidly erases the pretrained VLM's visual reasoning capabilities, even when only LoRA parameters are updated. All-layer MSE distillation preserves VLM knowledge. Anchor VLA retains 97% of the VLM's GQA accuracy at 10,000 steps (35.3% vs. 36.4%), while simultaneously achieving 98.4% task success rate on LIBERO Spatial, demonstrating that robot action performance and visual-linguistic reasoning are not fundamentally at odds. This behavioral observation is corroborated by the layer-wise CKA analysis in Fig. 5a: the Action-Only Finetuning's text-token representations diverge sharply from the pretrained VLM in the later transformer layers (CKA drops to 0.34 at layer 24), directly explaining the catastrophic VQA degradation, whereas Anchor VLA maintains CKA > 0.95 across all layers, consistent with its near-complete retention of GQA accuracy.

Action Information across layers (Fig. 5b). We quantify how much action-relevant information is linearly accessible from the model's hidden states at each layer using linear probing. For every layer, we extract the hidden state and mean-pool over vision and text token positions to obtain a single feature vector per sample. We then fit a ridge regression model to predict the ground-truth discretized action (7-dimensional, 256 bins per dimension) from these pooled features. A higher R^2 (R is the coefficient of determination) indicates that more action information is linearly decodable from that layer's representations, even without passing through the action head. This metric measures how effectively the VLM backbone serves as a prior for action generation. Anchor-Align VLA achieves the best of both objectives, Fig. 5b: it maintains strong language preservation in the output layers (CKA $=$ 0.91) while attaining the highest action decodability of any method (peak $\mathbf{R^2} = 0.60$ at layer 22), demonstrating that the alignment loss routes action-relevant information through the network's later layers without overwriting the pretrained language geometry. The two metrics are complementary: language preservation captures what the model retains from pretraining, while action decodability captures what the model gains from finetuning. Together, they reveal whether a given training strategy achieves high task performance by destroying pretrained representations (as in standard finetuning) or by constructively enriching them (as in our approach).

Anchoring is critical for robustness. To isolate the contribution of representation anchoring, we train an alignment-only variant that uses the same direction-word alignment objective but removes the layer-wise MSE distillation loss (λ_{MSE}=0). As shown in Fig. 6, removing anchoring causes consistent degradation across all three evaluation axes: standard success drops by 3.2%, PRO language rephrase falls by 6.2%, and PRO object swap drops by 1.2%. Without anchoring, the backbone's pretrained representations drift during finetuning, and the alignment objective alone cannot compensate. This confirms that anchoring and alignment play complementary roles; anchoring preserves the representational erosion that alignment leverages.

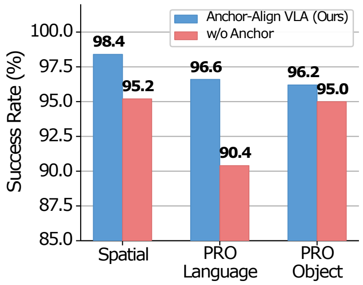


Fig. 6: Effect of removing anchoring while keeping alignment. Removing the MSE distillation loss causes consistent degradation across all evaluation axes, confirming that anchoring and alignment play complementary roles.

5 Conclusion

In this work, we showed that finetuning pretrained VLMs into VLAs introduces two failure modes: erosion of pretrained vision-language representations and inconsistency between language and action predictions on the same observations. We addressed these with Anchor-and-Align, which uses Vision-Language Anchoring to preserve the pretrained VLM representation space and Language-Action Alignment using `ACL`-derived supervision from the same demonstrations used for control learning. Across LIBERO, LIBERO-Pro, and LIBERO-Plus, Anchor-and-Align improves both in-distribution task success and out-of-distribution robustness. The same gains transfer to a real xArm7 setup. Overall, we show that effective VLA finetuning should preserve pretrained multimodal representations and should explicitly synchronize language with action, rather than optimizing for control alone.

References

1. Bjorck, J., Castañeda, F., Cherniadev, N., Da, X., Ding, R., Fan, L., et al.: Gr00t n1: An open foundation model for generalist humanoid robots. arXiv preprint arXiv:2503.14734 (2025) 3, 5
2. Black, K., Brown, N., Driess, D., Esmail, A., Equi, M., Finn, C., et al.: π_0: A vision-language-action flow model for general robot control. arXiv preprint arXiv:2410.24164 (2024) 3, 7, 11
3. Bo, Y., Soni, A., Srivastava, S., Khosla, M.: Evaluating representational similarity measures from the lens of functional correspondence. arXiv preprint arXiv:2411.14633 (2024) 12
4. Brohan, A., Brown, N., Carbajal, J., Chebotar, Y., Dabis, J., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Hsu, J., et al.: Rt-1: Robotics transformer for real-world control at scale. arXiv preprint arXiv:2212.06817 (2022) 3
5. Bu, Q., Yang, Y., Cai, J., Gao, S., Ren, G., Yao, M., Luo, P., Li, H.: Univla: Learning to act anywhere with task-centric latent actions. arXiv preprint arXiv:2505.06111 (2025) 5
6. Chen, K., Xie, S., Ma, Z., Sanketi, P.R., Goldberg, K.: Robo2vlm: Visual question answering from large-scale in-the-wild robot manipulation datasets. arXiv preprint arXiv:2505.15517 (2025) 4
7. Chi, C., Xu, Z., Feng, S., Cousineau, E., Du, Y., Burchfiel, B., Tedrake, R., Song, S.: Diffusion policy: Visuomotor policy learning via action diffusion. Int. J. Robot. Res. (2023) 7, 11
8. Das, A., Datta, S., Gkioxari, G., Lee, S., Parikh, D., Batra, D.: Embodied question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1–10 (2018) 4
9. Doshi, R., Walke, H., Mees, O., Dasari, S., Levine, S.: Scaling cross-embodied learning: One policy for manipulation, navigation, locomotion and aviation. arXiv preprint arXiv:2408.11812 (2024) 3
10. Driess, D., Springenberg, J.T., Ichter, B., Yu, L., Li-Bell, A., Pertsch, K., Ren, A.Z., Walke, H., Vuong, Q., Shi, L.X., et al.: Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. arXiv preprint arXiv:2505.23705 (2025) 3, 8, 10, 11
11. Driess, D., Xia, F., Sajjadi, M.S., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., et al.: Palm-e: An embodied multimodal language model. arXiv preprint arXiv:2303.03378 (2023) 3
12. Fei, S., Wang, S., Shi, J., Dai, Z., Cai, J., Qian, P., Ji, L., He, X., Zhang, S., Fei, Z., Fu, J., Gong, J., Qiu, X.: LIBERO-Plus: In-depth robustness analysis of vision-language-action models. arXiv preprint arXiv:2510.13626 (2025) 8
13. Gao, Q., Pi, X., Liu, K., Chen, J., Yang, R., Huang, X., Fang, X., Sun, L., Kishore, G., Ai, B., et al.: Do vision-language models have internal world models? towards an atomic evaluation. arXiv preprint arXiv:2506.21876 (2025) 4
14. Goyal, A., Hadfield, H., Yang, X., Blukis, V., Ramos, F.: VLA-0: Building state-of-the-art VLAs with zero modification. arXiv preprint arXiv:2510.13054 (2025) 8, 11
15. Hudson, D.A., Manning, C.D.: Gqa: A new dataset for real-world visual reasoning and compositional question answering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6700–6709 (2019) 8, 13
16. Kim, M., Finn, C., Liang, P.: Fine-tuning vision-language-action models: Optimizing speed and success. arXiv preprint arXiv:2502.19645 (2025) 3, 8, 10, 11

17. Kim, M., Pertsch, K., Karamcheti, S., Xiao, T., Balakrishna, A., Nair, S., Rafailov, R., Foster, E., Lam, G., Sanketi, P., Vuong, Q., Kollar, T., Burchfiel, B., Tedrake, R., Sadigh, D., Levine, S., Liang, P., Finn, C.: Openvla: An open-source vision-language-action model. arXiv preprint arXiv:2406.09246 (2024) 1, 3, 5
18. Lee, J., Duan, J., Fang, H., Deng, Y., Liu, S., Li, B., Fang, B., Zhang, J., Wang, Y.R., Lee, S., et al.: Molmoact: Action reasoning models that can reason in space. arXiv preprint arXiv:2508.07917 (2025) 4, 8, 10, 11, 12
19. 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) 3
20. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014) 8
21. Liu, B., Zhu, Y., Gao, C., Feng, Y., Liu, Q., Zhu, Y., Stone, P.: LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In: NeurIPS (2023) 8
22. Liu, F., Emerson, G., Collier, N.: Visual spatial reasoning. Transactions of the Association for Computational Linguistics **11**, 635–651 (2023). https://doi.org/10.1162/tacl_a_00566, https://aclanthology.org/2023.tacl-1.37/ 8
23. Majumdar, A., Ajay, A., Zhang, X., Putta, P., Yenamandra, S., Henaff, M., Silwal, S., Mcvay, P., Maksymets, O., Arnaud, S., et al.: Openeqa: Embodied question answering in the era of foundation models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16488–16498 (2024) 4
24. O'Neill, A., Rehman, A., Maddukuri, A., Gupta, A., Padalkar, A., Lee, A., Pooley, A., Gupta, A., Mandlekar, A., Jain, A., et al.: Open x-embodiment: Robotic learning datasets and rt-x models: Open x-embodiment collaboration 0. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). pp. 6892–6903. IEEE (2024) 3
25. Pertsch, K., Stachowicz, K., Ichter, B., Driess, D., Nair, S., Vuong, Q., Mees, O., Finn, C., Levine, S.: FAST: Efficient action tokenization for vision-language-action models. arXiv preprint arXiv:2501.09747 (2025) 7, 11
26. Qu, D., Song, H., Chen, Q., Chen, Z., Gao, X., Ye, X., Lv, Q., Shi, M., Ren, G., Ruan, C., et al.: Eo-1: Interleaved vision-text-action pretraining for general robot control. arXiv preprint arXiv:2508.21112 (2025) 4
27. Qu, D., Song, H., Chen, Q., Yao, Y., Ye, X., Ding, Y., Wang, Z., Gu, J., Zhao, B., Wang, D., et al.: Spatialvla: Exploring spatial representations for visual-language-action model. arXiv preprint arXiv:2501.15830 (2025) 3
28. Reed, S., Zolna, K., Parisotto, E., Colmenarejo, S.G., Novikov, A., Barth-Maron, G., Gimenez, M., Sulsky, Y., Kay, J., Springenberg, J.T., et al.: A generalist agent. arXiv preprint arXiv:2205.06175 (2022) 3
29. Shukor, M., Aubakirova, D., Capuano, F., Kooijmans, P., Palma, S., Zouitine, A., Aractingi, M., Pascal, C., Russi, M., Marafioti, A.: SmolVLA: A vision-language-action model for affordable and efficient robotics. arXiv preprint arXiv:2506.01844 (2025) 8, 11
30. Singh, A., Natarajan, V., Shah, M., Jiang, Y., Chen, X., Batra, D., Parikh, D., Rohrbach, M.: Towards vqa models that can read. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8317–8326 (2019) 4
31. Team, O.M., Ghosh, D., Walke, H., Pertsch, K., Black, K., Mees, O., Dasari, S., Hejna, J., Kreiman, T., Xu, C., et al.: Octo: An open-source generalist robot policy. arXiv preprint arXiv:2405.12213 (2024) 3

32. Wang, Y., Ding, P., Li, L., Cui, C., Ge, Z., Tong, X., Song, W., Zhao, H., Zhao, W., Hou, P., et al.: Vla-adapter: An effective paradigm for tiny-scale vision-language-action model. arXiv preprint arXiv:2509.09372 (2025) 1, 3, 5, 7, 8, 9, 10, 11
33. Wen, J., Zhu, Y., Li, J., Tang, Z., Shen, C., Feng, F.: Dexvla: Vision-language model with plug-in diffusion expert for general robot control. arXiv preprint arXiv:2502.05855 (2025) 3, 4
34. Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R., Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., Qiu, Z., et al.: Qwen2.5 technical report. arXiv preprint arXiv:2412.15115 (2024) 7, 8
35. Yang, J., Tan, R., Wu, Q., Zheng, R., Peng, B., Liang, Y., Gu, Y., Cai, M., Ye, S., Jang, J., et al.: Magma: A foundation model for multimodal ai agents. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 14203–14214 (2025) 4, 12
36. Zawalski, M., Chen, W., Pertsch, K., Mees, O., Finn, C., Levine, S.: Robotic control via embodied chain-of-thought reasoning. arXiv preprint arXiv:2407.08693 (2024) 4
37. Zhou, X., Xu, Y., Tie, G., Chen, Y., Zhang, G., Chu, D., Zhou, P., Sun, L.: LIBERO-PRO: Towards robust and fair evaluation of vision-language-action models beyond memorization. arXiv preprint arXiv:2510.03827 (2025) 8
38. Zhou, Z., Zhu, Y., Wen, J., Shen, C., Xu, Y.: Vision-language-action model with open-world embodied reasoning from pretrained knowledge. arXiv preprint arXiv:2505.21906 (2025) 4
39. Zhou, Z., Zhu, Y., Zhu, M., Wen, J., Liu, N., Xu, Z., Meng, W., Cheng, R., Peng, Y., Shen, C., et al.: Chatvla: Unified multimodal understanding and robot control with vision-language-action model. arXiv preprint arXiv:2502.14420 (2025) 4, 5, 12
40. Zitkovich, B., Yu, T., Xu, S., Xu, P., Xiao, T., Xia, F., Wu, J., Wohlhart, P., et al.: Rt-2: Vision-language-action models transfer web knowledge to robotic control. In: Proceedings of the 7th Conference on Robot Learning (CoRL). Proceedings of Machine Learning Research, vol. 229, pp. 2165–2183. PMLR (2023) 1, 3, 4