FROM SPARSE TO DENSE: DIAGNOSING TASK-EQUIVALENT LANGUAGE ROBUSTNESS IN VISION-LANGUAGE-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language-Action (VLA) models are commonly trained and evaluated with concise task commands, leaving open whether learned control remains accessible when the same executable intent is expressed through richer, task-equivalent language. An audit of LIBERO and SimplerEnv finds that their native instructions concentrate on direct, low-context expressions. To study this uncovered regime, we construct a three-level task-equivalent benchmark with controlled dense language and evaluate models under matched rollouts that vary only the language realization. These evaluations reveal a substantial Sparse–Dense Gap whose magnitude depends strongly on the model and task. Further diagnostics show that paired sparse–dense action training yields limited dense gains and that action fine-tuning reduces inherited visual-language capability. Guided by these findings, **STAR**, a **S**parse-to-dense **T**ask **A**lignment **R**egularization framework, uses a lineage-matched pre-action VLM and the source VLA as complementary representation and action anchors. Across LIBERO, STAR raises dense macro success from 7.2% to 24.3% while retaining 69.7% sparse success, and it improves the corresponding sparse–dense balance in SimplerEnv, demonstrating that task-equivalent dense-language access is a distinct robustness problem that can be systematically measured and improved.

1 INTRODUCTION

Vision-Language-Action (VLA) models map visual observations and language instructions to robot actions, bringing pretrained visual-language knowledge into robotic control Zitkovich et al. (2023); Kim et al. (2024); Pan et al. (2026). Advances in policy scale, action modeling, and cross-embodiment learning have substantially broadened their control capabilities Mees et al. (2024); Black et al. (2026). Existing VLA robustness studies evaluate paraphrases, linguistic perturbations, object references, irrelevant context, and broader changes in evaluation conditions Pugacheva et al. (2025); Kim et al. (2026); Hou & Zhao (2026); Fei et al. (2025); Zhou et al. (2026). A central question remains at the language–control interface. Once a VLA has learned a task, can it still access that control capability when the same executable intent is expressed through another valid language form?

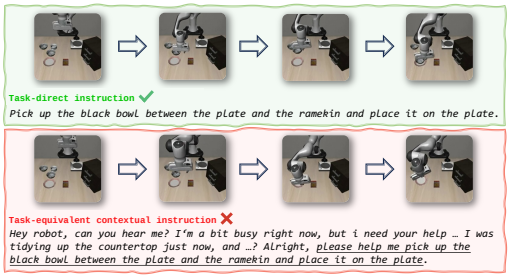

Figure 1: Motivating case study of task-equivalent language robustness. In a matched rollout, OpenVLA completes the task when it is stated directly but fails when the same executable intent is embedded in richer conversational context.

In the native interfaces used by LIBERO and SimplerEnv, tasks are presented through concise, task-direct commands Liu et al. (2023); Li et al. (2025). The same executable intent can also be conveyed through additional context, cross-sentence references, or indirect language. These forms are absent from the native instruction sets. We therefore probe this broader interface with the matched case

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

study in Figure 1: the source command is replaced by a task-equivalent, context-rich realization while the task and rollout conditions remain fixed. OpenVLA succeeds on the native command but fails on the richer realization. This contrast motivates a central requirement: learned control should remain accessible across task-equivalent language forms.

To study this failure systematically, we distinguish the native task-direct form from task-equivalent realizations that preserve the executable intent while requiring greater contextual interpretation. We refer to these forms as **sparse** and **dense**, respectively, and define their matched success difference as the **Sparse–Dense Gap**. The resulting diagnostics expose two linked obstacles. Direct mixed action training improves access through dense language but does not consistently retain the established sparse route, while action fine-tuning in the OpenVLA lineage weakens inherited visual-language capability. An effective adaptation must broaden the language interface while retaining both established control and the representation learned before action specialization. **STAR**, or Sparse-to-dense Task Alignment Regularization, addresses this objective through paired sparse–dense learning, representation alignment with the pre-action VLM, and action alignment with the source VLA.

To evaluate this robustness at controlled levels of difficulty, we construct a three-level benchmark of progressively richer task-equivalent instructions. Experiments on LIBERO and SimplerEnv show that STAR improves access through dense language while largely retaining established sparse behavior. Component and data ablations connect this behavior to the two alignment targets and to the composition of dense training data.

Our contributions are summarized as follows:

- We formulate task-equivalent language robustness, audit the native LIBERO and SimplerEnv instruction interfaces, and measure the resulting failure through the Sparse–Dense Gap.
- Matched robot evaluations, general VQA, and direct mixed training identify the capabilities that adaptation must preserve.
- We propose STAR, using a pre-action VLM for representation supervision and a source VLA for action-distribution supervision.
- We introduce a three-level task-equivalent benchmark and evaluate STAR across LIBERO and SimplerEnv with cross-model comparisons and ablations.

2 Diagnosing Task-Equivalent Language Robustness

This section uses OpenVLA Kim et al. (2024) as a representative autoregressive VLA for controlled diagnosis. Its matched pre-action Prismatic VLM and action-specialized policy allow us to isolate the Sparse–Dense Gap, trace inherited visual-language capability across action fine-tuning, and test whether paired action supervision balances dense access with sparse retention. We first define task equivalence and instruction density and audit the native evaluation interfaces. These diagnostics establish the requirements for the method developed in the next section.

2.1 Task-Equivalent Sparse and Dense Instructions

For an executable task τ, two instructions are task-equivalent when they preserve its target objects, manipulation or spatial relations, action order, goal state, and success condition. Their wording, discourse structure, and required interpretation may differ. Equivalence does not require identical action trajectories. The *Semantic Equivalence Score* (SES) rates preservation of these executable task constraints on a 1–5 scale, with higher scores indicating stronger preservation.

The *Interaction Density Score* (IDS) characterizes the contextual interpretation required to recover the task from an instruction on a 1–5 scale. Scores 1 and 2 describe direct or low-context expressions. Scores 3, 4, and 5 respectively require contextual filtering, cross-sentence reference resolution, and semantic inference. SES measures whether task identity is preserved, whereas IDS measures how much interpretation the language interface requires.

Source	N	Avg. IDS	IDS distribution (%)	
			1–2	3–5
LIBERO	40	1.10	100.0	0.0
SimplerEnv	68	1.00	100.0	0.0
All	108	1.04	100.0	0.0

Table 1: IDS statistics for native instruction instances.

Applying this rubric to all 108 native instruction instances from LIBERO and SimplerEnv places every instance at IDS 1–2 (Tab. 1); none reaches IDS 3–5. The native evaluation interfaces therefore cover direct and low-context task expressions, leaving richer task-equivalent forms outside their original coverage.

Within a task-equivalent pair (s, d), we call the task-direct source form s *sparse*. We call d *dense* when it preserves the same executable task but requires IDS 3–5 interpretation. The distinction concerns the path from language to the task, rather than utterance length or a change in task complexity. Section 4 instantiates the dense side with Simple, Middle, and Hard instructions at IDS 3, 4, and 5, using SES to verify task preservation.

Policy	Sparse	Simple	Gap (pp)
OpenVLA	83.2	17.6	−65.6

Table 2: Sparse–Dense Gap on LIBERO-Spatial (%). Simple uses IDS = 3; Gap is Simple minus Sparse.

Benchmark	Prismatic VLM	OpenVLA
VQAv2	79.1	31.7
GQA	64.0	27.5

Table 3: VQA accuracy (%) before and after OpenVLA action fine-tuning.

Method	Sparse	Dense
OpenVLA	83.2	17.6
Direct Mix	79.8	27.0

Table 4: Direct Mix on LIBERO-Spatial (%). Dense uses IDS = 3.

2.2 Establishing the Sparse–Dense Gap

Table 2 tests whether task-equivalent language alone changes execution under a matched rollout protocol. The OpenVLA policy fine-tuned on LIBERO-Spatial is evaluated on all ten tasks, with each sparse source command paired to a task-equivalent Simple instruction at IDS = 3. The executable task, scene, initial state, success detector, and rollout protocol remain fixed, and only the language realization changes. Let $R_\pi(I)$ denote the success rate of policy π under instruction form I. We report the signed Sparse–Dense Gap as $R_\pi(d) - R_\pi(s)$, so a negative value denotes reduced access through the denser realization.

Across 500 evaluation episodes per instruction form, success decreases from 83.2% to 17.6%. This corresponds to a signed gap of -65.6 percentage points, or a 65.6-point drop. Because the executable constraints and rollout conditions are matched, this difference isolates sensitivity to the language realization. The policy can execute the task through its task-direct source command. The same learned capability becomes substantially less accessible through a task-equivalent instruction that requires contextual filtering.

2.3 Tracking Visual-Language Capability

OpenVLA is obtained by action fine-tuning a Prismatic VLM Karamcheti et al. (2024); Kim et al. (2024). The matched Prismatic checkpoint serves as the pre-action reference in Tab. 3. On VQAv2 and GQA Goyal et al. (2017); Hudson & Manning (2019), accuracy declines from 79.1% and 64.0% before action fine-tuning to 31.7% and 27.5% afterward. These declines show that action specialization reduces access to visual-language capability inherited from the base VLM. The experiments track this capability alongside robot-task success.

2.4 Testing Paired Action Supervision

Having established the access gap, Tab. 4 tests whether paired sparse–dense action supervision alone provides a balanced solution. Direct Mix continues action training on sparse and dense instructions that share the same observations and target actions. At IDS = 3, dense success increases by 9.4 points, from 17.6% to 27.0%, while sparse success decreases by 3.4 points, from 83.2% to 79.8%. Paired action supervision improves access through the Simple realization, but a 52.8-point gap remains between the two task-equivalent forms.

Together, the diagnostics establish three objectives for robustness adaptation: extend access to an established task through higher-density instructions, retain the visual-language capability inherited before action specialization, and preserve the control capability already available through the source language interface. The next section develops a method around these objectives.

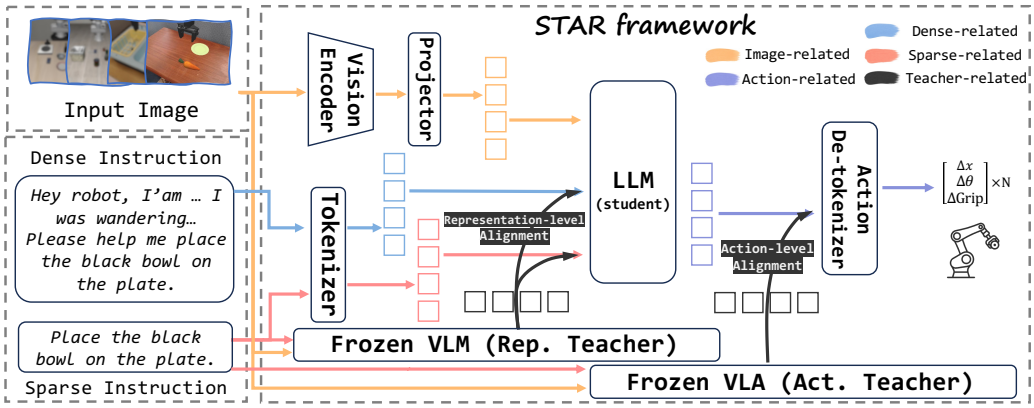

Figure 2: STAR aligns a task-equivalent sparse–dense pair to same-form representations from the frozen pre-action VLM and a shared sparse-conditioned action distribution from the frozen source VLA. Inference uses only the adapted policy.

3 METHOD

To meet these objectives, we propose **STAR**, or **S**parse-to-dense **T**ask **A**lignment **R**egularization. STAR combines paired sparse–dense action learning with complementary targets for the inherited representation space and source action distribution. The lineage-matched pre-action VLM provides representation targets for both forms, while the source VLA provides the action-distribution target of the established sparse controller (Fig. 2).

Let g_{ϕ_0} be the pre-action VLM, f_{θ_0} the sparse-capable source VLA, and f_θ the student initialized from f_{θ_0}. Each example pairs sparse $I_{s,b}$ and dense $I_{d,b}$ instructions with observation x_b and target actions $\mathbf{a}_b$. Paired action loss trains both routes on the demonstration. Representation alignment matches each route to its same-form VLM target. Action alignment matches both routes to the source VLA's sparse-conditioned distribution.

3.1 PAIRED SPARSE–DENSE ACTION LEARNING

Paired supervision trains the student to produce the demonstrated action trajectory from either instruction form. For a batch of B examples, where K_b is the length of target sequence $\mathbf{a}_b$, we use

$$\mathcal{L}_{\text{act}} = -\frac{1}{B}\sum_{b=1}^{B}\frac{1}{K_b}\sum_{k=1}^{K_b}\sum_{q\in\{s,d\}}\log p_\theta(a_{b,k} \mid x_b, I_{q,b}, a_{b,<k})\,. \tag{1}$$

The dense term learns the contextual route, while the sparse term continues supervision on the source route. Since action labels specify neither the pretrained representation space nor the source policy's full distribution, STAR adds a teacher for each target.

3.2 REPRESENTATION ALIGNMENT WITH THE PRE-ACTION VLM

The frozen representation teacher g_{ϕ_0} precedes action fine-tuning in the source lineage. For each $q \in \{s, d\}$, teacher and student process $(x_b, I_{q,b})$. Let $h_{\phi_0}(x, I)$ and $h_\theta(x, I)$ be their mean-pooled layer-16 instruction representations. STAR aligns each route to its same-form target:

$$\mathcal{L}_{\text{rep}} = \frac{1}{2B}\sum_{b=1}^{B}\sum_{q\in\{s,d\}}\left[1-\cos(h_\theta(x_b, I_{q,b}), h_{\phi_0}(x_b, I_{q,b}))\right]. \tag{2}$$

This penalizes deviations from how the pretrained VLM represents each form without forcing sparse and dense hidden states to be identical. Paired action learning maps both forms to the demonstrated actions.

3.3 Action Alignment with the Source VLA

The frozen source VLA f_{θ_0} is the action-distribution teacher. We distill Hinton et al. (2015) its sparse-conditioned distribution into both student routes. Let $p_\theta^{(T)}$ and $p_{\theta_0}^{(T)}$ denote their distributions at temperature T. We compare them at the same prediction position under the shared target prefix:

$$\mathcal{L}_{\mathrm{KD}} = \frac{T^2}{2B} \sum_{b=1}^{B} \frac{1}{K_b} \sum_{k=1}^{K_b} \sum_{q \in \{s,d\}} \mathrm{KL}\Big(p_{\theta_0}^{(T)}(\cdot \mid x_b, I_{s,b}, a_{b,<k}) \,\|\, p_\theta^{(T)}(\cdot \mid x_b, I_{q,b}, a_{b,<k}) \Big). \tag{3}$$

The shared target anchors both routes toward the source control distribution and supplies token-level information absent from a single demonstration.

3.4 Overall Objective and Inference

The full objective combines the three terms:

$$\mathcal{L}_{\mathrm{STAR}} = \mathcal{L}_{\mathrm{act}} + \lambda_{\mathrm{rep}} \mathcal{L}_{\mathrm{rep}} + \lambda_{\mathrm{KD}} \mathcal{L}_{\mathrm{KD}}. \tag{4}$$

Inference uses only the adapted policy.

4 Dense Instruction Benchmark and Evaluation Protocol

From each sparse source we construct task-equivalent **Simple**, **Middle**, and **Hard** instructions targeting IDS 3, 4, and 5. Simple embeds a direct command in conversational context. Middle requires cross-sentence reference resolution. Hard requires the action and relevant objects to be inferred from indirect language (Fig. 3).

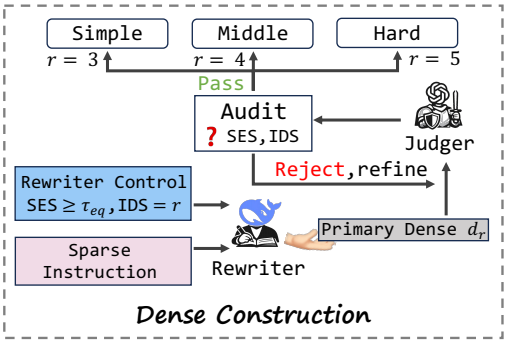

Figure 3: Two-stage dense-instruction construction and rubric-based review.

DeepSeek-V4-Flash DeepSeek-AI et al. (2026) first rewrites each source under level-specific task-preservation rules. The review follows rubric-based LLM evaluation Zheng et al. (2023). GPT-5.6-Sol acts as a separate judge and refiner. It assigns SES and IDS, checks the executable task constraints, and verifies the target density. Candidates with task-constraint violations or target-density mismatches are refined and re-evaluated. Persistent failures return to the rewriter.

The 324 realizations are balanced across dense levels (Tab. 5). Middle has the largest mean word count. Hard is shorter but requires more indirect task recovery, separating instruction density from verbosity.

(a) By benchmark

Benchmark	N	First-pass (%)
LIBERO	120	79.2
Bridge VM	12	66.7
Fractal VA	132	76.2
Fractal VM	60	66.7

(b) By instruction mode

Mode	Words
Sparse	7.4
Simple	43.4
Middle	67.0
Hard	57.7

Table 5: Statistics for 108 sparse sources and 324 accepted dense realizations. N counts dense realizations. Words is the sample-weighted mean. First-pass is the proportion accepted before refinement, averaged across Simple, Middle, and Hard.

These realizations form the evaluation benchmark. STAR uses a separately generated, nonoverlapping training pool. Every policy receives the same held-out variants.

5 EXPERIMENTS

The experiments answer five questions: how task-equivalent language robustness varies across models and task suites; whether STAR improves dense access beyond Direct Mix while retaining sparse performance; whether the same strategy extends to SimplerEnv; how the alignment targets and dense-data composition affect the result; and how much VQA capability remains after STAR adaptation.

5.1 EXPERIMENTAL SETUP

Benchmarks. We evaluate four 10-task LIBERO suites commonly used for VLA evaluation: Spatial, Object, Goal, and LIBERO-10 (Long) Liu et al. (2023); Kim et al. (2025). SimplerEnv evaluates Bridge and Fractal policies in visually matched simulation. This setting extends the study to different robot embodiments and training distributions Li et al. (2025).

Evaluation protocol. We report success rates using the standard LIBERO setup with 50 evaluation rollouts per task Kim et al. (2025). SimplerEnv Bridge uses 24 trials per task and instruction condition. Fractal results are aggregated over task families, with separate Visual Matching and Variant Aggregation settings Li et al. (2025). Every method–instruction condition in VA uses 1,992 rollouts. Suite and task-family averages are macro averages. Cross-model diagnostics use each released VLA unchanged. Matched adaptation fixes the OpenVLA backbone across OpenVLA, Direct Mix, and STAR.

Training details. For the OpenVLA experiments, its matched pre-action Prismatic checkpoint serves as the representation teacher, and a sparse-capable OpenVLA checkpoint serves as the source VLA Karamcheti et al. (2024); Kim et al. (2024). On LIBERO, OpenVLA is fine-tuned separately on sparse data from each suite. Each resulting checkpoint initializes the student and serves as the action-distribution teacher. On SimplerEnv, we first fine-tune OpenVLA on sparse target-domain instructions. The resulting Sparse-FT checkpoint initializes the student and serves as the action-distribution teacher. The pre-action Prismatic checkpoint remains the representation teacher. We mean-pool the exact instruction-token positions at transformer layer 16 and adapt the student with rank-32 LoRA Hu et al. (2021). Training runs for 30k steps in BF16 with learning rate 10^{-5}, $\lambda_{\text{rep}} = 0.1$, $\lambda_{\text{KD}} = 0.5$, and distillation temperature $T = 1$. Supervision covers seven action tokens followed by EOS, and knowledge distillation is computed at these eight prediction positions over the full vocabulary.

Unchanged released policies diagnose model-level variation, the matched OpenVLA block isolates adaptation, and SimplerEnv tests the same source-relative objective beyond LIBERO.

5.2 LIBERO RESULTS

LIBERO provides two complementary tests of the Sparse–Dense Gap and STAR. The first block of Tab. 6 evaluates task-equivalent sparse–dense sensitivity across OpenVLA-OFT, CogVLA, InstructVLA, and $\pi_{0.5}$ Kim et al. (2025); Yang et al. (2026); Intelligence et al. (2025). The second block fixes the OpenVLA backbone and asks whether STAR improves the adaptation balance beyond Direct Mix. Here S denotes Sparse. The symbols D_{S}, D_{M}, and D_{H} denote Simple, Middle, and Hard. Average is the four-suite macro.

The gap depends on both model and task. The cross-model block asks whether the access failure is specific to OpenVLA. All four policies remain strong on sparse commands, but dense success varies sharply across suites and levels. OpenVLA-OFT drops under dense instructions, InstructVLA weakens from Simple to Middle and Hard, and CogVLA varies by suite. The flow-based $\pi_{0.5}$ exceeds 60% on 11 of the 12 dense suite–level conditions, with Goal-Hard at 12.4%. These results establish model- and task-dependent sensitivity and show that strong dense access is attainable under a different policy design.

STAR improves the matched OpenVLA balance. The matched block asks whether STAR moves beyond paired action training alone. Across all four suites, Direct Mix raises dense macro success

Table 6: LIBERO success rates (%) for sparse (S) and three dense levels (D_{S}, D_{M}, and D_{H}). Gray text marks existing VLA models evaluated without dense adaptation. In the matched block, OpenVLA is fine-tuned separately on each suite. The shaded row is the reference, and bold marks the block best. Average is the four-suite macro. The row beneath STAR reports STAR − OpenVLA in percentage points.

Method	Spatial				Object				Goal				Long				Average			
	S	D_{S}	D_{M}	D_{H}	S	D_{S}	D_{M}	D_{H}	S	D_{S}	D_{M}	D_{H}	S	D_{S}	D_{M}	D_{H}	S	D_{S}	D_{M}	D_{H}
Sparse–Dense Gap across VLA Models																				
OpenVLA-OFT	97.6	17.6	21.6	13.8	98.4	24.8	27.8	30.4	97.9	22.2	5.4	3.4	94.5	0.6	2.4	0.6	97.1	16.3	14.3	12.1
CogVLA	98.6	19.8	2.4	9.2	98.8	73.2	59.0	65.8	96.6	9.0	0.8	1.0	95.4	35.6	2.8	7.4	97.4	34.4	16.3	20.9
InstructVLA	97.3	79.0	22.4	2.8	99.6	84.2	43.8	28.0	96.5	68.0	27.0	4.0	89.8	39.6	29.2	7.0	95.8	67.7	30.6	10.5
$\pi_{0.5}$-LIBERO	98.8	98.2	94.8	84.4	98.2	97.0	95.0	60.6	98.0	92.6	66.2	12.4	92.4	92.0	91.2	78.4	96.9	95.0	86.8	59.0
Matched OpenVLA Adaptation																				
OpenVLA	83.2	17.6	13.6	1.8	84.8	24.4	17.0	1.0	**74.8**	2.2	0.8	0.4	**50.6**	5.4	1.8	0.0	**73.4**	12.4	8.3	0.8
Direct Mix	79.8	27.0	13.0	5.0	**85.0**	38.6	19.8	3.8	72.2	8.0	4.6	0.8	50.0	10.6	3.4	0.0	71.8	21.1	10.2	2.4
STAR	**83.4**	**66.6**	**41.8**	**13.8**	81.0	**59.2**	**28.8**	**19.2**	69.0	**22.0**	**7.2**	**0.6**	45.4	**21.6**	**8.0**	**3.0**	69.7	**42.4**	**21.5**	**9.2**
Δ vs. OpenVLA	+0.2	+49.0	+28.2	+12.0	−3.8	+34.8	+11.8	+18.2	−5.8	+19.8	+6.4	+0.2	−5.2	+16.2	+6.2	+3.0	−3.7	+30.0	+13.2	+8.4

from 7.2% to 11.2%, while sparse macro success changes from 73.4% to 71.8%. STAR reaches 24.3% dense macro success while retaining 69.7% sparse macro success. Its dense mean exceeds Direct Mix on every suite: 40.7% versus 15.0% on Spatial, 35.7% versus 20.7% on Object, 9.9% versus 4.5% on Goal, and 10.9% versus 4.7% on Long. Direct Mix improves dense access but leaves Goal and Long below 5%, whereas STAR establishes a stronger sparse–dense balance across the complete comparison. Goal and Long remain the most difficult settings.

5.3 SimplerEnv Results

SimplerEnv asks whether the same adaptation objective extends beyond LIBERO. Sparse-FT initializes STAR and supplies action targets, while the corresponding pre-action VLM supplies representation targets. We evaluate dense access and sparse retention relative to the control available in Sparse-FT.

Bridge. Bridge starts from a source policy with limited task success. Sparse-FT reaches 5.2% under sparse instructions and a dense mean of 0.7% in Tab. 7. STAR raises these values to 8.3% and 4.5%, respectively, and outperforms Sparse-FT at all three dense levels. The consistent gains show that STAR broadens language access from this control baseline.

Mode	Method	Spoon on Towel	Carrot on Plate	Stack Cube	Eggplant in Basket	Avg.
Sparse	OpenVLA	4.2	0.0	0.0	12.5	4.2
	Sparse-FT	4.2	4.2	4.2	8.3	5.2
	STAR	8.3	8.3	4.2	12.5	**8.3** (+4.1)
Simple	OpenVLA	0.0	0.0	0.0	0.0	0.0
	Sparse-FT	0.0	4.2	4.2	0.0	2.1
	STAR	8.3	8.3	4.2	8.3	**7.3** (+7.3)
Middle	OpenVLA	0.0	0.0	0.0	0.0	0.0
	Sparse-FT	0.0	0.0	0.0	0.0	0.0
	STAR	4.2	4.2	4.2	4.2	**4.2** (+4.2)
Hard	OpenVLA	0.0	0.0	0.0	0.0	0.0
	Sparse-FT	0.0	0.0	0.0	0.0	0.0
	STAR	0.0	4.2	0.0	4.2	**2.1** (+2.1)

Table 7: Bridge SimplerEnv success rates (%). Sparse-FT initializes and supervises STAR. OpenVLA has no Bridge target-data fine-tuning. Avg. is the task macro. Gray marks OpenVLA, bold marks STAR averages, and parentheses report STAR − OpenVLA (pp).

Fractal. Fractal asks how the result changes between Visual Matching (VM) and Variant Aggregation (VA). Under VA, STAR preserves sparse success at 42.5% and raises dense mean success

from 12.5% to 24.1%. Under VM, sparse success rises from 29.2% to 32.3%, and dense mean success rises from 0.7% to 2.4%. STAR broadens dense access most in VA, where the source policy already provides stronger control. Put in Drawer remains unresolved for both Sparse-FT and STAR, connecting the outcome to the control behavior available in the source policy.

Mode	Method	Visual Matching (VM)					Variant Aggregation (VA)				
		Coke Can	Move Near	O./C. Drawer	Put in Drawer	**Avg.**	Coke Can	Move Near	O./C. Drawer	Put in Drawer	**Avg.**
Sparse	OpenVLA	18.0	56.3	63.0	0.0	**34.3**	60.8	67.7	28.8	0.0	39.3
	Sparse-FT	58.3	8.3	50.0	0.0	29.2	65.9	63.2	39.8	0.0	42.2
	STAR	58.3	4.2	66.7	0.0	32.3 (−2.0)	66.9	63.5	39.5	0.0	**42.5** (+3.2)
Simple	OpenVLA	0.0	0.0	0.0	0.0	0.0	23.9	22.5	10.3	5.6	15.6
	Sparse-FT	0.0	0.0	0.0	0.0	0.0	32.7	29.7	13.3	0.0	18.9
	STAR	0.0	0.0	0.0	0.0	0.0	49.5	45.8	41.6	0.0	**34.2** (+18.6)
Middle	OpenVLA	0.0	0.0	4.2	0.0	1.1	15.6	14.8	7.5	0.0	9.5
	Sparse-FT	0.0	0.0	0.0	0.0	0.0	8.8	21.7	29.2	0.0	14.9
	STAR	0.0	0.0	16.7	0.0	**4.2** (+3.1)	14.7	27.7	55.6	0.0	**24.5** (+15.0)
Hard	OpenVLA	8.3	0.0	0.0	0.0	2.1	7.4	8.3	1.7	0.0	4.4
	Sparse-FT	8.3	0.0	0.0	0.0	2.1	5.7	6.2	2.9	0.0	3.7
	STAR	12.5	0.0	0.0	0.0	**3.1** (+1.0)	13.2	8.7	31.9	0.0	**13.5** (+9.1)

Table 8: Fractal SimplerEnv success rates (%) under Visual Matching (VM) and Variant Aggregation (VA). Sparse-FT initializes and supervises STAR. OpenVLA has no Fractal target-data fine-tuning. VM drawer scores average six variants. Gray marks OpenVLA, bold marks the highest nonzero macro average, and parentheses give STAR − OpenVLA (pp).

Table 9: STAR ablations on LIBERO-Spatial (%). Panel (a) starts from suite-fine-tuned OpenVLA. Gray marks the reference, checkmarks enabled targets, and bold the panel best.

(a) Alignment targets

Variant	VLM Rep.	VLA KD	S	D_S	D_M	D_H
OpenVLA	×	×	83.2	17.6	13.6	1.8
Rep. only	✓	×	64.3	58.1	21.7	12.4
Act. only	×	✓	77.0	54.4	29.6	9.0
STAR	✓	✓	**83.4**	**66.6**	**41.8**	**13.8**

(b) Density-level mixture

Mixture	S	D_S	D_M	D_H
Simple-heavy	84.8	**68.6**	**44.2**	9.6
Balanced	83.4	66.6	41.8	13.8
Hard-heavy	**85.6**	56.8	41.8	**14.0**

(c) Dense exposure

Exposure	S	D_S	D_M	D_H
25%	**86.6**	54.6	35.0	7.8
50%	82.6	58.8	36.0	7.6
100%	83.4	**66.6**	**41.8**	**13.8**

5.4 Component and Data Ablations

These ablations ask how the two alignment targets, density-level sampling, and amount of dense exposure shape STAR's sparse–dense balance.

Panel (a) of Tab. 9 separates the two teacher targets. Representation-only improves all dense levels but lowers sparse success to 64.3%; action-only retains 77.0% sparse success while improving each dense level. Combining them gives the best result in every column, showing that representation and action alignment play complementary roles.

Panel (b) tests density-level sampling. Simple-heavy is strongest on Simple and Middle, whereas Hard-heavy is strongest on Hard. The balanced mixture stays within 2.4 points of the best result in every dense column, providing broad coverage for the main STAR model.

Panel (c) tests the amount of paired dense exposure. Full exposure exceeds 25% exposure by 12.0 points on Simple, 6.8 points on Middle, and 6.0 points on Hard, while sparse success stays between 82.6% and 86.6%. More paired exposure improves every dense level without displacing the sparse route.

5.5 General VQA after STAR Adaptation

The diagnostic in Tab. 3 showed that VQA accuracy declines after action fine-tuning. Here we test how much of that capability remains after STAR adaptation. STAR raises OpenVLA accuracy from 31.7% to 58.1% on VQAv2 and from 27.5% to 40.4% on

Benchmark	Pre-action VLM	OpenVLA	STAR
VQAv2	79.1	31.7	58.1
GQA	64.0	27.5	40.4

Table 10: Standard VQA accuracy (%).

GQA (Tab. 10), recovering 26.4 and 12.9 percentage points, respectively. This partial recovery complements the robot-control gains.

6 Related Work

6.1 Generalist VLA Models and Action Modeling

RT-2 and OpenVLA tokenize actions, whereas Octo, π_0, CogACT, FAST, and MemoryVLA use transformer, diffusion, flow, and memory-based generation Zitkovich et al. (2023); Mees et al. (2024); Kim et al. (2024); Black et al. (2026); Li et al. (2024); Pertsch et al. (2025); Shi et al. (2026). SpatialVLA develops explicit spatial representations for VLA policies Qu et al. (2025). These works broaden the tasks and embodiments a policy can execute. We study whether an already learned task remains accessible through denser task-equivalent language.

6.2 Language Robustness and Evaluation for VLAs

Canonical VLA benchmarks use narrow task commands Liu et al. (2023); Wanna et al. (2026). LIBERO-Plus, LIBERO-PRO, LIBERO-X, and VLA-Arena broaden evaluation. Language diagnostics test paraphrases, semantic changes, references, irrelevant context, and difficulty Fei et al. (2025); Zhou et al. (2026); Wang et al. (2026); Zhang et al. (2026); Pugacheva et al. (2025); Kim et al. (2026); Hou & Zhao (2026). Our benchmark holds the executable task and rollout conditions fixed, then orders task-equivalent forms by the interpretation needed to recover the task.

6.3 Language Adaptation and Capability Retention

Language adaptation uses relabelling, augmentation, consistency, rewriting, or action-expert pretraining Kuramshin et al. (2026); Zhan et al. (2026); Luo et al. (2026); Shodiev et al. (2026); Xu et al. (2026). INT-ACT diagnoses visual-language erosion Fang et al. (2025). Other work retains pretrained capability through joint supervision, isolation, alignment, domain recovery, or parameter merging Zhou et al. (2025); Hancock et al. (2026); Driess et al. (2025); Yang et al. (2026); Kachaev et al. (2025); Dey et al. (2025); Yadav et al. (2026). For the task-equivalent interface shift, STAR pairs pre-action representation targets with the sparse-conditioned action distribution of the source VLA.

7 Conclusion

Task-equivalent language robustness asks whether a learned task remains accessible through another valid language form. IDS reveals low-density source coverage. The Sparse–Dense Gap reveals model- and task-dependent access failures. STAR addresses the resulting adaptation problem with paired action learning, pre-action representation alignment, and source-action alignment. STAR raises LIBERO dense macro success from 7.2% to 24.3% as sparse success changes from 73.4% to 69.7%. On Fractal VA, dense mean success rises from 12.5% to 24.1% while sparse success changes from 42.2% to 42.5%. These results establish a stronger balance between dense access and source-interface retention.

Limitations. Our simulation study covers constructed language and OpenVLA-derived policies. It does not capture unconstrained human interaction, and the two frozen teachers increase training-time computation.

AI USE STATEMENT

In this work, we used generative AI tools during benchmark construction: DeepSeek-V4-Flash generated first-pass dense-instruction rewrites, and GPT-5.6-Sol performed rubric-based scoring and refinement. OpenAI Codex was used for manuscript language editing and LaTeX formatting. The authors independently designed the research question, method, benchmark rules, experiments, and evaluation protocol, and collected and verified the references. All AI-assisted outputs were reviewed by the authors, who take responsibility for the final content of this work.

REFERENCES

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael 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, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control, 2026. URL https://arxiv.org/abs/2410.24164.

DeepSeek-AI, Anyi Xu, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, Chenchen Ling, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chengyu Hou, Chenhao Xu, Chenze Shao, Chong Ruan, Conner Sun, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Donghao Li, Dongjie Ji, Erhang Li, Fang Wei, Fangyun Lin, Fangzhou Yuan, Feiyu Xia, Fucong Dai, Guangbo Hao, Guanting Chen, Guoai Cao, Guolai Meng, Guowei Li, Han Yu, Han Zhang, Hanwei Xu, Hao Li, Haofen Liang, Haoling Zhang, Haoming Luo, Haoran Wei, Haotian Yuan, Haowei Zhang, Haowen Luo, Haoyu Chen, Haozhe Ji, Hengqing Zhang, Honghui Ding, Hongxuan Tang, Huanqi Cao, Huazuo Gao, Hui Qu, Hui Zeng, J Yang, JQ Zhu, Jia Luo, Jia Song, Jia Yu, Jialiang Huang, Jialu Cai, Jian Liang, Jiangting Zhou, Jiasheng Ye, Jiashi Li, Jiaxin Xu, Jiewen Hu, Jieyu Yang, Jin Chen, Jin Yan, Jingchang Chen, Jingli Zhou, Jingting Xiang, Jingyang Yuan, Jingyuan Cheng, Jingzi Zhou, Jinhua Zhu, Jiping Yu, Joseph Sun, Jun Ran, Junguang Jiang, Junjie Qiu, Junlong Li, Junmin Zheng, Junxiao Song, Kai Dong, Kaige Gao, Kang Guan, Kexing Zhou, Kezhao Huang, Kuai Yu, Lean Wang, Lecong Zhang, Lei Wang, Leyi Xia, Li Zhang, Liang Zhao, Lihua Guo, Lingxiao Luo, Linwang Ma, Linyan Zhu, Litong Wang, Liyu Cai, Liyue Zhang, Longhao Chen, MS Di, MY Xu, Max Mei, Miaojun Wang, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingming Li, Mingxu Zhou, Minmin Han, Ning Wang, Panpan Huang, Panpan Wang, Peixin Cong, Peiyi Wang, Peng Zhang, Qiancheng Wang, Qihao Zhu, Qingyang Li, Qinyu Chen, Qiushi Du, Qiwei Jiang, Rui Tian, Ruifan Xu, Ruijie Lu, Ruiling Xu, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, Runqian Chen, Runqiu Yin, Runxin Xu, Ruomeng Shen, Ruoyu Zhang, Ruyi Chen, SH Liu, Shanghao Lu, Shangmian Sun, Shangyan Zhou, Shanhuang Chen, Shaofei Cai, Shaoheng Nie, Shaoqing Wu, Shaoyuan Chen, Shengding Hu, Shengyu Liu, Shiqiang Hu, Shirong Ma, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, Shuying Yu, Songyang Zhou, Tao Ni, Tao Yun, Tian Jin, Tian Pei, Tian Ye, Tianle Lin, Tianran Ji, Tianyi Cui, Tianyuan Yue, Tingting Yu, Tun Wang, W Zhang, WL Xiao, Wangding Zeng, Wei An, Weilin Zhao, Wen Liu, Wenfeng Liang, Wenjie Pang, Wenjing Luo, Wenjing Yao, Wenjun Gao, Wenkai Yang, Wenlve Huang, Wenqing Hou, Wentao Zhang, Wenting Ma, Xi Gao, Xiang He, Xiangwen Wang, Xianzu Wang, Xiao Bi, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaokang Zhang, Xiaotao Nie, Xiaowen Sun, Xiaoxiang Wang, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xingchen Liu, Xingkai Yu, Xingyou Li, Xinyu Yang, Xinyu Zhang, Xu Chen, Xuanyu Wang, Xuecheng Su, Xueyin Chen, Xuheng Lin, Xuwei Fu, YC Yan, YQ Wang, YW Ma, Yanfeng Luo, Yang Zhang, Yanhong Xu, Yanru Ma, Yanwen Huang, Yao Li, Yao Li, Yao Xu, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Qian, Yi Shao, Yi Yu, Yichao Zhang, Yifan Ding, Yifan Shi, Yijia Wu, Yiliang Xiong, Yiling Ma, Ying He, Ying Tang, Ying Zhou, Yingjia Luo, Yinmin Zhong, Yishi Piao, Yisong Wang, Yixiang Zhang, Yixiao Chen, Yixuan Tan, Yixuan Wei, Yiyang Ma, Yiyuan Liu, Yonglun Yang, Yongqiang Guo, Yongtong Wu, Yu Wu, YuKun Li, Yuan Cheng, Yuan Ou, Yuanfan Xu, Yuanhao Li, Yuduan Wang, Yuehan Yang, Yuer Xu, Yuhan Wu, Yuhao Meng, Yuheng Zou, Yukun Zha, Yunfan Xiong, Yupeng Chen, Yuping Lin, Yuqian Cao, Yuqian Wang, Yushun Zhang, Yuting Yan, Yutong Lin, Yuxian Gu, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuxuan Zhou, Yuyang Zhou, Yuzhen Huang, ZF Wu, Zehao Wang, Zehua Zhao, Zehui Ren, Zekai Zhang, Zhangli Sha, Zhe Fu, Zhe Ju, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zheren Gao, Zhewen Hao, Zhibin Gou, Zhicheng Ma, Zhigang Yan, Zhihong

Shao, Zhixian Huang, Zhixuan Chen, Zhiyu Wu, Zhizhou Ren, Zhongyu Wu, Zhuoshu Li, Zhuping Zhang, Zian Xu, Zihao Wang, Zihua Qu, Zihui Gu, Zijia Zhu, Zilin Li, Zipeng Zhang, Ziwei Xie, Ziyi Gao, Ziyi Wan, Zizheng Pan, and Zongqing Yao. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026. URL https://arxiv.org/abs/2606.19348.

Sombit Dey, Jan-Nico Zaech, Nikolay Nikolov, Luc Van Gool, and Danda Pani Paudel. Revla: Reverting visual domain limitation of robotic foundation models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8679–8686. IEEE, 2025.

Danny Driess, Jost Springenberg, Brian Ichter, LILI YU, Adrian Li-Bell, Karl Pertsch, Allen Ren, Homer Walke, Quan Vuong, Lucy Xiaoyang Shi, and Sergey Levine. Knowledge insulating vision-language-action models: Train fast, run fast, generalize better. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 102867–102888. Curran Associates, Inc., 2025. doi: 10.52202/085713-3439. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/94e936034d12bcd04834ec2773f02aff-Paper-Conference.pdf.

Irving Fang, Juexiao Zhang, Shengbang Tong, and Chen Feng. From intention to execution: Probing the generalization boundaries of vision-language-action models, 2025. URL https://arxiv.org/abs/2506.09930.

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. Libero-plus: In-depth robustness analysis of vision-language-action models, 2025. URL https://arxiv.org/abs/2510.13626.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

Asher Hancock, Xindi Wu, Lihan Zha, Olga Russakovsky, and Anirudha Majumdar. Actions as language: Fine-tuning vlms into vlas without catastrophic forgetting. In *International Conference on Learning Representations*, volume 2026, pp. 75333–75360, 2026.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Yuchen Hou and Lin Zhao. Langgap: Diagnosing and closing the language gap in vision-language-action models, 2026. URL https://arxiv.org/abs/2603.00592.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Drew A. Hudson and Christopher D. Manning. 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 (CVPR)*, June 2019.

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.

Nikita Kachaev, Mikhail Kolosov, Daniil Zelezetsky, Alexey K. Kovalev, and Aleksandr I. Panov. Don't blind your vla: Aligning visual representations for ood generalization, 2025. URL https://arxiv.org/abs/2510.25616.

Siddharth Karamcheti, Suraj Nair, Ashwin Balakrishna, Percy Liang, Thomas Kollar, and Dorsa Sadigh. Prismatic VLMs: Investigating the design space of visually-conditioned language models. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 23123–23144. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/karamcheti24a.html.

Chanyoung Kim, Minwoo Kim, Minseok Kang, Hyunwoo Kim, and Dahuin Jung. Libero-para: A diagnostic benchmark and metrics for paraphrase robustness in vla models, 2026. URL https://arxiv.org/abs/2603.28301.

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.

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-tuning vision-language-action models: Optimizing speed and success. *arXiv preprint arXiv:2502.19645*, 2025.

Artur Kuramshin, Özgür Aslan, Cyrus Neary, and Glen Berseth. Task robustness via re-labelling vision-action robot data, 2026. URL https://arxiv.org/abs/2606.10918.

Qixiu Li, Yaobo Liang, Zeyu Wang, Lin Luo, Xi Chen, Mozheng Liao, Fangyun Wei, Yu Deng, Sicheng Xu, Yizhong Zhang, Xiaofan Wang, Bei Liu, Jianlong Fu, Jianmin Bao, Dong Chen, Yuanchun Shi, Jiaolong Yang, and Baining Guo. Cogact: A foundational vision-language-action model for synergizing cognition and action in robotic manipulation, 2024. URL https://arxiv.org/abs/2411.19650.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Oier Mees, Karl Pertsch, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. Evaluating real-world robot manipulation policies in simulation. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 3705–3728. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/li25c.html.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 44776–44791. Curran Associates, Inc., 2023. doi: 10.52202/075280-1939. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/8c3c666820ea055a77726d66fc7d447f-Paper-Datasets_and_Benchmarks.pdf.

Jingzhou Luo, Yifan Wen, Yongjie Bai, Xinshuai Song, Yang Liu, and Liang Lin. Rovla: Multi-consistency constraints for robust vision-language-action models, 2026. URL https://arxiv.org/abs/2605.19678.

Oier Mees, Dibya Ghosh, Karl Pertsch, Kevin Black, Homer Rich Walke, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, et al. Octo: An open-source generalist robot policy. In *First workshop on vision-language models for navigation and manipulation at ICRA 2024*, 2024.

Hewen Pan, Cong Wei, Dashuang Liang, Zepeng Huang, Pengfei Gao, Ziqi Zhou, Lulu Xue, Pengfei Yan, Xiaoming Wei, Minghui Li, et al. Ufvideo: Towards unified fine-grained video cooperative understanding with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 31833–31844, 2026.

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, 2025. URL https://arxiv.org/abs/2501.09747.

Daria Pugacheva, Andrey Moskalenko, Denis Shepelev, Andrey Kuznetsov, Vlad Shakhuro, and Elena Tutubalina. Bring the apple, not the sofa: Impact of irrelevant context in embodied ai commands on vla models, 2025. URL https://arxiv.org/abs/2510.07067.

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

Hao Shi, Bin Xie, Yingfei Liu, Lin Sun, Fengrong Liu, Tiancai Wang, Erjin Zhou, Haoqiang Fan, Xiangyu Zhang, and Gao Huang. Memoryvla: Perceptual-cognitive memory in vision-language-action models for robotic manipulation. In *International Conference on Learning Representations*, volume 2026, pp. 18567–18602, 2026.

Damir Shodiev, Aleksei Staroverov, Nikita Kachaev, Alexey K. Kovalev, and Aleksandr I. Panov. Vla grounder: Language-conditioning space optimization for black-box vla models, 2026. URL https://arxiv.org/abs/2607.04517.

Guodong Wang, Chenkai Zhang, Qingjie Liu, Jinjin Zhang, Jiancheng Cai, Junjie Liu, and Xinmin Liu. Libero-x: Robustness litmus for vision-language-action models, 2026. URL https://arxiv.org/abs/2602.06556.

Selma Liliane Wanna, Agnes Luhtaru, Jonathan Salfity, Ryan Barron, Juston Moore, Cynthia Matuszek, and Mitch Pryor. Limited linguistic diversity in embodied ai datasets. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11058–11086, 2026.

Kechun Xu, Zhenjie Zhu, Anzhe Chen, Rong Xiong, and Yue Wang. Apt: Action expert pretraining improves instruction generalization of vision-language-action policies, 2026. URL https://arxiv.org/abs/2606.12366.

Yajat Yadav, Zhiyuan Zhou, Andrew Wagenmaker, Karl Pertsch, and Sergey Levine. Robust fine-tuning of vision-language-action robot policies via parameter merging. In *International Conference on Learning Representations*, volume 2026, pp. 56385–56417, 2026.

Shuai Yang, Hao Li, Bin Wang, Yilun Chen, Yang Tian, Tai Wang, Hanqing Wang, Feng Zhao, Yiyi Liao, and Jiangmiao Pang. Vision-language-action instruction tuning: From understanding to manipulation. In *International Conference on Learning Representations*, volume 2026, pp. 152896–152943, 2026.

Zhihao Zhan, Yuhao Chen, Jiaying Zhou, Qinhan Lyu, Hao Liu, Keze Wang, Liang Lin, and Guangrun Wang. Stable language guidance for vision–language–action models. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 4137–4159, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-390-6. doi: 10.18653/v1/2026.acl-long.190. URL https://aclanthology.org/2026.acl-long.190/.

Borong Zhang, Jiahao Li, Jiachen Shen, Yuhao Zhang, Yishuai Cai, Lu Liu, Hailu Ji, Yuanpei Chen, Juntao Dai, Jiaming Ji, and Yaodong Yang. Vla-arena: An open-source framework for benchmarking vision-language-action models, 2026. URL https://arxiv.org/abs/2512.22539.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623. Curran Associates, Inc., 2023. doi: 10.52202/075280-2020. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/91f18a1287b398d378ef22505bf41832-Paper-Datasets_and_Benchmarks.pdf.

Xueyang Zhou, Yangming Xu, Guiyao Tie, Yongchao Chen, Guowen Zhang, Duanfeng Chu, Pan Zhou, and Lichao Sun. Libero-pro: Towards robust and fair evaluation of vision-language-action models beyond memorization, 2026. URL https://arxiv.org/abs/2510.03827.

Zhongyi Zhou, Yichen Zhu, Minjie Zhu, Junjie Wen, Ning Liu, Zhiyuan Xu, Weibin Meng, Yaxin Peng, Chaomin Shen, Feifei Feng, and Yi Xu. ChatVLA: Unified multimodal understanding and robot control with vision-language-action model. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 5377–5395, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-332-6. doi: 10.18653/v1/2025.emnlp-main.273. URL https://aclanthology.org/2025.emnlp-main.273/.

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 (eds.), *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183. PMLR, 06–09 Nov 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.