**Robotics: Science and Systems 2026**
**Sydney, Australia, July 13–17, 2026**

# GuidedVLA: Specifying Task-Relevant Factors via Plug-and-Play Action Attention Specialization

Xiaosong Jia[*†,1,2], Bowen Yang[*,3], Zuhao Ge[*,1,2], Xian Nie[*,3], Yuchen Zhou[*,1,2], Cunxin Fan[*†,3], Yufeng Li[3], Yilin Chai[3], Chao Jing[1,2], Zijian Liang[3], Qingwen Bu[4], Haidong Cao[1,2], Chao Wu[1,2], Qifeng Li[3], Zhenjie Yang[3], Chenhe Zhang[1,2], Hongyang Li[4], Zuxuan Wu[✉1,2], Junchi Yan[✉3], Yu-Gang Jiang[✉1,2]

[1]Institute of Trustworthy Embodied AI (TEAI), Fudan University
[2]Shanghai Key Laboratory of Multimodal Embodied AI
[3] Shanghai Jiao Tong University
[4] OpenDriveLab, The University of Hong Kong

* Core Contributors † Project Lead ✉ Correspondence Authors

https://guidedvla.github.io/project_page/

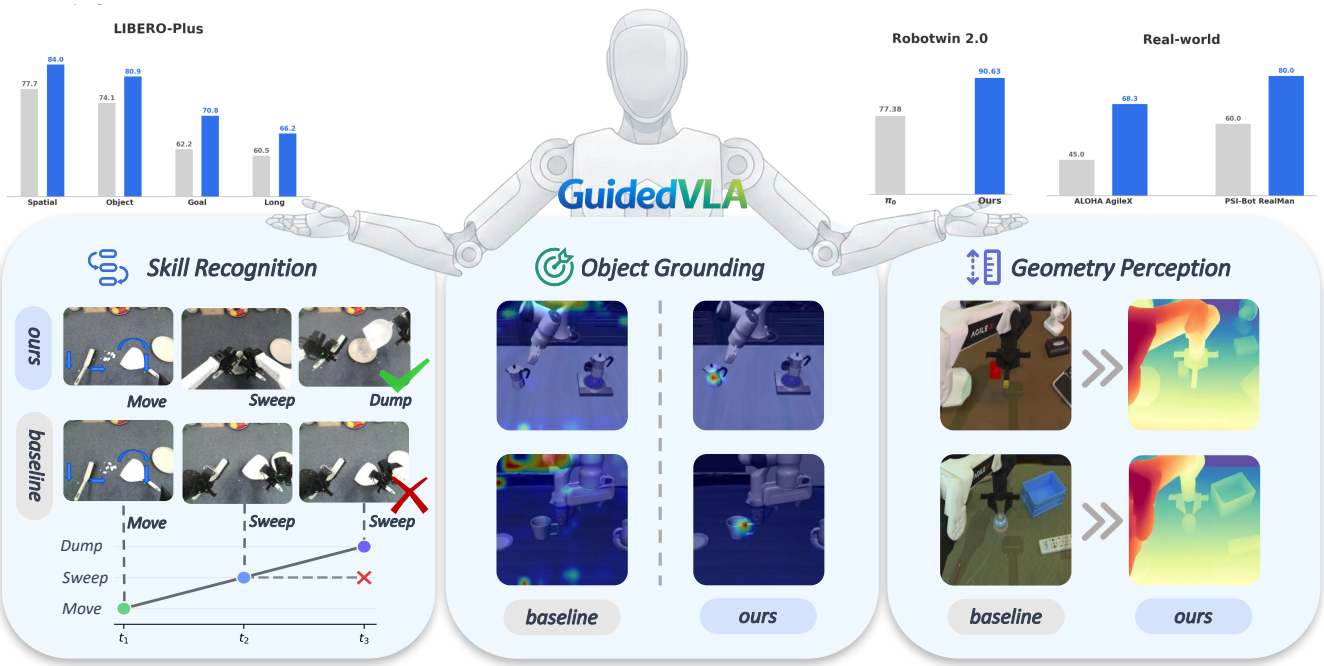



Fig. 1: We present **GuidedVLA**, a VLA paradigm in which the action decoder is explicitly guided to capture task-relevant information such as object grounding, spatial geometry, and temporal skill logic. Across simulation and real-robot experiments, GuidedVLA significantly improves success rates in both in-domain and out-of-domain settings, demonstrating the effectiveness of specifying action-decoder attention heads with explicit guidance.

***Abstract*—Vision-Language-Action (VLA) models aim for general robot learning by aligning action as a modality within powerful Vision-Language Models (VLMs). Existing VLAs rely on end-to-end supervision to implicitly enable the action decoding process to learn task-relevant features. However, without explicit guidance, these models often overfit to spurious correlations, such as visual shortcuts or environmental noise, limiting their generalization. In this paper, we introduce GuidedVLA, a framework designed to manually guide the action generation to focus on task-relevant factors. Our core insight is to treat the action decoder not as a monolithic learner, but as an assembly of functional components. Individual attention heads are supervised by manually defined auxiliary signals to capture distinct factors. As an initial study, we instantiate this paradigm with three specialized heads: object grounding, spatial geometry, and temporal skill logic. Across simulation and real-robot experiments, GuidedVLA improves success rates in both in-domain and out-of-domain settings compared to strong VLA baselines. Finally, we show that the quality of these specialized factors correlates positively with task performance and that our mechanism yields decoupled, high-quality features. Our results suggest that explicitly guiding action-decoder learning is a promising direction for building more robust and general VLA models.**

## I. Introduction

Vision-Language-Action (VLA) models [104, 45, 8] represent a significant step toward generalist robot policies by

integrating action as a specialized modality within the rich feature space of Vision-Language Models (VLMs). By leveraging the massive pre-training of VLMs [78, 79, 4, 51], these models can inherit high-level semantic knowledge and reasoning capabilities essential for complex tasks. However, current VLA training typically relies on end-to-end supervision where the action decoder is expected to implicitly learn task-relevant factors from demonstration data [10]. According to pioneering studies in the field of computer vision and imitation learning, end-to-end learning without explicit guidance may lead to shortcut learning [30, 29] or causal confusion [21].

In practice, we observe that the action decoder of VLA often latches onto spurious correlations, such as background textures or incidental camera artifacts, as shown in Fig. 1. While some cross-attention heads in the action decoder occasionally attend to relevant regions, this behavior is highly stochastic and varies across different heads and scenarios. This randomness suggests that although VLM backbones provide a robust feature stream, **the action decoder does not learn a stable, causal understanding of the task, but instead relies on a shifting set of features** and thus struggles to generalize.

Because end-to-end supervision alone makes the decision-making process of VLA opaque and inconsistent, we propose **GuidedVLA**, a framework that transforms the action decoder from a monolithic learner into an assembly of functionally specialized components. Instead of allowing the cross-attention heads to develop roles implicitly, we manually specify the information each head should capture by supervising them with distinct, task-relevant auxiliary signals.

While this paradigm is designed to be general and extensible, in this work, we instantiate it by supervising three primary factors, as in Fig. 1: (i) **object grounding**, ensuring action tokens attend to task-relevant regions; (ii) **skill recognition**, enabling action tokens to identify the intended sub-skill or phase of a multi-step behavior; and (iii) **geometry perception**, allowing action tokens to leverage 3D spatial information. Our probing experiments reveal that current VLAs are brittle across all three factors, and that the proposed guidance mechanism effectively resolves these deficiencies.

Across multiple simulation benchmarks and real-world experiments, GuidedVLA achieves a significant performance boost for $\pi_0$ [8], surpassing other recent feature-training methods in the field [96, 75]. Furthermore, we provide a quantitative evaluation showing a strong positive correlation between factor understanding and overall success rates. Finally, we validate that partitioning factors into specialized attention heads produces better-decoupled features than a mixture approach where all heads are jointly supervised.

In summary, we make the following contributions:

- We propose GuidedVLA, a general paradigm for VLA that mitigates overfit risk by specifying task-relevant factors through functional attention specialization.
- We instantiate this framework by designing three specialized heads: object grounding, skill recognition, and geometry perception and demonstrate through probing that such explicit guidance resolves the inherent brittleness and stochasticity of unguided VLA decoders.
- We provide extensive evaluations across multiple simulation benchmarks and real-robot tasks, showing that GuidedVLA significantly improves state-of-the-art baselines in both in-domain and out-of-distribution scenarios.
- We offer quantitative insights into head specialization, validating that our approach yields high-quality features that correlate positively with task performance.

## II. Related Work

**Vision-Language-Action (VLA) Models**: Vision-Language-Action (VLA) models aim to map visual observations and language instructions to low-level robot actions by combining pretrained vision-language models with large-scale robot demonstrations [2, 10, 104, 23, 31, 45, 6, 101, 90]. One important research direction focuses on scaling embodied data through multi-source datasets [67, 81, 43, 20, 63, 40, 48], standardized multi-task benchmarks [64, 59, 28, 39, 92], and evaluations under distribution shift [28, 65, 24]. Another line of work improves training and inference recipes, including multimodal prompting [42, 25], parameter-efficient adaptation [44, 87, 74, 34], and inference-time acceleration [88, 9, 41, 62, 99]. In parallel, prior work strengthens the action pathway through alternative action parameterizations and learning objectives, including diffusion- or flow-based generation [18, 8, 7, 60, 17, 55, 14, 86], action chunking for temporal abstraction [98], and discrete or compressed action tokenizers to better match control bandwidth [69, 85].

**Auxiliary Tasks for Robotics Models** Structured intermediate representations improve policy robustness under distribution shift. Object-centric methods factor manipulation around task-relevant entities such as object poses, keypoints, and relations [36, 32, 58, 33, 89, 53, 50, 49, 68, 13]. Skill-based representations support long-horizon reasoning by decomposing tasks into reusable subgoals [2, 54, 61, 35, 27, 97, 1]. Geometry-aware policies that operate on 3D representations, including point clouds and 3D scene tokens, achieve strong viewpoint and instance generalization [94, 83, 56, 100, 71, 19, 91, 70, 22, 26, 76, 93, 47, 52, 46, 37, 66, 102, 5]. Our work complements these approaches by supervising object-centric structure, skills, and geometry into separable internal pathways within the action decoder of VLA.

## III. Method

### A. Problem Setup and Motivation

Recent representative Vision-Language-Action (VLA) models [8, 45, 104] extend Vision-Language Models (VLMs) by introducing action tokens $\boldsymbol{a}$ alongside vision tokens $\boldsymbol{v}$ and language tokens $\boldsymbol{l}$. The action generation process is trained to regress robot trajectories by denoising $\boldsymbol{a}$ conditioned on $(\boldsymbol{v}, \boldsymbol{l})$.

Although VLMs provide rich semantic features in $(\boldsymbol{v}, \boldsymbol{l})$, the action tokens $\boldsymbol{a}$ do not inherently learn to extract task-critical information. As in Figure 1, action token attention often diffuses over irrelevant background regions. This motivates our central designs: ***How can we guide action decoding to extract task-relevant information?***

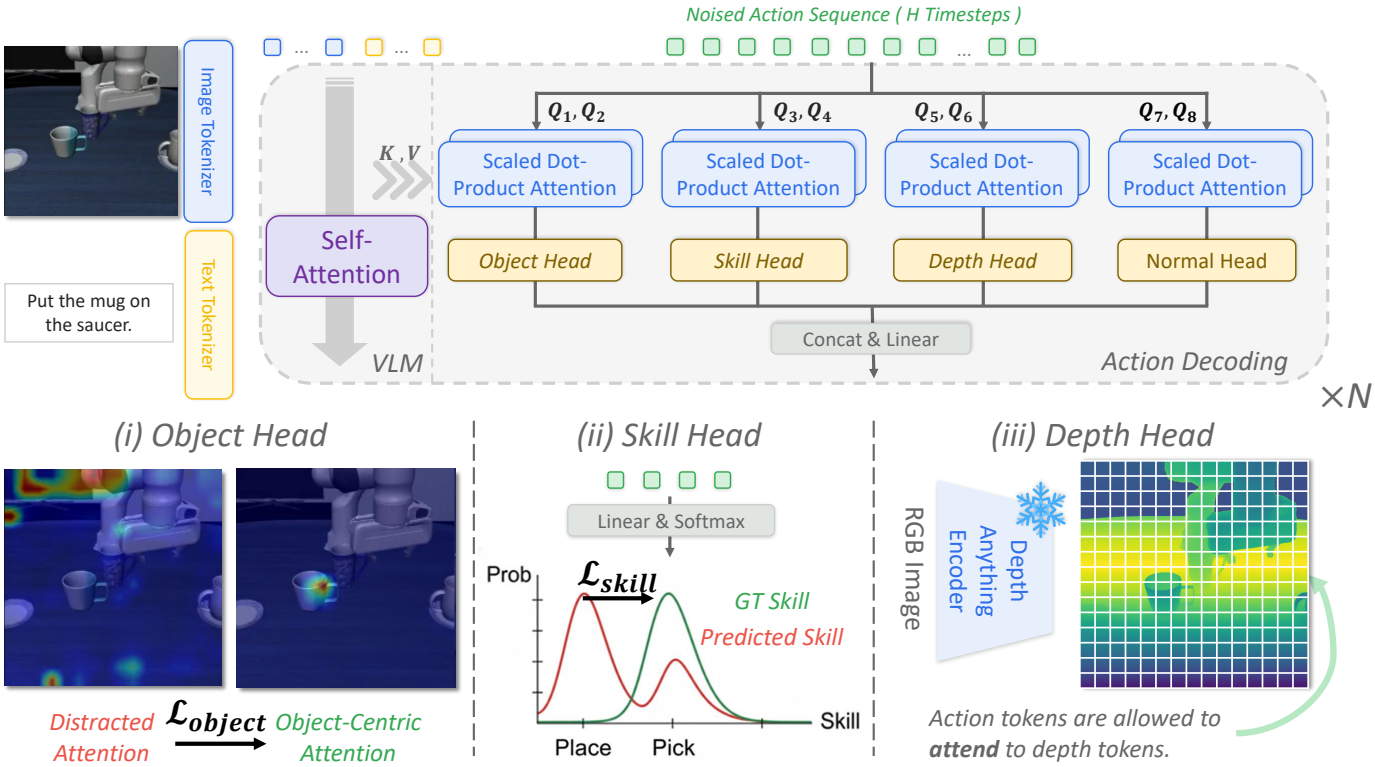



Fig. 2: **Architecture of GuidedVLA.** We introduce explicit, structured guidance into the multi-head attention layers of the VLA action decoder. Instead of relying on implicitly entangled representations, we repurpose dedicated attention heads to specialize in distinct task-relevant factors: **(i) Object Head** supervises its attention maps to explicitly ground task-relevant objects and suppress distractors via $\mathcal{L}_{object}$; **(ii) Skill Head** aligns internal feature representations with temporal skill phases (e.g., Pick → Place) through auxiliary classification $\mathcal{L}_{skill}$; **(iii) Depth Head** injects geometric cues via cross attention only to features from a depth encoder. These guidance forces the policy to explicitly aware spatial, temporal, and geometric structures.

**Algorithm 1** Decoupled Attention with Guided Heads

**Require:** Action hidden states $X_L$ at layer $L$
**Require:** Head sets: $\mathcal{H}_o$ (object), $\mathcal{H}_s$ (skill), $\mathcal{H}_d$ (depth)
**Require:** Joint cache $(K, V)$; Depth cache $(K^d, V^d)$
**Ensure:** Fused attention output $A_L$
$Q \leftarrow \text{Proj}_\text{Q}(X_L)$

Stage 1: Factor-Specific Attention
$P_o, A_o \leftarrow \text{Attn}(Q[\mathcal{H}_o], K, V)$ ▷ Object Head
$A_s \leftarrow \text{Attn}(Q[\mathcal{H}_s], K, V)$ ▷ Skill Head
$A_d \leftarrow \text{Attn}(Q[\mathcal{H}_d], K^d, V^d)$ ▷ Depth Head

Stage 2: Per-Head Supervision
Apply $\mathcal{L}_{object}$ (Eq. 4) to $P_o$
Apply $\mathcal{L}_{skill}$ (Eq. 7) to $A_s$

Stage 3: ControlNet-style Residual Fusion
$A_L^{\text{specified}} \leftarrow \text{Proj}_\text{O}\big(\text{Concat}(A[:])\big)$
$A_L \leftarrow \text{ZeroConv}(A_L^{\text{specified}}) + A_L^{\text{main}}$ ▷ Merge with Main Branch

### B. What to Guide: Three Task-Relevant Factors

We identify three factors correlated with robotics tasks, based on our preliminary probing experiments in Sec. V-B:

1) **Object Grounding**: whether action tokens can attend to the correct task-relevant regions (e.g., affordance).
2) **Skill Recognition**: whether action tokens correctly identify the current sub-skill or temporal phase within a complex robotics task (e.g., long-horizon).
3) **Geometry Perception**: whether action tokens can utilize 3D spatial information when performing fine-grained tasks (e.g., click bell).

These factors are complementary: grounding localizes the target, skill recognition defines the behavior, and geometry provides the spatial constraints for execution. Together, they constitute a comprehensive semantic interface between high-level VLM representations and low-level control.

### C. How to Guide: Attention Head Specialization

To capture decoupled task-relevant information, the Multi-Head Attention (MHA) [80] adopted in the action decoder offers a natural solution with minimal structure modification: we explicitly assign specific heads to capture certain factors by **applying different supervision signals on different heads**.

Because large-scale pretrained backbones already exist [45, 8], we can equip them with specified heads while preserving pretrained capabilities, thanks to the natural decoupling characteristics of multi-head attention. Specifically, for those pretrained backbones, we propose a **ControlNet-style** [95] residual adapter strategy, as illustrated in Fig. 3. To add a

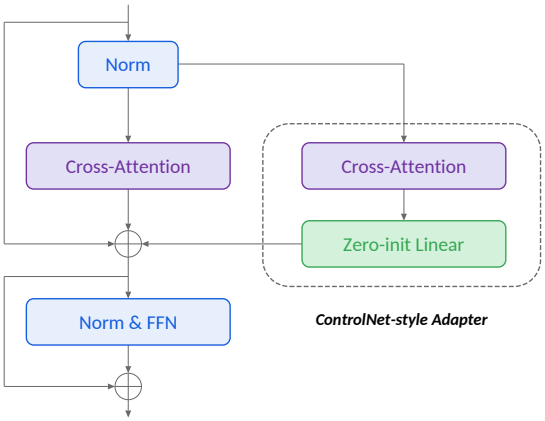



Fig. 3: **ControlNet-style residual adapter for plug-and-play head specialization.** The pretrained main attention branch is kept as the behavior-preserving path, while a factor-specific attention branch is fused through a zero-initialized projection. The adapter copies weights from the base policy and gradually injects task-relevant biases during training.

supervised head $\mathrm{Attn}_{\text{specified}}$, we introduce a zero-initialized projection ZeroConv before fusing with the main branch attention features $\mathrm{Attn}_{\text{main}}$:

$$\mathrm{Attn}(\mathbf{x}) = \mathrm{Attn}_{\text{main}}(\mathbf{x}) + \mathrm{ZeroConv}\left(\mathrm{Attn}_{\text{specified}}(\mathbf{x})\right). \quad (1)$$

Since ZeroConv is initialized to zero, the control branch initially contributes no signal. This ensures the model retains its pre-trained behavior at the start of training, while gradually learning to inject factor-specific biases as optimization proceeds.

We now describe the specific objectives and mechanisms for the three specific factors, as in Alg. 1.

*1) **Object Head (Visual Grounding)**:* Intuitively, action decoding benefits from attending to semantically meaningful regions, such as the object to be grasped and the target destination. To enforce this, we guide a subset of heads $\mathcal{H}_{\text{obj}}$ to concentrate their attention mass on ground-truth object-region masks. Given attention probabilities $\boldsymbol{P}$ from action queries to all keys, we use mean-head aggregation and first average the selected object heads:

$$\bar{P}_{b,t,k} = \frac{1}{|\mathcal{H}_{\text{obj}}|} \sum_{h \in \mathcal{H}_{\text{obj}}} P_{b,h,t,k}. \quad (2)$$

Let $M_{b,k} \in [0,1]$ denote the object-region target on the full key axis, with non-object image patches and non-image tokens assigned zero weight. The object mass for action query $t$ is

$$m_{b,t} = \sum_{k} \bar{P}_{b,t,k} M_{b,k}. \quad (3)$$

We minimize the negative log object mass over samples whose target object is visible:

$$\mathcal{L}_{\text{object}} = -\frac{1}{\sum_b v_b |\mathcal{T}_a|} \sum_b v_b \sum_{t \in \mathcal{T}_a} \log\left(\max(m_{b,t}, \epsilon)\right), \quad (4)$$

where $v_b$ indicates that at least one labeled object patch is available for sample $b$, $\mathcal{T}_a$ is the set of action queries, and $\epsilon$ is a small numerical constant. To construct $M$, we use foundation models like grounding SAM [73] to annotate the object to be grasped or the target destination as interested areas and assign zero weight to all other key positions, while allowing the model to decide how to distribute attention within the interest area. The implementation details are provided in Appendix C, and the stage-aware mask construction is described in Appendix I1.

*2) **Skill Head (Temporal Logic Intent)**:* Skills capture high-level, temporally extended semantics that modulate the model's action behaviors in long-horizon tasks. To encode this, we designate a subset of heads $\mathcal{H}_{\text{skill}}$ to specialize in intent recognition. We pool the selected skill-head output features over guided layers, heads, and action queries:

$$\bar{\mathbf{f}}_b = \frac{1}{|\mathcal{L}_g||\mathcal{H}_{\text{skill}}||\mathcal{T}_a|} \sum_{\ell \in \mathcal{L}_g} \sum_{h \in \mathcal{H}_{\text{skill}}} \sum_{t \in \mathcal{T}_a} \mathbf{f}_{b,\ell,h,t}. \quad (5)$$

We then project the pooled feature to a skill probability distribution and apply a KL-divergence loss against the ground-truth soft label $\mathbf{y}$, which represents the skill distribution over a future horizon:

$$\hat{\mathbf{p}}_b = \mathrm{softmax}(\boldsymbol{W}\bar{\mathbf{f}}_b + \mathbf{b}) \quad (6)$$

$$\mathcal{L}_{\text{skill}} = \frac{1}{B} \sum_{b=1}^{B} \sum_{k} y_{b,k} \left(\log y_{b,k} - \log \hat{p}_{b,k}\right) \quad (7)$$

Regarding the annotation of skill types at each time-step, we combine foundation models and manual correction. For LIBERO, the skill distribution covers three effective task-level skill classes plus one null/background class for unannotated or transition frames. The construction and implementation details of the skill-label are provided in Appendix I2 and Appendix C.

*3) **Depth Head (3D Structure)**:* Since standard vision encoders (e.g., SigLIP) in VLA [8] are trained with 2D supervision and lack explicit 3D awareness, we design specialized depth heads. Instead of a loss, we use a structural constraint: we extract features from a frozen depth encoder (e.g., DA3 [57]) on the primary camera view, $F_{\text{Depth}}$, and project them into depth-aware keys and values, $K_{\text{Depth}}$ and $V_{\text{Depth}}$. The query still comes from the action decoder; we constrain the action-query heads in $\mathcal{H}_{\text{depth}}$ to attend only to these depth-derived keys and values:

$$\mathcal{H}_{\text{depth}} : \mathrm{softmax}\left(\frac{Q_{\text{act}}[\mathcal{H}_{\text{depth}}](K_{\text{Depth}})^{\top}}{\sqrt{d_h}}\right) V_{\text{Depth}}. \quad (8)$$

This design forces specific heads to specialize in 3D geometry processing. The details of the implementation and the design ablation experiments are provided in Appendix C and Appendix E2.

In summary, we adopt a mixed loss:

$$\mathcal{L} = \mathcal{L}_{\text{FM}} + \lambda_{\text{object}}\mathcal{L}_{\text{object}} + \lambda_{\text{skill}}\mathcal{L}_{\text{skill}}. \quad (9)$$

where $\mathcal{L}_{\text{FM}}$ is the flow matching loss, and $\lambda_{\text{object}}$ and $\lambda_{\text{skill}}$ are the coefficients for the auxiliary objectives that supervise distinct subsets of attention heads. For geometric perception, we inject depth keys and values for depth heads rather than using a loss term. The remaining unsupervised heads are left free to capture purely data-driven patterns, preserving the model's flexibility and expressivity, as in Fig. 2.

### *D. Guidance Dataset Construction*

To reduce the annotation burden of factor guidance, we build a highly automatic factor annotation pipeline, as shown in Fig. 4. For object grounding, Qwen3-VL [3] first identifies the

TABLE I: **LIBERO-Plus Benchmark Results.** The proposed model achieves the highest average success rate, with a significant boost compared to its base model $\pi_0$. Notably, **single-head ablations reveal task-specific alignment**: the object head is strongest among single-head variants on the *Object* and *Long* suites, the skill head gives the best single-head result on the *Goal* suite, and the depth head performs best on the *Spatial* suite.

| | Perturbation Dimensions | | | | | | | Task Suites | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Model** | **Camera** | **Robot** | **Language** | **Light** | **Background** | **Noise** | **Layout** | **Spatial** | **Object** | **Goal** | **Long** | **Total** |
| OpenVLA [45] | 0.8 | 3.5 | 23.0 | 8.1 | 34.8 | 15.2 | 28.5 | 19.4 | 14.0 | 15.1 | 14.3 | 15.6 |
| OpenVLA-OFT [44] | 56.4 | 31.9 | 79.5 | 88.7 | 93.3 | 75.8 | 74.2 | 84.0 | 66.5 | 63.0 | 66.4 | 69.6 |
| NORA [38] | 2.2 | 37.0 | 65.1 | 45.7 | 58.6 | 12.8 | 62.1 | 47.6 | 34.4 | 38.8 | 36.3 | 39.0 |
| WorldVLA [12] | 0.1 | 27.9 | 41.6 | 43.7 | 17.1 | 10.9 | 38.0 | 32.5 | 28.6 | 31.8 | 8.2 | 25.0 |
| UniVLA [11] | 1.8 | 46.2 | 69.6 | 69.0 | 81.0 | 21.2 | 31.9 | 55.5 | 36.7 | 40.7 | 39.9 | 43.9 |
| $\pi_0$-Fast [69] | 65.1 | 21.6 | 61.0 | 73.2 | 73.2 | 74.4 | 68.8 | 74.4 | 72.7 | 57.5 | 43.4 | 61.6 |
| RIPT-VLA [77] | 55.2 | 31.2 | 77.6 | 88.4 | 91.6 | 73.5 | 74.2 | 85.8 | 64.3 | 58.0 | 67.5 | 68.4 |
| DreamVLA [96] | 65.0 | 40.9 | 63.5 | 85.7 | 82.7 | 85.0 | 74.0 | 79.7 | 79.0 | 61.7 | 59.8 | 69.9 |
| AdaMoE [75] | 53.8 | 17.5 | 20.6 | 73.7 | 73.8 | 58.6 | 65.8 | 51.0 | 57.9 | 53.3 | 38.1 | 50.1 |
| Spatial Forcing [47] | 20.1 | 13.4 | 40.9 | 29.1 | 33.4 | 25.7 | 39.3 | 52.9 | 31.0 | 28.2 | 5.4 | 29.1 |
| VLA-Adapter [84] | 36.2 | 37.9 | 74.6 | 70.6 | 76.1 | 58.0 | 69.7 | 85.0 | 46.3 | 56.0 | 50.4 | 59.1 |
| $\pi_0$ [8] | 62.3 | 39.8 | 63.1 | 86.0 | 82.8 | 82.4 | 69.6 | 77.7 | 74.1 | 61.4 | 60.1 | 68.2 |
| w/ object head | 71.7 | 45.8 | 63.5 | 92.4 | 86.9 | 85.1 | 77.4 | 80.6 | **82.5** | 67.1 | 64.0 | 73.4 |
| w/ skill head | 70.0 | 45.0 | 61.7 | 90.2 | 83.0 | **88.4** | 76.3 | 79.8 | 78.9 | 68.9 | 62.7 | 72.5 |
| w/ depth head | 68.1 | 43.9 | **65.8** | 90.7 | 83.4 | 85.6 | 72.8 | 81.4 | 79.0 | 65.4 | 61.8 | 71.7 |
| w/ all heads **(Ours)** | **73.7** | **51.4** | 62.6 | **94.6** | **89.0** | 85.2 | **79.9** | **84.0** | 80.9 | **70.8** | **66.2** | **75.4** |

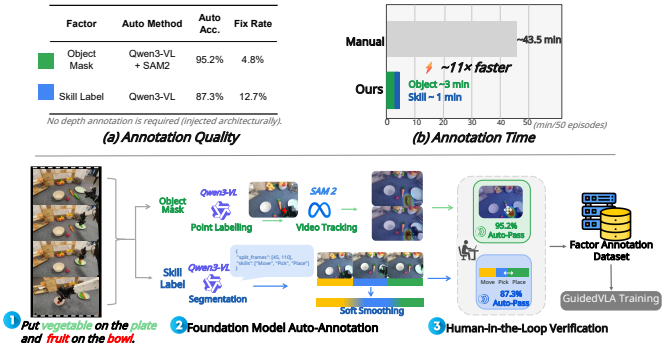



Fig. 4: **Automatic factor annotation pipeline.** Object masks are initialized by Qwen3-VL point prompts and propagated by SAM2, skill labels are generated by Qwen3-VL from stage descriptions and a predefined skill list, and depth guidance uses frozen depth features without requiring depth labels. The pipeline substantially reduces human annotation time while preserving a human verification step for supervision quality.

task-relevant object from the stage description and proposes foreground point prompts; SAM2 [72] then propagates the corresponding masks through the video segment, followed by human verification. For skill recognition, Qwen3-VL assigns stage-level skill labels from a predefined skill list, which are then converted into soft targets in Eq. 7. Depth guidance does not require manual depth annotation because the depth head directly consumes features from a frozen pretrained depth encoder. In our annotation method, 92% of the episodes require no human correction; annotating 50 episodes takes about 4 minutes with our pipeline, compared to around 43.5 minutes under manual annotation. The implementation details are in Appendix I.

## IV. Experiments

### A. Simulation Experiments

**LIBERO-Plus** [28] is a robustness-oriented benchmark built upon LIBERO [59]. It is designed to **evaluate generalist manipulation policies under distribution shifts**. It introduces perturbations along seven dimensions: camera viewpoint, robot initial state, language variation, lighting condition, background texture, sensor noise, and object layout to expose failure modes under generalization scenario beyond in-domain evaluation. We compare with state-of-the-art baselines in Table I.

**RoboTwin 2.0** [16] offers a multi-task evaluation platform across diverse robot embodiments, and leverages extensive scene/object randomization to scale data and enable out-of-distribution testing. As in Fig. 5, we evaluate on eight representative tasks **under randomized, unseen settings (out-of-domain task instructions, environments, and object placements)** using the Agilex Piper dual-arm setup.

### B. Real-World Experiments

We conduct real-world experiments on two dual-arm platforms to evaluate both in-domain action generation and cross-platform generalization against baselines.

**Platforms**: Platform A is an ALOHA AgileX dual-arm system, equipped with two Intel Orbbec Dabai wrist cameras (one per arm) and an additional Intel Orbbec Dabai third-person camera. Platform B is a PSI-Bot dual-arm platform, using Intel RealSense D435 cameras for visual observations. Figure 6 summarizes the hardware setups and qualitative task rollouts.

**Tasks**: On ALOHA AgileX, we design three household tasks: (1) *pick up fruits and vegetables*: classify and place pepper/carrot on plate, strawberry in bowl, (2) *stack the bowls*: assemble

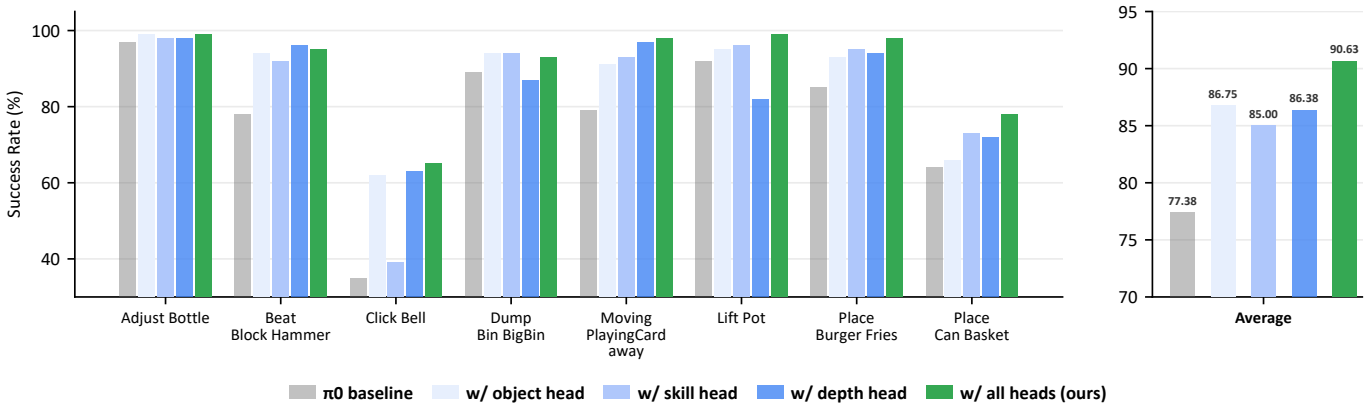



Fig. 5: **RoboTwin 2.0 Benchmark Performance.** Success rates across 8 manipulation tasks comparing the $\pi_0$ baseline, single-head experts, and our full model. While specific heads excel at aligned tasks (e.g., depth head for geometry-heavy Beat Hammer Block), the full model (purple) integrates these capabilities to achieve the best overall average performance (90.63%).

two bowls and place on rack, and (3) *clean the tabletop*: sweep trash with broom/dustpan, pour into tray. On PSI-Bot RealMan, we design three chemistry-lab manipulation tasks: (4) *place beaker in heating mantle*: grasp a beaker and insert it into the heating mantle, (5) *stack beakers*: nest small beakers inside a large one, and (6) *heat beaker*: place an asbestos mesh on the iron stand and then place the beaker on the mesh. These laboratory tasks focus on the manipulation challenges posed by transparent, rigid objects and tight geometric constraints. They do not evaluate complete safety-critical chemical procedures; in particular, *heat beaker* requires placing the beaker onto the heating setup rather than controlling the heating process itself. Detailed success criteria are provided in Appendix L2.

**Evaluation protocol**: For each task and model, we perform 20 trials. A trial is successful if the entire task is completed.

**Generalization evaluation**: We evaluate three generalization settings on both platforms: *in-domain generalization*, *scene generalization*, and *lighting generalization*. Here, *in-domain generalization* focuses on object position variations within the training distribution, while preserving task semantics and scene layout. *Scene generalization* introduces distracting objects into the workspace, testing robustness to clutter and semantic interference. *Lighting generalization* varies illumination intensity and color temperature, assessing sensitivity to perceptual shifts. Results are summarized in Table II, with detailed setting definitions in Appendix M.

## V. Analysis

In this section, we aim to answer the following questions:

1) Do VLAs under-utilize vision-language representations in action decoding process, and can explicit factor guidance close this gap? (Section V-B)
2) Does our proposed GuidedVLA improve baseline performance under both in-distribution and out-of-distribution evaluations? (Section V-A)
3) Which factors (object, skill, geometry) matter most for which task types? (Section V-A)
4) Does our attention head specialization indeed lead to learning decoupled features? (Section V-C)
5) How different architectural choices for guidance influence performance? (Section V-E)

### A. Task-suite Analysis and Cross-benchmark Generalization

We analyze how each factor contributes to different task suites and use representative results from simulation and real-world evaluations to explain *why* each specified head helps.

**Object Head: Visual Generalization.** Tasks involving clutter or distractors necessitate a precise understanding of object instance identities. On the LIBERO-Plus *Object* suite, which stresses object-level distinctions, the object head yields the strongest single-head result (82.5%, +8.4% over $\pi_0$, Table I; full results in Appendix B). This aligns with the intuition that explicit object-centric representations mitigate grounding failures, allowing the policy to filter out irrelevant visual cues that confuse the baseline.

**Skill Head: Temporal Coherence.** Long-horizon manipulation requires maintaining "stage awareness" to transition correctly between sub-skills. On LIBERO-Plus, the skill head gives the best single-head result on the *Goal* suite (68.9%) and remains above $\pi_0$ on the *Long* suite (62.7% vs. 60.1%, Table I). Similarly, for the *Lift Pot* task (RoboTwin 2.0) involving a strict sequence of grasping, stabilizing, and lifting, this head achieves the best single-head success rate (96%, Fig. 5; full results in Appendix B). These results validate that explicit skill recognition provides the temporal scaffolding needed to prevent premature termination or mode collapse during phase transitions.

**Depth Head: Geometric Precision.** Tasks reliant on precise 3D localization, such as pressing or insertion, necessitate accurate depth estimation which 2D backbone alone often fails to provide. On RoboTwin 2.0, the *Click Bell* task requires precise Z-axis control to trigger the mechanism without collision; the depth head drastically improves performance from 35% to 63% (Fig. 5). We observe a similar trend in *Beat Hammer*

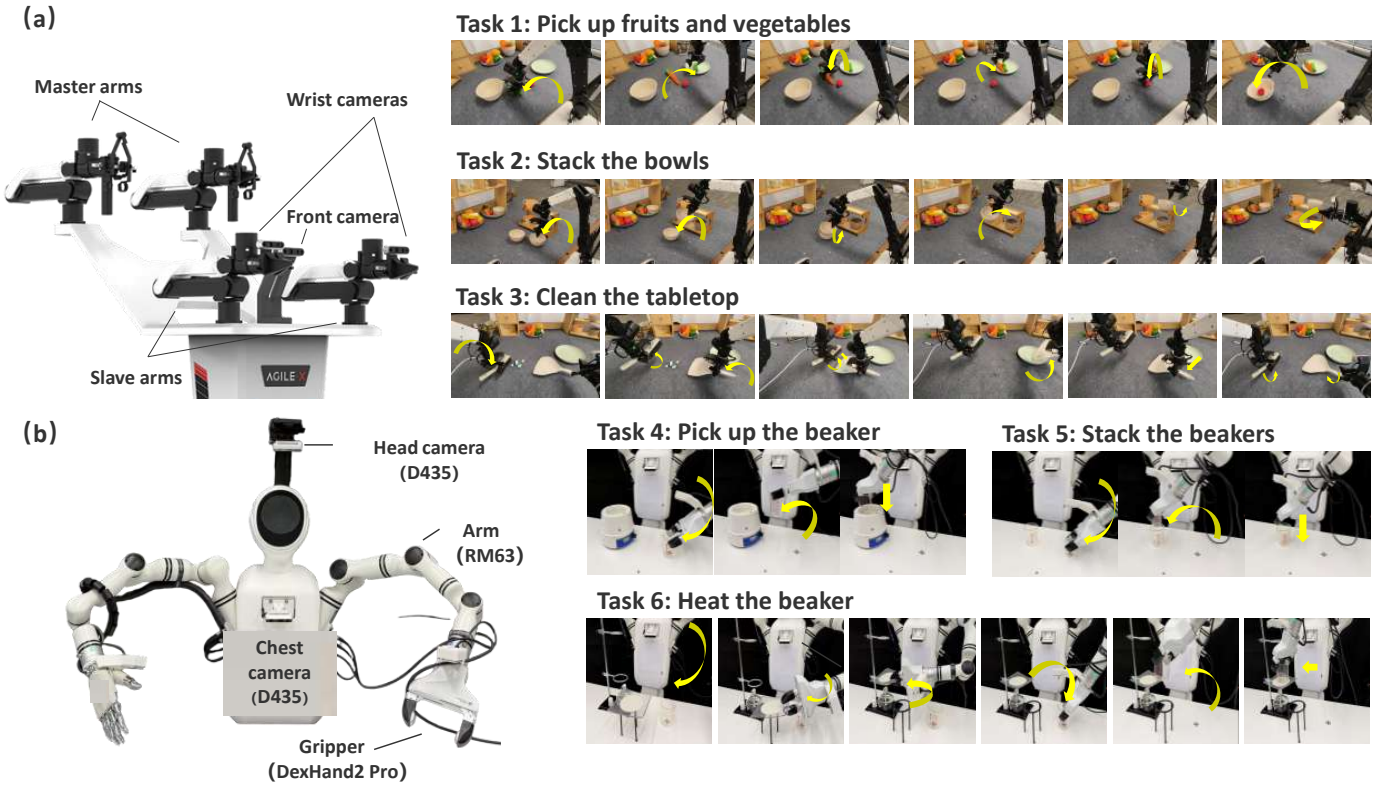



Fig. 6: **Real-world Robot Platforms and Evaluation Tasks.** (a) ALOHA AgileX dual-arm mobile manipulator with left/right wrist Orbbec Dabai cameras and a third-person Orbbec Dabai camera; we evaluate three household tasks: pick up fruits and vegetables, stack the bowls, clean the tabletop. (b) PSI-Bot equipped with RealMan RM63 arm(s) and DexHand2 Pro hands, with head/chest RealSense D435 cameras; we evaluate three lab tasks: pick up beaker, stack beakers, and heat beaker.

TABLE II: **Cross-Platform Real-World Generalization.** Success rates ($N = 20$) across three generalization settings on ALOHA and PSI-Bot platforms. Our method consistently outperforms the baseline, achieving performance gains across all settings (up to 52.7%) and demonstrating robustness under challenging out-of-domain conditions. Task 1–6 correspond to: (1) pick up fruits and vegetables, (2) stack the bowls, (3) clean the tabletop, (4) place beaker in heating mantle, (5) stack beakers, and (6) heat beaker. In-domain generalization includes variations in object positions within the training distribution.

| **Generalization** | **Method** | **ALOHA AgileX** | | | **PSI-Bot RealMan** | | | **Average** |
|---|---|---|---|---|---|---|---|---|
| **Setting** | | Task 1 | Task 2 | Task 3 | Task 4 | Task 5 | Task 6 | (%) |
| In-Domain | Base Policy | 10/20 | 11/20 | 9/20 | 12/20 | 12/20 | 13/20 | 55.8 |
| | **Ours** | **14/20** | **15/20** | **14/20** | **16/20** | **17/20** | **15/20** | **75.8** |
| Scene | Base Policy | 7/20 | 8/20 | 6/20 | 12/20 | 11/20 | 9/20 | 44.2 |
| | **Ours** | **13/20** | **12/20** | **11/20** | **15/20** | **16/20** | **14/20** | **67.5** |
| Lighting | Base Policy | 11/20 | 9/20 | 10/20 | 14/20 | 12/20 | 13/20 | 57.5 |
| | **Ours** | **13/20** | **16/20** | **15/20** | **17/20** | **18/20** | **16/20** | **79.2** |

*Block* (78% $\rightarrow$ 96%), where height alignment is critical. These gains confirm that explicit geometric cues compensate for the lack of 3D observability in standard VLA inputs.

**Full Model: Synergetic Generalization.** The full model integrates these complementary strengths—visual grounding, temporal coherence, and geometric precision—to achieve robust generalization across diverse domains. It raises the average success on RoboTwin 2.0 from 77.38% to 90.63% (Fig. 5) and demonstrates superior robustness in the real world (Table II; head-wise real-robot diagnostics in Appendix O2). Crucially, while single heads excel in their respective niches, the full model is the only variant that reliably generalizes across *all* dimensions of variability (scene, lighting, and position), highlighting that these factors are not redundant but mutually reinforcing.

### *B. Sensitivity Analysis: Does Factor Quality Matter?*

We move from a binary "with/without" comparison to a quantitative question: does better factor quality lead to higher success? We vary each factor's strength in controlled ablations

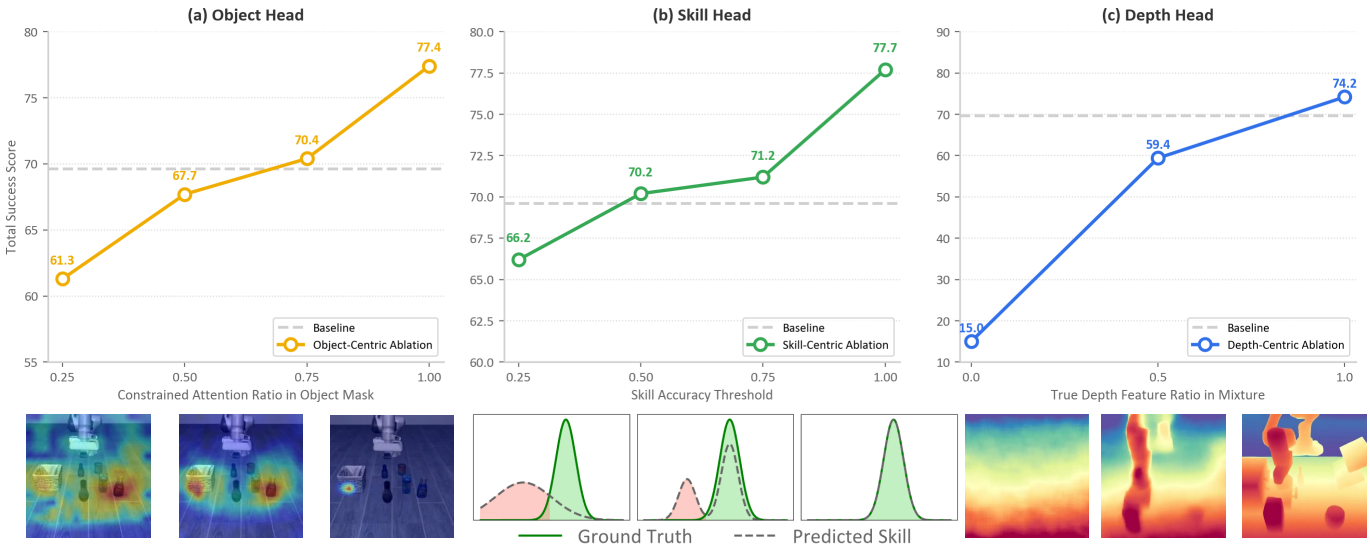



Fig. 7: **Higher Factor Quality Leads to Better Task Performance. Top:** Quantitative analysis on the LIBERO-Plus layout perturbation track shows that improving the quality of each specialized head consistently boosts success rates. **(a) Object Head:** as the proportion of attention focused on task-relevant object regions increases, success rises from 61.3% to 74.6%, highlighting the importance of precise object-centric attention. **(b) Skill Head:** higher skill-recognition accuracy, measured by a linear probe, correlates with improved performance (66.2% to 72.9%), indicating that better temporal understanding enhances control. **(c) Depth Head:** increasing the ratio of true depth features (versus noise) dramatically improves both qualitative depth estimation and quantitative success (15.6% to 76.7%), confirming that explicit 3D cues are critical for robust manipulation. **Bottom:** Qualitative visualizations show how changes along the x-axis metrics are reflected in the corresponding feature representations.

and measure continuous proxies aligned with the intended semantics, as summarized in Figure 7.

**Object Grounding.** We measure the fraction of attention mass falling inside the object/gripper mask, using the same supervision target as Eq. 4. As the mask-aligned attention ratio increases from 0.25 to 1.0, success rises from 61.3% to 74.6% (Figure 7 (a)). In contrast, $\pi_0$ exhibits low intrinsic object focus (26.5%), indicating that stronger spatial grounding directly correlates with performance. Under a stricter localization diagnostic, GuidedVLA increases object-region attention mass from 8.1% to 84.0% and argmax-hit accuracy from 2.2% to 84.7% over $\pi_0$.

**Skill Recognition.** We use a linear probe to predict task skill labels from action features, using probe accuracy as the quality metric. Raising the skill accuracy threshold from 0.25 to 1.0 increases success from 66.2% to 72.9% (Figure 7 (b)). The baseline $\pi_0$ remains low at 48.4%, confirming improved temporal representations translate into better performance.

**Geometry Perception.** We modulate the true depth feature ratio in the geometry stream as a proxy for geometric signal strength. Increasing this ratio from 0 to 1.0 yields a large success gain (15.6% $\rightarrow$ 76.7%, Figure 7 (c)), demonstrating that richer geometric cues substantially improve task outcomes.

Figure 7 summarizes these trends, showing that success increases monotonically with each factor's quality, not merely its presence. Figure 8 provides a qualitative analysis of factor features over time in a task. Details of these metrics and ablations are provided in Appendix D.

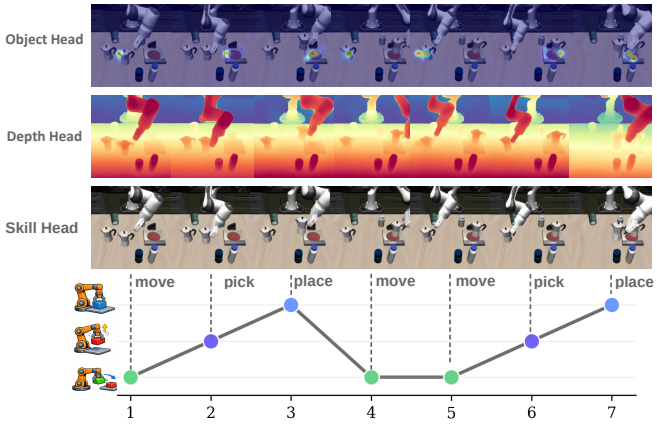



Fig. 8: **Visualization of Learned Representations in GuidedVLA.** From top to bottom: **(i)** Object attention focuses on the manipulation target (e.g., pot handle); **(ii)** Depth features encode explicit 3D structure; **(iii)** Skill predictions track the temporal progress of task phases. This confirms that each head specializes in its designated semantic factor as intended.

### *C. Specialization Enables Decoupled Feature Learning*

We have shown that each factor (object grounding, geometry, and skill) correlates with task success. A natural next question is: *can we guide multiple factors by training all heads with all factor objectives?* Our answer is **NO**: a naive mixed training protocol consistently underperforms (Figure 9). The gain is not just extra supervision or capacity: under matched control settings, GuidedVLA consistently outperforms the shared-head mixture alternative and other

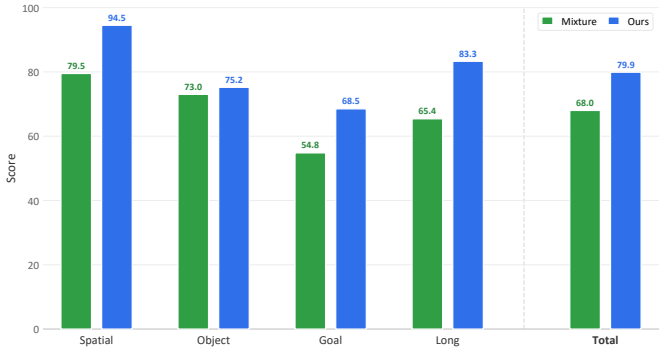
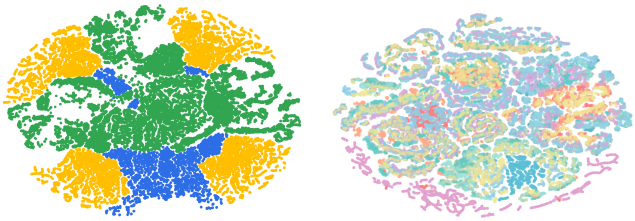



Fig. 9: **Comparison of GuidedVLA against Mixture Alternative.** Attention head specialization explicitly outperforms learning all objectives in a mixture.

non-factorized controls; additional architecture ablations are provided in Appendix F.

When object grounding, geometry, and skill objectives are all supervised through all attention heads, their features become entangled, as in Fig. 10. This coupling means that information from different factors is mixed, making it difficult to capture each factor clearly, thus leading to degraded performance.



(a) Ours (b) Mixture

Fig. 10: **t-SNE Visualization of Attention Outputs.** (a) Specialized attention heads (object: yellow, depth: blue, skill: green) form well-separated clusters, demonstrating factor disentanglement and minimal interference. (b) The mixture alternative shows overlapping clusters (different colors representing different heads), indicating entangled representations.

### *D. Comparison to Other Factor Guidance Approaches*

There exist two representative paradigms for providing task-relevant factors to VLA: DreamVLA [96] lets a VLM predict dynamic regions, depth maps, and semantic knowledge with extra queries. AdaMoE [75] applies a mixture of experts to automatically learn experts for different tasks.

As shown in Table I, our method outperforms DreamVLA (69.9%), VLA-Adapter (59.1%), AdaMoE (50.1%), and Spatial Forcing (29.1%) in overall average success rate (75.4%), with strong gains across most perturbation dimensions and task suites, especially in challenging settings such as camera, robot, and layout perturbations. We attribute these gains to our explicit attention head specialization, which enables the model to disentangle and robustly capture object grounding, skill recognition, and geometric cues.

### *E. Ablation of Design Choices*

In this section, we discuss the design choices for each specified head and the plug-and-play ControlNet-style residual adapter. We systematically conduct experiments on these choices in RoboTwin 2.0, with detailed results in Appendix E.
**Object Head**. Suppressing attention outside object regions outperforms enforcing a fixed spatial prior (e.g., Gaussian) inside object regions. This allows the model to flexibly learn which object parts are most relevant at each task stage.
**Skill Head**. Soft targets outperform one-hot labels, better handling ambiguous or mixed-intent segments and thus leading to more stable training and better performance.
**Depth Head**. Adopting a lightweight downsampling adapter outperforms directly using original, non-downsampled depth features, since the number of depth tokens can be large and make the learning process difficult.
**Extra Branch**. A zero-initialized control branch introduces auxiliary signals gradually and does not disrupt the base model at the start of training, outperforming fusion without this branch.

## VI. Conclusion

We present GuidedVLA, a method that makes the VLA action decoding process more robust by explicitly specifying task-relevant factors through attention head specialization. By assigning dedicated heads to object grounding, temporal skill logic, and geometric cues, GuidedVLA transforms the action decoder from an entangled black box into a set of semantically decoupled pathways. Across simulation benchmarks and real-robot evaluations, this design delivers consistent gains in both in-domain performance and robustness under distribution shift. Further analyses show that (i) higher-quality factor signals correlate with higher task success, and (ii) allocating a dedicated head per factor produces clearly decoupled features. Together, these results point toward training VLAs that are both more interpretable and more generalizable.

**Limitations and Future Work.** Our method relies on predefined factors, and automating factor discovery remains an open challenge, especially for continuous tasks where automatic skill labeling is difficult. Promising directions include automatic skill discovery [103, 82] and the use of continuous progress signals as latent skill targets [15].

## VII. Acknowledgment

This work is supported by the National Natural Science Foundation of China (Grant No. 62521004), the Science and Technology Commission of Shanghai Municipality (No. 24511103100) and the New Cornerstone Science Foundation through the XPLORER PRIZE. This work is also in part supported by Scientific Research Innovation Capability Support Project for Young Faculty (U40) of the Ministry of Education of China, SRICSPYF-ZY2025019.

## References

[1] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn,

Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

[2] Michael Ahn, Anthony Brohan, Noah Brown, Yevgen Chebotar, Omar Cortes, Byron David, Chelsea Finn, Chuyuan Fu, Keerthana Gopalakrishnan, Karol Hausman, et al. Do as i can, not as i say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

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

[4] Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*, 2024.

[5] Vineet Bhat, Yu-Hsiang Lan, Prashanth Krishnamurthy, Ramesh Karri, and Farshad Khorrami. 3d cavla: Leveraging depth and 3d context to generalize vision language action models for unseen tasks. *arXiv preprint arXiv:2505.05800*, 2025.

[6] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

[7] Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y Galliker, et al. $\pi_{0.5}$: A vision-language-action model with open-world generalization. In *9th Annual Conference on Robot Learning*, 2025.

[8] Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, et al. $\pi_0$: A vision-language-action flow model for general robot control. In *RSS*, 2025.

[9] Kevin Black, Manuel Y Galliker, and Sergey Levine. Real-time execution of action chunking flow policies. *arXiv preprint arXiv:2506.07339*, 2025.

[10] Anthony Brohan, Noah Brown, Justice Carbajal, Yevgen Chebotar, Joseph Dabis, Chelsea Finn, Keerthana Gopalakrishnan, Karol Hausman, Alex Herzog, Jasmine Hsu, et al. Rt-1: Robotics transformer for real-world control at scale. *arXiv preprint arXiv:2212.06817*, 2022.

[11] Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. Univla: Learning to act anywhere with task-centric latent actions. *arXiv preprint arXiv:2505.06111*, 2025.

[12] Jun Cen, Chaohui Yu, Hangjie Yuan, Yuming Jiang, Siteng Huang, Jiayan Guo, Xin Li, Yibing Song, Hao Luo, Fan Wang, Deli Zhao, and Hao Chen. Worldvla: Towards autoregressive action world model. *arXiv preprint arXiv:2506.21539*, 2025.

[13] Alexandre Chapin, Emmanuel Dellandréa, and Liming Chen. Storm: Slot-based task-aware object-centric representation for robotic manipulation. *arXiv preprint arXiv:2601.20381*, 2026.

[14] Jiayi Chen, Wenxuan Song, Pengxiang Ding, Ziyang Zhou, Han Zhao, Feilong Tang, Donglin Wang, and Haoang Li. Unified diffusion vla: Vision-language-action model via joint discrete denoising diffusion process. *arXiv preprint arXiv:2511.01718*, 2025.

[15] Shirui Chen, Cole Harrison, Ying-Chun Lee, Angela Jin Yang, Zhongzheng Ren, Lillian J. Ratliff, Jiafei Duan, Dieter Fox, and Ranjay Krishna. Topreward: Token probabilities as hidden zero-shot rewards for robotics. *arXiv preprint arXiv:2602.19313*, 2026.

[16] Tianxing Chen, Zanxin Chen, Baijun Chen, Zijian Cai, Yibin Liu, Zixuan Li, Qiwei Liang, Xianliang Lin, Yiheng Ge, Zhenyu Gu, Weiliang Deng, Yubin Guo, Tian Nian, Xuanbing Xie, Qiangyu Chen, Kailun Su, Tianling Xu, Guodong Liu, Mengkang Hu, Huan ang Gao, Kaixuan Wang, Zhixuan Liang, Yusen Qin, Xiaokang Yang, Ping Luo, and Yao Mu. Robotwin 2.0: A scalable data generator and benchmark with strong domain randomization for robust bimanual robotic manipulation. *arXiv preprint arXiv:2506.18088*, 2025. URL https://arxiv.org/abs/2506.18088.

[17] Baiye Cheng, Tianhai Liang, Suning Huang, Maanping Shao, Feihong Zhang, Botian Xu, Zhengrong Xue, and Huazhe Xu. Moe-dp: An moe-enhanced diffusion policy for robust long-horizon robotic manipulation with skill decomposition and failure recovery. *arXiv preprint arXiv:2511.05007*, 2025.

[18] Cheng Chi, Zhenjia Xu, Siyuan Feng, Eric Cousineau, Yilun Du, Benjamin Burchfiel, Russ Tedrake, and Shuran Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, 44(10-11):1684–1704, 2025.

[19] Yinpei Dai, Jayjun Lee, Yichi Zhang, Ziqiao Ma, Jed Yang, Amir Zadeh, Chuan Li, Nima Fazeli, and Joyce Chai. Aimbot: A simple auxiliary visual cue to enhance spatial awareness of visuomotor policies. *arXiv preprint*

*arXiv:2508.08113*, 2025.

[20] Sudeep Dasari, Frederik Ebert, Stephen Tian, Suraj Nair, Bernadette Bucher, Karl Schmeckpeper, Siddharth Singh, Sergey Levine, and Chelsea Finn. Robonet: Large-scale multi-robot learning. *arXiv preprint arXiv:1910.11215*, 2019.

[21] Pim De Haan, Dinesh Jayaraman, and Sergey Levine. Causal confusion in imitation learning. *Advances in neural information processing systems*, 32, 2019.

[22] Shengliang Deng, Mi Yan, Yixin Zheng, Jiayi Su, Wenhao Zhang, Xiaoguang Zhao, Heming Cui, Zhizheng Zhang, and He Wang. Stereovla: Enhancing vision-language-action models with stereo vision. *arXiv preprint arXiv:2512.21970*, 2025.

[23] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. In *International Conference on Machine Learning*, pages 8469–8488. PMLR, 2023.

[24] Frederik Ebert et al. Bridgedata v2: A dataset for robot learning at scale. *arXiv preprint arXiv:2308.12952*, 2023.

[25] Cunxin Fan, Xiaosong Jia, Yihang Sun, Yixiao Wang, Jianglan Wei, Ziyang Gong, Xiangyu Zhao, Masayoshi Tomizuka, Xue Yang, Junchi Yan, et al. Interleave-vla: Enhancing robot manipulation with interleaved image-text instructions. In *ICLR*, 2026.

[26] Qingyu Fan, Zhaoxiang Li, Yi Lu, Wang Chen, Qiu Shen, Xiao-xiao Long, Yinghao Cai, Tao Lu, Shuo Wang, and Xun Cao. Peafowl: Perception-enhanced multi-view vision-language-action for bimanual manipulation. *arXiv preprint arXiv:2601.17885*, 2026.

[27] Hung-Chieh Fang, Kuo-Han Hung, Chu-Rong Chen, Po-Jung Chou, Chun-Kai Yang, Po-Chen Ko, Yu-Chiang Wang, Yueh-Hua Wu, Min-Hung Chen, and Shao-Hua Sun. Learning skills from action-free videos. *arXiv preprint arXiv:2512.20052*, 2025.

[28] Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, et al. Libero-plus: In-depth robustness analysis of vision-language-action models. *arXiv preprint arXiv:2510.13626*, 2025.

[29] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In *International conference on learning representations*, 2018.

[30] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2 (11):665–673, 2020.

[31] Dibya Ghosh, Homer Rich Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, et al. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems*, 2024.

[32] Siddhant Haldar and Lerrel Pinto. Point policy: Unifying observations and actions with key points for robot manipulation. *arXiv preprint arXiv:2502.20391*, 2025.

[33] Cheng-Chun Hsu, Bowen Wen, Jie Xu, Yashraj Narang, Xiaolong Wang, Yuke Zhu, Joydeep Biswas, and Stan Birchfield. Spot: Se(3) pose trajectory diffusion for object-centric manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4853–4860, 2025. doi: 10.1109/ICRA55743.2025.11127562. arXiv:2411.00965.

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

[35] Aoshen Huang, Jiaming Chen, Jiyu Cheng, Ran Song, Wei Pan, and Wei Zhang. Skill-aware diffusion for generalizable robotic manipulation. *arXiv preprint arXiv:2601.11266*, 2026.

[36] Wenlong Huang, Chen Wang, Yunzhu Li, Ruohan Zhang, and Li Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. In *Conference on Robot Learning*, pages 4573–4602. PMLR, 2025.

[37] Wenlong Huang, Yu-Wei Chao, Arsalan Mousavian, Ming-Yu Liu, Dieter Fox, Kaichun Mo, and Li Fei-Fei. Pointworld: Scaling 3d world models for in-the-wild robotic manipulation. *arXiv preprint arXiv:2601.03782*, 2026.

[38] Chia-Yu Hung, Qi Sun, Pengfei Hong, Amir Zadeh, Chuan Li, U Tan, Navonil Majumder, Soujanya Poria, et al. Nora: A small open-sourced generalist vision language action model for embodied tasks. *arXiv preprint arXiv:2504.19854*, 2025.

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

[40] Tao Jiang, Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Jianning Cui, Xiao Liu, Shuiqi Cheng, Jiyang Gao, Huazhe Xu, and Hang Zhao. Galaxea open-world dataset and g0 dual-system vla model. *arXiv preprint arXiv:2509.00576*, 2025.

[41] Yuhua Jiang, Shuang Cheng, Yan Ding, Feifei Gao, and Biqing Qi. Asyncvla: Asynchronous flow matching for vision-language-action models. *arXiv preprint arXiv:2511.14148*, 2025.

[42] Yunfan Jiang, Agrim Gupta, Zichen Zhang, Guanzhi Wang, Yongqiang Dou, Yanjun Chen, Li Fei-Fei, Anima Anandkumar, Yuke Zhu, and Linxi Fan. Vima: General robot manipulation with multimodal prompts. *arXiv preprint arXiv:2210.03094*, 2(3):6, 2022.

[43] Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karam-

cheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, et al. Droid: A large-scale in-the-wild robot manipulation dataset. *arXiv preprint arXiv:2403.12945*, 2024.

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

[45] Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning*, pages 2679–2713. PMLR, 2025.

[46] Seungjae Lee, Yoonkyo Jung, Inkook Chun, Yao-Chih Lee, Zikui Cai, Hongjia Huang, Aayush Talreja, Tan Dat Dao, Yongyuan Liang, Jia-Bin Huang, et al. Tracegen: World modeling in 3d trace space enables learning from cross-embodiment videos. *arXiv preprint arXiv:2511.21690*, 2025.

[47] Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=euMVC1DO4k.

[48] Guangrun Li, Yaoxu Lyu, Zhuoyang Liu, Chengkai Hou, Jieyu Zhang, and Shanghang Zhang. H2r: A human-to-robot data augmentation for robot pre-training from videos. *arXiv preprint arXiv:2505.11920*, 2025.

[49] Hang Li, Qian Feng, Zhi Zheng, Jianxiang Feng, Zhaopeng Chen, and Alois Knoll. Language-guided object-centric diffusion policy for generalizable and collision-aware manipulation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12834–12841. IEEE, 2025.

[50] Jinming Li, Yichen Zhu, Zhibin Tang, Junjie Wen, Minjie Zhu, Xiaoyu Liu, Chengmeng Li, Ran Cheng, Yaxin Peng, Yan Peng, et al. Coa-vla: Improving vision-language-action models via visual-text chain-of-affordance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9759–9769, 2025.

[51] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[52] Peiyan Li, Yixiang Chen, Hongtao Wu, Xiao Ma, Xiangnan Wu, Yan Huang, Liang Wang, Tao Kong, and Tieniu Tan. Bridgevla: Input-output alignment for efficient 3d manipulation learning with vision-language models. *arXiv preprint arXiv:2506.07961*, 2025.

[53] Ziwen Li, Xin Wang, Hanlue Zhang, Runnan Chen, Runqi Lin, Xiao He, Han Huang, Yandong Guo, Fakhri Karray, Tongliang Liu, et al. Posa-vla: Enhancing action generation via pose-conditioned anchor attention. *arXiv preprint arXiv:2512.03724*, 2025.

[54] Yixing Liang, Anna Xie, Ziyun Feng, Yuke Zhu, Song-Chun Zhu, and Yunzhu Li. Skilldiffuser: Interpretable skill planning for latent diffusion-based manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16467–16476, 2024.

[55] Zhixuan Liang, Yizhuo Li, Tianshuo Yang, Chengyue Wu, Sitong Mao, Tian Nian, Liuao Pei, Shunbo Zhou, Xiaokang Yang, Jiangmiao Pang, et al. Discrete diffusion vla: Bringing discrete diffusion to action decoding in vision-language-action policies. *arXiv preprint arXiv:2508.20072*, 2025.

[56] Fanqi Lin, Haojie Lu, Haojian Fang, and Ping Luo. Manicm: Real-time 3d diffusion policy via consistency model for robotic manipulation. *arXiv preprint arXiv:2406.01586*, 2024.

[57] Haotong Lin, Sili Chen, Jun Hao Liew, Donny Y. Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

[58] Kevin Lin, Varun Ragunath, Andrew McAlinden, Aaditya Prasad, Jimmy Wu, Yuke Zhu, and Jeannette Bohg. Constraint-preserving data generation for one-shot visuomotor policy generalization. In *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pages 3631–3646. PMLR, 2025. URL https://proceedings.mlr.press/v305/lin25b.html.

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

[60] Songming Liu, Lingxuan Wu, Bangguo Li, Hengkai Tan, Huayu Chen, Zhengyi Wang, Ke Xu, Hang Su, and Jun Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. In *The Thirteenth International Conference on Learning Representations*, 2025.

[61] Jing Ma, Zhengyi Jiang, Rifat Hoque, Sangwoo Ahn, Pulkit Agrawal, and Kaiming Lee. Hierarchical diffusion policy for kinematics-aware multi-task robotic manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18081–18090, 2024.

[62] Yunchao Ma, Yizhuang Zhou, Yunhuan Yang, Tiancai Wang, and Haoqiang Fan. Running vlas at real-time speed. *arXiv preprint arXiv:2510.26742*, 2025.

[63] Ajay Mandlekar, Yuke Zhu, Animesh Garg, Jonathan Booher, Max Spero, Albert Tung, Julian Gao, John Emmons, Anchit Gupta, Emre Orbay, et al. Roboturk: A crowdsourcing platform for robotic skill learning through imitation. In *Conference on Robot Learning*,

pages 879–893. PMLR, 2018.

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

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

[66] Zehao Ni, Yonghao He, Lingfeng Qian, Jilei Mao, Fa Fu, Wei Sui, Hu Su, Junran Peng, Zhipeng Wang, and Bin He. Vo-dp: Semantic-geometric adaptive diffusion policy for vision-only robotic manipulation. *arXiv preprint arXiv:2510.15530*, 2025.

[67] Abby O'Neill, Abdul Rehman, Abhiram Maddukuri, Abhishek Gupta, Abhishek Padalkar, Abraham Lee, Acorn Pooley, Agrim Gupta, Ajay Mandlekar, Ajinkya Jain, 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)*, pages 6892–6903. IEEE, 2024.

[68] Mingjie Pan, Jiyao Zhang, Tianshu Wu, Yinghao Zhao, Wenlong Gao, and Hao Dong. Omnimanip: Towards general robotic manipulation via object-centric interaction primitives as spatial constraints. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17359–17369, 2025.

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

[70] Jingjing Qian, Boyao Han, Chen Shi, Lei Xiao, Long Yang, Shaoshuai Shi, and Li Jiang. Geopredict: Leveraging predictive kinematics and 3d gaussian geometry for precise vla manipulation. *arXiv preprint arXiv:2512.16811*, 2025.

[71] 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. In *Robotics: Science and Systems*, 2025.

[72] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollar, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=Ha6RTeWMd0.

[73] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, Zhaoyang Zeng, Hao Zhang, Feng Li, Jie Yang, Hongyang Li, Qing Jiang, and Lei Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.

[74] Ralf Römer, Yi Zhang, and Angela P Schoellig. Clare: Continual learning for vision-language-action models via autonomous adapter routing and expansion. *arXiv preprint arXiv:2601.09512*, 2026.

[75] Weijie Shen, Yitian Liu, Yuhao Wu, Zhixuan Liang, Sijia Gu, Dehui Wang, Tian Nian, Lei Xu, Yusen Qin, Jiangmiao Pang, et al. Expertise need not monopolize: Action-specialized mixture of experts for vision-language-action learning. *arXiv preprint arXiv:2510.14300*, 2025.

[76] Lin Sun, Bin Xie, Yingfei Liu, Hao Shi, Tiancai Wang, and Jiale Cao. Geovla: Empowering 3d representations in vision-language-action models. *arXiv preprint arXiv:2508.09071*, 2025.

[77] Shuhan Tan, Kairan Dou, Yue Zhao, and Philipp Krähenbühl. Interactive post-training for vision-language-action models. *arXiv preprint arXiv:2505.17016*, 2025.

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

[79] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[80] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

[81] Homer Rich Walke, Kevin Black, Tony Z Zhao, Quan Vuong, Chongyi Zheng, Philippe Hansen-Estruch, Andre Wang He, Vivek Myers, Moo Jin Kim, Max Du, et al. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning*, pages 1723–1736. PMLR, 2023.

[82] Weikang Wan, Yifeng Zhu, Rutav Shah, and Yuke Zhu. LOTUS: Continual imitation learning for robot manipulation through unsupervised skill discovery. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 2024. doi: 10.1109/ICRA57147.2024.10611129.

[83] Jason Z Wang, Kuan Fang, Tonghe Zhang, and Yunzhu Li. 3d diffuser actor: Policy diffusion with 3d scene representations. *arXiv preprint arXiv:2402.10885*, 2024.

[84] Junyao Wang, Qianli Liu, Bo Zhang, Yiran Zhang, Shengfa Li, Yuda Li, Ziwei Liu, Kaijie Wang, Yijiang Zhu, Jinxi Li, et al. Vla-adapter: An effective paradigm for tiny-scale vision-language-action model.

In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 18638–18646, 2026. doi: 10.1609/aaai.v40i22.38931.

[85] Yating Wang, Haoyi Zhu, Mingyu Liu, Jiange Yang, Hao-Shu Fang, and Tong He. Vq-vla: Improving vision-language-action models via scaling vector-quantized action tokenizers. *arXiv preprint arXiv:2507.01016*, 2025.

[86] Junjie Wen, Minjie Zhu, Jiaming Liu, Zhiyuan Liu, Yicun Yang, Linfeng Zhang, Shanghang Zhang, Yichen Zhu, and Yi Xu. dvla: Diffusion vision-language-action model with multimodal chain-of-thought. *arXiv preprint arXiv:2509.25681*, 2025.

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

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

[89] Shijie Wu, Yihang Zhu, Yunao Huang, Kaizhen Zhu, Jiayuan Gu, Jingyi Yu, Ye Shi, and Jingya Wang. Afforddp: Generalizable diffusion policy with transferable affordance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6971–6980, June 2025.

[90] Zhenjie Yang, Yilin Chai, Xiaosong Jia, Qifeng Li, Yuqian Shao, Xuekai Zhu, Haisheng Su, and Junchi Yan. Drivemoe: Mixture-of-experts for vision-language-action model in end-to-end autonomous driving. *arXiv preprint arXiv:2505.16278*, 2025.

[91] Hang Yu, Juntu Zhao, Yufeng Liu, Kaiyu Li, Cheng Ma, Di Zhang, Yingdong Hu, Guang Chen, Junyuan Xie, Junliang Guo, et al. Point what you mean: Visually grounded instruction policy. *arXiv preprint arXiv:2512.18933*, 2025.

[92] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*, pages 1094–1100. PMLR, 2020.

[93] Tianyuan Yuan, Yicheng Liu, Chenhao Lu, Zhuoguang Chen, Tao Jiang, and Hang Zhao. Depthvla: Enhancing vision-language-action models with depth-aware spatial reasoning. *arXiv preprint arXiv:2510.13375*, 2025.

[94] Yanjie Ze, Gu Zhang, Kangning Zhang, Chenyuan Hu, Muhan Wang, and Huazhe Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations. In *Robotics: Science and Systems (RSS)*, 2024. doi: 10.15607/RSS.2024.XX.067.

[95] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3836–3847, 2023.

[96] Wenyao Zhang, Hongsi Liu, Zekun Qi, Yunnan Wang, Xinqiang Yu, Jiazhao Zhang, Runpei Dong, Jiawei He, Fan Lu, He Wang, et al. Dreamvla: a vision-language-action model dreamed with comprehensive world knowledge. In *NeurIPS*, 2025.

[97] Ruihan Zhao, Tyler Ingebrand, Sandeep Chinchali, and Ufuk Topcu. Mos-vla: A vision-language-action model with one-shot skill adaptation. *arXiv preprint arXiv:2510.16617*, 2025.

[98] Tony Z Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[99] Yongsheng Zhao, Lei Zhao, Baoping Cheng, Gongxin Yao, Xuanzhang Wen, and Han Gao. Vla-rail: A real-time asynchronous inference linker for vla models and robots. *arXiv preprint arXiv:2512.24673*, 2025.

[100] Haoyu Zhen, Xiaowen Qiu, Peihao Chen, Jincheng Yang, Xin Yan, Yilun Du, Yining Hong, and Chuang Gan. 3d-vla: A 3d vision-language-action generative world model. In *International Conference on Machine Learning*, pages 61229–61245. PMLR, 2024.

[101] Ruijie Zheng, Yongyuan Liang, Shuaiyi Huang, Jianfeng Gao, Hal Daumé III, Andrey Kolobov, Furong Huang, and Jianwei Yang. Tracevla: Visual trace prompting enhances spatial-temporal awareness for generalist robotic policies. *arXiv preprint arXiv:2412.10345*, 2024.

[102] Hongyan Zhi, Peihao Chen, Siyuan Zhou, Yubo Dong, Quanxi Wu, Lei Han, and Mingkui Tan. 3dflowaction: Learning cross-embodiment manipulation from 3d flow world model. *arXiv preprint arXiv:2506.06199*, 2025.

[103] Yifeng Zhu, Peter Stone, and Yuke Zhu. Bottom-up skill discovery from unsegmented demonstrations for long-horizon robot manipulation. *IEEE Robotics and Automation Letters*, 7(2):4126–4133, 2022. doi: 10.1109/LRA.2022.3146589.

[104] Brianna Zitkovich, Tianhe Yu, Sichun Xu, Peng Xu, Ted Xiao, Fei Xia, Jialin Wu, Paul Wohlhart, Stefan Welker, Ayzaan Wahid, et al. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Conference on Robot Learning*, pages 2165–2183. PMLR, 2023.