EMBODIEDMIDTRAIN: BRIDGING THE GAP BETWEEN VISION-LANGUAGE MODELS AND VISION-LANGUAGE-ACTION MODELS VIA MID-TRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision-Language-Action Models (VLAs) inherit their visual and linguistic capabilities from Vision-Language Models (VLMs), yet most VLAs are built from off-the-shelf VLMs that are not adapted to the embodied domain, limiting their downstream performance. In this work, we propose *EmbodiedMidtrain* to bridge the gap between VLMs and VLAs. We first characterize the data distribution gap between them, showing that VLA data occupy compact regions that are largely separated from the broader VLM distribution, while the degree of alignment varies substantially both across and within VLM data sources. Then, we build a mid-training data engine that leverages a lightweight learnable proximity estimator to select the most VLA-aligned candidates from a large VLM pool, and produce *EmbodiedMidtrain-1M*, a VLM dataset that we further use to mid-train the VLM before downstream VLA fine-tuning. Experiments on both simulated robot manipulation benchmarks and real-world tasks show consistent VLA performance gains over off-the-shelf checkpoints across VLM backbones and VLA action heads. These gains extend to downstream tasks not used to guide data selection. Further analysis reveals that mid-training provides a stronger initialization for VLA fine-tuning, with gains emerging from the earliest steps and widening throughout training. Moreover, the data engine captures both dataset-level and sample-level alignment signals, favoring spatial reasoning over text-centric tasks while preserving the diversity of the VLM data. We will release all code, data and models for future research.

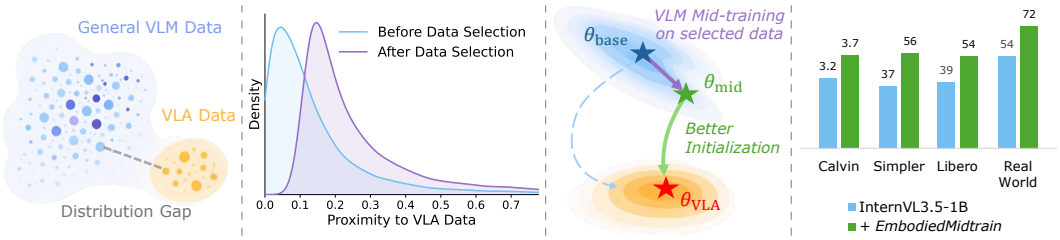
1 INTRODUCTION

Recent advances in robotics foundation models have given rise to Vision-Language-Action Models (VLAs), which unify vision perception, language understanding, and action generation within a single model to enable generalist robot control across diverse tasks, environments, and embodiments (Kim et al., 2024; Black et al., 2024; Gemini Robotics Team et al., 2025). Most VLAs leverage Vision-Language Models (VLMs) as their backbone, aiming to inherit rich visual and linguistic representations that facilitate rapid adaptation to embodied settings (Kim et al., 2025a; Black et al., 2025; NVIDIA et al., 2025).

Despite this progress, a fundamental gap exists: most VLAs initialize from general-purpose *off-the-shelf* VLMs that are not tailored toward embodied action generation. VLM pretraining (Liu et al., 2023b; Li et al., 2023a; Bai et al., 2023) covers broad vision-language tasks such as captioning, visual question answering, and document understanding, whereas VLA training consists of robotic manipulation trajectories grounded in physical interaction. This distributional mismatch means that, even when a VLM backbone provides a strong initialization for language and visual understanding, its internal representations may not be well-suited for the embodied reasoning that effective action generation demands (Yang et al., 2025). Bridging this gap remains an open challenge: simply fine-tuning VLMs on curated embodied data does not reliably translate into better VLA performance (Zhang et al., 2026), indicating the need for more principled VLM adaptation to the VLA domain.

In this work, we present *EmbodiedMidtrain*, a mid-training framework that aligns VLM data distribution with the VLA domain, enabling VLAs for robot manipulation to truly benefit from the

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.



(a) VLM-VLA Data Distribution Gap (b) VLM Data Distribution Shift (c) VLM Mid-training (d) Downstream VLA Gains

Figure 1: Overview of *EmbodiedMidtrain*. We analyze the data distribution gap between VLMs and VLAs, and select VLM samples with higher proximity to the VLA domain for mid-training, yielding a stronger initialization for downstream VLA fine-tuning.

abilities of their VLM backbones. We begin with a deep analysis of the VLM-VLA representation gap and identify a core issue (Figure 1a): the two domains exhibit a significant data distribution discrepancy. VLA data form compact clusters that are largely separated from the broad, diverse VLM distributions (Xing et al., 2025); yet this gap is not uniform, as some VLM samples are inherently more aligned with the VLA domain than others. This reveals that the bulk of VLM data concentrates on capabilities distant from VLA data, leaving downstream VLA fine-tuning to bridge a wide representational gap from a misaligned starting point.

Leveraging this understanding, we develop a mid-training data engine that **reshapes the VLM training distribution toward the VLA domain** before fine-tuning begins (Figure 1b) to bridge the gap. The key idea is that, rather than training on all available VLM data indiscriminately, we can focus on the subset that is most useful for robotics by aligning with the target VLA distribution, producing a stronger, more relevant initialization for downstream VLA learning. Concretely, we propose a lightweight proximity estimator: a learnable classifier on frozen VLM features that learns to distinguish VLA data from VLM data, and whose predicted scores serve as a continuous measure of each VLM sample's closeness to the VLA domain. Top-scoring samples are then assembled into a curated mid-training mixture, and the VLM is mid-trained on this distribution-aligned dataset to serve as a better initialization for VLA fine-tuning (Figure 1c). This pipeline is lightweight, effective, and requires no architectural changes to the underlying VLM or VLA.

Experiments on Calvin ABC-D (Mees et al., 2022), SimplerEnv Bridge (Walke et al., 2023), and Libero-10 (Liu et al., 2023a) show that our mid-trained 1.1B model achieves a significant performance boost (Figure 1d). Beyond raw performance gain, our analysis reveals several insights into what makes mid-training effective. First, the same curated data transfers across architectures: applying the mixture selected with InternVL3.5-1B to Qwen3VL-2B yields consistent gains, suggesting that proximity-based selection captures transferable properties across VLMs. Second, the learned proximity estimator substantially outperforms hand-crafted alternatives like feature-space distance and perplexity-based metrics. Third, training dynamics analysis shows that the mid-trained model outperforms the baseline from the earliest fine-tuning checkpoints and the gap widens over time, indicating mid-training provides a fundamentally better VLM initialization for VLAs.

Our main contributions are as follows:

- **A proximity-based mid-training pipeline and dataset to bridge the VLM-VLA gap.** We propose *EmbodiedMidtrain*, a mid-training pipeline with a data engine that learns a proximity estimator to score VLM samples by their closeness to the VLA domain, and selects the top-ranked candidates to curate a distribution-aligned mid-training dataset *EmbodiedMidtrain-1M*.
- **Consistent VLA performance gains across backbones, action heads, and target tasks.** Extensive experiments on simulation benchmarks (Calvin ABC-D, SimplerEnv Bridge, and Libero-10) and real-world tasks demonstrate that VLMs mid-trained on our *EmbodiedMidtrain-1M* dataset yield consistent improvements over the original backbone. We further demonstrate its transferability across VLM backbones, VLA action heads and data selection target tasks.
- **Insights on better VLM initialization for VLAs.** We show that mid-training produces an initialization whose advantages emerge from the earliest fine-tuning steps and amplify over time.

We further demonstrate that the learned proximity estimation outperforms hand-crafted alternatives by capturing fine-grained alignment with the VLA domain.

2 Related work

VLA Models. VLAs extend foundation models like VLMs to generate robot actions, leveraging the broad visual and language understanding inherited from VLMs (Kim et al., 2024; Black et al., 2024; NVIDIA et al., 2025). Existing designs differ in backbone choice and action generation mechanism. Token-based approaches such as OpenVLA (Kim et al., 2024) discretize robot actions into tokens for autoregressive generation; OpenVLA-OFT (Kim et al., 2025a) further introduces parallel decoding with learnable action embeddings for efficient continuous action prediction. Flow-matching and diffusion-based approaches such as π_0 (Black et al., 2024) and $\pi_{0.5}$ (Black et al., 2025) instead adopt a VLM backbone (PaliGemma (Beyer et al., 2024)) paired with a continuous action generation head. More recent models such as GR00T N1 (NVIDIA et al., 2025) employ a VLM backbone (Eagle-2 (Li et al., 2025)) with a dedicated cross-attention action decoder. While these designs differ architecturally, a common thread is that the VLM backbone is taken *off-the-shelf* from general-purpose training without specific preparation for the embodied domain, which brings a gap that our work identifies and addresses.

VLM Mid-training. Many VLMs adopt multi-stage training pipelines, where the model is further trained on curated vision-language data after initial multimodal pretraining or alignment and before final instruction tuning (Bai et al., 2023; Wang et al., 2024b; Chen et al., 2025c). In this paper, we refer to this intermediate stage as *VLM mid-training*. Similar to mid-training in LLMs, its goal is to adapt a foundation model toward desired domains or capabilities prior to final post-training or task-specific fine-tuning (Wang et al., 2025b; Grattafiori et al., 2024; Hu et al., 2024; Team OLMo et al., 2025). Our work studies this paradigm in the embodied setting, where mid-training is used to bridge the data distribution gap between general VLM pretraining data and VLA fine-tuning data.

Embodied-oriented VLMs. A broad line of work aims to improve VLMs' embodied capabilities, through both embodied-oriented dataset construction (Du et al., 2024; Zhou et al., 2025; Chen et al., 2025a; Yuan et al., 2025) and model-level adaptation (Cai et al., 2025; Ji et al., 2025; Chen et al., 2025b). Despite these efforts, Zhang et al. (2026) show that these gains on finetuning VLMs on embodied tasks do not reliably transfer to downstream VLA task performance, suggesting that current embodied VLM fine-tuning captures a different signal from what VLA execution requires. Alternatively, Vlaser (Yang et al., 2025) converts in-domain robot trajectories into VQA pairs for VLM fine-tuning. Collectively, these efforts highlight that bridging VLMs and VLAs remains an open problem: existing approaches either improve VLM-side embodied benchmarks without reliably translating into better VLA performance, or require large amounts of in-domain robot data (Yang et al., 2025; Kim et al., 2025b; Zhang et al., 2026). Our work takes a complementary perspective by operating on the diverse and abundant VLM data instead of crafting task-specific corpora.

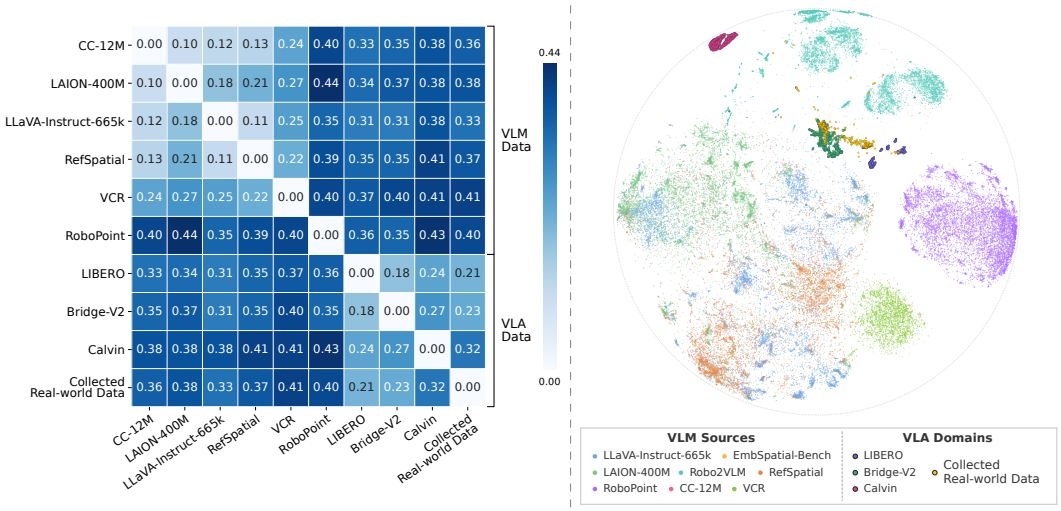
3 Data distribution gap between VLMs and VLAs

Since most VLAs inherit their visual and linguistic representations from VLM pretraining, the quality of this initialization is fundamentally shaped by the data the VLM was trained on. This motivates us to examine the gap between VLMs and VLAs from the perspective of their training data distributions, where we analyze VLM and VLA data in a shared representation space and characterize the gap with both qualitative and quantitative measurements.

To represent VLM and VLA data in a unified distribution space, we extract the last hidden states of InternVL3.5 VLM (Wang et al., 2025a) as the feature representation $h(\cdot)$ for each data sample. We first quantify the distance between each pair of datasets with **Maximum Mean Discrepancy** (MMD). Formally, the squared MMD between two datasets P and Q is:

$$\mathrm{MMD}^2(P, Q) = \mathbb{E}_{x,x' \sim P}[k(x, x')] + \mathbb{E}_{y,y' \sim Q}[k(y, y')] - 2\,\mathbb{E}_{x \sim P,\, y \sim Q}[k(x, y)]$$

where $k(x, y) = \exp\big(-\|h(x) - h(y)\|_2^2 \,/\, 2\sigma^2\big)$ is a Gaussian RBF kernel applied over the extracted features, with bandwidth σ set via the median heuristic (Gretton et al., 2012). Figure 2a reports the globally normalized pairwise MMD scores across all dataset pairs. We further visualize these features using t-SNE (van der Maaten & Hinton, 2008) in Figure 2b.



(a) Pairwise MMD distance matrix between VLM and VLA datasets

(b) t-SNE visualization of visual feature distributions across VLM and VLA datasets

Figure 2: **Distribution analysis of VLM and VLA datasets.** (a) Pairwise MMD distances quantify this distribution gap, with cross-group distances larger than within-group distances. (b) VLA datasets (shown with larger markers) form more compact and concentrated clusters that are separated from the broader and more dispersed VLM distributions.

The characteristics of data distribution gap between VLM and VLA. The distribution space of VLM data and VLA data is largely separated, with only a few near-neighbors. Figure 2a shows that MMD distances are generally smaller within the VLM group and within the VLA group than across the two groups, quantitatively confirming a clear distributional mismatch. Figure 2b further illustrates this pattern: VLA datasets form compact clusters that are mostly detached from the main regions occupied by VLM datasets, while only a small subset of the VLM data lies nearby. These results reveal a clear distribution mismatch between VLM data and VLA data that could affect the performance of downstream embodied tasks.

The data distribution gap exhibits substantial internal heterogeneity. Although VLM and VLA data are separated overall, their mismatch is not uniform within each dataset. Figure 2b shows that part of VLM sources lie noticeably closer to VLA domains than others, with a few local regions exhibiting clear cross-domain proximity despite the global separation. This suggests that the gap between VLM and VLA is better characterized as a spectrum of alignment, revealing substantial heterogeneity in how different parts of the VLM distribution relate to VLA data.

These insights suggest that **enhancing VLMs for embodied tasks requires reshaping the training data distribution toward the VLA domain**. Realizing this goal requires more than coarse dataset-level mixture adjustment; it calls for sample-wise selection within each dataset. This motivates a data-centric mid-training strategy that explicitly prioritizes VLM samples that are most compatible with the target embodied domain.

4 Data Engine for *EmbodiedMidtrain*

The distribution analysis in Section 3 reveals substantial differences between VLM and VLA data, together with heterogeneity across and within VLM sources. These observations motivate a mid-training data engine that prioritizes VLM examples according to their affinity to the target VLA domain. We construct the candidate pool from both general-purpose and embodied-oriented VLM sources to provide varied training examples, and perform selection at the sample level to account for variation within each source.

VLM data source. To support mid-training that bridges the gap between VLM capabilities and downstream VLA performance, we first construct a candidate pool spanning both general-purpose and embodied-oriented VLM datasets. For **general VLM data**, following standard VLM mid-

training data of Qwen-VL (Bai et al., 2023), we include image-captioning data from subsets of LAION-400M (Schuhmann et al., 2021), CC-12M (Changpinyo et al., 2021) with captions relabeled by BLIP (Li et al., 2022), instruction-following data from LLaVA-Instruct-665k (Liu et al., 2024), and Visual Commonsense Reasoning (VCR) (Zellers et al., 2019). For **embodied-oriented VLM data**, we additionally include RefSpatial (Zhou et al., 2025) for spatial referring and reasoning, evaluation examples from EmbSpatial-Bench (Du et al., 2024) for embodied spatial understanding, Robo2VLM (Chen et al., 2025a) for robotic visual question answering, and RoboPoint (Yuan et al., 2025) for spatial affordance prediction.

Proximity-Based Data Selection. Inspired by target-guided data selection for LLM pretraining (Xie et al., 2023), we frame sample scoring as a domain-membership problem. A lightweight binary classifier trained on frozen VLM features distinguishes target VLA samples from candidate VLM samples. We use its predicted probability of VLA membership as a proximity score for ranking candidate examples.

Let D_{VLM} and D_{VLA} denote the candidate VLM pool and target VLA training corpus, respectively. Given a selection budget K and a scoring function s, we select the subset with the highest aggregate proximity score:

$$D_{\mathrm{sel}} = \underset{\substack{D' \subseteq D_{\mathrm{VLM}} \\ |D'|=K}}{\arg\max} \sum_{x \in D'} s(x). \tag{1}$$

Because the objective is additive across samples, its solution is obtained by retaining the K highest-scoring candidates:

$$D_{\mathrm{sel}} = \mathrm{TopK}_{x \in D_{\mathrm{VLM}}}\, s(x). \tag{2}$$

The key question is how to define s. Let $\phi(x)$ denote the frozen VLM representation of sample x, and let p_{VLM} and p_{VLA} denote the two domain densities in this representation space. The density ratio $p_{\mathrm{VLA}}(\phi(x))/p_{\mathrm{VLM}}(\phi(x))$ measures the relative likelihood of a representation under the target domain compared with the candidate distribution.

A classical result connects this ratio to binary classification (Goodfellow et al., 2014). Under equal domain priors, minimizing the population binary cross-entropy objective over unrestricted scoring functions yields the optimal domain-membership score:

$$s^*(x) = \frac{p_{\mathrm{VLA}}(\phi(x))}{p_{\mathrm{VLA}}(\phi(x)) + p_{\mathrm{VLM}}(\phi(x))}. \tag{3}$$

This score is monotonically increasing in the density ratio. The result motivates using a learned classifier as a practical proxy for relative domain likelihood; the downstream usefulness of the resulting selection is evaluated empirically.

We instantiate the proximity estimator as a linear scoring function f on the frozen VLM's last-token, last-hidden-state representation, followed by a sigmoid:

$$s(x) = \sigma(f(\phi(x)))\,. \tag{4}$$

We train the estimator using VLA training samples as positives and candidate VLM samples as negatives, with equal weighting of the two domains:

$$\mathcal{L}_{\mathrm{cls}} = -\mathbb{E}_{y \sim D_{\mathrm{VLA}}}\left[\log s(y)\right] - \mathbb{E}_{x \sim D_{\mathrm{VLM}}}\left[\log(1 - s(x))\right]. \tag{5}$$

After training, we score all candidate VLM samples and retain the top K according to Equation 2. This produces *EmbodiedMidtrain-1M*, the corpus used for subsequent VLM mid-training. The selection procedure prioritizes high-scoring examples from the heterogeneous candidate pool, with its downstream effectiveness and selected-data characteristics examined in Section 6.

5 Experiments

We conduct experiments to demonstrate that the proposed mid-training data engine bridges the VLM-VLA gap. In particular, we show that mid-training with proximity-based data selection consistently improves multiple VLM backbones on downstream VLA tasks, and that the curated data mixture exhibits cross-backbone transferability.

5.1 SETUP

VLM mid-training. We first construct the mid-training dataset using the data engine described in Section 4. We leverage a balanced mixture of training data in VLA fine-tuning as the target VLA data for proximity estimator learning. To prevent overfitting, we apply early stopping at 90% validation accuracy. The resulting proximity estimator scores and selects candidate VLM samples to conduct mid-training on InternVL3.5-1B (Wang et al., 2025a) and Qwen3VL-2B (Bai et al., 2025a) using the LLaMA-Factory (Zheng et al., 2024) framework. We train all model parameters with a global batch size of 256 for 5,000 steps. See Appendix A for more implementation details.

VLA fine-tuning. After VLM mid-training, we initialize a VLA from the resulting VLMs and fine-tune it following the VLA training pipeline of VLM4VLA (Zhang et al., 2026), which enables controlled comparisons of VLM initializations by fixing the downstream robot data and using a shared lightweight two-branch MLP action head. By contrast, models such as $\pi_{0.5}$ (Black et al., 2025) introduce large-scale robot pretraining, making the contribution of VLM initialization harder to isolate. The architecture and training details are provided in Appendix C.

Evaluation benchmarks. We evaluate on three simulated manipulation benchmarks that cover multiple aspects of embodied control, following VLM4VLA's evaluation protocol for fair and reproducible comparison. **Calvin ABC-D** (Mees et al., 2022) trains on the ABC splits and evaluates on the unseen split D over 1,000 five-subtask sequences, testing generalization to novel scene configurations. **SimplerEnv Bridge** (Li et al., 2024; Walke et al., 2023) is a real-to-sim benchmark with four tabletop manipulation tasks and 24 randomized trials each, reporting mean success rate. **LIBERO-10** (Liu et al., 2023a), the most challenging suite in the LIBERO benchmark, comprises 10 long-horizon tasks evaluated on 50 trials per task.

5.2 BASELINES

We compare our method against two categories of baselines: expert VLA models and VLAs trained from off-the-shelf VLMs. These baselines are experimented under the framework of VLM4VLA for a unified comparison. Additional implementation details are provided in the Appendix C.

Expert VLA Baselines. We first compare with representative expert VLA models, including OpenVLA (Kim et al., 2024) and π_0 (Black et al., 2024). OpenVLA is built on Llama-2-7B with DINOv2 and SigLIP visual encoders, and models robot control by autoregressively predicting discretized action tokens. π_0 is based on Paligemma-1 VLM and models continuous robot actions using flow matching. We adapt VLM4VLA's implementation on the architecture of OpenVLA and π_0 and apply the same training recipe to make results comparable.

Off-the-shelf VLM Baselines. To evaluate how pretrained VLM abilities transfer to downstream robot manipulation tasks, we further consider VLAs directly finetuned from a diverse set of off-the-shelf VLMs spanning different architectures and scales. These VLMs include the Qwen2.5VL and Qwen3VL family (Bai et al., 2025b;a), which represents a strong line of general-purpose open-source VLMs; the Paligemma family (Paligemma-1 (Beyer et al., 2024) and Paligemma-2 (Steiner et al., 2024)), which is widely adopted for downstream adaptation; and KosMos-2 (Peng et al., 2023), which provides a grounding-oriented alternative.

5.3 RESULTS

Table 1 presents the main results. With proximity-based mid-training, our models achieve consistent and substantial improvements across all three benchmarks, while finetuned on the same robot action data, demonstrating that a well-curated mid-training mixture can significantly strengthen a VLM's readiness for downstream action learning. More details are provided in Appendix E.

Consistent performance gains across benchmarks. *EmbodiedMidtrain* improves both InternVL3.5-1B and Qwen3VL-2B on all three benchmarks relative to their off-the-shelf initializations. For InternVL3.5-1B, mid-training increases the average number of consecutively completed tasks on Calvin ABC-D by 0.541 and the success rates on SimplerEnv Bridge and LIBERO-10 by 19.8 and 15.2 percentage points, respectively.

Table 1: Main results across Calvin ABC-D, SimplerEnv-Bridge, and Libero-10. *EmbodiedMidtrain* consistently improves the performance of downstream VLA benchmarks for both Qwen3-VL-2B and InternVL3.5-1B backbones. * Results for Expert VLA Baselines and Off-the-shelf VLM Baselines are reported by VLM4VLA under the same framework.

Model	Size	Calvin (Tasks Completed in a Row)						Simpler↑	Libero↑
		1	2	3	4	5	Avg. Len.↑		
Expert VLA Baselines*									
OpenVLA (Llama-2)	7.7B	0.792	0.644	0.499	0.368	0.245	2.548	4.2	53.7
π_0 (Paligemma-1)	3.1B	0.896	0.785	0.686	0.610	0.532	3.509	60.4	46.0
Off-the-shelf VLM Baselines*									
Qwen2.5VL-3B	3.8B	0.922	0.842	0.766	0.700	0.626	3.856	48.0	43.0
Qwen2.5VL-7B	8.3B	0.935	0.864	0.807	0.758	0.693	4.057	46.8	45.0
Qwen3VL-4B	4.4B	0.933	0.857	0.790	0.719	0.644	3.943	56.3	44.4
Qwen3VL-8B	8.8B	0.940	0.868	0.797	0.746	0.684	4.035	58.3	46.2
Qwen3VL-30B-A3B	30B-A3B	0.939	0.877	0.820	0.757	0.682	4.075	44.8	46.8
Paligemma-1	2.9B	0.914	0.813	0.692	0.599	0.488	3.506	55.3	44.2
Paligemma-2	3.0B	0.901	0.775	0.669	0.575	0.486	3.406	57.3	46.2
KosMos-2	1.7B	0.878	0.721	0.591	0.498	0.408	3.096	60.4	55.0
VLM with *EmbodiedMidtrain* (Ours)									
Qwen3VL-2B	2.1B	0.887	0.747	0.612	0.527	0.432	3.205	38.5	33.8
+ *EmbodiedMidtrain*	2.1B	0.922	0.808	0.700	0.623	0.533	3.584	45.8	40.2
InternVL3.5-1B	1.1B	0.909	0.754	0.606	0.498	0.406	3.173	36.5	39.0
+ *EmbodiedMidtrain*	1.1B	0.935	0.838	0.737	0.653	0.551	**3.714**	**56.3**	**54.2**

Transferability across VLM backbones. To assess transfer across backbones, we mid-train Qwen3VL-2B on the same data selected using InternVL3.5-1B's feature space. The gains on Qwen3VL-2B are therefore obtained without training a new proximity estimator or reselecting data for this backbone. This suggests that proximity-based selection captures aspects of alignment with downstream VLA tasks that transfer across VLM architectures.

5.4 Real-World Experiments

We further demonstrate that the gains from mid-training transfer to real-world manipulation tasks. We collect 50 demonstrations for each of five manipulation tasks and evaluate with 10 rollouts per task on a Franka Research 3 robot with a Robotiq 2F-85 gripper. As shown in Figure 3, *EmbodiedMidtrain* increases the overall success rate from 54% to 72% across five tasks, improving performance on four real-world tasks.

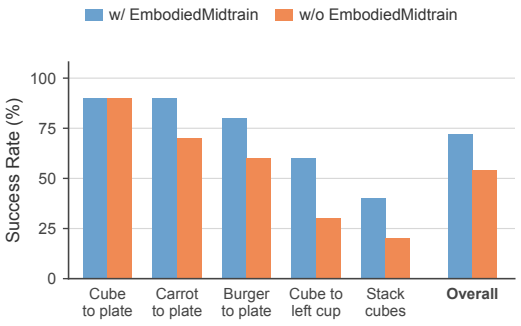


Figure 3: Real-world success rates with and without *EmbodiedMidtrain* across five manipulation tasks on InternVL-3.5-1B backbone.

For real-world experiments, we initialize the VLA with the same mid-trained VLM checkpoint used for the simulation experiments in Section 5.3, then fine-tune it on the collected real-world demonstrations. These results show that ***EmbodiedMidtrain* can select mid-training data using simulation data and still improve performance on different real-world manipulation tasks**. Details and qualitative examples are provided in Appendices D and L.

Additional experiments in Appendix H show transferability across VLA action heads and target tasks: downstream tasks not used to guide data selection still benefit from *EmbodiedMidtrain*. Together with the gains across VLM backbones, these findings suggest that the mid-training provides better initialization broadly transferrable on embodied manipulation tasks.

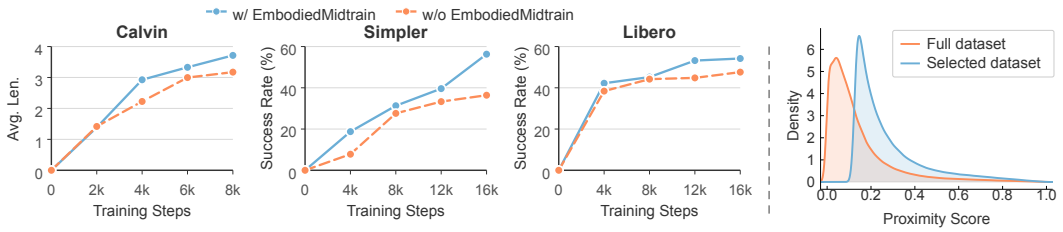


Figure 4: The training dynamics across VLA tasks for VLMs with and without *EmbodiedMidtrain*, showing downstream VLA task performances (left three panels) alongside the VLM data distribution shift of proximity score before and after data selection (right).

6 Analysis

Having established the effectiveness of our approach, we now examine its design choices and behavior in detail. We first ablate two core components of our data engine: the effect of data selection versus random sampling, and the choice of proximity measurement (Section 6.1). We then analyze the training dynamics of VLA fine-tuning to understand *when* the benefits of mid-training emerge (Section 6.2), and finally inspect the selected data to understand *what* the proximity estimator learns to prefer (Section 6.3).

6.1 Ablations

We ablate two central design choices in our data engine: the advantage of proximity-based selection over random sampling, and the effectiveness of different proximity measurements. Table 2 summarizes the results.

Table 2: Ablation results for random selection and different proximity measurements on mid-training InternVL3.5-1B backbone.

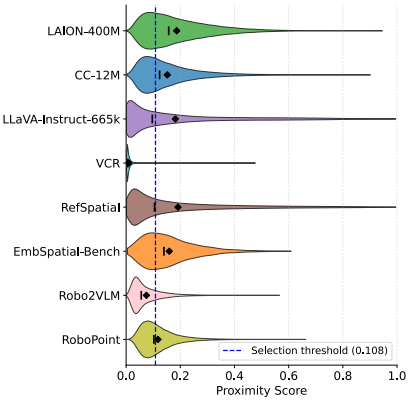
Setting	Calvin↑	Simpler↑	Libero↑
Random Selection	3.398	43.8	48.4
Proximity Measurements			
Feat.-space Avg. Dist.	3.126	53.1	51.2
VLA-cond. Perplexity	3.159	55.2	48.0
Delta Perplexity	3.221	39.6	54.2
Learned Estimator (*Ours*)	**3.714**	**56.3**	**54.2**

Random selection. Randomly sampling from the full candidate pool consistently underperforms our learned estimator on all three benchmarks, showing that naive mid-training on unfiltered data is insufficient to bridge the distribution gap. This suggests that the gains do not come merely from additional mid-training, but from identifying and retaining the subset of VLM data that is better aligned with the VLA domain. Proximity-based selection is therefore critical to unlocking the benefit of mid-training.

Proximity measurements. Beyond our learned proximity estimator, we evaluate three alternatives: *feature-space average distance* (mean distance to VLA samples in the frozen VLM's representation space), *VLA-conditioned perplexity* (perplexity under a VLM fine-tuned on text-form VLA data), and *delta perplexity* (perplexity change relative to the original VLM). All alternatives are less consistent than our learned estimator, which confirms that hand-crafted metrics only partially capture VLA alignment, whereas the learned estimator provides a more robust and transferable signal. Formal definitions and details of each alternative are provided in Appendix B.

6.2 Training Dynamics

Figure 4 compares VLA fine-tuning trajectories from the original InternVL3.5-1B backbone and our mid-trained variant, evaluated at intermediate checkpoints throughout training. The mid-trained model already achieves higher performance in the early stages of fine-tuning, providing direct evidence that **proximity-based mid-training yields a better initialization for VLA learning**. Moreover, this advantage is not a transient head start, since the mid-trained model consistently outperforms the baseline throughout the entire training trajectory, and the gap widens rather than narrowing over time. Together with the consistently stronger downstream performance of the mid-trained model, these



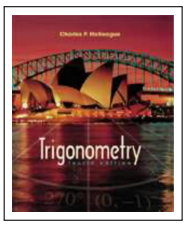
(a) Per-dataset proximity score



Q: *You are standing at the point marked by the coordinate point at point (0.878, 0.780). Which object is directly in front of you?*
A: *The white matte truck at lower right.*

Q: *Locate a point on the yellow metallic crane at upper right.* (format instructions...)
A: `[(0.976, 0.244)]`

(b) High-scoring sample



Q: *Who wrote this book?*
A: *Charles P. McKeague.*

Q: *What is the title of this book?*
A: *Trigonometry.*

(c) Low-scoring sample

Figure 5: Analysis of proximity-based data selection. (a) Distribution of proximity scores across VLM data sources. (b) A high-scoring sample from RefSpatial requiring spatial grounding and reasoning. (c) A low-scoring sample: a book cover with text-only VQA.

results indicate that the benefits of mid-training persist throughout VLA fine-tuning and translate into a backbone better suited to VLA tasks.

6.3 Analysis of selected VLM data

Figure 5a presents the proximity score distributions across eight candidate VLM datasets, visualized as violin plots. While all datasets concentrate in the low-to-moderate score range, the distribution shapes vary noticeably across datasets. Among them, RefSpatial achieves the highest average scores while VCR receives the lowest, indicating that the estimator assigns clear *dataset-level* preferences. At the same time, the within-dataset score spread shows that the estimator also performs fine-grained *sample-level* selection, retaining only the most VLA-aligned samples even from high-scoring datasets.

Figures 5b and 5c illustrate this with representative examples. The high-scoring sample requires grounding spatial references and exercising spatial reasoning crucial to embodied manipulation. The low-scoring sample involves recognizing text on a book cover, bearing little relevance to embodied tasks. This suggests that the proximity estimator learns to distinguish visual reasoning patterns that transfer to embodied control from those that do not, without requiring manual domain expertise.

The overall proximity score distribution before and after data selection is presented in the rightmost panel of Figure 4. Compared with the full candidate pool, the selected subset exhibits a distribution shift toward higher proximity scores, with part of the low-score mass removed, showing that the reshaped data distribution is more aligned with the VLA domain. Further analyses of dataset composition and diversity preservation are provided in Appendix F.

7 Conclusion

We presented *EmbodiedMidtrain*, a mid-training pipeline that bridges the data distribution gap between VLMs and VLAs through proximity-based data selection. Built on frozen VLM features, our lightweight estimator identifies the VLM samples most aligned with the VLA domain and uses them to construct a more effective mid-training set. Across simulation benchmarks and real-world tasks, this strategy consistently improves performance competitive with models trained at larger scales and training budgets. The gain transfers across VLM backbones, VLA action heads and target tasks.

Our analysis further shows that mid-training provides a stronger initialization for downstream VLA learning: the advantage emerges from the earliest stages of fine-tuning and continues to grow throughout training. We also find that the selected data exhibit meaningful preferences at both the dataset and sample levels, while preserving the diversity of the original VLM dataset. We hope these findings can inform future efforts to better prepare foundation models for embodied intelligence.

AI USE STATEMENT

In this work, we used generative AI tools including ChatGPT and Claude to polish the writing, and generate code to help implement the proposed method. We have not used generative AI for ideation, hypothesis formulation, dataset generation, or result interpretation. We have reviewed all the content generated by AI, including polished texts and codes. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the information needed to reproduce our results in the main text and appendix. The proximity-based data engine and mid-training procedure are described in Section 4, and the full experimental setup is given in Section 5.1. Implementation details and hyperparameters for VLM mid-training and VLA fine-tuning are reported in Appendices A, C and D. As stated in the abstract, we will release all code, data, and models to support reproduction.

REFERENCES

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond, 2023. URL https://arxiv.org/abs/2308.12966.

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. *arXiv preprint arXiv:2511.21631*, 2025a.

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

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

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. *arXiv preprint arXiv:2410.24164*, 2024.

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert 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. In Joseph Lim, Shuran Song, and Hae-Won Park (eds.), *Proceedings of The 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 17–40. PMLR, 27–30 Sep 2025. URL https://proceedings.mlr.press/v305/black25a.html.

Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 9490–9498. IEEE, 2025.

Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

Kaiyuan Chen, Shuangyu Xie, Zehan Ma, Pannag R Sanketi, and Ken Goldberg. Robo2vlm: Visual question answering from large-scale in-the-wild robot manipulation datasets, 2025a. URL https://arxiv.org/abs/2505.15517.

Xinyi Chen, Yilun Chen, Yanwei Fu, Ning Gao, Jiaya Jia, Weiyang Jin, Hao Li, Yao Mu, Jiangmiao Pang, Yu Qiao, Yang Tian, Bin Wang, Bolun Wang, Fangjing Wang, Hanqing Wang, Tai Wang, Ziqin Wang, Xueyuan Wei, Chao Wu, Shuai Yang, Jinhui Ye, Junqiu Yu, Jia Zeng, Jingjing Zhang, Jinyu Zhang, Shi Zhang, Feng Zheng, Bowen Zhou, and Yangkun Zhu. Internvla-m1: A spatially guided vision-language-action framework for generalist robot policy, 2025b. URL https://arxiv.org/abs/2510.13778.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, Lixin Gu, Xuehui Wang, Qingyun Li, Yiming Ren, Zixuan Chen, Jiapeng Luo, Jiahao Wang, Tan Jiang, Bo Wang, Conghui He, Botian Shi, Xingcheng Zhang, Han Lv, Yi Wang, Wenqi Shao, Pei Chu, Zhongying Tu, Tong He, Zhiyong Wu, Huipeng Deng, Jiaye Ge, Kai Chen, Kaipeng Zhang, Limin Wang, Min Dou, Lewei Lu, Xizhou Zhu, Tong Lu, Dahua Lin, Yu Qiao, Jifeng Dai, and Wenhai Wang. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling, 2025c. URL https://arxiv.org/abs/2412.05271.

Mengfei Du, Binhao Wu, Zejun Li, Xuanjing Huang, and Zhongyu Wei. EmbSpatial-bench: Benchmarking spatial understanding for embodied tasks with large vision-language models. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar (eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 346–355, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-short.33. URL https://aclanthology.org/2024.acl-short.33/.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. Blink: Multimodal large language models can see but not perceive, 2024. URL https://arxiv.org/abs/2404.12390.

Gemini Robotics Team, Abbas Abdolmaleki, Saminda Abeyruwan, Joshua Ainslie, Jean-Baptiste Alayrac, Montserrat Gonzalez Arenas, Ashwin Balakrishna, Nathan Batchelor, Alex Bewley, Jeff Bingham, Michael Bloesch, Konstantinos Bousmalis, Philemon Brakel, Anthony Brohan, Thomas Buschmann, Arunkumar Byravan, Serkan Cabi, Ken Caluwaerts, Federico Casarini, Christine Chan, Oscar Chang, London Chappellet-Volpini, Jose Enrique Chen, Xi Chen, Hao-Tien Lewis Chiang, Krzysztof Choromanski, Adrian Collister, David B. D'Ambrosio, Sudeep Dasari, Todor Davchev, Meet Kirankumar Dave, Coline Devin, Norman Di Palo, Tianli Ding, Carl Doersch, Adil Dostmohamed, Yilun Du, Debidatta Dwibedi, Sathish Thoppay Egambaram, Michael Elabd, Tom Erez, Xiaolin Fang, Claudio Fantacci, Cody Fong, Erik Frey, Chuyuan Fu, Ruiqi Gao, Marissa Giustina, Keerthana Gopalakrishnan, Laura Graesser, Oliver Groth, Agrim Gupta, Roland Hafner, Steven Hansen, Leonard Hasenclever, Sam Haves, Nicolas Heess, Brandon Hernaez, Alex Hofer, Jasmine Hsu, Lu Huang, Sandy H. Huang, Atil Iscen, Mithun George Jacob, Deepali Jain, Sally Jesmonth, Abhishek Jindal, Ryan Julian, Dmitry Kalashnikov, M. Emre Karagozler, Stefani Karp, Matija Kecman, J. Chase Kew, Donnie Kim, Frank Kim, Junkyung Kim, Thomas Kipf, Sean Kirmani, Ksenia Konyushkova, Li Yang Ku, Yuheng Kuang, Thomas Lampe, Antoine Laurens, Tuan Anh Le, Isabel Leal, Alex X. Lee, Tsang-Wei Edward Lee, Guy Lever, Jacky

Liang, Li-Heng Lin, Fangchen Liu, Shangbang Long, Caden Lu, Sharath Maddineni, Anirudha Majumdar, Kevis-Kokitsi Maninis, Andrew Marmon, Sergio Martinez, Assaf Hurwitz Michaely, Niko Milonopoulos, Joss Moore, Robert Moreno, Michael Neunert, Francesco Nori, Joy Ortiz, Kenneth Oslund, Carolina Parada, Emilio Parisotto, Amaris Paryag, Acorn Pooley, Thomas Power, Alessio Quaglino, Haroon Qureshi, Rajkumar Vasudeva Raju, Helen Ran, Dushyant Rao, Kanishka Rao, Isaac Reid, David Rendleman, Krista Reymann, Miguel Rivas, Francesco Romano, Yulia Rubanova, Peter Pastor Sampedro, Pannag R Sanketi, Dhruv Shah, Mohit Sharma, Kathryn Shea, Mohit Shridhar, Charles Shu, Vikas Sindhwani, Sumeet Singh, Radu Soricut, Rachel Sterneck, Ian Storz, Razvan Surdulescu, Jie Tan, Jonathan Tompson, Saran Tunyasuvunakool, Jake Varley, Grace Vesom, Giulia Vezzani, Maria Bauza Villalonga, Oriol Vinyals, René Wagner, Ayzaan Wahid, Stefan Welker, Paul Wohlhart, Chengda Wu, Markus Wulfmeier, Fei Xia, Ted Xiao, Annie Xie, Jinyu Xie, Peng Xu, Sichun Xu, Ying Xu, Zhuo Xu, Jimmy Yan, Sherry Yang, Skye Yang, Yuxiang Yang, Hiu Hong Yu, Wenhao Yu, Wentao Yuan, Yuan Yuan, Jingwei Zhang, Tingnan Zhang, Zhiyuan Zhang, Allan Zhou, Guangyao Zhou, and Yuxiang Zhou. Gemini robotics 1.5: Pushing the frontier of generalist robots with advanced embodied reasoning, thinking, and motion transfer, 2025. URL https://arxiv.org/abs/2510.03342.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, pp. 2672–2680, Cambridge, MA, USA, 2014. MIT Press.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song,

Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012. URL http://jmlr.org/papers/v13/gretton12a.html.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zheng Leng Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. Minicpm: Unveiling the potential of small language models with scalable training strategies, 2024. URL https://arxiv.org/abs/2404.06395.

Yuheng Ji et al. Robobrain: A unified brain model for robotic manipulation from abstract to concrete. *arXiv preprint arXiv:2502.21257*, 2025.

Alexander Khazatsky, Karl Pertsch, Suraj Nair, Ashwin Balakrishna, Sudeep Dasari, Siddharth Karamcheti, Soroush Nasiriany, Mohan Kumar Srirama, Lawrence Yunliang Chen, Kirsty Ellis, Peter David Fagan, Joey Hejna, Masha Itkina, Marion Lepert, Yecheng Jason Ma, Patrick Tree Miller, Jimmy Wu, Suneel Belkhale, Shivin Dass, Huy Ha, Arhan Jain, Abraham Lee, Youngwoon Lee, Marius Memmel, Sungjae Park, Ilija Radosavovic, Kaiyuan Wang, Albert Zhan, Kevin Black, Cheng Chi, Kyle Beltran Hatch, Shan Lin, Jingpei Lu, Jean Mercat, Abdul Rehman, Pannag R Sanketi, Archit Sharma, Cody Simpson, Quan Vuong, Homer Rich Walke, Blake Wulfe, Ted Xiao, Jonathan Heewon Yang, Arefeh Yavary, Tony Z. Zhao, Christopher Agia, Rohan Baijal, Mateo Guaman Castro, Daphne Chen, Qiuyu Chen, Trinity Chung, Jaimyn Drake, Ethan Paul Foster, Jensen Gao, Vitor Guizilini, David Antonio Herrera, Minho Heo, Kyle Hsu, Jiaheng Hu, Muhammad Zubair Irshad, Donovon Jackson, Charlotte Le, Yunshuang Li, Kevin Lin, Roy Lin, Zehan Ma, Abhiram Maddukuri, Suvir Mirchandani, Daniel Morton, Tony Nguyen, Abigail O'Neill, Rosario Scalise, Derick Seale, Victor Son, Stephen Tian, Emi Tran, Andrew E. Wang, Yilin Wu, Annie Xie, Jingyun Yang, Patrick Yin, Yunchu Zhang, Osbert Bastani, Glen Berseth, Jeannette Bohg, Ken Goldberg, Abhinav Gupta, Abhishek Gupta, Dinesh Jayaraman, Joseph J Lim, Jitendra Malik, Roberto Martín-Martín, Subramanian Ramamoorthy, Dorsa Sadigh, Shuran Song, Jiajun Wu, Michael C. Yip, Yuke Zhu, Thomas Kollar, Sergey Levine, and Chelsea Finn. Droid: A large-scale in-the-wild robot manipulation dataset, 2025. URL https://arxiv.org/abs/2403.12945.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Grace Lam, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. 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. In *Proceedings of Robotics: Science and Systems*, LosAngeles, CA, USA, June 2025a. doi: 10.15607/RSS.2025.XXI.017.

Taeyoung Kim, Jimin Lee, Myungkyu Koo, Dongyoung Kim, Kyungmin Lee, Changyeon Kim, Younggyo Seo, and Jinwoo Shin. Contrastive representation regularization for vision-language-action models, 2025b. URL https://arxiv.org/abs/2510.01711.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 12888–12900. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/li22n.html.

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 Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 19730–19742. PMLR, 23–29 Jul 2023a. URL https://proceedings.mlr.press/v202/li23q.html.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, 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. *arXiv preprint arXiv:2405.05941*, 2024.

Yifan Li, Yifan Du, Kun Zhou, Jinpeng Wang, Wayne Xin Zhao, and Ji-Rong Wen. Evaluating object hallucination in large vision-language models. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023b. URL https://openreview.net/forum?id=xozJw0kZXF.

Zhiqi Li, Guo Chen, Shilong Liu, Shihao Wang, Vibashan VS, Yishen Ji, Shiyi Lan, Hao Zhang, Yilin Zhao, Subhashree Radhakrishnan, Nadine Chang, Karan Sapra, Amala Sanjay Deshmukh, Tuomas Rintamaki, Matthieu Le, Ilia Karmanov, Lukas Voegtle, Philipp Fischer, De-An Huang, Timo Roman, Tong Lu, Jose M. Alvarez, Bryan Catanzaro, Jan Kautz, Andrew Tao, Guilin Liu, and Zhiding Yu. Eagle 2: Building post-training data strategies from scratch for frontier vision-language models, 2025. URL https://arxiv.org/abs/2501.14818.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. Libero: Benchmarking knowledge transfer for lifelong robot learning. *arXiv preprint arXiv:2306.03310*, 2023a.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916. Curran Associates, Inc., 2023b. doi: 10.52202/075280-1516. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/6dcf277ea32ce3288914faf369fe6de0-Paper-Conference.pdf.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2024. URL https://arxiv.org/abs/2310.03744.

Wufei Ma, Haoyu Chen, Guofeng Zhang, Yu-Cheng Chou, Jieneng Chen, Celso de Melo, and Alan Yuille. 3dsrbench: A comprehensive 3d spatial reasoning benchmark. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6924–6934, October 2025.

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 (RA-L)*, 7(3):7327–7334, 2022.

NVIDIA, Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi "Jim" Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, Joel Jang, Zhenyu Jiang, Jan Kautz, Kaushil Kundalia, Lawrence Lao, Zhiqi Li, Zongyu Lin, Kevin Lin, Guilin Liu, Edith Llontop, Loic Magne, Ajay Mandlekar, Avnish Narayan, Soroush Nasiriany, Scott Reed, You Liang Tan, Guanzhi Wang, Zu Wang, Jing Wang, Qi Wang, Jiannan Xiang, Yuqi Xie, Yinzhen Xu, Zhenjia Xu, Seonghyeon Ye, Zhiding Yu, Ao Zhang, Hao Zhang, Yizhou Zhao, Ruijie Zheng, and Yuke Zhu. Gr00t n1: An open foundation model for generalist humanoid robots, 2025. URL https://arxiv.org/abs/2503.14734.

Zhiliang Peng, Wenhui Wang, Li Dong, Yaru Hao, Shaohan Huang, Shuming Ma, and Furu Wei. Kosmos-2: Grounding multimodal large language models to the world. *arXiv preprint arXiv:2306.14824*, 2023.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. LAION-400M: Open dataset of clip-filtered 400 million image-text pairs, 2021. URL https://arxiv.org/abs/2111.02114.

Andreas Steiner, André Susano Pinto, Michael Tschannen, Daniel Keysers, Xiao Wang, Yonatan Bitton, Alexey Gritsenko, Matthias Minderer, Anthony Sherbondy, Shangbang Long, Siyang Qin, Reeve Ingle, Emanuele Bugliarello, Sahar Kazemzadeh, Thomas Mesnard, Ibrahim Alabdulmohsin, Lucas Beyer, and Xiaohua Zhai. Paligemma 2: A family of versatile vlms for transfer. *arXiv preprint arXiv:2412.03555*, 2024.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh,

Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 olmo 2 furious, 2025. URL https://arxiv.org/abs/2501.00656.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008. URL http://jmlr.org/papers/v9/vandermaaten08a.html.

Homer Walke, Kevin Black, Abraham Lee, Moo Jin Kim, Max Du, Chongyi Zheng, Tony Zhao, Philippe Hansen-Estruch, Quan Vuong, Andre He, Vivek Myers, Kuan Fang, Chelsea Finn, and Sergey Levine. Bridgedata v2: A dataset for robot learning at scale. In *Conference on Robot Learning (CoRL)*, 2023.

Jiayu Wang, Yifei Ming, Zhenmei Shi, Vibhav Vineet, Xin Wang, Yixuan Li, and Neel Joshi. Is a picture worth a thousand words? delving into spatial reasoning for vision language models. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.

Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org, 2020.

Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. Internvl3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025a.

Zengzhi Wang, Fan Zhou, Xuefeng Li, and Pengfei Liu. Octothinker: Mid-training incentivizes reinforcement learning scaling, 2025b. URL https://arxiv.org/abs/2506.20512.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Youguang Xing, Xu Luo, Junlin Xie, Lianli Gao, Heng Tao Shen, and Jingkuan Song. Shortcut learning in generalist robot policies: The role of dataset diversity and fragmentation. In *Conference on Robot Learning*, pp. 3239–3266, 2025.

Weiye Xu, Jiahao Wang, Weiyun Wang, Zhe Chen, Wengang Zhou, Aijun Yang, Lewei Lu, Houqiang Li, Xiaohua Wang, Xizhou Zhu, Wenhai Wang, Jifeng Dai, and Jinguo Zhu. Visulogic: A benchmark for evaluating visual reasoning in multi-modal large language models, 2025. URL https://arxiv.org/abs/2504.15279.

Ganlin Yang, Tianyi Zhang, Haoran Hao, Weiyun Wang, Yibin Liu, Dehui Wang, Guanzhou Chen, Zijian Cai, Junting Chen, Weijie Su, et al. Vlaser: Vision-language-action model with synergistic embodied reasoning. *arXiv preprint arXiv:2510.11027*, 2025.

Wentao Yuan, Jiafei Duan, Valts Blukis, Wilbert Pumacay, Ranjay Krishna, Adithyavairavan Murali, Arsalan Mousavian, and Dieter Fox. Robopoint: A vision-language model for spatial affordance prediction in robotics. 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. 4005–4020. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/yuan25c.html.

Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Jianke Zhang, Xiaoyu Chen, Qiuyue Wang, Mingsheng Li, Yanjiang Guo, Yucheng Hu, Jiajun Zhang, Shuai Bai, Junyang Lin, and Jianyu Chen. Vlm4vla: Revisiting vision-language-models in vision-language-action models. *arXiv preprint arXiv:2601.03309*, 2026.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Bangkok, Thailand, 2024. Association for Computational Linguistics. URL http://arxiv.org/abs/2403.13372.

Enshen Zhou, Jingkun An, Cheng Chi, Yi Han, Shanyu Rong, Chi Zhang, Pengwei Wang, Zhongyuan Wang, Tiejun Huang, Lu Sheng, et al. Roborefer: Towards spatial referring with reasoning in vision-language models for robotics. *arXiv preprint arXiv:2506.04308*, 2025.

Yifeng Zhu, Abhishek Joshi, Peter Stone, and Yuke Zhu. Viola: Imitation learning for vision-based manipulation with object proposal priors. *arXiv preprint arXiv:2210.11339*, 2022. doi: 10.48550/arXiv.2210.11339.

A Implementation Details for VLM Mid-training

We apply a simple-yet-effective learnable scoring function $f(\cdot)$ described in Section 4 as a single linear layer on top of the frozen VLM's last-token last-hidden-state representation, trained with binary cross-entropy loss. For proximity estimator training, we apply a batch size of 128. The training data are sampled in a balanced manner from the VLM candidate pool (negative sample) and the target VLA data (positive sample), as described in Section 5.1. Validation accuracy is computed on a held-out split of the union of VLM negatives and VLA positives. We apply early-stopping at $\geq 90\%$ validation accuracy, and experiments show that our binary classifier training usually stops at 75 to 100 steps. We then run data selection process with this proximity estimator on all VLM data to select a subset of top 1280k samples by proximity score for further VLM training.

In VLM mid-training, we leverage the LLaMA-Factory (Zheng et al., 2024) as the training framework. We perform full-parameter supervised fine-tuning on InternVL3.5-1B, unfreezing the vision encoder, multi-modal projector, and language model. Table 3 summarizes the key hyperparameters.

Table 3: Key hyperparameters for VLM mid-training.

Configuration	**Mid-training**
VLM backbone	InternVL3.5-1B
Trainable parameters	All (full fine-tuning)
Sequence length	1024
Optimizer	AdamW
Peak learning rate	5×10^{-5}
Learning rate schedule	Cosine decay
Warmup ratio	0.1
Numerical precision	bfloat16
Training steps	5,000
Global batch size	256

B Implementation Details for Alternative Proximity Measurements

We provide formal definitions and implementation details for the three alternative proximity measurements compared against our learnable proximity estimator in Section 6.1.

Feature-space Average Distance. For each candidate VLM sample x, we compute its average ℓ_2 distance to all VLA samples in the frozen VLM representation space:

$$d_{\text{avg}}(x) = \frac{1}{|\mathcal{D}_{\text{VLA}}|} \sum_{y \in \mathcal{D}_{\text{VLA}}} \|\phi(x) - \phi(y)\|_2 \,, \tag{6}$$

where $\phi(\cdot)$ denotes the last hidden state of the frozen VLM, following the same feature space used in Section 3. Samples with smaller $d_{\text{avg}}(x)$ are ranked as closer to the VLA distribution and selected first.

VLA-conditioned Perplexity. We fine-tune the VLM on converted in-domain VLA data in which robot actions are represented as text tokens, yielding a VLA-conditioned model with parameters θ_{VLA}. The perplexity of a candidate sample $x = (x_1, \ldots, x_T)$ under this model is:

$$\text{PPL}_{\text{VLA}}(x) = \exp\left(-\frac{1}{T} \sum_{t=1}^{T} \log p_{\theta_{\text{VLA}}}(x_t \mid x_{<t}) \right). \tag{7}$$

Samples with lower $\text{PPL}_{\text{VLA}}(x)$ are considered more compatible with the VLA domain and selected first.

Delta Perplexity. Delta perplexity measures the perplexity change induced by VLA fine-tuning:

$$\Delta\text{PPL}(x) = \text{PPL}_{\text{VLA}}(x) - \text{PPL}_{\text{VLM}}(x), \tag{8}$$

where $\mathrm{PPL}_{\mathrm{VLM}}(x)$ is the perplexity under the original pretrained VLM. A more negative $\Delta\mathrm{PPL}(x)$ indicates that the sample becomes more predictable after VLA adaptation, suggesting stronger alignment with the VLA domain. Samples are ranked by $\Delta\mathrm{PPL}(x)$ in ascending order for selection.

C Implementation Details for VLA Fine-tuning

Training Setup. We follow the VLM4VLA adaptation design and evaluation protocol (Zhang et al., 2026) for downstream VLA adaptation. Table 4 summarizes the key training settings for our InternVL3.5-1B-based VLA models. Across all experiments, models take a single-view image as visual input and do not use robot state information, and are fine-tuned end-to-end. Input images are first standardized to 224×224 and then resized to match the input resolution required by each VLM backbone. For InternVL3.5-1B, the final input resolution is 448×448.

For the baseline models reported in Table 1 including expert VLAs and off-the-shelf VLMs, we directly use the results from VLM4VLA.

Table 4: Key hyperparameters for VLA fine-tuning.

Configuration	**VLA Fine-tuning**
Trainable parameters	All (full fine-tuning)
Action chunk size	10 (Calvin ABC-D) 4 (Simpler-Bridge, Libero-10)
Optimizer	AdamW
Peak learning rate	5×10^{-5}
Learning rate schedule	Constant after linear warmup
Warmup steps	500
Weight decay	0
Gradient clipping	1.0
Numerical precision	bfloat16
Training steps	8,000 (Calvin ABC-D) 16,000 (Simpler-Bridge, Libero-10)
Global batch size	128 (Calvin ABC-D) 256 (Simpler-Bridge, Libero-10)

VLA Architecture. Following VLM4VLA (Zhang et al., 2026), we append a single learnable action-query embedding $\mathbf{q}$ to the image–instruction sequence. At timestep t, the current image I_t is encoded into visual tokens $\mathbf{V}_t$ by the VLM's vision encoder and multimodal projector. These tokens and the task instruction ℓ are arranged using the backbone-specific template $\mathcal{T}_m$. The final hidden state at the action-query position provides the control representation:

$$\mathbf{x}_t = [\mathcal{T}_m(\mathbf{V}_t, \ell); \mathbf{q}], \qquad \mathbf{h}_t = F_\theta(\mathbf{x}_t)_{\mathbf{q}} \in \mathbb{R}^d, \tag{9}$$

where F_θ is the VLM's language backbone and d is its hidden size. For InternVL3.5 and Qwen3VL, the image and instruction are placed in the user message, and the action query follows the assistant prefix.

A lightweight two-branch MLP decodes $\mathbf{h}_t$ into continuous arm actions and binary gripper states for each action chunk. We train the backbone, query embedding, and action head jointly, using Huber loss for normalized arm actions and binary cross-entropy on the gripper logits.

D Implementation Details for Real World Experiments

Hardware Setup. We collect demonstrations using a 3D Connexion SpaceMouse for teleoperation similar to the DROID (Khazatsky et al., 2025) set up, consisting of a 7-DoF Franka Research 3 robot with a Robotiq 2F-85 gripper, and an Intel Realsense D435i external camera. We collect 50 demonstrations for each of five tabletop manipulation tasks and evaluate with 10 rollouts per task.

Training Configuration. We adapt VLM4VLA's codebase on our real-world dataset, and the training configuration is provided in Table 5.

Table 5: Key hyperparameters for VLA fine-tuning on the Franka real-robot data.

Configuration	**Real-world Fine-tuning**
Trainable parameters	All (full fine-tuning)
Action chunk size	4
Action dimension	7 (3 transl. + 3 rot.-vec. + gripper)
Optimizer	AdamW
Peak learning rate	5×10^{-5}
Learning rate schedule	Constant after linear warmup
Warmup steps	500
Weight decay	0
Gradient clipping	1.0
Numerical precision	bfloat16
Training steps	2,000
Global batch size	256

Deployment Setting. We use DEOXYS CONTROL (Zhu et al., 2022) as the real-time control interface for executing VLA-predicted actions on the Franka robot arm. Individual actions from the predicted chunks are supplied at 20 Hz, while DEOXYS CONTROL performs low-level control within the robot's 1 kHz control loop.

Table 6: Text instructions used in the real-world evaluations. We collect 50 demonstrations per task and evaluate each task over 10 rollouts, with a limit of 300 steps per rollout. Representative executions are shown in Figures 8 and 9.

Task	**Text Description**
Cube to plate	put the red cube on the plate
Carrot to plate	put the carrot on the plate
Burger to plate	put the hamburger on the plate
Cube to left cup	put the red cube in the cup on the left
Stack cubes	stack the red cube on top of the green cube

E DETAILED EXPERIMENTAL RESULTS

Table 7 provides full per-scene breakdowns for SimplerEnv-Bridge, corresponding to the aggregate scores reported in Table 1. EmbodiedMidtrain improves most of the subtasks consistently.

Table 7: Detailed results on SimplerEnv-Bridge (per-scene success rates: Carrot, Eggplant, Spoon, Cube). * Results for Expert VLA Baselines and Off-the-shelf VLM Baselines are reproduced and reported by VLM4VLA.

Model	Size	Carrot	Eggplant	Spoon	Cube	Simpler↑
Expert VLA Baselines*						
OpenVLA (Llama-2)	7.7B	4.2	0.0	0.0	12.5	4.2
π_0 (Paligemma-1)	3.1B	62.5	100.0	54.2	25.0	60.4
Off-the-shelf VLM Baselines*						
Qwen2.5VL-3B	3.8B	20.8	91.7	79.2	0.0	48.0
Qwen2.5VL-7B	8.3B	12.5	100.0	75.0	0.0	46.8
Qwen3VL-4B	4.4B	54.2	95.8	75.0	0.0	56.3
Qwen3VL-8B	8.8B	58.3	95.8	79.2	0.0	58.3
Qwen3VL-30B-A3B	30B-A3B	29.2	79.2	70.8	0.0	44.8
Paligemma-1	2.9B	50.0	91.7	75.0	4.2	55.3
Paligemma-2	3.0B	75.0	75.0	79.2	0.0	57.3
KosMos-2	1.7B	37.5	100.0	75.0	29.2	60.4
VLM with *EmbodiedMidtrain* (Ours)						
Qwen3VL-2B	2.1B	37.5	75.0	41.7	0.0	38.5
+*EmbodiedMidtrain*	2.1B	41.7	95.8	45.8	0.0	45.8
InternVL3.5-1B	1.1B	0.0	100.0	45.8	0.0	36.5
+*EmbodiedMidtrain*	1.1B	20.8	91.7	70.8	41.7	**56.3**

F Selected Data Mixture Composition

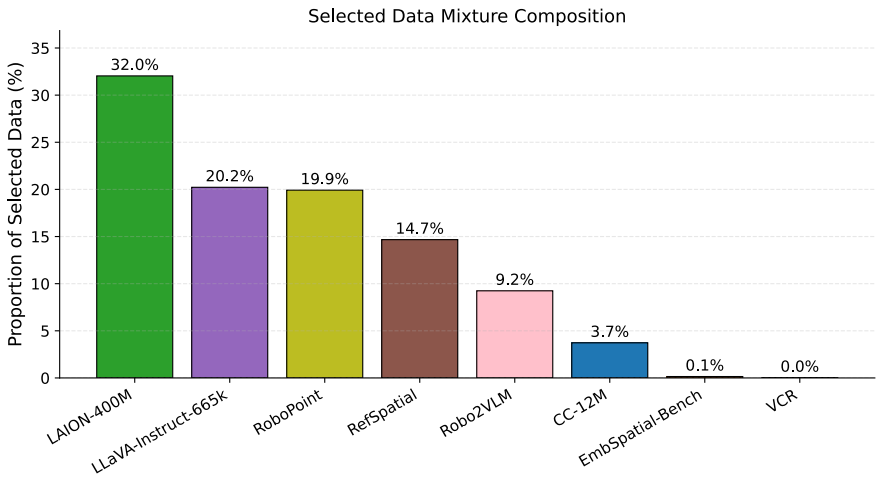


Figure 6: Composition of the selected mid-training data mixture after proximity-based selection, shown as the percentage of each VLM source dataset in the final mixture.

Figure 6 shows the composition of the mid-training mixture after proximity-based data selection. LAION-400M contributes the largest share, while EmbSpatial-Bench and VCR are nearly absent from the selected mixture.

F.1 Diversity preservation after data selection

We further verify that our proximity-based selection preserves the diversity of the original VLM dataset rather than collapsing the mid-training corpus onto a narrow region near the VLA domain. To quantify the diversity of a data subset, we follow Xing et al. (2025) to adopt the *uniformity* metric

Table 8: Diversity of VLM data, VLA data, and the VLM subset selected by EmbodiedMidtrain. The full VLM data pool comprises two disjoint subsets: general VLM data and embodied-oriented VLM data.

Dataset	Diversity
VLM data	1.85
General VLM Data	1.96
Embodied-oriented VLM Data	1.62
VLA data	1.26
Selected VLM data	1.93

Table 9: Performance of InternVL3.5-1B on VLM benchmarks before and after mid-training.

Model	POPE↑	BLINK↑	VisuLogic↑	3DSRBench↑	SpatialEval↑
InternVL3.5-1B	86.33	43.45	21.00	47.87	49.82
+*EmbodiedMidtrain*	86.29	40.45	24.90	49.51	48.00

proposed by Wang & Isola (2020). Formally, given a dataset D_i, we define its diversity as

$$S_{\text{diversity}}^{D_i} \triangleq \frac{1}{\mathbb{E}_{u,v\sim D_i}\left[e^{-t\|u-v\|_2^2}\right]}, \tag{10}$$

where u and v are feature representations drawn from D_i in the frozen VLM's last hidden state space (following Section 3), and t is a temperature parameter controlling the kernel bandwidth (we apply $t = 2$ in our calculation). A higher $S_{\text{diversity}}^{D_i}$ indicates greater spread among samples, reflecting higher diversity.

Table 8 reports the diversity scores. VLA data is the most concentrated, with a score of 1.26 that is consistent with the compact clusters observed in Section 3, while the general VLM pool is the most diverse at 1.96. Our selected subset reaches 1.93, closely matching the general VLM pool and substantially exceeding both the embodied-oriented subset at 1.62 and the VLA data. Proximity-based selection thus shifts the data distribution toward the VLA domain without collapsing it, retaining the broad visual and linguistic coverage that makes the mixture effective as a mid-training signal.

G VLM Capability After Mid-training

While our primary goal is to improve VLMs as initializations for downstream VLA adaptation, it is also important to examine how mid-training affects the original capability profile of the VLM itself. To this end, we evaluate the mid-trained model on VLM benchmarks including BLINK (Fu et al., 2024), POPE (Li et al., 2023b), VisuLogic (Xu et al., 2025), 3DSRBench (Ma et al., 2025) and SpatialEval (Wang et al., 2024a). Results are reported in Table 9.

Overall, mid-training largely preserves the original VLM capability profile while inducing a selective shift in performance. Performance remains nearly unchanged on POPE, improves on VisuLogic and 3DSRBench, and decreases moderately on BLINK and SpatialEval. This suggests that VLM mid-training does not uniformly maintain all original capabilities, but instead reorients the model toward skills that are more relevant to embodied downstream adaptation.

Detailed BLINK subtask results. We examine performance across BLINK subtasks to understand which capabilities change after EmbodiedMidtrain. Table 10 reports the five subtasks with the largest changes: two with declines and three with gains. The declines occur in tasks that require reasoning across multiple images or jointly presented image patches. These input formats are absent from our robotics fine-tuning data and rare in the original VLM pool, leaving few such examples in the selected mid-training data. This limited coverage may contribute to the observed declines. The three subtasks that improve (Spatial Relation, Counting, and Semantic Correspondence) involve spatial and object-level reasoning, consistent with the emphasis of our data selection. The results suggest a trade-off associated with the composition of the mid-training data, with the largest losses concentrated

in input formats that are uncommon in robotics. Our objective is to improve VLA initialization, rather than to improve performance uniformly across all general-purpose VLM tasks.

Table 10: Top-5 most affected BLINK sub-tasks before and after mid-training. The two largest drops occur on multi-image / image-mixing tasks, and the gains occur on spatial- and object-grounded tasks.

BLINK sub-task	**Pre**	**Post**	Δ
Visual Similarity	0.726	0.474	-0.252
Jigsaw	0.620	0.527	-0.093
Spatial Relation	0.678	0.706	$+0.028$
Counting	0.492	0.583	$+0.091$
Semantic Correspondence	0.230	0.345	$+0.115$

H Transferability across Target Tasks and VLA Action Heads

H.1 Transferability across target tasks

By construction, **no robot manipulation data used for downstream VLA fine-tuning are included in the VLM mid-training dataset**: the proximity classifier uses only a small subset of data from three simulation VLA datasets as target data for selection. Real-robot results in Section 5.4 show that our mid-training gains transfer from simulation to real-world tasks: the dataset selected using simulation data as the target consistently benefits downstream fine-tuning in the real world. In addition, we conduct leave-one-benchmark-out experiments in this section to provide further evidence of transferability across target tasks.

Leave-one-benchmark-out generalization. We train the proximity classifier using *only* the Bridge and Calvin VLA training data, holding LIBERO-10 entirely out of the selector. We then re-run sample selection, mid-train an InternVL3.5-1B with the selected mixture, and VLA-fine-tune on LIBERO. Table 11 reports the result: the held-out variant retains 95% of the full method's gain on LIBERO. This indicates that the selector identifies VLM samples useful for embodied tasks in general, rather than retrieving samples that resemble a specific evaluation benchmark.

H.2 Transferability across VLA action heads

In the main experiments, we follow VLM4VLA's architecture design using two-branch MLP action head. To demonstrate that the performance gain of EmbodiedMidtrain can transfer across different action head design, we additionally evaluate EmbodiedMidtrain with two qualitatively different action heads using a modular VLA training framework that supports multiple decoder heads: parallel decoding of OpenVLA-OFT (Kim et al., 2025a) style on SimplerEnv-Bridge, and flow matching of π_0 (Black et al., 2024) style on LIBERO-10. We evaluate both InternVL3.5-1B and Qwen3VL-2B on this transferability. Experiment result is shown in Table 12.

EmbodiedMidtrain consistently improves performance across VLA action heads for both VLM backbones. Together with the main results on VLM4VLA's two-branch MLP architecture in Table 1, these findings demonstrate that the performance gains from mid-training transfer across different VLA action heads.

I Compute Accounting and FLOP-Matched Comparison

Compute accounting. The mid-training stage costs approximately **60 GPU-hours** on NVIDIA L40S (5,000 steps at global batch size 256, full-parameter optimization). Each VLA fine-tuning run on the same hardware costs approximately **300 GPU-hours**. Mid-training therefore adds about 20% to a single VLA fine-tuning run. However, the mid-trained VLM checkpoint is a one-time cost that is amortized across multiple downstream VLA runs (different VLA datasets, different action heads, etc.). Across the experiments in this paper, the same mid-trained checkpoint is reused for more than ten downstream VLA runs, so the effective per-run overhead is much smaller than the 20% headline number suggests.

Table 11: Leave-one-benchmark-out generalization on LIBERO-10. All settings use InternVL3.5-1B. Excluding LIBERO from the selection target datasets retains most of the gain from mid-training.

Method	**Selection targets**	**LIBERO-10**
No mid-training	N/A	39.0
EmbodiedMidtrain	Bridge + CALVIN	53.4
EmbodiedMidtrain	Bridge + CALVIN + LIBERO	**54.2**

Table 12: Transfer across action heads. EmbodiedMidtrain consistently improves both parallel decoding and flow matching across two backbones, with gains of 4.0–12.5 points. These results suggest that its benefits extend beyond the two-branch MLP head used in the main experiments.

Setting	**Parallel decoding** SimplerEnv	**Flow matching** LIBERO-10
InternVL3.5-1B	36.5	88.4
+ *EmbodiedMidtrain*	**49.0**	**92.4**
Qwen3VL-2B	42.7	83.4
+ *EmbodiedMidtrain*	**50.0**	**87.6**

FLOP-matched comparison. To rule out the possibility that the gain is a compute artifact, we train an off-the-shelf InternVL3.5-1B baseline for additional VLA fine-tuning steps so that its total compute matches mid-training plus standard VLA fine-tuning. Table 13 reports the comparison. At identical total FLOPs, EmbodiedMidtrain substantially outperforms simply spending the additional budget on extended VLA fine-tuning ($+12.5$ on SimplerEnv, $+8.4$ on LIBERO-10). The gain is therefore not a compute artifact: at matched total compute, mid-training is the more effective use of additional budget.

Table 13: FLOP-matched comparison. The baseline is trained for extra VLA fine-tuning steps so that its total compute equals mid-training plus VLA fine-tuning. At identical total FLOPs, EmbodiedMidtrain substantially outperforms extended VLA fine-tuning.

Setting	**SimplerEnv**	**LIBERO-10**
InternVL3.5-1B (extra training to match total compute)	43.8	45.8
InternVL3.5-1B + *EmbodiedMidtrain*	**56.3**	**54.2**

J Proximity Selector Design Ablations

Selection size. We compare selection sizes $K \in \{320\,\mathrm{k}, 1.28\,\mathrm{M}, 2.56\,\mathrm{M}\}$ with mid-training on the full candidate pool, holding mid-training compute fixed. Table 14 shows gains over the off-the-shelf InternVL3.5-1B initialization for every selected subset. The default selection of $1.28\,\mathrm{M}$ examples yields the best performance on all three benchmarks, followed by $2.56\,\mathrm{M}$, as both of them outperform full-pool mid-training. The $320\,\mathrm{k}$ subset also outperforms the full pool on SimplerEnv and LIBERO-10, but falls below it on CALVIN.

Classifier stopping point. We stop training the proximity classifier when held-out classification accuracy reaches at least 90%, as described in Appendix A. To examine how training affects proximity scores, we score the full candidate pool using checkpoints with validation accuracies of approximately 70%, 90%, and 95%. Table 15 reports the within-source interquartile range (IQR) for three VLM sources and the fraction of all candidates with scores below 0.10. This threshold summarizes the score distribution. Data selection uses top-K ranking throughout.

The 70% checkpoint has the largest within-source IQRs and the smallest fraction of low-scoring candidates. Further training concentrates scores near zero: at 95% validation accuracy, 80.7% of candidates have scores below 0.10, and the IQR is smallest for all three sources. The default 90% checkpoint retains greater within-source score variation than the later checkpoint while satisfying the stopping criterion.

Table 14: Effect of selection size K on downstream VLA performance with InternVL3.5-1B. All mid-training variants use the same compute budget. Baseline denotes the model without mid-training. Full pool uses all candidate data without selection.

Benchmark	**Baseline**	**EmbodiedMidtrain** (K)			**Full pool**
		320 k	1.28 M	2.56 M	
CALVIN	3.173	3.211	**3.714**	3.455	3.398
SimplerEnv	36.5	47.9	**56.3**	51.0	43.8
LIBERO-10	39.0	48.6	**54.2**	50.4	48.4

Table 15: Proximity score statistics at three classifier checkpoints, indexed by validation accuracy. IQR denotes the interquartile range within each source. The final row reports the fraction of the full candidate pool with $s(x) < 0.10$.

Metric	**70%**	**90% (ours)**	**95%**
Score IQR, LAION-400M	0.293	0.158	0.042
Score IQR, LLaVA-Instruct	0.429	0.224	0.057
Score IQR, CC-12M	0.244	0.123	0.046
$P(s(x) < 0.10)$	21.0%	52.8%	80.7%

K Selection Granularity and Data Source Contributions

We evaluate the effects of selection granularity and candidate data sources using InternVL3.5-1B. Table 16 compares dataset-level selection, sample-level selection, and selection restricted to either general or embodied-oriented VLM sources. All mid-training variants use the same compute budget and hyperparameters.

Table 16: Effect of selection granularity and candidate data sources on downstream VLA performance with InternVL3.5-1B.

Setting	**Calvin**	**SimplerEnv**	**LIBERO-10**
InternVL3.5-1B (no mid-training)	3.173	36.5	39.0
+ Mid-training on the full pool (random selection)	3.398	43.8	48.4
+ Dataset-level selection (no sample filtering)	3.273	43.8	46.6
+ Sample-level selection within general VLM sources	3.636	49.0	51.8
+ Sample-level selection within embodied-oriented VLM sources	3.324	40.6	50.6
+ *EmbodiedMidtrain* (sample-level selection across all sources)	**3.714**	**56.3**	**54.2**

Dataset-level and sample-level selection. Dataset-level selection retains sources with high mean proximity scores without filtering individual samples and yields performance comparable to mid-training on the full pool. Sample-level selection improves over dataset-level selection by 0.441 in average task sequence length on Calvin and by 12.5 and 7.6 percentage points on SimplerEnv and LIBERO-10, respectively. These results support using individual proximity scores to identify useful samples within sources, beyond the information captured by dataset averages.

Contributions of general and embodied-oriented sources. Sample-level selection from general VLM sources (LAION-400M, CC-12M, LLaVA-Instruct, and VCR) outperforms selection from embodied-oriented sources (RefSpatial, RoboPoint, Robo2VLM, and EmbSpatial-Bench) on all three benchmarks. Selecting across both groups yields the best results, supporting the use of general and embodied-oriented data together for mid-training.

L Qualitative Examples

L.1 Simulation

Figure 7 shows representative rollouts on the SimplerEnv-Bridge *put carrot on plate* task. For both Qwen3VL-2B and InternVL3.5-1B, the VLA initialized from the off-the-shelf VLM fails to complete the task within the allotted horizon, while the variant initialized from an *EmbodiedMidtrain*-mid-trained VLM successfully grasps the carrot and transports it to the plate. These qualitative outcomes are consistent with the per-scene *Carrot* gains reported in Table 7, providing visual evidence that proximity-based mid-training yields backbones better suited to embodied control.

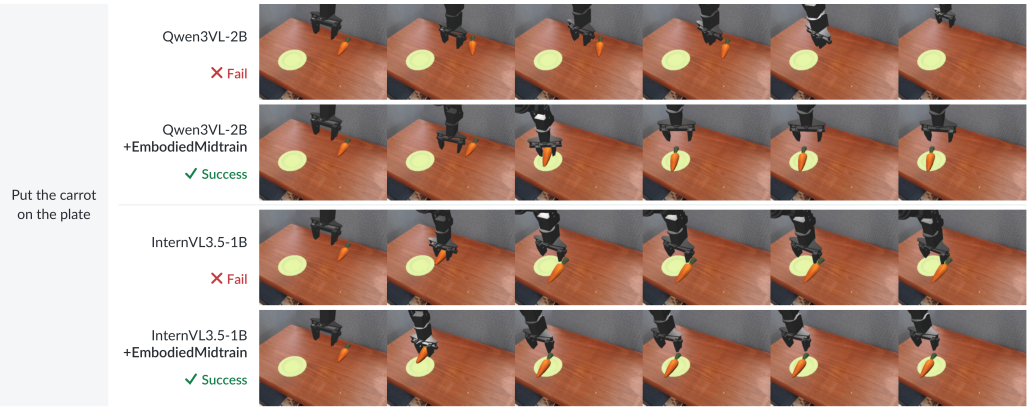


Figure 7: Qualitative rollouts on the SimplerEnv-Bridge *put carrot on plate* task. For each VLM backbone (Qwen3VL-2B and InternVL3.5-1B), we compare the VLA fine-tuned from the off-the-shelf VLM (top of each pair) against the same backbone after *EmbodiedMidtrain* (bottom of each pair). Without mid-training, both baselines fail to complete the task, whereas the mid-trained variants successfully grasp the carrot and place it on the plate.

L.2 Real-World Task

We examine the 50 evaluation rollouts from each initialization across the five tasks in Table 6. Figure 8 shows examples where *EmbodiedMidtrain* improves task completion.

Improvements in grasping and placement. In the carrot-to-plate example in Figure 8, the baseline initialized from InternVL3.5-1B reaches the carrot and closes its gripper too close to the edge, leaving the carrot on the table. With *EmbodiedMidtrain*, the robot successfully lifts the carrot and places it on the plate. Similarly, the baseline displaces the hamburger during its grasp attempt without delivering it to the plate, whereas the mid-trained model maintains its grasp through transport and placement. The cube-to-cup example illustrates a failure after a successful grasp: the baseline carries the cube near the cup but approaches at insufficient height, causing its gripper to push the cup toward the table edge. In the stacking example, the baseline moves the red cube toward the green cube without lifting it high enough to clear the nearby corn and fails to complete the stack. The mid-trained model lifts the red cube above the corn and successfully stacks it on the green cube.

Shared successes. Both models succeed in 90% runs of cube-to-plate task. In Figure 9(a), both policies grasp the red cube, lift it from the table, and release it onto the plate. Shared successes also occur on the carrot and hamburger tasks, where both models complete the same pick-and-place sequence.

Remaining failures. In the shared cube-to-plate failure in Figure 9(b), both models approach the cube but leave it on the table without establishing a successful grasp. A possible explanation is the cube's initial position near the edge of the camera view, where visual conditions may differ from those in the regions covered by most collected demonstrations.

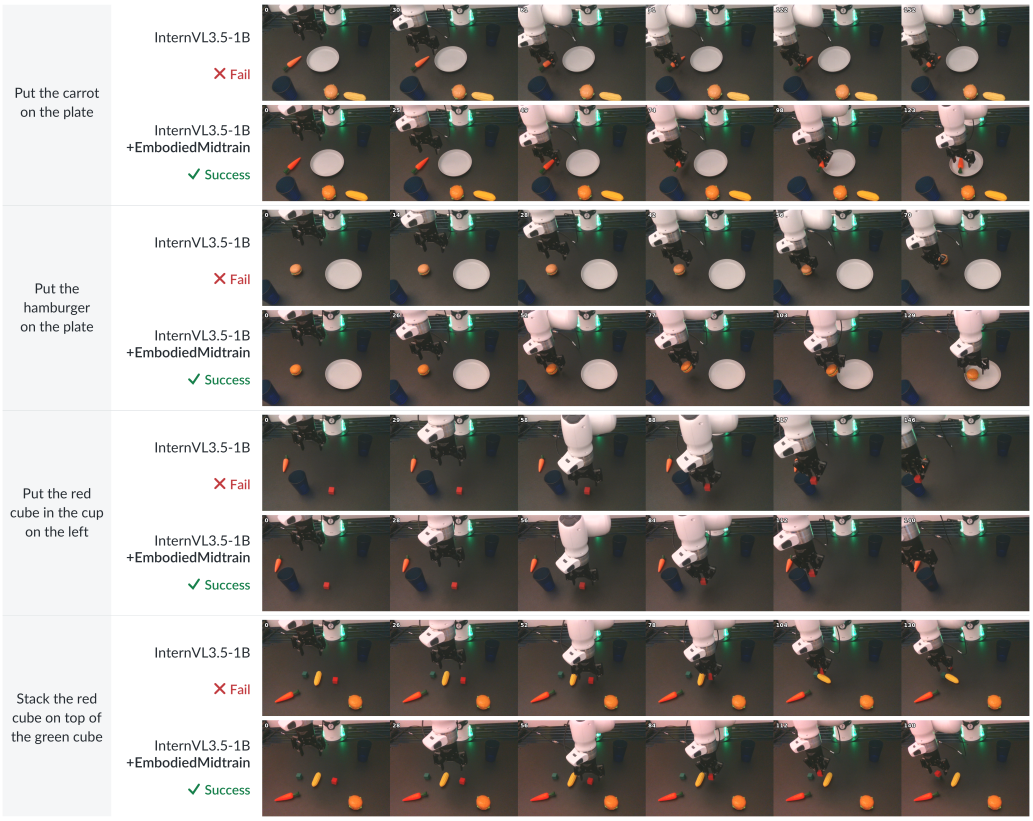


Figure 8: Representative real-world rollouts on InternVL3.5-1B. For each task, the top row shows a failed rollout from the off-the-shelf initialization, and the bottom row shows a successful rollout after *EmbodiedMidtrain*.

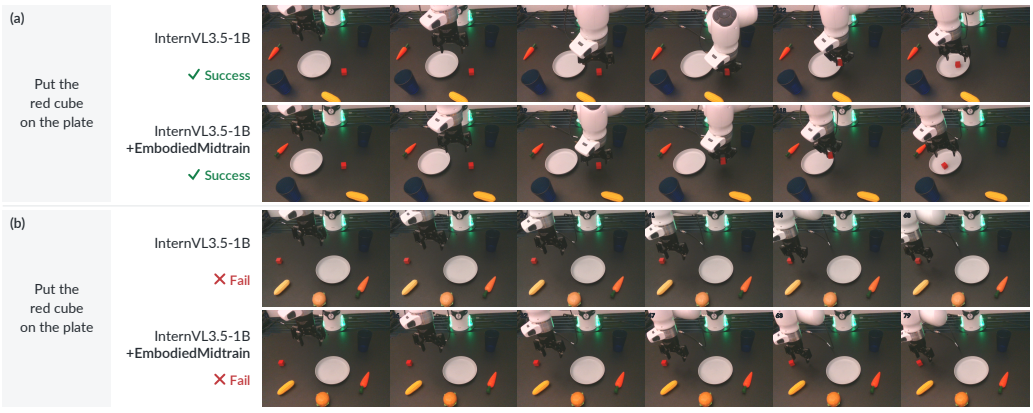


Figure 9: Shared successes and failures on the cube-to-plate task. (a) Both models grasp the red cube and place it on the plate. (b) Both approach the cube but leave it on the table. Frames are ordered chronologically within each row.