FAKEVLA: A PROBE FOR ANALYZING VISUAL REPRESENTATIONS FOR ROBOT ACTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision encoders are commonly evaluated frozen through downstream probes that reveal what information is encoded and how readily it can be used for prediction. In robotic manipulation, however, visual representations are typically evaluated within complete visuomotor policies, making it difficult to separate representation quality from downstream control. We propose FakeVLA, a framework for probing frozen visual representations through manipulation-specific tests that connect feature geometry to action selection. FakeVLA uses nearest-neighbor (NN) retrieval over demonstrations to test whether local visual neighborhoods recover manipulation-relevant physical states, finding that stronger state recovery is closely associated with higher Grasp and Place success. Further lightweight action predictors are used to test whether useful information remains decodable beyond local NN geometry. We evaluate generic, vision-language, and VLA-derived representations under controlled state and visual shifts, and compare probe trends with corresponding full-VLA policies whose vision modules remain frozen. Lower Drift is strongly associated with higher manipulation success, while texture causes the largest robustness degradation. Nonlinear predictors perform strongly under familiar observations but degrade more sharply under visual shifts than NN retrieval, showing that action decodability and local geometry capture distinct properties. Probe performance is also positively associated with full-VLA performance. Finally, strong within-domain performance does not imply sim-to-real alignment: cross-domain retrieval remains substantially weaker, although VLA pretraining and real-domain adaptation improve Sim→Real transfer.

1 INTRODUCTION

Learning-based robotic manipulation increasingly relies on large pretrained visual representations, including self-supervised encoders and vision-language models (VLMs) (Bjorck et al., 2025; Kim et al., 2024; Black et al., 2024; 2025). Yet, unlike in computer vision—where the usefulness of an encoder is routinely isolated and analyzed through downstream probes (Alain & Bengio, 2017; Kornblith et al., 2019; Chen et al., 2020; Grill et al., 2020; Oquab et al., 2024; Radford et al., 2021; Zhai et al., 2023)—robotic visual encoders are almost exclusively evaluated through the success rate of a complete visuomotor policy (Nair et al., 2022; Chi et al., 2023; Kim et al., 2024). This convention evaluates the representation indirectly, leaving the manipulation-relevant information intrinsically contained within the frozen visual features largely unexamined.

Research Gap. Standard VLA models map a visual observation I_t to an action a_t by composing a visual encoder f_θ with a policy π_ϕ:

$$I_t \xrightarrow{f_\theta} z_t \xrightarrow{\pi_\phi} a_t. \tag{1}$$

Evaluating systems purely on final action correctness tests the composition $\pi_\phi \circ f_\theta$, fundamentally entangling perception with action generation. Consequently, assessing a novel visual representation requires costly end-to-end policy optimization, and task failures cannot be cleanly attributed to perceptual deficits in f_θ versus capacity limitations in π_ϕ. This leaves a critical gap in understanding the intrinsic manipulation capabilities of different visual representations.

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.

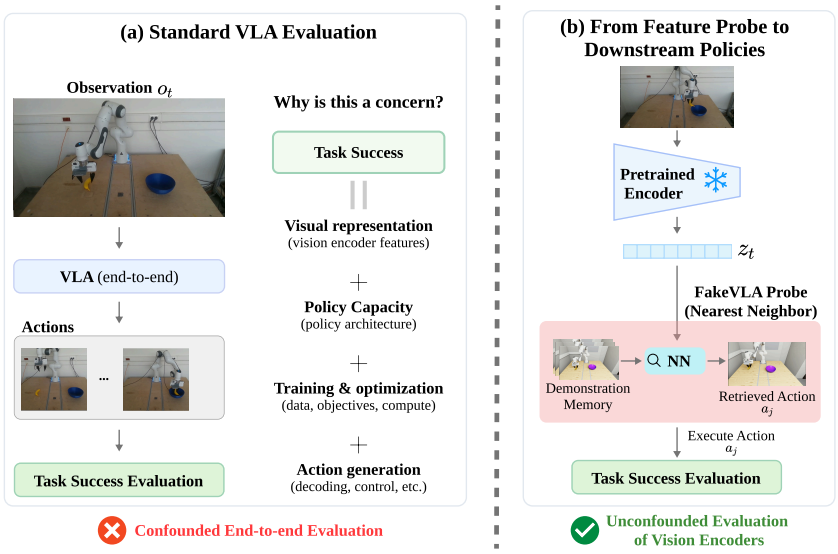


Figure 1: **Motivation for FakeVLA. (a)** Current VLA evaluations focus on task success rate, which conceals vision encoder's suitability for action estimation. **(b)** We propose FakeVLA as a visual probe that uses the visual encoder's visual features to retrieve the closest demonstration from an experience memory and selects the corresponding action.

Main Contributions. To address this gap, we introduce *FakeVLA*, a non-parametric probing framework that retrieves the nearest demonstration observation purely in visual feature space and reuses its associated action chunk. By applying this uniform retrieval mechanism across vision encoders, we eliminate downstream policy optimization from the comparison, making the manipulation-relevant local geometry of each representation directly observable.

To rigorously quantify this geometry, we introduce State–Vision Drift, a novel metric that measures the physical-state divergence between a neighbor retrieved via visual features and one retrieved via ground-truth physical state. Our framework provides concrete, quantifiable evaluations of: (1) spatial understanding for pick-and-place tasks, (2) representation capabilities across the sim-to-real gap, and (3) invariance against controlled environmental perturbations such as lighting, texture, and camera pose shifts.

Main Findings. Through extensive evaluations across off-the-shelf vision encoders, VLM-derived representations, and VLA visual encoders, FakeVLA reveals several properties of manipulation-relevant visual representations:

- **Local visual geometry tracks manipulation success.** StateNN–VisionNN Drift is strongly negatively correlated with both Grasp and Place success, showing that representations whose visual neighborhoods better preserve manipulation-relevant physical state also support more successful nearest-neighbor action retrieval. However, the absolute Drift scale varies across objects, making it most useful as a comparative diagnostic within a matched evaluation setting.
- **Manipulation relevance and visual robustness are distinct.** VLA-pretrained representations achieve the strongest average Grasp and Place performance, but this advantage is not uniform across distribution shifts. Texture produces the largest degradation across representations, while camera-pose perturbations substantially reduce the nominal advantage of VLA representations. Strong task performance therefore does not imply uniform invariance to visual changes.
- **Representation quality depends on how features are decoded.** Linear probes recover little useful action information, whereas nonlinear MLPs achieve strong performance under familiar observations but degrade sharply under visual perturbations. Nearest-neighbor FakeVLA remains more robust under these shifts, indicating that linear decodability, nonlinear action decodability, and local neighborhood structure capture different properties of the same frozen representation. Probe performance is moderately associated with corresponding full-VLA performance, making FakeVLA an informative diagnostic of downstream behavior alongside full-policy evaluation.

- **Strong within-domain geometry does not guarantee sim-to-real alignment.** Representations support high manipulation success when demonstration memory and execution remain within the same domain, but performance drops sharply for Sim→Real and Real→Sim retrieval. VLA pretraining provides a partial advantage in the Sim→Real direction, and real-domain adaptation further improves transfer, while simulation-only adaptation can strengthen in-domain performance without improving cross-domain compatibility.

2 Related Work

The full version of the related work is in the appendix section

Probing Visual Representations. Pretrained visual representations are commonly evaluated through frozen-feature probes, nearest-neighbor retrieval, and lightweight downstream predictors to assess the accessibility of task-relevant information. This paradigm has been applied to self-supervised encoders such as SimCLR (Chen et al., 2020), BYOL (Grill et al., 2020), and DINOv2 (Oquab et al., 2024), as well as robotic representations such as R3M (Nair et al., 2022) and VI-PRoM (Jing et al., 2023). Unlike recognition-oriented probes, robotic manipulation requires features that preserve physical-state distinctions relevant to action selection. We therefore evaluate frozen representations through manipulation-specific state correspondence and control.

Representation Geometry and Retrieval-Based Manipulation. Representation geometry can be examined through visual correspondence (Florence et al., 2018) and the recoverability of robot-related variables (Singh et al., 2026). Retrieval-based imitation provides a complementary behavioral probe: VINN (Pari et al., 2022) reuses actions associated with visually similar demonstrations, while subsequent methods extend retrieval-based control (Kwon et al., 2025; Pfeifer et al., 2026; Zhang et al., 2025). However, information accessible to a learned predictor need not be organized into neighborhoods suitable for direct action reuse. We therefore use Vision FakeVLA to probe local feature geometry through retrieval and physical-state correspondence, and compare its behavior with learned action prediction.

Vision-Language-Action Models and Frozen Visual Features. VLA models combine pretrained visual representations with learned robotic policies, differing in visual backbones and action generation (Table 1). Complete-policy evaluation entangles representation quality with policy architecture and optimization. We isolate visual representations through shared retrieval and learned-prediction settings, then examine whether the observed trends persist in full VLA control.

Table 1: Representative VLA models, their visual backbones, and action-generation mechanisms. For OpenVLA, OFT denotes the fine-tuning recipe used for the corresponding full policy; see Appendix B.

VLA Model	Visual Backbone	Action Generation
OpenVLA (Kim et al., 2024; 2025)	DINOv2 + SigLIP	Continuous action chunks (OFT)
$\pi_{0.5}$ (Black et al., 2025)	PaliGemma (SigLIP)	Flow matching
X-VLA (Zheng et al., 2026)	Florence-2-Large	Soft-prompted Transformer
GR00T N1.7 (Bjorck et al., 2025)	Cosmos-Reason2-2B (Qwen3-VL)	Flow-matching DiT

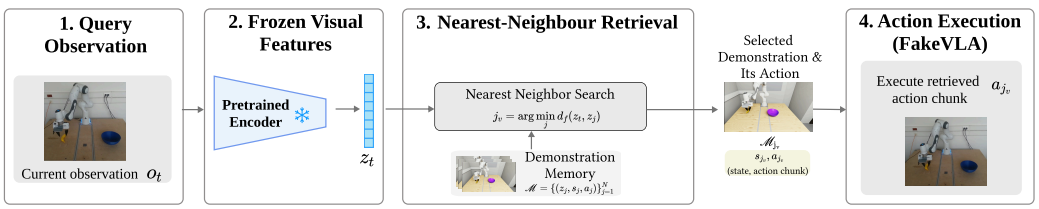


Figure 2: **An overview of FakeVLA.** FakeVLA is a visual probe that uses visual features of the current observation o_t to retrieve the visually closest demonstration o'_t from the demonstration memory. FakeVLA executes the action corresponding to the retrieved demonstration.

3 METHODOLOGY: FAKEVLA

FakeVLA probes manipulation-relevant information encoded in frozen visual representations independently of a separately trained action predictor. As illustrated in Figure 2, given a memory of robot demonstrations, Vision FakeVLA retrieves observations using visual features and executes their associated actions.

3.1 PROBLEM DEFINITION

We assume a dataset of K robot demonstrations, $\mathscr{D} = \{\tau_i\}_{i=1}^{K}$, where each trajectory $\tau_i = \{(I_{i,t}, s_{i,t}, a_{i,t})\}_{t=1}^{T_i}$ contains visual observations, corresponding physical states, and executed actions or action chunks. For a frozen visual encoder f_θ, we construct a demonstration memory from all demonstration observations:

$$\mathscr{M} = \left\{ (z_{i,t}, s_{i,t}, a_{i,t}) \;\middle|\; \begin{smallmatrix} i=1,\dots,K \\ t=1,\dots,T_i \end{smallmatrix} \right\}, \qquad z_{i,t} = f_\theta(I_{i,t}), \tag{2}$$

where $I_{i,t}$ is a visual observation, $z_{i,t}$ its representation, $s_{i,t}$ the corresponding physical state, and $a_{i,t}$ the associated action or action chunk. Our goal is to probe how effectively the visual representation supports action selection from the demonstration memory given a query image I.

3.2 FAKEVLA

Vision FakeVLA. Given demonstration memory $\mathscr{M}$, Vision FakeVLA retrieves the nearest demonstration to the current observation I_t in frozen visual feature space:

$$j^v(t) = \arg\min_j d_f(f_\theta(I_t), z_j), \tag{3}$$

where d_f is cosine distance. Features are ℓ_2-normalized and retrieved using FAISS inner-product search, which is equivalent to cosine nearest-neighbor search. The associated action $a_{j^v(t)}$ is executed and retrieval is repeated after each new observation. Nearest-neighbor control serves as a probe of whether local visual neighborhoods correspond to physically relevant states and actions, rather than as an assumed optimal policy.

State FakeVLA. For complementary analysis, State FakeVLA retrieves directly in physical state space. We define $s_t = [x_t, y_t, z_t, r_t, p_t, \psi_t, g_t]$, with end-effector position (x_t, y_t, z_t), roll–pitch–yaw (r_t, p_t, ψ_t), and binary gripper state $g_t \in \{0, 1\}$. It retrieves

$$j^s(t) = \arg\min_j d_s(s_t, s_j), \tag{4}$$

where d_s is the $L1$ distance between the physical states. Although the distance is computed over the full state, contributions from components other than Cartesian position (x, y, z) are negligible; we therefore report only the positional component.

Demonstration Memory. Demonstration memories are constructed from synchronized robot trajectories containing visual observations, physical states, and actions. Visual representations are computed offline using the encoder under evaluation, while the underlying demonstrations and retrieval procedure remain fixed across encoder comparisons.

4 EXPERIMENTS

We evaluate visual representations through three complementary research questions (RQs). **RQ1** asks whether frozen pretrained visual features organize observations according to manipulation-relevant physical state and remain robust under controlled distribution shifts. **RQ2** asks whether trends exposed by nearest-neighbor FakeVLA remain informative when the same frozen features are consumed by learned action predictors and complete VLA policies. **RQ3** examines how task-specific visual adaptation affects this structure within and across simulation and real-world domains.

4.1 Experimental Setup

Task and dataset. We collect 100 tabletop pick-and-place rollouts for each of the Apple and Banana objects in simulation, and an additional 100 Banana rollouts in the real world. Our simulation environment is built in Isaac Sim and closely replicates the real-world setup in terms of robot, workspace, camera configuration, and task geometry. In both domains, the pick and place locations remain fixed, while the robot's initial configuration is randomized across episodes. Each demonstration is saved as visual observations, robot states, and action sequences.

The simulation dataset is used as retrieval memory in RQ1; to train the Linear and MLP probes and adapt the non-visual components of the full VLAs in RQ2; and for VLA fine-tuning in RQ3. The visual encoder remains frozen throughout RQ2 and is made trainable for the adapted VLA variants in RQ3. The real-world dataset is used as retrieval memory and for VLA fine-tuning with a trainable visual encoder in RQ3.

Object selection. Apple and Banana were chosen to provide complementary object geometries commonly encountered in tabletop manipulation. Apple provides a compact, approximately isotropic shape, whereas Banana provides an elongated and curved geometry with a more direction-dependent grasp configuration. Evaluating both objects therefore tests whether representation behavior is consistent across substantially different geometric structures rather than being specific to a single object shape.

Visual encoders. We compare off-the-shelf representations from DINOv2, SigLIP, and DINOv2+SigLIP; VLM-derived representations from Prismatic and PaliGemma; and VLA-pretrained visual encoders from OpenVLA B, $\pi_{0.5}$, GR00T, and X-VLA B. A canonical feature readout is used for each encoder in the main comparison, while alternative pooling strategies and architectural variants are treated as ablations.

Environment Variables. To better evaluate robustness of vision encoders, we consider the following environmental factors: **Light Intensity**, **Light Color**, **Camera Pose**, and **Texture**. Light Intensity is varied over five levels $\{2000, 6000, 12000, 20000, 30000\}$, while Light Color is sampled from five distinct RGB illumination colors spanning red, blue, green, orange, and cyan/teal. Camera Pose is perturbed by up to ± 10 cm along the horizontal and vertical axes and $\pm 10^\circ$ about each rotation axis. Texture randomization changes the visual appearance of the table, stand, wall, plate, floor, and robot surfaces. These factors are varied independently to measure sensitivity to controlled appearance and viewpoint changes. We additionally consider two robot-initialization conditions without visual perturbations: **Dataset Init.**, where the robot is initialized from the $t = 0$ state of a demonstration episode, and **Unseen Init.**, where the initial robot state is not present in the demonstration dataset B.

4.2 Evaluation Metrics

We evaluate FakeVLA using three complementary metrics: **Grasp**, **Place**, and **StateNN–VisionNN Drift**.

Grasp and Place Success. Let grasp_i and place_i denote binary indicators of successful grasping and task completion in rollout i, respectively. Over N evaluation rollouts,

$$P_{\text{grasp}} = \frac{1}{N}\sum_{i=1}^{N} \text{grasp}_i, \qquad P_{\text{place}} = \frac{1}{N}\sum_{i=1}^{N} \text{place}_i. \tag{5}$$

Reporting both separates object-acquisition failures from failures occurring after a successful grasp.

StateNN–VisionNN Drift. To quantify the physical-state relevance of visual retrieval, we define StateNN–VisionNN Drift as the physical-state distance between the demonstrations selected by StateNN and VisionNN:

$$D_{\text{drift}}(t) = d^s\left(s_{j^s(t)}, s_{j^v(t)}\right), \tag{6}$$

with rollout-level Drift

$$\overline{D}_{\text{drift}} = \frac{1}{T}\sum_{t=1}^{T} D_{\text{drift}}(t). \tag{7}$$

Each rollout's Drift is averaged over all queried timesteps. The reported Drift for a condition is the mean of these rollout-level values across the corresponding evaluation trials. Lower Drift indicates stronger agreement between visual and physical-state neighborhoods. StateNN serves as a reference for physical-state correspondence rather than an oracle for the uniquely correct action; Drift therefore measures neighborhood agreement rather than manipulation quality in isolation.

4.3 Downstream Probes and Policies

Nearest-neighbor retrieval evaluates the local geometry of the frozen representation, but unsuccessful NN control does not imply that manipulation-relevant information is absent. We therefore evaluate the same features using a linear layer and an MLP, testing linear and nonlinear action decodability. Where available, we additionally evaluate the corresponding full VLA while keeping its visual encoder frozen. Together, these comparisons separate local neighborhood structure, learned action decodability, and complete-policy behavior. Each primary condition is evaluated over repeated rollouts. We report success rates (as defined in Section 4.2) with trial counts and use matched configurations for policy comparisons.

4.4 Exp. 1: Structure and Robustness of Frozen Representations (RQ1)

We first ask whether frozen pretrained visual representations organize observations according to manipulation-relevant state, and whether this structure remains useful under controlled distribution shifts. FakeVLA encodes the current observation with a frozen visual encoder, retrieves the nearest demonstration observation in feature space, and executes its associated action chunk.

We evaluate **Grasp**, **Place**, and **StateNN–VisionNN Drift** (**Drift**). Grasp and Place measure closed-loop pick-and-place success, while Drift measures the physical-state disagreement between the visually retrieved demonstration and the nearest demonstration in ground-truth state space. Lower Drift therefore indicates better alignment between visual neighborhoods and manipulation-relevant physical state.

We evaluate the frozen representations under the initialization and environmental conditions defined in the experimental setup. Full per-condition results are reported in Appendix C.

Results. Figure 3 shows that manipulation performance is closely related to the physical-state organization of the visual feature space. Across the 54 aggregated Apple encoder–condition observations (9 encoders $\times$ 6 conditions), StateNN–VisionNN Drift is strongly negatively correlated with Grasp ($r = -0.876$) and Place ($r = -0.805$), where r is the Pearson correlation coefficient; poorer agreement between visual and physical-state neighborhoods therefore tends to coincide with lower task success.

Robustness depends strongly on the distribution shift. Texture is the most disruptive condition: for Apple, mean Grasp falls from 86.7% to 19.6% and Place from 62.2% to 8.0%, while mean Drift increases from 54.1 mm to 241.7 mm. VLA-pretrained representations achieve the strongest average manipulation performance, although their nominal advantage is not preserved uniformly under camera-pose shift.

These results establish both local manipulation geometry and visual robustness are distinct properties of frozen representations, motivating Exp. 2, where we vary the action-decoding mechanism while keeping the representation fixed.

4.5 Exp. 2: From Representation Probes to Downstream Policies (RQ2)

Exp. 1 evaluates whether frozen representations organize observations into manipulation-relevant local neighborhoods. We next ask whether the same representations expose useful action information to different downstream decoders. We keep the visual representation frozen and compare four

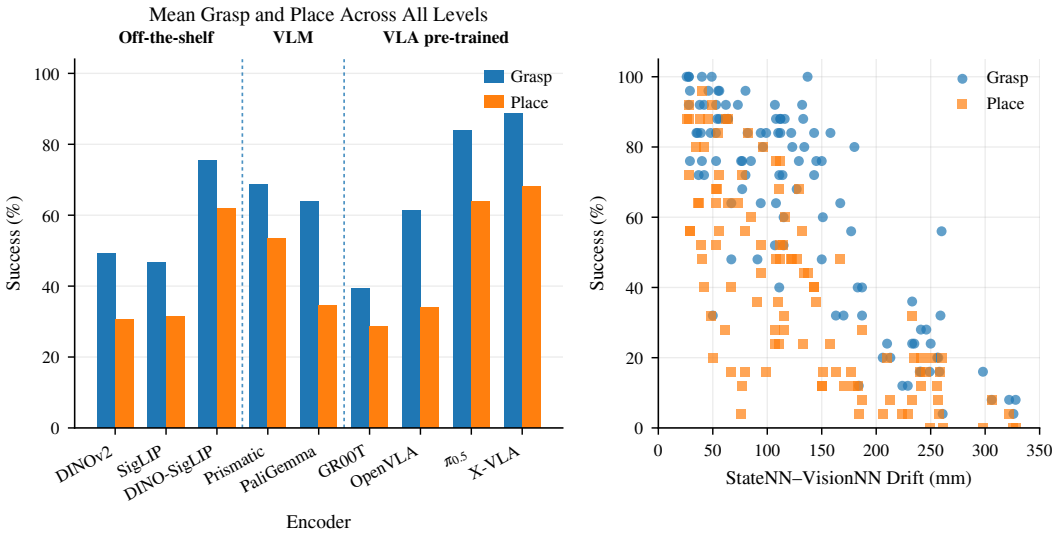
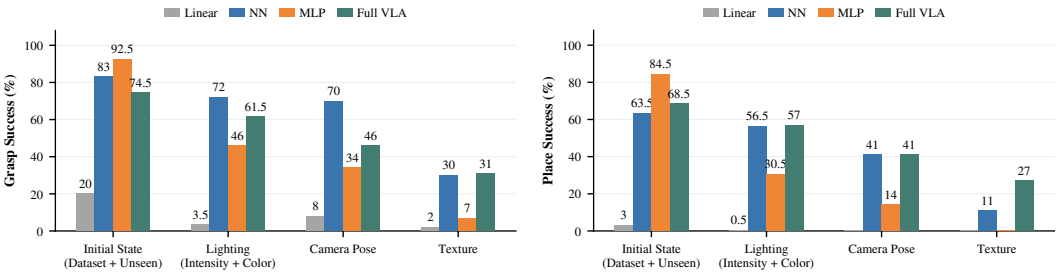


Figure 3: **Exp. 1: Overall evaluation of frozen representation quality.** Left: mean Grasp and Place success for Apple across the six evaluation conditions. Right: Grasp and Place success versus StateNN–VisionNN Drift for Apple and Banana. Lower Drift generally corresponds to higher task success, although the absolute Drift scale differs across objects. Full results are provided in Appendix Table 2.

action mappings: a **Linear** probe, nearest-neighbor **FakeVLA** (**NN**), a learned nonlinear **MLP**, and the corresponding **full VLA** with its base visual encoder frozen.

Linear, NN, and MLP policies provide complementary probes of the same frozen features. The linear probe measures whether actions can be recovered through a single global linear mapping, NN measures whether local feature-space neighborhoods directly support action reuse, and the MLP measures whether action-relevant information can be recovered through a learned nonlinear mapping. Full-VLA performance additionally incorporates the model-specific action-generation architecture and training.



(a) Grasp **(b) Place**

Figure 4: **Exp. 2: Comparison of action decoders over frozen visual representations.** Grasp and Place success are averaged over the four encoders with corresponding full-VLA evaluations. Dataset Init. and Unseen Init. are grouped as Initial State, while Light Intensity and Light Color are grouped as Lighting. Full results are provided in Appendix Table 3.

Results. Figure 4 shows that representation quality depends strongly on how frozen features are consumed. Linear probes perform poorly across nearly all conditions, while NN retrieval achieves substantially higher success, showing that useful local feature-space structure can exist without a successful global linear action mapping.

The MLP exposes more action-relevant information under familiar observations: over Dataset Init. and Unseen Init., it achieves 92.5% Grasp and 84.5% Place, compared with 83.0%/63.5% for NN.

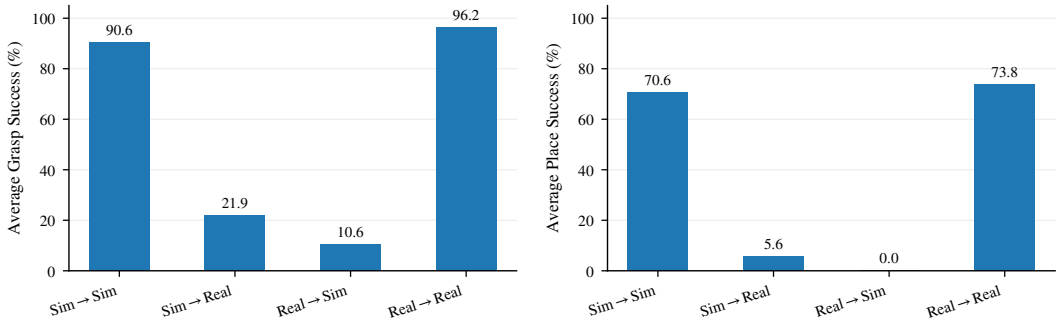
This advantage reverses under visual shifts: MLP Place falls to 18.8%, whereas NN retains 41.3% and the corresponding full VLAs 45.3%.

The lightweight probes nevertheless remain positively associated with full-VLA behavior. Across the six evaluation conditions for each of $\pi_{0.5}$, X-VLA, and GR00T (18 encoder–condition pairs), NN performance correlates with full-VLA performance at $r = 0.641$ for Grasp and $r = 0.627$ for Place. The corresponding MLP correlations are $r = 0.712$ and $r = 0.630$, respectively[1].

Thus, linear decodability, nonlinear action prediction, and local neighborhood structure expose different properties of the same frozen representation. Exp. 3 next asks whether this manipulation-relevant structure remains compatible across simulation and reality.

4.6 Exp. 3: Cross-Domain Representation Transfer and Adaptation (RQ3)

We finally ask whether manipulation-relevant visual geometry remains compatible across simulation and real-world domains. Unlike Exp. 1 and Exp. 2, we now explicitly modify the visual representation. For the supported VLA models, we compare the original pretrained visual encoder with variants fine-tuned on simulation or real-world demonstrations. Each representation is evaluated under four memory–query configurations: **Sim→Sim**, **Sim→Real**, **Real→Sim**, and **Real→Real**, where the first domain specifies the demonstration memory and the second specifies the query and execution domain. Full per-model results are reported in Appendix C Table 4.



(a) Grasp **(b) Place**

Figure 5: **Exp. 3: Within-domain and cross-domain retrieval performance.** Grasp and Place success are averaged over all evaluated representations and adaptations. The first domain denotes the demonstration-memory domain, while the second denotes the query and execution domain. Full results are provided in Appendix Table 4.

Results. Figure 5 reveals a large gap between within-domain and cross-domain retrieval. Sim→Sim achieves 90.6% Grasp and 70.6% Place, while Real→Real achieves 96.2%/73.8%. In contrast, Sim→Real falls to 21.9%/5.6%, and Real→Sim to 10.6% Grasp with no successful placements.

Among pretrained representations, generic/VLM features obtain 0% Grasp and Place under Sim→Real, whereas pretrained VLA representations average 30.0% Grasp and 5.0% Place, with X-VLA reaching 70%/20%. This advantage is directional: under Real→Sim, pretrained VLA representations average only 5.0% Grasp versus 15.0% for the generic/VLM group, and neither achieves successful placement. Adaptation further separates in-domain specialization from transfer. Real fine-tuning improves average Sim→Real performance from 30.0%/5.0% to 45.0%/15.0% Grasp/Place. Simulation fine-tuning instead raises Sim→Sim Place from 67.5% to 82.5%, while reducing Sim→Real performance to 12.5%/2.5%. Cross-domain Grasp remains more common than successful Place, showing that partial cross-domain correspondence does not necessarily support completion of the full pick-and-place task.

[1] r denotes the Pearson correlation coefficient; positive values indicate that higher probe performance tends to coincide with higher full-VLA performance.

5 DISCUSSION

Main Results

Across the three experiments, FakeVLA reveals complementary properties of frozen visual representations. First, local visual geometry is strongly associated with manipulation behavior 4.4: StateNN–VisionNN Drift correlates with both Grasp and Place, while severe visual shifts, particularly texture, simultaneously degrade task success and physical-state alignment. Second, representation usefulness depends on the downstream decoder 4.5: nonlinear MLPs recover strong action mappings under familiar observations but degrade much more under visual shifts, while NN retrieval provides a more direct measure of local representation geometry. Third, strong within-domain geometry does not imply cross-domain compatibility 4.6: both Sim→Sim and Real→Real succeed strongly while cross-domain retrieval remains substantially weaker.

Main Takeaways

Our results suggest that manipulation-relevant representation quality is multidimensional. Local physical-state geometry, action decodability, visual invariance, and cross-domain compatibility are related but distinct properties. Action-oriented pretraining generally improves manipulation performance, but does not guarantee robustness to viewpoint or appearance changes. This agrees with prior manipulation and VLA robustness studies that identify lighting, texture, and camera variation as important sources of failure (Burns et al., 2025; Fei et al., 2026; Garcia et al., 2023).

FakeVLA is therefore best viewed as a lightweight diagnostic rather than a replacement for complete-policy evaluation. Its nearest-neighbor probe disentangles learned action decoding from the comparison and directly exposes whether nearby visual observations correspond to manipulation states for which actions can be reused. The moderate positive relationship with full-VLA performance further suggests that this geometry is relevant downstream, although complete policy behavior also depends on the action architecture and training data.

Limitations

Our evaluation is limited to tabletop pick-and-place and two object geometries, with real-world experiments narrower than the simulation study. FakeVLA also intentionally evaluates one-neighbor action reuse; poor retrieval therefore does not imply that action-relevant information is absent, as demonstrated by the MLP results. Finally, family-level comparisons are descriptive because the evaluated encoders and VLAs differ in architecture, pretraining data, model scale, and downstream action generation.

6 CONCLUSION

We introduced FakeVLA, a non-parametric probe for evaluating whether frozen visual representations organize observations according to manipulation-relevant physical state. Across controlled simulation shifts, downstream action decoders, and sim-to-real transfer, our results show that representation quality cannot be captured by a single notion of task success. Local physical-state geometry strongly tracks manipulation behavior, but action decodability, visual robustness, and cross-domain compatibility remain distinct properties. VLA-pretrained representations generally provide stronger manipulation performance and improved Sim→Real retrieval, yet substantial sensitivity to visual and domain shifts remains. These results motivate evaluating visual representations directly, alongside complete policies, when designing robust visuomotor systems.

AI USE STATEMENT

In this work, we used generative AI tools for writing the code for the simulation environment. We have not used generative AI tools for generating core research ideas, formulating scientific claims, or conducting literature reviews, and synthetic data generation is not applicable to this work. Additionally, we used generative AI tools for polishing the text and improving overall readability. We have reviewed all AI-assisted work. Specifically, the AI-generated simulation code was manually verified and rigorously tested for correctness by the authors, and all AI-polished text was thoroughly proofread to ensure it accurately preserves our intended scientific meaning. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our study does not include any experiments with humans. We further report methodological limitations and potential failures to support responsible interpretation and use of our findings in Section 5.

REPRODUCIBILITY STATEMENT

The assumptions, algorithmic steps and experiment details are clearly described in the main text. The supplementary code contains the implementation and its dependency specification, and the configurations and scripts that reproduce Fake VLA NN, MLP and VLA experiments. All robotic demonstrations will be shared along with the code after acceptance.

REFERENCES

Guillaume Alain and Yoshua Bengio. Understanding intermediate layers using linear classifier probes. In *International Conference on Learning Representations (ICLR) Workshop Track*. OpenReview.net, 2017.

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. *arXiV: 2503.14734*, 2025.

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: 2410.24164*, 2024.

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

Kaylee Burns, Zach Witzel, Jubayer Ibn Hamid, Tianhe Yu, Chelsea Finn, and Karol Hausman. What makes pre-trained visual representations successful for robust manipulation? 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. 4525–4545. PMLR, 06–09 Nov 2025.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 2020.

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

Senyu Fei, Siyin Wang, Junhao Shi, Zihao Dai, Jikun Cai, Pengfang Qian, Li Ji, Xinzhe He, Shiduo Zhang, Zhaoye Fei, Jinlan Fu, Jingjing Gong, and Xipeng Qiu. Libero-plus: A progressive robustness benchmark for visual-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 38574–38583, June 2026.

Peter R. Florence, Lucas Manuelli, and Russ Tedrake. Dense object nets: Learning dense visual object descriptors by and for robotic manipulation. In *Proceedings of the 2nd Conference on Robot Learning (CoRL)*, volume 87 of *Proceedings of Machine Learning Research*, pp. 373–385. PMLR, 2018.

Ricardo Garcia, Robin Strudel, Shizhe Chen, Etienne Arlaud, Ivan Laptev, and Cordelia Schmid. Robust visual sim-to-real transfer for robotic manipulation. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 992–999, 2023. doi: 10.1109/IROS55552.2023.10342471.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Ya Jing, Xuelin Zhu, Xingbin Liu, Qie Sima, Taozheng Yang, Yunhai Feng, and Tao Kong. Exploring visual pre-training for robot manipulation: Datasets, models and methods. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 11390–11395. IEEE, 2023.

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. In *Proceedings of the 8th Conference on Robot Learning (CoRL)*, volume 270 of *Proceedings of Machine Learning Research*, pp. 2679–2713. PMLR, 2024.

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

Simon Kornblith, Jonathon Shlens, and Quoc V. Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2661–2671. Computer Vision Foundation / IEEE, 2019.

Owen Kwon, Abraham George, Alison Bartsch, and Amir Barati Farimani. RT-Cache: Training-free retrieval for real-time manipulation. In *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 1–8. IEEE, 2025.

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A universal visual representation for robot manipulation. In *Proceedings of the 6th Conference on Robot Learning (CoRL)*, volume 205 of *Proceedings of Machine Learning Research*, pp. 892–909. PMLR, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Hervé Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024, 2024.

Jyothish Pari, Nur Muhammad Shafiullah, Sridhar Pandian Arunachalam, and Lerrel Pinto. The surprising effectiveness of representation learning for visual imitation. In *Robotics: Science and Systems (RSS)*, 2022.

Quinn Pfeifer, Ethan Pronovost, Paarth Shah, Khimya Khetarpal, Siddhartha Srinivasa, and Abhishek Gupta. Difference-aware retrieval policies for imitation learning. In *International Conference on Learning Representations (ICLR)*, 2026.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021.

Mohit Shridhar, Lucas Manuelli, and Dieter Fox. CLIPort: What and where pathways for robotic manipulation. In *Proceedings of the 5th Conference on Robot Learning (CoRL)*, volume 164 of *Proceedings of Machine Learning Research*, pp. 894–906. PMLR, 2021.

Ishneet Sukhvinder Singh, Dhanoosh Pooranakumaran, Alex Nguyen, and Jia Qi Yip. Kepler-Encoder-v0.1: Towards a multimodal embedding model for robots. 2026.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, 2023.

Yuying Zhang, Wenyan Yang, Francesco Verdoja, Ville Kyrki, and Joni Pajarinen. ReMoBot: Retrieval-based few-shot imitation learning for mobile manipulation with vision foundation models. In *CoRL 2025 RemembeRL Workshop*, 2025.

Jinliang Zheng, Jianxiong Li, Zhihao Wang, Dongxiu Liu, Xirui Kang, Yuchun Feng, Yinan Zheng, Jiayin Zou, Yilun Chen, Jia Zeng, Ya-Qin Zhang, Jiangmiao Pang, Jingjing Liu, Tai Wang, and Xianyuan Zhan. X-VLA: Soft-prompted transformer as scalable cross-embodiment vision-language-action model. In *International Conference on Learning Representations (ICLR)*, pp. 60580–60606, 2026.

TABLE OF CONTENTS: APPENDIX

A Related Work: An Extended Discussion **13**

B Additional Experimental Details **14**

C Additional Experimental Results **16**

A RELATED WORK: AN EXTENDED DISCUSSION

Probing Visual Representations. A common practice in visual representation learning is to evaluate pretrained encoders through constrained downstream probes rather than solely through their pretraining objectives. This approach has been adopted across supervised, self-supervised, and vision-language pretraining to assess the transferability and task-relevant information encoded in frozen visual features. Methods such as SimCLR (Chen et al., 2020), BYOL (Grill et al., 2020), and DINOv2 (Oquab et al., 2024) learn visual representations without task-specific supervision and are commonly evaluated using frozen features, nearest-neighbor retrieval, or lightweight downstream predictors. Such evaluations ask not only whether information is present in a representation, but also how readily it can be accessed by a simple downstream mechanism.

Robotic manipulation increasingly relies on the same class of pretrained representations. R3M (Nair et al., 2022) learns visual features from large-scale human video and evaluates them as frozen inputs to downstream robot policies, while VI-PRoM (Jing et al., 2023) compares visual pretraining strategies under controlled manipulation evaluation. Vision-language models such as CLIP (Radford et al., 2021) and SigLIP (Zhai et al., 2023) provide another source of transferable features, and systems such as CLIPort (Shridhar et al., 2021) combine pretrained semantic features with spatial mechanisms designed for manipulation.

The relevant prediction target in manipulation differs from standard recognition probes. A useful representation must preserve distinctions required for selecting actions as the robot, object configuration, gripper state, and task progress change. Consequently, semantic separability alone does not establish that the representation is organized appropriately for control. Our work adopts the probing perspective of representation learning but evaluates frozen visual features through manipulation-specific targets: physical-state correspondence and action selection.

Representation Geometry and Retrieval-Based Manipulation. One way to evaluate a representation is to ask whether proximity in feature space corresponds to similarity in an underlying task-relevant space. Visual correspondence methods such as Dense Object Nets (Florence et al., 2018) directly evaluate geometric correspondence, while recent probing approaches such as Kepler (Singh et al., 2026) examine whether robot-related variables can be recovered from learned visual representations. These approaches test what information is accessible from visual features, but do not necessarily determine whether the local geometry of the representation supports action selection.

Retrieval-based imitation provides a direct behavioral test of this geometry. VINN (Pari et al., 2022) retrieves visually similar demonstration observations and reuses their associated actions, showing that pretrained visual neighborhoods can support manipulation without learning a fully parametric policy. RT-Cache (Kwon et al., 2025) extends retrieval-based control to larger memories, while methods such as DARP (Pfeifer et al., 2026) and ReMoBot (Zhang et al., 2025) combine retrieval with learned adaptation or trajectory reasoning.

These methods use retrieval primarily as a control or imitation mechanism: the objective is to select useful demonstrations and improve task execution. Our goal is different. We use retrieval as a diagnostic of the visual representation itself. In particular, Vision FakeVLA probes an encoder's ability to extract and organize manipulation-relevant physical state from visual observations by asking whether nearby features correspond to nearby physical states that admit similar actions. This distinction matters because a representation may contain action-relevant information without organizing it so that a single visual nearest neighbor yields an appropriate action, while strong local neighborhood structure may support direct action reuse even without a learned predictor. FakeVLA therefore fixes the retrieval rule across encoders and treats nearest-neighbor control as a controlled

probe rather than a complete manipulation policy. We quantify this local physical-state geometry by comparing the state retrieved in visual feature space with a state-space nearest-neighbor reference, and separately evaluate the same frozen features with learned action predictors to distinguish local neighborhood usability from broader action decodability.

Vision-Language-Action Models and Frozen Visual Features. Vision-language-action (VLA) models integrate pretrained visual or vision-language representations with policies trained on robot demonstrations, but differ substantially in both their visual backbones and action-generation mechanisms. OpenVLA uses a fused DINOv2+SigLIP representation with continuous action chunks (Kim et al., 2025), while $\pi_{0.5}$ uses PaliGemma with a SigLIP visual backbone and generate actions through flow matching (Black et al., 2024; 2025). X-VLA instead builds on Florence-2-Large and uses a soft-prompted Transformer for action generation (Zheng et al., 2026), whereas GR00T N1.7 uses Cosmos-Reason2-2B, based on Qwen3-VL, together with a flow-matching DiT action model (Bjorck et al., 2025).

These architectural differences make complete-policy comparisons difficult to interpret as comparisons of visual representation quality: performance may reflect the visual backbone, multimodal processing, action decoder, optimization procedure, or their interaction. This motivates evaluating the visual representations under a shared downstream setting.

Our evaluation therefore freezes the visual representations and compares them using the same nearest-neighbor retrieval mechanism and learned action predictors. We additionally evaluate the corresponding complete VLA policies on the same manipulation task to examine whether trends observed through controlled representation probes persist in end-to-end control.

B Additional Experimental Details

Simulation and communication stack. Our simulation environment is implemented in Isaac Sim 5.1. Simulation data collection and policy inference use ROS 2 as the communication layer between the environment, inference pipeline, and robot control components. The simulated setup is designed to reproduce the corresponding real-world robot, workspace, camera configuration, and task geometry as closely as possible.

Pretrained visual representations. For OpenVLA, we initialize from the OpenVLA-7B checkpoint. For the off-the-shelf encoders, we use `dinov2_vitl14_reg` for DINOv2 and `siglip-so400m-patch14-224` for SigLIP; the fused DINOv2+SigLIP representation uses these same two checkpoints. For $\pi_{0.5}$, we use the PyTorch-converted checkpoint originating from the Physical Intelligence JAX release. X-VLA is initialized from `2toINF/X-VLA-Pt`, and GR00T from the `nvidia/GR00T-N1.7-3B`.

For NN retrieval and the Linear and MLP probes, we use the CLS-token output for DINOv2, the pooled representation for SigLIP, and mean-pooled patch features for the remaining encoders. The same feature readout is used consistently across these probes for a given encoder.

Probe training and VLA adaptation. The Linear and MLP probes are trained using the same optimization setup. The MLP uses a three-layer architecture with GELU activations and LayerNorm, while the Linear probe consists of a single linear layer. Both are trained with AdamW using a learning rate of 1×10^{-4} and a batch size of 64 for up to 30 epochs, with early stopping based on validation loss and a patience of five epochs.

For VLA fine-tuning, we use a batch size of 8, a learning rate of 2×10^{-4}, and LoRA with rank 8, while otherwise retaining the default optimization settings provided by each model repository. Fine-tuning typically converges within approximately 4–6 epochs. The same training configuration is used for the VLA evaluations in RQ2, where LoRA is applied only to the non-visual components and the visual encoder remains frozen. In RQ3, the visual encoder is instead made trainable to study the effect of domain-specific visual adaptation.

OpenVLA terminology. We refer to the visual representation as OpenVLA throughout. OpenVLA-OFT denotes the fine-tuning procedure used to adapt the corresponding VLA policy;

it does not introduce a different visual backbone. Accordingly, adapted representations are reported as OpenVLA with the corresponding sim- or real-domain OFT adaptation.

Evaluation protocol. A fixed random seed governs the simulation setup for reproducibility. Within each evaluation condition, the robot's initial configuration is independently resampled for every rollout from the prescribed initialization distribution. Thus, the R rollouts correspond to distinct episode-level draws rather than repeated executions from a single initial state. The same seeded evaluation sequence is used across encoders and policies to provide matched comparison conditions. In Exp. 1 and Exp. 2, each evaluation condition is tested over $R = 25$ rollouts, while in Exp. 3 each domain-transfer configuration is evaluated over $R = 10$ rollouts. Real-world experiments are conducted under a consistent physical setup, including the same robot, camera configuration, workspace, and lighting conditions across evaluations, unless the corresponding experimental condition explicitly changes one of these factors.

Automated data collection. Simulation demonstrations are collected automatically without human intervention. Object poses are obtained directly from the simulator and are used to generate the required end-effector motions for each pick-and-place episode. Real-world collection follows the same high-level procedure: the object pose is recorded automatically, the robot executes the corresponding pick-and-place trajectory, and the robot performs the reset required for the subsequent episode. Thus, human intervention is not required between individual data-collection episodes.

Initialization diversity and overlap with the demonstration memory. The evaluation conditions are designed to distinguish retrieval from identical or near-identical observations from retrieval under genuinely unseen states. Dataset Init. (Level 1) intentionally starts from robot states present in the demonstration data and therefore represents the highest-overlap setting. In Unseen Init. (Level 2), the robot is initialized from joint configurations not present in the demonstration trajectories. Because the demonstration data occupy a restricted set of joint-space configurations, these independently sampled initial states have low likelihood of exactly overlapping with a stored demonstration state. From Level 3 onward, the same state-space variation is combined with explicit visual perturbations such as lighting, camera-pose, or texture changes, making exact observation matches with the demonstration memory even less likely. Thus, performance beyond Dataset Init. cannot be explained simply by retrieving an identical training image or state.

C Additional Experimental Results

This section provides the complete experimental results corresponding to the analyses presented in the main paper. The tables report the full per-encoder and per-condition results for the evaluated visual representations and downstream policies, complementing the aggregated results and figures discussed in the main text. We include these detailed results to facilitate comparison across experimental conditions and improve transparency and reproducibility.

Table 2: **Exp. 1: Frozen visual representations under controlled simulation shifts.** Grasp (G) and Place (P) are reported in %, and StateNN–VisionNN Drift (D) in mm.

	Vision Encoder	Dataset Init.			Unseen Init.			Light Int.			Light Color			Camera Pose			Texture		
		G	P	D	G	P	D	G	P	D	G	P	D	G	P	D	G	P	D
Apple	*Off-the-shelf Vision Encoders*																		
	DINOv2	76	48	40	72	40	42	64	40	67	32	20	50	48	36	91	4	0	261
	SigLIP	72	56	80	76	60	85	64	44	94	32	12	170	32	16	163	4	0	326
	DINO-SigLIP	92	72	28	84	80	35	84	52	39	84	64	36	92	88	38	16	16	240
	Vision Encoders of VLMs																		
	Prismatic	76	56	29	84	64	53	72	64	37	76	52	53	92	80	42	12	4	229
	PaliGemma	96	56	29	84	32	48	40	24	111	92	64	73	52	24	107	20	8	257
	Vision Encoders of VLAs																		
	OpenVLA	84	40	143	60	28	115	52	32	115	80	44	134	72	52	114	20	8	213
	$\pi_{0.5}$	96	72	56	96	88	46	88	64	64	100	96	40	100	44	137	24	20	250
	X-VLA	100	92	28	76	72	77	100	88	28	100	88	26	88	56	56	68	12	77
	Gr00t	88	68	54	64	48	108	40	28	187	16	12	241	20	12	256	8	4	322
Banana	*Off-the-shelf Vision Encoders*																		
	DINOv2	72	40	143	84	24	158	88	60	116	88	52	112	28	20	241	8	8	306
	SigLIP	76	36	145	88	24	133	80	12	180	56	20	260	56	16	177	8	0	328
	DINO-SigLIP	84	16	99	96	16	80	76	12	150	92	68	53	84	52	94	32	8	187
	Vision Encoders of VLMs																		
	Prismatic	24	20	210	24	8	233	16	0	249	12	4	184	12	4	224	20	4	206
	PaliGemma	92	28	107	84	48	122	64	48	167	80	48	123	84	36	110	32	16	259
	Vision Encoders of VLAs																		
	OpenVLA	48	16	67	88	28	61	76	4	76	48	12	150	60	16	151	40	12	183
	$\pi_{0.5}$	96	84	55	100	92	49	88	76	108	84	76	112	92	56	132	16	0	298
	X-VLA	88	88	64	80	80	96	92	88	62	84	84	82	68	48	127	88	48	112
	Gr00t	72	68	111	76	68	129	36	32	233	24	20	235	28	16	246	16	4	258

Table 3: **Exp. 2: From frozen visual representations to downstream action prediction.** Apple results. The visual encoder is frozen within each comparison. Grasp (G) and Place (P) are reported in %. NN denotes nearest-neighbor retrieval, Linear denotes a linear action probe, MLP denotes a nonlinear action predictor, and VLA denotes the corresponding full vision-language-action model. "N/A" indicates that no corresponding full VLA is included in the comparison.

Vision Encoder	Policy	Dataset Init.		Unseen Init.		Light Int.		Light Color		Camera Pose		Texture	
		G	P	G	P	G	P	G	P	G	P	G	P
DINOv2	NN	76	48	72	40	64	40	32	20	48	36	4	0
	Linear	84	0	88	16	28	0	28	0	24	0	0	0
	MLP	84	76	96	88	48	44	40	40	12	4	4	0
SigLIP	NN	72	56	76	60	64	44	32	12	32	16	4	0
	Linear	28	0	44	4	4	0	0	0	0	0	0	0
	MLP	88	76	80	60	32	24	4	0	24	0	0	0
DINO-SigLIP	NN	92	72	84	80	84	52	84	64	92	88	16	16
	Linear	32	0	28	0	16	0	0	0	0	0	0	0
	MLP	72	64	96	72	40	12	48	36	4	0	0	0
Prismatic	NN	76	56	84	64	72	64	76	52	92	80	12	4
	Linear	24	4	20	0	4	0	0	0	0	0	0	0
	MLP	88	60	84	20	40	8	64	40	12	8	0	0
PaliGemma	NN	96	56	84	32	40	24	92	64	52	24	20	8
	Linear	24	0	40	0	0	0	0	0	20	0	4	0
	MLP	100	84	100	96	32	32	8	4	32	16	0	0
OpenVLA	NN	84	40	60	28	52	32	80	44	72	52	20	8
	Linear	48	4	60	16	12	4	0	0	16	0	0	0
	MLP	80	80	68	60	48	32	40	28	48	28	0	0
	VLA	100	100	88	88	76	68	92	88	68	64	0	0
$\pi_{0.5}$	NN	96	72	96	88	88	64	100	96	100	44	24	20
	Linear	16	0	20	0	4	0	0	0	4	0	8	0
	MLP	100	100	100	96	72	56	28	12	24	20	0	0
	VLA	80	76	44	36	60	56	48	44	52	48	44	32
X-VLA	NN	100	92	76	72	100	88	100	88	88	56	68	12
	Linear	0	0	8	4	8	0	0	0	12	0	0	0
	MLP	100	96	96	96	92	52	52	48	40	4	28	0
	VLA	80	76	80	72	76	76	60	56	32	32	60	60
GR00T	NN	88	68	64	48	40	28	16	12	20	12	8	4
	Linear	4	0	4	0	0	0	4	0	0	0	0	0
	MLP	96	60	100	88	36	16	0	0	24	4	0	0
	VLA	76	60	48	40	36	28	44	40	32	20	20	16

Table 4: **Cross-domain evaluation of pretrained and adapted visual representations.** Grasp (G) and Place (P) are success rates (%). The first domain denotes the demonstration-memory domain and the second denotes the query and execution domain.

Encoder	Adaptation	Sim→Sim		Sim→Real		Real→Sim		Real→Real	
		G	P	G	P	G	P	G	P
DINOv2	Pretrained	80	40	0	0	0	0	100	70
SigLIP	Pretrained	90	40	0	0	10	0	90	90
DINOv2+SigLIP	Pretrained	80	30	0	0	50	0	100	100
PaliGemma	Pretrained	100	40	0	0	0	0	90	90
OpenVLA	Pretrained	90	80	0	0	10	0	100	100
OpenVLA	Sim-adapted (OFT)	100	100	0	0	10	0	100	90
OpenVLA	Real-adapted (OFT)	100	100	10	0	30	0	100	70
X-VLA	Pretrained	90	80	70	20	10	0	90	70
X-VLA	Sim fine-tuned	80	80	50	10	10	0	100	90
X-VLA	Real fine-tuned	100	100	80	40	10	0	100	100
GR00T	Pretrained	80	60	20	0	0	0	100	100
GR00T	Sim fine-tuned	70	60	0	0	0	0	90	90
GR00T	Real fine-tuned	100	80	40	10	20	0	100	100
$\pi_{0.5}$	Pretrained	100	50	30	0	0	0	100	0
$\pi_{0.5}$	Sim fine-tuned	90	90	0	0	0	0	80	0
$\pi_{0.5}$	Real fine-tuned	100	100	50	10	10	0	100	20

G: Grasp success (%), P: Place success (%). Fine-tuned VLA models are additionally evaluated in their respective fine-tuning domains, where they retain full task success.