SEGMENTATION-GUIDED SEMANTIC DISTILLATION FOR ROBOT MANIPULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Pretrained Vision Transformer (ViT) features may contain high-response background tokens and exhibit spatial-semantic misalignment, which can weaken the spatial perception of Vision-Language-Action (VLA) models and thereby impede spatially precise robotic control. Action-supervised adaptation does not reliably remove these artifacts. Therefore, we propose Segmentation-Guided Semantic Distillation (SGSD), which transfers dense features from a CNN segmentation teacher trained on robotic manipulation data directly into the VLA visual backbone. During policy fine-tuning, three auxiliary losses, namely spatially corresponding feature matching, directional alignment, and magnitude alignment, are jointly optimized with the action prediction loss. On five RoboTwin tasks and three real-world robot-arm tasks, SGSD improves success rates over frozen-backbone baselines by 17.5 and 18.7 percentage points, respectively. These results show that task-matched spatial distillation improves both foreground-aligned representations and manipulation success. Code and models are available at: `https://anonymous.4open.science/r/SGSD-Segmentation-Guided-Semantic-Distillation-0760`.

1 INTRODUCTION

Vision-language-action models (VLAs) are now a key paradigm in robotic manipulation. They link pretrained perception and language understanding to robot actions. This enables manipulation with limited demonstrations via large-scale visual knowledge (Kim et al., 2024). Their visual interface is critical. The action decoder must recover task-relevant objects, end effectors, and their spatial relations from the features it receives. The vision backbone usually uses Vision Transformers (Dosovitskiy et al., 2021), trained with image-text alignment or self-supervision (Radford et al., 2021; Caron et al., 2021; Oquab et al., 2023). These backbones provide robust semantic representations. However, recognizing what an image contains differs from representing where it occurs. A useful global representation may not provide an equally faithful local description for action.

Recent work identifies spatial artifacts in pretrained transformers. These artifacts expose this tension. Darcet et al. (2024) identify high-norm outlier tokens in ViT feature maps. Such tokens appear under full supervision (DeiT-III Touvron et al., 2022), text supervision (CLIP Radford et al., 2021), or self-supervision (DINOv2 Oquab et al., 2023). They often lie in low-information background regions. There, they support internal computation and carry global information, but weaken local content. Shi et al. (2026) trace this to *lazy aggregation*: under coarse semantic supervision, global attention can favor background tokens as shortcuts for image-level semantics. Large feature norms do not fully explain the mismatch. A background patch may match the global image representation yet poorly describe its own location. In manipulation, this can blur the distinction between an object, the active gripper, and an irrelevant part of the scene.

These artifacts are more challenging for VLA models than for purely visual downstream tasks. Robotic manipulation requires precise spatial perception. A policy must infer target object locations and poses. It must also infer their spatial relations to the robot and environment. When artifacts distort the visual backbone's feature maps, semantic ambiguity and spatial misalignment reach the action decoder. This can cause erroneous actions and task failure. To test this risk, we visualized feature responses from several pretrained visual backbones after adaptation to robot-manipulation

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.

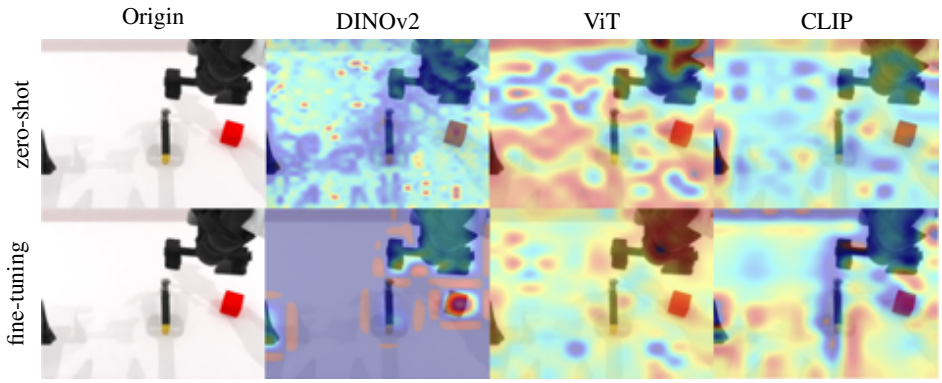


Figure 1: Spatial artifacts persist after adaptation on robot-manipulation data.

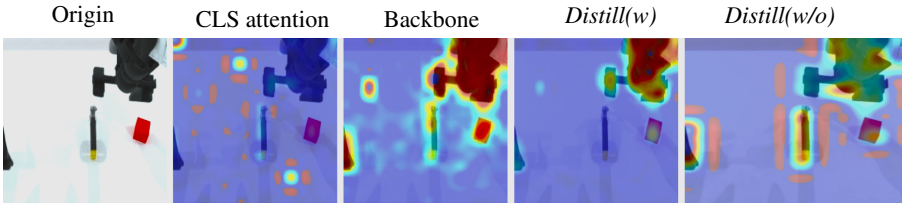


Figure 2: The learned projection alleviates artifacts in its output, whereas artifacts persist in the backbone. From left to right: original image, CLS attention, backbone (final ViT output), learnable projection distillation (*Distill*(w)), and parameter-free projection distillation (*Distill*(w/o)).

data. Figure 1 shows the results: fine-tuning reduced visible high-norm outlier tokens, but substantial spatial misalignment remained.

In vision, several studies address ViT artifacts, but they do not target VLA adaptation constraints. VLA deployment typically adapts an existing pretrained visual model from limited robot demonstrations. Spatial correction must therefore fit this adaptation setting while retaining transferable visual knowledge. Existing methods intervene at different representation-learning stages. Registers, LaSt-ViT and UniRefiner address visual artifacts through additional computation tokens, selective aggregation and lightweight post-training refinement, respectively (Darcet et al., 2024; Shi et al., 2026; Qiu et al., 2026). However, these approaches do not explicitly couple task-specific foreground supervision to action learning, leaving open how to correct the backbone's spatial representations during policy adaptation.

Our projection-based experiments reveal the importance of where this correction is placed. As shown in Figure 2, the learnable distillation layer suppresses background artifacts in its output but fails to endow the backbone with this capability. We therefore remove the distillation projection, directly supervise the backbone, and retain the policy's original input projection. Based on this observation, we propose *Segmentation-Guided Semantic Distillation* (SGSD), which transfers dense spatial priors from a task-trained convolutional segmentation teacher (based on a CNN architecture) directly to the final visual block during policy learning. Convolutional locality and pixel-level foreground supervision supply the spatial guidance needed to correct backbone features used for action prediction.

To balance spatial correction with pretrained feature retention, SGSD aligns feature values, directions and magnitudes with the teacher. Optimized jointly with the action prediction loss as an auxiliary regularizer, the distillation objective couples spatial correction to action learning, correcting anomalous background responses and improving the spatial alignment of foreground features. Most backbone parameters remain fixed, and our retention evaluations show substantially better preservation of pretrained discriminative capability than full fine-tuning. The teacher is used only during training, leaving the deployed policy unchanged.

Our main contributions are summarized as follows:

- We empirically verify that action-supervised fine-tuning alone does not eliminate high-norm outlier artifacts in ViT features. These artifacts manifest as anomalous background responses and spatial-semantic misalignment, which can degrade spatial perception in vision-language-action (VLA) models.
- We propose SGSD, which transfers dense spatial priors from a robot-data-trained CNN segmentation teacher directly into the VLA visual backbone. SGSD jointly optimizes feature, direction, and magnitude alignment with the action prediction loss, suppressing artifacts while preserving pretrained knowledge.
- On five RoboTwin tasks (Mu et al., 2025) and three real-world robot-arm tasks, SGSD improves success rates over frozen-backbone baselines by 17.5 and 18.7 percentage points, respectively.

2 Related work

Pretrained representations and robot policy learning. R3M and VC-1 study reusable visual representations for manipulation and embodied tasks (Nair et al., 2022; Majumdar et al., 2023). On the policy side, ACT predicts action chunks, whereas Diffusion Policy generates action sequences through conditional denoising (Zhao et al., 2023; Chi et al., 2023). OpenVLA transfers pretrained vision–language representations to robot control (Kim et al., 2024); OpenVLA-OFT improves its adaptation recipe through parallel decoding, action chunking and continuous-action regression (Kim et al., 2025). Other VLA designs include π_0, which combines a pretrained vision–language model with flow matching (Black et al., 2024), and TinyVLA, which couples compact multimodal models with a diffusion action decoder (Wen et al., 2024). Our contribution is an auxiliary representation target, not a new action decoder.

Artifacts and spatial correspondence in transformers. Registers provide additional tokens for internal computation (Darcet et al., 2024), and LaSt-ViT modifies aggregation during pretraining to reduce background bias (Shi et al., 2026). UniRefiner instead refines pretrained ViTs using contrastive registers and adaptation (Qiu et al., 2026). Denoising Vision Transformers targets grid artifacts associated with positional embeddings (Yang et al., 2024). These methods address related but distinct defects, with different correction interfaces. We study foreground supervision during robot policy adaptation and measure both representation changes and behavior. FeatUp learns high-resolution feature reconstruction (Fu et al., 2024); our interpolation serves only to align grids for the loss and does not recover independent high-resolution detail.

Feature distillation and auxiliary supervision. Knowledge distillation and intermediate feature hints establish the teacher–student matching paradigm (Hinton et al., 2015; Romero et al., 2015). AM-RADIO and Theia consolidate knowledge from vision foundation models, with Theia targeting robot learning (Ranzinger et al., 2024; Shang et al., 2024). Robot-DIFT distills correspondence-sensitive diffusion features for manipulation (Deng et al., 2026), and Crossway Diffusion studies auxiliary self-supervision for visuomotor policies (Li et al., 2023). Thus, neither feature distillation nor joint training with an action objective is new in isolation. SGSD instead couples a task-trained segmentation teacher with direct, projection-free supervision of the final visual block to address spatial artifacts within an existing perception–action pipeline.

3 Methodology

3.1 Direct supervision at the visual–action interface

Unless otherwise specified, the configurations in this section use ACT or Diffusion Policy with DINOv2, CLIP, or supervised ViT visual backbones. We apply the distillation objective to the output of the final visual block so that the policy loss and the spatial supervision update the same source representation (Figure 3). Let I denote an observation image, q the robot state, and A a demonstrated action chunk. For the twelve-block visual encoders evaluated here, $X^{(11)}$ denotes the final block

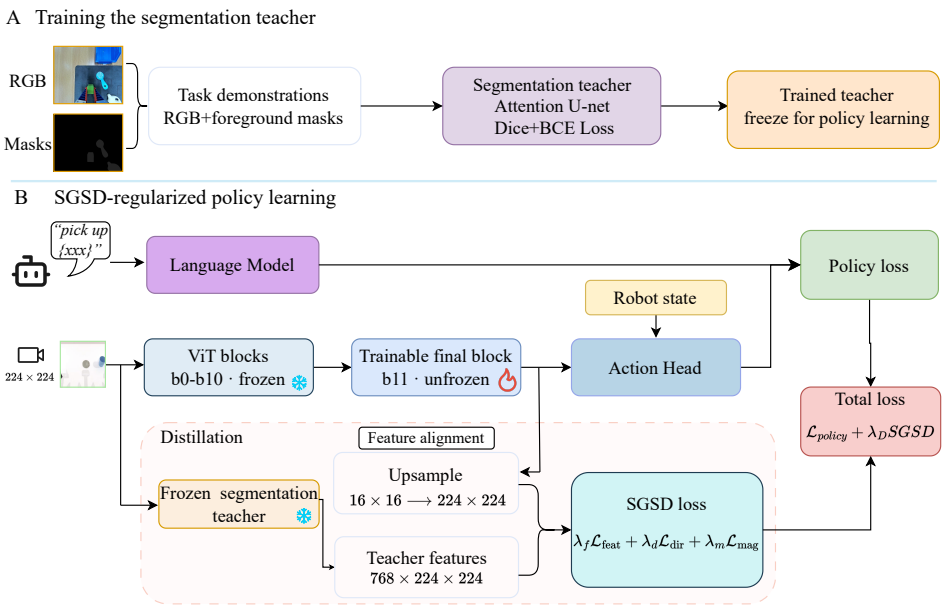


Figure 3: **SGSD training and action readout.** A frozen segmentation teacher supervises final-block patches through parameter-free upsampling. The policy reads features through its existing input projection. The teacher and alignment branch are used only during training. The diagram shows the DINOv2/ACT configuration.

output, including prefix tokens. We distinguish between the supervised patch grid and the normalized features read by the policy,

$$S = \mathrm{Grid}(X^{(11)}_{\mathrm{patch}}), \qquad Z = [\mathrm{LN}_{\mathrm{final}}(X^{(11)})]_{\mathrm{patch}}, \qquad \widehat{A} = g_\omega(\mathcal{R}(Z), q). \tag{1}$$

Here, $S \in \mathbb{R}^{C\times H\times W}$ is the pre-normalization patch grid, Z contains the post-normalization patch features, and $\mathcal{R}$ is the policy input interface. The model-specific interface $\mathcal{R}$ maps the normalized visual features to the representation consumed by the action decoder.

A learned distillation projection P_ϕ would supervise $P_\phi(S)$ rather than S. Figure 2 shows why these are different objectives: matching an auxiliary projected representation can leave visible artifacts in the backbone and action-input features. SGSD therefore omits this auxiliary mapping and matches S directly after parameter-free grid alignment. The policy input projection remains part of $\mathcal{R}$ and is retained at inference.

3.2 Segmentation teacher and parameter-free alignment

We train an Attention U-Net (Ronneberger et al., 2015; Oktay et al., 2018) on the same task demonstration images used for policy learning. Foreground masks from simulator annotations provide pixel-level supervision through a Dice-plus-binary-cross-entropy objective. The resulting convolutional decoder supplies a task-specific spatial prior. We distill dense decoder features rather than binary mask predictions, and evaluate the importance of their image and spatial correspondence through controlled permutations.

After segmentation training, the teacher is frozen and its final decoder output is $T = f^{\mathrm{up4}}_\psi(I) \in \mathbb{R}^{768\times 224\times 224}$. Decoder widths $1024 \to 896 \to 832 \to 768$ produce the required channel count directly, without an additional teacher-side projection. Since all evaluated student models share the same feature dimension $C = 768$ (e.g., an ACT model with DINO as the visual backbone), matching requires only spatial alignment,

$$\widetilde{S} = \mathcal{U}_{\mathrm{bilinear}}(S; 224, 224), \qquad \widetilde{T} = \mathrm{sg}(T). \tag{2}$$

At an input resolution of 224×224, DINOv2 ViT-B/14 produces a 16×16 patch grid, whereas supervised ViT-B/16 (Steiner et al., 2022) and CLIP ViT-B/16 produce 14×14 grids. We bilinearly

upsample the student to the teacher's 224×224 grid; sg stops teacher gradients. Corresponding image locations are matched without introducing learnable parameters in the distillation interface.

3.3 Feature, direction and magnitude alignment

For aligned feature vectors $s_u, t_u \in \mathbb{R}^C$, let Ω contain all batch and spatial positions and $P = |\Omega|$. We match feature values, directions and magnitudes,

$$\mathcal{L}_{\text{feat}} = \frac{1}{PC} \sum_{u \in \Omega} \|s_u - t_u\|_2^2 , \tag{3}$$

$$\mathcal{L}_{\text{dir}} = \frac{1}{P} \sum_{u \in \Omega} \left(1 - \frac{s_u^\top t_u}{\max(\|s_u\|_2, \varepsilon) \max(\|t_u\|_2, \varepsilon)} \right) , \tag{4}$$

$$\mathcal{L}_{\text{mag}} = \frac{1}{P} \sum_{u \in \Omega} (\|s_u\|_2 - \|t_u\|_2)^2 . \tag{5}$$

Feature MSE matches local vector values; cosine matching aligns their directions independently of scale; magnitude alignment aligns their norms. All three use the complete spatial grid, with no foreground-mask weighting. The direction target is the teacher representation, while magnitude alignment raises or lowers each student response toward its corresponding teacher norm.

The three terms reweight overlapping components of feature error: squared feature error already combines radial and angular differences (Appendix B). We use

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{policy}} + \lambda_D \left(\lambda_f \mathcal{L}_{\text{feat}} + \lambda_d \mathcal{L}_{\text{dir}} + \lambda_m \mathcal{L}_{\text{mag}} \right) , \qquad \lambda_D = 1. \tag{6}$$

The three terms are retained on the basis of closed-loop ablations (Section 4.4).

During training, each image is passed through the frozen teacher and the student, and Equation 6 is optimized jointly over the selected visual and policy parameters. For DINOv2 and supervised ViT, we update b_{11}, the input CLS token, and the final LayerNorm (Ba et al., 2016); for CLIP, we update b_{11} only. All other visual parameters remain frozen. At inference, the teacher and upsampling branch are removed, and only the adapted visual encoder and original policy are evaluated. ACT's existing input projection is retained. Details are provided in Appendix A.

4 Experiments

4.1 Experimental setup

We conduct extensive experiments to evaluate SGSD's effectiveness and applicability across tasks, visual backbones and policy architectures in both simulation and real-world setups.

Task success in simulation. We evaluate SGSD on five RoboTwin manipulation tasks, measuring closed-loop success over 100 rollouts per policy. Controlled comparisons examine the contribution of spatial supervision and its effects on backbone representations (§4.2).

Real-world evaluation. We additionally evaluate SGSD on three real-world manipulation tasks using five robot policies, each with and without SGSD. These experiments assess physical task execution across policies and in the presence of visual clutter (§4.3).

Implementation details. We train a task-specific convolutional teacher on foreground segmentation annotations from the policy demonstration images, then freeze it during policy training. The teacher provides dense intermediate features for the same observation that the student processes. Teacher and student features have matched channel widths; parameter-free bilinear upsampling aligns the student's spatial grid without a learned distillation projection. We jointly optimize the policy objective and feature, direction and magnitude alignment losses, using the same distillation loss configuration in simulation and real-world experiments. SGSD updates the final visual block while earlier block weights remain frozen; backbone-specific update sets and ablation configurations are detailed in Appendix A.

Table 1: **Success rates across five tasks.** Values are percentages over 100 rollouts. Action-only uses the SGSD visual update set without teacher supervision.

Task	Baseline (Frozen)	Action-only	SGSD
Object stand	10	13	**34**
Dual bottles	43	40	**63**
Open laptop	69	73	**83**
Move card	38	34	**50**
Hammer	54	36	**70**
Avg.	42.8	39.2	**60.0**

4.2 Simulation Experiments

Simulation task definition. We evaluate five RoboTwin manipulation tasks (Mu et al., 2025): placing an object on a display stand, manipulating two bottles, opening a laptop, moving playing cards aside, and bringing a grasped hammer into contact with a target block. These tasks cover object placement, bimanual coordination, articulated-object interaction and tool use, requiring the policy to resolve spatial relationships between objects and end effectors. We report full-task success using the simulator's task-specific completion criteria.

Simulation experiments. We use ACT with DINOv2 for the main five-task comparison, with simulator foreground masks providing teacher supervision. Policies are trained on the same demonstration collection under three adaptation strategies: *Baseline (Frozen)*, which keeps the visual backbone fixed; *Action-only*, which updates the same visual parameters as SGSD using only the policy objective; and *SGSD*, which additionally applies segmentation-guided feature supervision. The action-only comparison isolates the contribution of teacher supervision from the effect of unfreezing visual parameters.

Spatial artifact correction. On the hammer task, we measure foreground–background norm contrast, norm-hotspot overlap and episode-held-out foreground probe IoU. Layerwise responses and outlier locations in DINOv2, CLIP and supervised ViT provide additional spatial diagnostics (Appendices B and G; Figure 9).

Backbone and policy generality. We vary the visual encoder while holding ACT fixed, comparing DINOv2, CLIP, supervised ViT and register-equipped DINOv2 (Darcet et al., 2024). We separately vary the policy architecture, comparing ACT and Diffusion Policy with DINOv2. Each configuration is evaluated against its frozen-backbone baseline on the hammer task. Pretrained feature retention is measured by frozen-encoder kNN classification on CIFAR-100 and Oxford-IIIT Pet (Krizhevsky, 2009; Parkhi et al., 2012), using CLS and mean-patch readouts (Appendix A).

Simulation experiment results. SGSD achieves the highest success rate on all five tasks, increasing mean success from 42.8% with the frozen baseline and 39.2% with action-only adaptation to 60.0% (Table 1). With the same trainable visual parameters, SGSD improves hammer-task success from 36% to 70%, with a mean paired gain of 37.5 percentage points across three training seeds (Appendix D.1). These comparisons establish the benefit of spatial supervision beyond unfreezing the backbone for action learning. Appendix C and Figure 6 provide paired statistical results and supplementary task comparisons.

Spatial artifact correction. Figure 4 compares feature-norm maps and foreground–background contrast for the same observation. SGSD concentrates strong responses on the annotated foreground, correcting the scattered responses retained by frozen features and action-only adaptation. Aggregate hotspot overlap and foreground probe results confirm improved spatial alignment and foreground readability (Appendix G, Table 16). The correction concerns where strong responses occur, rather than simply reducing their magnitude. Additional task visualizations appear in Figure 10 of the same appendix.

Backbone and policy generality. With ACT held fixed, SGSD improves success for each visual encoder, including register-equipped DINOv2 (Table 2(a)). Task-specific supervision therefore adds value even when the encoder already includes register tokens. With DINOv2 held fixed, both ACT

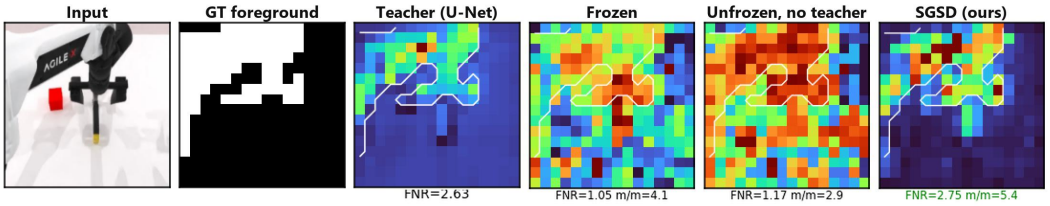


Figure 4: **Spatial organization of feature responses on the hammer task.** Teacher and student feature-norm maps are shown for the same frame; student maps are extracted from the final transformer block before LayerNorm. White contours mark annotated foreground. FNR is the foreground/background mean norm ratio; m/m is the maximum/mean patch norm ratio. Statistics describe frame 4, whereas Table 16 reports aggregate diagnostics. Colors are scaled independently.

Table 2: **Backbone and policy comparisons.** Hammer-task success (%) over 100 rollouts. (a) Visual backbones with ACT. (b) Policy architectures with DINOv2.

(a) Visual backbones

Backbone	Baseline (Frozen)	SGSD
DINOv2-B	54	**70**
CLIP-B	38	**49**
ViT-augreg2-B	18	**34**
DINOv2-reg4	30	**63**

(b) Policy architectures

Policy	Baseline (Frozen)	SGSD
ACT	54	**70**
Diffusion Policy	18	**36**

and Diffusion Policy improve (Table 2(b)). The latter uses pooled visual features globally, showing that SGSD's gains do not depend on the policy consuming a spatial feature grid.

For DINOv2 and CLIP, SGSD also retains substantially more pretrained discriminative capability than full fine-tuning. Mean kNN accuracy losses across the four dataset/readout combinations are 6.90 versus 78.29 points for DINOv2, and 3.23 versus 27.17 for CLIP. Restricted spatial adaptation thus improves control while reducing distortion of transferable features (Kumar et al., 2022); detailed comparisons appear in Appendix D.

4.3 Real-World Experiments

Real-world task definition. We define three real-world manipulation tasks: large-object sorting, small-object sorting, and small-object sorting in cluttered scenes, as illustrated in Fig. 5. The first two tasks are instruction-conditioned pick-and-place tasks in which a robot arm picks up a block, a basket, or a 3D-printed object according to the instruction and places it in a designated location in a storage box. In the cluttered setting, the robot must select the instructed object from multiple candidates and place it into the storage box. We decompose the task into *pick* and *place*, and report the success rate for each subtask separately.

Real-world experiments. We use Cutie (Cheng et al., 2024) to obtain foreground masks of the robotic arm and task-relevant objects (Fig. 5). These masks provide the teacher's segmentation annotations for real-world demonstrations.

We evaluate SGSD across five widely used robot policies: ACT (Zhao et al., 2023), Diffusion Policy (Chi et al., 2023), OpenVLA-OFT (Kim et al., 2025), π_0 (Black et al., 2024) and Tiny-VLA (Wen et al., 2024). ACT and Diffusion Policy use DINOv2 visual backbones; OpenVLA-OFT, π_0 and Tiny-VLA retain their original visual encoders. For each policy and task, we train configurations with and without SGSD using three random seeds and evaluate each trained model over 20 trials. Table 3 reports success counts pooled over the resulting 60 trials, separately for pick and place.

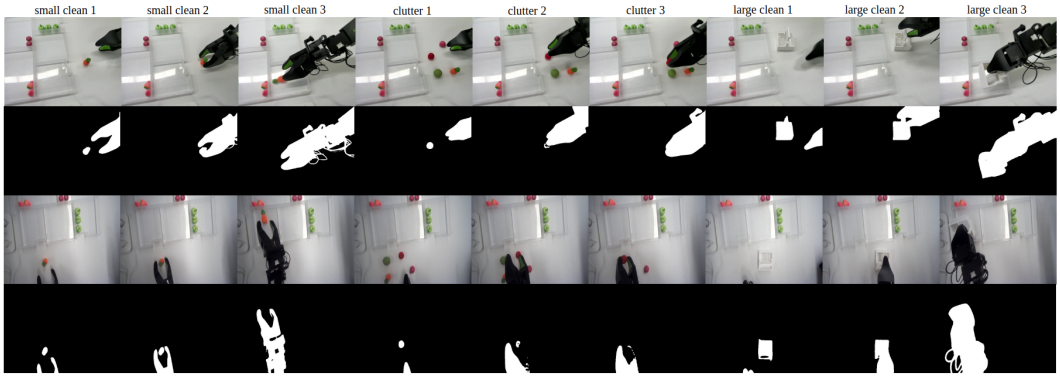


Figure 5: Real-world scenes and their corresponding segmentation maps.

Table 3: Real-World Manipulation Success Count. Each task is decomposed into pick and place. +SGSD denotes the proposed segmentation-guided semantic distillation.

Method	Large-object sorting		Small-object sorting		Cluttered small-object sorting		Avg.
	Pick	Place	Pick	Place	Pick	Place	
ACT	40	38	37	33	26	22	32.7
ACT+SGSD	49	48	46	42	43	40	44.7
Diffusion Policy	43	40	37	34	27	24	34.2
DP+SGSD	50	48	51	49	48	45	48.5
OpenVLA-OFT	40	37	38	34	28	26	33.8
OpenVLA-OFT+SGSD	48	36	44	41	43	40	42.0
π_0	48	45	40	35	32	27	37.8
π_0+SGSD	**55**	**54**	**51**	**48**	**49**	**45**	**50.3**
Tiny-VLA	46	45	40	36	30	25	37.0
Tiny-VLA+SGSD	52	50	47	43	43	40	45.8

Real-world experiment results. SGSD improves average success for all five policies, yielding an overall gain of 18.6 percentage points across policies and task stages (Table 3). The largest gains occur in cluttered small-object sorting. For Diffusion Policy, successful picks increase from 27 to 48 and successful placements from 24 to 45 out of 60 trials. The only exception is a 1-trial drop for OpenVLA-OFT in large-object place, which is within random variation and is outweighed by its 8.2-trial average gain. The improvements across policies, with the largest gains under clutter, demonstrate SGSD's effectiveness in physical manipulation.

We further test whether SGSD complements existing artifact mitigation methods. Using ACT and Diffusion Policy, we compare robotics-oriented visual backbones based on LaSt-ViT and Register, each with and without SGSD. The remaining training settings are matched within each comparison, and each configuration is evaluated over 20 trials (Table 4). SGSD improves every evaluated LaSt-ViT and Register configuration relative to its corresponding baseline: with ACT, adding SGSD raises the average success count from 12.2 to 15.0 out of 20 for LaSt-ViT and from 10.2 to 15.0 for Register. Both backbone–SGSD combinations also improve Diffusion Policy performance, demonstrating that task-specific spatial supervision provides additional benefits beyond existing artifact mitigation.

4.4 Ablation Studies

Effect of distillation objectives. We compare feature-only, two-term and complete distillation objectives while keeping the supervision location and trainable visual parameters fixed. Adding either direction or magnitude alignment improves success over feature MSE alone, while the complete objective performs best (Table 5).

Table 4: Real-World Manipulation Success Count. Each task is decomposed into pick and place. We compare Last-ViT, Register, and their combinations with SGSD under two baseline policies: ACT and Diffusion Policy (DP).

Method	Large-object sorting		Small-object sorting		Cluttered small-object sorting		Avg.
	Pick	Place	Pick	Place	Pick	Place	
ACT (DINO)	12	12	11	9	7	6	9.50
ACT + Last-ViT	13	11	14	13	13	9	12.1
ACT + Register	12	11	11	10	9	8	10.2
ACT + Last-ViT + SGSD	18	16	15	14	15	12	15.0
ACT + Register + SGSD	17	15	16	14	15	13	15.0
DP (DINO)	13	12	12	10	11	9	11.2
DP + Last-ViT	15	14	13	9	11	10	12.0
DP + Register	14	13	12	11	12	11	12.1
DP + Last-ViT + SGSD	17	15	15	13	14	12	14.3
DP + Register + SGSD	16	16	14	12	14	11	13.8

Table 5: **Loss-component ablations.** Hammer-task success over 100 rollouts. Coefficients correspond to feature, direction and magnitude alignment.

Objective	Coefficients	Success (%)
Feature MSE	$(1, 0, 0)$	47
Without magnitude	$(1, .5, 0)$	56
Without direction	$(1, 0, .5)$	58
Complete loss	$(1, .5, .5)$	**70**

Effect of teacher correspondence. To test whether the benefit depends on spatially aligned supervision, we shuffle teacher feature positions or pair student features with teacher features from a different image. These controls reduce success from 70% to 46% and 38%, respectively (Appendix E.2). Both controls alter the backbone but fail to reproduce SGSD's gains, showing that effective supervision depends on correspondence rather than feature displacement alone (Appendix F, Table 14).

Effect of supervision placement. To identify where spatial supervision best supports action learning, we compare intermediate-, final- and dual-block distillation. We examine how each choice affects foreground contrast at the backbone output, task success and pretrained feature retention. Final-block supervision preserves the foreground contrast that weakens downstream of intermediate supervision, while matching dual-block success with substantially less degradation of pretrained features. These findings motivate supervision at b_{11}; detailed comparisons are provided in Appendix E.1.

5 Conclusion

Pretrained vision transformers contain spatial artifacts that undermine spatial fidelity required for robotic manipulation and vision–language–action interfaces. These artifacts persist after action fine-tuning, and cleaner projected features can coexist with unresolved backbone defects. We propose Segmentation-Guided Semantic Distillation (SGSD) to inject task-specific spatial supervision into backbone learning. Experiments show SGSD suppresses anomalous background responses, improves foreground readability, and increases manipulation success across tasks, visual backbones, and policy architectures. It retains more pretrained discriminative capability than full fine-tuning while leaving inference unchanged, establishing spatial fidelity as essential for transferring vision foundation models to robot control.

REPRODUCIBILITY STATEMENT

Section 3 specifies the objective and adaptation sets. Appendix A records preprocessing, trajectory splits, optimizer settings and probe training; Appendix C defines shared-scene pairing and statistical tests. Extended tables distinguish supervision locations, checkpoint repetitions and training seeds.

AI USE STATEMENT

Generative AI tools assisted with English translation, manuscript restructuring, language editing, literature retrieval and bibliographic checks. The experimental values in this manuscript originate from author-provided records.

REFERENCES

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep Reinforcement Learning at the Edge of the Statistical Precipice. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021. https://arxiv.org/abs/2108.13264

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization. *arXiv preprint arXiv:1607.06450*, 2016. https://arxiv.org/abs/1607.06450

Yoav Benjamini and Yosef Hochberg. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society: Series B (Methodological), 57(1):289–300*, 1995. https://doi.org/10.1111/j.2517-6161.1995.tb02031.x

Kevin Black, Noah Brown, Danny Driess, et al. π_0: A Vision-Language-Action Flow Model for General Robot Control. *arXiv preprint arXiv:2410.24164*, 2024. https://arxiv.org/abs/2410.24164

Mathilde Caron, Hugo Touvron, Ishan Misra, et al. Emerging Properties in Self-Supervised Vision Transformers. *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. https://arxiv.org/abs/2104.14294

Ho Kei Cheng, Seoung Wug Oh, Brian Price, Joon-Young Lee, and Alexander Schwing. Putting the Object Back into Video Object Segmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3151–3161, 2024. https://openaccess.thecvf.com/content/CVPR2024/html/Cheng_Putting_the_Object_Back_into_Video_Object_Segmentation_CVPR_2024_paper.html

Cheng Chi, Siyuan Feng, Yilun Du, Zhenjia Xu, Eric Cousineau, Benjamin Burchfiel, and Shuran Song. Diffusion Policy: Visuomotor Policy Learning via Action Diffusion. *Robotics: Science and Systems (RSS)*, 2023. https://www.roboticsproceedings.org/rss19/p026.pdf

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision Transformers Need Registers. *International Conference on Learning Representations (ICLR)*, 2024. https://arxiv.org/abs/2309.16588

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. https://www.image-net.org/static_files/papers/imagenet_cvpr09.pdf

Yu Deng, Yufeng Jin, Xiaogang Jia, Jiahong Xue, Gerhard Neumann, and Georgia Chalvatzaki. Robot-DIFT: Correspondence-Sensitive Diffusion Features for Contact-Rich Robot Manipulation. *arXiv preprint arXiv:2602.11934*, 2026. https://arxiv.org/abs/2602.11934v2

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *International Conference on Learning Representations (ICLR)*, 2021. https://arxiv.org/abs/2010.11929

Stephanie Fu, Mark Hamilton, Laura Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. FeatUp: A Model-Agnostic Framework for Features at Any Resolution. *International Conference on Learning Representations (ICLR)*, 2024. https://proceedings.iclr.cc/paper_files/paper/2024/hash/c5601d99ed028448f29d1dae2e4a926d-Abstract-Conference.html

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the Knowledge in a Neural Network. *arXiv preprint arXiv:1503.02531*, 2015. https://arxiv.org/abs/1503.02531

Edward J. Hu, Yelong Shen, Phillip Wallis, et al. LoRA: Low-Rank Adaptation of Large Language Models. *International Conference on Learning Representations (ICLR)*, 2022. https://arxiv.org/abs/2106.09685

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, et al. OpenVLA: An Open-Source Vision-Language-Action Model. *arXiv preprint arXiv:2406.09246*, 2024. https://arxiv.org/abs/2406.09246

Moo Jin Kim, Chelsea Finn, and Percy Liang. Fine-Tuning Vision-Language-Action Models: Optimizing Speed and Success. *Robotics: Science and Systems (RSS)*, 2025. https://www.roboticsproceedings.org/rss21/p017.html

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)*, 2015. https://arxiv.org/abs/1412.6980

Alex Krizhevsky. Learning Multiple Layers of Features from Tiny Images. *Technical report, University of Toronto*, 2009. https://www.cs.toronto.edu/~kriz/cifar.html

Ananya Kumar, Aditi Raghunathan, Robbie Jones, Tengyu Ma, and Percy Liang. Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution. *International Conference on Learning Representations (ICLR)*, 2022. https://arxiv.org/abs/2202.10054

Xiang Li, Varun Belagali, Jinghuan Shang, and Michael S. Ryoo. Crossway Diffusion: Improving Diffusion-based Visuomotor Policy via Self-supervised Learning. *arXiv preprint arXiv:2307.01849*, 2023. https://arxiv.org/abs/2307.01849

Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization. *International Conference on Learning Representations (ICLR)*, 2019. https://arxiv.org/abs/1711.05101

Arjun Majumdar, Karmesh Yadav, Sergio Arnaud, et al. Where are we in the search for an Artificial Visual Cortex for Embodied Intelligence? *arXiv preprint arXiv:2303.18240*, 2023. https://arxiv.org/abs/2303.18240

Quinn McNemar. Note on the Sampling Error of the Difference between Correlated Proportions or Percentages. *Psychometrika, 12(2):153–157*, 1947. https://doi.org/10.1007/BF02295996

Yao Mu, Tianxing Chen, Zanxin Chen, et al. RoboTwin: Dual-Arm Robot Benchmark with Generative Digital Twins. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. https://arxiv.org/abs/2504.13059

Suraj Nair, Aravind Rajeswaran, Vikash Kumar, Chelsea Finn, and Abhinav Gupta. R3M: A Universal Visual Representation for Robot Manipulation. *Conference on Robot Learning (CoRL)*, 2022. https://arxiv.org/abs/2203.12601

Ozan Oktay, Jo Schlemper, Loic Le Folgoc, et al. Attention U-Net: Learning Where to Look for the Pancreas. *Medical Imaging with Deep Learning (MIDL)*, 2018. https://arxiv.org/abs/1804.03999

Maxime Oquab, Timothée Darcet, Théo Moutakanni, et al. DINOv2: Learning Robust Visual Features without Supervision. *arXiv preprint arXiv:2304.07193*, 2023. https://arxiv.org/abs/2304.07193

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and Dogs. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012. https://www.robots.ox.ac.uk/~vgg/publications/2012/parkhi12a/

Belinda Phipson and Gordon K. Smyth. Permutation P-values Should Never Be Zero: Calculating Exact P-values When Permutations Are Randomly Drawn. *Statistical Applications in Genetics and Molecular Biology, 9(1):Article 39*, 2010. https://doi.org/10.2202/1544-6115.1585

Congpei Qiu, Zhaoyu Hu, Wei Ke, Zhuotao Tian, Yanhao Wu, and Tong Zhang. UniRefiner: Teaching Pre-trained ViTs to Self-Dispose Dross via Contrastive Register. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. https://arxiv.org/abs/2605.19622

Alec Radford, Jong Wook Kim, Chris Hallacy, et al. Learning Transferable Visual Models From Natural Language Supervision. *International Conference on Machine Learning (ICML)*, 2021. https://arxiv.org/abs/2103.00020

Mike Ranzinger, Greg Heinrich, Jan Kautz, and Pavlo Molchanov. AM-RADIO: Agglomerative Vision Foundation Model—Reduce All Domains Into One. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. https://openaccess.thecvf.com/content/CVPR2024/html/Ranzinger_AM-RADIO_Agglomerative_Vision_Foundation_Model_Reduce_All_Domains_Into_One_CVPR_2024_paper.html

Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. FitNets: Hints for Thin Deep Nets. *International Conference on Learning Representations (ICLR)*, 2015. https://arxiv.org/abs/1412.6550

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 2015. https://arxiv.org/abs/1505.04597

Jinghuan Shang, Karl Schmeckpeper, Brandon B. May, Maria Vittoria Minniti, Tarik Kelestemur, David Watkins, and Laura Herlant. Theia: Distilling Diverse Vision Foundation Models for Robot Learning. *Conference on Robot Learning (CoRL)*, 2024. https://arxiv.org/abs/2407.20179v2

Cheng Shi, Yizhou Yu, and Sibei Yang. Vision Transformers Need More Than Registers. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. https://arxiv.org/abs/2602.22394v2

Leslie N. Smith and Nicholay Topin. Super-Convergence: Very Fast Training of Neural Networks Using Large Learning Rates. *arXiv preprint arXiv:1708.07120*, 2017. https://arxiv.org/abs/1708.07120

Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers. *Transactions on Machine Learning Research (TMLR)*, 2022. https://openreview.net/forum?id=4nPswr1KcP

Hugo Touvron, Matthieu Cord, and Hervé Jégou. DeiT III: Revenge of the ViT. *European Conference on Computer Vision (ECCV)*, 2022. https://arxiv.org/abs/2204.07118

Junjie Wen, Yichen Zhu, Jinming Li, et al. TinyVLA: Towards Fast, Data-Efficient Vision-Language-Action Models for Robotic Manipulation. *arXiv preprint arXiv:2409.12514*, 2024. https://arxiv.org/abs/2409.12514

Edwin B. Wilson. Probable Inference, the Law of Succession, and Statistical Inference. *Journal of the American Statistical Association, 22(158):209–212*, 1927. https://doi.org/10.1080/01621459.1927.10502953

Jiawei Yang, Katie Z. Luo, Jiefeng Li, et al. Denoising Vision Transformers. *European Conference on Computer Vision (ECCV)*, 2024. https://arxiv.org/abs/2401.02957

Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware. *Robotics: Science and Systems (RSS)*, 2023. https://arxiv.org/abs/2304.13705

A IMPLEMENTATION DETAILS

Teacher and data. The Attention U-Net teacher is trained on the same task demonstrations used for policy learning, with simulator foreground masks. Episodes are sorted by filename and split at 80%, with the remaining episodes used for early stopping. Training images and masks are horizontally flipped together with probability 0.5; RGB images additionally receive color jitter (brightness, contrast and saturation 0.2; hue 0.1). Images are resized bicubically to 224×224, masks by nearest-neighbor interpolation, and RGB values use ImageNet (Deng et al., 2009) normalization. The decoder widths are $1024, 896, 832, 768$, and the final `up4` tensor has shape $768 \times 224 \times 224$. Teacher optimization uses Adam (Kingma and Ba, 2015), learning rate 8×10^{-3}, weight decay 10^{-4} and batch size 8. OneCycleLR (Smith and Topin, 2017) uses 10% warmup and cosine annealing; training is capped at 10 epochs, with early-stopping patience 5. The dense feature target is fixed during policy learning.

Table 6: **Policy-training configuration.** The main configuration is the DINOv2 + ACT model with 70 anchor-task successes.

Item	Setting
Visual input	224×224 RGB; $C = 768$; DINOv2 B/14 or ViT/CLIP B/16.
Main visual updates	`blocks.11.*`, `cls_token`, `norm.*`; other blocks and patch/position embeddings fixed.
CLIP visual updates	`blocks.11.*` only.
Alignment	Student bilinear upsampling to 224×224, `align_corners=False`; teacher stop-gradient; no trainable distillation projection.
ACT input interface	Existing 1×1 convolution, $768 \to 512$, retained in training and inference.
Optimizer	AdamW (Loshchilov and Hutter, 2019); weight decay 10^{-4}; backbone and non-backbone learning rates both 10^{-5}.
ACT schedule	4,000 epochs; batch size 4; final checkpoint evaluated.
ACT objective	L1 action loss plus KL regularization (weight 10) and distillation ($\lambda_D = 1$).
ACT architecture	Chunk length 50; hidden size 512; feed-forward size 3,200; 4 encoder layers, 7 decoder layers, 8 heads; dropout 0.1; state dimension 14.
DP configuration	Conditional denoising plus distillation; batch size 32; 300 epochs; patch-mean DINOv2 features.
Anchor demonstrations	47 trajectories in the `demo_clean-50` collection; `cam_high`, `cam_left_wrist`, `cam_right_wrist`.
Evaluation	100 completed trials per arm; temporal aggregation for ACT; fixed scene-seed sequence; same-machine shared-scene pairing.

The `demo_clean-50` name denotes the collection protocol, not a guarantee of 50 stored trajectories per task. The main run saves at the end of its 4,000-epoch schedule; policy evaluation uses final checkpoints rather than selection by task success. The evaluation configuration records instruction type `unseen`; this setting is not evidence that the evaluated ACT and DP implementations perform language grounding. Teacher and probe statistics are treated as in-distribution diagnostics, not independent evidence of held-out representation generalization.

Policy data and adaptation controls. Policy episodes are randomly permuted and split 80:20 at trajectory level (37/10 for the 47-episode anchor collection, 40/10 for each 50-episode collection). The four non-anchor tasks reported here each record 50 episodes. ACT samples a random starting frame and the subsequent action chunk from each episode, scales RGB to $[0, 1]$ and applies the backbone's normalization; it uses no stochastic image augmentation. Action and state normalization statistics are computed over the stored collection. LoRA (Hu et al., 2022) is inserted into the attention `qkv` and `proj` linear layers of every block, with $\alpha = r$ and hence unit α/r scaling; the base weights remain frozen. DINOv2-reg4 uses `blocks.11.*`, `cls_token` and `norm.*` updates, leaving its register-token parameters fixed.

Segmentation probe and hotspot selection. The foreground probe is an affine $768 \to 1$ classifier on patch features. The recorded jobs sample 1,500 frames and hold out 30% of sampled episodes using split seed 0. Per-channel standardization uses only the probe-training split. Adam trains the classifier for 600 steps at learning rate 3×10^{-3}, using batches of up to 16,384 tokens and class-balanced binary cross-entropy. IoU is evaluated at probability 0.5. This episode-held-out probe measures in-distribution readability; its split is separate from the policy/teacher training splits. For

hotspot diagnostics, each image selects the top $\max(1, \mathrm{round}(\varphi N))$ patch scores or norms, where N is the grid size and φ is the foreground fraction of the diagnostic batch; intersection and union are aggregated over that batch.

Pretrained feature retention. We evaluate frozen features with cosine kNN ($k = 20$) on CIFAR-100 and Oxford-IIIT Pet, using their distributed training partitions as reference banks and test partitions for evaluation. CLS and mean-patch features from the encoder's final normalized output are separately ℓ_2-normalized. Neighbors vote with weights $\exp(\mathrm{cosine}/0.07)$, summed by class. CIFAR-100 images are resized to 224×224; Pet images are resized to a 256-pixel shorter side and center-cropped to 224×224. RGB normalization follows each backbone's pretrained configuration. No encoder or classifier is trained in this evaluation. Reported retention changes average the four dataset/readout accuracy differences from the corresponding pretrained encoder.

Training cost. The recorded hardware includes instances labeled RTX 4090 48G. Teacher training costs approximately 4–5 GPU-hours per task. The illustrative policy timing, about 10 hours with distillation versus 1.5 hours without it, is a single-run wall-clock comparison rather than a hardware-normalized benchmark. Teacher inference and dense 224×224 feature matching contribute additional training work; no such branch is evaluated when predicting actions.

B Representation objectives and diagnostic definitions

For nonzero feature vectors, squared error decomposes as

$$\|s - t\|_2^2 = (\|s\|_2 - \|t\|_2)^2 + 2\,\|s\|_2\,\|t\|_2\,\big(1 - \cos(s, t)\big). \tag{7}$$

The explicit direction and magnitude terms therefore reweight components already present in feature error. Feature MSE is averaged over both positions and channels; the magnitude term is averaged over positions only. This reduction convention helps explain the unequal scalar contributions in the recorded endpoint (Section 4.4).

For foreground and background sets $\mathcal{F}, \mathcal{B}$, the foreground/background norm ratio is

$$\mathrm{FNR}(F, Y) = \frac{|\mathcal{F}|^{-1} \sum_{u \in \mathcal{F}} \|F_u\|_2}{|\mathcal{B}|^{-1} \sum_{u \in \mathcal{B}} \|F_u\|_2}. \tag{8}$$

It describes mean amplitude contrast when both sets and the denominator are valid. It does not measure localization accuracy. The DINOv2 anchor diagnostic slice has foreground fraction 0.12442; the ViT/CLIP grid convention gives 0.12268. Norm-hotspot IoU measures overlap between a selected high-norm region and annotated foreground. Patch Score instead uses $r_u = \cos(F_u, F_{\mathrm{CLS}})$ (Shi et al., 2026). An anomalously high background score can indicate global-semantic content at a mismatched location; low scores alone do not define artifacts. This quantity is distinct from attention weights.

Reading features through LayerNorm. For one token x, LayerNorm (Ba et al., 2016) is

$$\mathrm{LN}_{\gamma,\beta}(x) = \gamma \odot \frac{x - \mu(x)\mathbf{1}}{\sqrt{v(x) + \epsilon}} + \beta. \tag{9}$$

For $a > 0$, the normalized expression for ax has denominator $\sqrt{v(x) + \epsilon/a^2}$. Positive rescaling is therefore approximately removed when the variance is nondegenerate and ϵ is small. Nevertheless, the post-LN norm can vary with centered direction and affine parameters. Neither post-LN norms nor interventions applied *after* normalization are necessarily irrelevant. Training-time magnitude alignment also changes joint optimization and is not equivalent to a single inference-time rescaling.

Hotspot reference and diagnostic scope. The reported reference $\varphi/(2 - \varphi)$ assumes both foreground and a random hotspot occupy fraction φ and takes the ratio of expected intersection to expected union. It is not generally the exact expected IoU of finite random sets. If the hotspot fraction is τ, the corresponding ratio is $\varphi\tau/(\varphi + \tau - \varphi\tau)$. Reference multiples require the matching selection convention. Mean and tail diagnostics can also differ: the frozen post-LN FNR is 1.0029, while the background share of high-norm outliers is 0.9625 against a background-area fraction of 0.8756. Such summaries describe different distributional properties; they do not independently identify a behavioral cause.

C EVALUATION AND STATISTICAL REPORTING

For models A, B, let $\mathcal{I}_{AB}$ be the set of scenes successfully instantiated and evaluated for both models on the same platform. The paired effect is

$$\Delta_{A\to B} = \frac{100}{n_{AB}} \sum_{i\in\mathcal{I}_{AB}} (y_i^B - y_i^A), \qquad n_{AB} = |\mathcal{I}_{AB}|, \quad y_i \in \{0,1\}. \tag{10}$$

For example, 54 versus 70 successes out of 100 gives a raw 16-point difference; 48 versus 63 successes on 86 shared scenes gives the reported paired 17.4-point effect. Comparisons with different scene intersections cannot be subtracted to create a new paired effect. Parenthetical differences in Table 7 instead subtract SGSD's raw success rate from each comparator's rate; because every policy has 100 rollouts, these differences are also percentage-point differences.

Let n_{01} count scenes on which only B succeeds and n_{10} those on which only A succeeds. The two-sided exact McNemar test uses the discordant pairs (McNemar, 1947):

$$p = \min\left\{1, 2\sum_{k=0}^{\min(n_{01},n_{10})} \binom{n_{01}+n_{10}}{k} 2^{-(n_{01}+n_{10})}\right\}. \tag{11}$$

With no discordant pairs, $p = 1$. Paired marginal success counts determine effect sizes but do not uniquely determine discordance counts. The accompanying source data provide scene-level outcomes and discordance counts for comparisons verified from the raw evaluation logs.

Single-policy uncertainty, where plotted, uses Wilson 95% intervals (Wilson, 1927). For k successes in n trials, $\widehat{p} = k/n$ and $z = 1.96$, the interval is

$$\frac{\widehat{p} + z^2/(2n) \;\pm\; z\sqrt{\widehat{p}(1-\widehat{p})/n + z^2/(4n^2)}}{1 + z^2/n}. \tag{12}$$

These intervals summarize scenario outcomes for a fixed trained model; they are not intervals for paired effects or across-seed performance. Overlap between marginal intervals is not used to infer paired significance. Limited training repeats warrant the distinction between evaluation and training variability (Agarwal et al., 2021).

Exploratory correlations. Spearman correlations in Appendix F use configurations as observations, with average ranks for ties. Two-sided p values compare absolute correlations under permutation: all permutations are enumerated for the five- and six-configuration analyses, while the larger diagnostic comparisons use 200,000 random permutations (seed 12345) with the $(k+1)/(B+1)$ correction (Phipson and Smyth, 2010), where k is the number of permutations at least as extreme as observed and B the number sampled. Undefined metrics are omitted separately for each comparison, and the effective configuration count is reported. These exploratory p values are unadjusted; configurations share training data and evaluation scenes, so they are not independent training replicates.

Multiplicity. The five reported frozen-versus-SGSD comparisons form a post hoc analysis family. Applying Benjamini–Hochberg at FDR 0.05 (Benjamini and Hochberg, 1995) to their reported p values retains the object-stand, dual-bottle, laptop and hammer effects; the card comparison remains outside the rejection set. This adjustment applies to the reported five-task family and does not control selection over the wider task and configuration search.

Task identifiers and aggregate effects. The shortened task names in Table 1 map, in row order, to `place_object_stand`, `pick_dual_bottles`, `open_laptop`, `move_playingcard_away` and `beat_block_hammer`. The five-task frozen/action-only/SGSD macro-average success rates are 42.8/39.2/60.0%, giving a raw frozen-to-SGSD difference of 17.2 percentage points. Averaging the five task-wise paired effects instead gives 17.5 points: $(24 + 20 + 14 + 12 + 100 \times 15/86)/5 \approx 17.49$. The distinction comes from the hammer comparison, whose shared-scene effect is $100 \times 15/86 \approx 17.44$ points, versus a raw difference of 16 points.

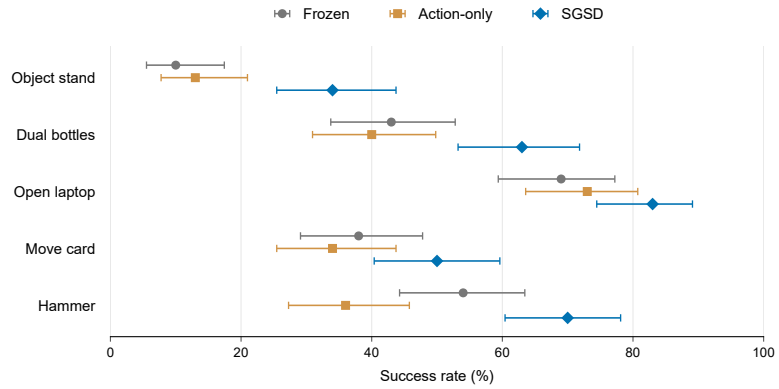


Figure 6: **Adaptation strategies across five tasks.** Success rates over 100 rollouts per policy; error bars show 95% Wilson intervals.

Table 7: **DINOv2 adaptation strategies on the hammer task.** Parameters count trainable visual weights. Parentheses indicate success-rate differences from SGSD in percentage points. Action-only rows specify their visual update sets; LoRA follows Hu et al. (2022).

Adaptation	Parameters	Success (%)
Baseline (Frozen)	0	54 (−16)
LoRA $r = 4$	221,184	40 (−30)
LoRA $r = 16$	884,736	34 (−36)
LoRA $r = 64$	3,538,944	0 (−70)
Action-only b_{11}	—	42 (−28)
Action-only b_8–b_{11}	—	42 (−28)
Action-only attention	—	37 (−33)
Full fine-tuning	∼86.6M	21 (−49)
SGSD	∼7.1M	**70**

Table 8: **Action-only adaptation.** Paired comparisons are frozen → action-only; Δ is in percentage points and p is the unadjusted exact McNemar value.

Task	Raw counts /100	Paired (n)	Δ	p
Object stand	10 → 13	9→13 (98)	+4.1	.4240
Dual bottles	43 → 40	43→40 (100)	-3.0	.7111
Open laptop	69 → 73	68→72 (99)	+4.0	.4545
Move card	38 → 34	38→34 (100)	-4.0	.5716
Hammer	54 → 36	47→32 (86)	-17.4	.0081

Table 9: **Paired anchor-task extensions.** Each configuration is compared with its frozen baseline. c denotes the input CLS token; LN denotes final LayerNorm.

Policy / backbone	Visual updates	Raw /100	Paired (n)	Δ	p
ACT / DINOv2	b_{11}, c, LN	54 → 70	48→63 (86)	+17.4	.0107
ACT / CLIP	b_{11}	38 → 49	32→46 (87)	+16.1	.0243
ACT / ViT augreg2	b_{11}, c, LN	18 → 34	18→31 (88)	+14.8	.0146
ACT / DINOv2-reg4	b_{11}, c, LN	30 → 63	29→59 (88)	+34.1	< 0.0001
DP / DINOv2	b_{11}, c, LN	18 → 36	16→31 (89)	+16.9	.0167

D EXTENDED POLICY AND RETENTION COMPARISONS

For DINOv2, last-block-only distillation gave 69 successes and mean kNN change -6.60 points, versus 70 and -6.90 for the main configuration. Switching from the main configuration to last-block-only gave a paired $+3.7$ points ($p = 0.5811$) on their shared scenes, despite the opposite one-trial difference in raw counts. In CLIP, adding CLS and final-LN updates instead reduced successes from 49 to 19, a paired -33.7 points ($p < 0.0001$). The register-equipped model had 30 frozen successes and 32 after action-only adaptation (paired effect 0.0, $p = 1$), compared with 63 under distillation. Its visual update set comprises the last block, input CLS token and final LayerNorm; the four register tokens remain fixed.

Retention beyond the main configuration. On DINOv2, action-only adaptation with b_{11}, c, LN changed mean kNN accuracy by $+0.37$ points. Distillation at b_8 with only that block updated gave 23 successes and a -30.68-point kNN change. Distillation at both b_5 and b_{11} with both blocks updated gave 69 successes and -61.52 points, compared with 69 and -6.60 for last-block-only distillation. Thus, similar task success did not imply similar preservation of pretrained features. Full fine-tuning of CLIP reduced success by 21.8 points ($35/87 \to 16/87$, $p = 0.0013$).

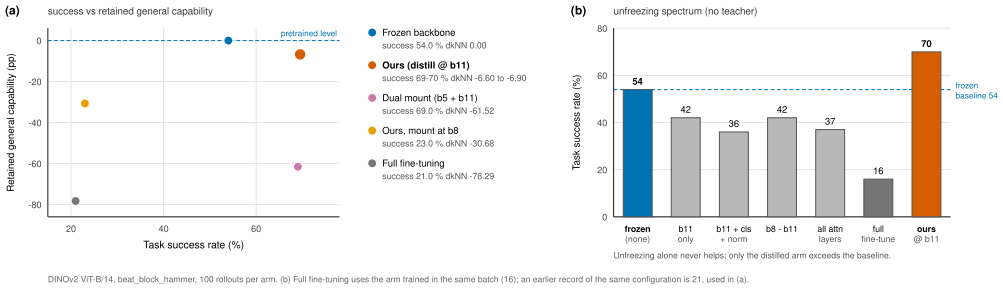


Figure 7: **Task performance and pretrained feature retention.** DINOv2 on the hammer task: (a) adaptation choices; (b) action-only update sets, with SGSD as reference. Success is unpaired; kNN changes are in percentage points. Full fine-tuning uses distinct 21-success and 16-success runs in (a) and (b), respectively.

D.1 TRAINING SEEDS, POLICY READOUT AND DOMAIN SHIFT

Table 10: **Independent training seeds.** Each row compares action-only adaptation with SGSD under the same visual update set. Paired effects use shared scenes within that seed.

Seed	Raw counts /100	Paired (n)	Δ	p
0	$36 \to 70$	$31 \to 64$ (88)	+37.5	< 0.0001
1 (same machine)	$24 \to 55$	$21 \to 52$ (89)	+34.8	< 0.0001
2 (same machine)	$38 \to 73$	$35 \to 70$ (87)	+40.2	< 0.0001

The same seed1 checkpoint evaluated on another machine gave 60 successes, with $24/88 \to 56/88$ paired successes and a 36.4-point gain. This is a repeat evaluation of seed1, not an additional training seed. Across the three reported training seeds, the mean paired gain is 37.5 percentage points with a sample standard deviation of 2.7 points ($n = 3$ training seeds). This summary uses unrounded within-seed, same-machine paired gains; it does not pool scene counts or average the two evaluations of seed1.

Replacing ACT's spatial readout with global pooling gave 23 frozen successes and 38 under distillation, with $21/89 \to 37/89$ paired successes ($+18.0$ points, $p = 0.0113$). For the frozen policy, changing full readout to pooling gave $47/86 \to 22/86$ (-29.1 points, $p < 0.0001$). The pooled and full-readout distilled policies had scene-level outcome correlation $r = 0.013$ on their shared scenes. These results show that spatially corresponding supervision can benefit a pooled policy, without implying identical downstream mechanisms.

Randomized-domain evaluation gave 0 frozen successes and 4 for SGSD. On 84 shared scenes, the paired difference was 4.8 points ($p = 0.1250$).

E WHERE AND WHAT TO DISTILL

E.1 SUPERVISION PLACEMENT AND TRAINABLE PARAMETERS

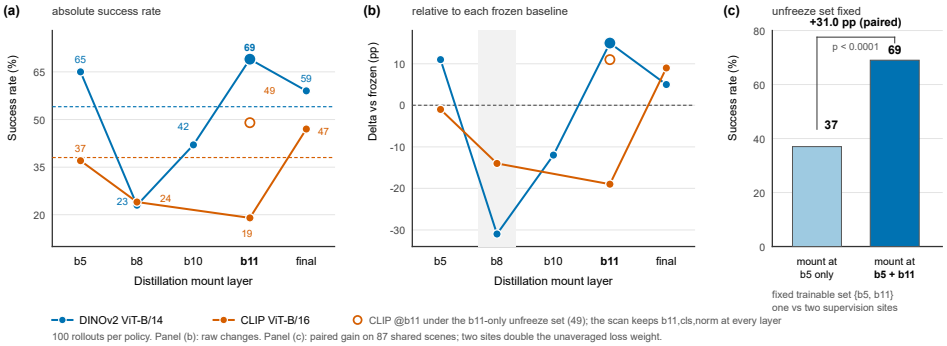


Figure 8: **Supervision-location comparisons.** (a) Raw success; dashed baselines are 54% (DINOv2) and 38% (CLIP); *final* denotes post-LN. Open CLIP markers update b_{11} only; filled markers also update CLS and final LN. (b) Raw changes from frozen baselines. (c) Fixed $\{b_5, b_{11}\}$ update set; the bracket reports the paired gain ($34 \rightarrow 61$ on 87 shared scenes), whereas bars show raw rates. Using both sites also doubles the unaveraged loss weight.

When supervision and the single trainable block were moved together to b_5, b_8, b_{10}, b_{11}, anchor successes were 65, 23, 42 and 69. Last-block distillation exceeded b_8 by 52.4 paired points ($p < 0.0001$) and b_{10} by 31.4 points ($p = 0.0002$); its 4.6-point difference from b_5 was not significant ($p = 0.5413$). An independently retrained post-LN distillation configuration gave 59 successes; moving to pre-LN last-block-only distillation gave a paired 7.9-point increase ($p = 0.0923$). The 59-success and 70-success models therefore refer to different supervision configurations.

With the trainable set fixed to $\{b_5, b_{11}\}$, changing supervision from b_5 alone to both blocks raised raw successes from 37 to 69 and paired successes from 34 to 61 on 87 shared scenes ($+31.0$ points, $p < 0.0001$). Parameter count is fixed, but the two unaveraged site losses approximately double total distillation strength, coupling supervision placement with its overall weight. At b_8, paired changes relative to the corresponding frozen baselines were $-31.8, -13.2, -20.2$ points for DINOv2, CLIP and ViT; the CLIP comparison had $p = 0.0576$. This trend does not imply that every intermediate block is unsuitable.

Local correction and downstream propagation. We examine the supervised block and the final block of the same intermediate-supervision models across three pretraining regimes (Table 11). Distillation at b_8 produces foreground-dominant norm responses at the supervised site, but this contrast weakens substantially by b_{11}. For DINOv2, last-block norm-hotspot overlap remains high even as mean norm contrast weakens; for CLIP and supervised ViT, both diagnostics deteriorate downstream. Correcting an intermediate representation therefore does not ensure that its spatial structure reaches the policy interface. In the DINOv2 single-block comparison, direct b_{11} supervision instead yields a final-block FNR of 2.9906 and 69 successes, compared with 1.1511 and 23 successes for b_8 supervision.

E.2 TEACHER CORRESPONDENCE AND TASK MATCHING

Image and spatial correspondence. Spatial permutation preserves the set of teacher feature vectors while disrupting their locations; cross-image permutation preserves batch-level teacher statistics while disrupting image pairing. The corresponding hammer-task models achieved 46 and 38 successes, compared with 70 for correctly aligned supervision. Against action-only adaptation, the wrong-image

Table 11: **Spatial responses after intermediate-block supervision.** All models are supervised at b_8. Diagnostics are measured before final LayerNorm; success uses 100 rollouts.

	Norm FNR		Norm-hotspot IoU		Success
Backbone	b_8	b_{11}	b_8	b_{11}	(%)
DINOv2	3.1642	1.1511	.4469	.4314	23
CLIP	3.1110	.9494	.4409	.1411	24
Supervised ViT	3.0562	1.0024	.4416	.1401	1

teacher changed paired successes from 34 to 35 on 91 shared scenes (+1.1 points, $p = 1.0000$). Thus, a teacher's marginal feature statistics alone did not reproduce the benefit of image-aligned supervision in this control.

Task-matched versus transferred teacher. We tested whether a teacher trained on `beat_block_hammer` could replace the dedicated teacher for `pick_dual_bottles`. Both teachers supplied features for the target task's own images, preserving image correspondence while changing the teacher's training task. All three policies were evaluated on the same 100 scenes (Table 12). The task-matched teacher improved over freezing by 20.0 points; the transferred teacher's 3.0-point difference was not significant. The direct comparison favored the task-matched teacher by 17.0 points ($p = 0.0015$).

Table 12: **Teacher transfer on dual-bottle manipulation.** All comparisons use the same 100 scenes; Δ is in percentage points and p is the unadjusted exact McNemar value.

Comparison	Successes /100	Δ	p
Frozen → task-matched teacher	43 → 63	+20.0	.0022
Frozen → transferred teacher	43 → 46	+3.0	.7359
Task-matched → transferred teacher	63 → 46	-17.0	.0015

Stronger foreground norm concentration was insufficient. The task-matched teacher had almost equal foreground and background mean norms (FNR 0.9843), and its norm-hotspot IoU was 0.1603, or 0.70 times the random reference. The transferred teacher had FNR 7.5181 and hotspot IoU 0.5494, or 2.43 times the reference (Table 13). The teacher with more foreground-concentrated norms therefore produced the weaker policy. These measurements describe intermediate `up4` feature norms, not the accuracy of the teacher's segmentation predictions. Both diagnostic records use dual-bottle images, but their foreground fractions differ slightly; they are not an identical-frame paired localization comparison.

Table 13: **Teacher feature norms on dual-bottle images.** φ is foreground fraction; the reference is $\varphi/(2-\varphi)$. Measurements use the teacher's `up4` features.

Teacher training task	φ	Hotspot IoU	IoU / reference	FNR
Dual bottles	.3742	.1603	0.70×	.9843
Hammer	.3692	.5494	2.43×	7.5181

Prospective decision rule and scope. The experiment ledger recorded a qualitative decision rule while the transfer run was in progress: a gain clearly above the task-matched teacher's 20-point gain would favor foreground norm concentration as a useful criterion; a comparable gain would support omitting task-specific teacher training; and a smaller gain would indicate a transfer cost. Thus, the design allowed the outcome that a shared teacher could replace the task-specific one. The observed result fell in the third category. This was a prospective qualitative rule.

Together, the correspondence and transfer controls support matching the teacher to both the observed image and its task distribution, rather than selecting teachers solely by foreground norm concentration. The transfer evidence comprises one target task and one direction, hammer to dual bottles; the reverse

direction and replication on other target tasks were not evaluated. It therefore establishes the benefit of the task-matched teacher in this comparison, not a universal requirement for separate teachers or the impossibility of a shared teacher. A nonsignificant gain over freezing does not establish equivalence to the frozen policy. Testing a jointly trained teacher and additional transfer directions would assess how broadly this task-matching result holds.

F MECHANISTIC DIAGNOSTICS AND THEIR LIMITS

Table 14: **Representation changes on 600 hammer-task frames.** Direction and magnitude are relative token distances from frozen features; CLS displacement measures parameter change.

Configuration	Success (%)	Direction	Magnitude	Cosine to frozen	CLS displacement
Frozen	54	0	0	1.000	0
Action-only	36	.940	.075	.550	.189
Wrong-image teacher	38	1.295	.602	.160	.824
Shuffled teacher positions	46	1.318	.617	.130	.977
Feature MSE only	47	1.305	.665	.147	1.173
Without direction loss	58	1.322	.635	.125	1.225
Complete loss	**70**	1.329	.549	.116	1.281

Table 15: **Post-LN inference interventions.** Comparisons are unperturbed → perturbed. SGSD norm suppression/injection uses the last-block-only model; rotation/scaling uses the main model. Token groups use predicted teacher masks; p values are unadjusted exact paired tests.

Policy	Intervention	Paired (n)	Δ	p
Frozen	High-norm suppression, $\sigma = 2$	$47 \rightarrow 44$ (87)	-3.4	.6636
Frozen	High-norm injection, $\sigma = 2$	$44 \rightarrow 41$ (87)	-3.4	.6636
SGSD	High-norm suppression, $\sigma = 2$	$60 \rightarrow 56$ (84)	-4.8	.5235
SGSD	High-norm injection, $\sigma = 2$	$61 \rightarrow 59$ (85)	-2.4	.8145
Frozen	Background tokens zeroed	$47 \rightarrow 45$ (87)	-2.3	.8450
Frozen	Foreground tokens zeroed	$45 \rightarrow 37$ (85)	-9.4	.0963
Frozen	Random tokens zeroed	$48 \rightarrow 41$ (89)	-7.9	.1435
SGSD	Norm-preserving rotation, dose .10	$60 \rightarrow 60$ (83)	0.0	1.0000
SGSD	Direction-preserving scaling, dose .10	$65 \rightarrow 61$ (87)	-4.6	.3877

Frozen weights need not produce fixed activations. In the main configuration, 154 parameter tensors in b_0–b_{10}, together with patch and position embeddings, remained identical to pretrained values. The trainable input CLS token nevertheless interacts with patches through those blocks. The recorded directional change at frozen b_8 was 0.558 under SGSD and 0.132 under action-only adaptation, consistent with that upstream path. CLS is not read directly by the policy. Its participation in feature propagation does not establish that it causes all behavioral gains, especially given the strong last-block-only DINOv2 result and CLIP's different update preference.

Across the six adapted configurations in Table 14, CLS and last-block weight displacement each had Spearman $\rho = 1.000$ with success (exact permutation $p = 0.0028$). Direction change had $\rho = 0.943$ ($p = 0.0167$); magnitude change had $\rho = 0.371$ ($p = 0.4972$). Within the five distilled configurations, the corresponding direction and magnitude correlations were 0.900 ($p = 0.0833$) and -0.100 ($p = 0.9500$).

Across 16 adaptation/placement configurations with at least one available diagnostic, last-block FNR, norm-hotspot IoU, background outlier share and probe IoU correlated with success at $\rho = 0.467, 0.404, -0.091, 0.420$, respectively ($p = 0.0809, 0.1352, 0.7573, 0.1191$; effective $n = 15, 15, 14, 15$ configurations). Background outlier share is undefined for the configuration with no detected outliers and is omitted from that correlation. A full-fine-tuning replication had post-LN Patch Score IoU 0.3862, close to SGSD's 0.4158, but only 16 successes. The main full-fine-tuning comparison had 21 successes.

Layerwise patterns. DINOv2's minimum norm FNR occurred at b_8 (0.6650); CLIP's at b_6 (0.7594); supervised ViT's at b_6/b_9 (approximately 0.9388). Using foreground/background mean Patch Score instead moved the respective minima to b_1, b_0, b_9. Cosine-score ratios are undefined when the background mean is zero and difficult to interpret across sign changes. The layer of the strongest anomaly is therefore metric-dependent, and does not by itself prescribe an effective distillation location.

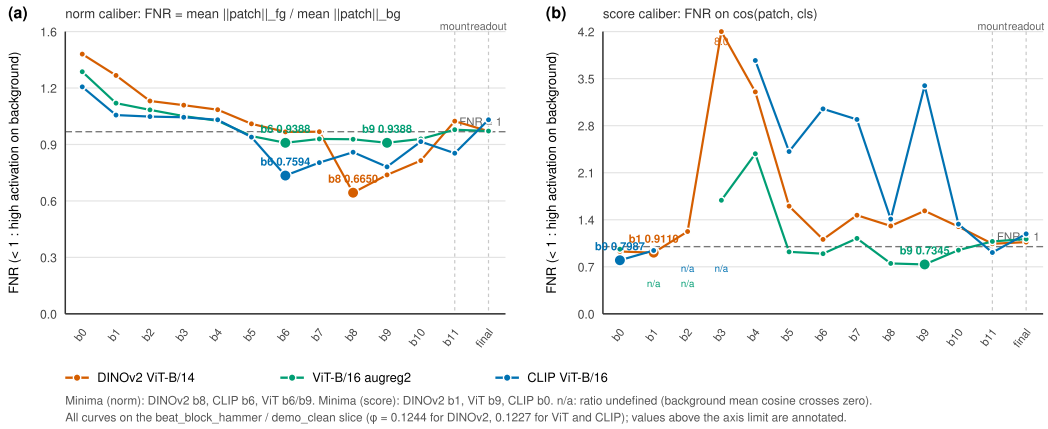


Figure 9: **Layerwise spatial diagnostics across pretraining regimes.** Left: foreground/background mean feature norm. Right: foreground/background mean CLS–patch cosine (Patch Score). The final block and post-LN readout are measured separately; undefined ratios are marked unavailable.

G Spatial diagnostics and supplementary visualizations

G.1 Backbone and policy-input features

Single-frame visualization and aggregate diagnostics. Figure 4 displays one hammer-task frame at the pre-LN final block. Its frozen, action-only and SGSD FNR values are 1.05, 1.17 and 2.75, respectively; Table 16 summarizes the diagnostic collection. In the displayed frame, SGSD's maximum/mean patch norm ratio is 5.4, compared with 4.1 for the frozen backbone. Improved spatial separation therefore coexists with a larger peak-to-mean ratio. The independently scaled heatmaps visualize the spatial distribution of responses, rather than absolute norm differences between models.

Table 16: **Spatial diagnostics on the hammer task.** Norm IoU is foreground overlap of norm hotspots; probe IoU measures foreground readability. Success uses 100 rollouts. The b_8 control changes both the supervised and trainable block.

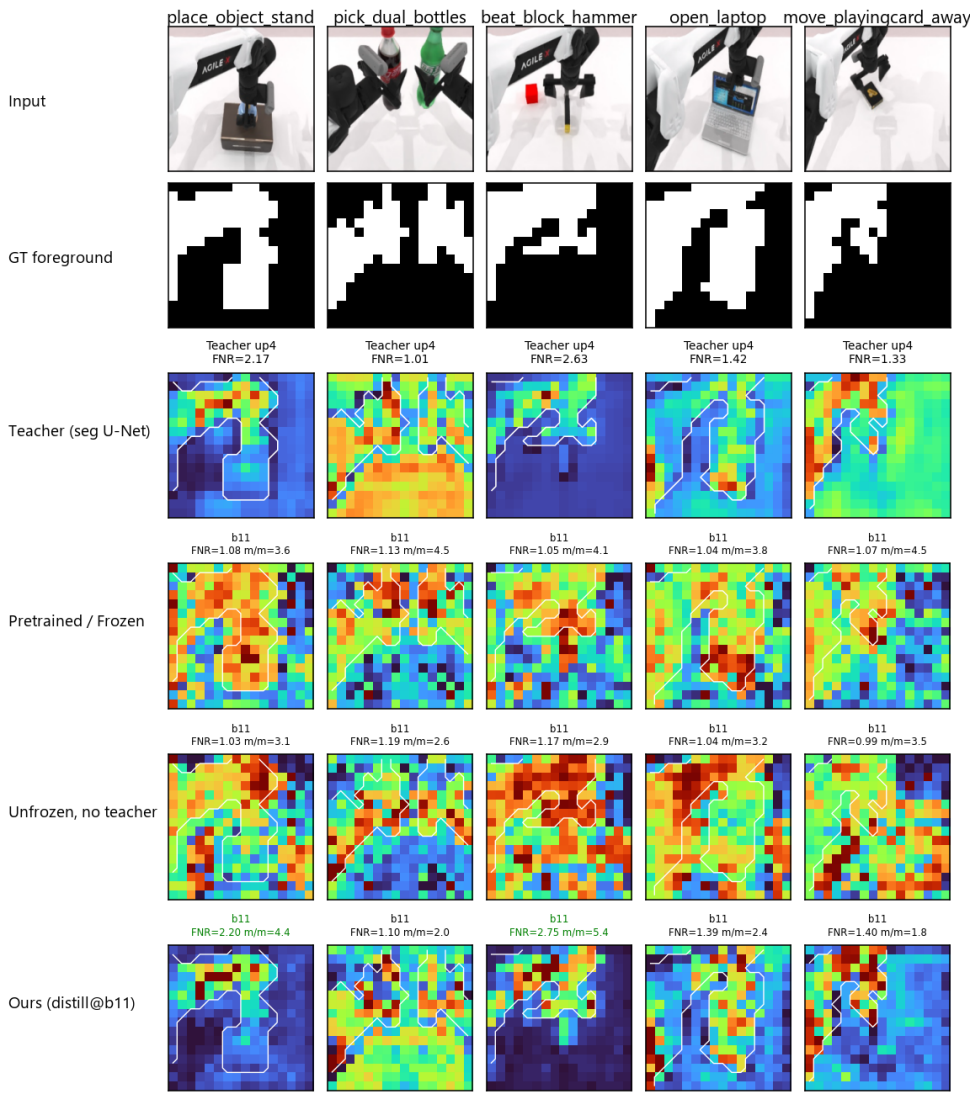
	Last block (b_{11})			Post-LN	Success
Configuration	FNR	Norm IoU	Probe IoU	Probe IoU	/100
Frozen	1.0582	.1540	.7804	.7736	54
Action-only adaptation	.9513	.0999	.7456	.7404	36
SGSD	2.9846	.4448	.8476	.8128	70
Distillation at b_8	1.1511	.4314	.8494	.8281	23

Figure 10 places backbone feature maps alongside closed-loop success on five tasks. The object-stand and hammer examples show stronger foreground-to-background norm contrast under SGSD than under either frozen or action-only adaptation. The corresponding success rates rise from 10% to 34% and from 54% to 70%, respectively. These observations connect correction in the backbone itself with improved task performance, complementing the projection-based comparison in Figure 2.

Aggregate diagnostics on the hammer task quantify this change (Table 16). Action-only adaptation lowers last-block norm-hotspot IoU from 0.1540 to 0.0999 and probe IoU from 0.7804 to 0.7456.

SGSD raises them to 0.4448 and 0.8476, respectively. Probe scores summarize the diagnostic split without repeated-probe uncertainty estimates.

The improvement also reached the normalized features actually read by the policy: probe IoU increased from 0.7736 to 0.8128 and Patch Score hotspot IoU from 0.1562 to 0.4158. Norm-hotspot IoU at this same readout instead fell from 0.0814 to 0.0019. This divergence matters because LayerNorm alters the relation between feature scale and content (Ba et al., 2016). Equally, the b_8 configuration achieved a higher probe score but much lower task success. Spatial readability improved under SGSD. Appendix F reports complementary inference interventions and layerwise analyses that examine the link between these diagnostics and behavior.



Success (%)	Stand	Bottles	Hammer	Laptop	Card
Frozen	10	43	54	69	38
Action-only	13	40	36	73	34
SGSD	34	63	70	83	50

Figure 10: **Backbone responses and task success.** Columns show five tasks; rows show the image, foreground mask, teacher, frozen backbone, action-only adaptation and SGSD. Student heatmaps show pre-LN b_{11} feature norms with foreground outlines; annotations describe individual frames, with independently scaled colors. Success rates below use 100 rollouts per policy, in the same task order.