The Locality Trade-off in Masked Patch Self-Distillation

Xi Weng[1], **Zhaoyu Liu**[1], **Lianyu Hu**[2], **Jin Song Dong**[1]*
[1]School of Computing, National University of Singapore
[2]Nanyang Technological University

Abstract

Masked patch self-distillation is a key component of modern vision foundation models, including DINOv2 and DINOv3, yet its implications for local visual information remain unclear. We show that its semantic benefits can coexist with a scale-dependent locality cost. Using two annotation-free diagnostics, we distinguish recoverable local appearance from spatial correspondence. In smaller models, adding iBOT substantially weakens patch correspondence while largely preserving appearance decodability. In larger models, correspondence improves, but pixel and texture evidence becomes less linearly accessible in deep features. Thus, scaling changes the form of the locality cost rather than uniformly eliminating it. Our analysis of Gram anchoring further shows that preserving patch relations does not guarantee simultaneous improvements in both properties. These findings motivate evaluating and designing visual representations around the complementary demands of semantic abstraction, appearance preservation, and spatial correspondence. Our code is available at `https://github.com/winci-ai/jea`

1 Introduction

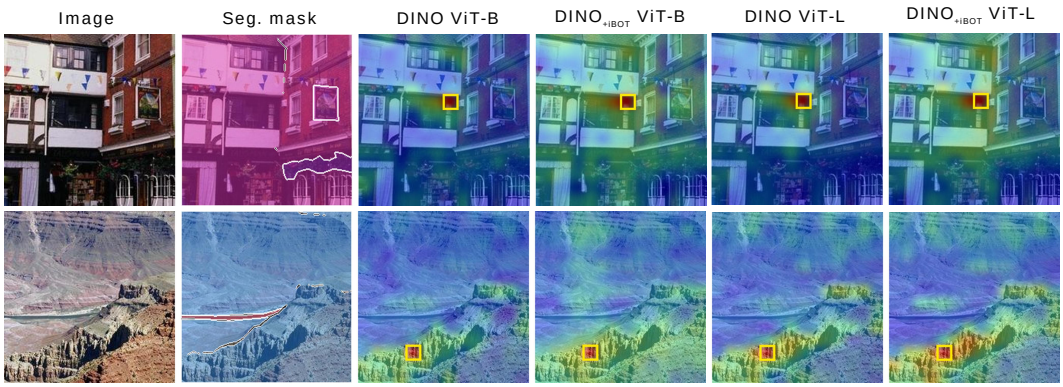

Figure 1: Anchor-patch similarity heatmaps for DINO and DINO$_{+\mathrm{iBOT}}$ on ADE20K examples, with semantic annotations shown for reference. Yellow boxes mark the anchor patches. For ViT-B, adding iBOT produces broader responses over the building and mountain, reducing distinctions associated with windows and rugged cliff structures. This effect is alleviated in ViT-L, where the +iBOT model also exhibits more spatially selective responses to these structures. The semantic masks do not resolve many of these within-region differences, illustrating why segmentation performance alone cannot characterize fine-grained local representations. Additional examples appear in Figure 10.

*corresponding author: dongjs@comp.nus.edu.sg

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Self-supervised learning has become a central approach to building general-purpose visual representations [5, 8–11, 20, 21, 24, 37–39]. From DINO [9] to DINOv2 [28] and DINOv3 [33], learning from unlabeled images has produced visual backbones that transfer across recognition, dense prediction, and geometric understanding. Their influence also extends to larger systems: DINOv2 contributes visual features to multimodal models such as Cambrian-1 [35] and serves as the backbone of Depth Anything V2 [41]. As these representations become shared components across applications, understanding which visual properties their training objectives preserve becomes increasingly important.

A key ingredient in this development is masked patch self-distillation. Introduced by iBOT [43] and adopted in DINOv2 and DINOv3, this objective trains masked student patches to predict teacher-generated targets. It extends self-distillation to patch tokens, enabling supervision at a finer spatial granularity. However, a patch-level objective does not necessarily preserve all forms of local evidence. Because its targets are learned representations rather than raw pixels, successful prediction may favor semantic consistency while becoming less sensitive to appearance differences within a semantic region. This raises a fundamental question: *what does masked patch self-distillation preserve locally, and how does the answer change with model scale?*

Figure 1 illustrates this question using images from ADE20K [42]. For ViT-B, DINO produces similarity responses that distinguish structures such as windows and rugged cliffs, whereas adding iBOT yields broader responses over the surrounding building or mountain. These differences can be obscured by semantic segmentation evaluation: human-annotated masks need not distinguish the visual structures contained within a labeled region. Interestingly, this qualitative loss of spatial specificity is alleviated in ViT-L, where the +iBOT model also exhibits responses that follow fine local structures. Yet sharper similarity maps alone cannot establish whether the underlying pixel and texture evidence remains recoverable.

To investigate this distinction, we compare DINO and $\text{DINO}_{+\text{iBOT}}$ across four backbone scales under matched training settings within each scale. The downstream results reveal a consistent semantic advantage alongside a more heterogeneous effect on local tasks. Adding iBOT generally improves tasks dominated by semantic understanding, while often reducing performance on tasks that require precise local distinctions. However, this pattern is not universal: on DAVIS video object tracking, the disadvantage observed in smaller models disappears and reverses slightly in larger models. Together with Figure 1, this suggests that a single notion of "local information" is insufficient to explain the effects of patch self-distillation.

We therefore introduce two complementary annotation-free diagnostics. *Local Appearance Decodability* measures how much pixel and texture evidence is linearly recoverable from frozen patch tokens. *Local-to-Global Correspondence* measures whether patches remain spatially distinguishable and matchable across views. These diagnostics reveal a scale-dependent dissociation. For ViT-S and ViT-B, iBOT substantially weakens correspondence while leaving appearance decodability largely unchanged or slightly improved. For ViT-L and ViT-H, correspondence is preserved or enhanced, but appearance decodability declines markedly in deep layers. The improved spatial specificity of larger models therefore does not imply that all forms of local evidence are preserved. Our results instead suggest that model scale changes which local properties are most affected by semantic patch supervision.

Finally, we examine Gram anchoring, introduced in DINOv3 [33] to stabilize dense features by regularizing pairwise patch similarities. We compare anchoring initialized from an earlier $\text{DINO}_{+\text{iBOT}}$ checkpoint with a variant initialized from DINO. The resulting changes again separate appearance from correspondence: the DINO-based anchor substantially improves correspondence in ViT-B, whereas in ViT-L it improves deep appearance decodability without consistently improving correspondence. These results suggest that the local properties encouraged by relational regularization depend on both the anchor representations and the model scale.

Our contributions are threefold:

- We characterize the scale-dependent locality cost of masked patch self-distillation, showing that its semantic benefits coexist with distinct changes in local visual abilities.
- We introduce two annotation-free diagnostics that separate local appearance decodability from spatial correspondence, revealing differences that downstream semantic evaluation and similarity visualizations alone cannot resolve.

- We analyze Gram anchoring through this distinction, showing that improvements in patch relations and recoverable appearance need not occur together.

2 Background, Related Work, and Notation

In this section, we provide a detailed description of the class-token joint-embedding architecture and masked patch self-distillation. Additional related work is discussed in Appendix H.

2.1 Class-Token Joint-Embedding Self-Distillation

Let $\mathcal{B} = \{x_i\}_{i=1}^{B}$ be a mini-batch of unlabeled images sampled from the data distribution $\mathcal{D}$. For each image x_i, a multi-crop augmentation pipeline [8] generates global and local views, denoted by $\mathcal{V}_i^g$ and $\mathcal{V}_i^\ell$, respectively. Following DINO [9], the student processes all views, whereas the teacher processes only the global views. We denote their respective crop sets by

$$\mathcal{C}_i = \mathcal{V}_i^g \cup \mathcal{V}_i^\ell, \qquad \mathcal{G}_i = \mathcal{V}_i^g. \tag{1}$$

Given a crop w, a vision transformer encoder [14] produces a class token and a sequence of patch tokens:

$$f_\theta(w) = \big(c_\theta(w), z_\theta(w,1), \ldots, z_\theta(w,P_w)\big), \tag{2}$$

where $c_\theta(w)$ denotes the class-token representation, $z_\theta(w,r)$ denotes the r-th patch-token representation, and P_w is the number of patch tokens in w. Let $a_i(u)$ denote the student input for crop u: it applies the sampled patch mask to global crops when iBOT is enabled and is the identity otherwise. Local crops and teacher inputs remain unmasked. The class-token representations are mapped to K-dimensional logit vectors by the student and teacher projection heads:

$$\boldsymbol{s}_i^u = g_\theta^{\mathrm{cls}}\big(c_\theta(a_i(u))\big), u \in \mathcal{C}_i, \quad \boldsymbol{t}_i^v = g_{\bar\theta}^{\mathrm{cls}}\big(c_{\bar\theta}(v)\big), v \in \mathcal{G}_i. \tag{3}$$

Here, θ and $\bar\theta$ collectively denote the parameters of the student and teacher encoders and projection heads, respectively. The teacher is initialized from the student and receives no gradient updates. Instead, after each student update, its parameters are updated by an exponential moving average:

$$\bar\theta \leftarrow \mu\bar\theta + (1-\mu)\theta, \tag{4}$$

where $\mu \in [0,1)$ is a momentum coefficient, typically increased toward 1 during training.

We use DINO [9] as the class-token self-distillation objective. DINO encourages agreement between different views of the same image by matching student predictions to stop-gradient teacher targets. The original method uses centering and sharpening of teacher outputs to help avoid collapse. To control for differences in the class-token objective when comparing DINO with DINO$_{+\mathrm{iBOT}}$, we instead construct teacher targets using Sinkhorn–Knopp normalization, following SwAV [8] and DINOv2 [28], while retaining temperature-controlled sharpening. Details of the Sinkhorn–Knopp target construction are provided in Appendix A.

2.2 Masked Patch Self-Distillation: iBOT

iBOT [43] complements class-token self-distillation with a masked patch prediction objective. The student predicts teacher-generated target distributions at masked patch positions, using the corresponding unmasked crop as the teacher input. Thus, the prediction targets are learned patch-level distributions rather than raw pixels.

For each global crop $u \in \mathcal{G}_i$, let $\mathcal{M}_i^u \subseteq \{1,\ldots,P_u\}$ denote the selected masked patch indices. We write $\widetilde{u}$ for the student input obtained by replacing the selected patch embeddings with a learnable mask token, so that $a_i(u) = \widetilde{u}$. The class and patch losses use the same masked student forward pass. The teacher receives the unmasked crop u. At each masked position $r \in \mathcal{M}_i^u$, the student and teacher patch heads produce logits

$$\boldsymbol{a}_{i,r}^{s,u} = h_\theta^{\mathrm{patch}}\big(z_\theta(\widetilde{u},r)\big), \quad \boldsymbol{a}_{i,r}^{t,u} = h_{\bar\theta}^{\mathrm{patch}}\big(z_{\bar\theta}(u,r)\big), \tag{5}$$

where both logit vectors belong to $\mathbb{R}^{K_p}$, and K_p is the output dimension of the patch heads. Since the student and teacher inputs share the same crop geometry, the index r identifies the same spatial position in both networks.

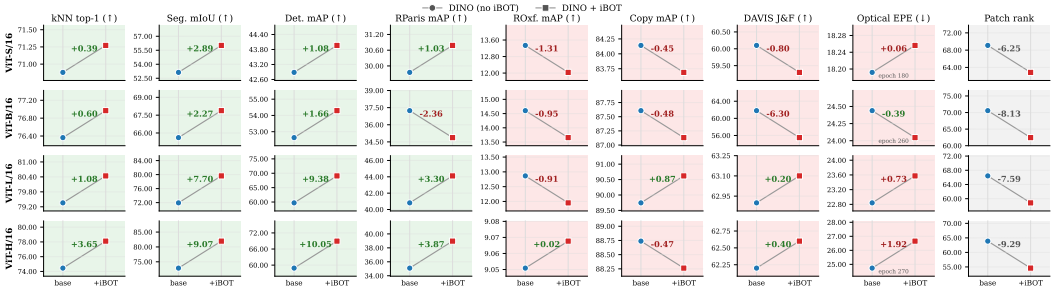

Figure 2: Downstream comparison of DINO and DINO$_{+\text{iBOT}}$ across ViT-S/16, ViT-B/16, ViT-L/16, and ViT-H/16, pretrained on ImageNet-1K for 300 epochs under matched settings within each scale. Adding iBOT generally benefits tasks dominated by semantic understanding, whereas its effects on tasks requiring fine-grained local information vary with model scale. In particular, the degradation in DAVIS tracking observed for smaller backbones reverses to modest gains for larger backbones. Task scores other than optical-flow EPE are percentages; normalized patch rank is multiplied by 100. Arrows indicate the preferred direction for task metrics: ↑ denotes higher is better and ↓ denotes lower is better. Patch effective rank measures spectral diversity. Numerical results and training trajectories are provided in Table 2 and Figure 11, respectively.

The student patch prediction is $\boldsymbol{p}_{i,r}^{s,u} = \operatorname{softmax}\left(\frac{\boldsymbol{a}_{i,r}^{s,u}}{\tau_s^p}\right)$, where $\tau_s^p > 0$ is the student patch temperature. Following DINOv2 [28], we construct teacher patch targets using Sinkhorn–Knopp normalization. Specifically, let $\boldsymbol{A}_{\mathcal{B}}^t$ stack the teacher patch logits at all selected masked positions in the mini-batch. The corresponding target matrix is $\boldsymbol{Q}_{\mathcal{B}}^t = \operatorname{sg}\left[\operatorname{SK}\left(\exp\left(\frac{\boldsymbol{A}_{\mathcal{B}}^t}{\tau_t^p}\right)\right)\right]$, where $\tau_t^p > 0$ is the teacher patch temperature, the exponential is applied elementwise, and SK uses the same normalization convention as in Appendix A. Each row provides the target distribution $\boldsymbol{q}_{i,r}^{t,u}$ for its corresponding patch. Normalization is performed jointly across the selected teacher patches, and all teacher targets are detached from gradient computation.

Defining $H(\boldsymbol{q}, \boldsymbol{p}) = -\sum_{k=1}^{K_p} q_k \log p_k$, the masked patch self-distillation loss is

$$\mathcal{L}_{\text{iBOT}} = \frac{1}{B}\sum_{i=1}^{B}\frac{1}{|\mathcal{G}_i|}\sum_{u\in\mathcal{G}_i}\frac{\sum_{r\in\mathcal{M}_i^u} H\left(\boldsymbol{q}_{i,r}^{t,u}, \boldsymbol{p}_{i,r}^{s,u}\right)}{\max\left\{1, |\mathcal{M}_i^u|\right\}}. \tag{6}$$

The loss is evaluated only at masked positions and averaged over global crops and images. Crops with no masked positions contribute zero. The student encoder and patch head are optimized by back-propagation, while their teacher counterparts are updated by exponential moving average.

3 The Downstream Signature of iBOT's Locality Cost

In this section, we compare DINO and DINO$_{+\text{iBOT}}$ on ViT-S/16, ViT-B/16, ViT-L/16, and ViT-H/16, all pretrained on ImageNet-1K [13] for 300 epochs following the DINOv2 recipe [28]. Within each scale, we keep the remaining training settings identical and enable or disable the iBOT branch together with its associated student masking.

Our evaluations cover complementary representation capabilities. ImageNet k-NN classification [9] primarily measures global semantic discrimination; VOC2012 segmentation [17] and VOC2007 detection [16] require semantic recognition together with spatial localization. RParis and ROxford retrieval [30] evaluate instance discrimination using both global structure and distinctive local cues, while Copydays copy detection [15] emphasizes visual identity under image transformations. DAVIS label propagation [9, 29] tests cross-frame patch matching and instance consistency, and Sintel optical flow [6] probes motion estimation through a learned head. Our Sintel scores measure fitting on training videos, not held-out generalization. We additionally report patch effective rank as a measure of spectral diversity [19, 31]. Benchmark interpretations and evaluation details are provided in Appendix B.1 and Appendix B.2.

Figure 2 reveals a recurring contrast: DINO$_{+\text{iBOT}}$ improves classification, segmentation, and detection across all scales, whereas DINO more often performs better on tasks emphasizing local visual

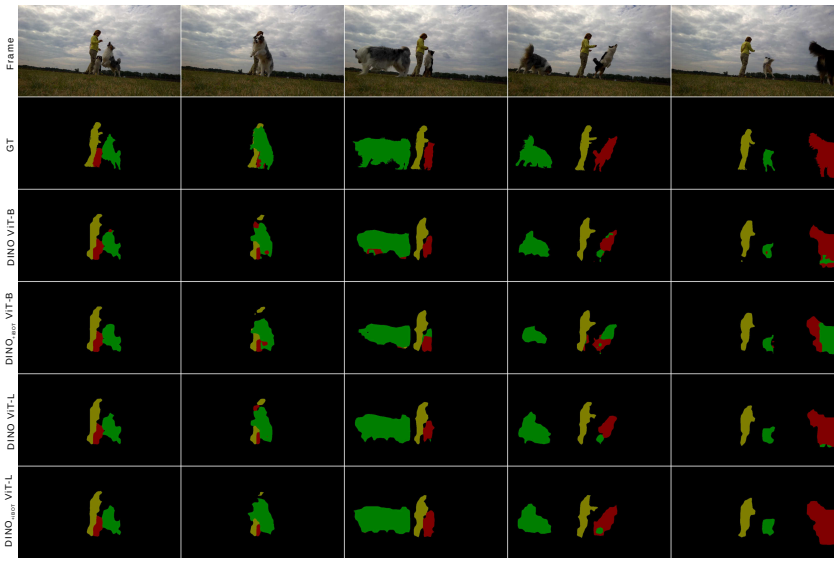

Figure 3: DAVIS video object tracking visualization. Here we compare tracking masks produced by DINO and DINO$_{+\text{iBOT}}$ on videos containing visually similar same-category instances and occlusions. More visualization results can be seen in Figure 12.

distinctiveness, including Copydays and ROxford retrieval. RParis generally benefits from iBOT, motivating separate reporting of the two retrieval benchmarks. However, the locality-sensitive results are not uniform. Most notably, iBOT reduces DAVIS performance on ViT-S/B but modestly improves it on ViT-L/H. Since our DAVIS evaluation relies on patch similarities for label propagation, this exception suggests that iBOT may improve patch correspondence in larger models. Patch effective rank decreases at every scale, showing that reduced spectral diversity need not imply weaker matching.

Figure 3 further visualizes DAVIS tracking predictions from DINO and DINO$_{+\text{iBOT}}$. With ViT-B/16, the +iBOT model exhibits two representative failure modes. First, when visually similar instances of the same category, such as two dogs, overlap, it is more prone to identity confusion and swapped tracking masks. Second, after an instance becomes occluded, it is less successful at recovering and maintaining the correct track. These failures are alleviated when scaling to ViT-L/16, consistent with the scale-dependent quantitative results. This qualitative evidence further motivates examining how iBOT affects instance-specific matching across model scales.

Overall, the downstream results indicate a recurring locality cost whose form depends on the task and model scale. They cannot determine whether the regressions reflect reduced appearance information, less effective matching geometry, or both. We therefore introduce two complementary annotation-free diagnostics: *Local Appearance Decodability* measures recoverable pixel and texture evidence, whereas *Local-to-Global Correspondence* measures whether patches remain spatially distinguishable and matchable across views.

4 Locating Lost Evidence via Annotation-Free Diagnostics

The downstream results in Section 3 reveal locality-sensitive regressions alongside scale-dependent exceptions, particularly the improved DAVIS performance of larger models. However, these benchmarks combine semantic recognition, appearance discrimination, and spatial matching, making it difficult to isolate which properties change under iBOT supervision. We therefore introduce two complementary annotation-free diagnostics. *Local Appearance Decodability* measures recoverable pixel and texture evidence, whereas *Local-to-Global Correspondence* measures whether patches remain spatially distinguishable and matchable across views. This distinction allows us to examine whether the downstream locality cost reflects changes in appearance accessibility, correspondence, or both.

4.1 Local Appearance Decodability

The first diagnostic measures how much local appearance information can be linearly recovered from frozen patch tokens. Given an image $w \in [0,1]^{3\times H\times H}$, let $z_\theta^{(\ell)}(w,r) \in \mathbb{R}^d$ denote its r-th patch

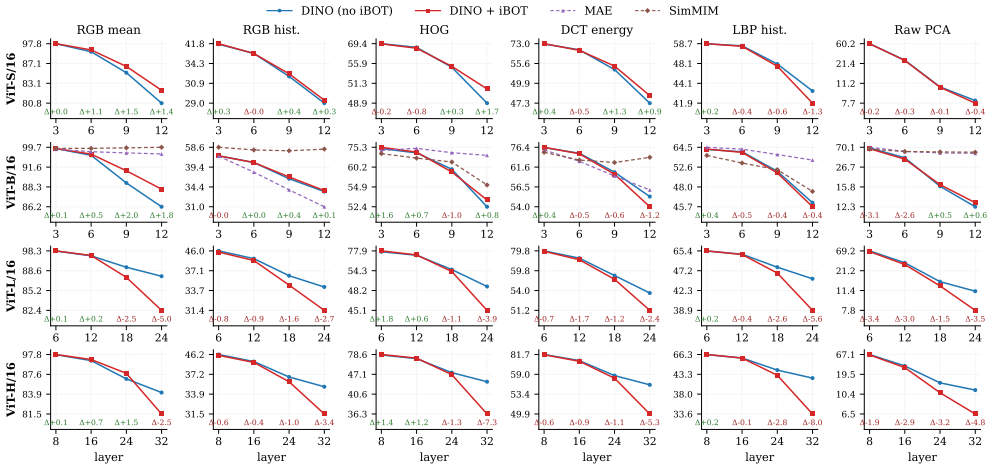

Figure 4: Quantitative results of Local appearance decodability across different layers. Δ denotes the difference between the $\text{DINO}_{+\text{iBOT}}$ and its corresponding DINO model under the same pretraining setup. To validate that the diagnostic indeed captures local visual evidence, we also include raw-pixel prediction models, such as SimMIM and MAE, which are also pretrained using ViT-B/16 on ImageNet. We also report the evolution of each descriptor metric during training in Figure 13, and show the decoding visualization in Figure 14.

token at layer ℓ, and let $\tilde{w}_r \in [0,1]^{3\times p\times p}$ be the spatially aligned RGB patch, where p is the patch size.

We construct an appearance target $y(w,r) = \phi(w,r) \in \mathbb{R}^m$ from RGB mean, RGB histograms, Histogram of Oriented Gradients (HOG), Discrete Cosine Transform (DCT) frequency-band energy, Local Binary Patterns (LBP) histograms, and a low-dimensional raw-pixel PCA projection. These descriptors capture complementary aspects of color, edge orientation, texture, spatial frequency, and pixel variation without semantic annotations. HOG and LBP use image-level neighborhood operations followed by patch-wise pooling; the other descriptors are computed within each aligned raw patch.

We fit a ridge linear decoder from frozen patch tokens to these targets using a probe-training split and evaluate it on held-out images. For each descriptor group g, we report the dimension-averaged coefficient of determination, $R^2_{g,\text{mean}}$. Higher values indicate greater linear accessibility of the corresponding appearance attributes. Descriptor construction, probe fitting, and evaluation details are provided in Appendix F.

We evaluate the 300-epoch backbones from Section 3 at multiple depths, since dense prediction commonly uses intermediate as well as final-layer features [9, 43]. We sample blocks 3/6/9/12 for ViT-S/B, 6/12/18/24 for ViT-L, and 8/16/24/32 for ViT-H. Figure 4 additionally includes ViT-B/16 MAE [22] and SimMIM [40] as pixel-reconstruction references. Their strong deep-layer RGB-mean and raw-PCA scores provide supporting evidence that the diagnostic captures appearance information beyond semantic discrimination.

Figure 4 reveals a clear dependence on model scale and depth. For ViT-S/B, adding iBOT largely preserves appearance decodability, with modest gains or losses depending on the descriptor and layer. In particular, final-layer RGB-mean and HOG scores improve for both backbones. Thus, the smaller models do not exhibit a systematic loss of linearly recoverable appearance information. Their weaker performance on locality-sensitive downstream tasks, including DAVIS, therefore cannot be explained simply by a broad decline in the appearance attributes measured here.

For ViT-L/H, iBOT provides small improvements in selected shallow or intermediate-layer descriptors, including HOG and RGB mean. However, this benefit does not extend to the final representations: all six descriptor scores decrease at the last block, with particularly pronounced losses in HOG and LBP for ViT-H. The appearance cost is therefore concentrated in deeper representations, rather than occurring uniformly throughout the backbone. These results indicate that iBOT makes pixel and texture evidence less linearly accessible in the deep layers of larger models.

Together with the downstream results, this pattern highlights why appearance decodability and correspondence must be examined separately. ViT-S/B retain broadly comparable appearance decodability

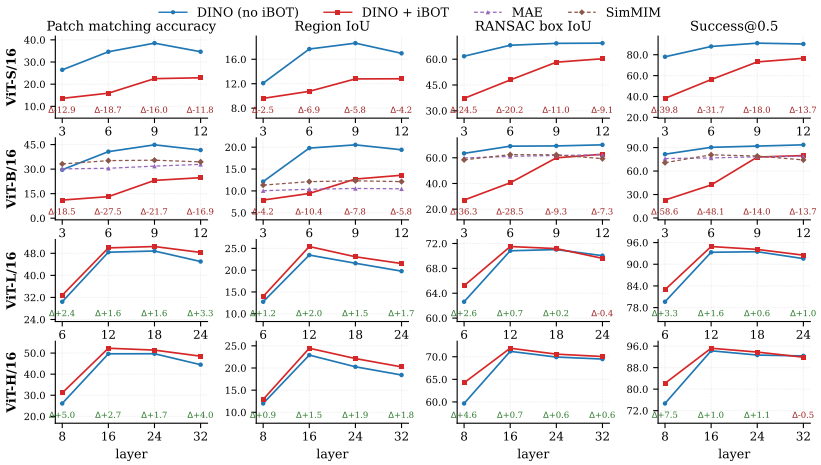

Figure 5: Quantitative results of local-to-global correspondence across different layers. Δ denotes the difference between the DINO$_{+\text{iBOT}}$ model and its corresponding DINO model under the same pretraining setup. We also report the evolution of each metric during training in Figure 16.

despite worse DAVIS tracking, whereas ViT-L/H improve on DAVIS despite reduced deep-layer appearance decodability. This dissociation suggests that recoverable appearance alone does not determine matching quality. We next directly evaluate patch correspondence to test whether its response to iBOT also changes with model scale.

4.2 Local-to-Global Correspondence

The second diagnostic evaluates whether patches remain spatially distinguishable and matchable across views. Given a source image x and a sampled crop box $B^\star$, we construct a local query $x_{B^\star}$ by cropping the image inside $B^\star$. Both the query and the source image are resized to the model input resolution, and patch tokens are extracted from a frozen layer ℓ.

Let $\tilde{z}_\ell^{\text{local}}(i)$ and $\tilde{z}_\ell^{\text{global}}(j)$ denote cosine-normalized patch tokens from the query and source image, respectively. Their matching scores are $A_\ell(i,j) = \tilde{z}_\ell^{\text{local}}(i)^\top \tilde{z}_\ell^{\text{global}}(j)$. The ground-truth source region consists of source patches whose grid centers lie inside $B^\star$.

We report four complementary metrics: **patch matching accuracy**, measuring whether query patches retrieve source patches inside $B^\star$; **region localization IoU**, measuring the overlap between the strongest source responses and the true crop support; **RANSAC geometric box IoU**, measuring the accuracy of the crop box estimated by RANSAC-style translation consensus with known crop dimensions; and **IoU$_{50}$ success rate**, measuring the fraction of estimated boxes reaching at least 0.5 IoU. Higher values indicate better performance under each criterion. Patch matching accuracy tests region membership, not exact pointwise correspondence; region IoU also uses the known reference-region size. Full definitions and sampling details are provided in Appendix G.

Unlike copy detection, which retrieves an image from a database, this diagnostic assumes that the source image is known and asks whether patch similarities can localize the query within it. It therefore isolates spatial matching more directly, without image-level aggregation or database distractors.

Figure 5 reveals a clear scale-dependent effect at epoch 300. For ViT-S/B, adding iBOT substantially reduces all four correspondence metrics across the sampled layers. At the final block and crop area ratio 0.1, patch matching accuracy drops by approximately 11.8 and 16.9 percentage points, respectively. For ViT-L/H, this broad degradation disappears: patch matching accuracy and region IoU improve across all sampled layers, with final-block matching gains of approximately 3.3 and 4.0 points. Geometric metrics generally improve or remain comparable, although small exceptions remain, including final-block box IoU for ViT-L and IoU$_{50}$ success for ViT-H. Thus, the larger-model benefit is clearest in patch matching and region localization.

The training trajectories in Figure 16 show that these differences develop during optimization. In ViT-S/B, several $+$iBOT metrics decline over extended periods of training, particularly toward the end, while DINO generally improves or maintains its correspondence performance. At deeper layers, an early $+$iBOT advantage can consequently turn into a substantial final-checkpoint deficit. These

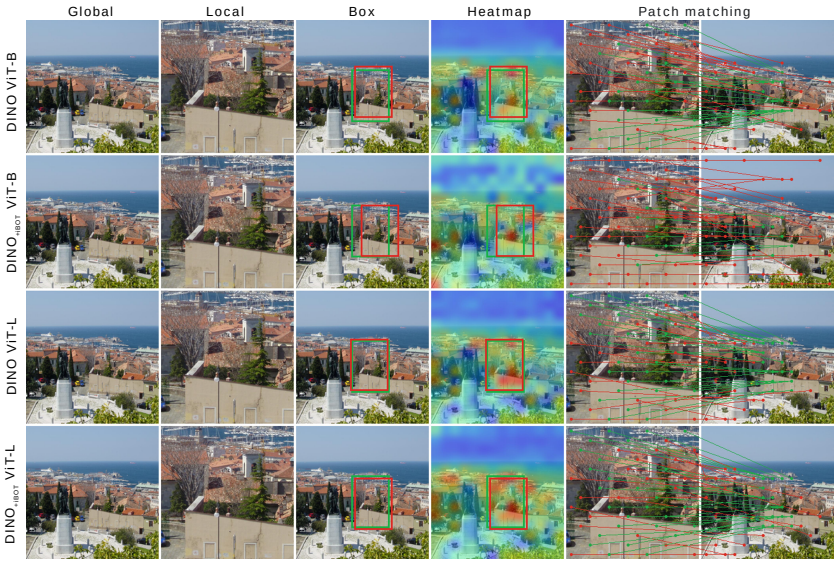

Figure 6: Local-to-global correspondence visualizations for DINO and DINO$_{+\text{iBOT}}$, showing RANSAC box estimates, similarity heatmaps, and patch matches. With ViT-B/16, adding iBOT weakens correspondence: reliable matches are largely confined to larger foreground objects, while smaller objects are incorrectly matched or poorly localized. This failure pattern is not observed in the ViT-L/16 examples, where the +iBOT model maintains correspondence for smaller objects. These illustrate the scale-dependent effect reported in Figure 5. Additional examples appear in Figure 17.

trajectories are not uniformly monotonic, but consistently reveal weaker late-training correspondence in the smaller +iBOT models. In ViT-L/H, the sustained deterioration is absent: correspondence generally improves or stabilizes, and the +iBOT models retain competitive or stronger performance. This scale dependence agrees with the DAVIS results in Section 3, where iBOT hurts tracking on ViT-S/B but modestly improves it on ViT-L/H.

Figure 15 examines different crop area ratios. For ViT-S/B, deficits in patch matching, region IoU, and geometric box IoU persist across the evaluated ratios, while the IoU$_{50}$ success gap largely closes for larger crops. For ViT-L/H, region IoU remains higher with iBOT, whereas the patch-matching advantage generally narrows as crop size increases and slightly reverses for ViT-L at ratio 0.5. The effect therefore depends on both model scale and the matching criterion, rather than following a universal relationship with crop size. Figure 6 provides qualitative examples of the correspondence failures that the diagnostic captures.

Together with Local Appearance Decodability, these results establish two distinct responses to iBOT. Smaller models retain broadly comparable appearance decodability while losing spatial matching ability. Larger models improve patch matching and region localization despite reduced deep-layer appearance decodability. Thus, preserving decodable pixel and texture evidence is insufficient to guarantee effective correspondence, and better correspondence does not require preserving all measured appearance attributes. The locality cost of iBOT consequently takes different forms across model scales.

5 How Gram Anchoring Changes Local Representations

Section 4 shows that iBOT affects local appearance and correspondence differently across model scales. We now examine whether Gram anchoring, introduced in DINOv3 [33], can alleviate these effects. By constraining pairwise patch similarities, Gram anchoring provides a useful intervention for testing whether preserving relational structure also preserves recoverable appearance.

We compare DINO$_{+\text{iBOT}}$, standard Gram anchoring, and DINO-Gram using ViT-B/16 and ViT-L/16. The Gram teacher is initialized from an epoch-250 checkpoint of DINO$_{+\text{iBOT}}$ for standard Gram anchoring and DINO for DINO-Gram. Regularization begins at epoch 260, targeting the late-stage deterioration observed in dense prediction. Both variants refresh the Gram teacher from the current EMA teacher at epochs 270, 280, and 290, so DINO-Gram differs in its initial anchor. Implementation details are provided in Appendix B.2.3.

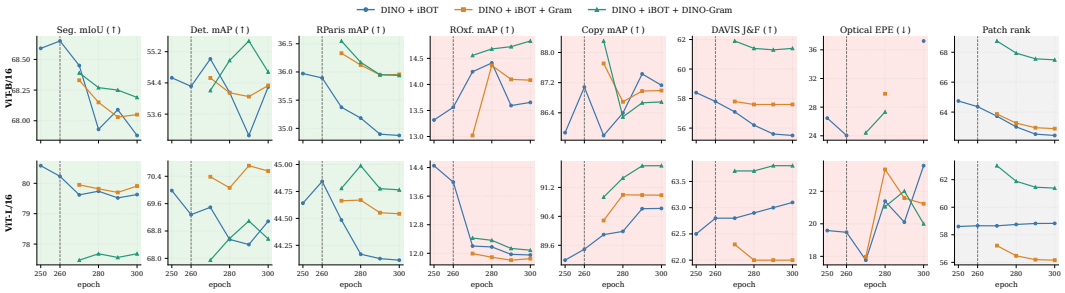

Figure 7: Downstream trajectories during epochs 250–300. Rows show ViT-B/16 and ViT-L/16; the dashed line marks Gram activation at epoch 260. Panels use independent linear vertical scales. Non-finite optical-flow results and EPE values above 100 are shown as gaps. Optical flow measures training-set fitting.

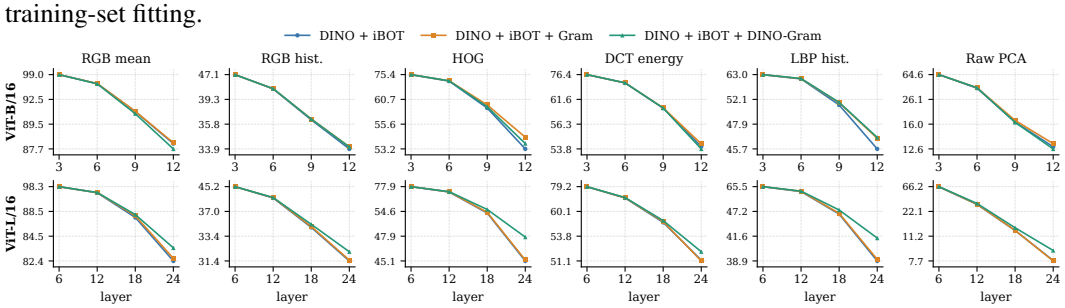

Figure 8: Local Appearance Decodability across layers at epoch 300. DINO-Gram improves deep-layer decodability more consistently for ViT-L/16 than for ViT-B/16. Values are $R^2 \times 100$; nonlinear asinh axes emphasize deep-layer differences.

5.1 Downstream Gains Do Not Imply Uniform Locality Recovery

Figure 7 shows that the downstream effects depend on both backbone scale and teacher choice. For ViT-B/16, the clearest improvement occurs on DAVIS: J&F increases from 55.5 to 57.6 with standard Gram anchoring and to 61.4 with DINO-Gram. Retrieval also improves, whereas segmentation and detection change relatively little. For ViT-L/16, standard Gram anchoring benefits segmentation and detection but reduces DAVIS performance. DINO-Gram instead improves DAVIS and copy detection while sacrificing some segmentation and detection performance.

This task dependence extends the observation in Section 3: dense semantic prediction and fine-grained correspondence do not measure interchangeable properties. In particular, stabilizing dense prediction does not establish that lost local evidence has been restored. The two annotation-free diagnostics reveal which aspects of locality change under each intervention.

5.2 Teacher Choice Affects Appearance and Correspondence Differently

For ViT-B/16, Gram anchoring primarily changes correspondence. Figure 8 shows small or mixed changes in appearance decodability, whereas Figure 9 shows clear improvements in deeper-layer correspondence. Final-layer patch matching accuracy increases from 24.8% to 30.1% with standard Gram anchoring and to 40.8% with DINO-Gram. The accompanying DAVIS gains indicate that this improvement extends beyond the diagnostic. This is consistent with the earlier finding that iBOT's main locality cost in smaller models concerns patch discriminability rather than recoverable pixel and texture evidence. Improving correspondence at this scale need not require a broad increase in appearance decodability.

For ViT-L/16, the relationship reverses. Standard Gram anchoring changes appearance decodability little, while DINO-Gram improves all six final-layer descriptors, particularly HOG and LBP. However, neither variant consistently improves correspondence: DINO-Gram reduces patch matching accuracy and region IoU, despite a small increase in RANSAC box IoU. Even with DINO-Gram, all six final-layer descriptor scores remain below those of DINO without iBOT, indicating partial appearance recovery. Thus, the intervention can improve the property that iBOT weakens in larger models without improving cross-view patch matching.

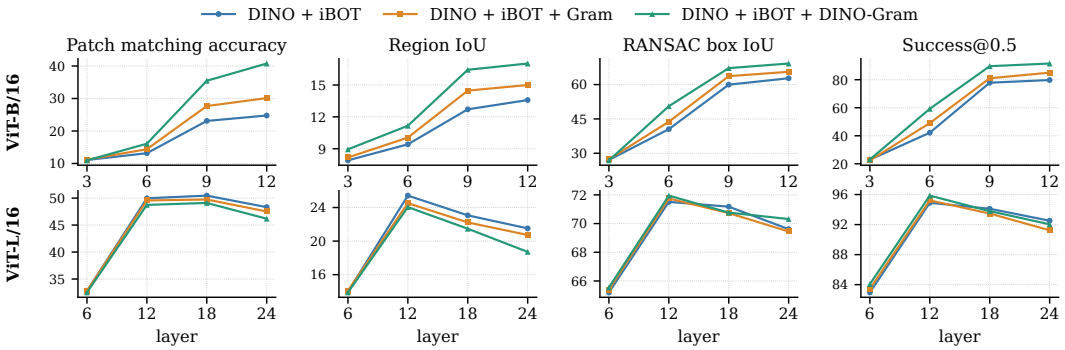

Figure 9: Local-to-Global Correspondence at epoch 300 with crop area ratio 0.1. Both Gram variants improve deep-layer correspondence for ViT-B/16, especially DINO-Gram. Neither consistently improves the four metrics for ViT-L/16. All values are percentages.

5.3 What Does Gram Anchoring Preserve?

A possible explanation is that Gram anchoring preserves the teacher's organization of patch similarities rather than local information in a general sense. An earlier $\text{DINO}_{+\text{iBOT}}$ teacher may restrain subsequent representational drift, but its similarities may already reflect the appearance or correspondence limitations induced by iBOT. An anchor can therefore stabilize a representation without supplying the evidence that it already lacks.

Changing the teacher to DINO may provide a different relational target. Our earlier diagnostics suggest why this change could have different effects across scales: DINO retains stronger correspondence in smaller models and stronger deep-layer appearance decodability in larger models. The observed benefits of DINO-Gram follow this distinction. We therefore hypothesize that the usefulness of an anchor depends on which property its teacher preserves, rather than on spectral diversity alone. Indeed, DINO-Gram increases patch effective rank at both scales, but improves patch matching only for ViT-B/16.

Finally, within-view similarity preservation does not directly enforce cross-view feature alignment. Gram matrices constrain relations among patches within an image, while correspondence additionally requires those relations to support consistent matching across views. Likewise, appearance-dependent similarities may retain texture or color differences without making them invariant to changes in viewpoint or context. This distinction offers a possible explanation for the ViT-L/16 result: improved appearance decodability can coexist with weaker patch matching. These mechanisms remain hypotheses, but the observed dissociation shows that Gram anchoring should be assessed separately for appearance preservation, correspondence, and downstream utility.

6 Conclusion

This work shows that the locality cost of masked patch self-distillation cannot be understood through a single measure of local feature quality. Spatial correspondence and recoverable appearance can respond differently to the same objective, and scaling can improve one while weakening the other. This distinction changes how progress toward general-purpose visual representations should be assessed: semantic transfer and visually coherent feature maps provide valuable evidence, but neither establishes that the representations retain the full range of local cues needed by downstream systems.

Our findings motivate training and evaluation practices that explicitly account for these complementary requirements. Future objectives could adapt their preservation targets to model scale and depth, while teacher selection could consider both appearance accessibility and correspondence rather than a single downstream score. The Gram anchoring results make this direction particularly relevant: preserving a teacher's relational structure inherits its preferences, which may benefit different local abilities unequally. For visual backbones serving multimodal and spatial reasoning systems, the broader challenge is to learn useful semantic invariances while keeping task-relevant visual distinctions accessible. The diagnostics developed here provide a way to measure that balance and guide its design.

7 Acknowledgments

This research is supported by the National Research Foundation Singapore under its AI Singapore Programme (Award Number: AISG3-RP-2022-030). Any opinions, findings, and conclusions expressed in this material are those of the author(s) and do not reflect the views of funding bodies.

References

[1] Nasir Ahmed, T. Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, C-23(1):90–93, 1974.

[2] Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15619–15629, 2023.

[3] Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. data2vec: A general framework for self-supervised learning in speech, vision and language. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

[4] Hangbo Bao, Li Dong, Songhao Piao, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.

[5] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *ICLR*, 2022.

[6] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012.

[7] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: Delving into high quality object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6154–6162, 2018.

[8] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. In *NeurIPS*, 2020.

[9] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

[11] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

[12] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2005.

[13] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR*, 2009.

[14] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[15] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM international conference on image and video retrieval*, pages 1–8, 2009.

[16] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, June 2010.

[17] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111(1):98–136, 2015.

[18] Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, 1981.

[19] Quentin Garrido, Randall Balestriero, Laurent Najman, and Yann Lecun. Rankme: Assessing the downstream performance of pretrained self-supervised representations by their rank. In *International conference on machine learning*, pages 10929–10974. PMLR, 2023.

[20] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent - a new approach to self-supervised learning. In *NeurIPS*, 2020.

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020.

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[23] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[24] Lei Huang, Yunhao Ni, Xi Weng, Rao Muhammad Anwer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Understanding whitening loss in self-supervised learning. *IEEE transactions on pattern analysis and machine intelligence*, 46(12):9479–9492, 2024.

[25] Ian T Jolliffe and Jorge Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions. Series A, Mathematical, physical, and engineering sciences*, 374(2065):20150202, 2016.

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and Larry Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014.

[27] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.

[28] 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.

[29] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbelaez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 DAVIS challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

[30] Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5706–5715, 2018.

[31] Olivier Roy and Martin Vetterli. The effective rank: A measure of effective dimensionality. In *2007 15th European signal processing conference*, pages 606–610. IEEE, 2007.

[32] Alexandre Sablayrolles, Matthijs Douze, Cordelia Schmid, and Hervé Jégou. Spreading vectors for similarity search. In *International Conference on Learning Representations*, 2019.

[33] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

[34] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018.

[35] Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai C Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *Advances in Neural Information Processing Systems*, 37:87310–87356, 2024.

[36] Chen Wei, Haoqi Fan, Saining Xie, Chao-Yuan Wu, Alan Yuille, and Christoph Feichtenhofer. Masked feature prediction for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

[37] Xi Weng, Lei Huang, Lei Zhao, Rao Muhammad Anwer, Salman Khan, and Fahad Khan. An investigation into whitening loss for self-supervised learning. In *NeurIPS*, 2022.

[38] Xi Weng, Yunhao Ni, Tengwei Song, Jie Luo, Rao Muhammad Anwer, Salman Khan, Fahad Shahbaz Khan, and Lei Huang. Modulate your spectrum in self-supervised learning. In *ICLR*, 2024.

[39] Xi Weng, Jianing An, Xudong Ma, Binhang Qi, Jie Luo, Xi Yang, Jin Song Dong, and Lei Huang. Clustering properties of self-supervised learning. In *Forty-second International Conference on Machine Learning*, 2025.

[40] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9653–9663, 2022.

[41] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *Advances in neural information processing systems*, 37:21875–21911, 2024.

[42] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

[43] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. iBOT: Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations*, 2022.

Appendix

Contents

A Class-Token Joint-Embedding Objectives **15**

B Downstream Benchmarks **15**

B.1 Capabilities Probed by the Downstream Benchmarks 15

B.2 Experimental Settings 16

C Additional Similarity Heatmap Visualization **20**

D Numerical Results and Trajectories of Downstream Evaluations **20**

E Additional DAVIS Video Object Tracking Visualization **23**

F Implementation and Visualization of Local Appearance Decodability **23**

F.1 Image Preparation and Probe Data 24

F.2 Local Appearance Targets 24

F.3 Fitting the Linear Decoder 25

F.4 Scoring and Interpretation 26

F.5 The Trajectories of Local Appearance Decodability 27

F.6 Visualization of Local Appearance Decodability 27

G Implementation and Visualization of Local-to-Global Correspondence **28**

G.1 Constructing Local Queries 29

G.2 Computing Patch Similarities 29

G.3 Matching and Localization Metrics 29

G.4 Crop Ratios and Sampling 31

G.5 Visualization and Training Trajectories 32

G.6 Additional Visualization of Local-Global Correspondence 34

H Extended Related Work **35**

H.1 From Image-Level to Patch-Level Self-Distillation 35

H.2 Prediction Targets in Masked Image Modeling 35

H.3 Gram Anchoring and the Preservation of Patch Relations 35

I Limitations **36**

J Licenses for Existing Assets **36**

A Class-Token Joint-Embedding Objectives

For a controlled comparison between DINO and DINO$_{+\mathrm{iBOT}}$, we use the same DINO loss configuration in both variants, following DINOv2 [28]. In particular, we construct the DINO teacher targets using Sinkhorn–Knopp normalization, as employed in SwAV [8] and DINOv2. Below, we detail this objective using the notation introduced in Section 2.1.

Student predictions. For a student crop $u \in \mathcal{C}_i$, the class-token head produces logits $\boldsymbol{s}_i^u \in \mathbb{R}^K$. The student prediction is obtained by applying temperature-scaled softmax:

$$p_{i,k}^u = \frac{\exp\left(s_{i,k}^u/\tau_s\right)}{\sum_{j=1}^K \exp\left(s_{i,j}^u/\tau_s\right)}, \qquad k = 1, \ldots, K, \tag{7}$$

where $\tau_s > 0$ is the student temperature and K is the output dimension of the class-token head.

Teacher targets. The teacher processes only global crops. We collect their logits $\boldsymbol{t}_i^v \in \mathbb{R}^K$, $v \in \mathcal{G}_i$, into a matrix $\boldsymbol{T} \in \mathbb{R}^{N \times K}$, where

$$N = \sum_{i=1}^B |\mathcal{G}_i| \tag{8}$$

is the total number of teacher crops in the mini-batch. Each row of $\boldsymbol{T}$ corresponds to one teacher crop. We construct the target matrix as

$$\boldsymbol{Q} = \mathrm{sg}\left[\mathrm{SK}\left(\exp\left(\frac{\boldsymbol{T}}{\tau_t}\right)\right)\right], \tag{9}$$

where the exponential is applied elementwise, $\tau_t > 0$ is the teacher temperature, and $\mathrm{sg}[\cdot]$ denotes the stop-gradient operation.

The Sinkhorn–Knopp operator SK alternately rescales rows and columns to produce per-crop probability distributions with balanced aggregate assignment mass across output dimensions. Under our scaling convention, the desired marginals are

$$\boldsymbol{Q}\mathbf{1}_K = \mathbf{1}_N, \qquad \boldsymbol{Q}^\top \mathbf{1}_N = \frac{N}{K}\mathbf{1}_K, \tag{10}$$

where $\mathbf{1}_d$ is the d-dimensional all-ones vector. In practice, a finite number of normalization iterations is used, ending with row normalization. Thus, each row sums to one, while the column-marginal constraint is satisfied approximately. In distributed training, the balancing statistics are aggregated across devices. The row corresponding to teacher crop v of image x_i defines its target distribution $\boldsymbol{q}_i^v$. The temperature τ_t controls the sharpness of these assignments.

Cross-view self-distillation loss. For each image, we match every global teacher target to student predictions from the other crops. The valid student–teacher pairs are

$$\mathcal{P}_i = \left\{(u, v) : u \in \mathcal{C}_i,\ v \in \mathcal{G}_i,\ u \neq v\right\}, \tag{11}$$

where $u \neq v$ excludes pairing the same crop across the two networks. The DINO objective is the average cross-entropy over images and valid view pairs:

$$\mathcal{L}_{\mathrm{base}} = \mathcal{L}_{\mathrm{DINO}} = -\frac{1}{B}\sum_{i=1}^B \frac{1}{|\mathcal{P}_i|} \sum_{(u,v)\in\mathcal{P}_i} \sum_{k=1}^K q_{i,k}^v \log p_{i,k}^u. \tag{12}$$

Gradients propagate only through the student predictions. The teacher targets, including the Sinkhorn–Knopp normalization, are computed without gradient propagation, and the teacher parameters are updated by the exponential moving average described in Section 2.1.

B Downstream Benchmarks

B.1 Capabilities Probed by the Downstream Benchmarks

The benchmarks place different demands on semantic discrimination, appearance information, and spatial correspondence. These categories describe their principal evaluation targets rather than mutually exclusive sources of information.

Semantic recognition and localization. ImageNet k-NN classification evaluates category discrimination using global image features, without explicitly testing localization or correspondence. Local cues may nevertheless contribute to recognition. VOC2012 segmentation [17] and VOC2007 detection [16] additionally require spatially accurate masks or boxes. Their targets emphasize category-correct localization: segmentation assigns the same label to visually different patches within an object, while detection does not evaluate its internal appearance. Consequently, improvements on these tasks do not establish better preservation of pixel or texture information.

RParis and ROxford instance retrieval. Both benchmarks require retrieving the same landmark across images, rather than merely recognizing its semantic category [30]. They therefore combine global landmark structure with distinctive local evidence. We report them separately because their responses to iBOT differ: RParis improves on ViT-S/L/H, whereas ROxford declines on ViT-S/B/L and remains nearly unchanged on ViT-H. One possible explanation is greater sensitivity to fine-grained instance discrimination on ROxford under our protocol. This remains an interpretation of the observed results, rather than evidence that RParis is purely semantic or ROxford exclusively local. Both use the Hard retrieval protocol.

Copydays copy detection. Copydays tests whether transformed images can be identified as copies of the same source [15]. Category similarity is insufficient; source-specific visual evidence must remain recognizable under transformations. However, our protocol compares aggregated image descriptors, so performance does not directly measure either pixel reconstruction or patch-to-patch matching.

DAVIS video object tracking. Our DAVIS evaluation propagates an initial object mask through similarities between frozen patch features across frames [9, 29]. It therefore emphasizes cross-frame matching and preservation of instance identity. Semantic coherence can assist propagation, but distinguishing same-category instances requires more specific information. The modest ViT-L/H gains motivate testing whether iBOT improves correspondence at larger scales; they do not independently establish this mechanism because tracking also depends on mask propagation and temporal aggregation.

Sintel optical flow. Sintel evaluates dense motion estimation, including accurate displacements and motion boundaries [6]. Appearance cues support matching, while context helps resolve ambiguity and occlusion. Pixel-level motion predictions do not require reconstructing pixel values, and our learned flow head introduces an additional readout dependency. Our flow head is evaluated on the same videos used to fit it, so this is a training-set readout diagnostic. The plotted comparisons retain EPE values at most 100 and use the latest common retained checkpoint: epochs 180, 260, 300, and 270 for ViT-S/B/L/H, respectively. They are therefore not uniformly final-epoch comparisons. The threshold is a reporting filter, not a definition of a valid EPE or a guarantee that all excluded evaluations failed numerically.

Patch effective rank. Effective rank measures the spread of the singular-value spectrum of the uncentered patch-token matrix [19, 31]. A decrease indicates greater concentration, but does not identify which information is lost or whether the remaining directions support better matching. We interpret it as a complementary spectral statistic rather than a direct measure of local capability.

B.2 Experimental Settings

This appendix documents the full pretraining and evaluation protocol used in Section 3. All pretraining runs use the DINOv2 codebase [28].

B.2.1 Pretraining

Backbones. We compare DINO and DINO$_{+\mathrm{iBOT}}$ using ViT-S/16, ViT-B/16, ViT-L/16, and ViT-H/16, implemented within the DINOv2 training framework [28]. The four backbones contain 12, 12, 24, and 32 transformer blocks, respectively. All models use a patch size of 16, SwiGLU feed-forward layers, four register tokens, and LayerScale initialized to 10^{-5}. We use uniform stochastic depth across blocks, with a drop-path rate of 0.3 for ViT-S/16 and ViT-B/16 and 0.4 for ViT-L/16 and ViT-H/16. The teacher shares the student's architecture and is updated by an exponential moving average, with momentum increasing from 0.994 to 1.0 according to a cosine schedule.

Training objectives. Both methods use the DINO class-token self-distillation objective [9]. $\text{DINO}_{+\text{iBOT}}$ additionally uses the masked patch self-distillation objective of iBOT [43], with its loss weight fixed at 1.0 throughout training. For each backbone, the paired runs share the same architecture, optimization settings, and data augmentation recipe; the intervention is the inclusion of the iBOT branch and its associated masking.

To reduce direct coupling between the class-token and patch-token heads, we follow the DINOv2 setup [28] and use separate projection heads for the two branches. Both objectives still update the shared backbone. Each head consists of a three-layer MLP with hidden dimension 2048 and bottleneck dimension 256, followed by an output layer with $131{,}072$ prototypes. The teacher temperature increases linearly from 0.04 to 0.07 over the first 30 epochs and remains at 0.07 thereafter. The final projection layers are frozen during the first epoch. The student temperature is 0.1 for both heads. For iBOT, we apply the standard settings: the masking probability is 0.5, with masking ratios sampled within $[0.1, 0.5]$. All models additionally apply the KoLeo regularizer [32] to the class-token features with a weight of 0.1.

Optimization and data augmentation. All models are pretrained from scratch on ImageNet-1K [13] for 300 epochs using AdamW with $\beta = (0.9, 0.999)$. The global batch size is 1024 for ViT-S/16, ViT-B/16, and ViT-L/16. Due to GPU memory constraints, both ViT-H/16 runs use a global batch size of 512. The peak learning rate follows the square-root scaling rule

$$\eta_{\text{peak}} = 4 \times 10^{-3} \sqrt{B/1024},$$

where B is the global batch size. This gives a peak learning rate of 4×10^{-3} for ViT-S/16, ViT-B/16, and ViT-L/16, and approximately 2.83×10^{-3} for ViT-H/16. We use a 10-epoch linear warmup followed by cosine decay to 10^{-6}. Weight decay increases from 0.04 to 0.4 according to a cosine schedule. The gradient-clipping threshold is 3.0, the patch-embedding learning-rate multiplier is 0.2, and the layerwise learning-rate decay factor is 0.9.

Each image produces two global crops of size 224×224 and eight local crops of size 96×96. The crop scale ranges are $[0.32, 1.0]$ for global crops and $[0.05, 0.32]$ for local crops. All runs use four NVIDIA H200 GPUs with `bf16` mixed precision and Fully Sharded Data Parallel (FSDP) training. The per-GPU batch size is 256 for ViT-S/16, ViT-B/16, and ViT-L/16, and 128 for ViT-H/16.

Compute resources. Table 1 summarizes the approximate peak GPU memory usage and wall-clock training time. Adding iBOT increases both memory consumption and training time at every backbone scale. A substantial source of this overhead is the patch-level prediction of distributions over $131{,}072$ prototypes, which requires large student logits and teacher targets for the masked patches, together with the associated training buffers. Consequently, the additional memory cost depends on the number of processed patches as well as the backbone size. The eight runs in Table 1 sum to approximately 3,964 H200 GPU-hours, excluding Gram experiments, downstream fitting, diagnostics, and exploratory runs. In particular, the ViT-H/16 measurements use half the batch size of the other backbones and should be interpreted with this difference in mind.

Table 1: Compute resources for 300-epoch pretraining on ImageNet-1K. All runs use four NVIDIA H200 GPUs with `bf16` mixed precision and FSDP. Batch size denotes the global batch size. Memory is the approximate peak usage per GPU, and time is the approximate wall-clock duration of a complete run.

Backbone	Objective	Batch size	Memory (GB)	Time
ViT-S/16	DINO	1024	28	1 d 7 h
	$\text{DINO}_{+\text{iBOT}}$	1024	100	2 d 5 h
ViT-B/16	DINO	1024	47	3 d 1 h
	$\text{DINO}_{+\text{iBOT}}$	1024	125	3 d 8 h
ViT-L/16	DINO	1024	100	5 d 15 h
	$\text{DINO}_{+\text{iBOT}}$	1024	129	6 d 5 h
ViT-H/16	DINO	512	86	9 d 8 h
	$\text{DINO}_{+\text{iBOT}}$	512	134	10 d 6 h

B.2.2 Downstream Evaluation

We evaluate the frozen teacher checkpoints on eight downstream benchmarks, counting RParis and ROxford separately. Below we describe the benchmark and the protocol used for each task.

ImageNet k-NN classification. We compute frozen class-token features on the ImageNet-1K training set and evaluate weighted k-NN classification on the validation set with cosine similarity. We use $k = 15$ and a softmax voting temperature of 0.07, and report top-1 accuracy following the DINO protocol [9]. The eval transform is the standard DINOv2 ImageNet evaluation transform: resize the shorter edge to 256, center crop 224×224, and normalize with ImageNet mean/std.

RParis and ROxford image retrieval. We evaluate on the revisited Paris6k and Oxford5k benchmarks [30]. Database and query images are resized to 224×224, and we extract a single image-level descriptor by concatenating the class token with a generalized-mean-pooled patch-token vector with exponent 4. We report the mean average precision (mAP) under the Hard evaluation protocol of Radenović et al. [30]. No fine-tuning or whitening is performed.

VOC2012 semantic segmentation. We use a linear segmentation protocol implemented in MMSegmentation. The frozen ViT backbone produces patch tokens at 512×512 input resolution; we concatenate features from 4 uniformly spaced block-chunk outputs after layer normalization and feed them to an FCN head with no intermediate convolutions. The concatenated feature dimensions are 1536, 3072, 4096, and 5120 for ViT-S/B/L/H, respectively. Training uses Pascal VOC2012 `train` for $10{,}000$ iterations with AdamW, learning rate 5×10^{-4}, a total batch size 16, weight decay 0, 200-iteration linear warmup, and a power-1 polynomial schedule. Random scaling in $[0.5, 2.0]$, 512×512 random crops, random horizontal flips, and photometric distortion are applied during training. Evaluation uses sliding-window inference with 512×512 windows and stride 341 on Pascal VOC2012 `val`, and we report mIoU.

VOC2007 object detection. We adopt a Cascade R-CNN[7] detector with a 4-conv-1-fc GIoU head and FPN neck on top of the frozen ViT, implemented in MMDetection. The backbone is fully frozen; FPN inputs are 4 uniformly spaced block-chunk outputs after layernorm. Training uses VOC2007 `train` with 1000×600 short-side resize, random horizontal flips, AdamW with learning rate 1×10^{-4}, a total batch size 16, weight decay 0.05, 500-iteration linear warmup from a 10^{-3} factor, and a multi-step decay at epochs 8 and 11 for 12 training epochs. Evaluation uses the VOC2007 `val` with the standard 11-point interpolation mAP [16].

Copydays copy detection. We follow the DINO copy-detection protocol on the Copydays dataset [15], with $10{,}000$ COCO [26] distractor images and $20{,}000$ COCO whitening images. Images are resized to 224×224, and we form a single descriptor by concatenating the class token with the GeM-pooled patch tokens (exponent 4). We center features, fit a PCA whitening with whitening exponent 0.5 on the COCO whitening set, and ℓ_2-normalize. Retrieval uses cosine similarity, and we report mAP on `strong` subset.

DAVIS video object tracking. We follow the DINO label-propagation tracking protocol on DAVIS-2017 [29]. For each video, the first-frame mask is propagated to subsequent frames via cosine similarity between frozen patch tokens of the target frame and the past 7 frames (plus the first frame). Frames are resized so that the shorter side is 480 and the longer side is rounded to a multiple of 64. We restrict matching to a spatial neighborhood of radius 12 patches around the query, aggregate from the top-5 neighbors, and use a softmax temperature of 0.1. We report the standard region–boundary score $\mathcal{J}\&\mathcal{F}$ on the DAVIS-2017 `val` split using the DAVIS official evaluation scripts [29].

Sintel optical flow. We use the PWCNet optical-flow head from MMFlow on top of the frozen ViT backbone with FPN. The encoder is fully frozen; FPN-projected feature maps with channels $\{32, 64, 96, 128\}$ at levels $\{2, 3, 4, 5\}$ feed the PWCNet [34] decoder. Training runs for 12 epochs on the Sintel `clean+final` training pass with 384×768 random crops, random affine transforms, color jitter, gamma jitter, random horizontal/vertical flips, AdamW with learning rate 1×10^{-4}, a total batch size 16, weight decay 0.05, 500-iteration linear warmup, and a multi-step decay at epochs 8 and 11. We report the average end-point error (EPE) on the `clean` pass of the same Sintel training videos, using the evaluation preprocessing pipeline. No disjoint validation split is used. These results measure the fitting behavior of a learned flow head on frozen features, not held-out optical-flow generalization.

Patch Effective Rank. To quantify the diversity of patch-level representations, we compute the normalized effective rank of the patch token matrix for each image. Given an image I_b, let its patch tokens be denoted as $\mathbf{X}_b = [\mathbf{x}_{b,1}, \mathbf{x}_{b,2}, \ldots, \mathbf{x}_{b,P}]^\top \in \mathbb{R}^{P\times D}$, where P is the number of patches and

D is the feature dimension of each patch token. We perform singular value decomposition on $\mathbf{X}_b$ and obtain its singular values $\{\sigma_{b,i}\}_{i=1}^{q}$, where $q = \min(P, D)$. The singular values are normalized as $p_{b,i} = \frac{\sigma_{b,i}}{\sum_{j=1}^{q} \sigma_{b,j}}$, which can be viewed as a probability distribution over spectral components. The entropy of this distribution is then computed by $H_b = -\sum_{i=1}^{q} p_{b,i} \log p_{b,i}$, with $0 \log 0 = 0$ and a small numerical floor in the implementation. Following the definition of effective rank, the patch effective rank is given by $\mathrm{ERank}(\mathbf{X}_b) = \exp(H_b)$. We report effective rank relative to the number of patches:

$$\mathrm{PER}(I_b) = \frac{\exp\left(-\sum_{i=1}^{q} p_{b,i} \log p_{b,i}\right)}{P}.$$

A larger value indicates a more evenly distributed singular-value spectrum; a smaller value indicates greater spectral concentration. Because $\mathbf{X}_b$ is not centered and the probabilities use singular values rather than their squares, this statistic is not a distribution of covariance variance. It does not by itself establish information loss, representational collapse, or poorer correspondence. The theoretical upper bound of the normalized score is q/P, which equals one when $D \geq P$ as in our evaluated backbones.

In our experiments, the patch effective rank is computed on $10{,}000$ randomly sampled ImageNet training images using subset seed 0. We report the mean normalized patch effective rank across images, multiplied by 100 in plots and tables.

B.2.3 Experimental Settings for Gram Anchoring

Models and comparison. We evaluate Gram anchoring using ViT-B/16 and ViT-L/16 under the 300-epoch ImageNet-1K pretraining setup described in Appendix B.2.1. The reference configuration is $\mathrm{DINO}_{+\mathrm{iBOT}}$. The two additional configurations introduce a Gram loss whose teacher is initialized from either the $\mathrm{DINO}_{+\mathrm{iBOT}}$ or DINO epoch-250 checkpoint of the same backbone architecture. We refer to these configurations as standard Gram anchoring and DINO-Gram, respectively. The DINO and iBOT objective weights remain 1.0.

Gram objective. Let $Z_s, Z_g \in \mathbb{R}^{N \times d}$ denote student and Gram-teacher patch features for a global crop, and let $\widehat{Z}_s, \widehat{Z}_g$ denote their row-wise ℓ_2-normalized versions. The student features come from the masked global-crop forward pass, whereas the Gram teacher processes the corresponding unmasked crop. All patch positions, including masked student positions, enter the Gram matrices. The objective matches within-crop pairwise similarities:

$$\mathcal{L}_{\mathrm{Gram}} = \frac{1}{N^2} \left\| \widehat{Z}_s \widehat{Z}_s^\top - \widehat{Z}_g \widehat{Z}_g^\top \right\|_F^2, \tag{13}$$

averaged over global crops in the batch. The loss uses backbone patch features rather than prototype-head outputs. We do not subtract the spatial feature mean or clamp negative similarities. Gradients propagate through the student features only.

Activation and teacher schedule. Gram regularization is inactive for the first 260 epochs. Its coefficient then increases from 0 to 2 over one epoch using a cosine transition and remains 2 thereafter. The initial Gram-teacher checkpoint is taken at epoch 250, before the late-stage deterioration motivating the intervention. For both standard Gram anchoring and DINO-Gram, we perform three subsequent teacher updates at epochs 270, 280, and 290. Each update copies the current EMA teacher into the Gram teacher. The Gram teacher is therefore fixed between updates, rather than fixed throughout the entire intervention. Thus, DINO-Gram uses DINO for initialization; its later anchors are the current run's EMA teacher, not a permanently fixed DINO checkpoint.

Optimization. Both backbones use a global batch size of 1024, distributed across four GPUs. The drop-path rates are 0.3 for ViT-B/16 and 0.4 for ViT-L/16. The AdamW optimizer, learning-rate and weight-decay schedules, crop sizes, and other pretraining settings follow Appendix B.2.1. Gram regularization is added to the existing objective:

$$\mathcal{L} = \mathcal{L}_{\mathrm{DINO}} + \mathcal{L}_{\mathrm{iBOT}} + 0.1\,\mathcal{L}_{\mathrm{KoLeo}} + \lambda_{\mathrm{Gram}}(t)\mathcal{L}_{\mathrm{Gram}}. \tag{14}$$

Evaluation and plotting. Downstream trajectories display the recorded checkpoints between epochs 250 and 300, with independent linear vertical scales for each panel. Unavailable evaluations

are not imputed. For optical flow, non-finite EPE values and values exceeding 100 are omitted from the displayed curves, with gaps retained between valid observations. This filtering does not establish successful optical-flow evaluation for checkpoints whose results are omitted.

Both annotation-free layer sweeps use epoch 300, without averaging across training checkpoints. We evaluate blocks 3, 6, 9, and 12 for ViT-B/16 and blocks 6, 12, 18, and 24 for ViT-L/16. Appearance decodability reports the dimension-averaged R^2 separately for each descriptor group. Correspondence uses crop area ratio 0.1 and reports patch matching accuracy, region IoU, RANSAC box IoU, and success at IoU 0.5. The plots compare recorded point estimates and do not imply statistical significance.

C Additional Similarity Heatmap Visualization



Figure 10: Additional anchor-patch similarity heatmaps for DINO and DINO$_{+\mathrm{iBOT}}$ using ViT-B/16 and ViT-L/16.

D Numerical Results and Trajectories of Downstream Evaluations

Table 2 reports the numerical values underlying Figure 2 for the four DINO/DINO$_{+\mathrm{iBOT}}$ pairs.

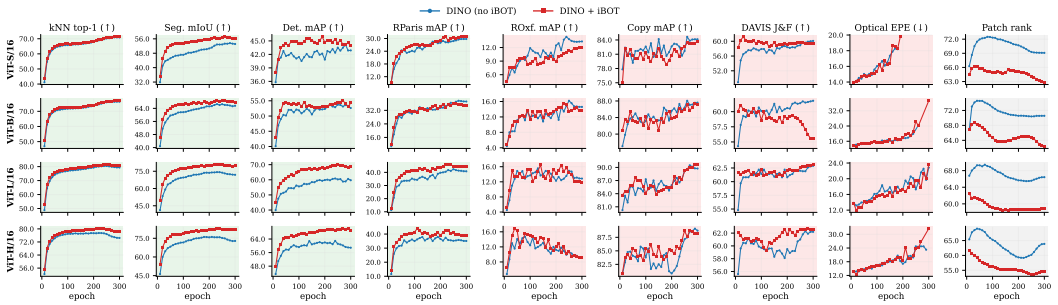

Figure 11: Training trajectories of DINO and DINO$_{+\mathrm{iBOT}}$ across four backbone scales. The effect of iBOT varies with the task, backbone scale, and training stage. Improvements in classification and dense semantic prediction coexist with mixed effects on retrieval and correspondence. Non-finite optical-flow results and EPE values exceeding 100 are excluded.

Arch.	Metric	Better	DINO	DINO$_{+\text{iBOT}}$	Δ
ViT-S/16	kNN top-1	↑	70.88	71.27	+0.39
	VOC2012 Seg. mIoU	↑	53.13	56.02	+2.89
	VOC2007 Det. mAP	↑	42.88	43.96	+1.08
	RParis Hard mAP	↑	29.75	30.78	+1.03
	ROxford Hard mAP	↑	13.34	12.03	-1.31
	Copy strong mAP	↑	84.14	83.69	-0.45
	DAVIS J&F	↑	60.10	59.30	-0.80
	Sintel clean EPE	↓	18.19	18.26	+0.06
	Normalized patch rank	–	69.10	62.85	−6.25
ViT-B/16	kNN top-1	↑	76.36	76.97	+0.60
	VOC2012 Seg. mIoU	↑	65.61	67.88	+2.27
	VOC2007 Det. mAP	↑	52.64	54.29	+1.66
	RParis Hard mAP	↑	37.24	34.87	-2.36
	ROxford Hard mAP	↑	14.60	13.65	-0.95
	Copy strong mAP	↑	87.61	87.13	-0.48
	DAVIS J&F	↑	61.80	55.50	-6.30
	Sintel clean EPE	↓	24.44	24.04	-0.39
	Normalized patch rank	–	70.57	62.44	−8.13
ViT-L/16	kNN top-1	↑	79.36	80.44	+1.08
	VOC2012 Seg. mIoU	↑	71.93	79.63	+7.70
	VOC2007 Det. mAP	↑	59.71	69.08	+9.38
	RParis Hard mAP	↑	40.81	44.11	+3.30
	ROxford Hard mAP	↑	12.86	11.95	-0.91
	Copy strong mAP	↑	89.74	90.62	+0.87
	DAVIS J&F	↑	62.90	63.10	+0.20
	Sintel clean EPE	↓	22.84	23.57	+0.73
	Normalized patch rank	–	66.42	58.83	−7.59
ViT-H/16	kNN top-1	↑	74.45	78.11	+3.65
	VOC2012 Seg. mIoU	↑	72.88	81.95	+9.07
	VOC2007 Det. mAP	↑	58.89	68.94	+10.05
	RParis Hard mAP	↑	35.10	38.96	+3.87
	ROxford Hard mAP	↑	9.05	9.07	+0.02
	Copy strong mAP	↑	88.74	88.26	-0.47
	DAVIS J&F	↑	62.20	62.60	+0.40
	Sintel clean EPE	↓	24.73	26.66	+1.92
	Normalized patch rank	–	63.88	54.60	−9.29

Table 2: Numerical results for the four backbone scales in Figure 2. All rows use epoch 300 except Sintel EPE, which uses the latest common retained checkpoint: epochs 180/260/300/270 for ViT-S/B/L/H. Sintel scores are evaluated on training videos rather than a held-out split. Performance metrics other than EPE are percentages. Normalized patch rank is the mean patch effective rank divided by the number of patches and multiplied by 100; it describes spectral dimensionality rather than task accuracy. Δ denotes DINO$_{+\text{iBOT}}$ minus DINO, computed before rounding. Green and red indicate improved and degraded task performance, respectively; patch-rank changes are shown without a performance interpretation.

At epoch 300, adding iBOT improves ImageNet kNN classification, VOC2012 segmentation, and VOC2007 detection at all four scales. RParis retrieval improves for ViT-S/16, ViT-L/16, and ViT-H/16, but decreases for ViT-B/16. The results on tasks requiring local discrimination are more mixed. ROxford retrieval decreases for ViT-S/16, ViT-B/16, and ViT-L/16, while remaining nearly unchanged for ViT-H/16. Copy detection decreases at three scales, with ViT-L/16 as an exception. DAVIS exhibits a clear change with model scale: iBOT reduces J&F for ViT-S/16 and ViT-B/16, but yields modest improvements for ViT-L/16 and ViT-H/16. At the selected optical-flow checkpoints, iBOT increases EPE for ViT-S/16, ViT-L/16, and ViT-H/16, but slightly reduces it for ViT-B/16.

Figure 11 further shows that these differences vary over training rather than following a uniformly widening semantic–locality trade-off. For example, the DAVIS advantage of iBOT on ViT-S/16 at earlier checkpoints reverses by epoch 300, whereas the ViT-B/16 gap becomes more negative late in training. Retrieval and copy-detection differences can also change sign. Meanwhile, normalized patch rank is lower with iBOT at epoch 300 for every backbone, including the larger models whose DAVIS performance improves. Thus, reduced spectral dimensionality alone does not imply weaker correspondence. These results motivate the separate diagnostics in Section 4: Local Appearance Decodability measures recoverable pixel and texture evidence, whereas Local-to-Global Correspondence measures whether patches remain spatially distinguishable and matchable across views.

E Additional DAVIS Video Object Tracking Visualization

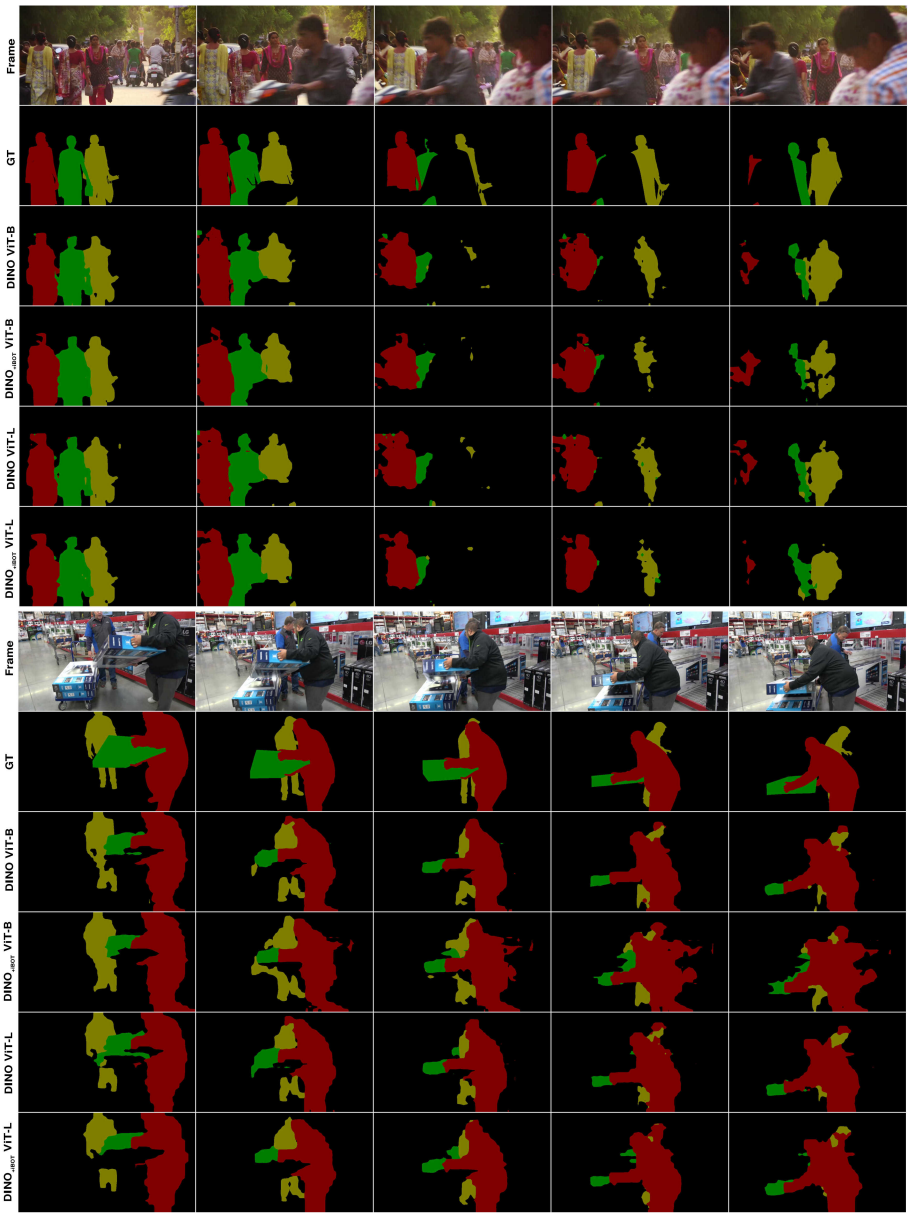

Figure 12: Additional DAVIS video object tracking visualization for DINO and DINO$_{+\text{iBOT}}$ using ViT-B/16 and ViT-L/16.

F Implementation and Visualization of Local Appearance Decodability

Local Appearance Decodability asks a simple question: *how accurately can a linear decoder recover the appearance of an image patch from its frozen patch token?* We describe each raw patch using six groups of pixel-derived attributes, covering color, edge orientation, texture, spatial frequency, and raw-pixel variation. We then train a linear decoder to predict these attributes and evaluate it on held-out images. The backbone remains frozen throughout this procedure, and no semantic annotations or model-generated targets are used.

F.1 Image Preparation and Probe Data

Aligning patch tokens with raw pixels. Each image undergoes a deterministic evaluation transform consisting of bicubic resizing of the shorter edge to 256, a 224×224 center crop, and conversion to an RGB tensor $w_{\text{raw}} \in [0,1]^{3\times H\times W}$. We use this tensor to construct the appearance targets. A copy of the same tensor is normalized using the ImageNet mean and standard deviation and passed to the backbone. Thus, the backbone input and the target image have exactly the same spatial crop.

For a backbone with patch size p, we divide the raw image into non-overlapping $p \times p$ patches. The raw patch at location r is denoted by $\tilde{w}_r \in [0,1]^{3\times p\times p}$, and its corresponding token from layer ℓ is $z_\theta^{(\ell)}(w,r) \in \mathbb{R}^d$. We check that the number of raw patches matches the number of patch tokens.

Separating training and validation images. We randomly select 20,000 ImageNet-1K training images using subset seed 0 and divide them into 10,000 probe-training images and 10,000 validation images. We use all $14 \times 14 = 196$ patches per image, yielding 1,960,000 token–target pairs per split. Splitting at the image level ensures that patches from the same image cannot appear in both sets. All fitted preprocessing statistics, including the raw-pixel PCA basis and the decoder's normalization statistics, are estimated using only the probe-training data.

The MAE and SimMIM curves use separate pretrained ViT-B/16 reference checkpoints with the same probe protocol. They are not controlled 300-epoch DINO ablations: their pretraining objectives and training budgets differ. Their role is to contextualize appearance decodability, not to isolate the causal effect of a pretraining objective.

F.2 Local Appearance Targets

At each patch location, we concatenate six descriptor groups:

$$y(w,r) = \phi(w_{\text{raw}}, r) = \big[\phi_{\text{rgb_mean}}, \phi_{\text{rgb_hist}}, \phi_{\text{hog}}, \phi_{\text{dct}}, \phi_{\text{lbp}}, \phi_{\text{raw_pca}}\big](w_{\text{raw}}, r). \tag{15}$$

Their dimensions are 3, 24, 9, 6, 16, and 64, respectively, giving a 122-dimensional target. Although the groups are concatenated for fitting, we evaluate and report each group separately. RGB statistics, DCT, and raw PCA use only the aligned raw patch. HOG and LBP are computed on the full image and then pooled within each patch, so their boundary responses can depend on a one-pixel neighborhood outside the patch.

RGB mean: average color and brightness. We average the pixel values within each patch independently for the red, green, and blue channels:

$$\phi^c_{\text{rgb_mean}}(\tilde{w}_r) = \frac{1}{p^2}\sum_{a,b} \tilde{w}^c_r(a,b), \qquad c \in \{R,G,B\}. \tag{16}$$

These three values describe the patch's average color, but do not retain the arrangement of pixels within it.

RGB histogram: distribution of color intensities. For each color channel, we divide $[0,1]$ into eight equal-width bins and count the fraction of pixels falling into each bin. A pixel value v is assigned to bin $\min(\lfloor 8v \rfloor, 7)$. Concatenating the three normalized histograms produces 24 values. Unlike the RGB mean, this descriptor distinguishes patches with similar average colors but different intensity distributions.

Grayscale descriptors. For HOG, DCT, and LBP, we convert the raw image to grayscale using

$$I = 0.2989\, w^R_{\text{raw}} + 0.5870\, w^G_{\text{raw}} + 0.1140\, w^B_{\text{raw}}. \tag{17}$$

HOG: distribution of edge orientations. Inspired by Histograms of Oriented Gradients (HOG) [12], we summarize the edge orientations within each patch. We compute horizontal and vertical gradients using 3×3 Sobel filters scaled by $1/8$ on the full grayscale image, with zero padding at the image boundary. Each pixel contributes to a nine-bin histogram of unsigned orientations in $[0,\pi)$, weighted by its gradient magnitude.

Specifically, the magnitude is $m = \sqrt{g_x^2 + g_y^2 + 10^{-12}}$, and the orientation is $\theta = \operatorname{atan2}(g_y, g_x) \bmod \pi$. The bin centers are $\nu_k = k\pi/9$. Votes are distributed between neighboring bins using triangular weights $\max(0, 1 - d_\pi(\theta, \nu_k)/(\pi/9))$, where d_π is circular distance modulo π. We pool the weighted votes over the patch and normalize by the pooled gradient magnitude, with a small numerical stabilizer. This yields nine values describing the relative prevalence of different edge directions. Our descriptor uses a single histogram per patch rather than the full spatial cell and block construction of conventional HOG.

DCT energy: variation across spatial frequencies. We apply an orthonormal two-dimensional discrete cosine transform (DCT) [1] to the grayscale patch. Low-frequency coefficients describe slowly varying intensity patterns, whereas higher-frequency coefficients describe more rapid spatial changes.

We exclude the DC coefficient, which represents the patch mean, and group the remaining coefficients into six radial frequency bands. For band $\mathcal{F}_k$, we compute

$$\phi_{\mathrm{dct}}^k(\tilde{w}_r) = \log\left(1 + \frac{1}{|\mathcal{F}_k|} \sum_{(u,v)\in\mathcal{F}_k} C_{uv}^2\right), \tag{18}$$

where C_{uv} is a DCT coefficient. Averaging accounts for different numbers of coefficients per band, and the logarithm compresses the range of energy values. The bands partition the radial coordinate $\sqrt{u^2 + v^2}$ into six equal-width intervals over $[0, \sqrt{2}(p-1)]$, with the maximum radius assigned to the last band and the DC coefficient excluded. This descriptor measures the distribution of spatial-frequency energy rather than reconstructing the original patch.

LBP histogram: local intensity patterns. Local binary patterns (LBP) describe texture through comparisons between a pixel and its neighbors [27]. For each grayscale pixel, we compare its intensity with those of its eight neighbors:

$$\mathrm{LBP}(a, b) = \sum_{t=0}^{7} 2^t \mathbf{1}[I_t(a, b) \geq I(a, b)]\,. \tag{19}$$

where $I_t(a, b)$ denotes the intensity of neighbor t. Neighbors are ordered clockwise from the top-left position (top-left, top, top-right, right, bottom-right, bottom, bottom-left, left). LBP codes are computed on the full grayscale image, with replicate padding at the image boundary, before patch-wise pooling.

The resulting code lies between 0 and 255. We coarsen these codes into 16 bins using $q_{\mathrm{lbp}} = \lfloor \mathrm{LBP}/16 \rfloor$ and normalize the histogram by the number of pixels in the patch. This is a compact histogram of local intensity patterns; it does not use uniform-pattern or rotation-invariant LBP mapping.

Raw-pixel PCA: dominant variation in RGB patches. The previous descriptors summarize predefined appearance attributes. To include a more general description of pixel variation, we also use principal component analysis (PCA) [25] on flattened RGB patches.

Let $v_r = \mathrm{vec}(\tilde{w}_r) \in \mathbb{R}^{3p^2}$. Using sampled probe-training patches, we estimate their mean μ_{pca} and the top 64 eigenvectors of their covariance matrix, collected in $U_{\mathrm{pca}} \in \mathbb{R}^{3p^2 \times 64}$. The target is

$$\phi_{\mathrm{raw_pca}}(\tilde{w}_r) = U_{\mathrm{pca}}^\top (v_r - \mu_{\mathrm{pca}}). \tag{20}$$

The mean and basis are then fixed and applied to both splits. Unlike the pooled color and texture histograms, this descriptor retains information about pixel positions through the flattened patch representation. It captures the leading modes of RGB variation rather than all raw-pixel information.

F.3 Fitting the Linear Decoder

For each backbone and evaluated layer, we fit a separate ridge-regression decoder [23]. Let $x_j \in \mathbb{R}^d$ be a frozen patch token and $y_j \in \mathbb{R}^{122}$ its appearance target. We standardize each feature dimension

using its probe-training mean μ_x and standard deviation σ_x, with a small variance floor for numerical stability:

$$\bar{x}_j = (x_j - \mu_x) \oslash \sigma_x, \qquad \bar{y}_j = y_j - \mu_y, \tag{21}$$

where $\oslash$ denotes element-wise division and μ_y is the probe-training target mean. Targets are centered but are not variance-standardized.

Given N sampled training pairs, the decoder is fitted by

$$W^\star = \arg\min_W \frac{1}{N} \sum_{j=1}^{N} \left\| W^\top \bar{x}_j - \bar{y}_j \right\|_2^2 + \lambda \|W\|_F^2. \tag{22}$$

The regularization term limits the magnitude of the decoder weights. The implementation accumulates feature covariance and feature–target cross-covariance statistics, then solves

$$\left(\frac{\bar{X}^\top \bar{X}}{N} + \lambda I_d \right) W^\star = \frac{\bar{X}^\top \bar{Y}}{N}, \tag{23}$$

where the rows of $\bar{X}$ and $\bar{Y}$ are the standardized features and centered targets. This avoids storing all sampled patch pairs simultaneously. The parameter λ corresponds to `ridge_alpha` in the implementation and is fixed at 10^{-3} for all reported probes.

For a validation token x_j, the prediction is

$$\hat{y}_j = (W^\star)^\top \big((x_j - \mu_x) \oslash \sigma_x \big) + \mu_y. \tag{24}$$

The backbone, preprocessing statistics, PCA basis, and decoder are all fixed during validation.

F.4 Scoring and Interpretation

Dimension-averaged coefficient of determination. We evaluate each target dimension using the coefficient of determination, R^2. Let $\mathcal{V}$ denote the validation patch pairs and let $\mu_{g,k}^{\mathrm{val}}$ be the validation mean of dimension k in descriptor group g. We compute

$$R_{g,k}^2 = 1 - \frac{\sum_{j \in \mathcal{V}} (\hat{y}_{j,g,k} - y_{j,g,k})^2}{\max\Big(\sum_{j \in \mathcal{V}} (y_{j,g,k} - \mu_{g,k}^{\mathrm{val}})^2, \epsilon \Big)}. \tag{25}$$

The validation mean is used only to compute the evaluation denominator; it is not used to fit the decoder or make predictions. We then average over the m_g dimensions within each group:

$$R_{g,\mathrm{mean}}^2 = \frac{1}{m_g} \sum_{k=1}^{m_g} R_{g,k}^2. \tag{26}$$

Computing R^2 before averaging gives each descriptor dimension equal weight, rather than allowing dimensions with larger target variance to dominate the group score.

Reading the scores. When the target's validation sum of squares exceeds ϵ, $R^2 = 1$ indicates perfect prediction, $R^2 = 0$ matches the squared error of predicting its validation mean, and $R^2 < 0$ indicates a larger error than that baseline. A higher group score therefore means that the corresponding appearance attributes are more linearly accessible from the patch tokens under this probe protocol. The numerical floor ϵ stabilizes dimensions with negligible validation variance.

What this diagnostic establishes. Local Appearance Decodability measures the linear accessibility of selected pixel-derived attributes. A lower score does not establish that the information has been completely removed: it may remain accessible to a nonlinear decoder or be encoded in a form not captured by these targets. Likewise, a higher score does not imply better spatial matching. A token can encode a patch's color or texture without preserving the feature relationships needed to match that patch across views. We therefore interpret this diagnostic together with Local-to-Global Correspondence (Section 4.2), which evaluates a complementary property of local representations.

F.5 The Trajectories of Local Appearance Decodability

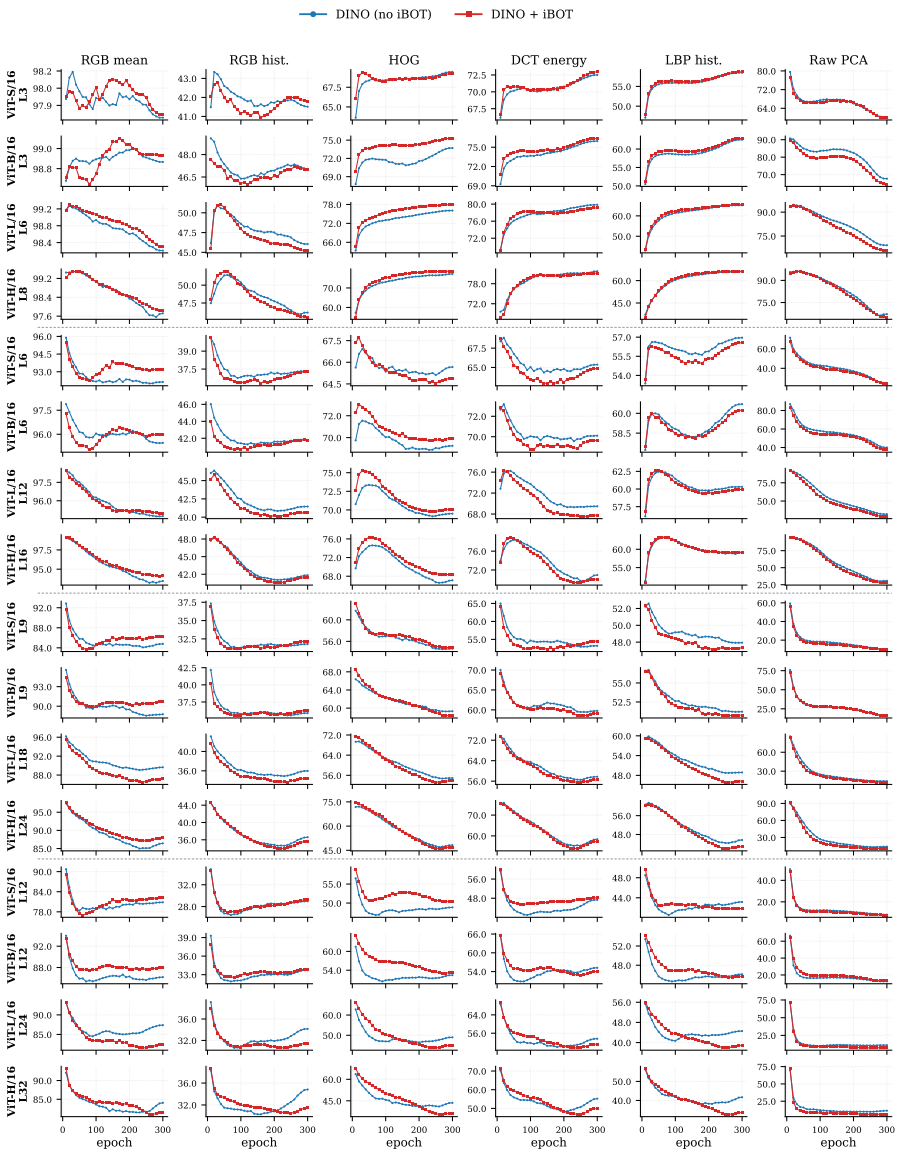

Figure 13: Training trajectories of local appearance decodability for each descriptor group.

The decodability trajectories reveal a scale-dependent effect of iBOT on local appearance information. For ViT-S/16 and ViT-B/16, adding iBOT generally preserves or slightly improves descriptor decodability, with no consistent loss of linearly recoverable appearance evidence. For ViT-L/16 and ViT-H/16, reduced final-layer decodability emerges during training and persists at the final checkpoint across descriptor groups, including RGB mean, DCT energy, HOG, and raw PCA. Together with the layer-wise results, these trajectories suggest that the appearance cost of iBOT is concentrated in the deeper layers of larger backbones, rather than being a uniform consequence of patch self-distillation at every model scale.

F.6 Visualization of Local Appearance Decodability

RGB-mean predictions are displayed directly as patch colors, clipped to $[0, 1]$. For HOG, DCT energy, and raw PCA, we visualize the ℓ_2 norm of each patch's descriptor vector as a heatmap. Each heatmap

is independently min–max normalized; colors therefore show relative spatial patterns within a panel, not calibrated prediction magnitudes across models. These maps do not reconstruct RGB pixels or display the individual descriptor components. Quantitative comparisons use the full descriptor vectors and R^2.

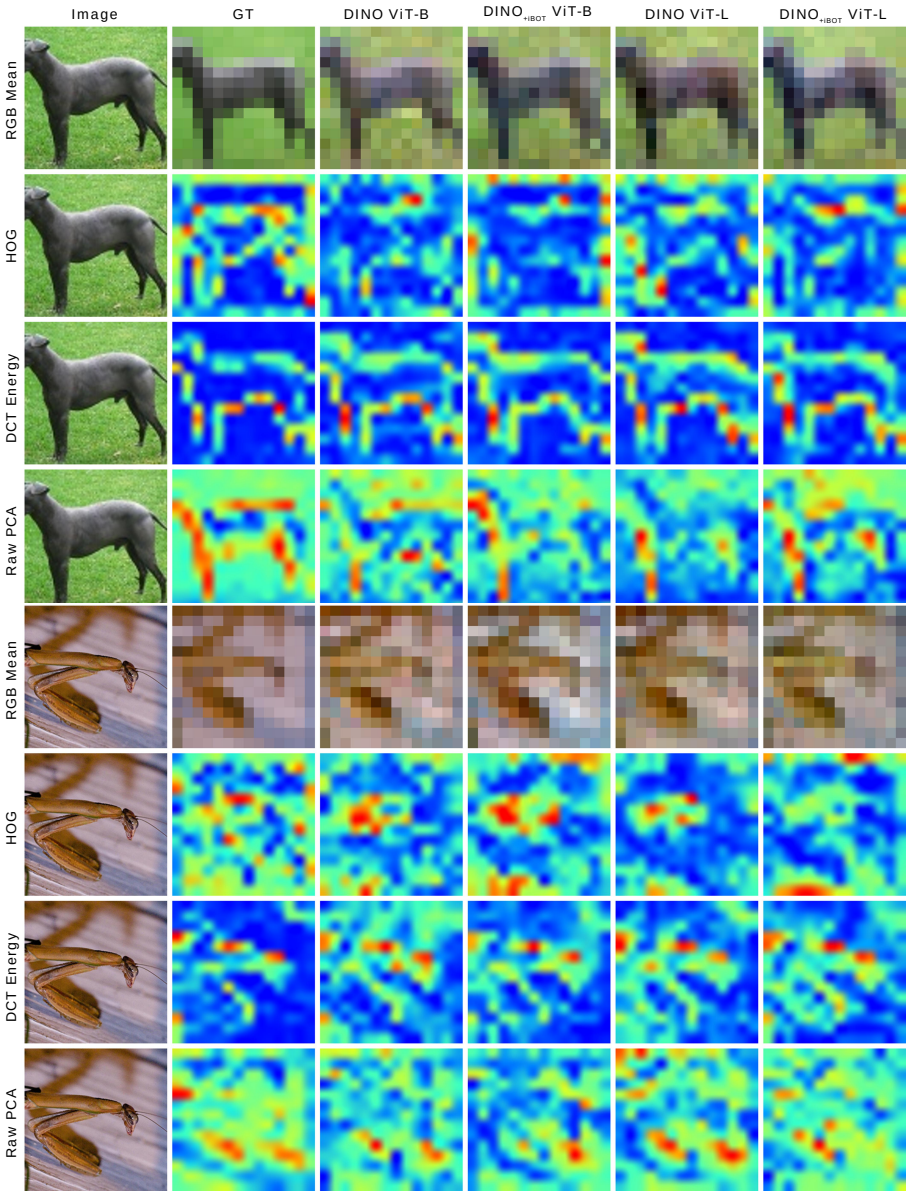

Figure 14: Visualization of local appearance decodability. Here we compare the RGB mean, HOG, DCT energy, and raw PCA results produced by ViT-B/16 and ViT-L/16 models of DINO and $\text{DINO}_{+\text{iBOT}}$.

G Implementation and Visualization of Local-to-Global Correspondence

Local-to-Global Correspondence asks whether a representation can locate a cropped image region within its source image. We extract frozen patch tokens from the crop and the full image, match them using cosine similarity, and evaluate whether the matches identify the correct source region and support a consistent geometric localization. The reference region is known from the crop operation, so the diagnostic requires no human annotations or learned matching head.

G.1 Constructing Local Queries

Source images and paired evaluation. We use the 157 source images from the Copydays dataset [15]. For each image, we sample four local crops, giving 628 queries per run. The default crop area ratio is $\rho = 0.1$. The crop specifications are cached and reused across models in each paired comparison, ensuring that DINO and DINO$_{+\mathrm{iBOT}}$ are evaluated on identical queries.

Crop size and position. For a source image of width W_{img} and height H_{img}, the target crop area is $A_{\mathrm{crop}} = \rho W_{\mathrm{img}} H_{\mathrm{img}}$. To introduce variation in crop shape, we sample the aspect ratio α log-uniformly from $[0.75, 1.25]$. The crop dimensions are

$$W_{\mathrm{crop}} = \mathrm{round}\sqrt{A_{\mathrm{crop}}\alpha}, \qquad H_{\mathrm{crop}} = \mathrm{round}\sqrt{A_{\mathrm{crop}}/\alpha}. \tag{27}$$

We clip these dimensions to the valid image range and sample the top-left corner (x_0, y_0) uniformly from valid integer positions. The resulting crop occupies approximately a fraction ρ of the source image, subject to rounding and clipping.

We express its reference box in normalized source-image coordinates:

$$B^\star = \left(\frac{x_0}{W_{\mathrm{img}}}, \frac{y_0}{H_{\mathrm{img}}}, \frac{x_0 + W_{\mathrm{crop}}}{W_{\mathrm{img}}}, \frac{y_0 + H_{\mathrm{crop}}}{H_{\mathrm{img}}} \right) = (x_0^\star, y_0^\star, x_1^\star, y_1^\star). \tag{28}$$

Both the full image and the crop are resized to 224×224 before feature extraction. Consequently, the same visual content appears at different scales and with different surrounding context in the two inputs.

G.2 Computing Patch Similarities

For a fixed backbone and layer ℓ, let $\mathcal{I}$ and $\mathcal{J}$ index the local-query and source-image patch tokens, respectively. Both inputs produce an $H_{\mathrm{grid}} \times W_{\mathrm{grid}}$ patch grid. Let $u(i) \in [0,1]^2$ denote the normalized center of query patch i, and let $r(j) \in [0,1]^2$ denote the normalized center of source patch j.

Reference source region. The reference region consists of source patches whose centers fall inside the sampled crop:

$$G(B^\star) = \{ j \in \mathcal{J} : x_0^\star \le r_x(j) < x_1^\star,\ y_0^\star \le r_y(j) < y_1^\star \}. \tag{29}$$

If the box contains no source-patch center, we use the source patch nearest to the box center as a fallback.

Similarity matrix. We extract patch tokens from the frozen backbone and normalize each token to unit ℓ_2 norm. The local-to-global similarity matrix is

$$A_\ell(i,j) = \tilde{z}_\ell^{\mathrm{local}}(i)^\top \tilde{z}_\ell^{\mathrm{global}}(j), \qquad i \in \mathcal{I},\ j \in \mathcal{J}, \tag{30}$$

where $\tilde{z}$ denotes a normalized token. For each query patch, its top-1 source match is

$$m(i) = \arg\max_{j \in \mathcal{J}} A_\ell(i,j). \tag{31}$$

All matching and localization metrics below are computed from this similarity matrix.

G.3 Matching and Localization Metrics

We evaluate complementary aspects of correspondence: whether individual matches fall inside the correct region, whether the strongest responses recover that region, and whether the matches jointly support the correct crop location.

Patch matching accuracy: retrieving the correct region. We measure the fraction of query patches whose top-1 match lies inside the reference source region:

$$T^1_{\text{patch}}(\ell) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{1}[m(i) \in G(B^\star)] \,. \tag{32}$$

A higher score indicates that query patches preferentially match their region of origin rather than unrelated parts of the source image.

This is a region-membership criterion: a match is counted as correct anywhere inside $G(B^\star)$. It does not require the source patch to coincide with the exact geometrically corresponding position. We therefore interpret this score together with the geometric localization metrics below.

For completeness, the top-k version counts a query patch as successful if at least one of its k highest-similarity source patches belongs to $G(B^\star)$:

$$T^k_{\text{patch}}(\ell) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{1}[M_k(i) \cap G(B^\star) \neq \emptyset] \,, \tag{33}$$

where $M_k(i)$ is the set of top-k matches. Our reported patch matching accuracy uses $k = 1$.

Region IoU: recovering the spatial extent of the crop. To obtain a response map over the source image, we assign each source patch its highest similarity to any query patch:

$$s(j) = \max_{i \in \mathcal{I}} A_\ell(i, j). \tag{34}$$

Thus, a source patch receives a high response when it strongly matches at least one patch in the query.

Let $n_G = |G(B^\star)|$. We select the n_G source patches with the highest response scores and denote this predicted patch set by $R(B^\star)$. Its overlap with the reference region is

$$T_{\text{region}}(\ell) = \frac{|R(B^\star) \cap G(B^\star)|}{|R(B^\star) \cup G(B^\star)|}. \tag{35}$$

A high score means that the strongest responses concentrate within the correct crop region.

The reference region size determines how many patches are selected, but not which patches are selected. This avoids choosing a similarity threshold and evaluates localization at a fixed region size. It does not make the task independent of crop ratio: larger crops occupy more of the source grid and can differ in matching difficulty.

Geometric localization: agreement on the crop position. We next test whether the top-1 matches support a consistent location for the query within the source image. This evaluation uses the known normalized crop dimensions

$$w^\star = x^\star_1 - x^\star_0, \qquad h^\star = y^\star_1 - y^\star_0.$$

For a geometrically correct match, the source position should approximately satisfy

$$r(m(i)) \approx (x^\star_0, y^\star_0) + (w^\star, h^\star) \odot u(i),$$

where $\odot$ denotes element-wise multiplication. Each query patch therefore casts a vote for the crop's top-left corner:

$$t_i = r(m(i)) - (w^\star, h^\star) \odot u(i). \tag{36}$$

Correct matches should produce similar votes, whereas inconsistent matches generally produce dispersed votes.

We select the location supported by the largest consensus set, following the robust-consensus principle of RANSAC [18]. Specifically, we evaluate at most 128 candidate votes, selected at uniformly spaced indices in the vote list. For each candidate, we count votes within an ℓ_∞ distance

$$\tau = \frac{1.5}{\min(H_{\text{grid}}, W_{\text{grid}})}. \tag{37}$$

The candidate with the largest count is selected as $\hat{t} = (\hat{t}_x, \hat{t}_y)$. This is a RANSAC-style translation consensus with known crop dimensions, rather than an unconstrained affine or homography estimation procedure.

The predicted box is

$$\hat{B} = (\hat{t}_x, \hat{t}_y, \hat{t}_x + w^\star, \hat{t}_y + h^\star), \tag{38}$$

with coordinates clipped to $[0, 1]$. We report its intersection-over-union with the reference box:

$$T_{\text{box}}(\ell) = \text{IoU}(\hat{B}, B^\star). \tag{39}$$

We also report the IoU_{50} success rate, obtained by averaging

$$T_{\text{succ}}(\ell) = \mathbf{1}[T_{\text{box}}(\ell) \geq 0.5] \tag{40}$$

over queries. Box IoU measures localization quality continuously, whereas IoU_{50} measures how often localization reaches a fixed overlap threshold.

Auxiliary inlier ratio. The fraction of votes supporting the selected location is

$$T_{\text{inlier}}(\ell) = \frac{1}{|\mathcal{I}|} \sum_{i \in \mathcal{I}} \mathbf{1}\left[\|t_i - \hat{t}\|_\infty \leq \tau\right]. \tag{41}$$

This measures internal geometric agreement among matches. A consistent set of incorrect matches can also produce a high inlier ratio, so this quantity should be interpreted alongside box IoU and localization success.

Aggregation and interpretation. We compute each metric separately for each query and then average over the 628 crops. Higher patch matching accuracy, region IoU, box IoU, and IoU_{50} indicate better performance under their respective criteria.

Together, these metrics assess cross-view region retrieval and geometric consistency under cropping and resizing. They do not measure whether pixel values or texture descriptors can be reconstructed from the tokens, nor do they establish robustness to arbitrary viewpoint changes. We therefore use this diagnostic alongside Local Appearance Decodability to distinguish spatial correspondence from recoverable appearance evidence.

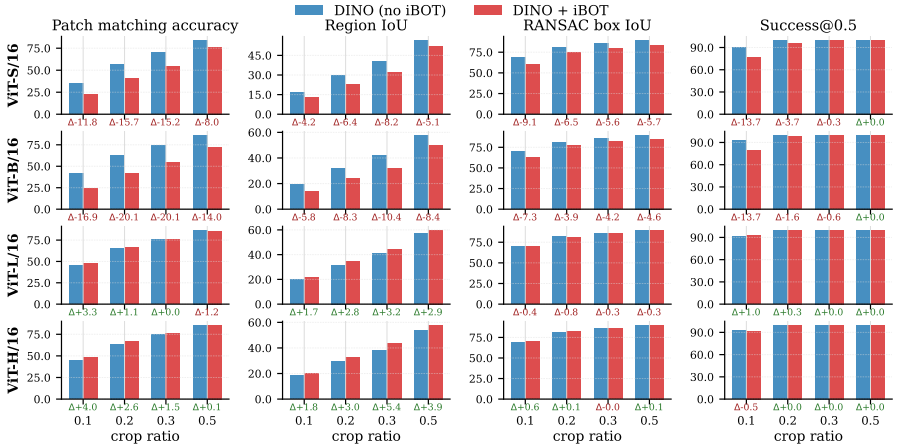

Figure 15: Local-to-global correspondence across crop area ratios. Each ratio specifies the approximate fraction of the source image used to construct a local query. Δ denotes $\text{DINO}_{+\text{iBOT}}$ minus DINO under matched pretraining settings and identical sampled crops. Positive values favor $\text{DINO}_{+\text{iBOT}}$ for all reported metrics.

G.4 Crop Ratios and Sampling

Figure 15 evaluates whether the paired differences persist when the query covers different fractions of the source image. Changing the crop ratio changes both the amount of available visual context and the scale difference between the resized query and source image. We therefore emphasize comparisons between models at the same crop ratio.

All reported correspondence results use a single query-sampling run with seed 0. The same sampled queries are used for each paired comparison. We do not estimate variability across sampling seeds or independent pretraining runs; the plots show point estimates, and small differences should not be interpreted as statistically established gains.

G.5 Visualization and Training Trajectories

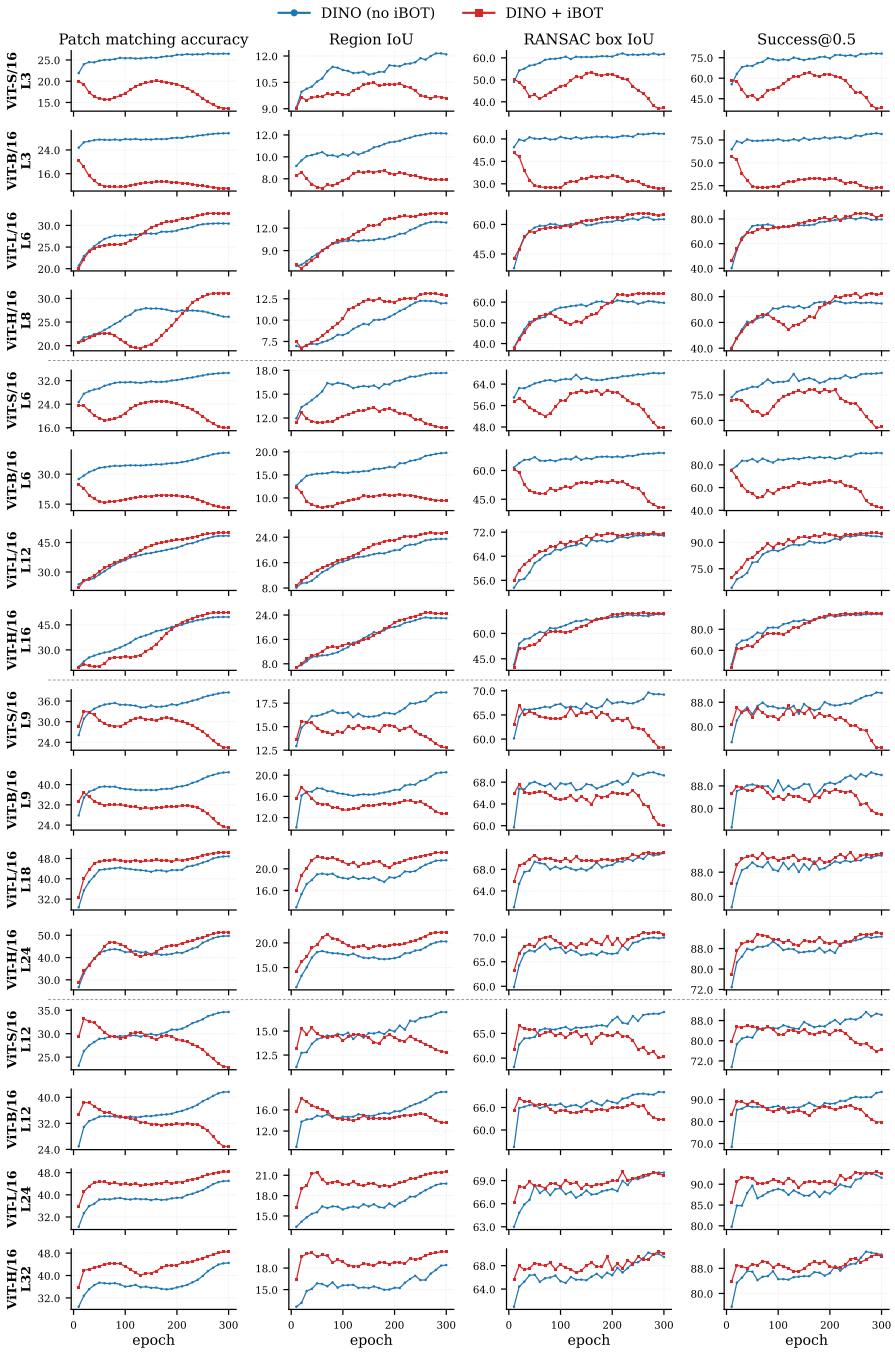

Figure 16: Training trajectories of local-to-global correspondence for DINO and DINO$_{+\text{iBOT}}$ across backbone scales. Rows show the backbone and evaluated layer: blocks 3/6/9/12 for ViT-S/B, 6/12/18/24 for ViT-L, and 8/16/24/32 for ViT-H. The sustained deterioration observed in smaller +iBOT models is alleviated in larger backbones.

Reading the qualitative examples. The qualitative results in Figure 17 show complementary aspects of the same matching process. Similarity heatmaps indicate where the query produces strong responses in the source image, patch matches show the individual associations, and estimated boxes

summarize their geometric consensus. These views help distinguish broadly distributed responses, incorrect individual matches, and failures to localize the query as a coherent region.

Figure 16 shows that the effect of iBOT depends on model scale. For ViT-S/16 and ViT-B/16, final-layer patch matching accuracy declines after an early peak under +iBOT training, while DINO generally improves or plateaus. Related deterioration appears in region-level and geometric localization metrics, although its magnitude varies. For ViT-L/16 and ViT-H/16, this sustained decline is no longer evident, and adding iBOT generally preserves or modestly improves correspondence, consistent with the DAVIS results.

These trajectories suggest that the correspondence cost observed in smaller models develops during optimization and can be alleviated at larger scales. However, this improvement coexists with reduced deep-layer appearance decodability in larger backbones. Thus, improved matching does not imply that pixel and texture evidence also becomes more accessible: the two diagnostics measure distinct properties of the learned representations.

G.6 Additional Visualization of Local-Global Correspondence

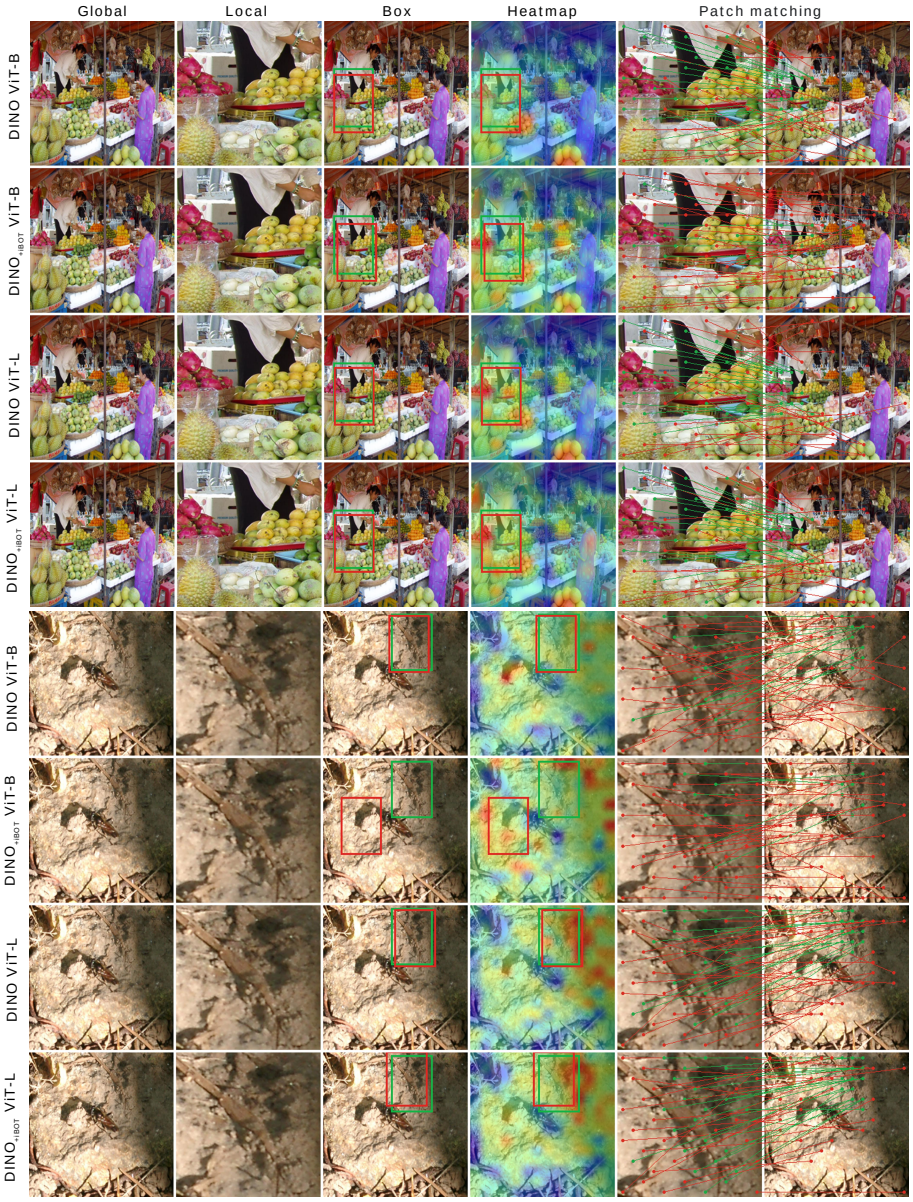

Figure 17: Additional visualization of local-global correspondence for ViT-B/16 and ViT-L/16 models of DINO and DINO$_{+\text{iBOT}}$.

H Extended Related Work

We organize the related literature around three questions: how self-distillation supervises patch representations, how prediction targets shape the information being learned, and how relational regularization changes these properties.

H.1 From Image-Level to Patch-Level Self-Distillation

DINO [9] learns visual representations by matching teacher and student predictions across augmented views. Despite using an image-level objective, it exhibits emergent spatial and semantic structure in its ViT features. iBOT [43] extends this approach with masked patch self-distillation: student tokens at masked locations predict the teacher's patch-level target distributions, alongside class-token distillation. DINOv2 [28] incorporates these image-level and patch-level objectives into a scalable training recipe, and DINOv3 [33] retains them while further scaling visual pretraining.

These developments establish the effectiveness of combining global and patch-level supervision for transferable visual representations. Our work examines the contribution of the patch objective through paired DINO and DINO$_{+\mathrm{iBOT}}$ experiments. The resulting semantic gains coexist with different local effects across model scales: correspondence deteriorates in smaller backbones, whereas larger backbones preserve or improve correspondence while exposing less linearly recoverable appearance evidence in deep layers. Thus, the spatial granularity of supervision alone does not determine which local properties are preserved.

H.2 Prediction Targets in Masked Image Modeling

Masked image modeling methods differ substantially in what they require the model to predict. MAE [22] and SimMIM [40] use pixel reconstruction, while MaskFeat [36] studies feature targets and finds HOG descriptors effective for visual pretraining. These objectives directly supervise aspects of image appearance. Other approaches predict learned targets: BEiT [4] uses discrete visual tokens, data2vec [3] predicts contextualized teacher representations, and I-JEPA [2] predicts target-block representations from a context block. iBOT similarly uses learned teacher targets, with an online tokenizer trained jointly with the student.

This distinction motivates our analysis of appearance decodability. Predicting a learned representation need not require retaining every pixel-derived attribute of the corresponding patch. However, the target type alone does not establish what remains accessible in the resulting features. Our experiments show that adding iBOT leaves appearance decodability largely unchanged or slightly improved in smaller models, while reducing it in deeper layers of larger models. The consequences of semantic patch supervision therefore depend on model scale and depth. Our diagnostic uses appearance descriptors as evaluation targets for frozen features; it does not introduce an additional reconstruction objective during pretraining.

H.3 Gram Anchoring and the Preservation of Patch Relations

DINOv3 [33] observes degradation in dense feature quality during extended training and introduces Gram anchoring to regularize pairwise patch similarities. The objective encourages the student's patch Gram matrix to match that of an earlier teacher checkpoint, preserving relational structure while allowing the feature vectors themselves to change.

Our experiments examine this intervention through the distinction between appearance and correspondence. We compare standard Gram anchoring with a variant initialized from a DINO checkpoint. In ViT-B, the DINO-based variant substantially improves correspondence, with limited and mixed changes in appearance decodability. In ViT-L, it improves deep-layer appearance decodability without consistently improving correspondence. These observations suggest that the effects of anchoring depend on both model scale and the reference representations.

This extends the analysis of relational regularization beyond aggregate dense-task performance. Matching within-view patch relations does not explicitly constrain cross-view consistency or the recoverability of pixel-derived attributes. Our results show that these properties need to be evaluated separately when assessing whether an intervention preserves local visual information.

I Limitations

Our study has several limitations. First, computational constraints restrict pretraining to ImageNet-1K for 300 epochs. Although we examine multiple backbone scales, this setting is substantially smaller than the data and compute regimes of DINOv2 and DINOv3. Whether the observed trade-offs persist under larger datasets and longer training remains to be established.

Second, we fix the iBOT loss weight to 1, following the standard weighting used in DINOv2 and DINOv3. Other weights or schedules may change the balance between semantic learning, appearance decodability, and correspondence. Our results therefore characterize this commonly used setting rather than an unavoidable consequence of masked patch self-distillation.

Third, our diagnostics measure specific aspects of local representations. Linear decodability does not capture information accessible only through nonlinear mappings, and crop-based correspondence does not cover the full range of viewpoint changes, deformations, and occlusions encountered in real-world matching. The region-membership score does not test exact pointwise matching, and geometric localization uses known crop dimensions. We also use single pretraining and query-sampling runs, so small differences are descriptive rather than statistically established. Our optical-flow results evaluate a learned head on the same Sintel videos used to fit it and do not measure held-out generalization. Finally, the observed dependence on model scale and Gram anchoring motivates possible explanations, but does not establish their underlying mechanisms. Further controlled experiments are needed to distinguish the roles of model capacity, optimization, and teacher representations. In particular, drop-path rates differ between ViT-S/B and ViT-L/H, and ViT-H uses a smaller batch size and scaled learning rate. The paired comparisons are controlled within each backbone, but cross-scale differences cannot be attributed to model capacity alone. Likewise, enabling iBOT also enables student masking; we do not separately isolate the effects of masking and its loss.

J Licenses for Existing Assets

This work uses existing datasets, benchmark annotations, model architectures, and public research code. We cite the corresponding papers throughout the main text and appendix, and use each asset only for research benchmarking under the terms stated by its original provider. The planned public release will contain our code, scripts, generated measurements, and derived metadata needed for reproduction; it will not redistribute raw third-party images or videos unless their original terms explicitly permit redistribution.

Table 3: Datasets and benchmarks used in the paper.

Asset	Use in this paper	Credit / source	License or terms of use
ImageNet-1K	Self-supervised pretraining, k-NN evaluation, and diagnostic image splits	Deng et al. [13]; official access terms	Access provided for non-commercial research and educational use under ImageNet terms. ImageNet does not own the underlying image copyrights, which remain with the original owners.
PASCAL VOC 2007 / 2012	VOC2007 object detection and VOC2012 semantic segmentation	Everingham et al. [16, 17]; VOC challenge pages	The official database-rights statement notes that VOC contains Flickr images, whose use must respect the corresponding Flickr terms of use.
MS COCO	Copy-detection distractors and whitening images	Lin et al. [26]; COCO terms of use	COCO annotations are released under CC BY 4.0; COCO images are subject to the Flickr terms and to the image-level license metadata provided by COCO.
$\mathcal{R}$Oxford5k / $\mathcal{R}$Paris6k	Image retrieval evaluation	Radenović et al. [30]; Revisited Oxford / Paris benchmark	Public benchmark data and annotations used for academic evaluation. The project page does not list a standard open-source license for the images; we do not redistribute the raw images.
ADE20K	Similarity heatmaps and reference semantic masks	Zhou et al. [42]; official terms of use	Images are provided for non-commercial research and education; copyright remains with their owners. The official terms specify BSD-3 terms for the annotations and software.
INRIA Copydays	Copy detection and local-to-global correspondence diagnostics	Douze et al. [15]; Copydays project page	INRIA holds copyright on the dataset images and distributes them "as is" without express or implied warranties. Used here only as a research benchmark; raw images are not redistributed.
DAVIS-2017	Video object tracking evaluation	Pont-Tuset et al. [29]; DAVIS 2017 repository	Released under the BSD license, per the official repository.
MPI Sintel	Optical flow evaluation	Butler et al. [6]; MPI-IS dataset page	Released under PS:License 1.0, with copyright held by the Max-Planck-Gesellschaft, per the official MPI-IS page.

Table 4: Existing codebases, model releases, and software assets used in the paper.

Asset	Use in this paper	Credit / source	License or terms of use
DINOv2	Pretraining framework and ViT implementation	Oquab et al. [28]; DINOv2 repository	Apache License 2.0 for the DINOv2 code and standard model weights, per the repository license.
MMSegmentation	Linear semantic segmentation evaluation	MMSegmentation repository	Apache License 2.0.
MMDetection	Frozen-backbone object detection evaluation	MMDetection repository	Apache License 2.0.
MMFlow	Frozen-backbone optical-flow evaluation and PWCNet implementation	MMFlow repository; PWCNet paper [34]	Apache License 2.0 for the MMFlow toolbox.
MAE	Raw-pixel masked-modeling baseline for local-appearance decodability	He et al. [22]; MAE repository	CC BY-NC 4.0, per the repository license.
SimMIM	Raw-pixel masked-modeling baseline for local-appearance decodability	Xie et al. [40]; SimMIM repository	MIT License, per the repository license.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes].

   Justification: The abstract and introduction describe a controlled empirical study of the scale-dependent effects of masked patch self-distillation. The claims concern downstream performance, the distinction between local appearance decodability and local-to-global correspondence, and the effects of Gram anchoring. These claims are supported by Sections 3, 4, and 5. Possible mechanisms are presented as hypotheses rather than established explanations.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes].

   Justification: Section I discusses the restricted pretraining data and compute scale, the fixed iBOT loss weight, the scope of the two diagnostics, and the absence of a conclusive mechanistic explanation for the observed effects. The conclusions are limited to the evaluated models and training settings.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A].

   Justification: The paper presents empirical analyses and diagnostic definitions. It does not introduce theoretical results requiring formal proofs.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes].

   Justification: Appendix B.2 describes the pretraining setup, model architectures, optimization hyperparameters, data, and downstream evaluation protocols. Appendices F and G provide the diagnostic definitions and implementation details needed to reproduce the annotation-free analyses.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [No].

   Justification: A complete public code release is planned, but an accessible release is not yet provided with this version. The release will include pretraining and evaluation scripts, configuration files, environment setup, and data-access instructions. The appendix documents the experimental protocols and cites the public datasets and codebases used.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes].

   Justification: Appendix B.2 specifies the pretraining objective variants, backbone details, optimizer, schedules, crop settings, and all downstream evaluation protocols. Appendices F and G further specify the diagnostic splits, sampling, probe fitting, and metrics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No].

Justification: The main comparisons use single pretraining runs, and the correspondence diagnostic uses a single sampling seed (0). We do not report repeated-run standard deviations, confidence intervals, or significance tests. Appendices G and I state these limits; small differences are interpreted descriptively.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes].

   Justification: Appendix B.2 reports the hardware setup, mixed-precision and sharding strategy, and Table 1 reports approximate training times and memory requirements for the 300-epoch DINO and DINO$_{+\text{iBOT}}$ runs across ViT-S/16, ViT-B/16, ViT-L/16, and ViT-H/16.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes].

   Justification: The work is a self-supervised vision representation analysis using standard public datasets and benchmark protocols, with no new human-subject data collection or deployment of a decision-making system.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [No].

    Justification: The paper discusses practical implications for downstream representation use in Appendix I, but it does not include a separate broader-impact discussion covering both positive and negative societal impacts. The work is foundational and diagnostic rather than a deployed application or a release of a high-risk capability.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A].

    Justification: The paper does not release a new high-risk model, image generator, language model, or scraped dataset. It analyzes self-supervised vision encoders using existing benchmark datasets and protocols.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes].

    Justification: Appendix J lists the datasets, benchmarks, codebases, and model releases used in the paper, together with their original credits, official sources, and licenses or terms of use. We follow these terms and do not redistribute raw third-party images or videos in the supplementary material unless permitted by the original asset provider.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A].

    Justification: The paper introduces new annotation-free diagnostics and documents their protocols in Appendices F and G, but it does not release a new dataset, model, or standalone asset in the current draft.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A].

Justification: The work does not involve crowdsourcing, user studies, or new research with human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A].

Justification: The paper does not conduct research with human subjects, so IRB approval or equivalent review is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A].

Justification: The core methodology and experiments do not use LLMs as an important, original, or non-standard component.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.