BELOW THE PATCH: SEPARATING INFORMATION FROM ACCESSIBILITY IN FROZEN VISION TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Vision transformers represent an image on a coarse patch grid, yet their frozen features are widely used for tasks requiring localization far below the patch stride. What, then, does "resolution" mean for a frozen representation? We argue that answering this question requires separating *information* from *accessibility*: a signal may be present in the input or representation while remaining inaccessible to a particular comparison and readout.

We study this distinction using controlled sub-pixel displacement. Raw 16×16 patches contain enough evidence to recover 2-D motion when local image structure is well conditioned (R^2 up to 0.98), even though a linear probe on the same pixels reports essentially zero. The ViT patch embedding is not the bottleneck either: on our grayscale input subspace it is injective, and a nonlinear decoder recovers displacement equally well from raw patch pairs and layer-0 token pairs ($R^2 \approx 0.875$). The apparent loss emerges when the two observations are collapsed into an aligned difference, where pixel and token inputs fail alike.

This bottleneck is structured rather than complete. The aligned difference retains the aperture-constrained component of motion—normal flow $\boldsymbol{d} \cdot \hat{\boldsymbol{g}}$—in a linearly accessible form. Tracking this coordinate through depth reveals that frozen checkpoints differ sharply in how much fine-scale geometry they keep accessible: MAE largely preserves it, whereas a label-supervised ViT reduces late-layer linear accessibility to nearly zero. Intervening on the fitted direction strongly controls the downstream readout, and combining the accessible scalar with the corresponding image geometry supports sub-pixel refinement on naturalistic video.

These results show that effective spatial resolution is not determined by patch stride alone. It depends on what information is present, how observations are compared, and what the readout can access.

1 INTRODUCTION

A vision transformer (ViT) represents an image on a coarse spatial lattice. A ViT-B/16 at 224 pixels, for example, produces only a 14×14 grid of tokens, with one token for each 16×16 image region. Yet frozen ViT features have become a substrate for tasks that demand localization much finer than this grid, and a growing line of work learns to upsample or otherwise refine them (Fu et al., 2024; Seo et al., 2026; Huang et al., 2025). Their success poses a basic question: *what does it mean for a coarse representation to contain fine-scale spatial information?*

There are at least three different questions hidden inside this one. A quantity may be *present* in the observations, *preserved* by the representation, and yet not be *accessible* to a particular comparison or readout. Conversely, a powerful decoder can combine representation content with its own inductive bias or auxiliary image information. Thus, obtaining a sub-patch output does not by itself tell us where the relevant information resides; neither does the failure of a probe establish that the information has been lost. The distinction between information and readout accessibility is well known in probing methodology (Hewitt & Liang, 2019; Voita & Titov, 2020; Pimentel et al., 2020). Here we ask how consequential that distinction becomes when the quantity of interest lives entirely *below the patch stride*.

We make the question concrete with sub-pixel displacement. This setting is useful because its geometry is understood. Under brightness constancy, local image changes constrain displacement

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

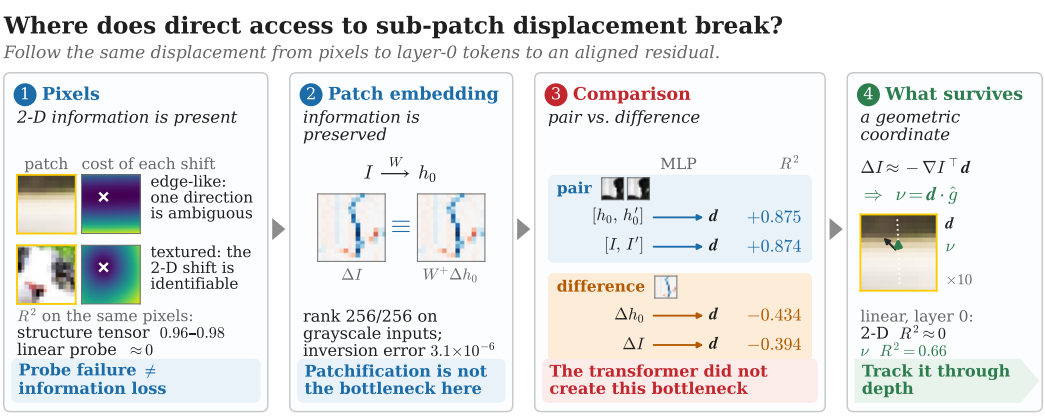

Figure 1: **Localising the direct-readout bottleneck below the patch. (1)** A patch is ambiguous when its gradients are nearly one-dimensional (the matching cost of an edge-like COCO patch is a trough) but texture identifies the 2-D shift (a basin; $\times$ marks the truth); except on the most edge-like quartile, a structure-tensor solve succeeds where a linear probe on the same pixels fails (Section 3.1). **(2)** On grayscale inputs the layer-0 projection is injective and its pseudo-inverse recovers the pixel difference essentially exactly (Section 3.2). **(3)** The same MLP recovers displacement from raw-pixel and token *pairs* but fails on both *differences*, so this direct-readout bottleneck is not introduced by the transformer (Section 3.3). **(4)** The difference still exposes $v_n = \boldsymbol{d} \cdot \hat{\boldsymbol{g}}$ linearly (sketch: the patch of (1), arrows $10\times$; Section 3.4), and Sections 4 to 6 follow this coordinate (construction: Section A).

through image gradients. A single gradient gives only one component of motion—the classical aperture problem—but a patch contains many gradients, and sufficiently diverse local structure can determine the full 2-D displacement. This gives us an unusual advantage for studying a learned representation: before asking what a probe can decode, we can establish what the observations themselves make identifiable.

That control immediately changes the interpretation. On well-conditioned raw patches, a classical structure-tensor estimator recovers 2-D displacement at $R^2 = 0.96$–0.98, while a linear probe on exactly the same pixels gives $R^2 \approx 0$. The probe has not discovered an information-theoretic limit; it has encountered a mismatch between the geometry of the problem and the geometry of the readout. We therefore use the rest of the paper to ask a more precise question:

> **As sub-patch displacement passes from pixels to tokens and through a frozen transformer, where does direct access to it change, and what part remains accessible?**

Figure 1 answers the first half by elimination. The patch embedding is not the critical step: on our grayscale input subspace the projection into a layer-0 token is injective, and the same nonlinear decoder recovers 2-D displacement from a layer-0 token *pair* as well as from the raw patch pair ($R^2 = 0.875$ versus 0.874), so whatever sub-patch evidence existed before the transformer is still available at its input. The crucial change comes from how the two observations are compared. Collapsing the pair into an aligned *difference* makes the same decoder fail for tokens and for raw pixels alike ($R^2 = -0.434$ and -0.394). This matched pixel-space control is the hinge of the paper: the failure is not produced by transformer processing, but by a difference that no longer exposes the reference-image structure needed to turn local intensity change into a unique 2-D displacement — restoring that geometry lets a classical estimator solve it again. Our claims therefore concern the *direct accessibility of displacement from the aligned difference*, not whether a frozen ViT contains it in some unrestricted sense.

Differencing does not, however, leave an unstructured remainder. One physically privileged coordinate stays directly accessible: the aperture-constrained normal flow $v_n = \boldsymbol{d} \cdot \hat{\boldsymbol{g}}$, the displacement projected onto the local gradient direction, linearly readable at layer 0 at $R^2 \approx 0.66$ where full 2-D displacement is not. This turns an apparent failure into a measurement: rather than asking whether a network "has" sub-patch information, we can track a specific fine-scale quantity present at its input

and ask what depth does to its accessibility. Across seven frozen checkpoints and three image domains the answer differs sharply, and the separation grows through depth rather than at the input: MAE preserves strong late-layer accessibility, whereas a label-supervised ViT from a comparable layer-0 start reduces it to about zero. The checkpoint comparison is observational, but a controlled ViT-S/16 study varying only the objective confirms that the objective can alter the *rate* at which accessibility is lost. Finally, accessibility is not a storage location: for MAE a single supervised direction suffices for all of the tested readout and editing it inside the frozen forward pass strongly moves the downstream estimate, yet its orthogonal complement still yields substantial signal after refitting — a *load-bearing access direction*, not the sole place displacement is represented. Combining the accessible scalar with the geometry that resolves its aperture ambiguity gives useful sub-pixel refinement on MPI-Sintel, degrading exactly where the local model should fail.

Contributions. Our contributions are conceptual as well as empirical:

1. **We localise an apparent loss of sub-patch resolution to the comparison/readout rather than patchification**: matched pixel and token controls show 2-D displacement recoverable from raw and layer-0 pairs but not directly from either aligned difference, separating what a representation contains from what a comparison makes accessible.
2. **We identify a geometry-matched coordinate for measuring sub-patch accessibility through depth**: normal flow stays linearly accessible in the aligned difference, and across checkpoints its accessibility diverges strongly through depth, with controlled training showing the objective can change its rate of loss.
3. **We distinguish a load-bearing access direction from a storage location, and connect it to behaviour**: interventions give the fitted direction a large, signed effect on the fixed readout while the complement retains much of the signal, and combining the coordinate with its image geometry supports real sub-pixel correspondence.

2 Related work

Recovering sub-patch structure from frozen features. Patch-based backbones support predictions finer than their token grids suggest: FeatUp, Upsample Anything, LoftUp, FeatSharp and AnyUp learn or optimise feature-upsampling operators (Fu et al., 2024; Seo et al., 2026; Huang et al., 2025; Ranzinger et al., 2025; Wimmer et al., 2026). They establish that high-resolution outputs can be recovered around frozen features; we ask instead which sub-patch information is already available, how the comparison and readout control access to it, and how that access changes through depth. FeatUp and Upsample Anything serve as our correspondence baselines.

Probing geometry and accessibility through depth. Probe performance need not measure what a representation contains: capacity, control tasks and description length all affect what success or failure licenses (Hewitt & Liang, 2019; Voita & Titov, 2020; Pimentel et al., 2020). We use the distinction in the failure direction, showing with matched pixel/token controls and an exact inverse that a failed readout need not imply information loss. Equivariance and geometric content in vision have been studied across layers and backbones (Lenc & Vedaldi, 2015; El Banani et al., 2024; Shkolnikov, 2026); our quantity is continuous, lies entirely below the patch stride, and is already accessible at the transformer input, so for most checkpoints it is kept or lost rather than built up (Figure 2). Image-level objectives can discard equivariant or patch-local information that a reconstruction loss keeps (Wang et al., 2024; Darcet et al., 2024); our public-checkpoint comparison is observational; Section 4.3 varies the objective directly.

Access directions and dictionaries. Finding a useful direction differs from locating all the information about a variable. Group-structured and sparse dictionaries study invariant or equivariant features and when learned dictionaries support interpretation or control (Gorton, 2024; Makelov et al., 2025; Chanin et al., 2025; Erdogan & Lucic, 2025; Ma et al., 2026; Gao et al., 2025). We use a supervised direction as a measurement instrument, not a claimed unique feature — load-bearing for the fixed readout, with substantial recoverability left in its complement — and for dictionaries we measure error along that direction rather than total reconstruction fidelity.

3 LOCALISING THE SUB-PATCH ACCESSIBILITY BOTTLENECK

A failed readout may mean the observations do not identify the quantity, the embedding discards it, or it survives but is hard to extract after the observations are combined. We separate these by following one displacement through raw patches, the patch embedding, a layer-0 token pair and their aligned difference (Figure 1; matched controls in Table 16).

3.1 PIXELS: 2-D DISPLACEMENT IS IDENTIFIABLE WHEN LOCAL STRUCTURE IS RICH

For a small rigid displacement $\boldsymbol{d}$, brightness constancy gives

$$I'(\boldsymbol{x}) - I(\boldsymbol{x}) \;\approx\; -\nabla I(\boldsymbol{x})^{\top}\boldsymbol{d}, \tag{1}$$

so a single pixel constrains only the projection $v_{\mathrm{n}} = \boldsymbol{d}\cdot\hat{\boldsymbol{g}}$ with $\hat{\boldsymbol{g}} = \nabla I/\|\nabla I\|$ — the aperture problem (Lucas & Kanade, 1981; Horn & Schunck, 1981). A patch stacks many such constraints into the structure tensor $M = \sum_{\boldsymbol{x}} \nabla I \nabla I^{\top}$, and $\boldsymbol{d}$ is identifiable where M has rank two (Shi & Tomasi, 1994): an edge leaves one direction weakly constrained, texture determines both (Figure 1, panel 1). This calibrates the probe. Stratifying by $\lambda_{\min}(M)$, a structure-tensor solve reaches $R^2 = 0.961$, 0.976 and 0.982 on the upper three quartiles at about $0.05\,\mathrm{px}$ median error, while a linear probe on the same pixels gives $R^2 \approx 0$ in every quartile (Table 18). The pixels do not lack 2-D displacement; its relation to local image change is conditioned by image geometry, which a generic linear probe cannot express. Real patches show the same signature, erring along the direction their own structure constrains least on all four image sources we test (Figures 5 and 7). Before any transformer is involved, probe failure and absence of information have already come apart.

3.2 PATCH EMBEDDING: THE AVAILABLE IMAGE EVIDENCE IS PRESERVED

Our inputs are grayscale, so a 16×16 patch has 256 values, mapped by a linear projection W into a 768-dimensional layer-0 token. On this input subspace W has rank $256/256$ and condition number 110, so distinct patches cannot collide. Numerically, W^{+} applied to the *actual* layer-0 token difference reconstructs the pixel difference to relative error 3.1×10^{-6}, and given the same reference-image gradients ∇I as the pixel-space solver, the reconstruction yields $R^2 = 0.944$ on the upper three quartiles at $0.096\,\mathrm{px}$ median error, identical to three decimals to solving from raw pixels (Figure 1, panel 2). This isolates the embedding; it does *not* show that the difference alone determines 2-D displacement, since the solver also receives ∇I. Positional embeddings cancel because aligned tokens share a grid position. What remains is how the two observations are compared.

3.3 PAIR VERSUS DIFFERENCE: DIRECT ACCESS CHANGES AT THE COMPARISON

We present the same observations kept as a pair or collapsed into the aligned difference used by the residual readout studied here (other systems compare by correlation, matching or cross-attention). Given the layer-0 token pair $[h_\ell(\boldsymbol{x}), h_\ell(\boldsymbol{x}')]$, a nonlinear decoder recovers 2-D displacement at $R^2 = 0.875$, matching 0.874 from the raw patch pair; given the difference, the same decoder scores -0.434 on tokens and -0.394 on pixels (Figure 1, panel 3; Table 17). The decisive fact is not that the difference decoders fail but that the same pair-to-difference transition appears before and after patch embedding: **the direct 2-D readout failure is not introduced by the transformer**; it appears when two observations are reduced to an aligned difference. This is not information destruction — the classical solver recovers 2-D from the pixel difference, and from the inverted token difference at $R^2 = 0.944$, once the reference gradients the difference lacks are restored. Collapsing a pair into a difference changes what is *directly accessible*, already before the transformer, and every claim below is bounded to this aligned-difference readout.

3.4 WHAT REMAINS ACCESSIBLE: NORMAL FLOW

The difference does not leave an unstructured remainder. Equation (1) singles out the normal flow $v_{\mathrm{n}} = \boldsymbol{d}\cdot\hat{\boldsymbol{g}}$, the component of motion normal to the local contour, which brightness change constrains directly. On MAE's layer-0 aligned residuals a linear probe gives $R^2 \approx 0$ for 2-D displacement but $R^2 = 0.66$ for normal flow (Figure 1, panel 4): a poor coordinate system for 2-D motion, the difference exposes the aperture-constrained component linearly. This is the quantity the rest of the

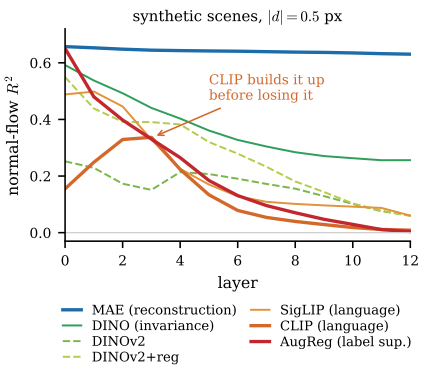

Figure 2: **Normal-flow accessibility evolves differently through depth.** Layerwise $R^2_{v_n}$ on synthetic scenes for all seven checkpoints at $|\boldsymbol{d}| = 0.5\,\text{px}$, the regime studied throughout. Normal flow is already accessible at layer 0 and the trajectories separate as computation proceeds: MAE stays close to its input accessibility while several checkpoints progressively lose direct linear access. CLIP is the exception, starting weak and building accessibility over the first blocks before losing it. At 2 px, outside the local regime, the trajectories change substantially (Figure 10), so we read no universal ordering off them.

paper tracks. At layer ℓ, with the target token first transported to the matching coarse patch, we fit a direction $\hat{\boldsymbol{w}} \in \mathbb{R}^{768}$ on the aligned residual $R_\ell(i) = h_\ell(\boldsymbol{x}'_i) - h_\ell(\boldsymbol{x}_i)$, so that $\hat{v_n}(i) = \hat{\boldsymbol{w}}^\top R_\ell(i)$, and recover the vector by orientation-weighted least squares over a 3×3 neighbourhood,

$$\hat{\boldsymbol{d}} \;=\; \arg\min_{\boldsymbol{d}} \sum_i w_i \big(\hat{v_n}(i) - \boldsymbol{d} \cdot \hat{\boldsymbol{g}}_i\big)^2, \qquad w_i = \|\nabla I\|_i, \tag{2}$$

which recovers a $0.5\,\text{px}$ displacement from frozen MAE tokens to $0.10\,\text{px}$ median error. For larger motion, $\boldsymbol{d} = kp + r$ with $p = 16$ and $k = \text{round}(\boldsymbol{d}/p)$: tokens are compared after nearest-patch transport and the readout estimates only the sub-patch remainder r.

Scope and hygiene. The same geometry predicts where this readout stops working. The signal is contrast-gated and periodic, so its range is bounded by the texture frequencies present: on our synthetic sweep (highest frequency 0.30 cycles/px) the predicted half-period ambiguity near 1.6 px is where pixel-space Lucas–Kanade breaks down (median error 0.01–0.05 px below 1 px, 0.57 px at 2 px) — a property of that band, not a universal limit. It also predicts transfer from global translations to varying flow, and that a local linear estimator should aid sub-pixel refinement but lose to searching an upsampled similarity surface at larger motion; Section 6 tests both. We compare checkpoints by raw $R^2_{v_n}(\ell)$; the layer-0-normalised ratio ρ only describes rates of loss. Bilinear warping leaks the applied shift (43% in our audit), so no evaluation pair is warped (Section A).

The original question — whether a patch-based representation "contains" sub-patch information — has become a measurable one: normal flow is linearly accessible at the transformer input, so we can ask **whether that accessibility survives, emerges or disappears through depth**.

4 Sub-patch accessibility diverges through depth

Normal flow is already linearly accessible from the aligned residual at the transformer input, so we can ask what learned computation does to that accessibility. We hold the target and readout class fixed and follow the same coordinate, $v_n = \boldsymbol{d} \cdot \hat{\boldsymbol{g}}$, layer by layer; the checkpoints turn out to differ less in how accessible it is at their input than in what their blocks do to it.

4.1 Following the same geometric coordinate through depth

We compare seven frozen checkpoints. Five share ViT-B/16 at 224 pixels — MAE (He et al., 2022), DINO (Caron et al., 2021), CLIP (Radford et al., 2021), SigLIP (Zhai et al., 2023) and a label-supervised AugReg ViT (Steiner et al., 2022) — so patch size, token count and architecture are matched; DINOv2 with and without registers (Oquab et al., 2024; Darcet et al., 2024) is ViT-B/14 and reported alongside. For every checkpoint and layer we fit the readout of Section 3.4 and report held-out $R^2_{v_n}(\ell)$ at $|\boldsymbol{d}| = 0.5\,\text{px}$ on analytic scenes, CelebA-HQ and COCO, three seeds each (Table 3). The checkpoints also differ in data, schedule and augmentation, so the comparison is observational.

Table 1: **The checkpoints differ sharply in what remains directly accessible after twelve blocks.** Each cell is $R^2_{v_n}(0) \to R^2_{v_n}(12)$ at $|\boldsymbol{d}| = 0.5$ px, three seeds. Differences between public checkpoints are observational; DINOv2 variants are ViT-B/14. Intervals: Table 3.

backbone	pretraining objective	synthetic $R^2(0) \to R^2(12)$	CelebA-HQ $R^2(0) \to R^2(12)$	COCO $R^2(0) \to R^2(12)$
MAE	pixel reconstruction	**0.657 → 0.631**	**0.578 → 0.605**	**0.530 → 0.519**
DINO	augmentation invariance	0.592 → 0.256	0.539 → 0.236	0.484 → 0.075
DINOv2	invariance + iBOT	0.252 → 0.060	0.207 → 0.082	0.173 → 0.009
SigLIP	sigmoid language contrastive	0.488 → 0.060	0.464 → 0.167	0.406 → 0.050
DINOv2+reg	invariance + registers	0.550 → 0.059	0.473 → 0.103	0.426 → 0.013
CLIP	language contrastive	0.155 → 0.009	0.414 → 0.015	0.180 → -0.006
AugReg ViT	label supervision	0.651 → 0.004	0.574 → 0.017	0.522 → -0.013

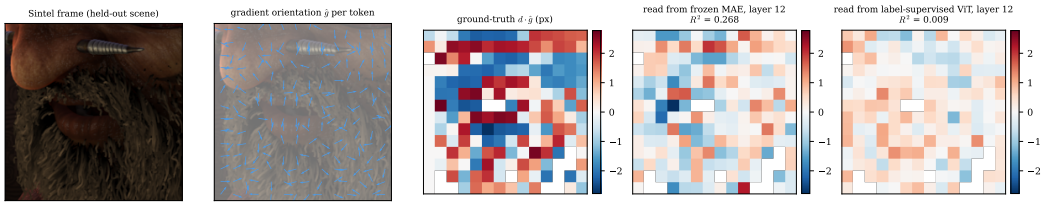

Figure 3: **The depth effect as a spatial field on non-uniform motion.** A held-out MPI-Sintel frame, its per-token $\hat{\boldsymbol{g}}$, the ground-truth $\boldsymbol{d} \cdot \hat{\boldsymbol{g}}$, and that field read linearly from frozen layer-12 residuals of MAE and of the supervised ViT. The crop is the first drawn under the evaluation seed (Section E); these R^2 use Sintel's own flow, not those of Table 1.

4.2 The separation is created largely through depth

Every checkpoint begins with measurable normal-flow accessibility, but subsequent computation treats it very differently (Figure 2): most trajectories decline from layer 0, MAE stays comparatively stable, DINO loses a substantial fraction, and several late-layer representations approach zero. The same holds beyond analytic scenes: Table 1 writes each endpoint pair as $R^2_{v_n}(0) \to R^2_{v_n}(12)$ on all three domains. On synthetic scenes MAE and the label-supervised ViT start from nearly the same value (0.657 and 0.651) but end at 0.631 and 0.004: the difference is created almost entirely by what twelve blocks do to an initially accessible signal, with DINO in between ($0.592 \to 0.256$). MAE keeps strong late-layer accessibility in every domain and DINO declines but stays measurable; among the rest the order is not stable — SigLIP and the DINOv2 variants exchange places, and SigLIP reaches 0.167 on CelebA-HQ while being much lower elsewhere — so we emphasise the depth-induced divergence rather than a seven-model ranking. Figure 3 shows the same separation as a spatial field under Sintel's non-uniform motion, a change of regime rather than a second estimate.

4.3 The training objective can change the rate of accessibility loss

Our own prior hypothesis was too simple: we expected *reconstruction* > *supervised* ≈ *contrastive* > *augmentation-invariance-heavy*, yet label supervision loses far more accessibility than DINO, and we do not retrofit the hypothesis. A controlled ViT-S/16 family at ImageNet-100 scale, with architecture, data, resolution and schedule fixed and only the objective varied (Section L), shows that the objective changes the *rate* at which linear normal-flow accessibility is lost: reconstruction retains substantially more than strong supervision, with invariance and milder supervision in between. It does not explain the public ordering, so causal language is reserved for that rate: fine-scale accessibility is something learned computation can preserve or suppress, not a quantity fixed at patchification.

The MAE trajectory is not an evaluation-mask artefact. MAE pretrains with 75% of patches masked while we present the full image, but sweeping its evaluation mask ratio to 0.75 and scoring only tokens visible in both members of a pair leaves layer-12 accessibility at 0.921 of its own layer-0 value (Table 10). High-norm outlier tokens, positional interpolation and specificity to global

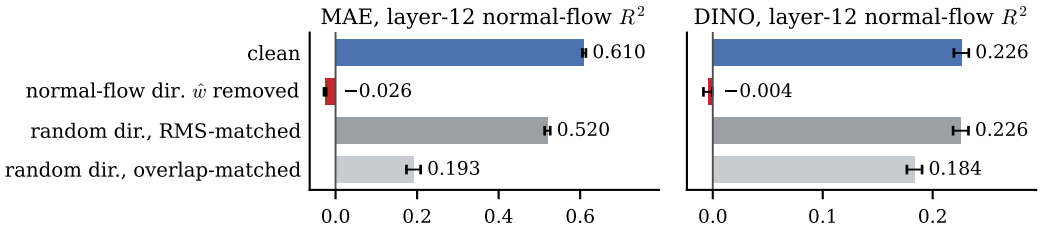

Figure 4: **The normal-flow direction is load-bearing for the fixed readout.** Removing $\hat{\boldsymbol{w}}$ at layer 6 collapses the layer-12 normal-flow readout, while random directions matched on norm or on first-order downstream overlap leave far more of it (Table 11; 95% bootstrap intervals, three seeds).

translation are ruled out in Section G; a periodic-stimulus diagnostic follows the same trend but is largely generic, so we do not count it as evidence (Section K).

Learned computation can thus preserve, attenuate or nearly eliminate access to a coordinate present at its input. What, then, does a successful readout direction mean inside the representation — a convenient route to the variable, or the place it is stored?

5 A LOAD-BEARING ACCESS DIRECTION IS NOT A STORAGE LOCATION

A scalar linear target always yields one ridge direction, so the existence of $\hat{\boldsymbol{w}}$ is no evidence for a one-dimensional code. We ask whether a nonlinear decoder loses anything when restricted to $R \cdot \hat{\boldsymbol{w}}$, what new decoders recover once $\hat{\boldsymbol{w}}$ is removed, and what happens when $\hat{\boldsymbol{w}}$ is edited inside the frozen forward pass with the readout held fixed.

5.1 ONE DIRECTION PROVIDES A STRONG ACCESS PATH BUT DOES NOT EXHAUST THE SIGNAL

For MAE at layer 12 a two-layer MLP on the full residual reaches $R^2 = 0.713$ and the same decoder given only the scalar $R \cdot \hat{\boldsymbol{w}}$ reaches 0.717: discarding the other 767 coordinates costs nothing, and DINO shows the same tendency less completely (0.316 against 0.277). That is a strong access path, not exclusive storage: projecting $\hat{\boldsymbol{w}}$ out and training new decoders *from scratch* on the 767-dimensional complement still recovers $R^2 = 0.518$ for MAE and 0.223 for DINO, about 73% and 70% of the full-residual MLP (Table 7). The direction is also a genuine axis of single-image activation rather than an artefact of differencing — for MAE it carries 0.66% of single-image activation variance and is the 31st of 768 principal components, while local contrast, orientation and position together explain $R^2 \leq 0.002$ of it (Table 8) — without this implying a uniquely semantic feature.

5.2 THE DIRECTION IS LOAD-BEARING FOR THE FIXED DOWNSTREAM READOUT

Recoverability after refitting and dependence of a *fixed* readout are different questions. With the layer-12 probes fit on clean activations and never refit, we project $\hat{\boldsymbol{w}}$ out of the residual stream at layer 6 and continue the frozen forward pass. Removing this one direction (7.9% of an MAE token's norm at layer 6) takes the MAE readout from $R^2 = 0.610$ to -0.026 and DINO's from 0.226 to -0.004 (Table 11). An RMS-matched random direction leaves MAE at 0.520 and DINO at 0.226, and a stronger random control matched on first-order downstream overlap does more collateral damage yet still leaves more of the readout than removing $\hat{\boldsymbol{w}}$ (Figure 4).

Moving the coordinate gives a signed effect. Deletion shows dependence, not that variation along $\hat{\boldsymbol{w}}$ moves the readout. Injecting $\alpha\,\sigma\,\hat{\boldsymbol{w}}$ at layer 6 for $\alpha \in [-2, 2]$, with σ the standard deviation of $h_\ell(\boldsymbol{x}) \cdot \hat{\boldsymbol{w}}$, moves MAE's layer-12 readout with the sign and size of the intervention at $5.79\,\mathrm{px}/\sigma$, against -0.063, $+0.023$ and $-0.026\,\mathrm{px}/\sigma$ for random directions each scaled by its own σ — 154× more effect per σ and 44× per unit norm (Table 15). The near-linearity is expected of a small perturbation read by a linear probe; what informs is the slope's size and specificity. Deletion and injection show that $\hat{\boldsymbol{w}}$ is a high-gain, load-bearing control direction for *this* readout; with the

Table 2: **Accessible normal flow supports sub-pixel refinement where the local model applies.** PCK@1 from the same coarse patch match, on MPI-Sintel and on translations up to 40 px; the last row is fit only on CelebA translations and transferred unchanged. Three seeds (intervals: Table 3).

	MPI-Sintel		$\|\boldsymbol{d}\| \leq 40$ px translation	
refinement	MAE	DINO	MAE	DINO
predict no motion	0.163	0.163	—	—
patch argmax (no refinement)	0.162	0.162	0.017	0.017
similarity-peak parabola	0.063	0.048	0.030	0.028
bilinear $\times 16$	0.134	0.156	0.082	0.080
Upsample Anything	0.120	0.109	0.040	0.043
FeatUp	0.231	0.205	0.108	**0.108**
normal-flow readout, in-domain	**0.353**	0.258	**0.114**	0.073
normal-flow readout, fit on CelebA	0.334	**0.261**	—	—

complement result it is not a unique storage location, and they act on the readout path, not on a native displacement output.

Target-aligned fidelity matters more than global reconstruction. The same distinction applies to unsupervised decompositions, which can preserve most residual variance while discarding the component a readout needs. Measured along the access direction, $R^2_{\hat{\boldsymbol{w}}} = 1 - \mathbb{E}[(\hat{\boldsymbol{w}}^\top (R - \hat{R}))^2] / \mathrm{Var}(\hat{\boldsymbol{w}}^\top R)$, a TopK sparse autoencoder (Gao et al., 2025) on MAE layer 12 with total reconstruction $R^2 = 0.712$ keeps only $R^2_{\hat{\boldsymbol{w}}} = 0.453$, losing under a third of the variance overall but more than half along $\hat{\boldsymbol{w}}$ (Section J): **global reconstruction fidelity is not a reliable proxy for the accessibility of a particular variable**, and scaling the tested TopK recipes narrows but does not close the gap.

A single direction can thus be a high-gain route the readout depends on without being where the variable is stored. If that coordinate is genuinely useful rather than merely decodable, combining it with the image geometry that resolves the aperture constraint should improve sub-patch localisation where that geometry applies.

6 From accessible geometry to sub-pixel correspondence

Normal flow alone does not determine a 2-D displacement, but Equation (2) supplies exactly the complementary information identified in Section 3: neighbouring gradient orientations resolve the local aperture ambiguity. If the scalar readout captures useful sub-patch geometry, the combination should improve localisation where the local motion model applies and stop helping where it breaks. This is a test of whether the accessible coordinate does real geometric work and fails where predicted, not a proposed correspondence method.

6.1 Accessible normal flow refines non-uniform motion

MPI-Sintel (Butler et al., 2012) provides exact per-pixel flow and occlusions; unlike the global translations used to fit the readout, its displacement field varies across the image, with median magnitude 2.4 px. Crops are taken at native resolution and scenes are split disjointly between fitting and evaluation. Every refinement starts from the same coarse patch match, and the upsampling baselines — bilinear, Upsample Anything (Seo et al., 2026) and FeatUp (Fu et al., 2024) — get the same ± 1-patch window and a parabolic sub-sample fit on their own similarity surface (Section C). We report PCK@1, the fraction of tokens whose endpoint lands within one pixel of the truth. The normal-flow readout reaches 0.353 for MAE and 0.258 for DINO against FeatUp's 0.231 and 0.205 (Table 2), with MAE's median endpoint error 1.49 against 2.10 px. The null is not trivial here: 16% of tokens move less than a pixel, so predicting zero already scores 0.16, but on tokens that move more than a pixel, where it scores exactly zero, the readout reaches 0.290 against FeatUp's 0.183.

6.2 THE GAIN COMES FROM THE STRUCTURE OF THE READOUT, NOT SUPERVISION ALONE

Our readout is supervised and the upsamplers are not, so the strongest alternative is a generic supervised decoder. Given the same pairs, labels, 3×3 neighbourhood of full residuals and image gradients, a two-layer MLP decoding 2-D directly reaches only $\mathrm{PCK@1} = 0.194$ (0.191 with depth and learning rate tuned) against 0.162 unrefined (Table 12). Matched contrasts on identical crops isolate the rest (Table 4): replacing an MLP per-token map with a linear scalar projection is worth $+0.041$ $[+0.035, +0.048]$ and replacing a learned aggregator with Equation (2) a further $+0.030$ $[+0.021, +0.040]$, while fitting the same linear map by ridge rather than SGD changes $\mathrm{PCK@1}$ by only $+0.0006$, inside the pre-specified equivalence margin. These are estimator contrasts — the MLP contains the linear solution, so its deficit reflects inductive bias and sample efficiency, not a need for linear decoding. The coordinate is also not arbitrary — crossing the geometry that supervises the scalar against the geometry the aggregator assumes drops even self-consistent rotated systems from 0.358 to 0.278, 0.191 and 0.166 at $30°$, $60°$ and $90°$ (Table 13 and Section D). Pixel-space Lucas–Kanade remains the reference (0.398 $[0.372, 0.428]$ against our 0.358 $[0.335, 0.383]$): frozen features do not beat the pixels' own local evidence, but after twelve blocks enough of it stays linearly accessible for useful sub-pixel estimation (Figure 6).

6.3 TRANSFER AND FAILURE FOLLOW THE PREDICTED GEOMETRY

The local coordinate transfers: a probe fit only on CelebA global translations and applied unchanged to Sintel reaches $\mathrm{PCK@1} = 0.334$ (MAE) and 0.261 (DINO); with no equivalence test run, we claim only similarity. **It degrades where local rigidity fails:** split at the median within-patch flow spread, it drops $0.512 \to 0.269$ against FeatUp's $0.320 \to 0.184$, somewhat faster, as an aggregator assuming locally shared displacement must. **Outside the local regime the advantage disappears** (Figure 8): on translations to $40\,\mathrm{px}$, searching an upsampled surface repairs coarse patch errors a local correction cannot — for DINO, FeatUp 0.108 against 0.073 — and on cross-instance SPair-71k correspondence (Min et al., 2019) the readout gives no gain (DINO: $\mathrm{PCK@0.05}$ 0.108 against 0.113 unrefined). **Removing the access direction removes the refinement:** deleting $\hat{\boldsymbol{w}}$ at layer 6 returns our refinement to about the unrefined baseline while FeatUp on the same edited features barely changes (Section C). Normal flow thus survives in some frozen representations in a form a simple readout can exploit for real sub-patch work and stops helping where the local geometry says it should: evidence about *accessible representation content*, not a superior correspondence method.

7 LIMITATIONS AND CONCLUSION

Limitations. Our claims concern one route from paired observations to an aligned residual, not everything decodable from a frozen ViT, and injectivity is exact only on the 256-dimensional grayscale input subspace. The aggregation assumes locally constant displacement; the $1.6\,\mathrm{px}$ transition belongs to the stimulus band that measured it; and the readout degrades as motion becomes non-rigid, does not help cross-instance correspondence, and loses its advantage once coarse matching errors dominate. The model comparison is associational, the controlled miniature isolates an effect on the *rate* of loss, not the public ordering, and our interventions manipulate a load-bearing readout path, not a unique storage location or a native displacement variable of the ViT.

Conclusion. A token grid is not, by itself, the spatial resolution of a representation. In our setting sub-patch displacement survives the patch projection — a layer-0 token pair exposes the same 2-D displacement as the raw patch pair — and what changes is *access*: collapsed into an aligned difference, full 2-D displacement is no longer directly exposed to a generic readout, while its aperture-constrained component, normal flow, stays simple enough to read linearly. Through that coordinate, frozen checkpoints differ sharply in whether fine-scale geometry stays accessible through depth; a single direction is a causally load-bearing route without confining the information; and the scalar, recombined with the geometry that resolves its ambiguity, does real sub-pixel work and fails where that geometry predicts. The broader lesson is that effective representational resolution depends on *what is present*, *how observations are compared* and *what the readout can reach*. Below the patch, the useful question is not whether information is "in the representation" but what computation leaves accessible, in which coordinates, and for which downstream operations.

AI USE STATEMENT

We used generative AI tools for coding and paper writing assistance. We did not use them to originate the paper's research question, to produce experimental results, or to write the bibliography. We have reviewed all AI-assisted work and take responsibility for the final content, including text, claims and artifacts.

REPRODUCIBILITY STATEMENT

All experiments use publicly available checkpoints and datasets (CelebA-HQ, COCO, MPI-Sintel, SPair-71k, ImageNet-100 for the controlled runs). The synthetic renderer is analytic and specified in Section A; the interpolation-free real-pair construction and its leak audit are given there. Section B gives the probe fitting procedure and the autoencoder recipe, and Section C gives the full correspondence protocol including our reimplementation of Upsample Anything and the bit-identity check against FeatUp's released module. Every table is generated directly from the stored result files by a single script.

REFERENCES

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 (ECCV)*, volume 7577 of *Lecture Notes in Computer Science*, pp. 611–625. Springer, 2012. doi: 10.1007/978-3-642-33783-3_44.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640. IEEE, 2021. doi: 10.1109/iccv48922.2021.00951.

David Chanin, Tomáš Dulka, and Adrià Garriga-Alonso. Feature hedging: Correlated features break narrow sparse autoencoders, 2025. URL https://arxiv.org/abs/2505.11756.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=2dnO3LLiJ1.

Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3D awareness of visual foundation models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21795–21806. IEEE, 2024. doi: 10.1109/cvpr52733.2024.02059.

Ege Erdogan and Ana Lucic. Equivariant sparse autoencoders: Mechanistic interpretability of neural networks on symmetric data, 2025. URL https://arxiv.org/abs/2511.09432.

Stephanie Fu, Mark Hamilton, Laura Brandt, Axel Feldman, Zhoutong Zhang, and William T. Freeman. FeatUp: A model-agnostic framework for features at any resolution. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=GkJiNn2QDF.

Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=tcsZt9ZNKD.

Liv Gorton. Group crosscoders for mechanistic analysis of symmetry, 2024. URL https://arxiv.org/abs/2410.24184.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15979–15988. IEEE, 2022. doi: 10.1109/cvpr52688.2022.01553.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2733–2743. Association for Computational Linguistics, 2019. doi: 10.18653/v1/d19-1275.

Berthold K.P. Horn and Brian G. Schunck. Determining optical flow. *Artificial Intelligence*, 17 (1-3):185–203, August 1981. ISSN 0004-3702. doi: 10.1016/0004-3702(81)90024-2. URL http://dx.doi.org/10.1016/0004-3702(81)90024-2.

Haiwen Huang, Anpei Chen, Volodymyr Havrylov, Andreas Geiger, and Dan Zhang. LoftUp: Learning a coordinate-based feature upsampler for vision foundation models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9913–9923. IEEE, 2025. doi: 10.1109/iccv51701.2025.00924.

Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 991–999. IEEE, 2015. doi: 10.1109/cvpr.2015.7298701.

Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI)*, volume 2, pp. 674–679, 1981.

George Ma, Zhongyuan Liang, Irene Y. Chen, and Somayeh Sojoudi. Do sparse autoencoders identify reasoning features in language models? In *International Conference on Machine Learning (ICML)*, 2026. URL https://openreview.net/forum?id=TCFtA9CI3U.

Aleksandar Makelov, George Lange, and Neel Nanda. Towards principled evaluations of sparse autoencoders for interpretability and control. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=1Njl73JKjB.

Juhong Min, Jongmin Lee, Jean Ponce, and Minsu Cho. SPair-71k: A large-scale benchmark for semantic correspondence, 2019. URL https://arxiv.org/abs/1908.10543.

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 Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=a68SUt6zFt.

Tiago Pimentel, Josef Valvoda, Rowan Hall Maudslay, Ran Zmigrod, Adina Williams, and Ryan Cotterell. Information-theoretic probing for linguistic structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4609–4622. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.acl-main.420.

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

Mike Ranzinger, Greg Heinrich, Pavlo Molchanov, Bryan Catanzaro, and Andrew Tao. FeatSharp: Your vision model features, sharper. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, volume 267 of *Proceedings of Machine Learning Research*, pp. 51156–51182. PMLR, 2025.

Minseok Seo, Mark Hamilton, and Changick Kim. Upsample anything: A simple and hard to beat baseline for feature upsampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 29707–29716, 2026.

Jianbo Shi and Carlo Tomasi. Good features to track. In *1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 593–600. IEEE Comput. Soc. Press, 1994. doi: 10.1109/cvpr.1994.323794.

Yakov Pyotr Shkolnikov. Do foundation models know geometry? Probing frozen features for continuous physical measurement. *arXiv preprint arXiv:2603.06459*, 2026.

Andreas Steiner, Alexander Kolesnikov, Xiaohua Zhai, Ross Wightman, Jakob Uszkoreit, and Lucas Beyer. How to train your ViT? Data, augmentation, and regularization in vision transformers. *Transactions on Machine Learning Research*, 2022. URL https://openreview.net/forum?id=4nPswr1KcP.

Elena Voita and Ivan Titov. Information-theoretic probing with minimum description length. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 183–196. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.14.

Qin Wang, Kai Krajsek, and Hanno Scharr. Equivariant representation learning for augmentation-based self-supervised learning via image reconstruction, 2024. URL https://arxiv.org/abs/2412.03314.

Thomas Wimmer, Prune Truong, Marie-Julie Rakotosaona, Michael Oechsle, Federico Tombari, Bernt Schiele, and Jan Eric Lenssen. AnyUp: Universal feature upsampling. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=Y9UAgPehqo.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11941–11952. IEEE, October 2023. doi: 10.1109/iccv51070.2023.01100. URL http://dx.doi.org/10.1109/ICCV51070.2023.01100.

A Measurement hygiene

Why we never warp. Producing a sub-pixel-shifted pair by resampling an existing image leaks the shift. Bilinear interpolation at fractional offset a has Nyquist gain $|1 - 2a|$, which is symmetric about $a = 0.5$ and independent of direction: the *magnitude* of the fractional offset is therefore recoverable from translation-invariant statistics of a single warped image. Fitting a ridge probe from the radially-averaged log power spectrum to $|r|$ on bilinearly warped pairs gives $R^2 = 0.43$; on analytically rendered pairs the same probe gives -0.15, i.e. nothing. Resampling both members identically does not remove the leak (0.43 again) unless absolute phase is also randomised independently of the label (-0.04).

Synthetic scenes. Scenes are sums of band-limited sinusoids ($|f| \leq 0.30$ cycles/px, six components with random amplitude, orientation and phase) plus analytic signed-distance shapes composited with a fixed-width antialiasing kernel. Because the scene is evaluated in closed form at transformed coordinates, any translation, rotation or scale is exact and carries no offset-dependent filtering. The antialiasing width is fixed rather than derivative-dependent so that the filter cannot itself encode the offset.

Real pairs. Photographs cannot be re-rendered. We instead shift at high resolution, where the shift is an *integer* translation and no interpolation occurs, then box-downsample both members by the same integer factor, so every offset receives an identical filter and the displacement at the target resolution is exactly k/s pixels. The crop origin is drawn from a range that does not depend on k; making it depend on k leaks the label through content, which we measured at signed $R^2 = 0.055$ in an earlier version of the sampler. On CelebA-HQ we additionally randomise orientation with rot90 and flips, because the dataset's face alignment otherwise correlates content with absolute position (signed leak 0.052, reduced to -0.018).

B Implementation details

Probes are ridge regressions fit by accumulating sufficient statistics ($X^\top X$, $X^\top Y$) in float64, with the penalty selected on a held-out split and the intercept unpenalised. Interventions are applied to

patch tokens only, leaving prefix tokens untouched, and the modified hidden state is run through the remaining blocks; we verified the round trip against a full forward pass to floating-point exactness. MAE is evaluated with its mask ratio set to zero and a monotone noise vector so that its internal shuffle is the identity. Sparse autoencoders follow Gao et al. (2025): TopK activation, auxiliary loss on latents that have not fired in 10^7 tokens at weight $1/32$, tied encoder initialisation, and unit-norm decoder columns with the gradient projected orthogonal to them. Where they use a constant learning rate we use Adam with a cosine schedule, and the 2048-latent, 2500-step cells use a simpler trainer without the auxiliary loss, tied initialisation or gradient projection.

C CORRESPONDENCE PROTOCOL IN FULL

Reimplementations, and how they were checked. FeatUp's adaptive convolution is a compiled CUDA extension; our pure-PyTorch replacement is bit-identical to a naive triple loop, and our JBU level is bit-identical to `featup.upsamplers.JBULearnedRange` loaded with the same released weights. Upsample Anything is reimplemented from the published equations and hyperparameter table; we report it both at the published setting and at the setting that minimises its *own* test-time objective (guidance reconstruction L_1 of 0.0018 against bilinear's 0.0295 and the published setting's 0.0114). The self-tuned variant performs worse at localisation, which is why both are reported.

FeatUp run end to end. As an additional check that our usage does not disadvantage it, we ran FeatUp's complete released pipeline — its own DINO ViT-S/16 featurizer, its ChannelNorm, and the full stack including the final channel-mixing residual that cannot transfer to a wider backbone — with correspondence performed entirely inside its own feature space. It reaches $\mathrm{PCK@1} = 0.083$ on the translation protocol, against 0.108 for its transferred kernels on ViT-B/16, so our usage flatters it rather than the reverse. On Sintel it reaches 0.160 against 0.164 for its own unrefined patch-level baseline.

Statistics. All intervals are percentile cluster bootstraps that resample *image pairs*, not tokens. The roughly 144 retained tokens of one image share its content, and a token-level bootstrap reports intervals several times too narrow. Seeds move the data, the probe fit and any random control direction together, so the reported spread covers all of those sources rather than sampling noise alone.

The estimator contrasts in detail. The four arms of Table 4 are trained and evaluated in one run on the same 800 training crops and identical evaluation crops, so per-token errors are paired and each contrast is a bootstrapped difference; equivalence is declared only by two one-sided tests against a $\pm 0.01\,\mathrm{PCK@1}$ margin fixed in advance. Fitting the linear map by ridge rather than SGD changes $\mathrm{PCK@1}$ by $+0.0006$ $[+0.0002, +0.0010]$, inside that margin; the two fits reach held-out scalar R^2 of 0.313 (SGD) and 0.314 (ridge), and their directions agree to $\cos(w) = 0.991$–0.996 across the three seeds. The MLP arm contains the linear solution, so its deficit at this training-set size is evidence about sample efficiency and inductive bias, not about the representation requiring a linear decoder.

The correspondence ablation in numbers. On the translation protocol, removing the normal-flow direction at layer 6 takes our refinement from $\mathrm{PCK@1} = 0.114$ to 0.020 for MAE and 0.073 to 0.017 for DINO — the unrefined patch match scores 0.017 — while FeatUp reading the same edited features goes only $0.108 \to 0.101$ and $0.108 \to 0.106$, and bilinear upsampling $0.082 \to 0.079$.

D THE CROSSED-GEOMETRY CONTROL IN FULL

With a learned aggregator, a deliberately *rotated* intermediate target costs nothing (0.243 against 0.247). A fixed-rotation account — that the aggregator simply absorbs the rotation, so only target–aggregator consistency matters — is ruled out by the crossed-geometry control: crossing the two geometries (Table 13), the self-consistent cells do *not* return to the unrotated value, falling from 0.358 to 0.278, 0.191 and 0.166 at 30°, 60°, 90°, and at 30° and 60° the *unrotated* aggregator reads a rotated scalar better than the matched one. Among the targets we tested the aperture-aligned coordinate is uniquely the easiest to exploit — consistent with the aperture geometry rather than an

Table 3: **The intervals the main text refers to.** *Top:* raw $R^2_{v_n}$ at layers 0 and 12 on synthetic scenes as mean $\pm$ standard deviation over three seeds, then the layer-0-normalised ratio $\rho(12)$ with its bootstrap interval on each of the three image sources. Per-seed spread of the raw R^2 was stored only for the synthetic runs; the normalised ratio carries an interval on all three, so we report each where it exists rather than implying a uniform analysis. This is the uncertainty behind Table 1, which shows point estimates only. *Bottom:* PCK@1 with cluster-bootstrap intervals over image pairs for the MPI-Sintel comparison of Section 6.

	synthetic, raw $R^2_{v_n}$		$\rho(12)$ with 95% interval		
backbone	layer 0	layer 12	synthetic	CelebA-HQ	COCO
MAE	0.657 ± 0.009	0.631 ± 0.008	0.959 $[0.954, 0.965]$	1.047 $[1.043, 1.051]$	0.979 $[0.972, 0.985]$
DINO	0.592 ± 0.008	0.256 ± 0.018	0.432 $[0.417, 0.450]$	0.438 $[0.428, 0.449]$	0.155 $[0.147, 0.164]$
DINOv2	0.252 ± 0.005	0.060 ± 0.007	0.239 $[0.212, 0.268]$	0.395 $[0.378, 0.412]$	0.055 $[0.044, 0.067]$
SigLIP	0.488 ± 0.008	0.060 ± 0.003	0.124 $[0.111, 0.137]$	0.360 $[0.350, 0.369]$	0.122 $[0.114, 0.130]$
DINOv2+reg	0.550 ± 0.007	0.059 ± 0.006	0.107 $[0.097, 0.118]$	0.217 $[0.208, 0.226]$	0.029 $[0.023, 0.035]$
CLIP	0.155 ± 0.007	0.009 ± 0.001	0.057 $[0.044, 0.070]$	0.037 $[0.032, 0.041]$	-0.033 $[-0.042, -0.023]$
AugReg ViT	0.651 ± 0.008	0.004 ± 0.002	0.006 $[0.002, 0.010]$	0.030 $[0.024, 0.036]$	-0.026 $[-0.030, -0.021]$

backbone	refinement	PCK@1	95% CI
MAE	aperture-aware readout	0.353	$[0.330, 0.378]$
MAE	FeatUp	0.231	$[0.217, 0.245]$
MAE	Upsample Anything	0.120	$[0.110, 0.131]$
MAE	bilinear $\times 16$	0.134	$[0.123, 0.147]$
MAE	no refinement	0.162	$[0.145, 0.182]$
DINO	aperture-aware readout	0.258	$[0.237, 0.281]$
DINO	FeatUp	0.205	$[0.192, 0.219]$
DINO	Upsample Anything	0.109	$[0.100, 0.120]$
DINO	bilinear $\times 16$	0.156	$[0.144, 0.169]$
DINO	no refinement	0.162	$[0.144, 0.181]$

Table 4: **Matched contrasts, each changing one factor.** Four arms trained and evaluated in one run on identical crops, so per-token errors are paired; each contrast is the bootstrapped difference between two of them, clustered on image pairs. Equivalence is declared only by TOST against a ± 0.01 PCK@1 margin fixed in advance, never from overlapping marginal intervals. Marginal values and per-seed numbers are in Table 3 and Section C. MPI-Sintel, 3 seeds $\times$ 240 crops.

contrast (one factor changed, all else held)	ΔPCK@1 [95% CI]	verdict
A per-token map: MLP $\rightarrow$ linear	$+0.0412$ $[+0.0353, +0.0477]$	difference detected
B aggregator: learned $\rightarrow$ analytic	$+0.0301$ $[+0.0210, +0.0395]$	difference detected
C optimiser: SGD $\rightarrow$ ridge	$+0.0006$ $[+0.0002, +0.0010]$	equivalent (TOST)

arbitrary basis, though the cross establishes the empirical privilege, not the mechanism. The earlier null reflects a pipeline far below the achievable ceiling, where the target's quality does not bind.

Finally, a classical pixel-space Lucas–Kanade refinement from the identical coarse match reaches PCK@1 $= 0.398$ $[0.372, 0.428]$, matching or beating every feature-based variant including ours (0.358 $[0.335, 0.383]$; our median is better, 1.474 against 1.583 px). Frozen features do *not* provide superior sub-pixel evidence here, but more *robust* evidence: they trade peak precision for fewer gross failures (Figure 6).

We draw the conclusion this licenses and no more. The correspondence experiment shows that the classical local quantity remains, after twelve transformer blocks, in a form a *linear* readout can reach, well enough to do real sub-pixel work at close to classical accuracy — an *accessibility* result about the representation, not a method contribution. It also rules out the simplest supervision-only explanation, since a supervised decoder with the same labels and $190\times$ more input dimensions does substantially worse than the linear projection. (The readout appears at 0.353 in Table 2 and 0.358 here: the contrast run fits its probe on a larger split and reserves a held-out split for tuning the learned arms, so all arms share one protocol. The difference is within the interval.)

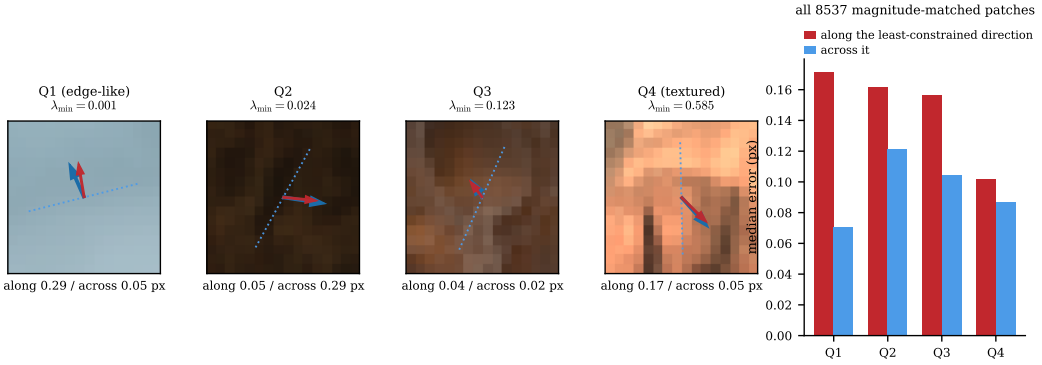

Figure 5: **The aperture signature is visible in *where* the error falls, not only in how large it is.** Real MPI-Sintel patches stratified by $\lambda_{\min}$ of the structure tensor; each example is the *median*-$\lambda_{\min}$ member of its quartile, so none is a best case. Blue arrow: true displacement. Red: the structure-tensor solve. Dotted: the direction the patch's own structure leaves least-constrained (the $\lambda_{\min}$ eigenvector). *Right:* over all 8537 patches, the median error resolved along that least-constrained direction against the error across it. For the edge-like quartile the error is 0.17 px along against 0.07 px across, a ratio of 2.44; for the textured quartile it is 0.10 against 0.09, a ratio of 1.17, and the total falls from 0.24 to 0.17 px. This is the aperture problem seen directly: an ill-conditioned patch is not uniformly worse, it is specifically uninformative about motion along its own edge. Two caveats. Displacement magnitude is restricted to 0.5–1.5 px, because raw endpoint error is not comparable across patches whose true displacement differs in size — without that matching the trend inverts, which is a confound rather than a finding. And individual patches are noisy: the Q2 example shown happens to err across rather than along. The claim is the aggregate, not any one patch.

Table 5: **A stimulus-side diagnostic: period recovery from one scalar projection, by checkpoint.** One grating is rendered at seven spatial frequencies and its offset swept over two periods; the scalar $h_\ell(\boldsymbol{x}) \cdot \hat{\boldsymbol{w}}$ is fitted with a sinusoid at the stimulus period, and the period is then recovered from that one number. No pairs are involved and the direction is never refit on this stimulus. Per-frequency values in Table 6.

backbone	pretraining objective	mean sinusoid R^2	median period error	periods within 5%
MAE	pixel reconstruction	**0.950**	**0.5%**	7/7
DINO	augmentation invariance	0.736	1.6%	6/7
AugReg ViT	label supervision	0.290	47.9%	3/7

E QUALITATIVE EXAMPLES

Every other figure in this paper is a diagram or a plot of aggregate numbers. These show the images themselves, with the scores attached. Crops come from batches drawn under a fixed seed on the held-out split, and where a single example is displayed the rule for choosing it is stated in the caption; nothing here is selected by score.

One point about these images. They are reproduced in colour because that is what the datasets contain and it makes the content legible, but *every measurement in this paper is taken on the luminance conversion*: probes see one channel, and the injectivity argument of Section 3.2 is stated on that 256-dimensional grayscale subspace. The colour is for the reader, not for the model. The synthetic renderer has no colour original and is shown as it is computed.

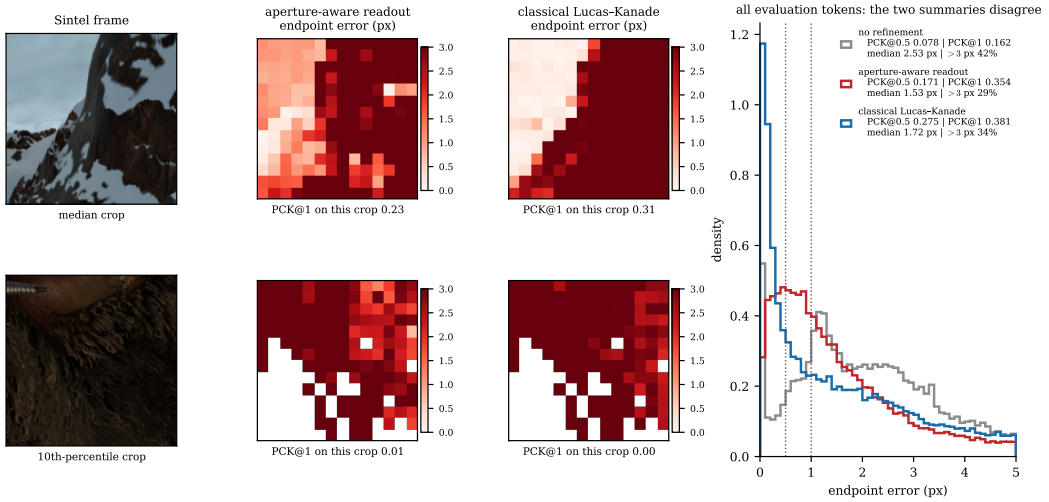

Figure 6: **Why the two summaries of the correspondence comparison disagree.** Table 4 reports that classical Lucas–Kanade beats our readout on PCK@1 while our *median* endpoint error is the better of the two, and the aggregate numbers do not say why. Per-token error maps on two crops — the one at the median per-crop PCK@1 and the one at the 10th percentile, a rule fixed before looking — and the error distribution over all evaluation tokens resolve it. Lucas–Kanade is markedly the more *precise* of the two where its linearisation holds: it places 27.5% of tokens within 0.5 px against our 17.1%, visible as the spike near zero. It is also the more *brittle*: 34% of its tokens land beyond 3 px against our 29%, and its median is worse (1.72 against 1.53 px). Reading a frozen feature is thus not a more accurate estimator than reading the pixels — it is a more robust one, trading peak precision for fewer gross failures. Both refinements improve substantially on the unrefined patch match (PCK@1 $= 0.162$, median 2.53 px). These are single-seed values on one evaluation draw, so they differ slightly from the three-seed numbers in Table 4.

Table 6: **Period recovery from one scalar projection.** Left: sweeping one grating's offset over two periods and reading the scalar $h_\ell(\boldsymbol{x}) \cdot \hat{\boldsymbol{w}}$ of frozen MAE at layer 12. The projection is sinusoidal *at the stimulus period*, which it recovers from that one number. Right: the same sweep on three backbones; period fidelity follows the accessibility ordering of Table 1, measured with a different stimulus and no paired inputs.

f (cyc/px)	period (px)	sinusoid R^2	linear R^2	period found
0.05	20.00	0.998	0.092	20.11
0.08	12.50	1.000	0.124	12.50
0.12	8.33	0.925	0.055	8.16
0.15	6.67	0.957	0.026	6.67
0.20	5.00	0.856	0.108	5.03
0.25	4.00	0.946	0.114	3.98
0.30	3.33	0.966	0.083	3.33

backbone	mean R^2	med. err	within 5%
MAE	0.950	0.5%	7/7
DINO	0.736	1.6%	6/7
AugReg ViT	0.290	47.9%	3/7

F Additional tables and figures

G What the depth effect is not

The measurement survives the obvious alternative explanations. It is not carried by outlier tokens: excluding the top 1% of tokens by norm (above 1.1–1.3× the median norm) changes retention by at most 0.003 in each of the four backbones tested (MAE, DINO and both DINOv2 variants). It is not a positional-interpolation artefact: DINOv2 evaluated at its native 518 pixels still decays by 4.96× and 2.97× at 0.5 and 2 px, against 3.84× and 3.15× at 224 pixels. And it is not specific to translation: a probe fit on translations reads rotated and scaled pairs at layer 12 as well as probes fit on them (MAE 0.627 and 0.626 against 0.618 and 0.610; DINO 0.251 and 0.202 against 0.246 and 0.198).

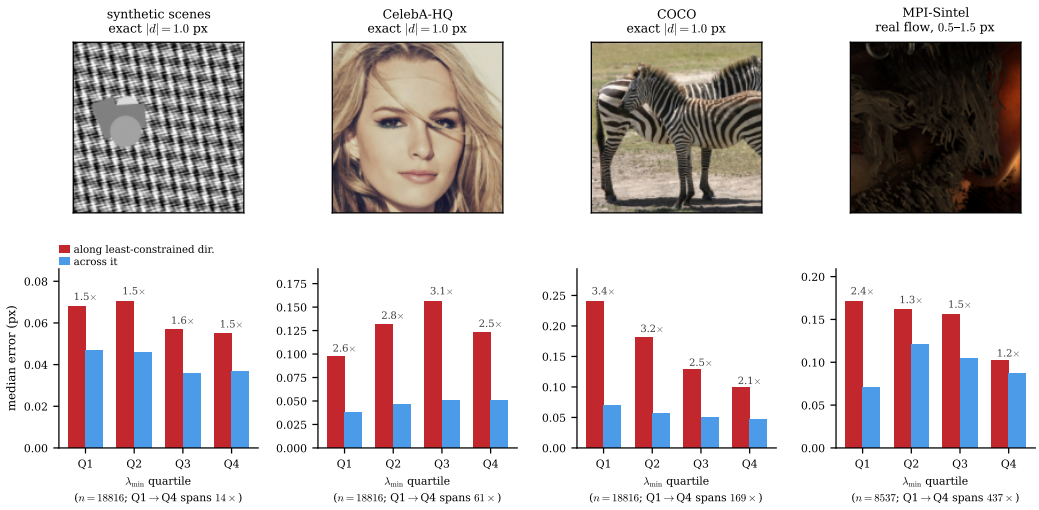

Figure 7: **The aperture signature replicates on four data sources; its dependence on conditioning does not.** *Top:* one example per source, the first drawn under a fixed seed. Synthetic scenes are rendered analytically; CelebA-HQ and COCO pairs are built by integer shifts at high resolution followed by identical box downsampling, so neither member is interpolated (Section A); MPI-Sintel supplies real per-pixel flow. The first three fix $|\boldsymbol{d}| = 1.0\,\mathrm{px}$ exactly so displacement magnitude cannot confound the comparison; Sintel is restricted to a 0.5–$1.5\,\mathrm{px}$ band for the same reason. *Bottom:* median structure-tensor error resolved along the direction the patch's own structure leaves least-constrained against the error across it, by $\lambda_{\min}$ quartile, with the ratio above each pair. **What replicates:** the error is larger along the least-constrained direction in every quartile of every source, by 1.2–3.4×. **What does not:** the anisotropy narrows with better conditioning on the two natural-image sources (COCO $3.4\times \to 2.1\times$, Sintel $2.4\times \to 1.2\times$) but is flat on the synthetic renderer ($1.5\times \to 1.5\times$) and on faces ($2.6\times \to 2.5\times$). One plausible account is the range of conditioning each source represents. Every source reaches $\lambda_{\min} \approx 0$, but the renderer's distribution sits far higher overall: the *upper* edge of its worst quartile, $\lambda_{\min} = 1.03$, exceeds the median of the best quartile of every real source here. Our data do not isolate this as the cause and we do not claim it. We note the limitation against ourselves: Table 1's synthetic column is measured on this renderer, whose patches substantially underrepresent the extreme ill-conditioning of natural images, which is why the paper reports three image domains rather than the rendered one alone.

Table 7: **A strong one-dimensional access path does not imply one-dimensional storage.** Normal-flow R^2 at layer 12. The 1-D MLP receives only $R \cdot \hat{\boldsymbol{w}}$; the complement MLP is refit *from scratch* after $\hat{\boldsymbol{w}}$ is projected out. For MAE the scalar matches the full nonlinear decoder and for DINO it keeps most of it, yet the orthogonal complement still supports about 70–73% of the full-residual MLP: $\hat{\boldsymbol{w}}$ is an efficient route to the variable, not the only place it is recoverable.

	linear, full	MLP, full	MLP, 1-D	linear, compl.	MLP, compl.	MLP compl. / full
MAE L12	0.629	0.713	**0.717**	0.459	0.518	73%
DINO L12	0.259	0.316	**0.277**	0.131	0.223	70%

H WHAT ORDERS DICTIONARY DISCOVERY ACROSS CELLS

Across the four backbone–layer cells, the target direction's raw variance share f does not order how well a dictionary aligns with it: DINO at layer 0 has the smallest share of the four, yet its best atom aligns better than MAE's at layer 12. The margin between the signal and what the dictionary still fails to explain, $f/(1 - R^2_{\mathrm{recon}})$, orders them better (Spearman $\rho = 0.80$). On four points with one inversion we offer that as a suggestion rather than a law; the point is that the diagnostic quantity is the error along the direction (Equation (3)) relative to the signal there, not the raw variance share.

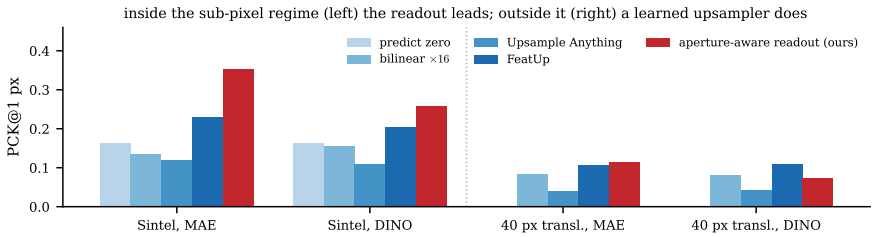

Figure 8: **The normal-flow readout helps in the regime its geometry predicts, not universally.** Left: MPI-Sintel, whose motion is predominantly below the patch stride; among the frozen-feature refinements shown the aperture-aware readout gives the strongest PCK@1. Right: translations extending to 40 px, where coarse matching errors dominate and a local residual correction no longer suffices; feature upsampling can then match or exceed it. The *predict zero* bar is shown on Sintel because 16% of tokens move less than one pixel.

Table 8: **One direction, and not because the probe was linear.** Left: normal-flow R^2 from the full residual, from the single supervised direction, from the best 4-D supervised subspace, and from a two-layer MLP given either all 768 dimensions or *only the one scalar* $R \cdot \hat{\boldsymbol{w}}$. Right: the same direction measured in the *single-image* activation — its variance share, its rank among the 768 principal components, and how much of it local contrast, gradient orientation and absolute position explain together. This is an independent run with 172,800 training tokens; Table 7 uses 576,000 and a different draw, which is why the MLP cells differ (MLPs are the most sample-hungry decoders here) while the linear cells agree to 0.02.

	normal-flow R^2 from the residual					$\hat{\boldsymbol{w}}$ in a single image		
cell	768-d ridge	1-D	4-D	MLP(768)	MLP(1-D)	var. share	PC rank	expl. by stats
MAE L12	0.648	0.648	0.648	0.679	**0.730**	0.66%	31/768	0.000
DINO L12	0.269	0.269	0.269	0.164	**0.289**	0.32%	45/768	0.002
MAE L6	0.658	0.658	0.658	0.699	**0.753**	1.43%	14/768	0.000

I DISCOVERY VERSUS DELIVERY, WORKED AT LAYER 0

Running the same grid at layer 0, where the residual is dominated by a deterministic positional term that an autoencoder reconstructs almost perfectly, separates two things that are easy to conflate.

At MAE layer 0 the dictionary *finds* the direction: its best decoder column reaches $|\cos| = 0.898$ with $\hat{\boldsymbol{w}}$, against 0.155 for the best of a random dictionary of the same size. The sparse code still delivers only 34% of what that direction delivers by ridge. At DINO layer 0 the autoencoder reconstructs 97.9% of the residual variance, and the normal-flow signal sits almost entirely in the 2.1% it throws away: $R^2 = 0.628$ from the discarded error against 0.116 from the reconstruction.

So the failure is not "dictionaries cannot represent geometry". Finding a direction and carrying it are different problems, and scaling addresses neither. (Raw variance share does not by itself order the four cells; see Section H.)

J WHY SPARSE DICTIONARIES MISS IT

The natural next move in an interpretability paper is to ask whether an unsupervised dictionary finds this direction on its own. It does not, and the reason is measurable — though not, as an earlier version of this work claimed, derivable from total reconstruction error.

A bound we withdraw. It is tempting to argue that preserving a direction carrying a fraction f of the residual variance requires $1 - R^2_{\text{recon}} < f$. That is not a necessary condition. A reconstruction

Table 9: **Scaling the dictionary does not close the gap.** A separate, larger scaling run from the one in Table 14: 1.15M training tokens against 0.58M, and a full width $\times$ steps grid. TopK sparse autoencoders on the aligned residual, trained following Gao et al. (2025) as described in Section B, with the auxiliary dead-latent loss enabled in every cell but 2048 $\times$ 2500; dictionary width $\times 8$ and training $\times 20$ relative to the smallest cell. "% sup" is the fraction of what *one* supervised direction achieves on the same input. "atom" is the best $|\cos|$ between any decoder column and $\hat{\boldsymbol{w}}$, against the best a random dictionary of the same size reaches. This run stores the code's flow R^2 rather than $R^2_{\hat{\boldsymbol{w}}}$, which is why the along-direction column appears only in Table 14.

cell	dict	steps	recon R^2	dead	codes R^2	% sup	atom $\cdot\hat{\boldsymbol{w}}$	random
MAE L12	2048	2500	0.645	0.00	0.273	42%	0.105	0.117
	2048	50000	0.672	0.00	0.304	47%	0.134	0.117
	8192	50000	0.684	0.00	0.323	50%	0.129	0.135
	16384	2500	0.674	0.00	0.286	44%	0.127	0.155
	16384	10000	0.693	0.00	0.319	49%	0.150	0.155
	16384	50000	0.691	0.00	0.319	49%	0.135	0.155
reference: one supervised direction = full 768-d ridge = 0.646; the direction holds $f = 0.0189\%$ of the residual variance								
DINO L12	2048	2500	0.494	0.00	0.121	44%	0.131	0.118
	2048	50000	0.537	0.00	0.167	60%	0.369	0.118
	8192	50000	0.561	0.00	0.177	64%	0.362	0.138
	16384	2500	0.553	0.00	0.152	55%	0.173	0.136
	16384	10000	0.577	0.00	0.172	62%	0.299	0.136
	16384	50000	0.576	0.00	0.177	64%	0.374	0.136
reference: one supervised direction = full 768-d ridge = 0.277; the direction holds $f = 0.0942\%$ of the residual variance								

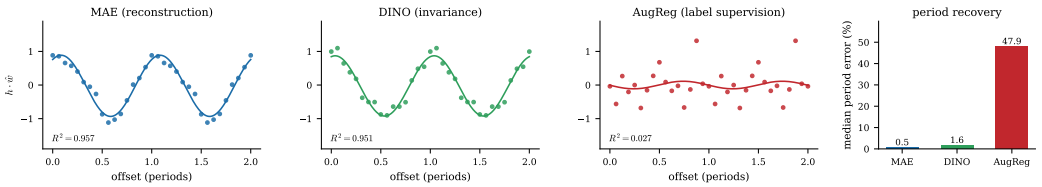

Figure 9: **The phase sweep at one frequency, three objectives.** A grating of period 6.67 px with its offset swept over two periods; each panel plots the single scalar $h_\ell(\boldsymbol{x}) \cdot \hat{\boldsymbol{w}}$ with its fitted sinusoid. The code is intact under reconstruction pretraining, degraded under augmentation invariance, and largely gone under label supervision. Right: the median error with which the stimulus period is recovered from that one number. Summary in Table 5.

returning the target component exactly while discarding everything orthogonal to it has $1-R^2 = 1-f$, which is enormous when f is small, and yet preserves the target perfectly. The inequality assumes the reconstruction error is isotropic, which is exactly what is at issue.

The quantity that is actually diagnostic. What matters is the error measured *along* the target direction,

$$R^2_{\hat{\boldsymbol{w}}} \;=\; 1 \;-\; \frac{\mathbb{E}\big[(\hat{\boldsymbol{w}} \cdot (R - \hat{R}))^2\big]}{\mathrm{Var}(\hat{\boldsymbol{w}} \cdot R)}, \tag{3}$$

which we measure directly rather than bound.

Table 14 gives it for TopK sparse autoencoders trained following Gao et al. (2025) as described in Section B, across dictionary widths 2048 to 16384 and 2500 to 50 000 steps on 0.58M tokens. The dictionaries are not capacity-starved: in the scaling run of Table 9, which records it, the dead-latent fraction is 0.00 in every cell. Their total reconstruction reaches $R^2 = 0.712$, i.e. they discard about 29% of the residual variance. But along $\hat{\boldsymbol{w}}$ they preserve only $R^2_{\hat{\boldsymbol{w}}} = 0.453$: **the error is concentrated in the direction that matters**, losing more than half the target component while losing under a third overall. Consistently, normal flow read from the reconstruction (0.277) is *worse* than normal flow read from what the autoencoder threw away (0.311).

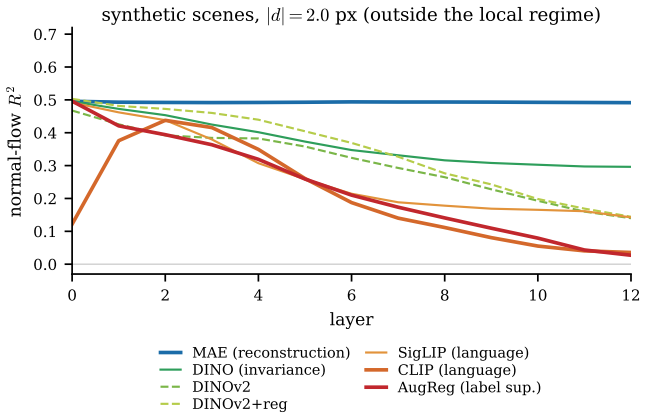

Figure 10: **Outside the local regime the layerwise trajectories change.** The same measurement as Figure 2 at $|\boldsymbol{d}| = 2.0\,\mathrm{px}$, beyond the half-period boundary of Section 3.4. We include it as a scope check rather than as evidence for any universal ordering: the quantity our local readout measures is regime dependent.

Table 10: **MAE's retention does not depend on the evaluation mask ratio.** Only tokens visible in both members of a pair are scored, so the comparison is like-for-like at every ratio. At MAE's own training value of 0.75 retention is still 0.921, against 0.432 for DINO and 0.006 for the supervised ViT.

evaluation mask ratio	0.00	0.25	0.50	0.75
$R^2_{v_n}(0)$	0.644	0.642	0.640	0.637
$R^2_{v_n}(12)$	0.620	0.616	0.608	0.586
retention	0.962	0.960	0.950	**0.921**

Scaling does not repair this within the range tested. Inside Table 14, width $\times 8$ and training $\times 20$ move $R^2_{\hat{\boldsymbol{w}}}$ from 0.387 to 0.453 and normal flow read from the reconstruction from 0.240 to 0.277, against 0.629 for one supervised direction on the same input. A separate grid at 1.15M tokens — twice the data, with the full width $\times$ steps sweep (Table 9) — reaches at best 50% of the supervised direction for MAE and 64% for DINO, and for MAE the best decoder atom's alignment with $\hat{\boldsymbol{w}}$ stays at or below what a random dictionary of the same size reaches in five of six cells. We state this as bounded: *under the tested TopK recipe and scaling range*, increasing width, data and training does not close the gap. It does not rule out other objectives, parameterisations or sparsity formulations.

A diagnostic that survives the withdrawn bound. The practical lesson is not a threshold but a measurement anyone can make before committing to dictionary learning: fit one supervised direction for the variable of interest and report $R^2_{\hat{\boldsymbol{w}}}$ of the candidate reconstruction along it, rather than its total R^2. In every cell we examined the two diverge sharply, and only the former predicts whether the code carries the variable.

K Periodic response under translation is largely generic

Because displacement enters through local phase, $h_\ell(\boldsymbol{x}) \cdot \hat{\boldsymbol{w}}$ oscillates as a grating is translated (Figure 9), and recovers the grating's period from that one scalar to 0.5% median error across a sixfold frequency band. This is a satisfying check but a weak claim, and the controls say why: random directions in the *same* network recover the period to 2.1%, and a random direction in an **untrained** ViT recovers it to 1.6%. Any filter bank with nonzero Fourier sensitivity oscillates under translation. We therefore report the periodic response as consistent with the mechanism, not as evidence for a dedicated phase code.

Table 11: **Removing one direction of 768 at layer 6 collapses the layer-12 readout; matched controls do not.** Three seeds move the data, the probe fit and the control direction together; intervals are a cluster bootstrap over image pairs. The per-seed column matters for the control rows, where the seed also redraws the control *direction*. The overlap-matched control is the RMS-matched random direction rescaled by 44.1 (MAE) until its first-order downstream effect equals that of $\hat{\boldsymbol{w}}$. Real image pairs.

condition	flow R^2 [95% CI]	per seed	orient. R^2	contrast R^2	agg. err (px)
MAE					
clean	0.610 [0.606, 0.614]	0.607, 0.611, 0.612	0.733	0.763	0.080
remove the normal-flow direction (1 of 768)	-0.026 [-0.028, -0.023]	-0.026, -0.026, -0.025	0.531	0.692	0.544
random direction, RMS-matched	0.520 [0.513, 0.527]	0.489, 0.461, 0.609	0.689	0.713	0.083
random direction, overlap-matched	0.193 [0.174, 0.209]	0.053, 0.072, 0.451	-1.516	-0.815	0.098
DINO					
clean	0.226 [0.219, 0.233]	0.228, 0.221, 0.229	0.648	0.737	0.365
remove the normal-flow direction (1 of 768)	-0.004 [-0.009, -0.000]	-0.003, -0.013, 0.003	0.646	0.734	0.522
random direction, RMS-matched	0.226 [0.219, 0.233]	0.227, 0.222, 0.228	0.640	0.734	0.367
random direction, overlap-matched	0.184 [0.177, 0.190]	0.181, 0.201, 0.169	0.268	0.653	0.372

Table 12: **What produces the advantage, and what does not.** All rows share labels, pairs, coarse match and evaluation; the learned rows also share the 3×3 context and the same gradients, tuned on a held-out split. Consecutive rows change one factor at a time, so each contrast is identified; Section 6 reads them off. MPI-Sintel, 3 seeds $\times$ 240 crops.

refinement	dims	PCK@1 [95% CI]	median px
no refinement	—	0.162 [0.145, 0.182]	2.432
MLP, 3×3 residual + gradients	6939	0.194 [0.176, 0.213]	2.230
the same, with depth and lr also tuned	6939	0.191 [0.173, 0.210]	2.260
learned scalar bottleneck, end-to-end	36	0.226 [0.207, 0.245]	2.025
supervised on a *rotated* target	36	0.243 [0.226, 0.260]	1.880
supervised on normal flow	36	0.247 [0.229, 0.263]	1.870
linear scalar + *learned* aggregation	36	0.336 [0.317, 0.356]	1.465
linear scalar + *analytic* aggregation, 2-D loss only	36	0.303 [0.283, 0.325]	1.650
the same, normal-flow supervised, fit by SGD	36	0.357 [0.334, 0.383]	1.476
the same, normal-flow supervised, fit by ridge	36	**0.358** [0.335, 0.383]	1.474
classical pixel-space Lucas–Kanade	—	**0.398** [0.372, 0.428]	1.583

L THE CONTROLLED OBJECTIVE MINIATURE

Seven public checkpoints differ in data, schedule and augmentation as well as objective, so the cross-model statement is associational. To test the objective directly we trained ViT-S/16 models in which data, architecture, schedule and resolution are identical and only the objective varies, with a random-initialised network as a reference. At ImageNet-100 scale (130k images, seven runs) retention orders as reconstruction-strong 1.08, reconstruction-mild 0.91, supervised-mild 0.70, invariance 0.66, supervised-strong 0.43, against 0.97 for random initialisation. The ratios have different denominators — reconstruction starts lower, $R^2_{v_n}(0) = 0.30$–0.40 against 0.56–0.60 — but the absolute drop from layer 0 to 12 gives the same order, from a gain of 0.02 for strong reconstruction to a loss of 0.32 for strong supervision.

Controlled manipulation of the objective therefore changes the *rate* at which linear accessibility is lost, and reconstruction protects it most. It does *not* reproduce the public checkpoints' absolute levels — strong supervision keeps 0.43 of its layer-0 value here, AugReg 0.006 — and closing that gap needs ImageNet-1k-scale pretraining, one to two orders of magnitude beyond this study's budget. We therefore keep associational wording for Table 1 and claim intervention only for the rate.

Table 13: **The aperture geometry is privileged, not merely self-consistent.** Crossing the geometry the per-token scalar is supervised on ($\boldsymbol{d} \cdot R_\theta \hat{\boldsymbol{g}}$) with the geometry the analytic aggregator assumes; everything else identical. Were a rotated coordinate a change of basis a matched aggregator undoes, the self-consistent column would sit at the unrotated value. Section 6 reads the result. Last column: paired cluster bootstrap. MPI-Sintel, 3 seeds $\times$ 240 crops.

	PCK@1, (scalar target, aggregator)				paired Δ
θ	$(0,0)$	$(0,\theta)$	$(\theta,0)$	(θ,θ)	$(\theta,\theta)-(0,0)$
30°	**0.358**	0.286	0.334	0.278	-0.080 $[-0.086, -0.074]$
60°	**0.358**	0.190	0.259	0.191	-0.167 $[-0.180, -0.156]$
90°	**0.358**	0.136	0.168	0.166	-0.192 $[-0.205, -0.179]$

Table 14: **The autoencoder's error is concentrated in the direction that matters.** TopK sparse autoencoders trained following Gao et al. (2025) as described in Section B, on 0.58M tokens (the larger 1.15M-token scaling grid, which records a dead-latent fraction of 0.00 in every cell, is Table 9). They discard 29–50% of the residual variance overall but 55–62% of it *along* $\hat{\boldsymbol{w}}$ (Equation (3)); for MAE, normal flow read from the reconstruction is worse than normal flow read from what they threw away.

cell	dict	steps	total recon R^2	$R^2_{\hat{\boldsymbol{w}}}$ (along $\hat{\boldsymbol{w}}$)	flow from recon	flow from error
MAE L12	2048	2500	0.647	**+0.387**	0.240	0.372
	2048	50000	0.675	**+0.442**	0.279	0.328
	16384	50000	0.712	**+0.453**	0.277	0.311
reference: one supervised direction = 0.629 on the same input						
DINO L12	2048	2500	0.502	**+0.376**	0.105	0.143
	2048	50000	0.540	**+0.414**	0.139	0.117
	16384	50000	0.607	**+0.413**	0.131	0.109
reference: one supervised direction = 0.259 on the same input						

Table 15: **A signed dose-response, which deletion cannot give.** Injecting $\alpha\,\sigma\,\hat{\boldsymbol{w}}$ into MAE at layer 6, with σ the standard deviation of $h_\ell(\boldsymbol{x}) \cdot \hat{\boldsymbol{w}}$, and reading the layer-12 normal-flow probe: the readout moves linearly and with the sign of the injection. Random directions, each injected in units of its own σ, do essentially nothing, so the effect is specific to $\hat{\boldsymbol{w}}$ and not to perturbing the stream.

α (σ units)	-2.00	-1.00	-0.50	-0.25	$+0.00$	$+0.25$	$+0.50$	$+1.00$	$+2.00$	slope (px/σ)	lin. R^2
readout shift (px)	-11.58	-5.79	-2.89	-1.45	$+0.00$	$+1.45$	$+2.89$	$+5.79$	$+11.58$	$\mathbf{+5.789}$	1.0000
random dir., own σ (seed 0)					—					-0.063	1.0000
random dir., own σ (seed 1)					—					$+0.023$	0.9998
random dir., own σ (seed 2)					—					-0.026	1.0000
the supervised direction is $\mathbf{154}\times$ more effective per σ and $\mathbf{44}\times$ per unit of injected norm											

Table 16: **Localising where direct access to 2-D sub-patch displacement changes.** Each row tests one candidate bottleneck with a matched control. Well-conditioned image patches identify 2-D displacement; the patch projection preserves the relevant image evidence; and the same nonlinear decoder recovers displacement equally well from raw-pixel and layer-0 token *pairs*. The sharp transition occurs when either pair is collapsed into an aligned *difference*: direct 2-D decoding then fails in pixel and token space alike, so this direct-readout failure is not introduced by transformer processing. It does not imply that the difference carries no displacement information: a geometry-aware estimator recovers 2-D once the missing reference-image gradients are supplied.

stage tested	decisive control	result	conclusion
image evidence	structure-tensor solve linear probe on the same pixels	$R^2 = 0.961$–0.982 in Q2–Q4 $R^2 \approx 0$	2-D is identifiable when structure is rich probe failure $\neq$ absence of information
patch embedding	rank and pseudo-inverse of W	rank $256/256$; W^+ error 3.1×10^{-6}	image evidence is preserved on our inputs
pair representation	same MLP on raw / token pairs	pixels 0.874; tokens 0.875	layer-0 tokens expose what the raw pair exposes
pair $\rightarrow$ difference	same MLP on raw / token differences	pixels -0.394; tokens -0.434	**direct 2-D decoding fails in either space**

Table 17: **The pair-to-difference transition is matched in pixel and token space.** The patch projection is injective on our grayscale inputs (rank 256/256, condition number 110), so the layer-0 token difference carries the same information as the pixel difference, and every input-matched decoder behaves accordingly. A classical structure-tensor solve succeeds on both *when it is also given the reference-image gradients*, which the difference alone does not contain; a generic MLP fails on both differences; and the same MLP given the *pair* instead succeeds on both, at layer 0 as on raw pixels. The transition therefore appears before the transformer, and it concerns what the difference makes directly accessible, not whether displacement information survives. The structure-tensor rows leave out the most ambiguous $\lambda_{\min}$ quartile; per-quartile results in Table 18.

decoder and input	2-D R^2	median err (px)
structure-tensor solve, raw pixel difference	0.944	0.096
structure-tensor solve, W^+(layer-0 token difference)	**0.944**	0.096
MLP, raw pixel *difference*	-0.394	—
MLP, layer-0 token *difference*	-0.434	—
MLP, raw patch *pair* $[A; B]$	**+0.874**	—
MLP, layer-0 token *pair* $[h(A); h(B)]$	**+0.875**	—

Table 18: **2-D recovery from raw patch pairs, by conditioning.** Stratified by $\lambda_{\min}$ of the patch structure tensor. Only the edge-like quartile is genuinely ambiguous; a linear probe scores zero in all four.

quartile	$\lambda_{\min}$ range	structure-tensor R^2	median err (px)	MLP R^2	linear R^2
Q1	0–0.73	-46.508	0.277	0.536	-0.000
Q2	0.73–1.7	**0.961**	0.064	0.984	0.000
Q3	1.7–3.1	**0.976**	0.053	0.991	-0.000
Q4	3.1–12	**0.982**	0.049	0.992	-0.000