How Vision Transformers Represent Position: A Mechanistic Comparison of Absolute and Rotary Positional Encodings

Anonymous authors
Paper under double-blind review

Abstract

Positional embeddings encode the spatial position of each image patch in a Vision Transformer (ViT), allowing the model to distinguish different spatial arrangements of visual content. However, we know little about how the choice of embedding shapes a vision transformer's internal representation. In this work, we show that Absolute Positional Embeddings (APE) and Rotary Positional Embeddings (RoPE) encode spatial position through different internal mechanisms: APE stores position in explicit features, while RoPE builds it inside attention and rebuilds it when it is withheld. Specifically, in APE models, sparse autoencoders recover causal row- and column-specific features, whereas interventions on RoPE reveal distinct rotary components that separately encode row and column information. Additionally, causal interventions selectively disabling RoPE's row or column rotations show that decodable row and column information can be independently removed and, once rotations are restored, partly reconstructed, in part by axis-specialized attention heads. Both mechanisms are causally important for classification: permuting APE's positional features across patches reduces top-1 accuracy by 24–31 points, and ablating the rotations of RoPE's primary rebuilding head reduces it more than ablating those of any other head. Our findings show that the choice of positional encoding fundamentally changes how spatial information is organized and processed inside Vision Transformers. More broadly, our work demonstrates how mechanistic analysis can reveal how models construct, reconstruct, and maintain structured representations, providing a basis for targeted interventions in vision models.

1 Introduction

Vision Transformers (ViTs) are now the standard backbone for image understanding, from self-supervised encoders (Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2025) to segmentation models (Kirillov et al., 2023) and the image encoders of vision–language models (Radford et al., 2021). Many of these applications depend on where objects are in an image, yet because self-attention is invariant to the order of its inputs, a ViT knows a patch's location only through its positional encoding (Vaswani et al., 2017; Dosovitskiy et al., 2021). Recent models have moved from learned absolute positional embeddings (APE) (Dosovitskiy et al., 2021) to rotary positional embeddings (RoPE) (Su et al., 2024; Fang et al., 2023; Heo et al., 2024; Wang et al., 2024; Siméoni et al., 2025), a choice made on the basis of accuracy and resolution extrapolation (Heo et al., 2024; Fuller et al., 2024). However, spatial reasoning remains a known bottleneck: vision–language models built on ViT encoders struggle with basic spatial relations, a gap recent work calls their spatial blindspot (Alam et al., 2026), and they make little use of spatial cues even when positional encodings are in place (Qi et al., 2025). The fixes proposed so far change the encoding itself and are judged only by benchmark scores, without a mechanistic account of how APE and RoPE represent position in ViTs, where that information is held, and how the model uses it.

At a technical level, the two encodings differ in how they provide position: APE adds a learned vector to each patch once, before the first block (Dosovitskiy et al., 2021), whereas RoPE leaves the patches unchanged and rotates queries and keys at every layer by angles set by each patch's row and column (Su et al., 2024; Heo et al., 2024). Most evidence about position inside trained ViTs comes

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.

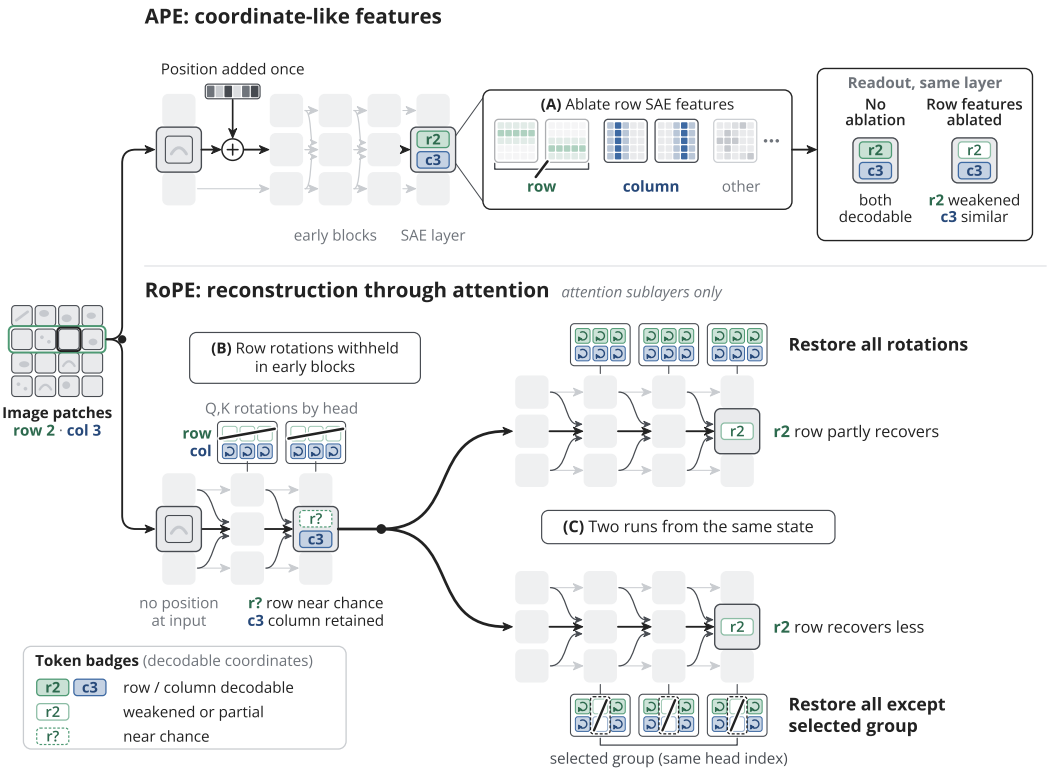

Figure 1: APE stores position in coordinate-like residual-stream features, whereas RoPE ViTs rebuild position through attention. Top: APE adds position once at the input, so at the SAE layer a patch token carries decodable row and column coordinates, and (A) ablating the row-selective SAE features weakens the row readout while the column readout stays similar. Bottom: RoPE adds no position at the input and instead rotates queries and keys, per head and separately for rows and columns, in every attention layer. (B) With the row rotations withheld in early blocks, the row coordinate is near chance while the column coordinate is retained. (C) Two runs continue from this state: restoring all rotations lets the row coordinate partly recover, and restoring all except those of a selected head group (the same head index in each block) lets it recover less.

from linear probes, which show that position can be decoded from a hidden state but not where it came from or whether the model uses it (Alain & Bengio, 2017; Belinkov, 2022). Mannes (2026) introduced a more direct measure, the spatial similarity–distance correlation (SSDC), which asks whether patches that are close on the image grid have similar hidden states; under random permutation at inference (RPI), which shuffles patch content while position stays attached to each token, it measures structure due to position alone. That work characterized this structure in small ViTs trained from scratch, but did not identify the components that produce it or test whether classification depends on it.

Across four APE and four RoPE ViTs, we show that the two encodings lead the network to represent position through different mechanisms. In APE models, position is stored in the residual stream: a sparse autoencoder (SAE) finds that it is driven by a small set of features, about 20 of 3,840, each tied to a band of rows or columns, which together form a coordinate system over the image grid. Downstream classification depends on these features: permuting them across patches, with the image unchanged, reduces ImageNet top-1 accuracy by 24–31 points. RoPE models contain no such features; position is instead computed inside attention by rotating queries and keys, with separate rotations for rows and columns, so that replacing the row rotations with the identity removes row position while leaving column position intact, and vice versa. Because these rotations are applied again at every block, the model recomputes position throughout the network: when they are withheld from the early blocks and then restored, later blocks rebuild 50–71% of the lost spatial structure (Figure 1).

These results mechanistically characterize APE and RoPE in vision models, and illustrate how position is represented inside vision transformers and how to intervene on it. For interpretability re-

search, position in an APE model can be read and edited through a few SAE features, whereas in a RoPE model the SAE is unreliable and the rotations themselves are the natural point of intervention; secondarily, for work on spatial reasoning, replacing one encoding with the other changes how position is represented throughout the network, a change that can be measured directly with the tools we use here rather than inferred from downstream benchmark scores.

Our contributions are:

- **Mechanistic evidence for APE.** In four pretrained APE ViTs, a sparse autoencoder isolates a small set of row- and column-selective features that tile the image grid, and removing one axis's features removes only that axis.
- **A causal decomposition of RoPE.** Replacing RoPE's rotations with the identity removes position exactly and reversibly, by axis, block or head, and shows that each axis is carried by its own rotations.
- **Evidence for positional rebuilding.** When rotations are withheld from the early blocks and then restored, later blocks rebuild much of the lost spatial structure, partly through attention heads that each restore mainly one axis.
- **Functional evidence.** Both mechanisms matter for ImageNet classification: moving APE's positional features between patches, or ablating the rotations of RoPE's main rebuilding head, degrades accuracy more than matched controls.

2 Related work

Positional encodings. RoPE was introduced for language models (Su et al., 2024) and adapted to images by Fang et al. (2023) and Heo et al. (2024), who found that it matches APE's accuracy at the training resolution and extrapolates better beyond it; three of our checkpoints come from their release. Other work compares encodings by their effect on extrapolation in ViTs (Fuller et al., 2024) and length generalisation in language models (Kazemnejad et al., 2023; Chi et al., 2023), or asks how language models learn position without one (Haviv et al., 2022). The closest mechanistic study, Barbero et al. (2025), analyses how Gemma uses the frequencies of its one-dimensional RoPE and tests changes by retraining. We instead compare two encodings side by side in trained ViTs, and edit the row and column rotations directly at inference, with no retraining.

Spatial structure in ViTs. Dosovitskiy et al. (2021) showed that the learned position embeddings of APE ViTs form a row-and-column pattern, and Mannes (2026) used SSDC and RPI to track spatial structure in ViT-S models trained from scratch. Both describe the structure without considering which components produce it or how the model uses it; we answer both questions using pretrained ViTs. Lastly, Darcet et al. (2024) found artefact tokens in ViTs that added register tokens remove; we test whether such prefix tokens carry position in RoPE models, and find that the image border does instead.

Probes, sparse autoencoders and interventions. Since probes show what can be decoded from a hidden state rather than what the model uses (Alain & Bengio, 2017; Belinkov, 2022), we identify positional features with sparse autoencoders, which were developed for language models (Cunningham et al., 2024; Bricken et al., 2023; Templeton et al., 2024) and have recently been applied to vision models (Lim et al., 2025; Fel et al., 2025), and test them with interventions. Causal analyses usually ablate or patch whole heads, layers or residual-stream states (Vig et al., 2020; Elhage et al., 2021; Meng et al., 2022; Wang et al., 2023); for RoPE we edit only the rotations, which removes position while leaving the rest of each head's computation intact.

3 Methods

3.1 Models

We study four APE-only ViT-B checkpoints (supervised (Dosovitskiy et al., 2021), DINO (Caron et al., 2021), AugReg (Steiner et al., 2022) and SAM (Chen et al., 2022)), four RoPE-only checkpoints from the `naver_in1k` (Heo et al., 2024), `sbb_in1k` (Wightman, 2019) and DINOv3 (Siméoni et al., 2025) recipes at ViT-B and ViT-S scale, and one hybrid ViT-B that uses both (Heo

et al., 2024). DeiT-III ViT-B (Touvron et al., 2022), an APE model trained with the `naver_in1k` recipe, is used for one comparison. Supervised ViT-B and naver ViT-B are the primary models; Table 1 in Appendix A lists every checkpoint.

3.2 Measuring spatial structure

SSDC. If a model represents where each patch sits, patches that are close on the image grid should have similar hidden states. The spatial similarity–distance correlation (SSDC) of Mannes (2026) measures this: for every pair of patch tokens we take the cosine similarity of their hidden states and compute its Spearman rank correlation with how close the two patches are on the grid (negative Manhattan distance), averaged over images. SSDC is near 1 when nearer patches are consistently more similar and near 0 when similarity ignores grid distance. We measure it on the hidden state entering each of the 12 blocks, read after the block's first LayerNorm, with blocks indexed from 0.

Probes. On the same hidden states we train linear probes to predict each patch's exact position (196 classes, chance 0.5%) or its row and column (14 classes each, chance 7.1%), and, for APE, ridge regressions on the row and column coordinates, reported as R^2. Since a probe shows only that position can be read out, we pair every probe result with an intervention; training details are in Appendix I.

Controlling for image content. Neighbouring patches often look alike, so SSDC would stay above zero even in a model with no positional information. We therefore compute SSDC and probes under random permutation at inference (RPI), which shuffles the patch tokens after the patch embedding while the positional signal stays attached to its place in the sequence, so any spatial structure that remains comes from position rather than from image content. Removing the positional signal as well gives a floor. For RoPE models, where we compare how much structure returns across windows and checkpoints, we rescale SSDC and probe accuracy so that the floor is 0 and the intact model is 1:

$$\text{normalised SSDC}_b = \frac{\text{observed}_b - \text{floor}_b}{\text{baseline}_b - \text{floor}_b}.$$

APE results are unnormalized, since each APE intervention is compared with controls in the same model.

Table 3 in Appendix B lists every experiment, and Appendix I gives compute, sample sizes and implementation checks.

4 Results

4.1 Localization: depth profile

When spatial structure appears. SSDC under RPI, measured at every block of the intact models, shows that the two encodings build spatial structure at different depths (Figure 2), and this sets where the later sections intervene. APE adds position at the input, so its structure is present from block 0 and, in supervised ViT-B, reaches 93% of its peak by block 2. RoPE adds nothing at the input, so SSDC starts at the floor and rises until every RoPE model has 68–100% of its peak by block 4. We therefore train the APE SAEs on the residual stream entering block 2, where the structure has formed but ten blocks remain in which to see the effect of an ablation, and read the RoPE interventions at block 4, the first block at which all four RoPE models have most of their structure. We chose block 2 on supervised ViT-B, and the other APE models are still building their structure there (54–90% of their peak), but it is early enough for them too: zeroing their positional features at block 2 leaves spatial structure low, with no recovery at any later block, in all four models (Appendix D.4). The difference in depth illustrates the contrasting mechanisms: APE's spatial structure is in place almost from the input, whereas RoPE's is assembled by attention over the first few blocks, a split that activation patching confirms at the level of individual components (Appendix C).

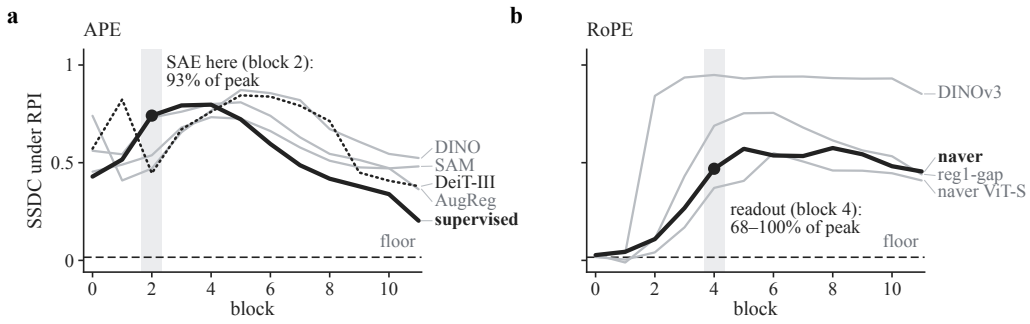

Figure 2: APE has spatial structure from the input; RoPE builds it over the early blocks. SSDC under RPI of the residual stream entering each block, for the intact models. Thick lines are the primary models (supervised ViT-B, naver ViT-B) and thin lines the other checkpoints; dashed is the floor with all positional signal removed; dotted is DeiT-III ViT-B (APE), trained with the same recipe as naver ViT-B. Shading marks where later sections act: the SAE at block 2 for APE (Section 4.2) and the block-4 readout for RoPE (Section 4.3).

4.2 APE: SPARSE AUTOENCODER FEATURES

In each APE model we train an SAE on the residual stream entering block 2 (Section 4.1) and, for each of the 196 grid positions, keep the most row-selective and the most column-selective feature. Row selectivity is the mean activation of a feature's most active row divided by its mean over the whole grid (1 is uniform, above about 6 is concentrated), and likewise for columns; details are in Appendix D.2. We then zero these features and compare against the SAE reconstruction and a random set of the same size.

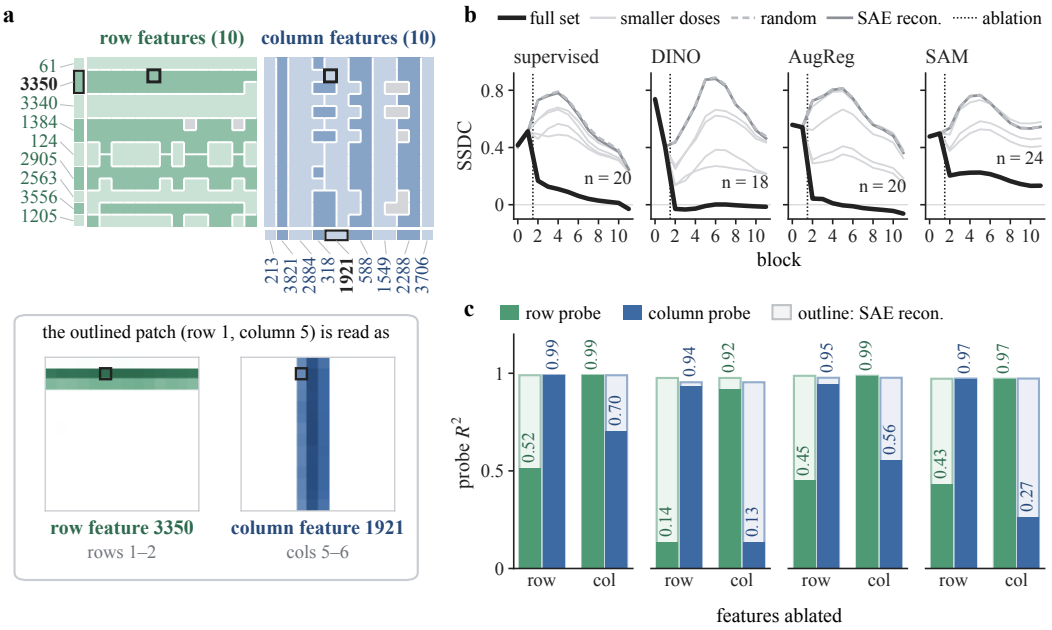

Figure 3: APE models store position in a small set of coordinate-like SAE features. We train an SAE on the residual stream entering block 2 and, for each grid position, keep its most row-selective and most column-selective feature. **(a)** Supervised ViT-B: each patch coloured by the row feature (green) and column feature (blue) that it is assigned. Nine of each tile the grid in ordered bands; grey cells belong to the tenth, which wins only scattered cells. The outlined patch (row 1, column 5) is read by row feature 3350 and column feature 1921, whose mean activation maps are shown below. **(b)** SSDC by block when the features are zeroed at block 2 (dotted line): the full set of n features (black), smaller sets, a random set of the same size and the SAE reconstruction with nothing zeroed. **(c)** Row and column probe R^2 after zeroing only the row or only the column features; outlines show the SAE reconstruction. Probes fit to shuffled labels read $R^2 \approx 0$.

The features tile the grid. The screen finds 20 such features in supervised ViT-B and 18–24 in the other models. In supervised ViT-B, nine of the ten row features and nine of the ten column features each cover a band of one or two adjacent rows or columns, and together they tile the grid in order (Figure 3a). The SAE has found something that looks close to a coordinate system, not a scatter of partly positional features.

Removing them removes the structure. Zeroing the full set drives SSDC to 0.23 or below from block 2 onward in all four models, while zeroing a random set leaves SSDC close to the SAE reconstruction baseline (Figure 3b), even when the random features are matched to the positional ones on firing rate and activation size (Appendix D.5). A position probe shows the same: in supervised ViT-B, it recovers a patch's exact position 22.9% of the time once the features are zeroed, against 99.7% intact and 99.1% after zeroing a random set (Appendix D.3). The SAE does not capture all of the position, since adding its reconstruction error back restores the probe to 57–93% while SSDC at block 2 stays low (0.07–0.25; Section 5). Removing fewer features removes less, and most of the drop in SSDC comes with the last few, as the bands in (a) predict: SSDC falls to its floor only once every band is gone. The spatial structure therefore depends on these particular features, not on removing features in general.

Rows and columns are stored separately. In all four models, zeroing only the row features sharply lowers a ridge-regression probe for the row coordinate while the column probe stays at $R^2 \geq 0.92$, and zeroing the column features does the reverse (Figure 3c; Appendix D.6). The targeted probe does not reach zero (row R^2 0.51 and column 0.71 in supervised ViT-B), so some position lies outside the features we screened (Section 5), but what the screen does find splits cleanly by axis. APE therefore stores a patch's row and its column in separate features, and either one can be removed while the other stays intact.

4.3 RoPE: ROTATIONS AND REBUILDING HEADS

RoPE adds no positional vector to the residual stream; instead, every block rotates each head's queries and keys by angles set by the patch's row and column, so position reaches the residual stream only through attention. An SAE is a poor tool here, since reconstruction alone shifts SSDC by up to $+0.214$ on naver ViT-B (Appendix D.7), but the rotations themselves can be edited exactly. Each ViT-B head has 32 rotation planes in its queries and keys, half turned by the row and half by the column, and setting a plane's rotation to the identity removes it reversibly, for one axis, a range of blocks or a single head (Appendix I).

Each axis has its own rotations. Rotations are the only source of position in these models, and each axis has its own planes. With every plane removed, SSDC falls to 0.011–0.018 and both probes to chance on all four RoPE models. Removing only the row planes sends the row probe to chance and leaves the column probe within 7 points of intact (the gap is largest on DINOv3 and naver ViT-S), removing the column planes does the reverse, and 16 random planes send neither axis to chance (Figure 4a; Appendix E). As in APE, position splits into row and column, but here the split lies in the operation that creates position rather than in features of the residual stream. Rotations encode only offsets between patches, so absolute position must be anchored somewhere. Row and column probes at block 4 are 95–100% accurate on patches at the image border but 14–56% in the interior, and edge patches stay at 72–91% when attention to the class and register tokens is blocked. The prefix tokens do add to the early structure, most of all on DINOv3, whose registers carry four-fifths of its block-4 SSDC, but the absolute position that RoPE models read comes from the edge of the image (Appendix E.2).

Later blocks rebuild position. Because every block applies rotations, later blocks can rebuild position when the early blocks go without it. With all rotations off in blocks 0–5 and restored afterwards, normalised SSDC at block 11 recovers to 0.50–0.71 on all four models, and less comes back as the window grows (Figure 4b). On naver ViT-B, the last six blocks contribute more spatial structure than the first six: keeping rotations only in the last six gives a normalised SSDC of 0.71, against 0.48 when they are kept only in the first six. DINOv3 is the opposite, building early and keeping what it builds (0.85 from the first six blocks against 0.50 from the last six). Removing one axis in blocks 0–5 gives the same picture per axis: by block 11 the missing axis returns to 0.21–0.47

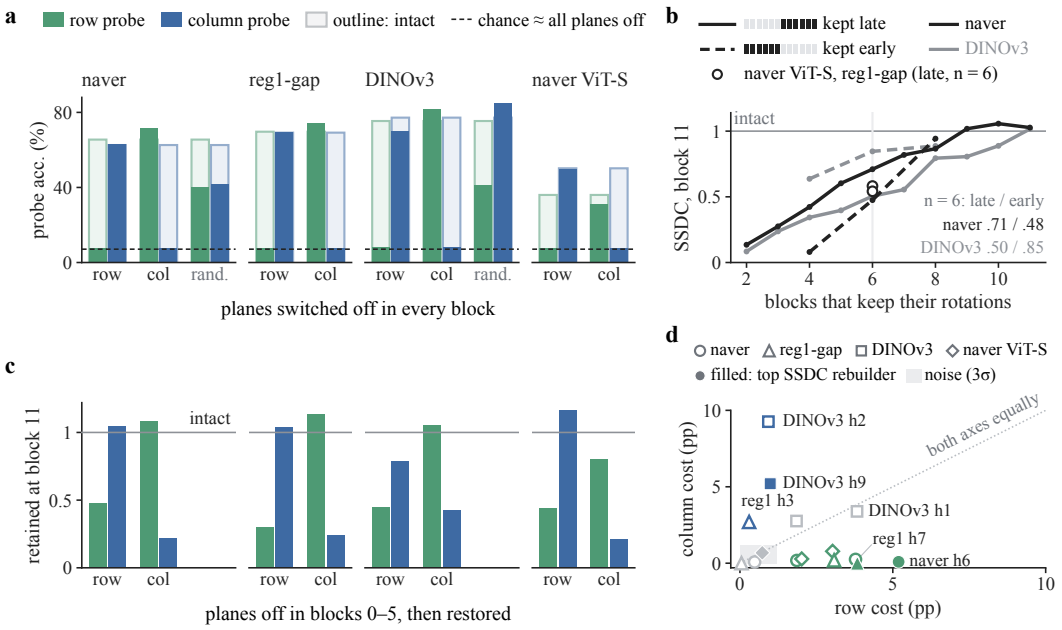

Figure 4: RoPE position comes from per-axis rotations, is recomputed through depth, and is rebuilt by axis-specialised heads. Replacing a rotation plane with the identity removes it exactly. **(a)** Row and column probe accuracy at block 4 with the row or the column planes removed in every block (outline: intact; dashed: chance, which is also where the probes sit with all planes off). The "rand." slot (naver and DINOv3) removes 16 random planes instead, which sends neither axis to chance. **(b)** Normalised SSDC at block 11 (0: all rotations off, 1: intact) when rotations are kept only in the last n blocks (solid) or only in the first n (dashed); hollow markers show the other two models for the last 6 blocks. **(c)** Row and column probes at block 11 when one axis's planes are off in blocks 0–5 and restored afterwards (1: intact). **(d)** Each head's recovery cost: probe points lost on the row (x) and column (y) probes when that head's rotations stay off in blocks 6–11 during recovery. A head is the same head index in each of those blocks. Colour marks the axis a head favours (grey: no clear preference, or both costs within noise); filled markers are the top rebuilder by SSDC, used in Section 4.5. Grey square: probe noise at three standard deviations; the filled marker inside it (naver ViT-S) has no axis-specific cost.

of intact, while the untouched one ends at or above intact on naver ViT-B and reg1-gap and near 0.8 on DINOv3 and naver ViT-S, the two models with cross-axis cost above (Figure 4c; Appendix E). So where APE adds position once at the input, RoPE models build it throughout the network, early or late depending on the model.

Particular heads rebuild each axis. Because rotations are applied separately in each head, we can ask which heads do the rebuilding. We keep one head's rotations off in blocks 6–11 while the rest recover from the 0–5 window, where a head is the same head index in each of those blocks, and rank heads by the SSDC this costs at block 11; the top-ranked head in each model is also the one whose loss costs the most ImageNet accuracy (Section 4.5). Repeating the test with only the row or only the column planes removed shows which axis each head rebuilds (Figure 4d; Appendix F). Most of the top heads rebuild one axis far more than the other, with gaps well above the probe noise of 1.2 points: head 6 of naver ViT-B costs 5.2 points of row recovery and 0.1 of column, head 2 of DINOv3 costs 9.3 points of column recovery and 0.9 of row, and reg1-gap has one head for each axis. The exceptions are one DINOv3 head that costs both axes about equally and the top head of naver ViT-S, which costs neither beyond the noise. Rebuilding is therefore not a general property of the late blocks but the work of particular heads, most of which restore a single axis, so the row–column split that the rotations impose carries through to the heads that recompute position.

4.4 Comparison: stored once versus recomputed

What both encodings share. Both encodings represent a row and a column coordinate, and in both the encoding is the only source of it: removing APE's embedding or all of RoPE's rotations

leaves SSDC below 0.03 at every block, and removing one axis's SAE features or rotation planes collapses that axis while largely sparing the other (Sections 4.2 and 4.3).

How they differ. If APE stores position and RoPE recomputes it, the two should differ in where spatial structure first appears, whether it takes the form of features, and whether it returns after being removed; all three differences hold. APE's structure is already present entering block 0, whereas RoPE's starts at the floor and is built by attention over the first few blocks (Figure 2); this holds even for DeiT-III and naver ViT-B, which share a training recipe and reach the same top-1 accuracy (Appendix G.1). APE's structure takes the form of highly selective SAE features in all four models, while the most selective features we find in RoPE models are far less selective (mean selectivity 5.2–9.8 against at most 3.4), although the SAE is also less reliable on RoPE (Appendix D.7). The most direct test is to remove position partway through the network. Once APE's features are zeroed at block 2, neither SSDC nor a position probe recovers at any later block (Figure 3b; Appendix D.4), because nothing downstream adds position again; once RoPE's rotations are withheld from blocks 0–5, later blocks rebuild 50–71% of the lost structure (Figure 4b), because every block applies the rotations again. APE therefore writes position into the residual stream once and the network carries it forward, whereas RoPE recomputes it at every block.

Both encodings in one model. The hybrid ViT-B, trained with both encodings, lets us compare the two within one model. With its rotations off, SSDC follows the intact model through block 6 and then falls to 0.09 by block 11 (intact: 0.43); with its embedding removed, SSDC stays at the floor until block 3 and then climbs to 0.42. Without its rotations the hybrid's position is even more readable than intact (exact-position probe 88.6% against 85.1%), yet top-1 falls from 80.5% to 37.4%, so in this model the embedding does not stand in for the rotations (Appendix G.2). Even within one network, then, each encoding produces its own mechanism: the embedding provides the spatial structure of the early blocks and the rotations that of the late blocks, so the mechanism follows the encoding even when both share the same weights and training.

4.5 Downstream impact: ImageNet classification

We test whether removing APE's positional features, or RoPE's main rebuilding head, costs more ImageNet top-1 accuracy (Deng et al., 2009) than removing matched components that carry no position (Figure 5).

APE: positional features. If the classifier uses APE's positional features, removing them should hurt more than removing other equally active features, and what should matter is the position they report rather than their presence alone. We test both by editing each model's features at block 2 with the SAE's reconstruction error added back, so that only the edited features change. Zeroing them costs 13.9–18.7 top-1 points more than zeroing random features matched on firing rate and activation size (Appendix H.1), so they matter more than active features that carry no position. Swapping them then shows that the position they encode is what matters. Moving each patch's features to the same patch of another image changes the image they came from but not the position they report, and costs at most 0.7 points; moving them between patches of the same image keeps their image but scrambles their position, and costs 24.8, 24.2 and 31.4 points more on supervised, AugReg and SAM ViT-B (paired standard errors 0.7 and 1.5 on supervised and SAM; Figure 5a). On single images, the second swap turns a coffee pot into a water jug while the pixels stay unchanged (Appendix H.1). Since we chose these features only by where on the grid they fire, never by their effect on the output, their effect on accuracy is an independent finding: ImageNet classification in APE models depends on the position that a small set of residual-stream features encodes.

RoPE: the top rebuilding head. In RoPE models the head that does the most rebuilding matters more for classification than any other head. We withhold rotations in blocks 0–5 as in Section 4.3 and then also remove one head's rotations in blocks 6–11, for every head in turn. The top rebuilder was fixed beforehand by its SSDC score (Section 4.3), so accuracy is an independent check of that ranking. On all three RoPE classifiers it is the costliest head: 4.3, 7.6 and 5.7 points on naver ViT-B, naver ViT-S and reg1-gap, against 3.4, 6.5 and 4.9 for the next head, and 3.1–4.6 points above the mean of the others (paired standard error 0.27 and 0.38 on the two naver models; Figure 5b). If the SSDC ranking carried no information, the chosen head would rank first in all three models

with probability 1/864. With the early rotations intact the same head costs only 0.4–1.2 points, and giving it rotations for shuffled positions costs about as much as removing them (3.5–6.2 points; Appendix H.2), so ImageNet classification in RoPE models depends on this head for rebuilding the position that the early blocks lost.

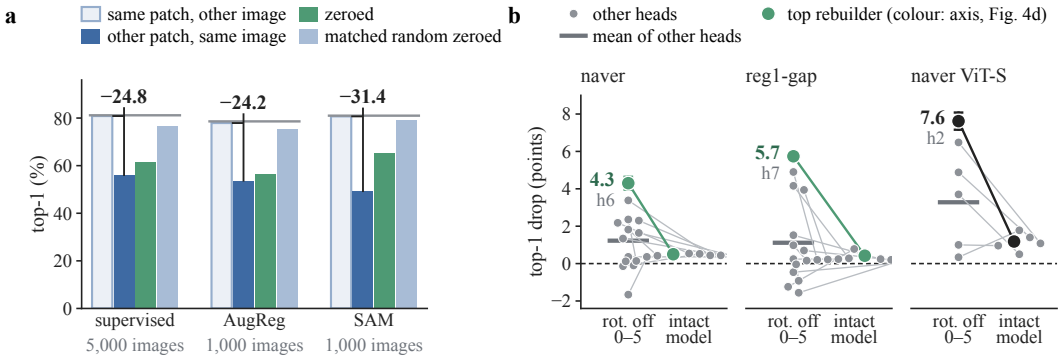

Figure 5: Both positional mechanisms matter for ImageNet classification. (a) APE: top-1 after editing the identified features at block 2 with the SAE error added back: swapped with the same patch of another image (outline) or with another patch of the same image (brackets: the gap), zeroed, or replaced by zeroing matched random features. Ticks: intact. DINO has no classification head. **(b)** RoPE: top-1 drop from removing one head's rotations in blocks 6–11, after rotations were off in blocks 0–5 (left) or in the unmodified model (right). Grey: every other head and their mean; coloured: the top rebuilder, fixed in advance (Section 4.3) and coloured by favoured axis as in Figure 4d, with its paired standard error. The panels are not on a common scale.

5 Conclusion and Discussion

Both encodings give a vision transformer a row and a column coordinate for each patch, but they store it in different places. An APE model writes position into the residual stream once, in about 20 row- and column-selective features that an SAE can isolate; zeroing these features removes spatial structure from every later block, and moving them between patches lowers top-1 accuracy by 24–31 points. A RoPE model has no comparable features. It computes position inside attention, through rotations that act on each axis separately and are applied again at every block, which lets later blocks rebuild much of the structure lost in the early ones; a few heads do this rebuilding, and most of them restore a single axis. Where a model keeps position is therefore set by its encoding, and this determines how position can be studied and edited: with SAE features in APE models, and with the rotations and the heads that use them in RoPE models.

Three limitations qualify these conclusions. Because we study APE through feature ablation and RoPE through rotation masking, we compare the two mechanisms qualitatively and not by effect size; only one pair of models, DeiT-III and naver ViT-B, shares a training recipe, and all models are ViT-B or ViT-S at 224×224. The APE features also do not account for all decodable position. When the SAE's reconstruction error is added back after the features are zeroed, the exact-position probe at block 2 recovers to 57–93%, although SSDC stays low and top-1 still falls by 16–22 points, so the features explain the spatial structure and most of its effect on classification but not everything a probe can read. Finally, each head in our RoPE experiments is one head index spanning six blocks, and the attention statistics we examined do not explain what the rebuilding heads compute (Appendix F.3). Masking single heads in single blocks would address this, and extending the analysis to larger models and higher resolutions, where SSDC becomes less sensitive (Appendix H.3), would show how far these mechanisms generalise.

Reproducibility Statement

All checkpoints are public and listed with exact identifiers in Appendix A.2; all images come from the ImageNet-1k validation split. Appendix I gives sample sizes, seeds, probe and masking protocols, the implementation checks every experiment runs, and the hardware used. Code, SAE weights,

the selected feature lists and all result files are available in the anonymous repository linked in Appendix I.

AI USE STATEMENT

We used generative AI tools to implement parts of the experiment and analysis code, to refine control experiments, and to help interpret results. We did not use them to generate data sets, formulate mathematical claims or write proofs; translation and dataset cleaning do not apply to this work. We also used them to create figures, search for and format related literature, and edit the text for readability. All AI-written code passed the implementation checks in Appendix I and was checked against the models' source code, every reported number was traced to a result file, and every citation was checked against its published version. We take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

REFERENCES

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

Nahid Alam, Leema Krishna Murali, Siddhant Bharadwaj, Patrick Liu, Timothy Chung, Drishti Sharma, Akshata A, Kranthi Kiran, Wesley Tam, and Bala Krishna S. Vegesna. The spatial blindspot of vision-language models. *arXiv preprint arXiv:2601.09954*, 2026.

Federico Barbero, Alex Vitvitskyi, Christos Perivolaropoulos, Razvan Pascanu, and Petar Veličković. Round and round we go! what makes rotary positional encodings useful? In *International Conference on Learning Representations*, 2025.

Yonatan Belinkov. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219, 2022.

Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nick Turner, Cem Anil, Carson Denison, Amanda Askell, et al. Towards monosemanticity: Decomposing language models with dictionary learning. *Transformer Circuits Thread*, 2023.

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

Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform ResNets without pre-training or strong data augmentations. In *International Conference on Learning Representations*, 2022.

Ta-Chung Chi, Ting-Han Fan, Alexander Rudnicky, and Peter Ramadge. Dissecting transformer length extrapolation via the lens of receptive field analysis. In *Annual Meeting of the Association for Computational Linguistics*, 2023.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, 2024.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations*, 2024.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. EVA-02: A visual representation for neon genesis. *arXiv preprint arXiv:2303.11331*, 2023.

Thomas Fel, Ekdeep Singh Lubana, Jacob S. Prince, Matthew Kowal, Victor Boutin, Isabel Papadimitriou, Binxu Wang, Martin Wattenberg, Demba Ba, and Talia Konkle. Archetypal SAE: Adaptive and stable dictionary learning for concept extraction in large vision models. In *International Conference on Machine Learning*, 2025.

Anthony Fuller, Daniel G. Kyrollos, Yousef Yassin, and James R. Green. LookHere: Vision transformers with directed attention generalize and extrapolate. In *Advances in Neural Information Processing Systems*, 2024.

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*, 2025.

Adi Haviv, Ori Ram, Ofir Press, Peter Izsak, and Omer Levy. Transformer language models without positional encodings still learn positional information. In *Findings of the Association for Computational Linguistics: EMNLP*, 2022.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, 2024.

Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. In *Advances in Neural Information Processing Systems*, 2023.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. In *IEEE/CVF International Conference on Computer Vision*, 2023.

Hyesu Lim, Jinho Choi, Jaegul Choo, and Steffen Schneider. Sparse autoencoders reveal selective remapping of visual concepts during adaptation. In *International Conference on Learning Representations*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Mahmoud Mannes. Positional encodings anchor spatial structure in vision transformers: A geometric perspective on robustness. *arXiv preprint arXiv:2606.00124*, 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, 2022.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

Jianing Qi, Jiawei Liu, Hao Tang, and Zhigang Zhu. Beyond semantics: Rediscovering spatial awareness in vision-language models. *arXiv preprint arXiv:2503.17349*, 2025.

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 *International Conference on Machine Learning*, 2021.

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.

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.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. Scaling monosemanticity: Extracting interpretable features from Claude 3 Sonnet. *Transformer Circuits Thread*, 2024.

Hugo Touvron, Matthieu Cord, and Hervé Jégou. DeiT III: Revenge of the ViT. In *European Conference on Computer Vision*, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, 2017.

Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. In *Advances in Neural Information Processing Systems*, 2020.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *International Conference on Learning Representations*, 2023.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, et al. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Ross Wightman. PyTorch image models. `https://github.com/huggingface/pytorch-image-models`, 2019.

APPENDIX

APPENDIX CONTENTS

A Background, models and data **14**

A.1 Patches, positions and the two schemes . . . 14

A.2 Models and data . . . 14

A.3 Terms used in the paper . . . 14

B Experiments at a glance **15**

C Localization: patching and ablation **16**

C.1 Protocol . . . 16

C.2 Results . . . 16

C.3 Alignment controls . . . 17

D Sparse autoencoder analyses **17**

D.1 Training and configuration . . . 17

D.2 Feature screen . . . 18

D.3 Dose-response . . . 19

D.4 Probes through depth . . . 19

D.5 Controls: matched features, the SAE error and swaps . . . 20

D.6 Axis separation . . . 22

D.7 RoPE: SAE reconstruction is not a reliable intervention . . . 22

E Rotation masking **23**

E.1 Removing one axis . . . 23

E.2 Absolute position: the border and the prefix tokens . . . 23

E.3 Masking windows . . . 24

F Head attribution **26**

F.1 Axis specialisation . . . 26

F.2 Carrying versus rebuilding . . . 26

F.3 Attention statistics . . . 26

G Comparison: a shared recipe and a shared model **27**

G.1 DeiT-III and naver ViT-B . . . 27

G.2 The hybrid checkpoint . . . 27

H Classification **28**

H.1 APE: every edit . . . 28

H.2 RoPE: every head . . . 29

H.3 Replication and higher resolution . . . 30

I Reproducibility: code, data, compute and protocol **30**

A Background, models and data

A.1 Patches, positions and the two schemes

A ViT-B/16 at 224×224 splits the image into $14 \times 14 = 196$ patches, embeds each as a vector, and prepends one or more extra tokens such as the class token; each block then mixes the tokens with self-attention and transforms them with an MLP. APE adds a learned vector $p_{r,c}$ to the patch at row r and column c, once, before block 0. After that, position is part of the patch vector, and the network is free to move it, copy it or discard it.

RoPE adds no positional vector to the residual stream. Inside each attention layer it splits the query and key vectors into two-dimensional planes and rotates each plane by an angle proportional to one of the patch's coordinates; a ViT-B RoPE head has 64 dimensions, which pair into 32 planes, 16 turned by the row and 16 by the column. The rotation touches queries and keys but not values, so position enters only through the attention weights, and it enters again at every block. Our experiments rely on this difference: APE's signal is added once and can only be reached through the representation it produces, while RoPE's is an operation repeated at every block that we can replace with the identity wherever we choose.

A.2 Models and data

Table 1 lists every checkpoint. All are loaded through `timm` (Wightman, 2019), except supervised ViT-B, which comes from the HuggingFace hub. Every model is evaluated at 224×224 input, a 14×14 patch grid; reg1-gap was trained at 256×256 and runs at 224 through `timm`'s dynamic image size. "Prefix" is the number of non-patch tokens.

For the RoPE models, `naver_in1k` is the ImageNet-1k recipe of Heo et al. (2024), which follows DeiT-III; `sbb_in1k` is the `timm` ImageNet-1k recipe with a register token and global average pooling; and `lvd1689m` is DINOv3's self-supervised pretraining. The `timm` RoPE implementations assign rotation planes to the two axes in opposite orders, and DINOv3 tiles the two components of each plane where the others interleave them, so we derive the plane-to-axis mapping at runtime from each model's frequency table.

All images come from the ImageNet-1k validation split (Deng et al., 2009), streamed from the HuggingFace hub in a fixed order; each experiment uses the first N images of the stream, with N given in Appendix I.

Table 1: Checkpoints. "IN-21k → IN-1k": supervised pretraining on ImageNet-21k, then fine-tuning on ImageNet-1k.

Short name	Checkpoint	PE	Heads	Prefix	Training
supervised ViT-B	`google/vit-base-patch16-224`	APE	12	1	IN-21k → IN-1k
DINO ViT-B	`vit_base_patch16_224.dino`	APE	12	1	DINO, IN-1k
AugReg ViT-B	`vit_base_patch16_224.augreg_in1k`	APE	12	1	IN-1k
SAM ViT-B	`vit_base_patch16_224.sam_in1k`	APE	12	1	IN-1k
DeiT-III ViT-B	`deit3_base_patch16_224.fb_in1k`	APE	12	1	DeiT-III, IN-1k
naver ViT-B	`vit_base_patch16_rope_224.naver_in1k`	RoPE	12	1	`naver_in1k`
naver ViT-S	`vit_small_patch16_rope_224.naver_in1k`	RoPE	6	1	`naver_in1k`
reg1-gap ViT-B	`vit_base_patch16_rope_reg1_gap_256.sbb_in1k`	RoPE	12	1	`sbb_in1k`
DINOv3 ViT-B	`vit_base_patch16_dinov3.lvd1689m`	RoPE	12	5	`lvd1689m`
hybrid ViT-B	`vit_base_patch16_rope_ape_224.naver_in1k`	both	12	1	`naver_in1k`

A.3 Terms used in the paper

Table 2 collects the terms the paper uses repeatedly, and Figure 6 shows how SSDC and RPI work.

Table 2: Terms used throughout.

Term	Meaning
Block b	The hidden state entering transformer block b (read after its first LayerNorm), blocks numbered 0–11. The patching experiments of Appendix C are the one exception: they read the output of block b.
SSDC	Rank correlation between how similar two patches' hidden states are and how close the patches are on the grid, averaged over patch pairs and images. 1: nearer patches are always more similar; 0: similarity ignores distance.
RPI	Random permutation at inference: patch contents are shuffled after the patch embedding, while position stays attached to each sequence index. Structure that survives comes from position, not image content.
Floor	The same measurement with all positional signal removed (APE: embedding zeroed; RoPE: every rotation replaced by the identity).
Normalised SSDC	$(\text{observed} - \text{floor})/(\text{intact} - \text{floor})$ at the same block: 0 is the floor, 1 the intact model. Used for RoPE only.
Probe	Linear classifier for a patch's exact position (196 classes), row or column (14 classes each), trained on hidden states under RPI.
Positional features	The row- and column-selective SAE latents kept by the screen of Appendix D.2: 18–24 per APE model.
Rotation plane	A pair of query/key dimensions that RoPE rotates together, by an angle set by either the row or the column. Masking a plane replaces its rotation with the identity.
Window	The blocks in which rotations are removed, e.g. "0–5": off in blocks 0 to 5, on from block 6.
Head	One head index across a range of blocks (e.g. head 6 in each of blocks 6–11), not a single head in a single block.
Rebuilder	A head whose rotations, when kept off after the 0–5 window, most reduce the SSDC that returns by block 11.
Carrier	A head whose rotations, when removed in every block of the unmodified model, most reduce SSDC at block 4.

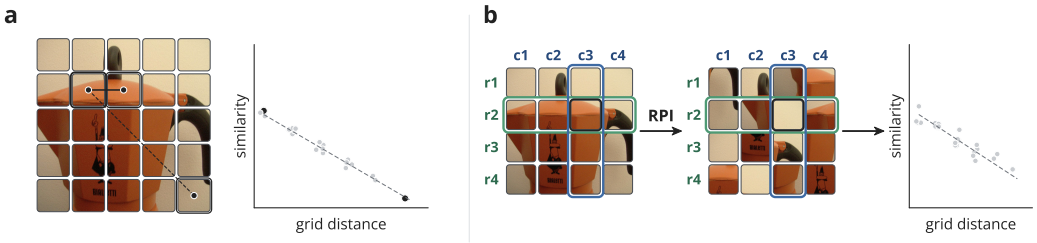

Figure 6: How SSDC and RPI work, on a real ImageNet image. **(a)** For every pair of patches we compare the cosine similarity of their hidden states with their distance on the grid; SSDC is the rank correlation between the two, averaged over images. **(b)** Under RPI the patch contents are shuffled before block 0 while each sequence index keeps its position, so neighbouring indices no longer hold neighbouring image content. Any SSDC that remains must come from position.

B Experiments at a glance

Table 3 lists every experiment with its intervention, controls, readouts and models.

Table 3: Experiments at a glance. "Floor": all positional signal removed. "SAE recon.": activations passed through the SAE with nothing edited. "+ error": the SAE's reconstruction error added back, so only the edited features change.

Experiment	Intervention	Controls	Readouts	Models
Depth profile (§4.1)	None	Floor	SSDC per block	all
Patching (App. C)	Copy or zero one attention or MLP output	Clean run, RPI run, misaligned and other-image donors	SSDC, top-1	sup. ViT-B, naver ViT-B
APE features (§4.2)	Zero SAE features at block 2, by dose or axis	SAE recon., random and matched random features	SSDC, probes	4 APE
APE controls (App. D.5)	Zero, mean-ablate or swap features, with error added back	Matched random features; embedding zeroed or permuted	SSDC, probes, top-1	4 APE, DeiT-III
Rotation planes (§4.3)	Identity on all, row or column planes	16 random planes, floor	SSDC, probes	4 RoPE
Border (App. E.2)	Probe split by distance to border; patch-to-prefix attention blocked	Intact	Probes, SSDC, top-1	4 RoPE
Windows (§4.3)	Rotations off in blocks 0–k, then restored; suffix and middle windows	Intact, floor	Normalised SSDC, probes	4 RoPE (full sweep on 2)
Heads (§4.3)	One head's rotations off in blocks 6–11 after the 0–5 window, all planes or one axis	All heads restored	SSDC, probes	4 RoPE
Classification (§4.5)	Every head's rotations off in 6–11; APE features swapped or zeroed	Other heads; intact model; shuffled rotations; same-patch swap; matched features	Paired top-1	3 RoPE, 3 APE

C Localization: patching and ablation

This appendix gives the patching and zero-ablation experiments behind Section 4.1, and controls showing that what the patches restore is position rather than image content.

C.1 Protocol

We run each image twice, once in normal patch order and once under RPI, copy one attention or MLP output from the normal run into the RPI run, and measure recovery as the fraction of the SSDC gap between the two runs that the patch closes. Zero-ablating the same output, with the residual connection kept, tests whether the component is needed at all. Each condition uses supervised ViT-B and naver ViT-B, 512 validation images and two seeds for image order and permutation. Unlike the rest of the paper, these experiments read SSDC from the *output* of block b rather than from the input to block b, so "block b" here corresponds to "block $b+1$" elsewhere.

C.2 Results

Figure 7 gives the full results.

APE: the first MLP. Patching the block-0 MLP alone restores 80–83% of the final-block SSDC lost under RPI in the two runs, and more than all of it at blocks 4 and 5 (121% and 105%). Patching later components restores accuracy but little structure: block-11 attention restores 97–100% of the lost accuracy.

RoPE: attention over several blocks. Patching block-5 attention restores 45% of the SSDC lost after that block, and zero-ablating attention in block 2, 3 or 4 lowers block-4 SSDC by 0.079–0.088 in both runs while changing accuracy on unpermuted images by less than 0.5 points.

How long attention keeps adding structure. Attention adds structure in more blocks in RoPE than in APE: under RPI it raises residual SSDC in blocks 0–7 in RoPE (most in blocks 4 and 5, by about 0.12 each) but only in blocks 0–2 in APE (most in block 1, by 0.222), and lowers it after block 3.

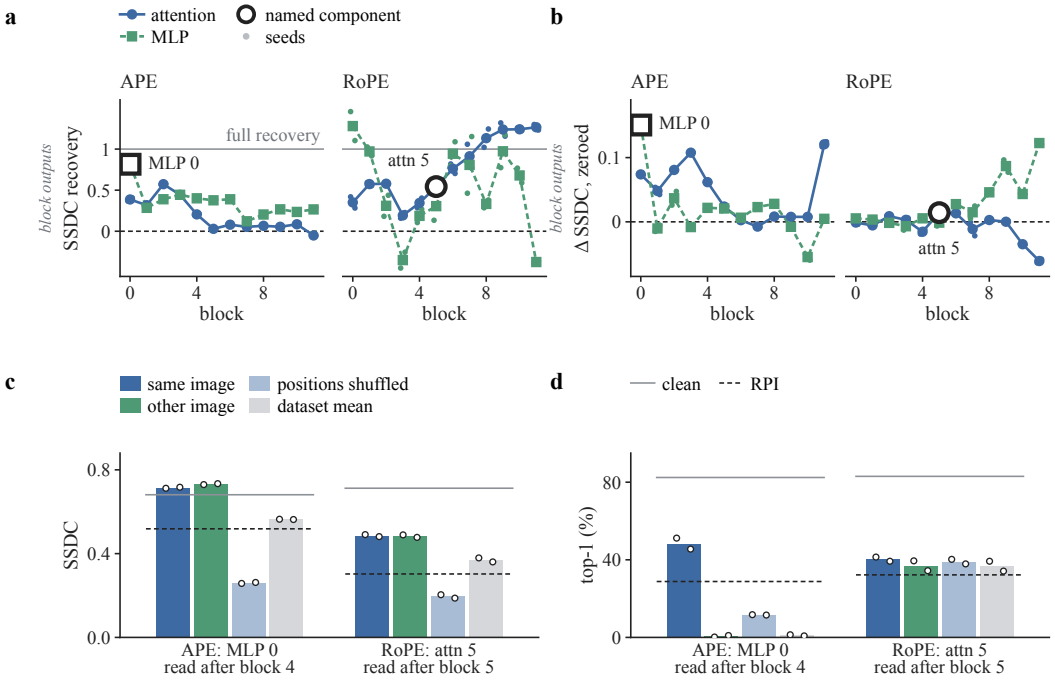

Figure 7: Patching locates APE's structure in the first MLP and RoPE's in attention over several blocks. Supervised ViT-B (APE) and naver ViT-B (RoPE), 512 images; small dots are the two seeds. **(a)** Fraction of the final-block SSDC gap between the normal and RPI runs closed by copying one attention (circles) or MLP (squares) output from the normal run; the components named in the text are ringed. **(b)** Change in final-block SSDC under RPI when the same output is zeroed. **(c)** SSDC after patching APE's block-0 MLP (read after block 4) or RoPE's block-5 attention (read after block 5) with activations from the same image, from another image at the same positions, from the same image with positions shuffled, or from the dataset mean; lines mark the clean and RPI runs. **(d)** Top-1 under the same donors. SSDC is read at block outputs here (Appendix C.1).

C.3 Alignment controls

Under RPI the patch is copied back into the permuted sequence at matching positions, so a patch could restore SSDC by restoring image content rather than position. We test this at APE's block-0 MLP and RoPE's block-5 attention with four donors (Figure 7c,d). Taking the donor activations from a different image, at the same positions, leaves the recovered SSDC unchanged (0.715 to 0.732 in APE, 0.486 to 0.484 in RoPE), while shuffling the donor's positions within the same image drops it to 0.260 and 0.196. A dataset-mean donor, which carries no image content at all, still restores 0.56 in APE. In APE the different-image donor also drops top-1 from 48.3% to 0.6%, so the recovered structure is positional: it survives a change of image that destroys classification, and disappears when positions are misaligned.

D Sparse autoencoder analyses

This appendix supports Section 4.2: the SAE configuration, the feature screen, the dose-response on all four APE checkpoints, the controls for the ablation, the axis test, and why the SAE is not a reliable instrument on RoPE models.

D.1 Training and configuration

Each APE model has one SAE, trained on its residual stream entering block 2 (Table 4).

Table 4: SAE configuration. One SAE per APE checkpoint, all with the same architecture and training setup; the learning rate and the held-out metrics differ by model.

Site	residual stream entering block 2, before LayerNorm
Architecture	TopK (Gao et al., 2025), $k = 64$, with the AuxK loss for dead latents and a unit-norm decoder
Latents	3,840 (5× the width of 768)
Training data	all 197 tokens (class token included) of 20,000 ImageNet images, scaled so that $\mathbb{E}\|x\| = \sqrt{768}$; 5% of images held out
Optimisation	AdamW (Loshchilov & Hutter, 2019), batch 512 tokens, 2 epochs, 5% linear warm-up

Model	Learning rate	Held-out FVU	Dead latents
supervised ViT-B	4×10^{-4}	0.142	3.3%
DINO ViT-B	2×10^{-4}	0.139	0.0%
AugReg ViT-B	4×10^{-4}	0.126	5.7%
SAM ViT-B	1×10^{-4}	0.114	0.0%

D.2 Feature screen

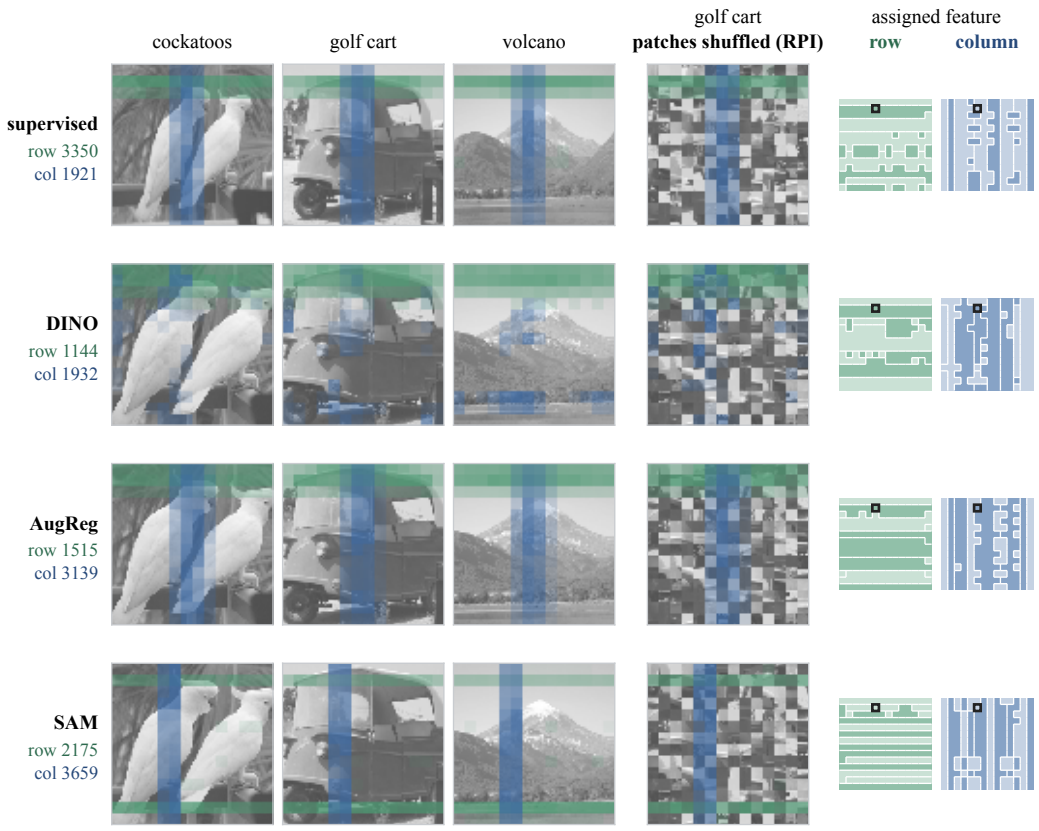

Figure 8: Positional features respond to position, not content. For each APE model, the row feature (green) and column feature (blue) that the screen assigns to the patch at row 1, column 5, overlaid on three ImageNet validation images (residual stream entering block 2). Each feature lights the same band on every image, and with the image's patches shuffled (fourth column) the content moves but the bands stay put. Right: the row and column feature assigned to every position; the outlined cell is row 1, column 5. DINO's column feature is shown to scale, fainter than the others.

For each of the 196 grid positions (excluding the class token) we shortlist SAE features by their mean activation at that position minus their mean activation elsewhere, score each candidate's 14×14 mean-activation map for row and column selectivity, and keep the best row feature and the best column feature. Row selectivity is the mean activation of a feature's most active row divided by

its mean over the whole grid, and likewise for columns. Many positions share a feature, so the screen keeps 20 features in supervised ViT-B, 18 in DINO, 20 in AugReg and 24 in SAM. Each feature counts as a row or a column feature by whichever selectivity is larger. On three of the four checkpoints a few features win positions on both axes, so the row and column subsets overlap slightly and their sizes sum to more than the full set.

Figure 8 shows what these features do on real images. For each model we take the row and column features the screen assigns to one patch (row 1, column 5, as in Figure 3a) and plot their activation on every patch of three ImageNet validation images. Each feature lights the same band on the cockatoos, the golf cart and the volcano, and shuffling the image's patches moves the content but not the band, so the features respond to where a patch is, not to what it shows. Two features are less clean: SAM's row feature has a second band at row 12, and DINO's column feature is weak and diffuse.

D.3 Dose-response

We zero features in selectivity order at increasing doses and train a linear probe for exact position on the residual stream under RPI (Table 5). On every checkpoint, the SAE round trip and a random set of the same size leave the probe near its intact value, and accuracy falls steadily as more of the selected features are removed. Most of the probe's drop in supervised ViT-B comes between 4 and 8 features, while most of the SSDC drop comes with the last few (Figure 3b): the probe loses precision as soon as a few bands go, but SSDC, which only needs nearby patches to be more similar than distant ones, holds until almost every band is gone.

Table 5: Exact-position probe accuracy (%) at block 2 as positional features are zeroed. Chance is 0.5%. Supervised ViT-B uses 10,000 images, the others 1,000. The random control zeroes as many random features as the full set. "—": dose not run.

	supervised	DINO	AugReg	SAM
Intact (no SAE)	99.99	95.9	99.94	100
SAE reconstruction	99.7	82.1*	99.96	100
Random features zeroed	99.1	83.4	99.8	100
4 positional features	95.9	58.7	95.0	—
8	60.5	28.4	64.7	—
12	44.0	18.2	53.7	—
16	33.2	6.6	36.8	—
Full set (20 / 18 / 20 / 24)	22.9	4.3	29.3	11.3

*A rerun with a validation-selected epoch (Table 7) finds a smaller reconstruction cost on DINO (99.9% to 96.8%). SAM's intermediate doses are 7 and 14 features (77.5% and 27.5%).

D.4 Probes through depth

Table 6 follows exact-position probe accuracy through all twelve blocks after the features are zeroed at block 2. Once the full set is gone, accuracy stays low all the way to block 11 on every model and never rises by more than 1.5 points, while a random set of the same size leaves every block at the reconstruction level.

Table 6: Exact-position probe accuracy (%) at every block after zeroing features at block 2. Linear probe for a patch's grid position (196 classes, chance 0.5%) on the residual stream entering each block, under RPI, 1,000 images per model. Blocks 0–1 come before the edit. The random control zeroes 20 random features (24 on SAM). Probes: learning rate 5×10^{-3}, weight decay 10^{-4}, batch 512, 20 passes, 10% of tokens held out; we report the best held-out accuracy over passes.

		Block											
Model	Condition	0	1	2	3	4	5	6	7	8	9	10	11
supervised	intact (no SAE)	100.0	100.0	100.0	98.9	95.1	71.3	44.7	32.1	26.8	21.5	15.9	8.1
	SAE reconstruction	100.0	100.0	100.0	98.6	92.5	66.6	40.7	30.2	24.0	20.0	15.1	7.5
	all 20 positional zeroed	100.0	100.0	21.5	20.5	18.5	14.4	9.7	7.3	6.6	5.8	5.0	3.2
	20 random zeroed	100.0	100.0	100.0	98.6	92.4	66.6	41.1	29.8	24.5	20.0	14.6	7.2
DINO	intact (no SAE)	69.2	93.7	96.6	95.0	90.1	84.4	79.7	75.7	72.9	70.4	69.6	68.8
	SAE reconstruction	69.2	93.7	85.8	90.8	89.2	87.3	84.8	82.8	80.3	77.9	77.4	77.3
	all 18 positional zeroed	69.2	93.7	3.8	3.9	3.8	3.9	4.0	4.2	4.3	4.7	5.1	5.3
	20 random zeroed	69.2	93.7	86.0	90.8	89.3	87.4	84.8	82.9	80.3	77.8	77.5	77.2
AugReg	intact (no SAE)	100.0	100.0	99.9	99.8	99.0	93.5	86.3	76.5	66.9	57.1	47.1	35.3
	SAE reconstruction	100.0	100.0	99.8	99.2	98.0	89.5	80.4	72.1	61.9	52.8	43.9	33.0
	all 20 positional zeroed	100.0	100.0	28.4	24.0	20.5	17.3	13.9	11.4	9.0	7.3	5.9	4.1
	20 random zeroed	100.0	100.0	99.8	99.3	97.8	89.2	80.6	70.9	61.7	52.7	44.1	32.9
SAM	intact (no SAE)	99.9	100.0	100.0	100.0	100.0	98.6	97.5	96.1	94.4	93.2	92.1	91.3
	SAE reconstruction	99.9	100.0	100.0	100.0	99.9	97.8	96.4	94.9	92.5	91.4	89.9	88.9
	all 24 positional zeroed	99.9	100.0	11.3	12.2	12.8	12.5	12.2	12.3	12.3	12.3	11.8	11.7
	24 random zeroed	99.9	100.0	100.0	100.0	99.9	97.7	96.3	94.8	92.4	91.2	89.6	88.8

D.5 Controls: matched features, the SAE error and swaps

Comparing against a random set leaves three questions open: whether it is a fair comparison, since most SAE latents fire rarely; whether the result depends on what the SAE fails to reconstruct; and whether zeroing hurts the model for reasons unrelated to position. Figure 9 shows the edits we use to answer them, and Table 7 and Figure 10 give the results.

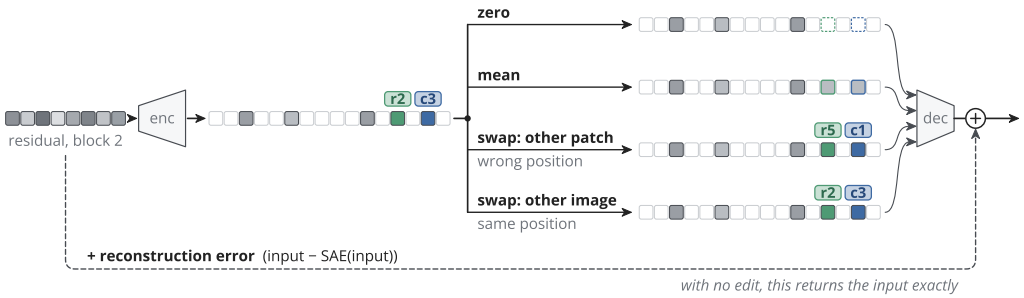

Figure 9: The edits applied to the positional features at block 2. Each patch's residual stream is encoded by the SAE; the positional features (coloured) are then zeroed, replaced by their mean, swapped with those of another patch in the same image, or swapped with those of the same patch in another image, and decoded. Adding the reconstruction error back (bottom path) makes the unedited model exactly the intact one, so only the edited features change.

Matched random features. The positional features fire on 14–31% of tokens, against a dictionary median below 0.5%, so a uniformly drawn random set is mostly silent. We therefore draw five random sets matched to the positional set on firing rate and activation size. Zeroing them leaves SSDC and the exact-position probe at the reconstruction level on every model, although it costs 2–5 top-1 points beyond the reconstruction (Appendix H.1), so they are active features that carry no position.

The SAE error. With the reconstruction error added back, the unedited model is exactly the intact one (identical SSDC and top-1), so only the edited features change. Zeroing the positional features then still leaves block-2 SSDC at 0.07–0.25, but the exact-position probe recovers to 57–93%. Linearly decodable position therefore also sits in what the SAE does not reconstruct, while the similarity structure that SSDC measures is carried by the features.

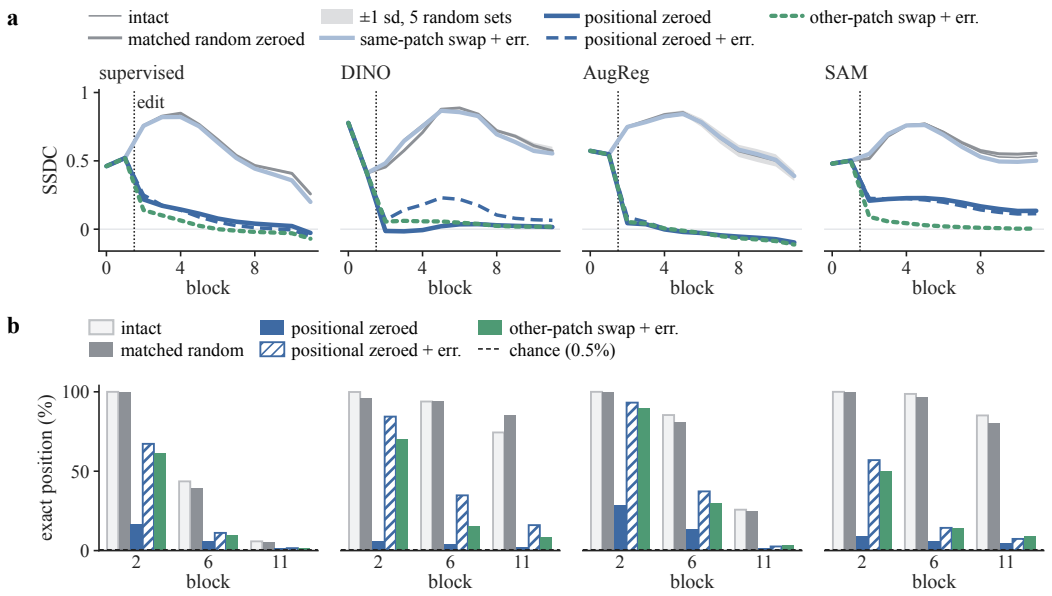

Figure 10: Only edits that remove or scramble the positional features remove spatial structure, and it does not come back. (a) SSDC under RPI entering each block after an edit at block 2 (dotted line), for the four APE checkpoints; the band is ±1 standard deviation over five matched random sets. **(b)** Exact-position probe accuracy at blocks 2, 6 and 11 (chance 0.5%). "+ err.": with the SAE reconstruction error added back. 1,000 images.

Mean ablation and swaps. Replacing the features with their mean has the same effect as zeroing them. Swapping them between images at the same patch changes nothing measurable, while swapping them between patches of the same image, so that each patch reports another patch's position, removes SSDC and most of the probe accuracy. What matters is the position the features report, not how strongly they fire.

No rebuilding. After the features are zeroed at block 2, the exact-position probe stays low through block 11 on every model (Appendix D.4), and SSDC does not recover. The one partial exception is DINO with the error added back, whose SSDC recovers to 0.22 at block 6 (intact: 0.86) before falling again.

Table 7: APE controls at the intervention site. SSDC and exact-position probe accuracy (%) at block 2 after each edit, 1,000 images. Matched random: SSDC is the mean of five draws, the probe uses the first draw. The embedding rows remove or permute APE's added embedding before block 0 and use no SAE.

	supervised		DINO		AugReg		SAM	
Edit at block 2	SSDC	probe	SSDC	probe	SSDC	probe	SSDC	probe
Intact	0.76	100.0	0.47	99.9	0.75	100.0	0.55	100.0
SAE reconstruction	0.75	100.0	0.44	96.8	0.74	99.9	0.54	100.0
Positional zeroed	0.22	16.6	−0.01	5.9	0.04	28.4	0.21	8.8
Uniform random zeroed	0.75	100.0	0.44	97.3	0.75	99.9	0.54	100.0
Matched random zeroed	0.76	100.0	0.45	96.1	0.75	99.9	0.52	100.0
With the SAE error added back								
Positional zeroed	0.25	67.2	0.07	84.4	0.09	93.2	0.23	57.0
Positional mean-ablated	0.24	—	0.09	—	0.10	—	0.15	—
Swapped, same patch of another image	0.76	99.9	0.48	97.4	0.75	99.9	0.55	100.0
Swapped, other patch of the same image	0.14	61.3	0.06	70.4	0.05	90.0	0.09	50.2
Matched random zeroed	0.76	—	0.48	—	0.75	—	0.54	—
Embedding zeroed	0.01	0.6	0.01	0.5	0.01	0.5	0.01	0.6
Embedding permuted	0.01	100.0	0.01	99.9	0.01	100.0	0.00	100.0

Permuting the added embedding across patches leaves exact position fully decodable, since each index still receives a fixed vector, but removes SSDC, because neighbouring indices no longer receive

similar vectors. SSDC therefore measures whether position is organised like the grid, which a probe for exact position does not.

D.6 Axis separation

Figure 3c summarises the axis test; Table 8 gives the numbers. The probes are ridge regressions on the normalised row and column coordinates, reported as test R^2 on a 600/200/200 split by image, with a control fit to shuffled labels. DINO, AugReg and SAM use 1,000 images.

Table 8: Zeroing one axis's features removes that axis. Test R^2 of ridge probes for the row and column coordinates at block 2. "Own axis off" zeroes the row features for the row probe and the column features for the column probe. Bold: the probe whose own features were removed. Probes fit to shuffled labels read $R^2 \approx 0$ throughout.

	Intact	SAE recon.	Random	Own axis off
Row probe				
supervised	0.992	0.990	0.989	**0.515**
DINO	0.990	0.976	0.976	**0.137**
AugReg	0.991	0.987	0.987	**0.451**
SAM	0.991	0.972	0.972	**0.432**
Column probe				
supervised	0.988	0.990	0.989	**0.705**
DINO	0.989	0.954	0.953	**0.134**
AugReg	0.974	0.977	0.978	**0.557**
SAM	0.981	0.973	0.973	**0.266**

Zeroing one axis's features drives that axis's probe to $R^2 \leq 0.56$ while the other stays above 0.92, and neither the reconstruction nor the random control moves either probe by more than 0.03.

Cost to the other axis. The cost to the other axis is at most 0.001 on supervised ViT-B and SAM, 0.031 on AugReg, and 0.017 and 0.055 on DINO, the largest of the four; the largest cross-axis cost among the RoPE models is also on a DINO-trained model (Section 4.3), which points to the training objective rather than the encoding.

Both axes together. Zeroing both subsets together in supervised ViT-B leaves the row probe at 0.514 and the column probe at 0.715, the same as zeroing each alone, so the two subsets act independently. No probe reaches zero, so some per-axis information lies outside the screened features (Appendix D.5). The supervised model loses more on the row axis than on the column axis (ΔR^2 0.475 against 0.285); since the screen and the SAE may resolve the two axes unevenly, we do not read this as a property of the model.

D.7 RoPE: SAE reconstruction is not a reliable intervention

The same SAE approach does not transfer to RoPE models, for two reasons. First, the reconstruction alone moves SSDC too much: passing naver ViT-B's residual stream through its SAE and back raises SSDC by up to 0.214 before any feature is removed, against 0.062 and 0.065 on the two APE models measured the same way. Second, on DINOv3 no dose moves SSDC enough: zeroing 22 features, the largest dose we tried, lowers SSDC only from 0.948 to 0.863 at the intervention site, well short of the 30% drop we set in advance as the bar for a usable edit.

The screen also finds nothing coordinate-like in RoPE models. At block 2, the screened APE features have a mean row and column selectivity of 5.2–9.8 (supervised 7.2 and 7.9, DINO 5.2 and 5.7, AugReg 6.6 and 7.5, SAM 9.8 and 9.1). On naver ViT-B the most selective features reach at most 3.4 at blocks 2, 4, 6, 8 and 10 (row 3.1, 3.4, 3.0, 3.2, 1.8; column 2.2, 3.3, 3.0, 1.9, 1.3), and on DINOv3 at most 1.8 at block 4. We therefore intervene on RoPE's rotations directly, which can be edited exactly by axis, block and head, with no reconstruction step.

E ROTATION MASKING

This appendix supports Sections 4.3 and 4.3: the axis-removal numbers behind Figure 4a, where RoPE models anchor absolute position, and every masking window behind Figure 4b,c. Figure 11 shows what the masking does.

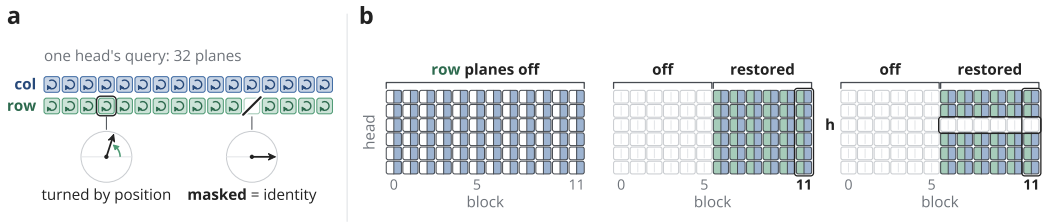

Figure 11: Rotation masking. (a) A RoPE head rotates 32 planes of its queries and keys, 16 by the patch's column and 16 by its row; masking replaces a plane's rotation with the identity. **(b)** The three scopes we use, shown on a grid of heads × blocks: one axis's planes off in every block (left); every rotation off in blocks 0–5 and restored from block 6, with SSDC and probes read at block 11 (middle); and the same, with one head index h also kept off in blocks 6–11 (right).

E.1 REMOVING ONE AXIS

Table 9: Removing one axis's rotation planes in every block, read at block 4 (1,000 images). Bold: the probe whose own planes were removed; chance is 7.1%. With every plane off (the floor), SSDC is 0.011–0.018 and both probes 7.5–8.1% on all four models.

	SSDC			Row probe (%)			Column probe (%)		
Model	intact	row off	col off	intact	row off	col off	intact	row off	col off
naver ViT-B	0.469	0.159	0.234	65.5	**7.8**	71.7	62.6	63.4	**7.8**
reg1-gap ViT-B	0.689	0.262	0.447	69.7	**7.8**	74.2	69.2	69.8	**7.6**
DINOv3 ViT-B	0.949	0.370	0.649	75.4	**8.2**	81.7	77.2	70.3	**8.3**
naver ViT-S	0.371	0.260	0.036	36.0	**7.8**	31.0	50.2	49.7	**7.9**

A random set of 16 planes, one draw on naver ViT-B and DINOv3, sends neither axis to chance: it leaves the row and column probes at 40.5% and 42.0% on naver and 41.2% and 85.0% on DINOv3. On naver it removes more SSDC than either axis group (0.050, against 0.159 and 0.234) while removing less axis information, so a drop in SSDC does not by itself measure how much positional information is gone. The random planes are matched to the axis groups in number, not frequency.

On DINOv3, removing the row planes also costs the column probe 7 points, and inside a 0–5 window the column probe falls from 83.9% to 64.7% at block 6 and from 78.7% to 65.1% at block 9 (two of the five failed implementation checks, Appendix I). This cost does not grow with the window: with row planes off over blocks 0–2, 0–5 and 0–8, the column probe stays at 78.0%, 79.5% and 80.7% while the row probe falls from 65.2% to 45.6% to 29.8%. It is a property of how DINOv3's planes divide between the axes, not of recovery.

E.2 ABSOLUTE POSITION: THE BORDER AND THE PREFIX TOKENS

Rotations make attention depend only on the offset between two patches, yet probes read each patch's absolute row and column (Table 9). Two sources could supply it: the image border, where a patch has neighbours on one side only, and the prefix tokens (the class token and, on DINOv3, four registers), which receive no rotation. We test both on the four RoPE models (Figure 12, Table 10). First, we split the row and column probes' test accuracy by the patch's distance to the nearest border along that axis. Second, we block attention from every patch to the prefix tokens in every block, by adding $-\infty$ to those attention logits, and repeat the probes, SSDC and top-1. Probes use a 60/20/20 split with the epoch chosen on validation, 1,000 images.

Absolute position is anchored at the border. At block 4, patches on the edge row or column are classified correctly 95–100% of the time, against 14–56% at the centre, and accuracy falls steadily

with distance from the edge. By block 11 the border advantage remains on every model. Blocking the prefix tokens does not remove this pattern: edge accuracy stays at 72–91% at block 4 while the centre falls further.

The prefix tokens still contribute. Blocking them lowers block-4 SSDC by a third on naver ViT-S and by four-fifths on DINOv3, whose four registers carry most of its early structure, and costs 1.8, 8.0 and 9.5 top-1 points on naver ViT-B, naver ViT-S and reg1-gap. By block 11, SSDC returns to its intact level on the three single-prefix models (0.38–0.46 against 0.36–0.43 intact) but not on DINOv3 (0.54 against 0.85). The border anchors absolute position; the prefix tokens add to it, mostly in the early blocks.

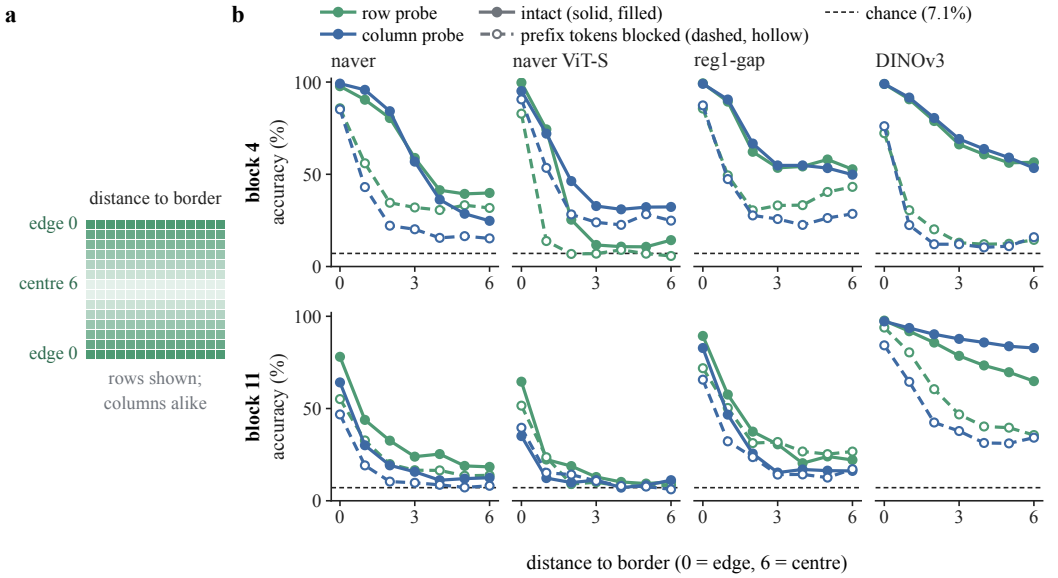

Figure 12: RoPE models read absolute position from the image border. **(a)** Distance to the nearest border along an axis, shown for rows (0: edge, 6: centre); columns are analogous. **(b)** Row (green) and column (blue) probe accuracy against that distance, at block 4 (top) and block 11 (bottom). Solid, filled: intact; dashed, hollow: attention from patches to the class and register tokens blocked in every block. Dashed black: chance (7.1%). 1,000 images.

Table 10: Blocking attention to the prefix tokens. SSDC at blocks 4 and 11, exact-position probe at block 4 (%), and top-1 (%). DINOv3 has no classification head.

	SSDC, block 4		SSDC, block 11		Exact probe, block 4		Top-1	
Model	intact	blocked	intact	blocked	intact	blocked	intact	blocked
naver ViT-B	0.47	0.26	0.40	0.38	28.4	11.5	84.6	82.8
naver ViT-S	0.37	0.25	0.36	0.42	15.6	7.2	81.3	73.3
reg1-gap ViT-B	0.69	0.40	0.43	0.46	31.6	13.5	83.9	74.4
DINOv3 ViT-B	0.95	0.19	0.85	0.54	31.9	5.5	—	—

E.3 Masking windows

Figure 13 shows every window in Section 4.3 and how much structure returns by block 11; Table 11 gives the values. A prefix window 0–k switches rotations off from block 0 through block k and restores them afterwards; a suffix window k–11 switches them off in the last blocks only; a middle window switches them off in an interior range. SSDC at block 11 is read on the input to block 11, so it reflects rotations in blocks 0–10 only; the suffix window 8–11 therefore removes three blocks' rotations, not four.

For windows that remove one axis, Table 12 gives probe retention at block 11 for the 0–5 window on all four models (the values in Figure 4c), and Table 13 varies the window length on naver ViT-B and DINOv3. The two tables use different floors: Table 12 uses the block-4 floor, which is at chance, because that run measured the floor only at block 4, while Table 13 uses the floor at block 11.

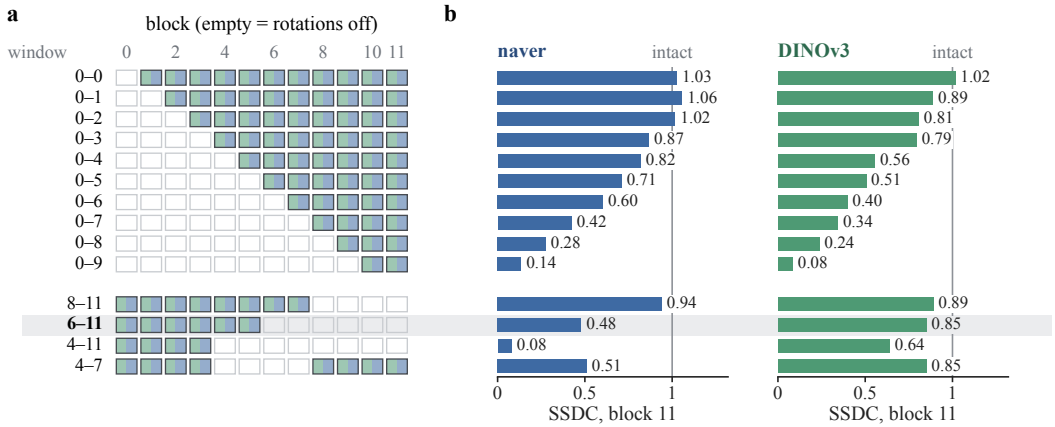

Figure 13: Which blocks need their rotations. (a) Each row is one masking window; empty cells are blocks with rotations off, and filled cells show the row (green) and column (blue) rotations that stay on. **(b)** Normalised SSDC at block 11 (0: floor, 1: intact) for naver ViT-B (left) and DINOv3 (right). naver ViT-B depends on its later blocks (removing 6–11 costs more than 0–5); DINOv3 on its earlier ones.

Table 11: Normalised SSDC at block 11 under each masking window (0: floor, 1: intact).

Rotations off	naver ViT-B	DINOv3 ViT-B	Rotations off	naver ViT-B	DINOv3 ViT-B
0–0	1.03	1.02	0–6	0.60	0.40
0–1	1.06	0.89	0–7	0.42	0.34
0–2	1.02	0.81	0–8	0.28	0.24
0–3	0.87	0.79	0–9	0.13	0.08
0–4	0.82	0.56	*suffix and middle*		
0–5	0.71	0.50	8–11	0.94	0.89
			6–11	0.48	0.85
			4–11	0.08	0.64
			4–7	0.51	0.85

Table 12: One axis off in blocks 0–5, then restored. Probe retention at block 11 (0: floor, 1: intact) for the removed axis and the other axis. Bold: the two models where the other axis ends clearly below intact.

	Row planes off		Column planes off	
Model	row (removed)	column (other)	column (removed)	row (other)
naver ViT-B	0.47	1.04	0.22	1.08
reg1-gap ViT-B	0.30	1.04	0.24	1.13
DINOv3 ViT-B	0.45	**0.79**	0.43	1.05
naver ViT-S	0.44	1.17	0.21	**0.80**

Table 13: Longer windows leave less of the removed axis. Probe retention at block 11 (0: floor at block 11, 1: intact). Inside the window the removed axis is at the floor in every case.

Model	Planes off	Row probe	Column probe
naver ViT-B	row, blocks 0–2	0.75	0.91
	row, blocks 0–5	0.47	1.05
	row, blocks 0–8	0.13	1.15
	column, blocks 0–5	1.06	0.24
DINOv3 ViT-B	row, blocks 0–2	0.65	0.78
	row, blocks 0–5	0.46	0.80
	row, blocks 0–8	0.30	0.81
	column, blocks 0–5	1.05	0.43

F HEAD ATTRIBUTION

This appendix supports Section 4.3. Throughout, a "head" is one head index across the blocks named.

F.1 AXIS SPECIALISATION

Table 14 gives the numbers behind Figure 4d. Each entry is the probe points lost when one head's rotations stay off in blocks 6–11 during recovery from a one-axis 0–5 window, relative to the same window with every head restored. Because SSDC and probes are read on the input to each block, a head whose rotations are removed from block 6 first affects the reading at block 7, so the cost is averaged over blocks 7–11. The row cost uses the row-removed window and the column cost the column-removed window. The heads are the top four by the SSDC rebuilding score (three on naver ViT-S), and the three-standard-deviation noise bar, measured by repeating the probe, is about 1.2 points.

Table 14: Cost of keeping one head's rotations off during recovery, in probe points (row window / column window), averaged over blocks 7–11. †: below the 1.2-point noise bar.

Model	Head: row cost / column cost			
naver ViT-B	h6: 5.19 / 0.09	h1: 3.78 / 0.25	h2: 1.86 / 0.18	h11: 0.48 / 0.08
DINOv3 ViT-B	h2: 0.93 / 9.26	h9: 0.99 / 5.20	h1: 3.83 / 3.39	h10: 1.85 / 2.76
naver ViT-S	h0: 3.03 / 0.78	h4: 2.02 / 0.28	h2: $0.73^{\dagger}$ / $0.68^{\dagger}$	—
reg1-gap ViT-B	h7: 3.84 / 0.00	h9: 3.08 / 0.19	h3: 0.30 / 2.70	h6: 0.05 / 0.00

On naver ViT-S, head 2, the top rebuilder by SSDC, costs neither axis more than the noise bar, yet it is the costliest head for classification (Appendix H.2). Axis specialisation is therefore clear on naver ViT-B, DINOv3 and reg1-gap, and absent for the top head on naver ViT-S.

F.2 CARRYING VERSUS REBUILDING

We also scored each head's *carrying* importance: the normalised SSDC lost at block 4 when that head index's rotations are off in every block, during ordinary inference. Table 15 compares it with the rebuilding score. The two cannot be compared directly: a head index shares no parameters across blocks, and the carrying score depends only on blocks 0–3 while the rebuilding score depends only on blocks 6–11, so they rank different sets of weights. Reading carrying at block 11 instead changes the ranking: on naver ViT-B the top rebuilder, head 6, becomes the second most important carrier, and the rank correlation between carrying and rebuilding rises from $+0.06$ to $+0.88$; on DINOv3 the top rebuilder, head 9, stays the least important carrier (-0.17 at block 4, $+0.27$ at block 11). The rebuilding scores are also close together: on DINOv3 the top three heads are within 0.004 of each other.

Table 15: Carrying and rebuilding rankings. "Concentration": share of SSDC lost at block 4 when the single most important carrying head is removed during ordinary inference. "Rank as carrier": where the top rebuilder sits in the carrying ranking (1 = most important). ρ: Spearman correlation of the two per-head rankings.

Model	Heads	Concentration	Top rebuilder	Rank as carrier	ρ
naver ViT-B	12	31% (head 2)	head 6	11 of 12	$+0.06$
DINOv3 ViT-B	12	88% (head 2)	head 9	11 of 12	-0.18
naver ViT-S	6	71% (head 4)	head 2	5 of 6	-0.09
reg1-gap ViT-B	12	46% (head 0)	head 7	7 of 12	-0.35

Concentration is not set by head count or scale: the three 12-head models span 31–88%, and the two naver checkpoints, which share a recipe, differ (31% and 71%).

F.3 ATTENTION STATISTICS

One might expect rebuilding heads to look further across the image than carrying heads, but they do the opposite: over blocks 6–11 on 64 images, the top two rebuilders attend nearer and with lower

entropy than the top two carriers on all four models (Table 16), after excluding any head that appears in both lists. The bracketed figure is the difference as a share of the largest head-to-head difference in that model. Each group holds one or two heads, so we report the direction, not a tested difference, and these statistics do not explain what the rebuilding heads compute.

Table 16: Top two rebuilders minus top two carriers, blocks 6–11. Negative: the rebuilding heads attend nearer, or with lower entropy.

Statistic	naver ViT-B	DINOv3 ViT-B	naver ViT-S	reg1-gap ViT-B
Mean distance (patches)	−0.05 (4%)	−0.87 (41%)	−0.19 (18%)	−0.64 (80%)
Entropy (nats)	−0.03 (6%)	−0.30 (44%)	−0.08 (26%)	−0.11 (17%)
Attention to prefix	+0.00	+0.02	+0.02	−0.02

G Comparison: a shared recipe and a shared model

This appendix supports Section 4.4.

G.1 DeiT-III and naver ViT-B

DeiT-III ViT-B (APE) and naver ViT-B (RoPE) are trained with the same recipe and reach the same top-1 on our 5,000 evaluation images (84.58% and 84.64%), so their comparison is not confounded by training. DeiT-III's SSDC is 0.57 entering block 0, rises to 0.84 at blocks 5–6 and falls to 0.38 at block 11; naver's starts at 0.03 and reaches 0.46 (Figure 2). Zeroing DeiT-III's embedding leaves SSDC at the floor (at most 0.011) at every block and top-1 at 23.5%, so, like the other APE models, it does not rebuild position once its only source is gone. Permuting the embedding across patches instead keeps exact position decodable (99.7% at block 2) but removes SSDC and costs 52 top-1 points.

G.2 The hybrid checkpoint

The hybrid ViT-B has both an added embedding and rotations. With every rotation removed, SSDC exceeds the intact curve in blocks 1–4, stays at 96–97% of it through block 6, and then collapses, reaching 0.228 at block 8 against 0.557 intact (Figure 14). With the embedding removed instead, SSDC starts near zero and climbs with depth, passing the rotations-off curve at block 8 (0.354 against 0.228). The embedding supplies the hybrid's early structure and the rotations its late structure, the same depth signatures as the separate APE and RoPE models.

Position does not need the rotations to be readable, but the classifier needs them. At block 4, the exact-position probe reads 85.1% intact, 88.6% with every rotation off, 4.1% with the embedding removed, and 0.8% with both removed (chance 0.5%). Top-1 is 80.5% intact and 37.4% with the rotations off. On this checkpoint the top rebuilder costs no more accuracy than ordinary heads, so it is not in Figure 5b, and we did not run the all-heads test on it. All hybrid values are raw: with the rotations off the hybrid stays above its intact SSDC in early blocks, so the floor used for normalisation elsewhere does not exist here.

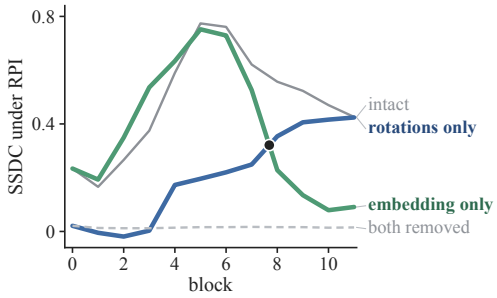

Figure 14: In the hybrid, the embedding builds early structure and the rotations late structure. SSDC under RPI entering each block of the hybrid ViT-B: intact, rotations off, embedding removed, and both removed. 1,000 images.

H CLASSIFICATION

This appendix supports Section 4.5: every APE arm, every RoPE head, a replication from an earlier run, and why that section reports accuracy rather than SSDC.

H.1 APE: EVERY EDIT

Table 17 gives top-1 for every APE edit. Supervised ViT-B and DeiT-III use 5,000 images, AugReg and SAM 1,000 (binomial standard error about 1.3 points); DINO has no classification head. On supervised ViT-B, paired over the same 5,000 images, swapping the features to other patches costs 24.8 ± 0.7 points more than swapping them to the same patch of another image, and zeroing them costs 15.1 ± 0.7 more than zeroing a matched random set, both with the error added back. On SAM the same two gaps are 31.4 ± 1.5 and 13.9 ± 1.3 points over 1,000 images; per-image results for AugReg were not kept, so we give no paired errors there.

Table 17: Top-1 accuracy (%) under each APE edit. Edits act at the residual stream entering block 2, except the embedding rows. Matched random: mean $\pm$ standard deviation over five draws.

Edit	supervised	AugReg	SAM	DeiT-III
Intact	81.16	78.6	81.0	84.58
SAE reconstruction	79.34	78.3	80.5	—
Positional zeroed	50.32	50.9	54.8	—
Uniform random zeroed	78.94	77.4	80.5	—
Matched random zeroed	73.90 ± 0.75	73.8 ± 0.4	78.8 ± 0.6	—
With the SAE error added back				
Positional zeroed	61.52	56.6	65.2	—
Positional mean-ablated	62.78	59.8	65.1	—
Swapped, same patch of another image	80.84	77.9	80.7	—
Swapped, other patch of the same image	56.04	53.7	49.3	—
Matched random zeroed (one draw)	76.60	75.3	79.1	—
Embedding zeroed	24.88	28.3	40.0	23.52
Embedding permuted	28.50	29.7	23.1	32.12

The matched random features cost 2–5 points beyond the reconstruction, far more than the uniformly drawn ones, yet leave spatial structure intact (Table 7), so they measure what removing active but position-free features costs. The same-patch swap costs at most 0.7 points while the other-patch swap costs 25–32, so the classifier depends on which position the features report.

Figure 15 shows the same effect on single images. The pixels are untouched and only the 20 positional features are edited. Moving them to the same patch of another image leaves every prediction intact, while moving them between patches of the same image turns a coffee pot into a water jug and a chiton into velvet. Of the first 64 validation images, the intact model classifies 54 correctly; the same-patch swap keeps all 54, the other-patch swap 35 and zeroing 39.

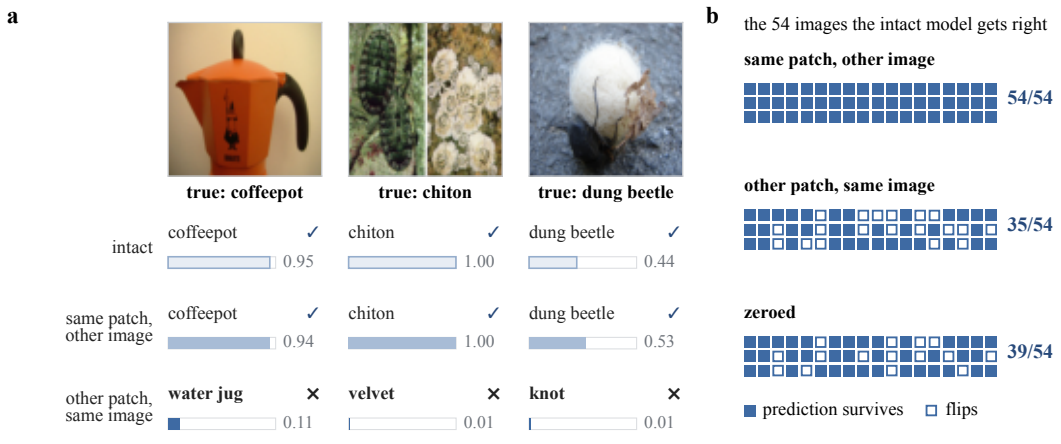

Figure 15: Moving the positional features changes the prediction, although the image does not change. Supervised ViT-B, with its 20 positional features edited at block 2 and the SAE error added back. **(a)** Top-1 prediction and the probability of the true class, intact, with the features swapped with the same patch of another image, and with them swapped between patches of the same image. The examples are the first three validation images that the intact model classifies correctly and the other-patch swap does not, skipping images that show people. **(b)** The 54 of the first 64 validation images that the intact model classifies correctly: filled if the prediction survives the edit.

H.2 RoPE: every head

Table 18: Top-1 cost (points) of removing one head's rotations in blocks 6–11. Bold: the top rebuilder, fixed in advance by its SSDC score. Reference rows give top-1 (%). "Shuffled" replaces rotations with those of a fixed random permutation of patch positions instead of the identity (two seeds for all blocks).

	naver ViT-B		reg1-gap ViT-B		naver ViT-S	
Head	window	intact	window	intact	window	intact
0	1.34	0.12	0.70	0.20	3.70	0.50
1	3.38	0.44	0.24	0.24	1.00	0.96
2	1.64	0.30	3.94	0.22	**7.62**	**1.18**
3	−0.14	0.52	4.16	0.28	4.88	1.40
4	2.18	0.54	−1.56	0.18	6.48	1.08
5	2.36	0.14	0.98	0.16	0.34	1.78
6	**4.30**	**0.50**	1.52	0.24		
7	−0.10	0.12	**5.74**	**0.42**		
8	0.36	0.36	−0.92	−0.04		
9	1.82	0.42	4.90	0.20		
10	−1.66	0.36	−1.24	0.78		
11	2.30	0.44	−0.46	0.10		
Top rebuilder, shuffled	3.52	0.42	5.86	0.40	6.22	1.02
Reference top-1 (%)						
Intact	84.64		83.88		81.32	
Window (0–5 off)	70.52		56.28		48.68	
Window, shuffled	67.16		57.36		53.38	
All rotations off	38.84		29.50		24.04	
All shuffled (two seeds)	34.20 / 33.40		22.58 / 22.74		20.78 / 20.72	

Table 18 gives, for every head, the top-1 cost of removing its rotations in blocks 6–11, either after the 0–5 window ("window") or in the unmodified model ("intact"). All runs use 5,000 images at 224×224; paired standard errors of the window costs are 0.20–0.36 points on naver ViT-B and 0.37–0.46 on naver ViT-S. Per-image results for reg1-gap were overwritten during a rerun, so it has no paired errors; its heads' window costs have a standard deviation of 2.6 points.

The top rebuilder is the costliest head in the window on all three models, and in the intact model every head costs at most 0.8 points on the two ViT-B models and 1.8 on naver ViT-S. Replacing the

rotations with shuffled ones costs about as much as removing them, both for all heads and for the top rebuilder alone, so the costs come from missing position rather than from attention receiving rotations it never saw in training.

H.3 Replication and higher resolution

An earlier run of the window test, on a different set of 5,000 images and with three random heads per model, gives the same top-rebuilder costs: 4.40, 5.82 and 8.76 points on naver ViT-B, reg1-gap and naver ViT-S, against 4.30, 5.74 and 7.62 here. Its absolute accuracies are 2–6 points lower on every arm because it used a 256×256 repack of the validation set (Appendix I). That run also covered 320×320 input, where the top rebuilder costs 7.84, 6.08 and 7.80 points after the window.

At 320×320, SSDC loses sensitivity. Baseline SSDC at the last block falls by about a third on the three models trained at 224 while top-1 barely moves, and by 4% on reg1-gap, which was trained at 256. On naver ViT-B at 320, a 1,000-image run with the top rebuilder (head 6) and a control head (head 0) gives top-1 losses of 6.95 and 3.13 points, while their SSDC costs are 0.005 and 0.003, a gap of 0.002. SSDC does not separate them; accuracy does. reg1-gap and naver ViT-S rank their heads the same way under both measures at both resolutions, so this is SSDC losing sensitivity in one setting rather than a general dissociation, and it is why Section 4.5 uses accuracy.

I Reproducibility: code, data, compute and protocol

Code. The code, with a script per experiment and the result files (JSON with per-block SSDC, probe accuracies and top-1, plus per-image correctness for the accuracy runs) from which every table and figure is generated, is available at `https://anonymous.4open.science/r/positional-encodings-spatial-structure-A56B/README.md`.

Models and data. Every checkpoint is public (Table 1). Images are the ImageNet-1k validation split, streamed from the HuggingFace hub. The rotation-masking runs (Section 4.3; Appendices E–F) use a 256×256 repack (`benjamin-paine/imagenet-1k-256x256`); the control runs and the patching experiments use the original images (`ILSVRC/imagenet-1k`). Each experiment takes the first N images of the stream in a fixed order, with fixed seeds for RPI permutations and random feature draws. The SAE weights and the lists of selected positional features will be released with the code.

Compute. The experiments ran on four machines (Table 19). Everything runs in float32 with `timm` 1.0.29; PyTorch 2.10 for the earlier runs and 2.14 for the control runs.

Table 19: Hardware per group of experiments.

Experiments	Hardware
SAE training, feature screen, APE dose-response, axis and depth probes (§4.2; App. D.1–D.4, D.6)	1× NVIDIA T4 (16 GB)
Patching, zero-ablation and alignment controls (App. C)	1× NVIDIA RTX A4000 (16 GB)
Rotation masking, windows, heads, first accuracy run (§4.3; App. E, F)	1× NVIDIA H100 (80 GB)
Control runs: APE controls, border, DeiT-III, all-heads accuracy (App. D.5, E.2, G.1, H)	2× NVIDIA H100 (80 GB)

Sample sizes. SSDC and probes use 1,000 ImageNet-1k validation images unless stated (10,000 for the supervised ViT-B dose-response). The patching experiments use 512 images with two seeds. Top-1 uses 5,000 images for supervised ViT-B, DeiT-III and every RoPE model, and 1,000 for AugReg and SAM, whose binomial standard error is about 1.3 points.

Probes. Linear probes use AdamW (Loshchilov & Hutter, 2019) at learning rate 5×10^{-3} for 20 passes with batch size 512, on standardised features, with the same settings for every arm. The earlier runs split images 80/20 and report the best test accuracy over epochs. The control runs split

images 60/20/20 and choose the epoch on the validation split. Across the more than 400 probes in the control runs, the old protocol would have reported 0.1 points more on average and at most 0.7, so it does not change any comparison. The APE axis probes are ridge regressions on the normalised row and column coordinates, split 600/200/200 by image.

Floors and normalisation. On the four RoPE-only models the floor sits at 0.008–0.029 SSDC at every block, against intact values of 0.37–0.95. Block 0 comes before any rotation, so its intact value equals its floor and normalised values are undefined there. The APE floor zeroes the added embedding (SSDC at most 0.021 at any block). Hybrid results stay raw (Appendix G.2).

Rotation masking. Each ViT-B RoPE head pairs its 64 query and key dimensions into 32 rotation planes, 16 per axis, assigned to axes at runtime from each model's frequency table. Setting a plane to the identity (sine 0, cosine 1 in the model's cached tables) is exact and reversible and can target one axis, block range or head. Per-head masking uses the fact that the cached rotation tensor has no head dimension: broadcasting a per-head tensor in its place applies the identity to the chosen heads only. We checked this against the source code of both attention classes these checkpoints use.

Implementation checks. Every experiment runs a set of checks before its main loop: that the planes partition correctly, that hooks fire the expected number of times, that readouts sit at the intended block, and that a no-op mask gives a bit-identical forward pass. Of 318 checks in the earlier runs, 313 passed. The five failures are reported where they apply: three record that the hybrid's rotations-off SSDC sits above its intact value (Appendix G.2), which comes from its second positional signal, and two record DINOv3's cross-axis cost (Appendix E). All 15 checks passed in each of the three control runs on RoPE models. On the APE side, adding the SAE error back with no edit reproduces the intact model's SSDC and top-1 exactly.

Missing per-image results. Paired standard errors use per-image correctness saved for every arm. For reg1-gap, a rerun after a machine restart overwrote the per-image files of the earlier arms, so only their means are available; we give the spread across heads instead.