INTUITION VECTORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large self-supervised vision models learn representations that support scene segmentation and the semantic decomposition of physical objects. We ask whether their representational geometry supports transfer to visual reasoning problems without any task-specific fine-tuning. We hypothesized that relational representations may bridge perception and abstract reasoning by encoding similarities and transformations among visual inputs such that an intuitive, implicit form of reasoning may be performed via latent vector arithmetic. Specifically, we examine DINOv3, MAE, and random pixel projections on abstract and naturalistic Bongard problems, ARC-AGI-1 and ARC-AGI-2, and novel ARC-GEN instances. On both Bongard benchmarks, the accuracy of a simple nearest-centroid readout of frozen visual embeddings is within four percentage points of the task-specific baselines reported with the original benchmarks. In ARC, latent difference vectors summarizing demonstration input–output transformations, which we refer to as *intuition vectors*, show greater alignment with test vectors from the same task, whereas those from unrelated tasks are near orthogonal. This latent geometry is operational: transporting a query along its intuition vector consistently improves exact-output retrieval, reaching 70.7% on ARC-AGI-2 evaluation. Across 397,000 ARC-GEN instances from 794 tasks, single-pair intuition vectors identify the generating task with approximately 87% leave-one-out accuracy. These findings suggest that latent vector arithmetic over frozen visual representations supports implicit rule inference across varied problem domains without a generative model component, indicating that inferring an abstract transformation and generating its instance-specific consequence may be separable capacities.

1 INTRODUCTION

Large pretrained models acquire internal representations with structure that extends beyond their direct training objectives. In language models, semantic concepts exhibit systematic geometric organization (Park et al., 2024; Marks & Tegmark, 2024; Arditi et al., 2024; Engels et al., 2025; Park et al., 2025). Vision models similarly contain structured representations of visual features and concepts (Haas et al., 2024; Fel et al., 2026b;a; Cohen et al., 2020; Sorscher et al., 2022), while video models encode physics-related structure (Joseph et al., 2026). Congruently, a line of recent work increasingly explores the construction of modular architectures that reuse pretrained representations instead of learning both representation and task-specific computation from scratch (Zhou et al., 2025; Zheng et al., 2025a; Gui et al., 2026; Gabeur et al., 2026).

Abstract reasoning systems, however, are typically designed in a manner that is specific to a task, benchmark, or problem instance. Successful approaches train models from scratch (Hu et al., 2025; Wang et al., 2025), fine-tune pretrained models (Franzen et al., 2024; 2025), introduce specialized architectures (Zheng et al., 2025c; Macfarlane & Bonnet, 2025; Wang et al., 2025), or adapt model parameters to individual problems at test time (Akyürek et al., 2025; Franzen et al., 2024; 2025). Given emerging evidence for relational representations across architectures (Shang et al., 2026) and pretrained models (Lepori et al., 2024), and for the expression of novel concepts through existing representation spaces (Sorscher et al., 2022), this motivates our research question: how far can abstract reasoning be supported by relational structure already present in pretrained representations, without task-specific representation or generation training?

In order to address this question, we study how simple geometric operations in the latent spaces of frozen DINOv3 and MAE encoders can support abstract reasoning. Our approach is to interrogate

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.

three components of inferential processing: abstract concept discrimination, implicit rule representation locally within and globally across tasks, and rule application given a novel input. Different linear vector manipulations accompany each inferential computation. Specifically, nearest-centroid classification probes concept structure in Bongard-LOGO and Bongard-OpenWorld, while input-output embedding differences, which we refer to as intuition vectors, represent abstract rules in ARC. We ask whether these vectors align across examples, operationally guide exact-output retrieval when used to transport a query representation, and preserve task identity across instances. Random pixel projections provide a low-level control, allowing us to distinguish transferable task-level structure from the instance-specific information required to resolve an exact output.

2 Related work

2.1 Emergent representational structure

The plethora of evidence for emergent, informative structure across artificial model representations (Park et al., 2025; Marks & Tegmark, 2024; Arditi et al., 2024; Engels et al., 2025; Haas et al., 2024; Fel et al., 2026c; Cohen et al., 2020; Sorscher et al., 2022; Joseph et al., 2026; Huh et al., 2024) has prompted debate over the extent to which such structure is coherent and functionally meaningful (Hewitt & Liang, 2019; Vafa et al., 2024; Gröger et al., 2026). Here, we take a practical approach, focusing on whether these representations are generally useful beyond the setting in which they were constructed and identified (Mahowald et al., 2024), specifically general abstract reasoning.

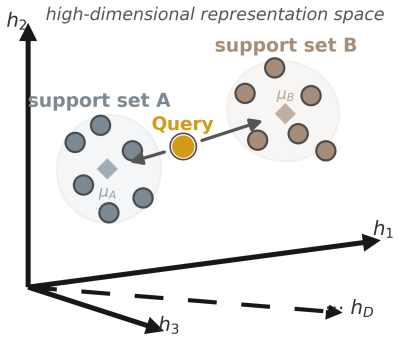
2.2 Bongard problems

Bongard Problems are few-shot visual concept-learning tasks in which a rule distinguishing positive from negative examples must be inferred (Bongard, 1970). Bongard-LOGO uses procedurally generated geometric images to study abstract and compositional reasoning (Nie et al., 2020), while Bongard-OpenWorld extends the setting to natural images and open-vocabulary concepts requiring richer semantic knowledge (Wu et al., 2024). Methods for Bongard-LOGO typically use dedicated few-shot models, classifying queries by distances to support examples or class prototypes (Nie et al., 2020; Raghuraman et al., 2024). We instead operate directly on pretrained representations, using nearest-centroid discrimination without task-specific representation learning.

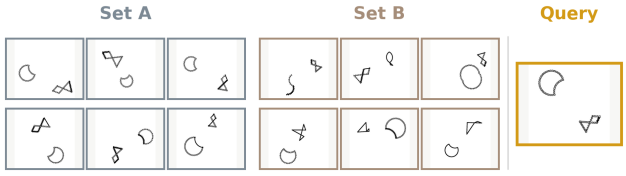
2.3 ARC-AGI 1&2 and ARC-GEN

ARC-AGI 1 and 2 consist of abstract grid-transformation tasks in which a few input-output examples define an abstract rule to be applied to a new input, with performance measured by exact output match (Chollet, 2019; Chollet et al., 2026b). In an effort to extend the available pool of task instances from ARC-1, ARC-GEN (Moffitt, 2025) was introduced. Solution strategies may be broadly classified in several ways (Chollet et al., 2026a). For example, as inductive or transductive methods or hybrids (Li et al., 2024). Inductive methods infer an intermediate task representation, often an executable or latent program, before applying it to the test input (Greenblatt, 2024; Macfarlane & Bonnet, 2025; Hu et al., 2026). Transduction-based methods instead predict the output directly, using ARC-specific architectures (Hu et al., 2025; Wang et al., 2025; Jolicoeur-Martineau, 2025), fine-tuned pretrained models (Franzen et al., 2024; 2025), or task-specific test-time training (Akyürek et al., 2025; Franzen et al., 2025). One transductive strategy based on compression is notable for its lack of any pretraining (Liao & Gu, 2025) however it is ARC-specific and requires test-time task-specific optimization.

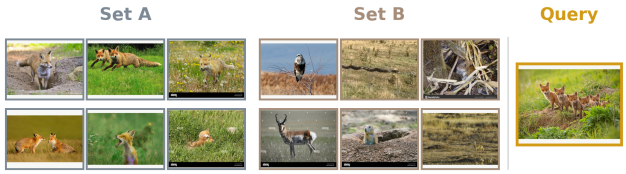
We study a complementary setting: asking whether frozen pretrained representations contain relational structure that reflects transformations demonstrated within a task. Specifically, we examine the geometry of input-output differences within and across tasks, and ask whether this structure can be used to guide exact-output retrieval. Our investigation aims to dissect distinct components of the putative reasoning process thus we focus on large pretrained models without the capacity to generate outputs. ARC requires output generation thus this reasoning benchmark may encourage a conflation of representing and manipulating abstract relational structure with generating the required output. While the present work examines whether these two computational capacities are theoretically and empirically separable, we do not claim an ARC solution and leave the incorporation of a generative component into our approach to future work (Zheng et al., 2025b).



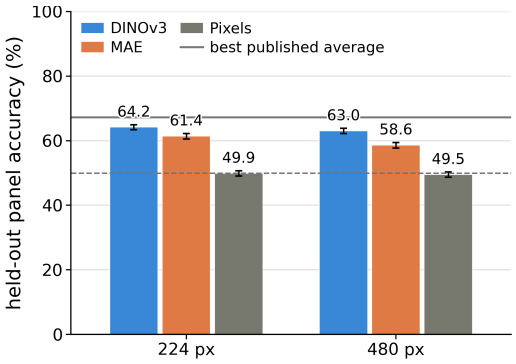
(a) Nearest-centroid readout in representation space.



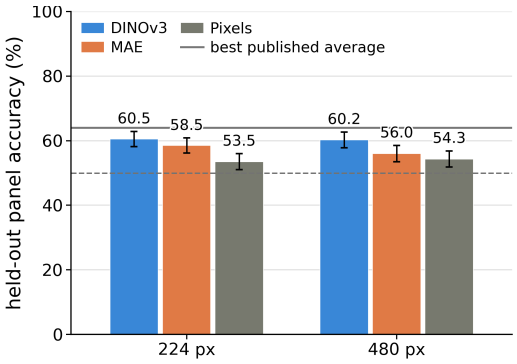
(b) Example Bongard-LOGO problem.



(c) Example Bongard-OpenWorld problem.



(d) Bongard–LOGO, average over four test subsets.



(e) Bongard–OpenWorld, all 200 problems.

Figure 1: Visual concept discrimination in frozen representation spaces on Bongard–LOGO and Bongard-OpenWorld. **(a)** The two support sets define class centroids, and a held-out query is assigned to its nearer centroid. **(b, c)** Representative Bongard-LOGO and Bongard-OpenWorld problems. **(d, e)** Best observed nearest-centroid accuracy for DINOv3, a size-matched MAE, and random-projection pixel controls at 224 and 480 px. For each encoder, dataset, and resolution, the displayed layer maximizes dataset-level evaluation accuracy. LOGO values are equal-weight means across its four disjoint test subsets; OpenWorld values pool 200 problems. Error bars show SEM across problems for encoders and the mean problem-level SEM across ten projections for pixel projections. Gray lines mark external published references (LOGO: 67.2%(Nie et al., 2020); OpenWorld: 64.0% (Wu et al., 2024)); dashed lines mark 50% chance. Per-subtask results and full layer profiles appear in Supp. Fig. S1 and S2, respectively.

3 Methods

3.1 Representations and controls

We evaluated frozen DINOv3-Large/16 ((Siméoni et al., 2025)) and ViT-MAE-Large/16 encoders ((He et al., 2021)). Both models were pretrained using self-supervised learning (SSL) objectives: DINO emphasizes representation invariance through self-distillation without labels, whereas MAE learns representations by reconstructing masked image content. No model parameters were fitted on any benchmark. Images were evaluated at 224 and 480 px, and the class-token representation was recorded after each of the 24 transformer blocks. MAE was run without masking. ARC grids were rendered with the standard ten-color palette, centered on a distinct padding color; the 224-px rendering used a 32×32 canvas with 7-px cells, and the 480-px rendering a 30×30 canvas with 16-px cells. As a low-level control, we applied the encoders' standard channel-wise normalization to each rendered RGB canvas, flattened it, and mapped it to 1024 dimensions (matching the dimensionality of the encoders) using a fixed Gaussian random projection with ten seeds.

3.2 BONGARD CONCEPT DISCRIMINATION

We used the standard Bongard-LOGO test subsets and all 200 Bongard-OpenWorld problems. Each problem contains seven images per class. We held out one query per class, formed centroids from the remaining six images, and averaged the two nearest-centroid decisions to obtain problem-level accuracy (Fig. 1). For each encoder, resolution, and benchmark, we report the layer with the highest benchmark-level mean accuracy.

3.3 ARC INTUITION VECTORS AND OUTPUT RETRIEVAL

For a task with K support input-output pairs $\{(x_i, y_i)\}_{i=1\ldots K}$ and frozen representation E_l at layer l, we define the *intuition vector* (Fig. 2a)

$$v_l = \frac{1}{K}\sum_{i=1}^{K}\left[E_l(y_i) - E_l(x_i)\right] \quad .$$

This intuition vector is proposed to implicitly represent the underlying transformation rule and the construction of the intuition vector as a form of intuitive induction. We evaluated ARC-AGI-1 (400 training and 400 evaluation tasks) and ARC-AGI-2 (1,000 training and 120 evaluation tasks). Transformation alignment was the angle between v_l and the difference $E_l(y_*) - E_l(x_*)$ in the held-out pair (x_*, y_*). Cross-task vectors and cell-shuffled answer grids, each averaged over ten randomizations, served as controls. To test whether the intuition vector was operational, i.e. intuition vector addition could be used to identify the correct answer grid, we translated the query grid representation to $\hat{z} = E_l(x_*) + v_l$ and ranked all output grids in the corresponding dataset by Euclidean distance to $\hat{z}$. Ranking from the untranslated query $E_l(x_*)$ provided the retrieval baseline.

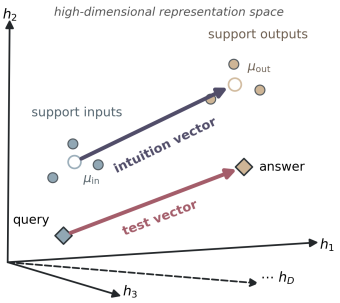
We repeated this retrieval analysis on ARC-GEN using intuition vectors computed from the original ARC training pairs, 79,400 unique held-out generated query-answer pairs from 794 tasks (100 per task), and a shared pool of 368,442 unique output grids. For retrieval, we sampled 100 eligible generated pairs per task after globally excluding pairs matching any original ARC pair. Task-property summaries were computed from the remaining unique generated pairs. Encoder layers were selected by retrieval performance on the original ARC evaluation pairs and then held fixed for ARC-GEN: layer 16 for DINOv3 and MAE at 224 px, 15 for DINOv3 and 18 for MAE at 480 px (zero-indexed). We report top-1 accuracy, the gain over untranslated-query retrieval, and the relative residual $\|E_l(x_*)+v_l-E_l(y_*)\|/\|E_l(x_*)-E_l(y_*)\|$. For task-property analyses, we used Spearman correlations and marginal 95% intervals from 2,000 task-bootstrap samples; for the pixel-advantage analysis reported here, two-sided p values were Bonferroni-adjusted across 40 associations (10 task-property summaries $\times$ four pixel–encoder comparisons).

For the population-level ARC-GEN analysis, we formed intuition vectors for 397,000 generated task instances from 794 tasks (500 distinct pairs per task) and assigned each to the nearest task centroid under leave-one-out evaluation. Task classification accuracy was reported at the same ARC-selected layers used for retrieval.

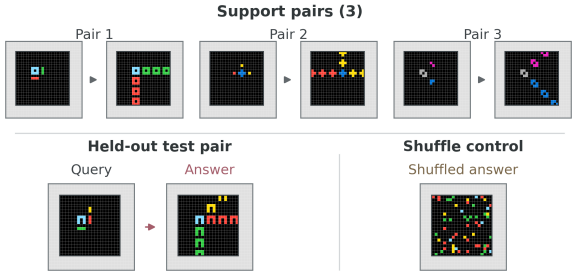
4 RESULTS

4.1 EMBEDDING-SPACE NEIGHBORHOODS DISCRIMINATE ABSTRACT CONCEPTS

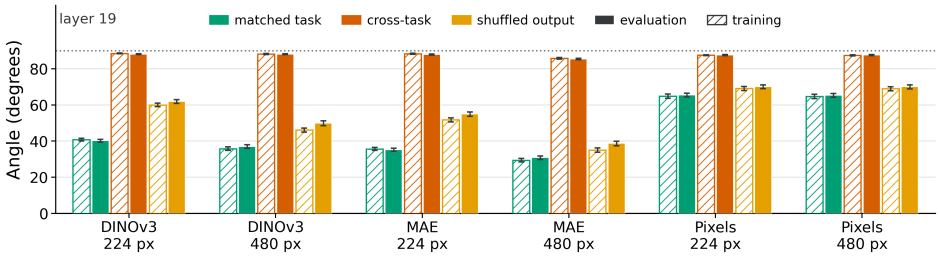
Bongard problems provide a direct test of whether relative position in a frozen embedding space supports abstract classification. For each problem, we form positive and negative support centroids and assign each held-out panel to the nearer centroid (Fig. 1a). We apply this readout independently at every layer of frozen DINOv3 and MAE encoders, without task-specific training or language supervision. At 224 px, DINOv3 gives the strongest observed results: Bongard–LOGO reaches 64.2% accuracy at layer 23 (SEM $= 0.8$ percentage points), only 3.0 points below the 67.2% published reference (Nie et al., 2020), and Bongard-OpenWorld reaches 60.5% at layer 21 (SEM $= 2.4$ points), 3.5 points below its 64.0% reference (Wu et al., 2024) (Fig. 1d, 1e). At the same resolution, MAE reaches 61.4% on LOGO at layer 23 and 58.5% on OpenWorld at layer 10 (For complete layer profiles see supp. Fig. S2). As a low-level control, we apply the same nearest-centroid readout to pixels projected to the encoder dimensionality and average over ten random projections. Pixel ac-



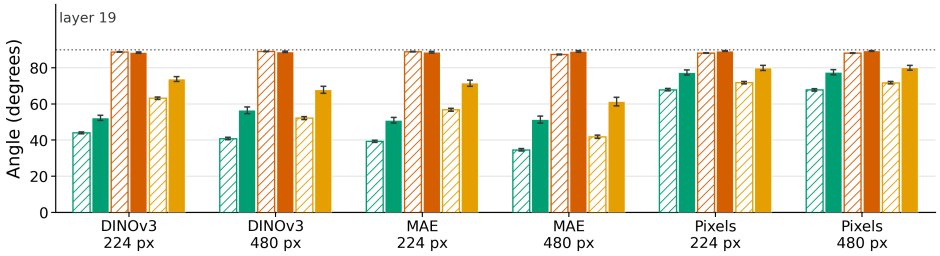
(a) Intuition and test vectors.



(b) Three ARC support pairs ($K = 3$), held-out pair (bottom left), and shuffled-output control (bottom right).



(c) ARC-1: every encoder, resolution and pixel control.



(d) ARC-2: the same comparison.

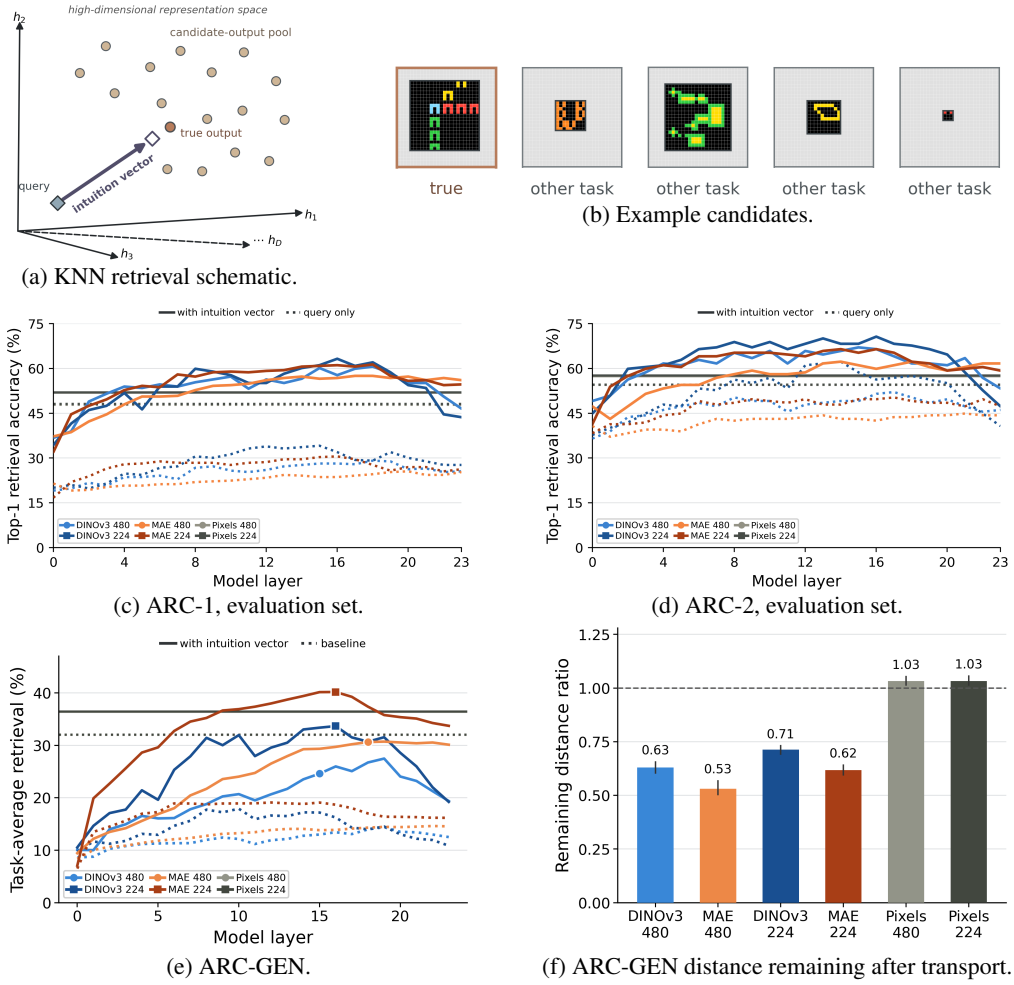
Figure 2: Intuition vector alignment in frozen DINOv3 and MAE representations of ARC-1 and ARC-2. **(a)** The intuition vector connects the support-input and support-output centroids; the test vector connects the held-out query and answer. **(b)** A task with three support pairs, its held-out test pair, and a cell-shuffled answer control, which preserves grid shape and color counts while disrupting spatial structure. **(c,d)** Intuition to test vector angles at layer 19 for both encoders at 224 and 480 px, alongside resolution-matched random-projection pixel controls. Bars compare matched tasks with cross-task and shuffled-output controls, each averaged over 10 randomizations. Filled bars denote evaluation tasks and open hatched bars training tasks; ARC-1 has $n = 400$ tasks per dataset, and ARC-2 has $n = 1{,}000$ training and $n = 120$ evaluation tasks. Bars show mean $\pm$SEM across tasks; the dotted line marks orthogonality. Panel **(c)** provides the key for both panels. Layer 19 was selected globally to maximize separation from both controls; it is within $1.9°$ of every setting's individual optimum. Pixel controls are layer-independent. Full layer-wise results in Supp. Fig. S3, S4.

curacy remains near chance, ranging from 49.5% to 49.9% on LOGO and from 53.5% to 54.3% on OpenWorld across the two resolutions (see subtask breakdown in Supp. Fig. S1).

Together, these results show that pretrained representation geometry exposes abstract concept information across both synthetic drawings and natural images that is beyond pixel similarity.

4.2 Intuition vectors align across examples within tasks

We next ask whether relational geometry can encode transformations rather than only static concepts. For each ARC-AGI task, the intuition vector is the mean support output-input embedding



(a) KNN retrieval schematic.

(b) Example candidates.

(c) ARC-1, evaluation set.

(d) ARC-2, evaluation set.

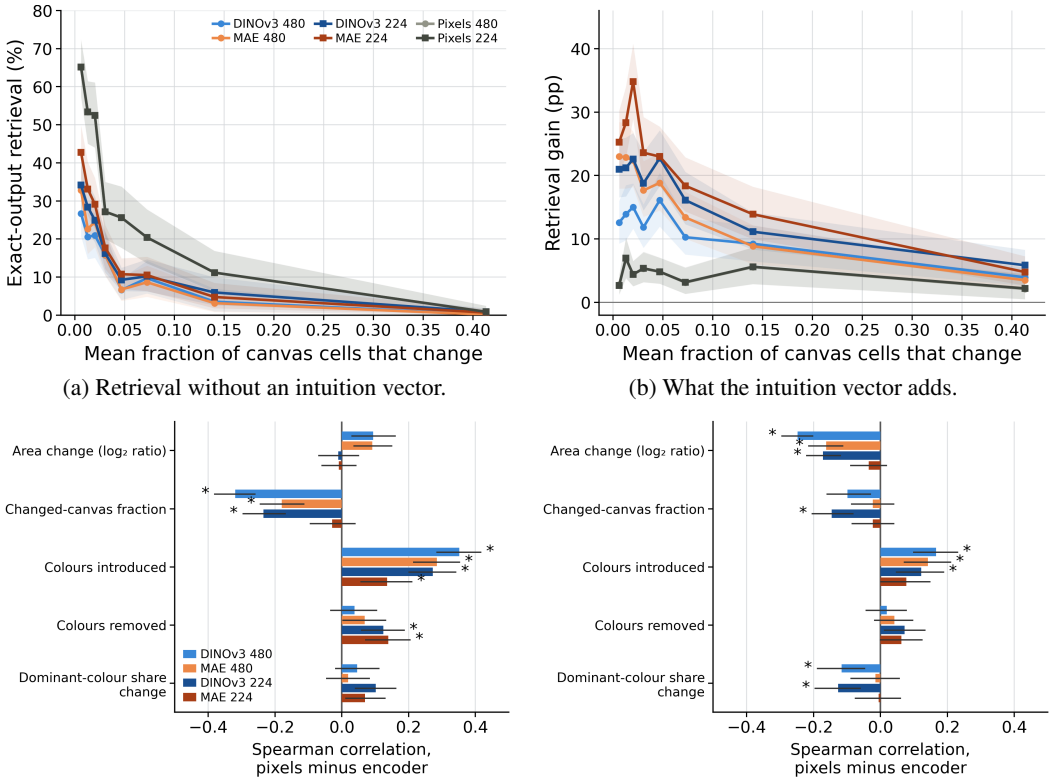
(e) ARC-GEN.

(f) ARC-GEN distance remaining after transport.

Figure 3: ARC output retrieval. **(a)** The inferred intuition vector transports the held-out query representation, and all outputs in the same dataset are ranked by distance to the transported query. **(b)** Candidate pool is the true output (outlined) competing with every other output in that dataset; four alternatives are shown. **(c, d)** Top-1 retrieval accuracy across the 24 transformer blocks on the ARC-1 and ARC-2 evaluation sets. Curves compare DINOv3 and a size-matched MAE at 224 and 480 px; flat lines are the corresponding layer-independent random-projection pixel baselines. Solid lines use the translated query; dotted lines use the original query, isolating the contribution of the intuition vector. Training set results appear in Supp. Fig. S6. **(e)** ARC-GEN retrieval from a 368,442 grid candidate pool,across all 24 layers. Markers identify selected encoder layers. **(f)** ARC-GEN remaining distance between the transported query and answer, relative to the original query-answer distance: the median of task medians with 95% task-bootstrap intervals. Lower is better, 1 denotes no improvement over the original query.

difference, and similarly, the matched test vector is the held-out answer embedding minus its query embedding (Fig. 2a). At layer 19, matched angles range from $29.4°$ to $40.8°$ across ARC-1, from $34.6°$ to $44.0°$ on ARC-2 training, and from $51.0°$ to $56.5°$ on the more challenging ARC-2 evaluation set. Cross-task controls remain approximately orthogonal in every setting ($85.4°$–$89.1°$; Fig. 2), and this task-specific separation persists across model depth (Supplementary Figs. S3 and S4).

Cell-shuffled answers preserve grid dimensions and color counts while disrupting spatial organization (Fig. 2b). Averaged across encoders, resolutions, and datasets, the shuffled-minus-matched angle grows from approximately $4.5°$ at layer 0 to $15°$ at layer 23 on both ARC benchmarks. Pixel controls show weaker alignment (matched angles $64.7°$–$77.5°$) and smaller effects of shuffling ($2.5°$–$4.7°$), while their cross-task angles remain near orthogonal ($87.5°$–$89.3°$; supp. Fig. S5).



(a) Retrieval without an intuition vector.

(b) What the intuition vector adds.

(c) Average task properties.

(d) Variation between instances.

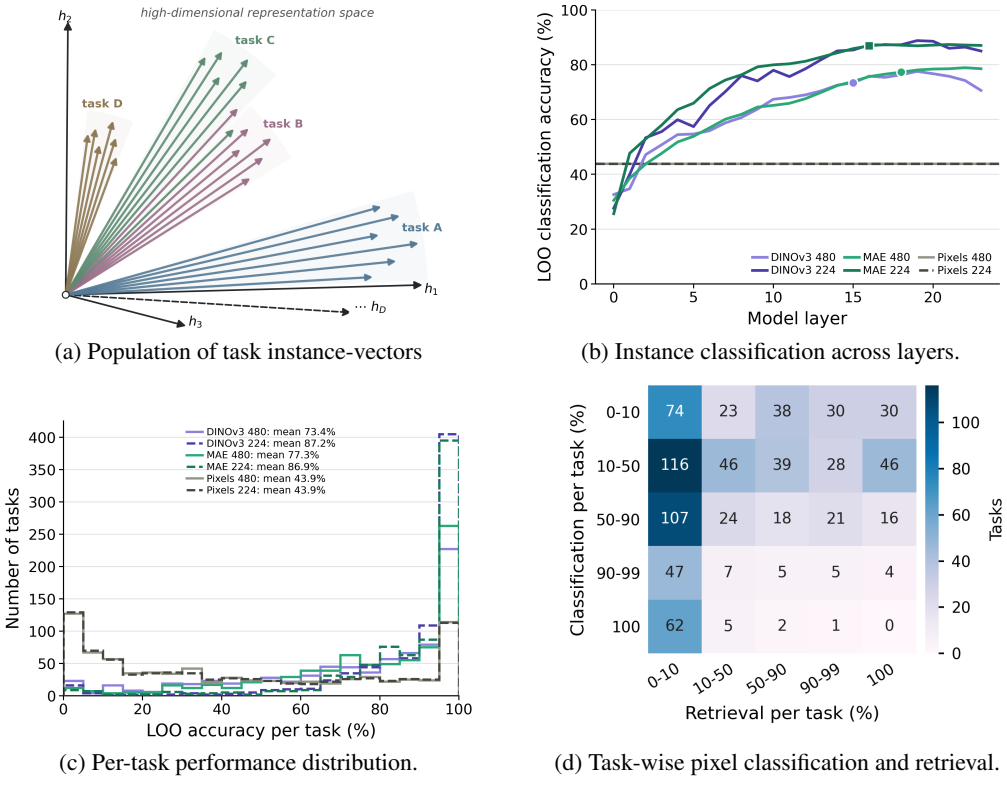
Figure 4: Task properties associated with intuition-vector gain and pixel advantage on ARC-GEN. All panels analyse 794 tasks with 100 generated queries each; encoder layers were selected on original ARC queries and held fixed. **(a,b)** Baseline retrieval (without the intuition vector) and its gain from adding the vector, in percentage points, against the fraction of canvas cells that change. Eight shared, fixed, near-equal-count bins use predictors computed from separate clean instances; shading shows marginal 95% task-bootstrap intervals. **(c,d)** Spearman correlations between task properties and transported pixel advantage (pixel minus resolution-matched encoder retrieval); positive values favour pixels. Panel **(c)** uses task-level means and panel **(d)** between-instance variation. Whiskers show marginal 95% task-bootstrap intervals, and $*$ denotes Bonferroni-adjusted two-sided Spearman $p < .05$ across the 40 correlations (10 task-property summaries $\times$ four pixel-encoder comparisons).

Thus, pre-trained intuition vectors align with test transformations from the same task, with deeper layers increasingly distinguishing the correct spatial organization from shuffled outputs with the same grid dimensions and color counts.

4.3 Intuition vector transport guides output retrieval

We next test whether intuition vectors can guide output selection rather than merely align with held-out transformations. We add the support-derived intuition vector to the query embedding and rank every output in the same dataset by distance to the transported query (Fig. 3a, 3b). At the best observed DINOv3-224 layer (16), top-1 accuracy rises from 32.0% to 63.2% on ARC-1, a 31.3-point gain among 1,782 candidates, and from 56.3% to 70.7% on ARC-2, a 14.4-point gain among 526 candidates; top-2 accuracy reaches 74.7% and 78.4%, respectively (Fig. 3). The same pattern appears across encoders, resolutions, and training sets (supp. Fig. S6). Surprisingly, random-projection pixels also retrieve relatively well but gain little from transport: at 224 px, accuracy rises only from 48.1% to 52.0% on ARC-1 and from 54.6% to 57.7% on ARC-2, remaining 11.3 and 13.0 points below DINOv3 after transport.

We replicate this transport gain at scale on ARC-GEN, evaluating 79,400 generated queries from 794 tasks against 368,442 candidate grids, with encoder layers selected on the original ARC eval-



(a) Population of task instance-vectors

(b) Instance classification across layers.

(c) Per-task performance distribution.

(d) Task-wise pixel classification and retrieval.

Figure 5: Population-level organization of single instance ARC-GEN difference vectors (794 tasks; 500 instances each; 397,000 total) and its dissociation from exact-output retrieval. DINOv3, MAE, and random-projection pixel controls were evaluated at 224 and 480 px. **(a)** Cross tasks difference vector schematic. **(b)** Leave-one-out nearest-centroid accuracy across model layers. Markers denote layers selected independently by retrieval on the original ARC evaluation pairs; pixel controls are layer-independent.**(c)** Per-task accuracy distributions at these layers; legends report means. **(d)** Per-task retrieval and classification performance correlation.

uation tasks. Intuition-vector transport increases top-1 retrieval by 11.6–21.5 points across encoder configurations, compared with approximately 4.4 points in pixel space (Fig. 3e, supp. Fig. S7); top-2 gains are similarly larger (14.7–24.1 versus approximately 5.1 points). It also reduces the median query-answer distance to 53%–71% of its original value in encoder space, whereas pixel transport leaves approximately 103% of the original distance (Fig. 3f). Thus, pixel similarity can discriminate among existing grids, but relational transport contributes far more in pretrained embedding spaces and remains effective across a much larger set of task instances.

4.4 Pixel projections and intuition vectors support different task regimes

On ARC-GEN, pixel-space baseline retrieval substantially exceeds the encoder baselines. However, transport improves retrieval in the encoder spaces while changing pixel-space performance only modestly. We thus ask: which transformations benefit from intuition-vector transport?

Baseline retrieval is strongest for tasks in the smallest-change bin and declines as transformations alter more of the canvas. Encoder-space transport gains are less concentrated near this minimum: their mean gain often peaks in the subsequent small-to-moderate-change bins and remains larger than the pixel-space gain across the full range of transformations (Fig. 4a, 4b). Although these gains eventually decline for the most extensive transformations, this pattern suggests that intuition vectors do more than reinforce the low-level similarity already available from the query. To characterize this difference directly, we correlated task properties with the transported pixel advantage, defined as pixel minus resolution-matched encoder retrieval. Mean changed-canvas fraction predicted a smaller pixel advantage in three of four comparisons ($\rho = -.18$ to $-.32$, Bonferroni-adjusted $p \leq$

1.4×10^{-5}), whereas the number of newly introduced colors predicted a larger pixel advantage in all four ($\rho = .14$–$.35$, adjusted $p \leq .0052$). Greater between-instance variation in the log output-to-input area ratio also predicted a smaller pixel advantage in three of four comparisons ($\rho = -.16$ to $-.25$, adjusted $p \leq 1.7 \times 10^{-4}$; Fig. 4c, 4d). As a consistency check, when ARC answers competed only with 99 partially recolored copies of themselves, pixel retrieval fell 11–22 points below intermediate DINOv3 layers (480 px) on ARC-1 and ARC-2 evaluation from 5% corruption onward (App.A.2, Fig.S8), in line with the correlations above.

Together, these results suggest that pixel retrieval is most competitive when outputs preserve substantial low-level overlap or contain salient palette changes. Encoder-space transport becomes more informative when transformations extend over more of the canvas or vary in scale across instances. Importantly, this is a relative effect: retrieval becomes harder as transformations grow more complex, but the initial advantage of pixel similarity diminishes more rapidly. This pattern is consistent with, although it does not by itself establish, a higher degree of abstraction in the pretrained encoders.

4.5 Task-level identifiability dissociates from exact-output retrieval

The preceding analyses ask whether a support-derived intuition vector aligns with a test transformation and whether transporting a query along that vector improves retrieval of its exact output. Here, we instead ask a population-level question: whether single-pair vectors preserve task identity across many instances and hundreds of competing tasks. We tested this using leave-one-out nearest-centroid classification, whose success requires within-task consistency relative to between-task separation. Across 397,000 difference vectors from 794 ARC-GEN tasks, accuracy reached 73.4%–87.2% for pretrained representations, compared with 43.9% for pixels, and generally increased through model depth (Fig. 5).

Task identification and exact-output retrieval were nevertheless not equivalent. This dissociation was clearest in pixel space: 138 tasks exceeded 90% classification and 181 exceeded 90% transported retrieval, but only 10 achieved both, compared with 31.5 expected under independence (observed/expected $= 0.32$, 95% CI $[0.15, 0.53]$). The two groups differed in area change, changed-canvas fraction, and color removal (supp. Fig. S7c). Thus, task identification and exact-output retrieval rely on different representational strengths. Pixel space preserves low-level similarity that can support exact matching, but its task signatures are less consistent and transport produces only modest gains. Pretrained intuition vectors provide a more transferable transformation signal, supporting stronger task identification and larger transport gains across a broader range of changes. The remaining limitation is execution: translating this task-level relational signal into the precise grid-level changes required by a particular query. Exploratory single-cell analyses in DINOv3 (480 px) are consistent with this account: translation, recoloring, and cell-addition vectors all change with the surrounding grid, retaining little similarity to their empty-grid counterparts when context cells share their color (layer 17, cosine-sim. 0.24, 0.53, and 0.04, respectively), even though operations measured from the same grid remain directionally compositional (App.A.3, Supp. Figs.S9, S10).

5 Conclusion

Together, these analyses show that frozen, off-the-shelf visual representations support abstract reasoning through simple geometric operations, beyond pixel similarity. Intuition vectors align with held-out transformations, form task-identifiable populations, and consistently guide exact-output selection when used to transport query representations. They therefore provide an operational task-level signal, not merely a decodable one. However, exact solution still requires an execution step that translates this relational signal into the precise grid-level output. Representing a transformation and executing its consequence thus appear to be distinct capacities.

5.1 Limitations

The single-cell analyses are exploratory and limited to DINOv3 at 480 px; whether local operations are similarly context-dependent in other encoders, and how such context dependence coexists with stable task-level signatures, remains open. Further, the generality of these findings to other architectures, model sizes, and domains remains for future work.

AI USE STATEMENT

We used generative AI tools to implement methods, create and edit software code, improve scientific figures of our results, create and polish writing. We did not use AI tools to draft the paper, propose or refine hypotheses, develop conceptual frameworks, generate synthetic data sets, or interpret results. Formulating or proving mathematical claims is not applicable to this work. AI-generated code was verified and tested; all reported numbers are regenerated by the released code, whose tests check manuscript values against saved results. We take responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

We further link to our anonymous analysis code: `https://anonymous.4open.science/r/Intuition-Vectors-522D`.

REFERENCES

Ekin Akyürek, Mehul Damani, Adam Zweiger, Linlu Qiu, Han Guo, Jyothish Pari, Yoon Kim, and Jacob Andreas. The surprising effectiveness of test-time training for few-shot learning, 2025. URL https://arxiv.org/abs/2411.07279.

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Panickssery, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction, 2024. URL https://arxiv.org/abs/2406.11717.

Mikhail M. Bongard. *Pattern Recognition*. Spartan Books, New York, 1970.

François Chollet, Mike Knoop, Gregory Kamradt, and Bryan Landers. Arc prize 2025: Technical report, 2026a.

Francois Chollet, Mike Knoop, Gregory Kamradt, Bryan Landers, and Henry Pinkard. Arc-agi-2: A new challenge for frontier ai reasoning systems, 2026b. URL https://arxiv.org/abs/2505.11831.

François Chollet. On the measure of intelligence, 2019. URL https://arxiv.org/abs/1911.01547.

Uri Cohen, SueYeon Chung, Daniel D. Lee, and Haim Sompolinsky. Separability and geometry of object manifolds in deep neural networks. *Nature Communications*, 11(1):746, February 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-14578-5. URL https://www.nature.com/articles/s41467-020-14578-5.

Joshua Engels, Eric J. Michaud, Isaac Liao, Wes Gurnee, and Max Tegmark. Not all language model features are one-dimensionally linear, 2025. URL https://arxiv.org/abs/2405.14860.

Thomas Fel, Matthew Kowal, Mozes Jacobs, Dron Hazra, Usha Bhalla, Lee Sharkey, Lucius Bushnaq, Satchel Grant, Tal Haklay, Thomas Icard, Can Rager, Michael Pearce, Daniel Wurgaft, Aiden Swann, Fenil Doshi, Siddharth Boppana, Curt Tigges, Nick Cammarata, Thomas Serre, Vasudev Shyam, Owen Lewis, Thomas McGrath, Jack Merullo, Ekdeep Singh Lubana, and Atticus Geiger. Structuring sparsity: Block-sparse featurizers capture visual concept manifolds, 2026a. URL https://arxiv.org/abs/2606.25234.

Thomas Fel, Binxu Wang, Michael A. Lepori, Matthew Kowal, Andrew Lee, Randall Balestriero, Sonia Joseph, Ekdeep S. Lubana, Talia Konkle, Demba Ba, and Martin Wattenberg. Into the rabbit hull: From task-relevant concepts in dino to minkowski geometry, 2026b. URL https://arxiv.org/abs/2510.08638.

Thomas Fel, Binxu Wang, Michael A. Lepori, Matthew Kowal, Andrew Lee, Randall Balestriero, Sonia Joseph, Ekdeep S. Lubana, Talia Konkle, Demba Ba, and Martin Wattenberg. Into the Rabbit Hull: From Task-Relevant Concepts in DINO to Minkowski Geometry, May 2026c. URL http://arxiv.org/abs/2510.08638. arXiv:2510.08638 [cs.CV].

Daniel Franzen, Jan Disselhoff, and David Hartmann. The llm architect: Solving arc-agi is a matter of perspective. Technical report, ARC Prize 2024, 2024. URL https://da-fr.github.io/arc-prize-2024/the_architects.pdf.

Daniel Franzen, Jan Disselhoff, and David Hartmann. Product of experts with llms: Boosting performance on arc is a matter of perspective, 2025. URL https://arxiv.org/abs/2505.07859.

Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T. Barron, Kyle Genova, Nithish Kannen, Sherry Ben, Yandong Li, Mandy Guo, Suhas Yogin, Yiming Gu, Huizhong Chen, Oliver Wang, Saining Xie, Howard Zhou, Kaiming He, Thomas Funkhouser, Jean-Baptiste Alayrac, and Radu Soricut. Image generators are generalist vision learners, 2026. URL https://arxiv.org/abs/2604.20329.

Ryan Greenblatt. Getting 50% (sota) on arc-agi with gpt-4o. Redwood Research Blog, June 2024. URL https://www.redwoodresearch.org/blog/getting-50-sota-on-arc-agi-with-gpt. Published June 17, 2024.

Fabian Gröger, Shuo Wen, and Maria Brbić. Revisiting the platonic representation hypothesis: An aristotelian view. *arXiv preprint arXiv:2602.14486*, 2026.

Ming Gui, Johannes Schusterbauer, Timy Phan, Felix Krause, Josh Susskind, Miguel Angel Bautista, and Björn Ommer. Adapting self-supervised representations as a latent space for efficient generation, 2026. URL https://arxiv.org/abs/2510.14630.

René Haas, Inbar Huberman-Spiegelglas, Rotem Mulayoff, Stella Graßhof, Sami S. Brandt, and Tomer Michaeli. Discovering interpretable directions in the semantic latent space of diffusion models, 2024. URL https://arxiv.org/abs/2303.11073.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners, 2021. URL https://arxiv.org/abs/2111.06377.

John Hewitt and Percy Liang. Designing and interpreting probes with control tasks. In *Proceedings of EMNLP-IJCNLP*, pp. 2733–2743, 2019.

Keya Hu, Ali Cy, Linlu Qiu, Xiaoman Delores Ding, Runqian Wang, Yeyin Eva Zhu, Jacob Andreas, and Kaiming He. Arc is a vision problem!, 2025. URL https://arxiv.org/abs/2511.14761.

Keya Hu, Ali Cy, Linlu Qiu, Xiaoman Delores Ding, Runqian Wang, Yeyin Eva Zhu, Jacob Andreas, and Kaiming He. Arc is a vision problem! In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2537–2546, 2026.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representation hypothesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 20617–20642, 2024.

Alexia Jolicoeur-Martineau. Less is more: Recursive reasoning with tiny networks, 2025. URL https://arxiv.org/abs/2510.04871.

Sonia Joseph, Quentin Garrido, Randall Balestriero, Matthew Kowal, Thomas Fel, Shahab Bakhtiari, Blake Richards, and Mike Rabbat. Interpreting physics in video world models, 2026. URL https://arxiv.org/abs/2602.07050.

Michael A. Lepori, Alexa R. Tartaglini, Wai Keen Vong, Thomas Serre, Brenden M. Lake, and Ellie Pavlick. Beyond the doors of perception: Vision transformers represent relations between objects, 2024. URL https://arxiv.org/abs/2406.15955.

Wen-Ding Li, Keya Hu, Carter Larsen, Yuqing Wu, Simon Alford, Caleb Woo, Spencer M. Dunn, Hao Tang, Michelangelo Naim, Dat Nguyen, Wei-Long Zheng, Zenna Tavares, Yewen Pu, and Kevin Ellis. Combining induction and transduction for abstract reasoning, 2024. URL https://arxiv.org/abs/2411.02272.

Isaac Liao and Albert Gu. Arc-agi without pretraining, 2025.

Matthew V Macfarlane and Clement Bonnet. Searching latent program spaces, 2025. URL https://arxiv.org/abs/2411.08706.

Kyle Mahowald, Anna A. Ivanova, Idan A. Blank, Nancy Kanwisher, Joshua B. Tenenbaum, and Evelina Fedorenko. Dissociating language and thought in large language models. *Trends in Cognitive Sciences*, 28(6):517–540, 2024. doi: 10.1016/j.tics.2024.01.011.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets, 2024. URL https://arxiv.org/abs/2310.06824.

Michael D. Moffitt. Arc-gen: A mimetic procedural benchmark generator for the abstraction and reasoning corpus, 2025. URL https://arxiv.org/abs/2511.00162.

Weili Nie, Zhiding Yu, Lei Mao, Ankit B Patel, Yuke Zhu, and Anima Anandkumar. Bongard-logo: A new benchmark for human-level concept learning and reasoning. *Advances in neural information processing systems*, 33:16468–16480, 2020.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models, 2024. URL https://arxiv.org/abs/2311.03658.

Kiho Park, Yo Joong Choe, Yibo Jiang, and Victor Veitch. The geometry of categorical and hierarchical concepts in large language models, 2025. URL https://arxiv.org/abs/2406.01506.

Nikhil Raghuraman, Adam W. Harley, and Leonidas Guibas. Support-set context matters for bongard problems, 2024. URL https://arxiv.org/abs/2309.03468.

Jiaqi Shang, Gabriel Kreiman, and Haim Sompolinsky. Unraveling the geometry of visual relational reasoning. *Scientific Reports*, 16(1):22146, May 2026. ISSN 2045-2322. doi: 10.1038/s41598-026-47123-3. URL https://doi.org/10.1038/s41598-026-47123-3.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. Dinov3, 2025. URL https://arxiv.org/abs/2508.10104.

Ben Sorscher, Surya Ganguli, and Haim Sompolinsky. Neural representational geometry underlies few-shot concept learning. *Proceedings of the National Academy of Sciences*, 119(43):e2200800119, 2022. doi: 10.1073/pnas.2200800119. URL https://www.pnas.org/doi/abs/10.1073/pnas.2200800119.

Keyon Vafa, Justin Y. Chen, Jon Kleinberg, Sendhil Mullainathan, and Ashesh Rambachan. Evaluating the world model implicit in a generative model. In *Advances in Neural Information Processing Systems*, 2024.

Guan Wang, Jin Li, Yuhao Sun, Xing Chen, Changling Liu, Yue Wu, Meng Lu, Sen Song, and Yasin Abbasi Yadkori. Hierarchical reasoning model, 2025. URL https://arxiv.org/abs/2506.21734.

Rujie Wu, Xiaojian Ma, Zhenliang Zhang, Wei Wang, Qing Li, Song-Chun Zhu, and Yizhou Wang. Bongard-openworld: Few-shot reasoning for free-form visual concepts in the real world. In *International Conference on Learning Representations*, volume 2024, pp. 20688–20718, 2024.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion transformers with representation autoencoders, 2025a. URL https://arxiv.org/abs/2510.11690.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. Diffusion Transformers with Representation Autoencoders, October 2025b. arXiv:2510.11690 [cs.CV].

Mingyang Zheng, Weibing Wan, and Zhijun Fang. Abstract visual reasoning based on algebraic methods. *Scientific Reports*, 15(1):3482, January 2025c. ISSN 2045-2322. doi: 10.1038/s41598-025-86804-3. URL https://doi.org/10.1038/s41598-025-86804-3.

Gaoyue Zhou, Hengkai Pan, Yann LeCun, and Lerrel Pinto. Dino-wm: World models on pre-trained visual features enable zero-shot planning, 2025. URL https://arxiv.org/abs/2411.04983.

A Appendix

A.1 Additional figures

A.2 Near-miss ARC 1&2 retrieval

Near-miss pools probe finer discrimination. Intuition vector calculation and application is identical to ARC1&2 procedure but here, the candidate pool contains the true output and 99 independently recolored variants at each corruption level. The analysis uses corruption fractions $\{0.01, 0.05, 0.10, 0.30, 0.60\}$; $\max(1, \mathrm{round}(\rho HW))$ cells are sampled without replacement and each is replaced uniformly by one of the other nine ARC colors. Corruption occurs before rendering and never changes padding. All test queries are included, regardless of whether whole-pool retrieval succeeds. We evaluated DINOv3 at 480 px (layers 6, 12, 17, and 22) and the pixel control on the identical corrupted grids, averaging the latter over the ten projection seeds; MAE and 224-px renderings were not evaluated in this setting. Retrieval accuracy uses the binomial standard error $\sqrt{\hat{p}(1-\hat{p})/N}$; for pixels, this is combined with the SEM across projection seeds. Consequently, error bars describe query-level sampling variation and do not cluster multiple queries from the same task.

A.3 Low-level compositionality analyses

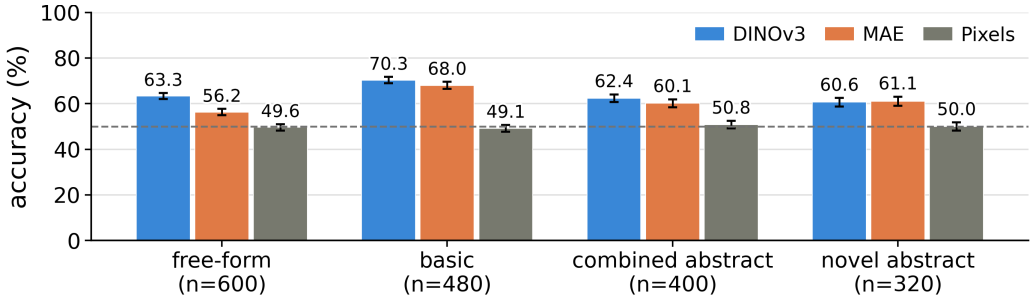
A.3.1 Scope, construction, and controls

These exploratory analyses use the frozen DINOv3-Large/16 encoder at 480 px; MAE and the 224-px rendering are not evaluated. Grids use the same 30×30 canvas as the main analyses. Unless otherwise stated, scalar results are reported at layer 17. Translation, recoloring, and addition vectors are differences between the relevant endpoint embeddings, and composition is evaluated using cosine similarity and the relative residual $\|M - \widehat{M}\|/\|M\|$.

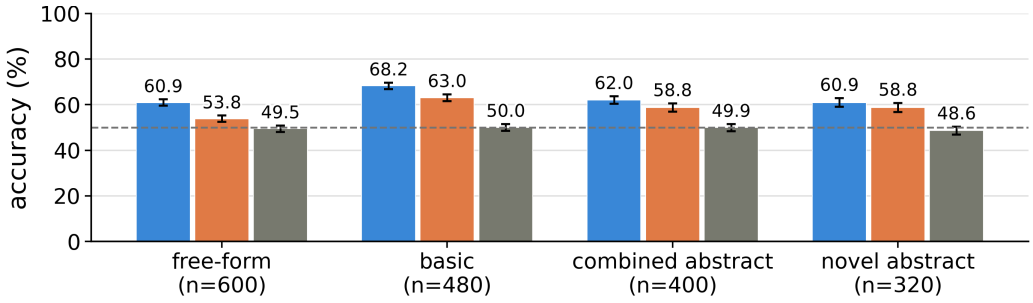
For the context-free analysis, the local cosine matrix uses the 784 source positions at which all four unit translations are valid, four translations of a blue cell, and its eight directed recolorings. For the layerwise composition test, translation primitives are averaged over valid positions and all nine foreground colors for each displacement d, and recoloring primitives are averaged over all positions for each ordered color pair. We test all 796 nonzero displacements satisfying $\|d\|_2 \leq 16$ and all 72 directed recolorings, sampling eight valid source positions without replacement for each combination. This yields $458{,}496$ joint transformations per layer. The shuffled control independently permutes displacement and recoloring assignments while preserving both marginals and is averaged over ten permutations.

Contextual analyses use nine interior anchor positions. For each anchor and occupancy, four independently generated placement sets contribute eight layouts each, giving 32 layouts per anchor and $9 \times 32 = 288$ spatial units. Positions are sampled uniformly without replacement from unoccupied cells, and conditions cross all nine non-black ARC colors. In the addition-transfer analysis, contexts contain one to five cells and all 81 ordered added-cell/context-color pairs are tested, producing 23,328 transitions per occupancy and 116,640 overall. In the shared-context composition analysis, the anchor is the one-cell base B and two to five sampled cells form the jointly added set R. Wrong-location controls average ten permutations of the added sites while retaining the base, occupancy, and colors. Layerwise analyses use a balanced subset of 16 layouts per anchor (144 spatial units).

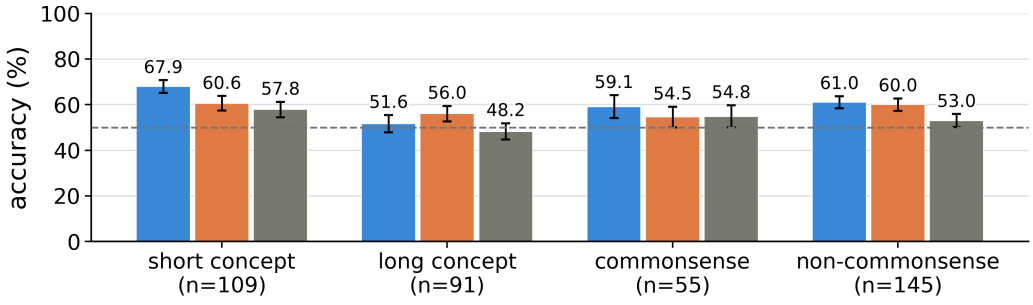
Color conditions sharing a spatial layout are not treated as independent replicates. Values are first averaged over colors within each anchor-layout unit and then across spatial units; 95% confidence



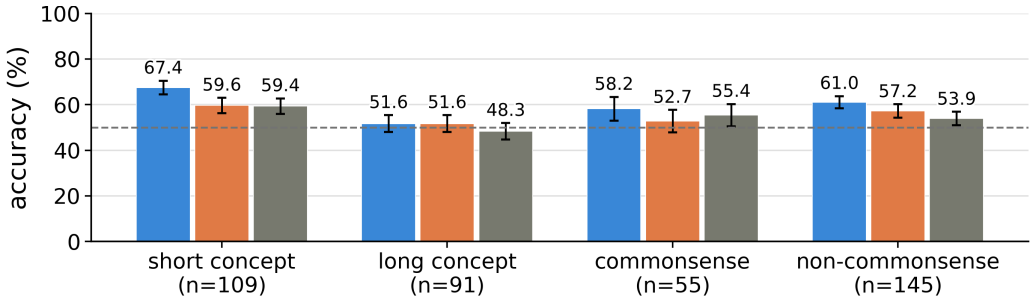
(a) Bongard–LOGO test subsets, 224 px.



(b) Bongard–LOGO test subsets, 480 px.

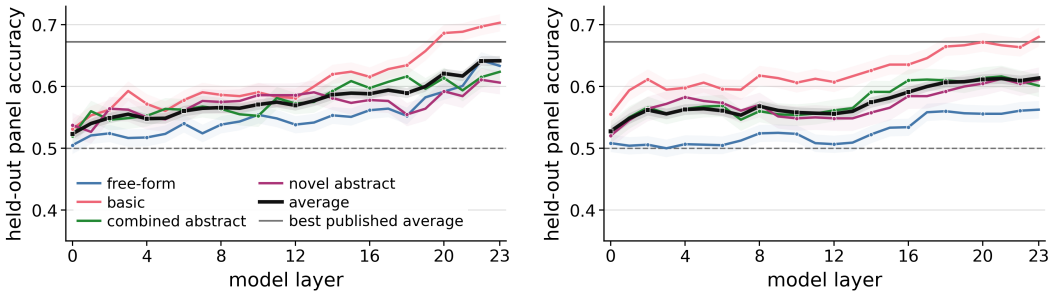


(c) Bongard–OpenWorld subgroups, 224 px.



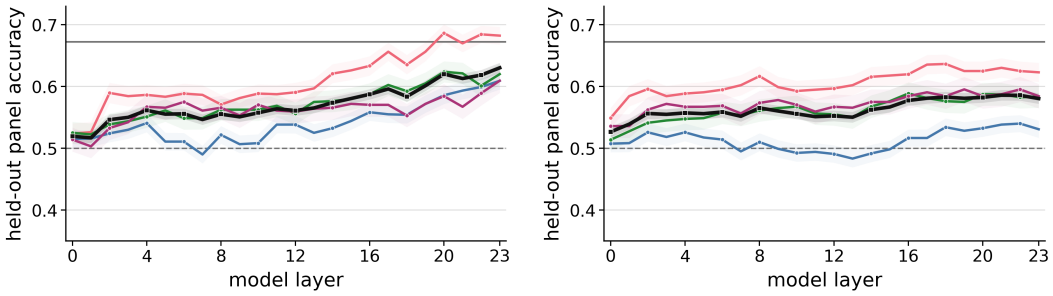
(d) Bongard–OpenWorld subgroups, 480 px.

Figure S1: Per-subtask nearest-centroid accuracy on the Bongard benchmarks, at the settings selected for Figure 1(d,e). **(a,b)** The four disjoint Bongard–LOGO test subsets at 224 and 480 pixels. **(c,d)** The Bongard–OpenWorld subgroups at 224 and 480 pixels. Bars compare DINOv3, a MAE encoder of matched size, and a random-projection pixel control; the legend in **(a)** applies to every panel. The encoder layer is the one that maximizes the dataset-level mean within each dataset and resolution and is then held fixed across all subtasks, so no subtask is selected on its own; the layer choice is a descriptive maximum on the evaluation data and the intervals shown do not include selection uncertainty. Error bars are the SEM across problems for the encoders and, for pixels, a conservative upper bound on the SEM of the mean over ten projection seeds. The dashed line marks 50% chance. No published reference line is drawn, because published values are reported per subset rather than for these groupings. The OpenWorld subgroups form two overlapping partitions of the same 200 problems – concept length (short/long) and commonsense requirement – and must not be averaged together. The count beneath each group is its number of problems; every problem contributes two held-out decisions, one per class.



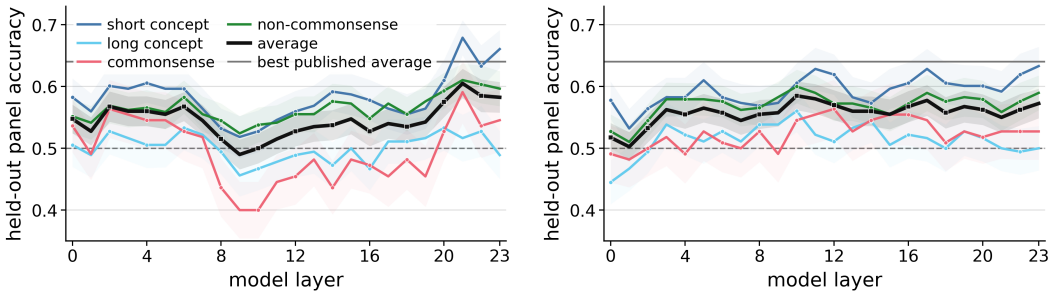
(a) DINOv3, LOGO, 224 px.

(b) MAE, LOGO, 224 px.



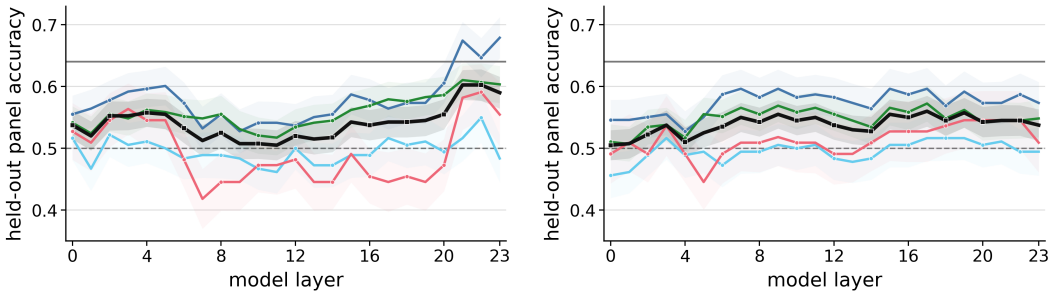
(c) DINOv3, LOGO, 480 px.

(d) MAE, LOGO, 480 px.



(e) DINOv3, OpenWorld, 224 px.

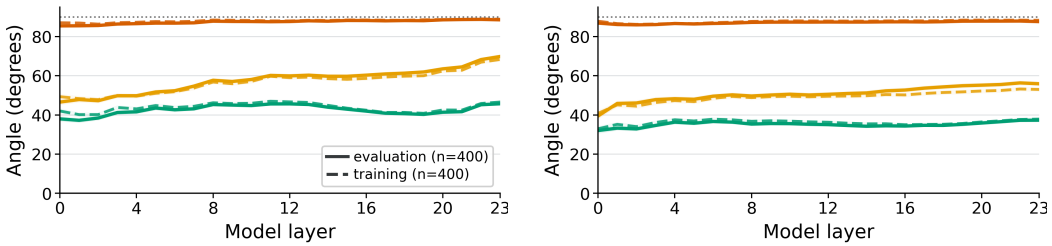
(f) MAE, OpenWorld, 224 px.



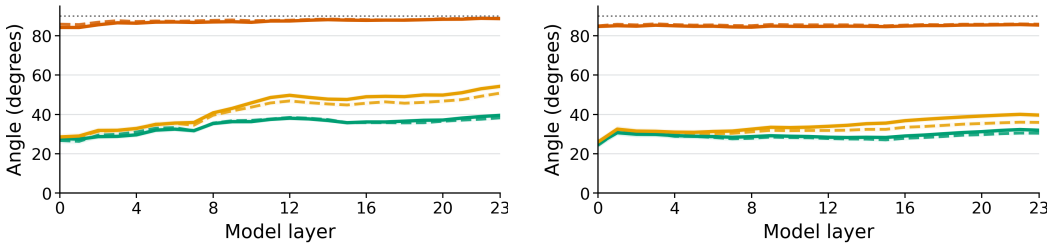
(g) DINOv3, OpenWorld, 480 px.

(h) MAE, OpenWorld, 480 px.

Figure S2: Nearest-centroid accuracy across all 24 transformer blocks, for DINOv3 (left column) and a MAE encoder of matched size (right column). **(a–d)** Bongard–LOGO at 224 and 480 pixels, with one curve per disjoint test subset and a black curve for their equal-weight mean. **(e–h)** Bongard–OpenWorld at 224 and 480 pixels, with one curve per subgroup and a black curve for all 200 problems. The OpenWorld subgroups form two overlapping partitions of the same problems – concept length and commonsense requirement – so their curves are not independent strata. The legend in **(a)** applies to every LOGO panel and the one in **(e)** to every OpenWorld panel. Shading denotes the SEM across problems. The gray line marks the best published average (LOGO 67.2%, the equal-weight mean of the per-subset maxima in (Nie et al., 2020); OpenWorld 64.0% (Wu et al., 2024)) and the dashed line marks 50% chance. All panels share the same axes. The bars in Figure 1(d,e) report the maximum of each black curve; because that layer is chosen on the same evaluation data, the peaks shown here are descriptive maxima rather than independently selected estimates.

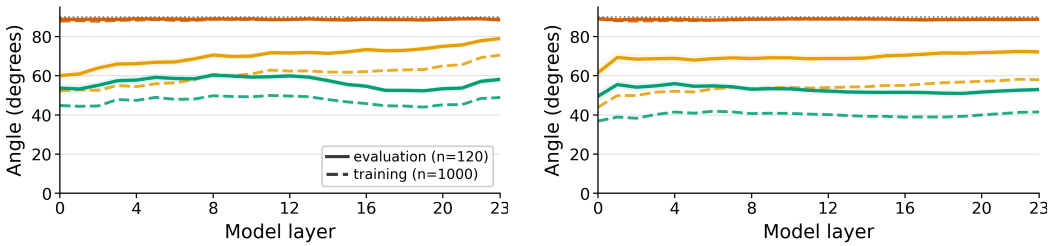


(a) DINOv3, 224 px. (b) MAE, 224 px.

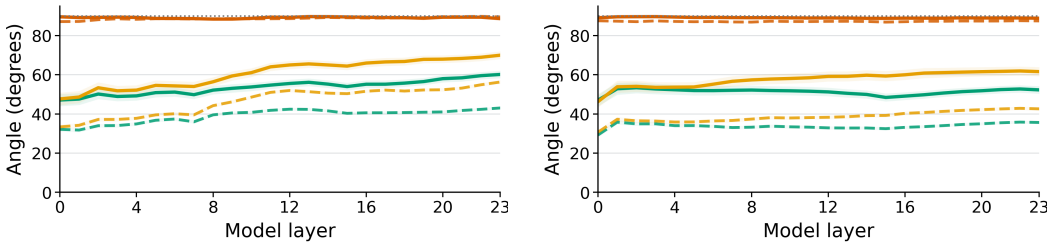


(c) DINOv3, 480 px. (d) MAE, 480 px.

Figure S3: Intuition–test-vector angles across all 24 transformer blocks on ARC-1. Green curves show matched tasks, red curves cross-task controls, and orange curves cell-shuffled answers. Solid and dashed curves denote the evaluation and training sets, respectively; shading shows the SEM across tasks and the dotted line marks 90°. Both sets contain $n = 400$ tasks. Cross-task and shuffled-output controls average ten randomizations. Figure 2(c) reports the corresponding values at the globally selected layer 19.



(a) DINOv3, 224 px. (b) MAE, 224 px.



(c) DINOv3, 480 px. (d) MAE, 480 px.

Figure S4: Intuition–test-vector angles across all 24 transformer blocks on ARC-2, using the same conventions as Figure S3. The training set contains $n = 1{,}000$ tasks and the evaluation set $n = 120$, producing wider evaluation uncertainty. Figure 2(d) reports the corresponding values at the globally selected layer 19.

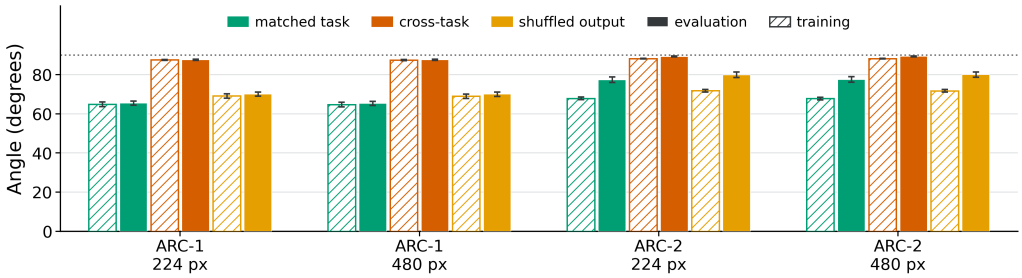
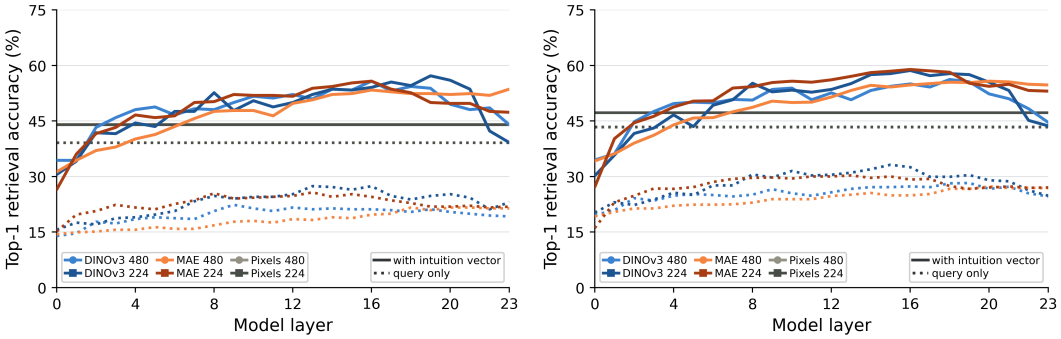


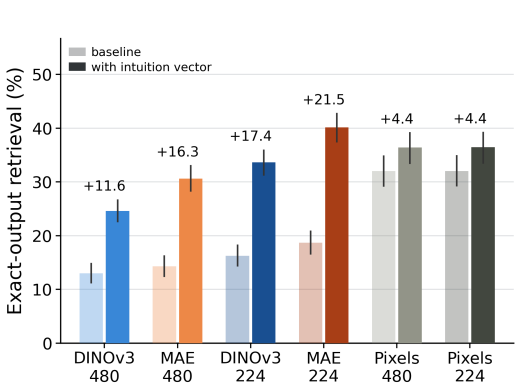
Figure S5: Intuition–test-vector angles in random-projection pixel space for ARC-1 and ARC-2 at 224 and 480 px. Pixel representations have no layer axis and are therefore shown categorically. Green bars show matched tasks, red bars cross-task controls, and orange bars cell-shuffled answers. Filled bars denote evaluation tasks and open hatched bars training tasks. Bars average ten projection seeds; error bars combine the mean within-seed task SEM with uncertainty across projection seeds. Results at 224 and 480 px differ by at most $0.2°$.



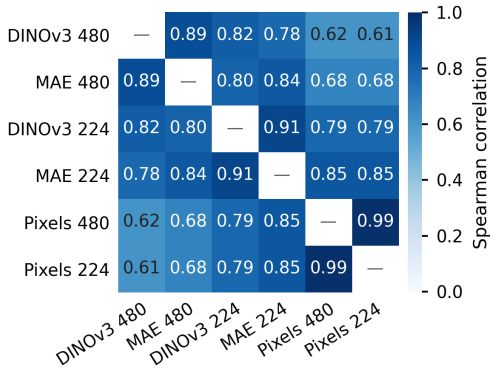
(a) ARC-1, training set.

(b) ARC-2, training set.

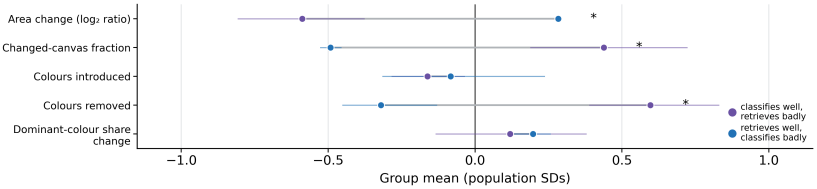
Figure S6: Output retrieval on the ARC training sets, matching Figure 3(c,d) panel for panel. Top-1 accuracy across all 24 transformer blocks, comparing DINOv3 and a MAE encoder of matched size at both rendering resolutions, with the layer-independent random-projection pixel controls drawn as flat lines. Solid curves add the intuition vector to the held-out query and dotted curves rank against the unperturbed query. The training sets are larger than the evaluation set, at 400 against 400 tasks for ARC-1 and 1,000 against 120 for ARC-2, so these curves are the smoother estimate of the same effect; neither set was used to select a layer, an encoder or a resolution.



(a) Retrieval at the selected layer.

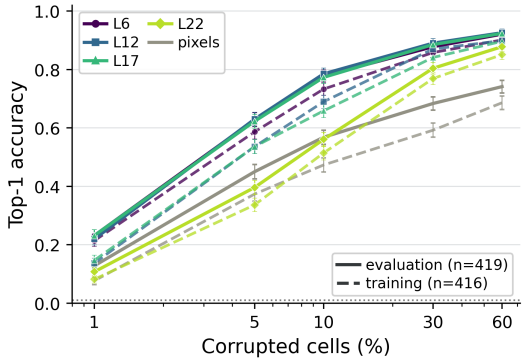


(b) Agreement between cells across tasks.

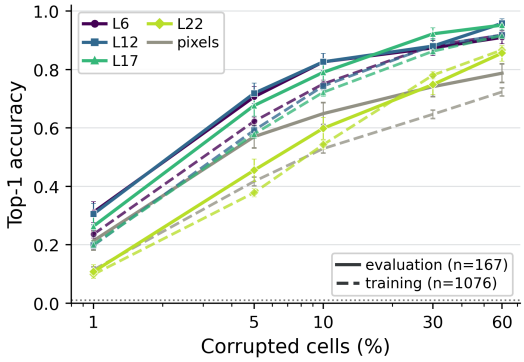


(c) Properties of discordant pixel-space tasks.

Figure S7: Additional ARC-GEN retrieval and classification diagnostics. DINOv3, MAE, and random-projection pixel controls are compared at 224 and 480 px. Retrieval uses 100 generated queries from each of 794 tasks and a 368,442-grid candidate pool; encoder layers were selected on original ARC queries and then held fixed. **(a)** Exact-output retrieval without the intuition vector (pale) and with it (solid) at each encoder's selected layer, alongside the pixel controls. Whiskers are marginal 95% task-bootstrap intervals. **(b)** Pairwise Spearman correlations between representations' task-wise transported retrieval rates. **(c)** Standardized task properties for the two pixel-space extremes in Figure 5d: high classification with low retrieval ($\geq$ 90% and $\leq$ 10%; $n = 109$) and the reverse ($n = 60$). Points and whiskers show group means and 95% bootstrap intervals; asterisks denote Bonferroni-corrected two-sided Mann–Whitney $p < .05$.



(a) Noisy-grid discrimination on ARC-1.



(b) Noisy-grid discrimination on ARC-2.

Figure S8: Nearest-neighbor discrimination after intuition-vector translation on noisy ARC-1 and ARC-2 for DINOv3 (480 px) and 10 random projections of pixel-space. **(a,b)** Top-1 retrieval of the true answer from a 100-grid pool containing the answer and corrupted variants, shown as a function of the fraction of corrupted cells. Colors identify sampled model layers; error bars are binomial SEM estimates. For pixelsbars are mean binomial SEM combined with the SEM across seeds.

intervals are percentile intervals from 2,000 bootstrap resamples of spatial units. The exhaustive context-free analyses are treated as fixed-design descriptive summaries, and their ten shuffles estimate the control mean rather than a sampling interval. For position decoding, a standardized ridge classifier ($\alpha = 100$) predicts the nine-way anchor position from $\Delta_\ell(a; C)$ at layer 17 with five context cells. Four-fold grouped cross-validation holds out one independently generated placement set per fold and keeps all color variants of a layout in the same fold. Confidence intervals use the same spatial-unit bootstrap.

A.3.2 Context-dependent compositionality of single-cell transformations

We begin with the occupancy-preserving operations, translation and recoloring, on otherwise empty synthetic 30×30 ARC-format grids. Let $S(p, c) = E_\ell(\mathrm{grid}(p, c)) - E_\ell(\varnothing)$ represent one cell of color c at position p (Fig. S9a). Differences between these states define individual low-level transformations, while averages over controlled positions or colors define canonical family directions. Unlike intuition vectors, these averages estimate a predefined operation across known realizations rather than infer a latent task from its demonstrations. At layer 17, opposite unit translations are strongly opposed (cosines -0.90 for up–down and -0.85 for left–right), whereas orthogonal translations and translation–recoloring pairs are near orthogonal. Recoloring directions are mutually aligned to varying degrees (Fig. S9b), separating the two transformation families in local geometry.

We then predict individual joint translation–recoloring changes by summing family-average primitives: the translation for the required displacement averaged over valid positions and colors, and the directed recoloring averaged over positions. Across $458{,}496$ balanced transformations per layer, the matched prediction has mean cosine near 0.999 and relative residual 0.03–0.05 in early layers. It remains strong at layer 17 (cosine 0.948, residual 0.298), whereas shuffled displacement–recoloring assignments have near-zero cosine and residuals of approximately 1.4–2.6 (Fig. S9c). Thus, context-free translation and recoloring admit stable, approximately additive primitives.

Having established that canonical family directions predict individual transformations on empty grids, we next examine the individual transformation vectors directly. We compare each transformation measured on an empty grid with the corresponding transformation in a grid containing one to five static cells (Fig. S9d). Context reduces transfer in both families, especially when context colors match the moved cell or a recoloring endpoint. At layer 17 with five context cells, matched-color translation has mean cosine 0.239 with its isolated counterpart, and matched-endpoint recoloring has mean cosine 0.528 (Fig. S9e). The five-cell layerwise analysis shows increasing context sensitivity after approximately layer 12 (Fig. S11a).

Despite this change in the primitive vectors, transformations measured from the same contextual source remain nearly additive. At layer 17, the cosine between a joint transformation and the sum of its contextual translation and recoloring decreases only from 0.9984 with one context cell to 0.9977 with five; the relative residual remains 0.049–0.053. Empty-grid primitives degrade to cosine 0.646 and residual 1.166 at five cells, and an incorrect contextual recoloring also performs worse (Fig. S9f). Across layers, contextual primitives remain essential through the middle and deep blocks (Fig. S11c). Translation and recoloring are therefore locally compositional, but their vectors depend on the state in which they are measured.

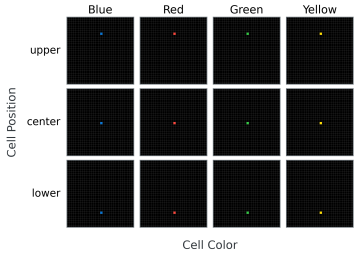
A.3.3 Context dependence of cell addition

We finally examine single-cell addition on occupied synthetic grids; removal reverses the same pair of endpoints and therefore negates the difference vector. For an added cell a and context C, define

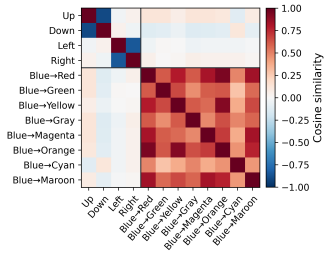
$$\Delta_\ell(a; C) = E_\ell(C \cup \{a\}) - E_\ell(C), \qquad A_\ell(a) = E_\ell(\{a\}) - E_\ell(\varnothing).$$

The design crosses nine anchor positions, full-canvas random contexts containing one to five cells, and all ordered foreground-color pairs (Fig. S10a; Appendix A.3.1).

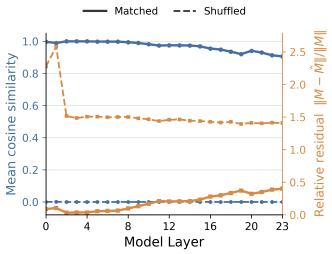
Addition is strongly context-dependent. At layer 17 with five context cells, the mean cosine between $A_\ell(a)$ and $\Delta_\ell(a; C)$ is 0.044 (95% CI: 0.030–0.058) when context and added cells share a color, versus 0.610 (95% CI: 0.607–0.612) when they differ (Fig. S10b). A five-cell analysis across depth shows that this context effect appears after the first block and persists thereafter (Fig. S11a), consistent with the preceding translation and recoloring results.



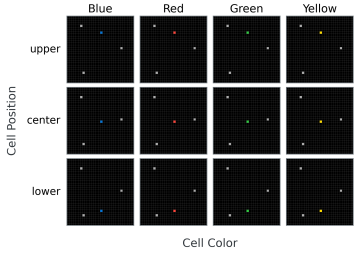
(a) Context-free single-cell design.



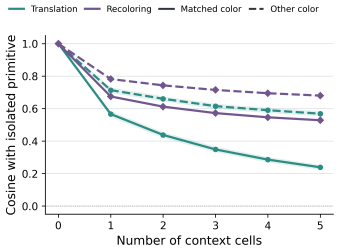
(b) Context-free primitive geometry.



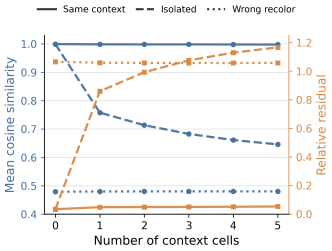
(c) Context-free primitive composition.



(d) Single-cell design in a shared context.

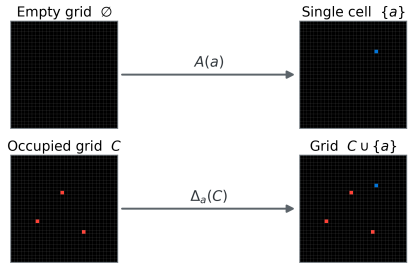


(e) Transfer of isolated primitives into context.

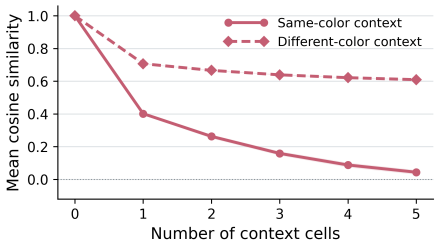


(f) Composition around the same context.

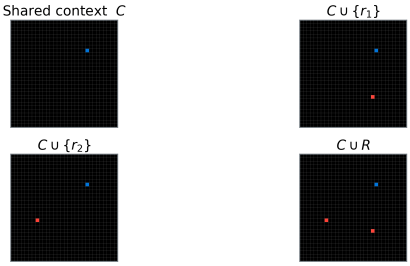
Figure S9: Primitive transformations in DINOv3-Large at 480 px compose locally but depend on their source configuration. Layer numbers are zero-indexed. **(a,d)** Synthetic single-cell example designs without and with shared context. **(b)** Layer-17 geometry of four unit translations and eight recolorings from blue. **(c)** Layer-wise prediction of individual joint transformations from family-average translation and recoloring vectors, with shuffled controls. **(e)** Similarity between contextual primitives and their empty-grid counterparts by transformation family and color context. **(f)** Joint contextual transformations predicted using primitives measured in the same context, on an empty grid, or with an incorrect contextual recoloring. Blue and orange denote cosine similarity and relative residual, respectively; bands and error bars denote 95% confidence intervals.



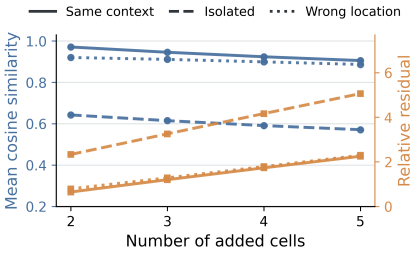
(a) Context-free and contextual cell addition.



(b) Loss of context-free transfer with grid occupancy.



(c) Single-cell additions measured from a shared context.



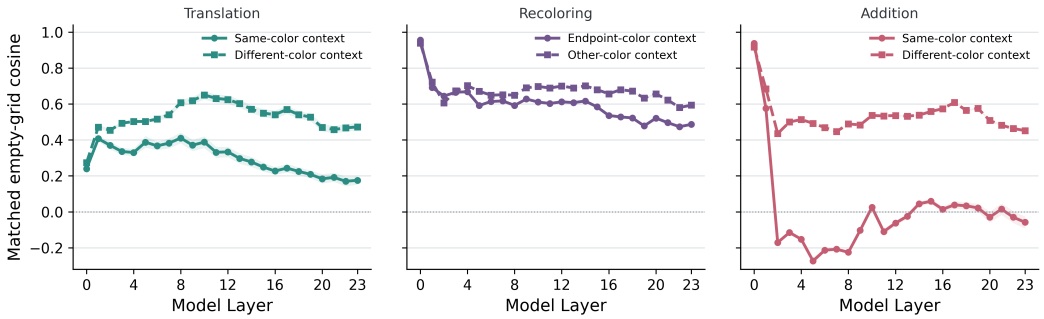
(d) Context-conditioned composition.

Figure S10: Cell-addition vectors in DINOv3-Large at 480 px are context-dependent and directionally aligned, but not magnitude-additive. Layer numbers are zero-indexed. **(a)** The same cell is added to an empty or occupied grid. **(b)** Similarity between isolated and contextual addition vectors as occupancy increases, separated by color context. **(c)** Multiple additions measured separately and jointly from a shared source context. **(d)** The joint addition is compared with sums of context-conditioned vectors, isolated additions, and a wrong-location control. Blue and orange denote cosine similarity and relative residual, respectively; bands and error bars denote 95% confidence intervals.

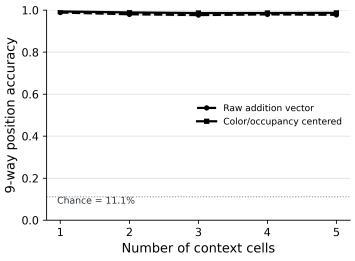
To test composition around a shared source state, an anchor cell forms a one-cell base $B = \{b\}$ and $R = \{r_1, \ldots, r_k\}$ contains $k \in \{2, \ldots, 5\}$ randomly positioned additions. We compare the joint transition $B \to B \cup R$ with the sum of the k changes obtained by adding each r_i separately to the same B (Fig. S10c). At layer 17 and $k = 5$, the sum remains directionally aligned with the joint transition (cosine 0.905, 95% CI: 0.904–0.906), but its relative residual reaches 2.249 (95% CI: 2.237–2.262). Across layers, cosine declines to approximately 0.85 by the final block while the residual remains near 2.5 (Fig. S11c). Contextual additions therefore remain in broad directional superposition without being magnitude-additive.

Summing isolated-cell additions performs worse at $k = 5$ (cosine 0.571, residual 5.060). A wrong-location control, which permutes spatial layouts while preserving the base, colors, and occupancy, remains comparatively close (cosine 0.887, residual 2.291; Fig. S10d). These aggregate measures are therefore dominated by occupancy- and color-related components. Nevertheless, a cross-validated linear decoder recovers the added cell's nine-way location from $\Delta_\ell(a; C)$ with 97.8% accuracy, versus 11.1% chance (Fig. S11b; Appendix A.3.1). Spatial information remains linearly accessible even when it contributes little to aggregate cosine geometry.

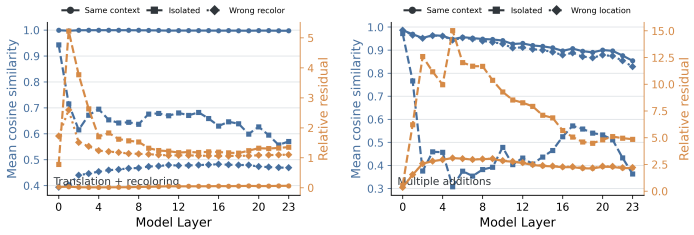
Together, the single-cell analyses identify context as a boundary of primitive transfer. Translation and recoloring compose accurately when measured from the same state; addition vectors remain aligned but are not magnitude-additive. At a higher level, each ARC intuition vector averages instance transformations measured from different support grids. The coexistence of source-dependent local vectors with a shared task-level average offers one possible account of the dissociation observed in the main paper: a representation can preserve task identity across instances without specifying the exact, query-dependent output. Establishing that connection directly will require experiments that link the controlled single-cell setting to individual ARC tasks.



(a) Layerwise context sensitivity with five context cells.



(b) Cross-validated decoding of the added cell's position.



(c) Layerwise contextual composition at the largest tested set size.

Figure S11: Additional analyses of context-dependent single-cell geometry in DINOv3-Large at 480 px. Position decoding uses four-fold grouped cross-validation over independently sampled spatial layouts, with all color variants of a layout confined to the same fold. Layerwise analyses use a balanced subset containing every unit direction in each of four placement sets. The transfer analyses use five context cells; the multi-addition analysis uses five additions from a one-cell base. Bands and error bars denote 95% spatial-bootstrap confidence intervals. Layer numbers are zero-indexed.