VISION TRANSFORMERS LEARN WORD MEANING FROM IMAGES ALONE

Anonymous authors
Paper under double-blind review

ABSTRACT

Can a vision model learn what words mean from pixels alone? We show that the two largest DINOv3 vision transformers organize words rendered in images by word meaning, despite being pretrained using self-supervision on images only, without paired captions, class labels, or other language supervision. We derive word representations from embedding differences between word images and matched controls, then test whether semantically related words are neighbors within each model's embedding space, while controlling for spelling similarity. We find that brand names in the same industry group together, as do nouns from the same semantic domain. Country-capital pairs exhibit consistent vector offsets, similarly to word embeddings trained on text. These word representations also align with the representations of the visual object categories they name. DINOv3-H and DINOv3-7B are, to our knowledge, the first self-supervised vision models to show substantial semantic structure for rendered words. Comparisons with text and speech encoders reveal shared structure among representations of the same words. Contrastive alignment with either encoder further strengthened DINOv3-H's word-level semantic structure, while leaving the other finetuned vision models largely unchanged. Image-only training can therefore teach vision models not just what words look like, but also how their meanings relate.

1 INTRODUCTION

Can knowledge of semantic relations among words emerge in models trained on images alone? Self-supervised vision models learn representations organized by the semantic categories of depicted objects (Oquab et al., 2024; Siméoni et al., 2026). Written words can also appear within their training images, but are encountered only as pixels, without paired captions, class labels, or other language supervision (Fan et al., 2025). Distinguishing these words by appearance does not establish knowledge of how their meanings relate. Word embeddings trained on text can encode such relations in their geometry (Mikolov et al., 2013), based on co-occurrence patterns in the textual input. We ask whether image-only training can produce similar structure. For example, is a vision model's representation of country names systematically offset with those of the capitals?

Training without language supervision does not mean training without exposure to language: written words may still appear as pixels within images. CLIP, by contrast, learns to recognize text through image-caption training (Radford et al., 2021), and in that model certain neurons respond to the same concepts either depicted or written (Goh et al., 2021). However, earlier comparisons found that DINOv2's predictions were unaffected by the meaning of overlaid words (Wang & Ma, 2025), and that its representations grouped images by visual content rather than written word semantics (Jalali et al., 2025). More recently, Mason et al. (2026) showed that DINOv3 representations distinguish rotated from mirrored text. While this result concerns visual structure rather than word meaning, it suggests that DINOv3 represents written strings more richly than earlier results might imply. We therefore ask whether its representations also capture semantic relations among words.

Self-supervised vision models already learn semantic categories of depicted objects (Oquab et al., 2024; Siméoni et al., 2026). But recognizing an object, or distinguishing one written word from another, does not establish knowledge of semantic relations among words. We find that this capability emerges in the largest DINOv3 models but is largely absent from the earlier self-supervised vision models we test. Thus, image-only training can recover some of the semantic structure typically studied

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.

Model	Capital germany	Synonym petrol	Category porsche
DINOv3-H	.87 **berlin**	.68 **gasoline**	.72 **ferrari**
	.62 vienna	.46 patrol	.26 terrace
	.57 edelweiss	.29 patio	.23 germany
DINOv3-7B	.78 **berlin**	.70 patrol	.45 **ferrari**
	.58 austria	.68 **gasoline**	.41 **volkswagen**
	.57 vienna	.40 engine	.29 **renault**
Web-DINO-1B	.70 hungary	.61 privet	.52 polished
	.67 guyana	.52 polite	.48 carlsberg
	.65 generous	.51 gerbil	.47 precise
DINOv2-G	.62 meatloaf	.55 patrol	.54 tadpole
	.62 gatorade	.54 barrel	.53 careless
	.61 heineken	.52 trench	.51 colombia
CLIP-B	.62 **berlin**	.44 patrol	.54 porch
	.44 austria	.36 pepsi	.46 perch
	.43 poland	.32 chevrolet	.39 epoch

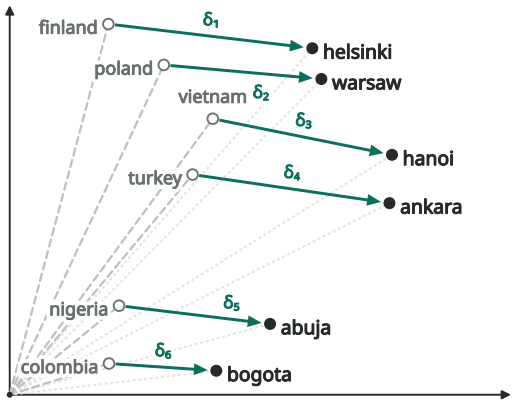


Figure 1: **A vision transformer trained on only images ranks the capital first. (a)** Three nearest rendered words per query, with cosine similarities between centered word directions shown at left. Bold marks meaning partners or same-industry brands. **(b)** Country-to-capital offsets in DINOv3-H, illustrating the vector-offset analogy relations studied by Mikolov et al. (2013). Gray lines indicate centered word directions τ; their common origin is for layout purposes only. The six pairs, hand-picked from 60, illustrate the relation rather than quantify its consistency. See Appendix C.1.

through language, despite having no objective that requires the model to read or match language. More broadly, models can acquire capabilities that their training objective never explicitly targets when the relevant structure is present in the training data. Understanding such unexpected capabilities is therefore a general problem in large-scale pretraining, not one specific to text recognition.

We analyze rendered words using word directions: embedding differences between word images and matched controls, averaged across backgrounds and centered over the vocabulary. Controlling for spelling similarity and rendering conditions, we find: (1) In DINOv3-H and DINOv3-7B, country-capital pairs are neighbors with consistent offsets, brands and nouns group by category, and synonyms are retrieved above chance (Figure 1). These models perform comparably to CLIP-B on several measures, while earlier self-supervised models primarily group words by visual similarity. To our knowledge, no prior work has shown that vision models pretrained without language supervision encode semantic relations among rendered words. (2) In these large DINOv3 models, word directions align with centered mean image embeddings of the corresponding visual object, without caption supervision or a fitted mapping. (3) Comparisons with text and speech encoders reveal shared representational structure. Contrastive finetuning on image-text or image-speech pairs, using images with no detected text, improves synonym retrieval and noun-category grouping in DINOv3-H, but changes the other finetuned models minimally.

2 Related Work

Self-supervised vision and language supervision. Self-distillation produces representations organized by semantic category (Caron et al., 2021; Oquab et al., 2024; Siméoni et al., 2026), while masked reconstruction (He et al., 2022) places greater emphasis on local visual features and texture (Park et al., 2023). At scale, self-supervised image encoders can match the performance of language-supervised encoders when used in visual question answering systems (Fan et al., 2025). A controlled comparison at a smaller scale, holding architecture and training images fixed, finds that caption supervision strengthens representations of fine-grained object categories and of text (Liu et al., 2025).

Rendered words in vision models. CLIP learns to recognize text in images as a by-product of image-text contrastive training (Radford et al., 2021). Certain neurons respond to a concept whether it is depicted or written (Goh et al., 2021). Learned projections can disentangle written and visual concepts in CLIP, and its image encoder matches rendered class names to corresponding natural images (Materzyńska et al., 2022). Representations of rendered text in CLIP's image encoder show semantic alignment with a language model comparable to that of CLIP's text encoder (Vennam et al., 2024). CLIPPO uses a single image encoder for images and rendered alt-text, but still

trains contrastively on image–alt-text pairs (Tschannen et al., 2023). Earlier comparisons find that DINOv2's predictions are unaffected by the meaning of overlaid words (Wang & Ma, 2025), and that its embeddings group images by visual content rather than the written word (Jalali et al., 2025). Our aim is to characterize semantic relations learned without language supervision, not to propose a method for language understanding.

Representational convergence. The Platonic Representation Hypothesis proposes that models trained on different modalities converge toward shared representations as they scale (Huh et al., 2024). However, image-caption neighborhood agreement weakens on larger evaluation sets and mainly reflects coarse semantic overlap (Koepke et al., 2026). After correcting for model-size biases, Gröger et al. (2026) find significant agreement in local neighborhoods but little evidence of global convergence under spectral measures.

The geometry of word meaning. Words used in similar contexts tend to have related meanings (Harris, 1954); distributional semantic models leverage contextual co-occurrence to learn a representation of word meanings. In such models, word embeddings encode semantic relations as similar vector offsets, for example between countries and their capitals (Mikolov et al., 2013). We ask whether comparable geometric structure emerges in representations of rendered words learned from images alone. Because word form can weakly covary with meaning, as documented for sound (Monaghan et al., 2014), we test these relations while controlling for spelling similarity. Separately, we test visual word-referent grounded alignment (Bender & Koller, 2020), by testing whether word representations align with representations of the depicted object categories they name.

3 METHODS

Our experiments first test whether vision models represent semantic relations among rendered words, controlling for spelling and rendering effects (Section 4). We then compare word directions with representations of the visual object categories they name (Section 5). Finally, we compare the semantic structure of word directions with representations from text and speech encoders, and test whether contrastive finetuning toward these representations through written or spoken captions strengthens that structure (Section 6).

3.1 MODELS

We measure fourteen models (Table 1): four DINOv3 models and Web-DINO-1B, trained without labels or captions on images alone, DINOv3-7B by self-distillation and the smaller three by distillation from it (Siméoni et al., 2026; Fan et al., 2025); three DINOv2 models, the predecessors to DINOv3 (Oquab et al., 2024); three MAE models, trained by masked reconstruction (He et al., 2022); CLIP-B and SigLIP2-B, trained with language supervision on image-caption pairs (Radford et al., 2021; Tschannen et al., 2025); and an untrained ViT-B as the baseline for visual form alone. The encoder f maps an image x to its final-layer class token, or for SigLIP2-B to the mean of its patch tokens, with no projection head and no per-image normalization.

3.2 WORDS AND IMAGES

Two vocabularies are used. The *lexical vocabulary* tests whether related words are neighbors (Section 4) and supports comparisons with text and speech encoders (Section 6). The *class-name vocabulary* tests whether a rendered class name is close to images of the class it names (Section 5). Table 3 in Appendix A gives every list, its size, and an example.

Lexical vocabulary. The lexical vocabulary contains 1,481 words and names and 80 pseudowords, all lowercase and three to ten letters long. The words come mainly from WordNet, with a target frequency of at least one occurrence per million words to favor familiar words. The vocabulary is divided into eight lists, used in three ways:

Relation lists. Five lists are generated for partner retrieval: whether a word's partner is among its ten nearest neighbors (Section 3.4). Three lists use triplets containing an anchor, a synonym, and a competitor. In the one-edit list (1), the competitor differs from the anchor by one letter insertion,

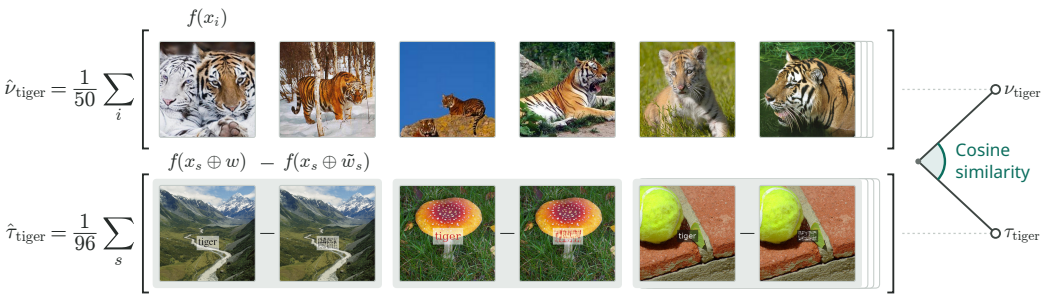


Figure 2: **Visual and word directions for *tiger*.** The upper row illustrates the uncentered visual class mean $\hat{\nu}_{\text{tiger}}$ of Eq. (2), corresponding to the mean final-layer representation over 50 ImageNet-1k validation images for a class. The lower row is the uncentered word difference $\hat{\tau}_{\text{tiger}}$ of Eq. (1), averaged over 96 backgrounds. The word image $x_s \oplus w$ shows word w on background image x_s; its control $x_s \oplus \tilde{w}_s$ has the same background, with the letters' opacity mask replaced by a phase-scrambled patch. Stacked card edges mark terms not drawn. Centering over all 1,000 classes gives the visual direction ν_{tiger}; centering over the 200 class names gives the word direction τ_{tiger}. The rays show the cosine similarity between these directions, with schematic lengths and angle.

deletion, or substitution (150 items; anchor: *torch*, synonym: *flashlight*, competitor: *touch*). This creates a controlled competition between semantic relatedness and close orthographic form. In the antonym list (2), the competitor is an antonym (50 items; *brave, bold, cowardly*). In the negation list (3), the competitor is a similarly spelled word that typically corresponds to the anchor's unnegated base (35 items; disappear, vanish, appear); exceptions are described in Appendix A. The singular–plural list (4) contains 40 pairs related in both form and meaning (*road–roads*). The country–capital list (5) contains 60 pairs expressing a factual relation (*germany–berlin*). Retrieval always uses all 1,560 other vocabulary items as candidates, not only items from the same list.

Noun and brand lists. The noun list (6) contains 384 common nouns from six categories (animal, device, food, person, plant, structure). The brand list (7) contains 192 brand names from six industries. These lists are used for neighbor purity: the fraction of a word's ten nearest neighbors that belong to the same category or industry (Section 3.4).

Pseudowords. The 80 pseudowords (8) are constructed nonwords and are never used as queries. They serve as additional retrieval candidates and are included in the mean used for centering.

Class-name vocabulary. The class-name vocabulary contains 200 single-word ImageNet-1k class names, each chosen to clearly name one class. We prefer classes whose images contain little detected text, reducing the chance that a rendered name matches writing already present in the class images. The names are rendered on the same 96 backgrounds. For each of the 1,000 ImageNet classes, a visual direction is computed from the 49,904 validation images not used as backgrounds. Each name's word direction is then compared with the visual direction of the class it names (Appendix D.1). Forty-three names also occur in the lexical vocabulary, but their word directions differ because the two vocabularies use different centering means (Section 3.3).

Form controls. The spelling controls test whether shared letters explain similarity between related words. We screen meaning pairs for inflectional and derivational relatives. Their average letter overlap within each relation is close to that of random pairs matched in length. We group words with similar character subsequences into four *spelling clusters*, requiring equal numbers from each category or industry in every cluster. Cluster membership alone therefore cannot predict those labels. We also measure category membership among neighbors within each cluster and compare the scores with category labels shuffled among words of similar length (Appendix C.2).

Rendering. Every word, name, and pseudoword is rendered on the same 96 ImageNet-1k validation backgrounds with no detected text. The 24 settings combine six fonts with four vertical position bands and vary size, color, and backing. Each setting is used on four backgrounds. Typography

and text center are fixed across words on each background, so words are compared under the same rendering conditions.

Control image. Each word image has a control with the same background, position, color, and backing. Fourier phase randomization replaces the letters' opacity mask with a textured patch in the same bounding box. Subtracting their representations measures the response to intact lettering relative to this matched overlay. Averaging these differences across backgrounds and subtracting the vocabulary mean gives one word direction (Appendix A).

3.3 Word and visual directions

Word directions. Let S index the 96 backgrounds and x_s denote background image s. For a word, name, or pseudoword w, write $x_s \oplus w$ for its word image, where $\oplus$ denotes rendering with the settings assigned to that background. The matched control is $x_s \oplus \tilde{w}_s$, where $\tilde{w}_s$ denotes the phase-scrambled opacity mask of the lettering for word w on background s. The *word direction* τ_w is the average difference between the word and control representations, centered over vocabulary V:

$$\hat{\tau}_w \;=\; \frac{1}{|S|}\sum_{s\in S}\big[\, f(x_s \oplus w) - f(x_s \oplus \tilde{w}_s)\,\big], \qquad \tau_w \;=\; \hat{\tau}_w - \frac{1}{|V|}\sum_{v\in V}\hat{\tau}_v, \tag{1}$$

Here, $\hat{\tau}_w$ is the intermediate, uncentered difference vector. For lexical and cross-encoder comparisons, V contains 1,481 words and names and 80 pseudowords. For word-class comparisons, it contains the 200 class names. The uncentered estimate is a mean of paired differences, as in difference-in-means probes (Rimsky et al., 2024; Marks & Tegmark, 2024) and concept directions under the linear representation hypothesis (Park et al., 2024). The matched control is used to improve signal-to-noise by controlling for background and rendering effects. Directly averaging the word-image embeddings without control subtraction leads to the same qualitative conclusions.

Visual directions. For a class c, let I_c index its retained class images x_i. Their uncentered class mean is $\hat{\nu}_c$. Centering it over the set C of all 1,000 classes gives the *visual direction* ν_c:

$$\hat{\nu}_c \;=\; \frac{1}{|I_c|}\sum_{i\in I_c} f(x_i), \qquad \nu_c \;=\; \hat{\nu}_c - \frac{1}{|C|}\sum_{c'\in C}\hat{\nu}_{c'}. \tag{2}$$

Centering over all 1,000 classes keeps the reference fixed when we choose the 200 classes with single-word names. The resulting visual direction is the difference between that class's mean and the mean of all the classes (Kim et al., 2018). Figure 2 compares the word and visual directions for *tiger*.

3.4 Measures

We use cosine similarity to find nearest neighbors. For the noun and brand lists, *neighbor purity* is the average fraction of a word's ten nearest neighbors that belong to the same category or industry. For noun purity, neighbors are drawn from the other 383 nouns; for brand purity, from the other 191 brands. For the five relation lists, *partner retrieval* asks whether a word's designated partner, such as its synonym or capital, is among its ten nearest neighbors in the full lexical vocabulary. The query itself is excluded. *Top-10 accuracy* is the fraction of queries whose designated target is among the ten nearest candidates.

For word–class alignment, *matched cosine similarity* is the average similarity between each class name's word direction and the visual direction of the class it names. Word-to-class retrieval ranks all 1,000 visual directions for each of the 200 class names, while class-to-word retrieval ranks the 200 word directions for each named class. Both use top-10 accuracy. All comparisons are made directly in the same embedding space; no mapping between word and visual directions is fitted. See Appendix B.1 for further details.

3.5 Alignment and finetuning

Encoders. We compare vision models' word directions with word representations from Qwen3-Embedding-0.6B, -4B, and -8B for text (Zhang et al., 2025), and w2v-BERT 2.0 for speech (SEAMLESS Communication Team et al., 2025). For text, each word, name, and pseudoword is placed in

Table 1: **DINOv3-H and DINOv3-7B are comparable to CLIP-B.** Neighbor purity is the fraction of the ten nearest neighbors sharing a noun category or brand industry; top-10 accuracy is the fraction of paired targets retrieved among the ten nearest vocabulary items. Underlining marks a value above the untrained ViT-B by more than twice the standard error of the difference, bold marks column maxima, and small numbers are standard errors over words or pairs (Appendix B).

	Neighbor purity		Top-10 accuracy			
Model	Nouns	Brands	Capital	Synonym	One-edit word	Plural or base
DINOv3-B	0.194 ±.012	0.184 ±.016	0.000 ±.000	0.000 ±.000	0.653 ±.039	0.333 ±.054
DINOv3-L	0.223 ±.014	0.235 ±.023	0.117 ±.041	0.013 ±.009	0.820 ±.031	0.733 ±.051
DINOv3-H	0.269 ±.017	**0.351** ±.027	**0.600** ±.063	0.060 ±.019	0.607 ±.040	0.813 ±.045
DINOv3-7B	0.308 ±.020	0.334 ±.031	**0.600** ±.063	0.100 ±.024	0.620 ±.040	0.853 ±.041
Web-DINO-1B	0.187 ±.012	0.167 ±.016	0.000 ±.000	0.007 ±.007	0.547 ±.041	0.293 ±.053
DINOv2-B	0.179 ±.012	0.164 ±.015	0.000 ±.000	0.000 ±.000	0.353 ±.039	0.067 ±.029
DINOv2-L	0.174 ±.011	0.167 ±.015	0.017 ±.017	0.007 ±.007	0.440 ±.041	0.093 ±.034
DINOv2-G	0.181 ±.011	0.173 ±.014	0.000 ±.000	0.013 ±.009	0.367 ±.039	0.080 ±.031
MAE-B	0.182 ±.011	0.182 ±.015	0.000 ±.000	0.000 ±.000	0.507 ±.041	0.040 ±.023
MAE-L	0.186 ±.012	0.188 ±.014	0.000 ±.000	0.000 ±.000	0.547 ±.041	0.080 ±.031
MAE-H	0.180 ±.012	0.183 ±.014	0.017 ±.017	0.007 ±.007	0.553 ±.041	0.093 ±.034
CLIP-B	0.297 ±.019	0.343 ±.027	0.550 ±.064	**0.200** ±.033	0.853 ±.029	0.987 ±.013
SigLIP2-B	**0.318** ±.019	0.299 ±.025	0.183 ±.050	0.060 ±.019	**0.947** ±.018	**1.000** ±.000
Untrained ViT-B	0.183 ±.011	0.183 ±.016	0.017 ±.017	0.007 ±.007	0.333 ±.038	0.053 ±.026
Qwen3-0.6B	0.709 ±.022	0.632 ±.031	0.600 ±.063	0.687 ±.038	0.213 ±.033	0.893 ±.036
Qwen3-4B	0.711 ±.021	0.746 ±.028	0.983 ±.017	0.733 ±.036	0.260 ±.036	0.893 ±.036
Qwen3-8B	0.711 ±.021	0.743 ±.027	0.967 ±.023	0.740 ±.036	0.240 ±.035	0.867 ±.039
w2v-BERT 2.0	0.201 ±.012	0.178 ±.017	0.000 ±.000	0.013 ±.009	0.453 ±.041	0.240 ±.049
Chance	0.164	0.162	0.006	0.006	0.006	0.006

the same 96 neutral contexts, such as "The word is *w*." We average the resulting text embeddings. For speech, four shared carrier sentences are spoken in eight voices at three rates, giving 96 utterances per word. We pool only the frames in which the word is spoken, then average over utterances. Both averages are centered over the full vocabulary of 1,561 words, names, and pseudowords. Appendix E.2 gives the templates, voices, and pooling details.

Measure of alignment. We test whether vision, text, and speech encoders agree about which words are related, despite having different input modalities. For each encoder, we form a matrix of cosine similarities between its centered vectors for the same vocabulary. Mutual 10-NN alignment measures the average fraction of each word's ten nearest neighbors shared by two encoders (Huh et al., 2024). Linear centered kernel alignment (linear CKA) compares the full pattern of pairwise similarities, using the normalized inner product of the two centered matrices (Kornblith et al., 2019). Appendix B.1 defines both scores, and Appendix B explains how agreement is compared with chance.

Finetuning. To test whether language supervision can build on existing word relationships, we finetune self-supervised vision models to match COCO images to either their written captions (Lin et al., 2014; Chen et al., 2015) or synthesized readings from SPEECH-COCO (Havard et al., 2017). We exclude images with detected text to limit direct supervision from writing in images. The text and speech runs use the same images and captions. A learned linear projection maps the vision model's image class token to the embedding dimension of the frozen text encoder (Qwen3-Embedding-0.6B) or speech encoder (w2v-BERT 2.0). Symmetric InfoNCE loss (Radford et al., 2021) trains the vision model and projection to match the corresponding caption's text or speech vector. We discard the projection and repeat the same word tests after finetuning, so only the vision model's weights have changed (Appendix E.3).

4 Lexical Structure

Our first experiment evaluates whether word directions of rendered words are semantically structured; results are shown in Table 1, Figure 1a, and Appendix C.

Semantic relations. DINOv3-H and DINOv3-7B group rendered words by meaning. In both, 60% of countries have their capital among their ten nearest neighbors (c.f. 55% in CLIP-B, with language supervision, but 0% for DINOv2), and nouns and brands lie near words of their own category about as often as in CLIP-B. DINOv3-L shows a weaker version of the same structure. The other eight self-supervised models, Web-DINO-1B among them, perform on par with the untrained ViT-B, using only structural inductive bias. However, synonym retrival is limited in vision models: top-10 accuracy is 20% in CLIP-B and 10% in DINOv3-7B, compared with 69–74% in the Qwen3 text encoders given the vocabulary as text. Across depth, noun purity rises from mid-depth in DINOv3-H, DINOv3-7B, CLIP-B, and SigLIP2-B, while capital retrieval emerges only in the last two measured depths of DINOv3-H, DINOv3-7B, and CLIP-B; the remaining models stay near the untrained ViT-B baseline, except DINOv3-L at the final depth (Appendix C.3).

Spelling. Sensitivity to visual overlap between words does not by itself imply a representation of spelling. Every model retrieves one-edit words, and even the untrained ViT-B achieves 33% top-10 accuracy. Most one-edit pairs differ by a single letter substitution, leaving the other letters in the same positions and therefore much of the rendered image unchanged. DINOv2 and MAE show this pattern particularly clearly: they retrieve one-edit words readily, yet retrieve plural and base partners far less often. Thus, their success on one-edit words can be explained largely by pixel- and patch-level similarity rather than by a general ability to recognize similar spellings. Insertions and deletions provide less visual overlap because they change the word width and, with centered text, shift the remaining letters (Appendix C.1).

Sensitivity to word form also does not predict semantic retrieval. SigLIP2-B achieves 100% top-10 accuracy for plural or base partners but only 18% for capitals, compared with 60% capital retrieval in DINOv3-H. Conversely, the Qwen3 text encoders achieve 69–74% synonym accuracy but only 21–26% one-edit accuracy. Thus, strong retrieval of spelling-related words is neither necessary nor sufficient for strong semantic retrieval. We therefore interpret semantic structure relative to the untrained ViT-B baseline, which captures similarity arising from visual form alone.

5 Word-Class Alignment

We test whether word directions τ_w align with their corresponding visual class directions ν_w by measuring cosine similarity in the same space. Results are in Table 2; additional experiments and ablations are in Appendix D.3–D.5.

Table 2: **Word directions in DINOv3-H and DINOv3-7B align with the classes they name.** Matched cosine similarity is the mean cosine similarity of the 200 word directions with the visual directions of the classes they name. Word-to-class ranks the 1,000 visual directions for a word direction; class-to-word ranks the 200 word directions for a named class; top-10 accuracy is the share of queries whose target is among the ten nearest, and median rank is the median of the target's rank over the queries, so a median rank of one means the target is nearest for at least half of them.

		Word to class		Class to word	
Model	Matched cosine similarity	Top-10 accuracy	Median rank	Top-10 accuracy	Median rank
DINOv3-B	0.002 ±.003	0.010 ±.007	486	0.060 ±.017	94.5
DINOv3-L	0.025 ±.004	0.120 ±.023	335	0.180 ±.027	68
DINOv3-H	0.234 ±.014	0.645 ±.034	**1**	0.680 ±.033	**1**
DINOv3-7B	0.120 ±.012	0.325 ±.033	154	0.405 ±.035	32.5
Web-DINO-1B	0.001 ±.002	0.020 ±.010	477	0.045 ±.015	95
DINOv2-B	-0.002 ±.003	0.010 ±.007	527.5	0.060 ±.017	103.5
DINOv2-L	-0.001 ±.002	0.000 ±.000	485	0.045 ±.015	99.5
DINOv2-G	0.001 ±.002	0.005 ±.005	463	0.045 ±.015	94
MAE-B	-0.011 ±.006	0.005 ±.005	559	0.020 ±.010	113
MAE-L	0.003 ±.004	0.015 ±.009	491	0.045 ±.015	98.5
MAE-H	0.002 ±.004	0.005 ±.005	461.5	0.050 ±.015	96.5
CLIP-B	**0.237** ±.010	**0.855** ±.025	**1**	**0.845** ±.026	**1**
SigLIP2-B	0.184 ±.009	0.630 ±.034	2	0.735 ±.031	2
Untrained ViT-B	0.005 ±.008	0.005 ±.005	511.5	0.045 ±.015	99
Chance	0.000	0.010		0.050	

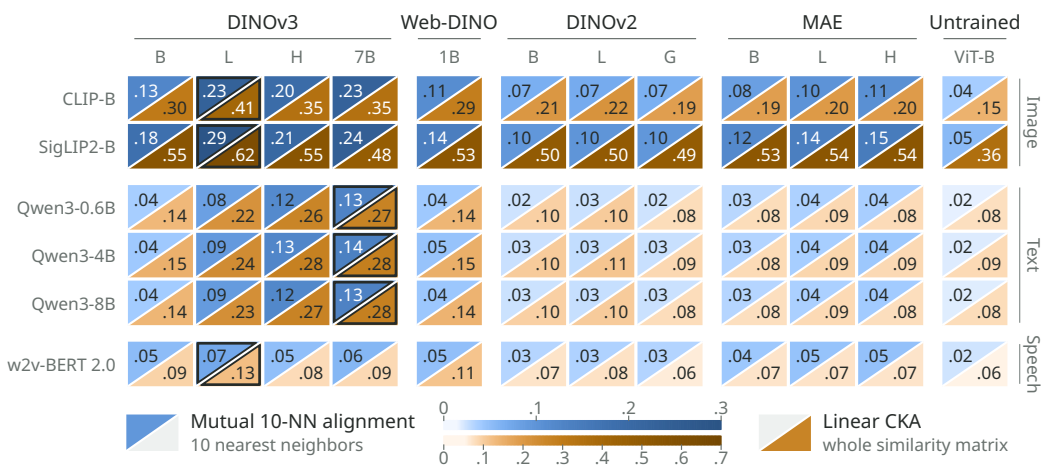


Figure 3: **DINOv3-H and DINOv3-7B are most aligned with text encoders.** Each cell compares a column model and row encoder on the lexical vocabulary; the image rows, CLIP-B and SigLIP2-B, receive rendered word images. The upper triangle shows mutual 10-NN alignment (shared nearest neighbors; chance 0.006), and the lower triangle shows linear CKA between centered similarity matrices (chance 0.005–0.052). Each measure uses a logarithmic scale, outlines mark row maxima, and every cell exceeds a null that relabels words of equal length (Appendix B).

Named classes. DINOv3-H and DINOv3-7B show substantial word–class alignment; DINOv3-H is on par with CLIP-B despite no alignment signal during training. Its word direction ranks the named class first among 1,000 for over half of the 200 class names (median rank $= 1$). This alignment persists when visual directions are estimated only from images with no detected text (0.230 versus 0.234 with all images; Appendix D.3). SigLIP2-B also aligns, and DINOv3-L weakly. The remaining models show no significant matched alignment despite retrieving one-edit words.

Model size. DINOv3-H exceeds DINOv3-7B on matched cosine similarity and on retrieval by a factor of 1.7 to 2.0, although DINOv3-7B is no weaker in Table 1 and DINOv3-H was distilled from it (Siméoni et al., 2026). We cannot explain this gap. Web-DINO-1B and DINOv2-G, at a billion parameters or more, show no word-class alignments, so size alone does not determine model behavior.

6 Alignment and Finetuning

We now compare each model with text encoders, given the 1,561 words, names, and pseudowords as text, and with a speech encoder, given them as audio (Section 3.5; results in Figure 3).

Text encoders. DINOv3-H and DINOv3-7B share 12–14% of their ten nearest neighbors with the text encoders, the highest agreement among the self-supervised vision models. Linear CKA shows the same pattern. Although text encoders never see the rendered words, their representations retain spelling information: Qwen3 encoders retrieve 21–26% of one-edit partners (Table 1). Vision models without detectable semantic structure share at most 5% of their neighbors with text encoders, suggesting that word form contributes to agreement but does not fully explain it.

Shared word form. CLIP-B, SigLIP2-B, and the speech encoder order the models differently. Every vision-only model agrees more with CLIP-B and SigLIP2-B than with any text encoder, and MAE-H reaches a linear CKA of 0.54 with SigLIP2-B, above DINOv3-7B's 0.48, although MAE shows no semantic structure. Two image encoders see the same rendered letters, and a word's sound follows its spelling, so we read this agreement as largely reflecting word form. DINOv3-L, the self-supervised model that finds the most one-edit words, has the largest values against all three, yet shares fewer nearest neighbors with every text encoder than DINOv3-H or DINOv3-7B.

Finetuning. Contrastive finetuning on images with no detected text improves synonym retrieval and noun grouping in DINOv3-H, while the other nine finetuned models change minimally (Appendix E.3). Using four rendered images per word, synonym top-10 accuracy rises from about 2% to 11% with text supervision and 8% with speech supervision, averaged over five typography draws. After text finetuning, both synonym retrieval and noun purity exceed the pretrained model's scores using all 96 images. Thus, caption supervision makes it easier for DINOv3-H to recover its lexical structure from fewer rendered examples, without yielding comparable gains in the other models.

7 Discussion

The scope and possible sources of semantic knowledge. DINOv3-H and DINOv3-7B exhibit lexical semantic structure connecting written words to one another and to depicted object categories. Not all relations are equally strong: Category membership and country-capital associations are more clearly expressed than synonymy. Similarity and association are distinct (Hill et al., 2015), so factual relationships do not imply equally strong synonym knowledge. These findings do not establish taxonomic hierarchies or sentence-level composition. Nor do consistent offsets improve analogy retrieval over the country-only baseline, reinforcing the distinction between geometric regularity and analogical reasoning (Linzen, 2016).

The source of this structure remains unresolved. Words can acquire associations from surrounding visual content or through text embedded in frames; the latter could provide linguistic co-occurrence signals even for semantic relations not directly instantiated in the depicted scene. Embedded text influences CLIP training (Lin et al., 2024); it is also a potential source of semantic information in DINOv3's images, which are drawn mainly from public Instagram posts (Siméoni et al., 2026). Image-only training therefore need not mean learning exclusively from depicted, concrete objects; vision models' knowledge of abstract semantics remains an open question.

What finetuning changes. Finetuning improves synonym retrieval and noun purity in DINOv3-H, with little change in the other finetuned models. After text finetuning, just four rendered images per word yield higher scores on both measures than all 96 images before finetuning. This is consistent with a partial functional-dormancy interpretation: caption supervision makes the existing lexical structure easier to recover. However, these results do not distinguish eliciting existing relations from learning new ones. The interpretation connects to work on eliciting latent capabilities (Donoway et al., 2025), although the evidence here concerns representations rather than downstream behavior. Claims of representational convergence also remain limited by the cross-encoder null, which controls for word length but not all shared properties of word form.

Implications for downstream systems and safety. Our results demonstrate that the absence of language supervision does not guarantee that written text is inert to systems built on a vision encoder. Image-only pretraining can produce lexical and factual associations, and later finetuning toward language can strengthen some of them. Harmful requests rendered in images can bypass safeguards in vision-language systems (Gong et al., 2025). If scaling broadens the semantic knowledge of image-only pretrained encoders, evaluations should examine both emerging capabilities and their downstream use (Shevlane et al., 2023). Our experiments establish neither instruction following nor harmful behavior, but motivate testing whether rendered instructions influence downstream systems.

8 Conclusion

We show that vision transformers pretrained without explicit language supervision can encode semantic relations among written words. In DINOv3-H and DINOv3-7B, word directions capture category membership and country-capital associations under spelling controls, with weaker evidence for synonymy. These directions also align with representations of the depicted object categories they name and share neighborhood structure with text encoders. Contrastive finetuning on images screened for text raises synonym retrieval and the grouping of nouns by category in DINOv3-H, lowers its capital retrieval, and changes the other finetuned models little. These findings establish word-level semantic structure, rather than sentence understanding or instruction following. Image-only pretraining can therefore teach vision models relationships among written words and their visual referents, extending what they learn beyond the appearance of text.

REFERENCES

Youngmin Baek, Bado Lee, Dongyoon Han, Sangdoo Yun, and Hwalsuk Lee. Character region awareness for text detection. In *CVPR*, 2019. arXiv:1904.01941.

Emily M. Bender and Alexander Koller. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *ACL*, pp. 5185–5198, 2020.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. arXiv:2104.14294.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Elizabeth Donoway, Hailey Joren, Arushi Somani, Henry Sleight, Julian Michael, Michael Deweese, John Schulman, Ethan Perez, Fabien Roger, and Jan Leike. Quantifying elicitation of latent capabilities in language models. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, pp. 150412–150443. Curran Associates, Inc., 2025. doi: 10.52202/085713-5029. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/dd16e008534dc52fab72fdedad069d5c-Paper-Conference.pdf.

David Fan, Shengbang Tong, Jiachen Zhu, Koustuv Sinha, Zhuang Liu, Xinlei Chen, Michael Rabbat, Nicolas Ballas, Yann LeCun, Amir Bar, and Saining Xie. Scaling language-free visual representation learning. In *ICCV*, 2025. arXiv:2504.01017.

Gabriel Goh, Nick Cammarata, Chelsea Voss, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. Multimodal neurons in artificial neural networks. *Distill*, 6(3), 2021. doi: 10.23915/distill.00030.

Yichen Gong, Delong Ran, Jinyuan Liu, Conglei Wang, Tianshuo Cong, Anyu Wang, Sisi Duan, and Xiaoyun Wang. FigStep: Jailbreaking large vision-language models via typographic visual prompts. In *AAAI*, 2025. arXiv:2311.05608.

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

Zellig S. Harris. Distributional structure. *Word*, 10(2–3):146–162, 1954.

William Havard, Laurent Besacier, and Olivier Rosec. SPEECH-COCO: 600k visually grounded spoken captions aligned to MSCOCO data set. In *GLU Workshop on Grounding Language Understanding*, pp. 42–46, 2017. arXiv:1707.08435.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. arXiv:2111.06377.

Felix Hill, Roi Reichart, and Anna Korhonen. SimLex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41(4):665–695, 2015. ISSN 0891-2017. doi: 10.1162/COLI_a_00237.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. Position: The platonic representation hypothesis. In *ICML*, 2024. arXiv:2405.07987.

Mohammad Jalali, Bahar Dibaei Nia, and Farzan Farnia. Towards an explainable comparison and alignment of feature embeddings. In *ICML*, 2025. arXiv:2506.06231.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV). In *ICML*, 2018. arXiv:1711.11279.

A. Sophia Koepke, Daniil Zverev, Shiry Ginosar, and Alexei A. Efros. Back into Plato's cave: Examining cross-modal representational convergence at scale. *arXiv preprint arXiv:2604.18572*, 2026.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *ICML*, 2019. arXiv:1905.00414.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, pp. 740–755, 2014. doi: 10.1007/978-3-319-10602-1_48.

Yiqi Lin, Conghui He, Alex Jinpeng Wang, Bin Wang, Weijia Li, and Mike Zheng Shou. Parrot captions teach CLIP to spot text. In *ECCV*, 2024. arXiv:2312.14232.

Tal Linzen. Issues in evaluating semantic spaces using word analogies. In *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP (RepEval)*, pp. 13–18, 2016.

Yiming Liu, Yuhui Zhang, Dhruba Ghosh, Ludwig Schmidt, and Serena Yeung-Levy. Data or language supervision: What makes CLIP better than DINO? In *Findings of EMNLP*, 2025. arXiv:2510.11835.

Samuel Marks and Max Tegmark. The geometry of truth: Emergent linear structure in large language model representations of true/false datasets. In *COLM*, 2024. arXiv:2310.06824.

Sebastian Ray Mason, Anders Gjølbye, Phillip Chavarria Højbjerg, Lenka Tětková, and Lars Kai Hansen. Large vision models can solve mental rotation problems. In *ICASSP*, pp. 1571–1575, 2026. doi: 10.1109/ICASSP55912.2026.11460589.

Joanna Materzyńska, Antonio Torralba, and David Bau. Disentangling visual and written concepts in CLIP. In *CVPR*, 2022. arXiv:2206.07835.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *NeurIPS*, 2013. arXiv:1310.4546.

Padraic Monaghan, Richard C. Shillcock, Morten H. Christiansen, and Simon Kirby. How arbitrary is language? *Philosophical Transactions of the Royal Society B*, 369(1651):20130299, 2014.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, et al. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024. arXiv:2304.07193.

Kiho Park, Yo Joong Choe, and Victor Veitch. The linear representation hypothesis and the geometry of large language models. In *ICML*, 2024. arXiv:2311.03658.

Namuk Park, Wonjae Kim, Byeongho Heo, Taekyung Kim, and Sangdoo Yun. What do self-supervised vision transformers learn? In *ICLR*, 2023. arXiv:2305.00729.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. arXiv:2103.00020.

Nina Rimsky, Nick Gabrieli, Julian Schulz, Meg Tong, Evan Hubinger, and Alexander Matt Turner. Steering Llama 2 via contrastive activation addition. In *ACL*, 2024. arXiv:2312.06681.

SEAMLESS Communication Team, Loïc Barrault, Yu-An Chung, et al. Joint speech and text machine translation for up to 100 languages. *Nature*, 637(8046):587–593, 2025. doi: 10.1038/s41586-024-08359-z. SeamlessM4T; arXiv:2308.11596.

Toby Shevlane, Sebastian Farquhar, Ben Garfinkel, Mary Phuong, Jess Whittlestone, Jade Leung, Daniel Kokotajlo, Nahema Marchal, Markus Anderljung, Noam Kolt, Lewis Ho, Divya Siddarth, Shahar Avin, Will Hawkins, Been Kim, Iason Gabriel, Vijay Bolina, Jack Clark, Yoshua Bengio, Paul Christiano, and Allan Dafoe. Model evaluation for extreme risks. *arXiv preprint arXiv:2305.15324*, 2023. doi: 10.48550/arXiv.2305.15324. URL https://arxiv.org/abs/2305.15324.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, et al. DINOv3. *Transactions on Machine Learning Research (TMLR)*, 2026. ISSN 2835-8856. URL https://openreview.net/forum?id=2NlGyqNjns. Featured Certification.

Michael Tschannen, Basil Mustafa, and Neil Houlsby. CLIPPO: Image-and-language understanding from pixels only. In *CVPR*, 2023. arXiv:2212.08045.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, et al. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Sreeram Vennam, Shashwat Singh, Anirudh Govil, and Ponnurangam Kumaraguru. Emergence of text semantics in CLIP image encoders. In *NeurIPS Workshop on Unifying Representations in Neural Models (UniReps)*, 2024. URL https://openreview.net/forum?id=eEtFRl7ksZ.

Hanzhang Wang and Qingyuan Ma. Textural or textual: How vision-language models read text in images. In *ICML*, 2025. URL https://proceedings.mlr.press/v267/wang25bw.html.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

A Data and Rendering

Word selection. We sampled the vocabulary mainly from WordNet and chose round list sizes. Eight lists record the words, their roles within an item, and their category and spelling-cluster labels (Table 3). We checked for repeated words, rendering fit, morphological relations, letter overlap, and equal category counts within spelling clusters.

The construction records also describe language-model proposals for synonym pairs and antonym and negation triples, followed by screening. One-edit words were found by searching possible letter edits. Noun candidates combined a WordNet pool with language-model proposals. Separate procedures formed plural pairs, filtered a country-capital list, and generated pseudowords.

Word frequency. We estimated English word frequencies using `wordfreq`, aiming for at least one occurrence per million words. The estimates do not distinguish different meanings of a word. The final lists include 15 capital names, 152 nouns, and 121 brand names below this target. Within three-word items, the median ratio of highest to lowest frequency is 6.31 in the synonym and one-edit list, 8.93 in the synonym and antonym list, and 21.38 in the negation list.

Spelling within items. All 150 one-edit words have Levenshtein edit distance one from their anchor: 135 substitutions, seven insertions, and eight deletions. Synonyms in this list are on average 1.23 letters longer than the anchor. A frequency above one per million does not by itself establish that a one-edit word is an ordinary dictionary word; the list includes *ain*, *lan*, and *icc*.

Letter overlap. We measure overlap as the number of distinct letters shared by two words divided by the number of distinct letters in either word. Repeated letters are counted once. For each meaning relation, we compare the average overlap with that of random word pairs matched in length. The largest excess is 0.0125. This is an average for a relation, so individual pairs may share more letters. We also use WordNet relations and spelling rules to check for inflectional and derivational relatives. All meaning pairs pass these checks.

Table 3: The eight vocabulary lists. Items are triples in the first three lists, pairs in the next two, and individual words or pseudowords in the remaining lists. Each triple defines two pairs sharing an anchor. The Words column includes names and pseudowords. Each appears in one list only. The Pairs column counts pairs before the query direction is reversed.

List	Items	Words	Pairs	Example and purpose
Synonym and one-edit comparison	150	450	300	*torch–flashlight*, *torch–touch*: meaning and spelling
Synonym and antonym	50	150	100	*brave–bold*, *brave–cowardly*: synonymy and opposition
Synonym and apparent negation/base	35	105	70	*disappear–vanish*, *disappear–appear*: meaning and similar form
Singular and plural	40	80	40	*road–roads*: form with related meaning
Country and capital	60	120	60	*germany–berlin*: factual association
Common nouns	384	384	—	Category membership
Brands	192	192	—	Industry membership
Pseudowords	80	80	—	Retrieval candidates and centering
Total	—	1,561	570	

Pairs. Each three-word item defines two pairs with the same anchor. For example, *torch*, *flashlight*, and *touch* define the pairs *torch–flashlight* and *torch–touch*. When *flashlight* is the retrieval target, *touch* remains among the candidates. The five relation lists contain 570 pairs before the query direction is reversed. The directions used in each analysis are specified in Appendix B.1. Synonym pairs occur in all three lists of three-word items.

The relation lists contain 905 distinct words and names. No word, name, or pseudoword appears twice across the eight lists. Names are lowercased for rendering. Forty-three of the separate 200

class names also occur in the first vocabulary. Their rendered images are identical, but their word directions differ because they are centered over different vocabularies.

Exceptions to the relation labels. Some words in the negation list only resemble negations or share a stem with the recorded base. Examples include *injury–jury*, *invent–vent*, *distract–tract*, and *impending–pending*. The pair *harmless–harmful* changes a suffix, and *invaluable* does not simply negate *valuable*. The full list, therefore, compares synonyms with words of similar spelling but is not a uniform test of negation. The plural list also contains exceptions. The pair *ala–alas* is not a standard singular–plural pair, and some plurals have distinct lexicalized senses.

Categories and spelling clusters. The noun categories are animal, device, food, person, plant, and structure, with 64 words each. The brand industries are beverage, carmaker, electronics, internet and software, retail chain, and video game, each with 32 names. Each list has four spelling clusters. Every noun category has 16 words in each cluster, and every brand industry has eight. Category and spelling-cluster labels therefore have Cramér's $V = 0$. This balance does not remove associations between category and individual spelling features, such as suffixes or word length.

The recorded noun-clustering procedure represents each word by TF–IDF-weighted character sequences of two to four letters, including word boundaries. It reduces the vectors to 50 dimensions using truncated singular value decomposition, normalizes them, and fits four k-means centers. Words are then assigned to the centers, with an equal number from each semantic category in each cluster. A word's assignment therefore depends on its spelling, the other candidate words, and the required category counts.

Pseudowords. We generated 80 four- to eight-letter pseudowords from consonant onsets, vowel sequences, and endings, filtering out `wordfreq` and WordNet entries, given names, and close spelling matches to specified words or accepted pseudowords. They serve as additional retrieval candidates, contribute to vocabulary centering, and are included in cross-encoder comparisons, but are neither queried nor scored as correct partners in word-pair tests. They are excluded from noun and brand candidate sets. Scrambled images, not pseudowords, provide the matched visual controls.

Background images. We use 96 ImageNet-1k validation photographs from 96 different classes. We screen candidates for image variation and central clutter, and rank them using CLIP-B prompts that describe image quality. We screen for text using CRAFT (Baek et al., 2019), then manually inspect every selected background to confirm that it contains no visible text.

Typography. Every word, name, and pseudoword fits inside the central 224-pixel crop-safe window of a 256-pixel image at every font size. The widest measures approximately 187 pixels. We pair each of the six fonts with each of the four vertical position bands, yielding 24 settings. Each font size and color occurs equally often, and half the settings use a backing panel. Each setting is used on four distinct backgrounds. We render all words, names, and pseudowords on these same 96 backgrounds with the assigned typography. Each background has one setting, rather than all 24. The text center is fixed on each background, while the text width and panel width vary with the word.

The fonts are DejaVu Sans, PT Sans, DejaVu Serif, Nimbus Roman, DejaVu Sans Mono, and URW Gothic. Small, medium, and large text use font sizes of 13, 20, and 28 pixels. The colors are near-black, near-white, red, blue, green, and purple. Text is centered horizontally. Its vertical center lies within one of four equal bands of the crop-safe window. The backing is bare or a translucent panel, dark behind near-white text and white behind the other colors. Five of the ten pairs of typographic factors are exactly balanced. The strongest association among the others is between font and color, with Cramér's V = 0.32. Rendering the 1,481 words and names and 80 pseudowords on 96 backgrounds, each with a word image and control image, plus the 96 bare backgrounds, gives 299,808 images.

Control image. We replace the letters' opacity mask inside the text bounding box with a phase-scrambled mask. We keep the constant Fourier term, draw the other phases uniformly from $[-\pi, \pi]$, take the real part of the inverse transform, rescale to the original mask's mean and standard deviation, and clip to $[0, 1]$. Word and control images share the background, bounding box, position, color, and panel. Taking the real part and clipping can change the Fourier amplitudes, opacity sum, and variance.

The subtraction therefore compares intact and scrambled lettering while allowing some differences in visual form to remain. The phases are fixed for each word or pseudoword on each background.

Centering. Before centering, the squared norm of the vocabulary mean of the uncentered difference vectors accounts for a median of 88% of their mean squared norm across the fourteen models, ranging from 27% in CLIP-B to 93% in DINOv2-G. We subtract this mean from $\hat{\tau}_w$ to obtain the word direction τ_w in (1). A word's direction therefore depends on the rest of the vocabulary. For lexical and cross-encoder comparisons, the mean includes 1,481 words and names and 80 pseudowords. For word–class comparisons, it includes only the 200 class names.

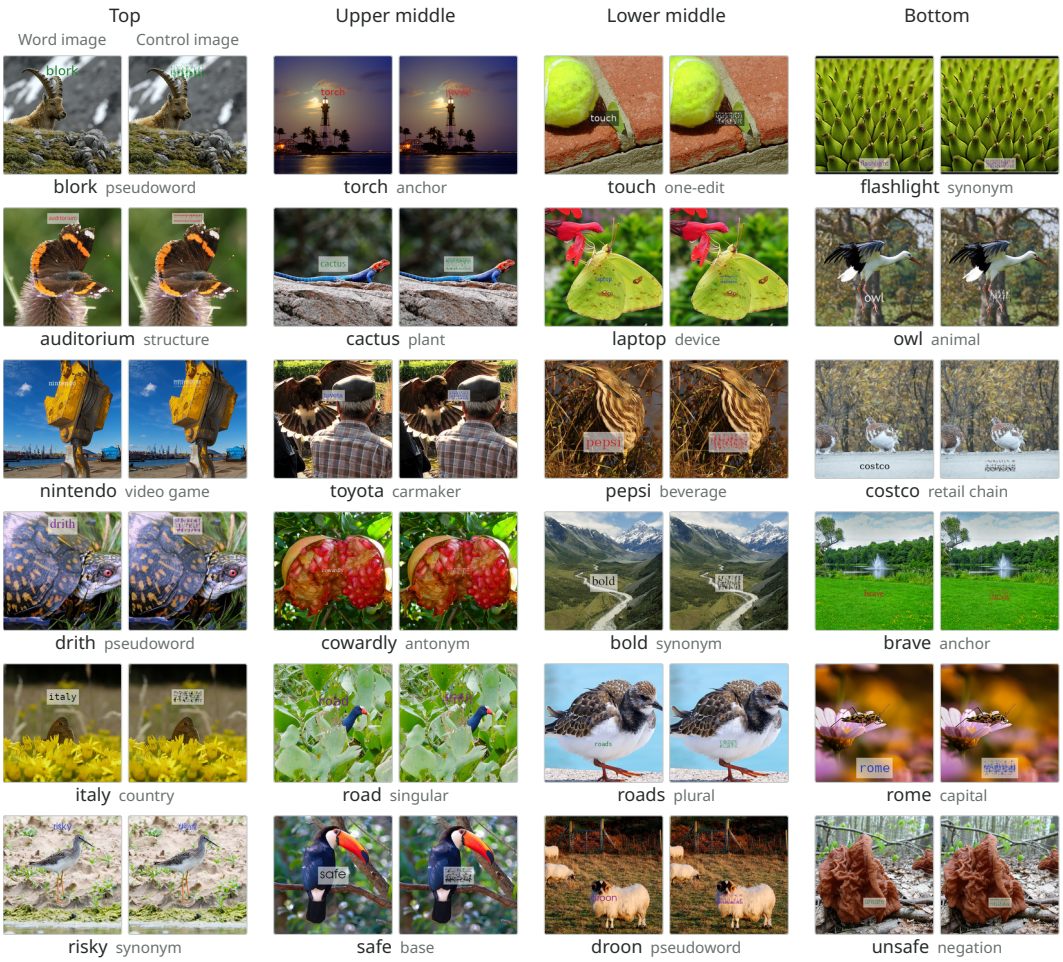


Figure 4: **Word images and control images.** Each row shows one font and each column one vertical position band, covering the 24 typography settings. Each cell shows a word image beside its control image. The control uses the same background, typography, and text color, with the letters' opacity mask replaced by a phase-scrambled copy. The words and pseudowords are examples from the eight lists in Table 3, with their roles indicated.

B MEASURES AND STATISTICAL TESTS

We define the measures used to assess lexical structure and alignment, then describe the statistical tests, uncertainty estimates, and comparisons with the untrained baseline.

B.1 MEASURES

Neighbors and retrieval. We rank candidates by cosine similarity, excluding the query itself. For neighbor purity, we find each noun's ten nearest neighbors among the other nouns and each brand's among the other brands. Purity is the fraction that share the query's category or industry, averaged across queries. Partner retrieval searches the full vocabulary for a word's designated partner. We report top-10 accuracy, the fraction of queries whose partner ranks among the ten nearest candidates, and median rank across queries, resolving ties in the target's favor. The main lexical results use the first word of each pair as the query. We report both directions separately in Appendix C.1 and pool them for the finetuning analysis (Section 6).

Word and visual directions. Three measures compare the word directions of the 200 class names with the visual directions. Word-to-class retrieval queries with the word direction of each class name and ranks the 1,000 visual directions, so top-10 chance is 0.010. Class-to-word retrieval runs the other way, from the visual direction of each of the 200 named classes to the 200 word directions, so top-10 chance is 0.050. Matched cosine similarity ranks no candidates and instead averages, over the 200 class names, the cosine similarity between a name's word direction and the visual direction of the class it names, both taken from the same model and layer. Its chance is zero, and its null reassigns classes to words (Appendix B.2).

Alignment across encoders. Mutual 10-NN alignment and linear CKA compare two models or encoders through the cosine similarities each gives the same 1,561 words, names, and pseudowords, since their vectors share no basis and often differ in width. Every model and encoder E gives a matrix S_E of cosine similarities between its centered vectors, and $N_{10}^E(w)$ is the set of the ten other words nearest to w under S_E. Mutual 10-NN alignment (Huh et al., 2024) between two encoders A and B is

$$\frac{1}{n}\sum_{w}\frac{\left|N_{10}^A(w)\cap N_{10}^B(w)\right|}{10}, \qquad n = 1{,}561, \tag{3}$$

where the sum over w covers the full vocabulary. The score is symmetric in A and B, with an expected value of 0.006 when the two neighborhoods are unrelated. Linear CKA (Kornblith et al., 2019) centers each matrix over the vocabulary, $K_E = HS_EH$ with $H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$. Here, H is the centering matrix, I is the $n \times n$ identity matrix, and $\mathbf{1}$ is a column vector of n ones. The score is

$$\frac{\langle K_A, K_B\rangle_F}{\|K_A\|_F\,\|K_B\|_F}, \tag{4}$$

Here, $\langle\cdot,\cdot\rangle_F$ is the Frobenius inner product, the sum of products of corresponding matrix entries. The Frobenius norm $\|\cdot\|_F$ is the square root of the sum of squared entries. Linear CKA is one when K_A and K_B are proportional. As in Kornblith et al. (2019), we keep the diagonal of each matrix, each word's similarity with itself. When one encoder's words are relabeled uniformly at random, linear CKA has expectation $\operatorname{tr} K_A \operatorname{tr} K_B/\big((n-1)\,\|K_A\|_F\,\|K_B\|_F\big)$, because the rows and columns of a centered matrix sum to zero. The trace tr is the sum of the diagonal entries. That chance depends on the two matrices and is 0.005 to 0.052 across the cells of Figure 3. The two measures weigh different things: mutual 10-NN alignment depends only on each word's ten nearest neighbors, and linear CKA on all the similarities, weighted toward the principal components of largest variance.

B.2 STATISTICAL TESTS

Permutation tests and corrections. Our permutation tests keep the vectors fixed and shuffle the labels or assignments relevant to each measure. For matched cosine similarity, we randomly reassign classes to words 2,000 times. For category structure, we shuffle category labels among words of similar length (Appendix C.2). With R random permutations, the one-sided p-value is $(r+1)/(R+1)$, where r counts scores at least as large as the observed score. The two-clue analysis

instead evaluates every combination of sign flips across the twelve clue grids, using a two-sided test for the residual and one-sided tests for the other outcomes (Appendix D.5). Tables report uncorrected p-values. Where indicated, we apply Holm correction separately across all fourteen models in the within-item comparisons and across the thirteen pretrained models in the two-clue analysis.

Null for alignment across encoders. To control for word length, we shuffle word labels among words of equal length in one encoder while keeping the other encoder's similarity matrix fixed. We recompute both alignment measures for 1,000 permutations, using the same permutations for every comparison in Figure 3. For every comparison, including those involving the untrained ViT-B, the observed score exceeds all corresponding permutation scores. All comparisons remain significant after Benjamini–Hochberg correction within each measure.

Standard errors and intervals. Standard errors reflect variation across words, classes, pairs, or items, with images, typography, and the held-out split fixed. We use binomial standard errors for top-10 and nearest-centroid accuracy. For item-level similarities, we divide the standard deviation across items by the square root of the item count. For neighbor purity and matched cosine similarity, we recompute scores on 1,000 random subsets of 80% of the words, sampled without replacement. For purity, neighbors are drawn only from the selected subset. We multiply the standard deviation across subsets by $\sqrt{m(n-1)/(n(n-m))}$, where n is the full sample size and $m = \lfloor 0.8n \rfloor$, to estimate the full-sample standard error. For the two-clue analysis, we report 95% percentile bootstrap intervals from 20,000 resamples of the twelve clue grids.

Comparisons with the untrained ViT-B. We underline values that exceed the untrained ViT-B by more than twice the estimated standard error of the difference. We estimate this error as $\sqrt{s^2 + s_0^2}$, where s and s_0 are the standard errors for the trained and untrained models. The rule does not account for pairing across models or adjust for multiple comparisons. Values without standard errors are not underlined.

C LEXICAL STRUCTURE

We examine how query direction and spelling affect word-pair retrieval, test category grouping on held-out words, and trace lexical structure across model depth.

C.1 PAIRS

Both query directions. Table 4 reports partner retrieval with either word in a pair used as the query. Reversing the query direction lowers country-capital retrieval in DINOv3-H and DINOv3-7B but raises it in CLIP-B. Synonym retrieval improves in both large DINOv3 models when the partner is used as the query. The main results in Table 1 use forward queries only and combine the plural and base pairs in one column.

Projection in Figure 1b. The panel shows a two-dimensional projection of six country-capital pairs in DINOv3-H, selected after inspecting the full set for familiar names and a legible layout. We normalize the word directions to unit length and center the displayed endpoints. The horizontal axis follows the mean country-to-capital offset of the undisplayed pairs. The vertical axis is the leading principal component of the displayed pairs' midpoints after removing the horizontal component. For display, we stretch the horizontal coordinates by a factor of 1.6 relative to the vertical coordinates.

Comparisons within an item. Table 5 compares each word's similarity to a synonym and a competing word, keeping the query fixed. The competitor is a one-edit word, an antonym, or the base form of a negation. Two-sided exact sign tests assess whether the synonym is nearer in more or fewer than half the items. All vision models favor one-edit words over synonyms, although synonym similarity is higher in DINOv3-H, DINOv3-7B, CLIP-B, and SigLIP2-B than in the other vision models. No vision model favors synonyms over antonyms after Holm correction. Negated words are usually closer to their base forms than to their synonyms in DINOv3, CLIP-B, SigLIP2-B, and Web-DINO-1B; DINOv2, MAE, and the untrained ViT-B show no clear preference.

DINOv2 and MAE retrieve one-edit words more readily than plurals or base forms (Table 4). The way we center the words matters for this comparison. Most one-edit words differ by a single letter without changing word length, so much of the text remains in the same position. Plurals and base forms add or remove letters, changing the word's width. Centering then shifts the whole word, including the letters shared by both forms, and reduces their visual overlap. Retrieving one-edit words therefore does not, on its own, demonstrate recognition of similar spelling beyond this overlap.

Table 4: Partner retrieval at the final layer. Top-10 accuracy is the fraction of queries whose partner is among the ten nearest candidates in the full vocabulary, excluding the query. Queries are (a) the first word of each pair (anchor, negation, singular noun, or country) and (b) its partner. Small numbers are standard errors; underlining marks values that exceed the untrained ViT-B by more than twice the standard error of the difference. Bold marks each column's maximum within a panel. Text and speech encoders are excluded from highlighting.

	Meaning					Spelling		
Model	Capital	Synonym	Synonym (antonym list)	Antonym	Synonym (negation list)	One-edit word	Plural	Base
(a) Forward								
DINOv3-B	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.653 ±.039	0.450 ±.079	0.200 ±.068
DINOv3-L	0.117 ±.041	0.013 ±.009	0.000 ±.000	0.020 ±.020	0.029 ±.028	0.820 ±.031	0.950 ±.034	0.486 ±.084
DINOv3-H	**0.600** ±.063	0.060 ±.019	0.020 ±.020	0.000 ±.000	0.086 ±.047	0.607 ±.040	**1.000** ±.000	0.600 ±.083
DINOv3-7B	**0.600** ±.063	0.100 ±.024	0.020 ±.020	0.060 ±.034	0.086 ±.047	0.620 ±.040	**1.000** ±.000	0.686 ±.078
Web-DINO-1B	0.000 ±.000	0.007 ±.007	0.020 ±.020	0.000 ±.000	0.000 ±.000	0.547 ±.041	0.475 ±.079	0.086 ±.047
DINOv2-B	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.029 ±.028	0.353 ±.039	0.100 ±.047	0.029 ±.028
DINOv2-L	0.017 ±.017	0.007 ±.007	0.020 ±.020	0.000 ±.000	0.000 ±.000	0.440 ±.041	0.125 ±.052	0.057 ±.039
DINOv2-G	0.000 ±.000	0.013 ±.009	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.367 ±.039	0.125 ±.052	0.029 ±.028
MAE-B	0.000 ±.000	0.000 ±.000	0.020 ±.020	0.020 ±.020	0.000 ±.000	0.507 ±.041	0.050 ±.034	0.029 ±.028
MAE-L	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.029 ±.028	0.547 ±.041	0.125 ±.052	0.029 ±.028
MAE-H	0.017 ±.017	0.007 ±.007	0.000 ±.000	0.000 ±.000	0.029 ±.028	0.553 ±.041	0.125 ±.052	0.057 ±.039
CLIP-B	0.550 ±.064	**0.200** ±.033	**0.200** ±.057	**0.120** ±.046	**0.143** ±.059	0.853 ±.029	**1.000** ±.000	0.971 ±.028
SigLIP2-B	0.183 ±.050	0.060 ±.019	0.020 ±.020	0.020 ±.020	0.086 ±.047	**0.947** ±.018	**1.000** ±.000	**1.000** ±.000
Untrained ViT-B	0.017 ±.017	0.007 ±.007	0.000 ±.000	0.020 ±.020	0.000 ±.000	0.333 ±.038	0.025 ±.025	0.086 ±.047
Qwen3-0.6B	0.600 ±.063	0.687 ±.038	0.920 ±.038	0.220 ±.059	0.943 ±.039	0.213 ±.033	1.000 ±.000	0.771 ±.071
Qwen3-4B	0.983 ±.017	0.733 ±.036	0.960 ±.028	0.220 ±.059	0.914 ±.047	0.260 ±.036	1.000 ±.000	0.771 ±.071
Qwen3-8B	0.967 ±.023	0.740 ±.036	0.960 ±.028	0.120 ±.046	0.857 ±.059	0.240 ±.035	1.000 ±.000	0.714 ±.076
w2v-BERT 2.0	0.000 ±.000	0.013 ±.009	0.020 ±.020	0.000 ±.000	0.029 ±.028	0.453 ±.041	0.225 ±.066	0.257 ±.074
Chance	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.006
(b) Reverse								
DINOv3-B	0.033 ±.023	0.000 ±.000	0.020 ±.020	0.000 ±.000	0.029 ±.028	0.687 ±.038	0.475 ±.079	0.257 ±.074
DINOv3-L	0.050 ±.028	0.013 ±.009	0.000 ±.000	0.000 ±.000	0.057 ±.039	0.807 ±.032	**1.000** ±.000	0.571 ±.084
DINOv3-H	0.483 ±.065	0.113 ±.026	0.020 ±.020	0.020 ±.020	0.114 ±.054	0.653 ±.039	0.950 ±.034	0.800 ±.068
DINOv3-7B	0.450 ±.064	0.133 ±.028	0.040 ±.028	0.020 ±.020	0.114 ±.054	0.720 ±.037	**1.000** ±.000	0.943 ±.039
Web-DINO-1B	0.000 ±.000	0.013 ±.009	0.000 ±.000	0.000 ±.000	0.029 ±.028	0.580 ±.040	0.650 ±.075	0.229 ±.071
DINOv2-B	0.000 ±.000	0.007 ±.007	0.000 ±.000	0.000 ±.000	0.029 ±.028	0.373 ±.039	0.125 ±.052	0.000 ±.000
DINOv2-L	0.000 ±.000	0.013 ±.009	0.040 ±.028	0.000 ±.000	0.057 ±.039	0.407 ±.040	0.100 ±.047	0.029 ±.028
DINOv2-G	0.000 ±.000	0.020 ±.011	0.020 ±.020	0.000 ±.000	0.029 ±.028	0.380 ±.040	0.125 ±.052	0.057 ±.039
MAE-B	0.000 ±.000	0.013 ±.009	0.020 ±.020	0.000 ±.000	0.000 ±.000	0.500 ±.041	0.225 ±.066	0.229 ±.071
MAE-L	0.000 ±.000	0.007 ±.007	0.020 ±.020	0.000 ±.000	0.029 ±.028	0.547 ±.041	0.225 ±.066	0.229 ±.071
MAE-H	0.000 ±.000	0.020 ±.011	0.020 ±.020	0.000 ±.000	0.000 ±.000	0.573 ±.040	0.275 ±.071	0.343 ±.080
CLIP-B	**0.600** ±.063	**0.173** ±.031	**0.180** ±.054	**0.120** ±.046	**0.143** ±.059	0.893 ±.025	**1.000** ±.000	**0.971** ±.028
SigLIP2-B	0.150 ±.046	0.067 ±.020	0.060 ±.034	0.040 ±.028	0.114 ±.054	**0.920** ±.022	**1.000** ±.000	**0.971** ±.028
Untrained ViT-B	0.000 ±.000	0.000 ±.000	0.000 ±.000	0.020 ±.020	0.000 ±.000	0.327 ±.038	0.025 ±.025	0.086 ±.047
Qwen3-0.6B	0.583 ±.064	0.680 ±.038	0.900 ±.042	0.300 ±.065	0.971 ±.028	0.220 ±.034	1.000 ±.000	0.857 ±.059
Qwen3-4B	0.983 ±.017	0.740 ±.036	0.960 ±.028	0.260 ±.062	0.971 ±.028	0.320 ±.038	1.000 ±.000	0.886 ±.054
Qwen3-8B	0.983 ±.017	0.767 ±.035	0.960 ±.028	0.200 ±.057	0.971 ±.028	0.273 ±.036	1.000 ±.000	0.886 ±.054
w2v-BERT 2.0	0.000 ±.000	0.013 ±.009	0.020 ±.020	0.020 ±.020	0.000 ±.000	0.533 ±.041	0.250 ±.068	0.171 ±.064
Chance	0.006	0.006	0.006	0.006	0.006	0.006	0.006	0.006

Table 5: Word-pair comparisons at the final layer. Counts give the number of items in which the synonym is closer to the query than the named competitor. Cosine similarities to the anchor are expressed in standard deviations above the vocabulary-wide pairwise mean, with standard errors over items. Top-10 hits count successful synonym retrieval in either direction among 75 concrete and 75 abstract anchors. Stars mark uncorrected $p < 0.05$ favoring synonyms (two-sided sign test) or concrete anchors (one-sided Fisher exact test). Underlining marks similarities that exceed the untrained ViT-B by more than twice the standard error of the difference. Bold marks the largest synonym similarity and the largest hit count across both anchor groups. Text and speech encoders are excluded from tests and highlighting.

	Synonym against one-edit word			Synonym against antonym	Synonym against base	Top-10 hits for synonyms	
Model	Count of 150	One-edit word	Synonym	Count of 50	Count of 35	Concrete	Abstract
DINOv3-B	1	2.32 ±.04	0.22 ±.08	31	3	0	0
DINOv3-L	2	3.24 ±.06	0.34 ±.09	30	3	3	0
DINOv3-H	14	3.01 ±.11	0.91 ±.11	27	2	14*	3
DINOv3-7B	17	2.82 ±.10	0.88 ±.11	23	2	**17***	5
Web-DINO-1B	5	2.30 ±.06	0.31 ±.08	30	8	1	1
DINOv2-B	6	1.85 ±.07	0.25 ±.08	31	16	1	0
DINOv2-L	9	2.03 ±.07	0.27 ±.08	34*	19	2	1
DINOv2-G	12	1.60 ±.05	0.33 ±.08	31	17	4	1
MAE-B	10	1.58 ±.04	0.39 ±.08	27	14	1	1
MAE-L	5	1.68 ±.04	0.43 ±.08	30	13	1	0
MAE-H	7	1.63 ±.03	0.46 ±.08	27	15	3	1
CLIP-B	13	5.51 ±.15	**1.75** ±.14	33*	1	15	**17**
SigLIP2-B	1	5.07 ±.09	1.01 ±.10	29	1	6	8
Untrained ViT-B	30	1.79 ±.12	0.33 ±.09	24	15	1	0
Qwen3-0.6B	119	2.10 ±.12	4.58 ±.19	48	22	48	60
Qwen3-4B	118	2.60 ±.14	5.50 ±.22	49	20	55	59
Qwen3-8B	124	2.45 ±.14	5.86 ±.22	50	26	53	65
w2v-BERT 2.0	16	2.25 ±.09	0.24 ±.09	27	4	2	0
Chance	75	0.00	0.00	25	17.5	0.96	0.96

C.2 GROUPING BY CATEGORY

Held-out words. Table 6 reports neighbor purity and nearest-centroid accuracy for nouns and brands. The latter tests whether category grouping extends to held-out words. We split the words evenly within each combination of category and spelling cluster, then average the unit-length word directions in one half to form the centroids. Each word in the other half is assigned to the nearest centroid by cosine similarity. DINOv3-H, DINOv3-7B, CLIP-B, and SigLIP2-B have the highest accuracy and neighbor purity among the vision models for both nouns and brands, followed by DINOv3-L.

Table 6: Category grouping at the final layer. Neighbor purity is the fraction of a word's ten nearest neighbors in the noun or brand list that share its category or industry. Nearest-centroid accuracy measures category assignment on held-out words. Permutation tests control for word length (Appendix C.2); small numbers are standard errors. Underlining marks values that exceed the untrained ViT-B by more than twice the standard error of the difference. Bold marks column maxima. Text and speech encoders are excluded from tests and highlighting.

	Neighbor purity				Nearest-centroid accuracy			
	Nouns		Brands		Nouns		Brands	
Model	Value	p	Value	p	Value	p	Value	p
DINOv3-B	0.194 ±.012	0.0010	0.184 ±.016	0.0345	0.214 ±.030	0.1594	0.219 ±.042	0.2964
DINOv3-L	0.223 ±.014	0.0005	0.235 ±.023	0.0005	0.297 ±.033	0.0005	0.281 ±.046	0.0165
DINOv3-H	0.269 ±.017	0.0005	**0.351** ±.027	0.0005	0.365 ±.035	0.0005	0.427 ±.050	0.0005
DINOv3-7B	0.308 ±.020	0.0005	0.334 ±.031	0.0005	0.396 ±.035	0.0005	0.469 ±.051	0.0005
Web-DINO-1B	0.187 ±.012	0.0180	0.167 ±.016	0.3873	0.198 ±.029	0.4278	0.198 ±.041	0.4353
DINOv2-B	0.179 ±.012	0.0925	0.164 ±.015	0.5182	0.245 ±.031	0.0420	0.188 ±.040	0.5317
DINOv2-L	0.174 ±.011	0.2884	0.167 ±.015	0.4153	0.245 ±.031	0.0490	0.208 ±.041	0.4123
DINOv2-G	0.181 ±.011	0.0695	0.173 ±.014	0.1879	0.271 ±.032	0.0090	0.188 ±.040	0.5447
MAE-B	0.182 ±.011	0.0630	0.182 ±.015	0.0340	0.250 ±.031	0.0325	0.167 ±.038	0.7141
MAE-L	0.186 ±.012	0.0230	0.188 ±.014	0.0100	0.255 ±.031	0.0155	0.188 ±.040	0.5042
MAE-H	0.180 ±.012	0.0980	0.183 ±.014	0.0340	0.229 ±.030	0.1259	0.229 ±.043	0.0875
CLIP-B	0.297 ±.019	0.0005	0.343 ±.027	0.0005	**0.474** ±.036	0.0005	**0.500** ±.051	0.0005
SigLIP2-B	**0.318** ±.019	0.0005	0.299 ±.025	0.0005	0.406 ±.035	0.0005	0.396 ±.050	0.0005
Untrained ViT-B	0.183 ±.011	0.0390	0.183 ±.016	0.0410	0.182 ±.028	0.7336	0.146 ±.036	0.9015
Qwen3-0.6B	0.709 ±.022	-	0.632 ±.031	-	0.906 ±.021	-	0.812 ±.040	-
Qwen3-4B	0.711 ±.021	-	0.746 ±.028	-	0.927 ±.019	-	0.948 ±.023	-
Qwen3-8B	0.711 ±.021	-	0.743 ±.027	-	0.922 ±.019	-	0.938 ±.025	-
w2v-BERT 2.0	0.201 ±.012	-	0.178 ±.017	-	0.266 ±.032	-	0.135 ±.035	-
Chance	0.164	-	0.162	-	0.167	-	0.167	-

Word form. The permutation tests in Table 6 assess category grouping while accounting for differences in word length across categories. We shuffle category labels within word-length terciles, also preserving spelling clusters for nearest-centroid accuracy. We use 2,000 permutations for each test. The observed scores for DINOv3-H, DINOv3-7B, CLIP-B, and SigLIP2-B exceed every permutation score for both measures on nouns and brands. The untrained ViT-B also shows significant neighbor purity, so we assess the trained models relative to this baseline (Section 4). Restricting noun neighbors to the query's spelling cluster likewise leaves DINOv3-H and DINOv3-7B above the untrained baseline.

C.3 Depth

Table 7 reports noun neighbor purity and country-to-capital retrieval at successive depths in each model. Noun purity increases from the middle layers onward in DINOv3-H, DINOv3-7B, CLIP-B, and SigLIP2-B. DINOv3-L separates from the untrained baseline mainly at the final layer, while the other vision models remain close to it throughout. Country-capital retrieval improves later: DINOv3-H and DINOv3-7B rise sharply at the last two sampled depths, while CLIP-B improves at the final layer. These models also begin to rank capitals first at those depths.

Table 7: Noun neighbor purity (a) and country-to-capital top-10 accuracy (b) across model depth. Relative depth is the layer index divided by the number of transformer layers; column labels give the mean across models at each sampled depth. The input embedding layer is excluded. Small numbers are standard errors. Underlining marks values that exceed the untrained ViT-B at the corresponding depth by more than twice the standard error of the difference. Bold marks each model's maximum across depths when it exceeds chance.

(a) Noun neighbor purity.

	Relative depth								
Model	0.10	0.23	0.33	0.44	0.56	0.67	0.77	0.90	1.00
DINOv3-B	.166 ±.009	.172 ±.010	.186 ±.011	.192 ±.012	.197 ±.012	.189 ±.012	**.198** ±.012	.195 ±.012	.194 ±.012
DINOv3-L	.185 ±.012	.180 ±.012	.176 ±.010	.183 ±.011	.181 ±.011	.198 ±.012	.193 ±.012	.207 ±.014	**.223** ±.014
DINOv3-H	.183 ±.011	.179 ±.011	.176 ±.011	.181 ±.011	.197 ±.012	.212 ±.013	.212 ±.014	.233 ±.015	**.269** ±.017
DINOv3-7B	.180 ±.011	.187 ±.010	.181 ±.012	.199 ±.012	.209 ±.013	.219 ±.015	.233 ±.015	.285 ±.018	**.308** ±.020
Web-DINO-1B	.178 ±.010	.179 ±.010	.176 ±.011	.189 ±.012	.197 ±.012	.198 ±.013	**.203** ±.013	.200 ±.013	.187 ±.012
DINOv2-B	.160 ±.010	.174 ±.010	.178 ±.011	.186 ±.011	.199 ±.013	.199 ±.013	**.199** ±.012	.191 ±.012	.179 ±.012
DINOv2-L	.158 ±.010	.166 ±.011	.166 ±.010	.179 ±.011	.192 ±.011	.186 ±.011	**.199** ±.013	.191 ±.012	.174 ±.011
DINOv2-G	.169 ±.010	.172 ±.010	.181 ±.011	**.187** ±.012	**.187** ±.012	.185 ±.012	.186 ±.012	.180 ±.011	.181 ±.011
MAE-B	.173 ±.011	.180 ±.011	**.188** ±.010	.171 ±.011	.179 ±.010	.180 ±.010	.186 ±.011	.184 ±.011	.182 ±.011
MAE-L	.185 ±.011	.180 ±.010	.183 ±.011	.184 ±.011	.188 ±.011	.192 ±.011	**.194** ±.011	.188 ±.012	.186 ±.012
MAE-H	.164 ±.010	.175 ±.011	.175 ±.010	.178 ±.011	.182 ±.010	**.184** ±.011	.184 ±.011	.182 ±.012	.180 ±.012
CLIP-B	.177 ±.009	.180 ±.011	.183 ±.011	.191 ±.012	.217 ±.014	.226 ±.014	.241 ±.016	.289 ±.019	**.297** ±.019
SigLIP2-B	.176 ±.010	.183 ±.011	.188 ±.012	.204 ±.012	.227 ±.014	.229 ±.015	.250 ±.018	.288 ±.020	**.318** ±.019
Untrained ViT-B	.171 ±.011	.175 ±.011	.176 ±.011	.176 ±.011	.179 ±.010	.180 ±.011	.179 ±.011	.182 ±.011	**.183** ±.011
Chance	.164	.164	.164	.164	.164	.164	.164	.164	.164

(b) Capital top-10 accuracy.

	Relative depth								
Model	0.10	0.23	0.33	0.44	0.56	0.67	0.77	0.90	1.00
DINOv3-B	.000 ±.000	**.017** ±.017	**.017** ±.017	.000 ±.000	**.017** ±.017	**.017** ±.017	**.017** ±.017	.000 ±.000	.000 ±.000
DINOv3-L	.017 ±.017	.000 ±.000	.000 ±.000	.033 ±.023	.017 ±.017	.017 ±.017	.017 ±.017	.033 ±.023	**.117** ±.041
DINOv3-H	.000 ±.000	.017 ±.017	.033 ±.023	.017 ±.017	.000 ±.000	.017 ±.017	.017 ±.017	.167 ±.048	**.600** ±.063
DINOv3-7B	.000 ±.000	.000 ±.000	.000 ±.000	.017 ±.017	.033 ±.023	.017 ±.017	.033 ±.023	.383 ±.063	**.600** ±.063
Web-DINO-1B	.017 ±.017	.017 ±.017	.017 ±.017	.017 ±.017	**.033** ±.023	.017 ±.017	.000 ±.000	.000 ±.000	.000 ±.000
DINOv2-B	.017 ±.017	.017 ±.017	**.033** ±.023	**.033** ±.023	.000 ±.000	.000 ±.000	.000 ±.000	.000 ±.000	.000 ±.000
DINOv2-L	.017 ±.017	.017 ±.017	.017 ±.017	**.050** ±.028	.000 ±.000	.000 ±.000	.017 ±.017	.017 ±.017	.017 ±.017
DINOv2-G	.017 ±.017	.000 ±.000	**.067** ±.032	.033 ±.023	.017 ±.017	.000 ±.000	.000 ±.000	.000 ±.000	.000 ±.000
MAE-B	.017 ±.017	.000 ±.000	.000 ±.000	.000 ±.000	.000 ±.000	.000 ±.000	**.033** ±.023	.000 ±.000	.000 ±.000
MAE-L	.000 ±.000	.000 ±.000	.033 ±.023	**.050** ±.028	.033 ±.023	.017 ±.017	.017 ±.017	.017 ±.017	.000 ±.000
MAE-H	.000 ±.000	.000 ±.000	.017 ±.017	**.033** ±.023	**.033** ±.023	**.033** ±.023	.017 ±.017	.017 ±.017	.017 ±.017
CLIP-B	.000 ±.000	.000 ±.000	.000 ±.000	.033 ±.023	.017 ±.017	.017 ±.017	.033 ±.023	.033 ±.023	**.550** ±.064
SigLIP2-B	.000 ±.000	.033 ±.023	.033 ±.023	.033 ±.023	.050 ±.028	.017 ±.017	.017 ±.017	.033 ±.023	**.183** ±.050
Untrained ViT-B	**.033** ±.023	.000 ±.000	.000 ±.000	.017 ±.017	.017 ±.017	**.033** ±.023	**.033** ±.023	.017 ±.017	.017 ±.017
Chance	.006	.006	.006	.006	.006	.006	.006	.006	.006

D WORD-CLASS ALIGNMENT

D.1 THE CLASS NAMES

Selection. A class name is a lowercase word of three to ten letters listed in WordNet for exactly one ImageNet-1k class. We exclude *crane*, for example, because it names both a bird and a machine. We take at most one name per class, preferring frequent and less ambiguous names, and require an estimated frequency of at least one occurrence per million words. We exclude inflected plurals, brands, offensive words, and words whose frequency comes from a proper name. Conventional object names such as *binoculars* are retained. A name must denote its class when read alone, and writing must not dominate the class photographs. We checked these judgments with a language model; they were not independently validated by human raters.

The 200 names. Of 384 candidate names, 224 satisfy the selection rules. We keep the 200 whose classes have the highest share of validation images with no detected text, to reduce the possibility that a rendered word aligns with writing in its class photographs. The names include animals such as *tiger*, devices such as *microphone*, vehicles, buildings, clothing, musical instruments, foods, and landscape words such as *volcano*. Forty-three names also occur in the vocabulary of Section 3.2 and use the same word images. Their word directions differ because the centering mean in (1) is taken over the full vocabulary for lexical comparisons and over the 200 names for word–class comparisons.

D.2 THE CLASS IMAGES

The class images come from the ImageNet-1k validation split, as do the 96 backgrounds, which are left out of the class images. Oquab et al. (2024) report removing near-duplicates of benchmark validation sets from the DINOv2 pretraining data, so these images are less likely to be in that data. Siméoni et al. (2026) describe the DINOv3 curation as following that work but do not report the same deduplication. Every class keeps at least 49 images, center-cropped and resized to 256 pixels as the backgrounds were.

Table 8: Word-class alignment at the final layer over the 200 class names. All images is the matched cosine similarity. Null mean is the mean over 2,000 random reassignments of classes to words, and p is the one-sided permutation probability. No detected text is the same measure over the 195 names whose classes keep at least 32 images with no detected text, with visual directions rebuilt from those images. Small numbers are standard errors over words. Underlining marks a value in the All images and No detected text columns above the untrained ViT-B by more than twice the standard error of the difference, and bold marks the largest value in those columns.

	Matched cosine similarity			
Model	All images	Null mean	p	No detected text
DINOv3-B	0.002 ±.003	-0.0001	0.237	0.002 ±.003
DINOv3-L	<u>0.025</u> ±.004	-0.0003	<0.001	0.023 ±.004
DINOv3-H	<u>0.234</u> ±.014	-0.0004	<0.001	<u>0.230</u> ±.015
DINOv3-7B	<u>0.120</u> ±.012	-0.0045	<0.001	<u>0.116</u> ±.013
Web-DINO-1B	0.001 ±.002	-0.0001	0.306	0.001 ±.002
DINOv2-B	-0.002 ±.003	0.0000	0.747	-0.002 ±.003
DINOv2-L	-0.001 ±.002	0.0000	0.667	-0.001 ±.002
DINOv2-G	0.001 ±.002	-0.0001	0.300	0.001 ±.002
MAE-B	-0.011 ±.006	-0.0002	0.958	-0.011 ±.006
MAE-L	0.003 ±.004	0.0001	0.254	0.004 ±.004
MAE-H	0.002 ±.004	0.0004	0.317	0.003 ±.004
CLIP-B	**<u>0.237</u>** ±.010	0.0001	<0.001	**<u>0.233</u>** ±.010
SigLIP2-B	<u>0.184</u> ±.009	0.0000	<0.001	<u>0.182</u> ±.009
Untrained ViT-B	0.005 ±.008	0.0030	0.419	0.006 ±.008
Chance	0.000			0.000

D.3 ALIGNMENT AND RETRIEVAL IN FULL

Table 8 gives word-class alignment and Table 9 retrieval in both directions, with standard errors and permutation probabilities as in Appendix B.2.

Images with no detected text. Writing in the class images could align a word direction with its class, so we rebuild each class mean from the images in which the CRAFT detector (Baek et al., 2019), at the settings used for the backgrounds, finds no text. Only classes that keep at least 32 such images enter, and we center the rebuilt class means over these classes to obtain their visual directions. Word directions keep their centering over all 200 names, and the names of the classes left out are dropped.

Background classes. We exclude the 96 background photographs from the class-image pool, so no photograph is used to estimate both a word direction and a visual direction. A background may still depict the class named by an overlaid word. This occurs for 22 names, each on one background. Recomputing their word directions from the other 95 backgrounds changes the mean matched cosine similarity over these 22 names by less than 0.006 in every model. The largest absolute change is 0.005803, in DINOv3-H. This check measures the effect of including the one matching background.

Class diagnostics. Two diagnostics describe the class images and exclude no class. Within-class cosine similarity is the mean cosine similarity between the representations of two images of one class, after subtracting the mean representation over all images. In DINOv3-H its median over the 200 named classes is 0.36, and it is lowest for classes such as *bee* and *church*. The second diagnostic is the cosine similarity between a visual direction and the nearest other visual direction. In DINOv3-H it is highest for pairs such as *corn* and *ear*, the class of an ear of corn, and *laptop* and *notebook*. *Ear* and *notebook* have no selected name but remain retrieval candidates, so a word direction for *corn* or *laptop* can rank a near class above its own.

Table 9: Retrieval in both directions at the final layer. Word to class ranks the 1,000 visual directions for each word direction, and class to word ranks the 200 word directions for each named class. Top-10 accuracy is the share of queries whose target is among the ten nearest, and median rank is taken over queries. No detected text gives top-10 accuracy with visual directions rebuilt from images with no detected text, over 195 names and 838 classes. Small numbers are standard errors over queries. Underlining marks an accuracy above the untrained ViT-B by more than twice the standard error of the difference, and bold marks the largest accuracy and the smallest median rank in each column.

	Word-to-class retrieval, 1,000 classes			Class-to-word retrieval, 200 class names		
Model	Top-10 accuracy	Median rank	No detected text	Top-10 accuracy	Median rank	No detected text
DINOv3-B	0.010 ±.007	486	0.010 ±.007	0.060 ±.017	94.5	0.067 ±.018
DINOv3-L	0.120 ±.023	335	0.103 ±.022	0.180 ±.027	68	0.179 ±.027
DINOv3-H	0.645 ±.034	**1**	0.651 ±.034	0.680 ±.033	**1**	0.677 ±.033
DINOv3-7B	0.325 ±.033	154	0.328 ±.034	0.405 ±.035	32.5	0.390 ±.035
Web-DINO-1B	0.020 ±.010	477	0.015 ±.009	0.045 ±.015	95	0.041 ±.014
DINOv2-B	0.010 ±.007	527.5	0.010 ±.007	0.060 ±.017	103.5	0.062 ±.017
DINOv2-L	0.000 ±.000	485	0.005 ±.005	0.045 ±.015	99.5	0.056 ±.017
DINOv2-G	0.005 ±.005	463	0.005 ±.005	0.045 ±.015	94	0.062 ±.017
MAE-B	0.005 ±.005	559	0.005 ±.005	0.020 ±.010	113	0.026 ±.011
MAE-L	0.015 ±.009	491	0.015 ±.009	0.045 ±.015	98.5	0.056 ±.017
MAE-H	0.005 ±.005	461.5	0.015 ±.009	0.050 ±.015	96.5	0.056 ±.017
CLIP-B	**0.855** ±.025	**1**	**0.851** ±.025	**0.845** ±.026	**1**	**0.831** ±.027
SigLIP2-B	0.630 ±.034	2	0.636 ±.034	0.735 ±.031	2	0.718 ±.032
Untrained ViT-B	0.005 ±.005	511.5	0.005 ±.005	0.045 ±.015	99	0.046 ±.015
Chance	0.010		0.012	0.050		0.051

D.4 ALIGNMENT BY DEPTH

We repeat word-class alignment and word-to-class retrieval at nine depths (Table 10). Through relative depth 0.77, matched cosine similarity stays within 0.032 of zero in every model, and the nine models whose final-layer alignment does not exceed its null stay that close at every depth. At every depth before the last, word-to-class top-10 accuracy is at most 0.055 in every model but SigLIP2-B, which at the next-to-last depth already reaches most of its final top-10 accuracy.

Table 10: Matched cosine similarity (a) and word-to-class top-10 accuracy (b) at nine depths over the 200 class names. Each column is one depth, labeled by the mean over models of the layer index divided by the layer count, and a column's depths differ across models by at most 0.042. The last two columns are one layer apart in the 12-layer models, three in the 24-layer models, and four in the 32- and 40-layer models. Chance is zero in (a) and 0.010 in (b). This table prints no standard errors, so no value is underlined, and bold marks each model's largest value across depths.

(a) Matched cosine similarity.

	Relative depth								
Model	0.10	0.23	0.33	0.44	0.56	0.67	0.77	0.90	1.00
DINOv3-B	-0.019	0.011	**0.022**	0.018	-0.003	0.012	0.007	-0.002	0.002
DINOv3-L	0.008	0.007	0.006	-0.002	0.003	0.002	0.012	0.013	**0.025**
DINOv3-H	0.016	0.029	-0.005	-0.003	0.000	0.005	0.013	0.041	**0.234**
DINOv3-7B	0.007	0.027	0.022	0.006	0.002	0.003	0.003	-0.006	**0.120**
Web-DINO-1B	0.014	**0.023**	0.020	0.007	0.005	0.007	0.003	0.003	0.001
DINOv2-B	-0.003	**0.011**	0.010	0.005	-0.009	-0.010	-0.009	-0.002	-0.002
DINOv2-L	-0.004	0.014	**0.016**	0.011	0.001	-0.000	0.000	-0.004	-0.001
DINOv2-G	0.005	**0.026**	0.021	0.005	0.007	-0.002	-0.001	0.001	0.001
MAE-B	-0.011	**0.031**	0.022	0.028	0.019	0.017	0.005	-0.005	-0.011
MAE-L	**0.026**	0.021	0.018	0.013	0.008	0.004	-0.000	-0.001	0.003
MAE-H	0.008	0.017	0.014	**0.019**	0.008	0.003	0.003	0.003	0.002
CLIP-B	0.022	0.030	0.022	0.014	0.016	0.007	0.003	0.036	**0.237**
SigLIP2-B	0.032	0.020	0.013	0.011	-0.001	-0.002	0.008	0.114	**0.184**
Untrained ViT-B	0.002	0.002	0.000	**0.007**	0.006	0.004	0.005	0.006	0.005

(b) Word-to-class top-10 accuracy.

	Relative depth								
Model	0.10	0.23	0.33	0.44	0.56	0.67	0.77	0.90	1.00
DINOv3-B	0.005	**0.020**	0.005	0.015	0.005	0.015	**0.020**	0.010	0.010
DINOv3-L	0.000	0.000	0.005	0.010	0.005	0.020	0.025	0.030	**0.120**
DINOv3-H	0.015	0.020	0.010	0.010	0.005	0.015	0.010	0.050	**0.645**
DINOv3-7B	0.010	0.000	0.005	0.025	0.010	0.005	0.010	0.015	**0.325**
Web-DINO-1B	0.000	0.010	0.000	0.005	0.005	**0.025**	0.020	0.010	0.020
DINOv2-B	0.005	0.005	0.005	0.010	**0.020**	**0.020**	0.015	0.005	0.010
DINOv2-L	**0.010**	**0.010**	0.000	0.005	0.005	0.000	0.000	0.000	0.000
DINOv2-G	0.005	0.005	**0.015**	0.005	0.005	0.000	0.000	0.005	0.005
MAE-B	0.010	0.010	0.000	0.010	**0.020**	0.015	0.010	0.010	0.005
MAE-L	**0.020**	0.010	0.005	**0.020**	0.015	0.005	0.015	0.010	0.015
MAE-H	0.000	0.010	**0.015**	0.010	0.000	**0.015**	0.010	0.005	0.005
CLIP-B	0.005	0.000	0.015	0.005	0.005	0.005	0.015	0.055	**0.855**
SigLIP2-B	0.020	0.010	0.000	0.005	0.005	0.005	0.020	0.465	**0.630**
Untrained ViT-B	0.010	**0.015**	0.000	0.000	0.005	0.005	0.005	0.005	0.005
Chance	0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010	0.010

D.5 TWO-CLUE ANALYSIS

We test whether two words shown together identify a target more clearly than either word alone. For example, *input* and *sound* together suggest *microphone*. We ask whether each clue improves the pair's alignment with the target relative to competing words. Only CLIP-B shows a contribution from both clues after correction across the thirteen pretrained models (Figure 5).

Table 11: Example clue grid. Each row and column supplies one clue, and each cell contains a target word. For *microphone* (yellow), the broad-clue comparison uses *speaker* (pink), and the narrow-clue comparison uses the three other input devices (blue). Gray targets share neither clue and enter only the comparison with all seven alternatives.

	Narrow clue			
Broad clue	*sound*	*picture*	*text*	*motion*
input	***microphone***	*camera*	*keyboard*	*sensor*
output	*speaker*	*projector*	*printer*	*motor*

Target · Narrow comparison · Broad comparison · Neither clue

Design. Table 11 illustrates how we choose the clues and targets. Each grid combines two broad clues with four narrow clues, giving eight targets. In this example, the broad clue specifies whether a device takes a signal in or gives it out, and the narrow clue specifies the signal type. Neither clue identifies a target on its own: *sound* could mean *microphone* or *speaker*, while *input* could mean any of the four devices in the first row. The analysis includes 96 items from twelve grids. Six pilot grids used during development are evaluated separately and excluded from the reported results.

Renderings. Table 12 shows the four versions of each clue pair: both clues intact, only the broad clue intact, only the narrow clue intact, and neither intact. We replace each absent clue with a phase-randomized version of its letters, following the control procedure in Appendix A. We render all four versions on the same 96 backgrounds and render the target word separately with its own control. For each condition, we subtract the control representation and average the differences over backgrounds. With f denoting the image encoder, we abbreviate these averages as

$$\begin{aligned} \hat{\tau}_c &= f(c\ \text{intact}) - f(\text{neither intact}), \qquad c \in \{\text{both},\ \text{broad},\ \text{narrow}\}, \\ \hat{\tau}_{\text{target}} &= f(\text{target}) - f(\text{target control}). \end{aligned} \tag{5}$$

We compare these vectors only with the eight target words in the same grid and do not center them over a vocabulary.

Table 12: Renderings for the clue pair *input* and *sound* and its target, *microphone*. Gray bars indicate phase-randomized letters. Difference vectors are defined in (5).

Rendering	Broad clue	Narrow clue	Difference vector
Both intact	*input*	*sound*	$\hat{\tau}_{\text{both}}$, used in both clue comparisons
Broad intact	*input*		$\hat{\tau}_{\text{broad}}$, baseline for the narrow-clue gain
Narrow intact		*sound*	$\hat{\tau}_{\text{narrow}}$, baseline for the broad-clue gain
Neither intact			The control subtracted from the three difference vectors above
Target alone	*microphone*		$\hat{\tau}_{\text{target}}$, against its own control image

Clue gains. We measure each clue's contribution by comparing the target with words that share the other clue. For *microphone*, the broad clue *input* must distinguish it from *speaker*, which also matches *sound*. The narrow clue *sound* must distinguish it from *camera*, *keyboard*, and *sensor*, which all match *input*. We call these the broad and narrow comparisons, respectively.

For each clue condition, the margin is the cosine similarity of its difference vector to the target vector minus the mean similarity to the competing target vectors. The clue gain is the increase in this margin when the missing clue is added. Thus, the broad-clue gain compares $\hat{\tau}_{\text{both}}$ with $\hat{\tau}_{\text{narrow}}$, and the narrow-clue gain compares $\hat{\tau}_{\text{both}}$ with $\hat{\tau}_{\text{broad}}$. We report the smaller gain and apply Holm correction to the larger of the two p-values across the thirteen pretrained models. This requires evidence that

both clues contribute. A single margin against all seven alternatives cannot establish this, because the narrow clue alone distinguishes the target from six of them.

Results. Figure 5 shows the gain from adding each clue. CLIP-B is the only model with significant gains for both clues after correction. SigLIP2-B has bootstrap intervals above zero for both gains, but does not pass the corrected test. DINOv3-H and DINOv3-7B have intervals above zero only for the narrow-clue gain, and the reported tests do not establish that these gains exceed the untrained ViT-B's. For these two models, the test detects a contribution from the narrow clue but no contribution from the broad clue that distinguishes targets sharing the narrow clue.

We also test whether the pair contributes anything beyond the sum of the two single-clue vectors. We subtract that sum from the pair vector and measure the residual's margin against all seven competing targets. No model has a residual margin significantly different from zero after correction. CLIP-B and MAE-H have negative residual margins with bootstrap intervals below zero, but these do not pass the corrected test either. The absence of a significant residual margin does not establish that the pair vector equals the sum of the single-clue vectors.

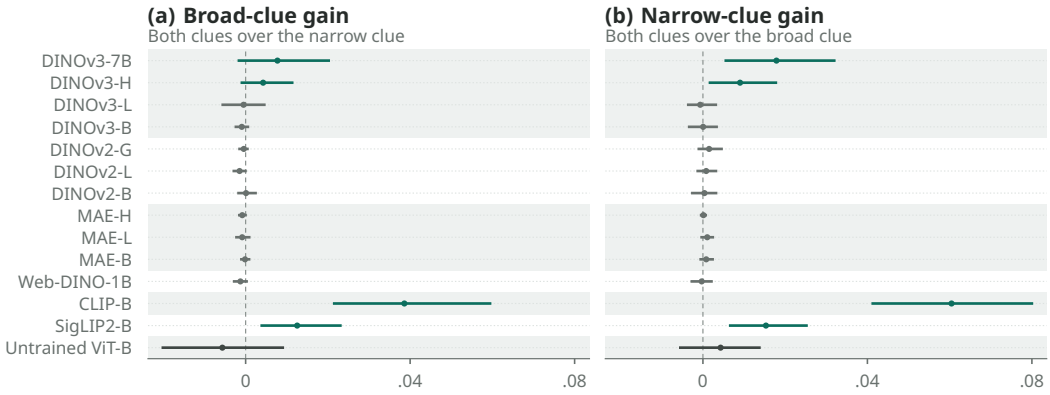


Figure 5: **Contribution of each clue to target-word alignment.** Points show clue gains, with 95% bootstrap intervals over the twelve grids. The dashed line marks zero gain. Colored points identify the four models whose margin against all seven alternatives has a corrected $p < 0.10$; the untrained ViT-B is shown in darker gray. **(a)** Broad-clue gain: both clues compared with the narrow clue alone. **(b)** Narrow-clue gain: both clues compared with the broad clue alone.

Limits. These comparisons concern alignment with a rendered target word; they do not test alignment with images of the object it names. Word co-occurrence during training could also account for the gains. The items were selected by the authors, and the small number of independent grids limits generalization. Visual differences beyond letter identity may also affect the comparison: they occur at both clue positions in the pair image, but at only one in each single-clue image.

E MODELS, ENCODERS, AND FINETUNING

E.1 MODELS AND REPRESENTATIONS

Representation. Table 13 lists the fourteen models and the four encoders of Appendix E.2. The representation f of Section 3.3 is the hidden state after the final transformer block, before the model's final layer normalization. It is the class token in every model but SigLIP2-B, which has no class token and uses the mean of the patch tokens, and neither CLIP-B's visual projection nor SigLIP2-B's attention pooling head is applied.

Table 13: The fourteen models and, under the rule, the three text encoders and the speech encoder, with the checkpoint identifier, layer count, and width of each. The untrained ViT-B takes only its architecture from the listed configuration, and its weights are randomly initialized. A speech vector concatenates four relative-position bins of the listed width, so it has 4,096 dimensions.

Model	Checkpoint or configuration identifier	Layers	Width
DINOv3-B	`facebook/dinov3-vitb16-pretrain-lvd1689m`	12	768
DINOv3-L	`facebook/dinov3-vitl16-pretrain-lvd1689m`	24	1024
DINOv3-H	`facebook/dinov3-vith16plus-pretrain-lvd1689m`	32	1280
DINOv3-7B	`facebook/dinov3-vit7b16-pretrain-lvd1689m`	40	4096
Web-DINO-1B	`facebook/webssl-dino1b-full2b-224`	40	1536
DINOv2-B	`facebook/dinov2-base`	12	768
DINOv2-L	`facebook/dinov2-large`	24	1024
DINOv2-G	`facebook/dinov2-giant`	40	1536
MAE-B	`facebook/vit-mae-base`	12	768
MAE-L	`facebook/vit-mae-large`	24	1024
MAE-H	`facebook/vit-mae-huge`	32	1280
CLIP-B	`openai/clip-vit-base-patch16`	12	768
SigLIP2-B	`google/siglip2-base-patch16-224`	12	768
Untrained ViT-B	`google/vit-base-patch16-224` (configuration only)	12	768
Qwen3-0.6B	`Qwen/Qwen3-Embedding-0.6B`	28	1024
Qwen3-4B	`Qwen/Qwen3-Embedding-4B`	36	2560
Qwen3-8B	`Qwen/Qwen3-Embedding-8B`	36	4096
w2v-BERT 2.0	`facebook/w2v-bert-2.0`	24	1024

Input processing. Every image is 256×256 pixels, and we give each model the preprocessing published with its checkpoint. For DINOv2 the image is center-cropped to 224×224, which keeps the text at its rendered size and trims the edge of the background. For the other models it is resized to 224×224, with bicubic interpolation for Web-DINO-1B and CLIP-B and bilinear interpolation for the rest. Color channels are normalized with the means and standard deviations published with each checkpoint.

Averaging and precision. Representations are not scaled to unit length before paired subtraction or class averaging. Every model runs in float32 except DINOv3-7B, which runs in bfloat16. MAE runs with a mask ratio of zero, so every patch is kept. Depth profiles read the same token at nine depths, from about a tenth of the blocks to the final block (Table 10). The untrained ViT-B has the ViT-B architecture of Table 13, with 12 attention heads and 16×16 patches, one random initialization, and the preprocessing of the ViT-B checkpoint.

E.2 TEXT AND SPEECH VECTORS

Carriers. A text or speech encoder never sees a rendered image, so its vector for a word, name, or pseudoword is built by placing it inside a fixed set of carrier sentences, as a word direction is built on a fixed set of backgrounds. The set is the same across the vocabulary, because a carrier chosen per word, such as a sentence in which the word occurs, would bring corpus co-occurrence into the measurement. The carriers vary in syntactic type, because an average keeps whatever its carriers share, and carriers of one type would move every average toward that type.

Text. Each of the 1,561 words, names, and pseudowords is placed in 96 carriers, the 24 templates of Table 14 in four variants each. The variants differ in punctuation, tense, or a function word, as in "The word is *w*." and "The word was *w*.". The templates supply the variation and the variants are replicates, as the typography settings and their four backgrounds are for the word images. Every target position is quoted, named, or otherwise permissive, so no carrier is ungrammatical for an adjective, a verb, a brand, a capital, or a pseudoword. A carrier's vector is the frozen encoder's state at the last token, the pooling that Qwen3-Embedding-0.6B, -4B, and -8B were trained with. We add no instruction prefix, since the prefix is meant only for queries in asymmetric retrieval, and this comparison is symmetric.

Table 14: The 24 carrier templates for the text encoders, shown in their first variant with *w* standing for the word, name, or pseudoword. Each template has four variants, for 96 carriers per word.

#	Template	#	Template
1	*w*	13	They wrote *w*.
2	"*w*"	14	Someone said *w*.
3	the word "*w*"	15	She read *w* out.
4	The word is *w*.	16	He typed *w* quickly.
5	Read *w* aloud.	17	I heard *w* again.
6	Spell *w*.	18	*w*, and then *w*.
7	*w* is spelled out.	19	First *w*. Then the rest.
8	It says *w* there.	20	*w*, *w*, *w*.
9	*w* is what it says.	21	Next up: *w*.
10	This one is *w*.	22	Here it is, *w*.
11	That would be *w*.	23	A sign showing *w*.
12	The answer is *w*.	24	Written on it: *w*.

Speech. Each word, name, or pseudoword is spoken in 96 utterances by a VITS synthesizer trained on VCTK, in eight voices at three speaking rates (0.9, 1.0, and 1.1) and four carrier sentences that place the word first, in the middle, or last: "*w* was written there.", "They wrote *w* on it.", "Someone said *w* out loud.", and "The label reads *w*.". Synthesis adds no sampling noise, so an utterance is fixed by its voice, rate, and carrier. The frozen encoder is w2v-BERT 2.0 (SEAMLESS Communication Team et al., 2025), and only the frames of the word are kept, located by the synthesizer's phoneme durations. These frames are pooled into four bins of equal relative position and concatenated, so the vector keeps where in the word each sound falls, which a plain mean over frames would discard. For text and speech alike, the 96 vectors for each word are averaged and the mean is centered over the full vocabulary of 1,561 words, names, and pseudowords.

E.3 Contrastive finetuning

Objective. A run trains one of the eleven self-supervised models end to end against a frozen encoder. The image side is the model's class token passed through a linear projection, without bias, to the encoder width of 1,024. The caption side is the frozen encoder's vector for the caption. A written caption is embedded by Qwen3-Embedding-0.6B (Zhang et al., 2025), read at the last token without an instruction prefix, and a spoken caption by w2v-BERT 2.0 (SEAMLESS Communication Team et al., 2025), as the mean of its frames over the utterance. Both sides are scaled to unit length, and the loss is symmetric InfoNCE over a batch of 1,056 pairs, with a learned inverse temperature initialized at $1/0.07$ and capped at 100 (Radford et al., 2021).

Optimization and schedule. The optimizer is AdamW with $\beta = (0.9, 0.98)$, $\epsilon = 10^{-6}$, and weight decay 0.1, with the gradient norm clipped at 1.0. The learning rate is 5×10^{-5} for the model and 5×10^{-4} for the projection and temperature, with linear warmup and then cosine decay to zero, under bfloat16 mixed precision. Each training image is a random crop covering 90 to 100 percent of its area, resized to 224 pixels with bicubic interpolation and normalized with the ImageNet channel statistics, with no other augmentation. The schedule is three epochs of the filtered corpus, 1,310 steps (Table 15). Warmup takes a tenth of the schedule and the interval between validations on held-out pairs a twelfth, 131 and 109 steps. A run keeps the weights with the best validation loss and stops after three validations without improvement. The projection and temperature are discarded after training, so a finetuned model differs from its pretrained model only in its weights.



Figure 6: **Finetuning raises noun purity and synonym retrieval in DINOv3-H.** Open circles show scores before finetuning; filled circles show scores after training with written (upper, dark) or spoken (lower, light) captions. Panels show **(a)** noun neighbor purity, **(b)** synonym top-10 accuracy, and **(c)** agreement between word directions estimated from two sets of backgrounds (split-half cosine similarity). Scores use four word images, averaged over five randomly sampled typography settings. Gray vertical lines show scores before finetuning using all 96 word images. The dashed line marks chance noun purity.

Corpus and text filter. SPEECH-COCO (Havard et al., 2017) pairs each COCO image (Lin et al., 2014) with about five synthesized spoken captions, each a reading of one of that image's COCO captions (Chen et al., 2015) in one of eight named voices at one of three speaking rates. The speech run takes the recording and the written run the text of the same caption, so the two runs see the same images and captions. Every image is scanned with the CRAFT detector (Baek et al., 2019) at the settings used for the backgrounds, and any image with detected text is excluded, which leaves 461,195 captions, each with its recording. The filter is required because captions often transcribe writing in the image, and a model trained on such pairs could group rendered words by matching them to captions. Validation holds out two percent of the images with all their captions, so no image is on both sides.

Checkpoints. Table 15 gives the step of the checkpoint kept for every run. Every self-supervised model but DINOv3-7B is trained once against each of the two encoders, and every run keeps its weights from the step with the best validation loss. Four of the runs that stopped early did so at step 1,308, the last validation of the schedule.

Table 15: The step of the checkpoint kept for each run. An asterisk marks an early stop.

	SPEECH-COCO	
Model	written	speech
DINOv3-B	1,199	1,308
DINOv3-L	763*	1,199
DINOv3-H	436*	981*
Web-DINO-1B	218*	545*
DINOv2-B	1,199	1,090
DINOv2-L	981*	981*
DINOv2-G	218*	872*
MAE-B	1,308	1,199
MAE-L	872*	981*
MAE-H	763*	872*

Before and after finetuning. Figure 6 shows every finetuned model in three readouts before and after both runs. A finetuned model is measured exactly as its pretrained model, with the same word images, control images, and preprocessing. Word directions from four word images use the four backgrounds of one typography setting. The setting is drawn at random five times, with the same draws before and after finetuning,

and the five draws give four distinct settings, one of them twice. A top-10 accuracy from four word images is the mean over the draws. In DINOv3-H the standard deviation of capital top-10 accuracy over the draws is 0.10 before finetuning, larger than its change after the written run. Mutual 10-NN alignment and linear CKA are computed against the same text and speech encoders as in Figure 3.

F AI USE STATEMENT

We used generative AI tools to assist in writing code across the experimental pipeline, including preprocessing, rendering, representation extraction, finetuning, statistical analysis, and plotting. They also assisted in editing and reviewing manuscript text, finding and reviewing literature, and checking and formatting references. Language models proposed synonym pairs, antonym and negation triples, and noun candidates, as described in Appendix A. The authors reviewed every AI-proposed item retained in the dataset.

The authors reviewed all AI-assisted work, checked the code and its outputs, and verified cited claims against their sources and bibliography entries against publisher records. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

G REPRODUCIBILITY STATEMENT

Code and reproduction instructions is available at:

`https://anonymous.4open.science/r/vision-word-semantics-345F`

The release will include environment and configuration files, curated word lists, and instructions for obtaining external datasets and model weights. Appendix A describes the data and rendering procedures; Appendix B defines the measures and statistical tests; and Appendix E specifies model checkpoints, input processing, encoders, and the finetuning procedure.