Steerable Visual Representations

Jona Ruthardt[1*], Manu Gaur[2*],
Deva Ramanan[2], Makarand Tapaswi[3†], and Yuki M. Asano[1†]

[1] University of Technology Nuremberg
[2] Carnegie Mellon University
[3] International Institute of Information Technology, Hyderabad

Abstract. Pretrained Vision Transformers (ViTs) such as DINOv2 and MAE provide generic image features that can be applied to a variety of downstream tasks such as retrieval, classification, and segmentation. However, such representations tend to focus on the most salient visual cues in the image, with no way to direct them toward less prominent concepts of interest. In contrast, Multimodal LLMs can be guided with textual prompts, but the resulting representations tend to be language-centric and lose their effectiveness for generic visual tasks. To address this, we introduce *Steerable Visual Representations*, a new class of visual representations, whose global and local features can be steered with natural language. While most vision-language models (e.g., CLIP) fuse text with visual features after encoding (*late* fusion), we inject text directly into the layers of the visual encoder (*early* fusion) via lightweight cross-attention. We introduce benchmarks for measuring *representational steerability*, and demonstrate that our steerable visual features can focus on any desired objects in an image while preserving the underlying representation quality. Our method also matches or outperforms dedicated approaches on anomaly detection and personalized object discrimination, exhibiting zero-shot generalization to out-of-distribution tasks.

Project Website: jonaruthardt.github.io/project/SteerViT

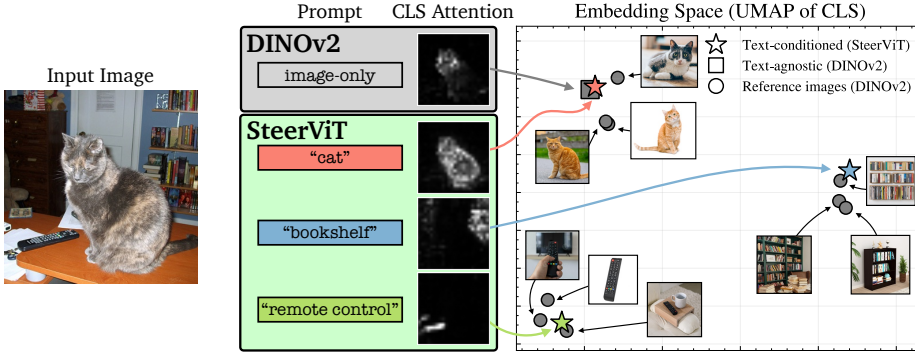


Fig. 1: Steering visual representations with language. While DINOv2 primarily encodes the salient object, producing a "cat" representation, SteerViT can be steered with text to shift its attention (middle) and global feature semantics (right) towards the queried visual concept (e.g., "bookshelf" or "remote control").

[*] Equal contribution. [†] Equal advising.

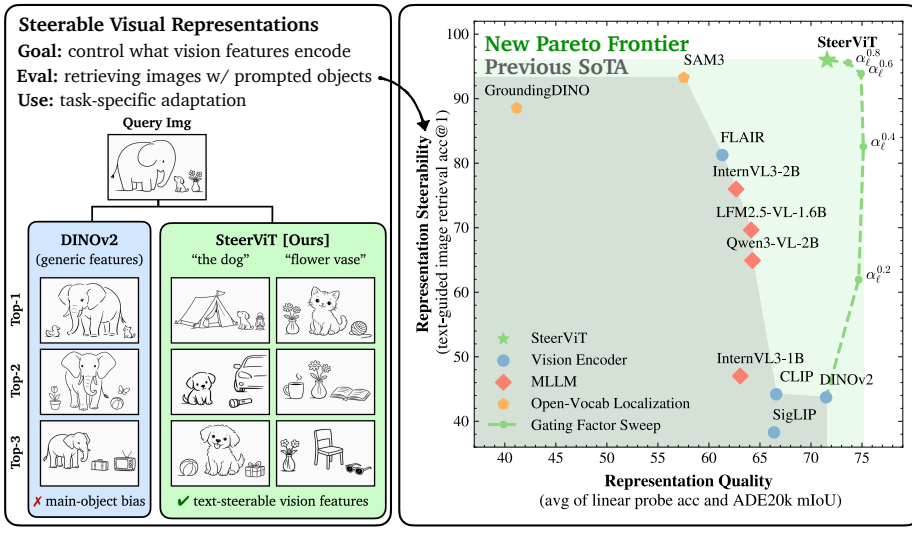


Fig. 2: SteerViT produces high-quality visual representations that can be steered by text. Left: Traditional (non-steerable) representations like DINOv2 tend to focus on the dominant object in an image and retrieve images with the same object. SteerViT can adapt to a text prompt, enabling retrieval of images even with small objects of interest. **Right:** We compare SteerViT to prior work in terms of its ability to adapt to text (measured by text-guided image retrieval (refer Sec. 4.1)) and the quality of the visual representation (measured by the accuracy of linear probing for the CLS feature and semantic segmentation for patch features). While models typically trade off steerability for representation quality, SteerViT preserves both. By modulating a gating factor (Eq. (2)), SteerViT achieves a new Pareto frontier.

1 Introduction

Pretrained Vision Transformers (ViTs) such as DINOv2 [28], MAE [10], and SigLIP [39] provide a generic representation of an image that can be applied to a variety of downstream tasks such as retrieval, classification, and segmentation. However, such representations tend to focus on the most prominent object in the image, likely due to the well-known photographer bias and object-centric vision datasets [35]. Consider the indoor scene from Figure 1: DINOv2 encodes the image focusing on the dominant salient object ("cat"), neglecting smaller or less prominent objects like the "remote control" or "bookshelf".

Given the lack of other input, focusing on the dominant object is reasonable. However, tasks such as fine-grained localization may require greater consideration of less prominent visual concepts. We argue that generic visual representations that can be *steered* using task-specific priors have significant utility.

In this work, we seek to steer pretrained vision transformers with natural language. We posit three desiderata that steerable representations should satisfy:

– **Steerability:** The representation should adapt to the input text. In particular, it should be steerable in *what* it encodes (so that it captures objects

Table 1: Only SteerViT satisfies all three desiderata. ✓ = satisfied, ✗ = not, ◐ = partial. *Trainable MM Params* reflects multimodal training; unimodal ViTs train solely on images (0). *V-L Fusion* indicates where modalities interact: outside (*late*) or within (*early*) visual encoder layers

Method Family	Text Steerable	Feature Quality	V-L Fusion	Trainable MM Params
Unimodal ViT (DINOv2)	✗	✓	✗	0
Cross-modal (CLIP)	✗	✓	Late	∼200M
OV Localize (SAM3)	✓	✗	Late	∼200M–1B
MLLM (Qwen3-VL)	◐	◐	Late, in LLM	≥1B
SteerViT (Ours)	✓	✓	Early, in ViT	**21M**

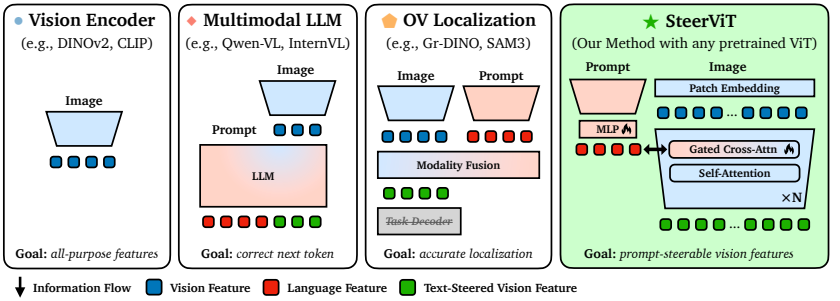


Fig. 3: Taxonomy of visual encoding. Standard vision encoders produce query-agnostic visual features. MLLMs and OV Localization models *late fuse* text after the visual encoder, modeling vision-language interactions inside the LLM or task-aligned encoder. SteerViT, instead, directly steers the internal features of a frozen ViT using text prompts (*early fusion*) via lightweight cross-attention layers.

irrespective of saliency; Sec. 4.1) and *how* it organizes the embedding space (so that clusters can be defined by specific attributes such as supercategories or object parts; Sec. 4.5).

– **Representation quality:** The steered representation should support diverse visual tasks, such as retrieval, classification, and segmentation (Sec. 4.3).
– **Early vision-language fusion:** Language should influence visual processing even at early layers of feature extraction.

While our first two desiderata are capabilities that are straightforward to measure (Fig. 2), the last is an architectural constraint that we hypothesize is crucial. When considering existing multimodal vision-language architectures (see Fig. 3), most encode visual features independently and employ *late fusion* with language. One reason is quite practical; existing language models are trained at massive scales on unimodal (text) data, which is far more plentiful than high-quality paired (text *and* image) data. It is not clear how to early-fuse text into the visual encoding process, and it is far easier to late-fuse them (for example, as in Multimodal LLMs (MLLMs) (e.g., [1, 3, 44])).

However, this implies that in existing models, *text does not influence the visual encoding process at inference time and only interacts with its outputs.* In human vision, studies indicate that people parse images differently depending on prior priming using a text prompt, often manifested as top-down task-guided attention [5] (example in Sec. A).

When considering prior art, we find that no existing approach simultaneously satisfies all three desiderata (see Tab. 1). MLLMs come closest, as they fuse visual and language representations in early layers of the language model. However, this paradigm typically yields language-dominant multimodal representations with diminished visual fidelity and limited controllability, as illustrated in Fig. 2.

Instead, with Steerable Visual Representations (SteerViT), we *invert* this paradigm: we condition a visual encoder on language input, producing a *vision-centric multimodal representation*. Specifically, we interleave lightweight trainable cross-attention layers [1] within frozen ViT blocks that attend to text prompts. This allows the visual encoding process to be heavily influenced or "steered" by text. We adopt referential image segmentation as the training objective to encourage vision-language alignment.

Our method achieves a Pareto improvement over prior approaches (Fig. 2), producing visual features that are steerable with text while preserving their underlying representation quality. This is accomplished by freezing both the visual and text encoders and adding only 21M trainable parameters via cross-attention, two orders of magnitude fewer than MLLMs (see Tab. 1).

Analogous to how prompting adapts (M)LLMs to novel tasks without retraining, language can adapt our *steerable visual representations* to novel domains without fine-tuning. Across diverse tasks spanning text-guided image retrieval (Sec. 4.1), personalized object discrimination (Sec. 4.4), and industrial anomaly detection (Sec. 4.6), SteerViT, without task-specific training, matches or outperforms strong baselines, including DINOv2, SAM3, billion-scale MLLMs and even dedicated methods. These results suggest that conditioning vision on language – rather than language on vision – could be a new paradigm for efficient multimodal vision understanding.

In summary, we make the following contributions:

1. We introduce Steerable Visual Representations (SteerViT), a framework that equips *any* pretrained visual encoder with text-steerable representations via a simple grounding pretext task, adding only 21M parameters.
2. We show that SteerViT achieves a Pareto improvement over prior approaches, steering visual features with text while preserving feature quality.
3. We demonstrate that text prompts enable zero-shot generalization, allowing SteerViT to transfer to new domains (e.g., personalized object discrimination or industrial anomaly detection) without task-specific training.

2 Related Work

Visual representation families. We summarize popular approaches against our desiderata in Tab. 1 and compare their architectures in Fig. 3. Unimodal self-supervised encoders (DINOv2 [28], MAE [10]) learn rich visual features but are inherently query-agnostic. Cross-modal encoders (CLIP [30], SigLIP [39], CoCoOp [43]) use text to provide training supervision; the visual encoder still cannot be steered with text. MLLMs come closest, offering moderate steerability and visual quality, but their representations reside in language space and require

billions of parameters. SteerViT inverts this paradigm: we condition vision on language, add only ∼21M trainable parameters to a frozen ViT, while yielding stronger visual features (Fig. 2).

Text-conditioned visual features. To our knowledge, *no prior work* steers a visual encoder effectively with text while preserving its representation quality. The closest attempt, FLAIR [37], applies text-conditioned attention pooling over a frozen SigLIP encoder (late fusion), resulting in suboptimal steerability and underperforming unimodal encoders on standard vision benchmarks (Fig. 2). Concurrent works condition visual features on text but target narrow pipelines. TIE [34] injects query tokens into the image encoder to reduce visual tokens in MLLMs, optimizing for document understanding. ELIP [40] prepends text in the ViT to improve text-to-image retrieval re-ranking. In contrast, SteerViT is a general framework for producing steerable visual representations that transfer across a wide variety of tasks.

3 SteerViT: Steering Vision Transformers with Text

Next, we describe the architectural modifications and post-training to obtain steerable representations from a pretrained ViT.

3.1 Architecture

As illustrated in Fig. 4, our SteerViT consists of four components:

A. Visual encoder. For an image $X_v \in \mathbb{R}^{H\times W\times 3}$, a ViT produces a sequence of N patch tokens $Z_v \in \mathbb{R}^{N\times d_v}$ and optionally a `[CLS]` token (d_v is the embedding dimension). All original ViT parameters remain frozen throughout training and new capacity results exclusively through the interleaved cross-attention layers described below. While most experiments adopt DINOv2 ViT-B/14 [28] as our backbone, we show that our approach also improves steerability of SigLIP and MAE.

B. Text encoder. We adopt a frozen, pretrained text encoder (RoBERTa-Large [24]) to produce token-level embeddings $Z_t \in \mathbb{R}^{L\times d_t}$ for a given input conditioning prompt X_t, where L is a variable number of text tokens and d_t is the text embedding dimension.

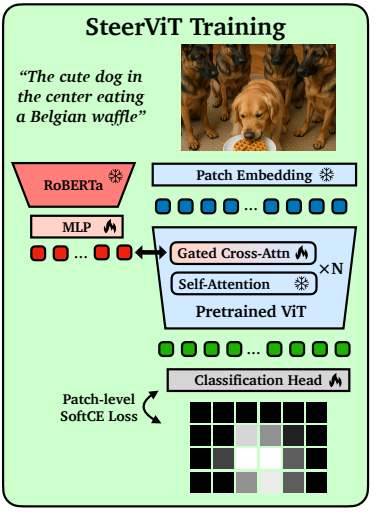


Fig. 4: Steering any ViT using text conditioning. Our method adds lightweight **vision**-to-**language** cross-attention layers within pretrained ViT blocks and applies a patch-level segmentation proxy objective to fuse prompt cues into patch tokens.

C. Multimodal adapter. Each text embedding Z_t^i is ℓ_2-normalized and fed through a trainable two-layer MLP that projects the sequence into a visual-aligned embedding space $H_t \in \mathbb{R}^{L\times d_v}$.

D. Gated cross-attention layers. To fuse textual conditioning into the ViT's residual stream, we invert the gated cross-attention (CA) formulation from

Flamingo [1] by allowing hidden vision states to attend to language tokens (CA is language→vision in [1]). We insert CA layers into every other Transformer encoder block (e.g., 6 CA layers for 12 ViT-B blocks). The visual patch tokens $Z_v^{(\ell)}$ at layer ℓ are queries and the adapted text tokens H_t are keys and values:

$$\hat{Z}_v^{(\ell)} = \mathrm{CA}(Z_v^{(\ell)}, H_t) = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V, \quad Q = Z_v^{(\ell)}W_Q,\ K = H_tW_K,\ V = H_tW_V. \tag{1}$$

The output is integrated into the residual stream through a tanh gate with a layer-specific learnable scalar α_ℓ, initialized to zero:

$$Z_v^{(\ell+1)} = Z_v^{(\ell)} + \tanh(\alpha_\ell) \cdot \hat{Z}_v^{(\ell)}. \tag{2}$$

Since $\tanh(0) = 0$, the model is identical to the frozen ViT at initialization. Despite the zero-initialization, the gate still receives a learning signal since $\frac{\partial Z_v^{(\ell+1)}}{\partial \alpha_\ell} = \mathrm{sech}^2(\alpha_\ell) \cdot \hat{Z}_v^{(\ell)}$, and $\mathrm{sech}^2(0) = 1$. Thus, α_ℓ can move away from zero during optimization, gradually activating the conditioning pathway and allowing the model to incorporate language-based clues.

3.2 Training Objective

In order to encourage the vision encoder to leverage and incorporate language clues, we design a pretext task requiring consideration of the prompt to be solved. We select referential segmentation for this purpose, where, given an image X_v and a prompt X_t referring to a target object or entity, the model predicts which patches correspond to the referred region.

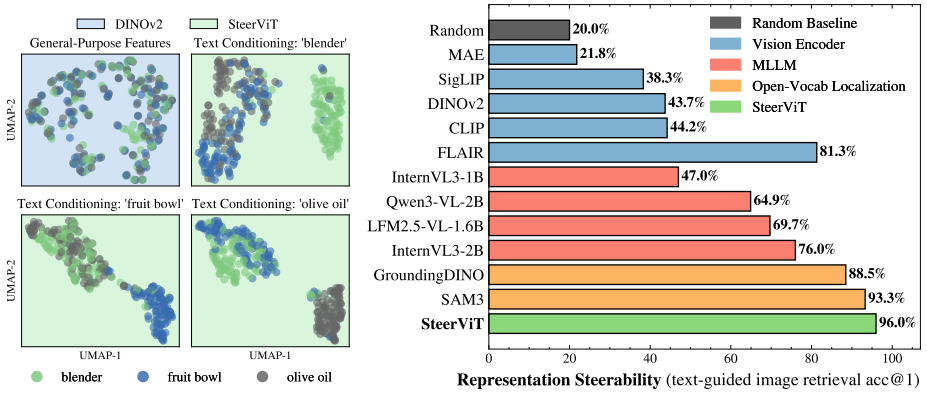
The ground-truth y_i is the fraction of foreground pixels of a pixel-wise binary segmentation mask that is patchified to match the ViT's $n \times n$ grid. A linear classifier head maps each patch representation $Z_v^i \in \mathbb{R}^d$ to a mask probability p_i through softmax. We adopt the soft cross-entropy loss to train our model:

$$\mathcal{L} = -\sum_{i=1}^{n \times n} y_i \log p_i \,, \tag{3}$$

as it encourages the CA layers to route textual information into the corresponding visual patch tokens, producing steered representations. Performing segmentation on a patch- rather than pixel-level reduces the training complexity and forgoes the need for pixel-level decoders.

3.3 Training Data

We train on a mixture of referential segmentation and grounding datasets spanning diverse visual domains and textual expression styles, comprising 162k unique images and 2.28M image-text pairs. Specifically, we use RefCOCO/+/g [17,38], Visual Genome [19], LVIS [9], and Mapillary Vistas [27]. Details are in supplement D.1.



(a) Feature distribution across three object types in the "kitchen" scene.

(b) Non-salient object retrieval performance on CORE across model families.

Fig. 5: COnditional REtrieval (CORE) benchmark. Left: While DINOv2 features form scene-level clusters, appropriate prompting of SteerViT yields object-specific clusters. **Right:** Substantial differences in steerability between model families exist, with OV localization methods and SteerViT offering the greatest adaptability.

4 Experiments

In this section, we empirically validate properties of our learned steerable representations and show applications to diverse downstream tasks.

Baselines. We compare SteerViT against baselines spanning multiple model families that differ in how, and whether, they incorporate text into visual features (see Sec. D.2 for feature-extraction details): (1) **Unimodal vision encoders**: DINOv2 [28] and MAE [10] produce query-agnostic features with no text conditioning. (2) **Cross-modal encoders**: CLIP [30] and SigLIP [39]; we fuse visual and text embeddings via post-hoc element-wise addition (late fusion). (3) **MLLMs**: InternVL3 [44], Qwen3-VL [3] and LFM-2.5-VL [2]; we extract prompt-specified vision features by following the last-token summary pooling of E5-V [15]. (4) **Open-vocabulary (OV) localization**: SAM3 [6] and GroundingDINO [23]; we use and evaluate the intermediate multimodal state.

Where supported, all models process images at 336×336 resolution; for models with fixed input dimensions, we first resize to this resolution.

4.1 COnditional REtrieval: Steering Global Semantics with Text

We propose *CORE (COnditional REtrieval)*, a text-conditioned image retrieval benchmark to measure how well a model steers its global features with text.

We select 100 images each for three indoor and three outdoor scenes from the SUN397 dataset [11] and inpaint five objects contextually fitting each scene into each image using the FLUX.2 image editing model [20] (e.g., a fruit bowl in a kitchen; a backpack in the park; see B.1.1 for details). We frame the problem as one-vs-all retrieval: given a query image containing an inpainted object Ω in

scene S (e.g., a fruit bowl in a kitchen), the goal is to retrieve other images of S that also contain Ω. This quantifies how well a model can steer its global features away from shared scene-level similarities (e.g., all images depict a kitchen) toward a specified non-salient object (e.g., the fruit bowl). Both query and gallery images are encoded while conditioned on the same brief description of Ω. We measure the top-1 retrieval accuracy over the remaining 495 samples of S with non-identical underlying images (pre-editing). Details about the experimental setup and results on a per-scene basis are reported in B.1.1.

Query-agnostic encoders collapse to salient concepts. Query-agnostic encoders fail at conditional retrieval because their features collapse to the dominant scene concept (Figure 5). MAE barely exceeds random chance (20%) and DINOv2, despite its object-centric representations, achieves only 44% acc@1. Cross-modal visual encoders (CLIP, SigLIP) also perform poorly despite operating in a shared vision-language space. In contrast, SteerViT achieves 96% retrieval accuracy, confirming that text conditioning shifts the global representation from the scene level ("kitchen") to the queried concept ("fruit bowl").

Figure 5a illustrates this on "kitchen" scenes for three objects via UMAP [26]. DINOv2 embeddings (top left) show no object separability, whereas our text-conditioned embeddings form distinct clusters for different text prompts. This confirms that SteerViT can reorganize its embedding space around the queried concept.

Late fusion does not enable steerability. Although CLIP and SigLIP operate in a shared vision-language space, their visual features are extracted independently of the query. Post-hoc element-wise addition of text yields a negligible 0.02% boost over their vision-only representations, confirming that late fusion cannot steer frozen visual features. In contrast, by conditioning intermediate representations on text (i.e., early fusion), SteerViT improves retrieval accuracy from 43.7% (vanilla DINOv2) to 96.0%. While FLAIR's trained attention pooling offers greater steerability (81.3%) than typical cross-modal encoders, it still falls short of SteerViT by 14.7 points.

MLLMs and OV models are steerable, but inefficient or specialized. MLLMs can moderately steer their visual features but at substantial computational cost. SteerViT outperforms both InternVL3-1B and InternVL3-2B by 49 and 20 percentage points, respectively, while only adding 21M parameters via cross-attention blocks compared to billion-parameter-scale LLMs (Tab. 1). Open-vocabulary localization models (GroundingDINO, SAM3) are highly steerable, with SAM3 nearly matching SteerViT's retrieval accuracy. However, their intermediate representations are optimized for localization and lack the generality needed for downstream transfer, as discussed in Sec. 4.3.

Conditioning on a random class breaks retrieval. To verify that steerability is genuinely text-driven rather than an artifact of training, we condition each model on a random (incorrect) object class (Tab. 7). Doing this has no effect on CLIP and SigLIP performance, confirming their features are unconditionally visual and not steerable. However, performance degrades drastically for FLAIR and SteerViT (-29.4 and -47.7 percentage points), indicating strong

text-dependence and corroborating that steerability is primarily prompt-driven. OV models see similar large drops, consistent with their deep text integration. MLLMs show only mild sensitivity, with InternVL3-1B declining by −7.6%.

Steerability transfers to real-world conditional retrieval. To assess whether the behavior observed on CORE transfers beyond our controlled inpainting setup, we additionally evaluate on the *Focus Object* split of GeneCIS [36], a zero-shot benchmark for conditional image retrieval in real images. Unlike CORE, which explicitly controls object presence and scene context, GeneCIS requires identifying the image that matches the reference scene while also containing the queried object. The retrieval gallery contains distractors that either share the scene but omit the object or contain the object in a different scene. Despite this more challenging setting, SteerViT transfers well in zero-shot evaluation, reaching 25.4% R@1, compared to 9.6% for DINOv2 and 18.7% for the benchmark's specialized baseline (cf. Tab. 2). These results complement CORE and show that language-based representational steering remains effective beyond a controlled synthetic benchmark.

Table 2: GeneCIS benchmark. SteerViT outperforms DINOv2 and specialized methods in challenging real-world conditional similarity benchmark.

Method	Focus Object ↑		
	R@1	R@2	R@3
DINOv2	9.6	19.5	28.4
SteerViT	**25.4**	**39.1**	**49.9**
Specialized	18.7	30.3	37.4

In summary, SteerViT and OV localization models can reliably be steered through text whereas standard ViTs collapse to salient concepts, with post-hoc late fusion providing no benefits. Although MLLMs offer moderate steerability, they bear significant computational cost and diminished visual feature quality.

4.2 MOSAIC Localization: Text enables Targeted Attention

We examine how SteerViT routes and aggregates global representations via attention to query-relevant tokens. For this, we construct a benchmark by stitching together four images from PASCAL-VOC [7] into a single 2×2 mosaic, resulting in a total of 363 composite images with reduced saliency of each primary subject. The `[CLS]`-to-patch attention scores in the final self-attention block highlight how global attention is redirected to regions specified by text prompt.

Qualitative analysis. As revealed by Fig. 6, DINOv2's attention map focuses primarily on the prominent "pony" and "airplane" entities, confirming the saliency bias hypothesized in Sec. 1. In contrast, SteerViT conditioned on "person" routes attention to the barely visible person in the top-left. When prompted with "potted plant", it distributes attention among the class instances in the bottom-right image. Additional qualitative examples are in Figure 14.

Quantitative evaluation. We quantify this effect by measuring the area under the precision-recall-curve (PR-AUC) of the attention heatmaps and the ground-truth segmentation masks for each object type appearing in the mosaic instance. DINOv2 cannot be actively steered and primarily focuses on prominent

Fig. 6: Text enables targeted attention. Attention maps on a four-image mosaic demonstrate that text conditioning with SteerViT redirects self-attention to the queried concept whereas DINOv2 attends to the most prominent objects. Note that the `[CLS]`-token of SteerViT was not directly optimized for targeted attention and remains frozen in its original state.

objects, resulting in a low PR-AUC of 14.3%. SteerViT can be steered with text to focus on objects of interest, achieving a substantially higher score of 50.2%.

Finding 1. SteerViT steers its visual features with text towards queried concepts, whereas standard encoders collapse to prominent visual cues.

4.3 Preserving Visual Representation Quality while Steering

While previous sections revealed steerability for SteerViT and OV localization, this is only useful if it preserves the downstream transfer, an important property of good visual representations. A naïve solution that overwrites visual features with text yields perfect steerability but destroys the underlying representation.

To assess this trade-off, we contrast CORE performance (cf. Sec. 4.1) with vision-centric downstream tasks intended to measure representational quality. In addition to training linear probes on global features across three fine-grained classification datasets (ImageWoof [13], Waterbirds [32], StanfordCars [18]), binary object-of-interest segmentation on ADE20k [41] gauges the dense encoding performance. Here, promptable models are conditioned on the superclass (e.g., "dog" for ImageWoof) and the ADE20k class name, respectively. For Fig. 2 and 7, both scores are averaged. Additional details are in supplement D.3.

Steerability ↔ Representation quality trade-off. Figure 2 (right) maps models along these two axes, revealing three regimes: (1) **Open-vocabulary localization methods** (SAM3 and GroundingDINO) achieve high steerability but produce localization-specific features that score poorly on generic vision

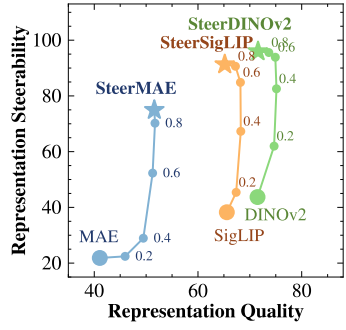


Fig. 7: Post-hoc steerability-visual quality modulation. Scaling CA gates α_ℓ at inference allows continuous interpolation between vanilla ViT and SteerViT characteristics. A factor of 0.6 provides the optimal steerability-quality trade-off for SigLIP- and DINOv2-based models.

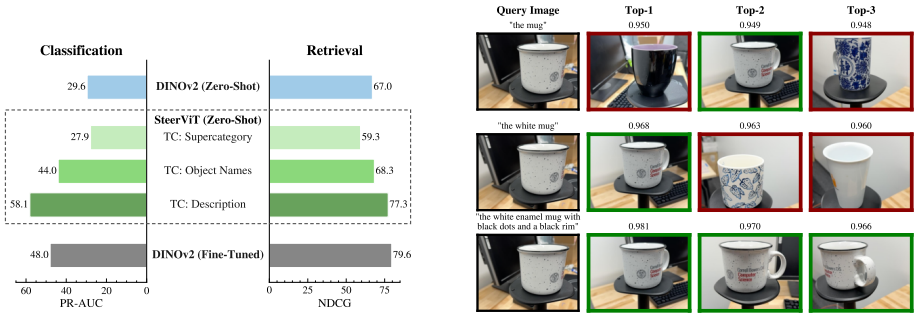


Fig. 8: Text controls feature granularity. Left: The quality of personalized representations produced by SteerViT improves drastically with more detailed prompts for text conditioning (TC), even surpassing a supervised fine-tuned DINOv2. **Right:** Retrieved images are better when providing SteerViT with more detailed descriptions.

tasks. (2) **MLLMs** perform reasonably on classification but falter on dense prediction tasks and require billions of parameters for moderate steerability. (3) **Query-agnostic encoders** (DINOv2, SigLIP) produce rich transferable features but cannot be steered. SteerViT bridges this gap, achieving high steerability while fully preserving the representation quality of the underlying ViT.

Cross-attention gate as a continuous control knob between ViT and SteerViT. The tanh-gated cross-attention mechanism (Eq. (2)) provides an implicit control knob for text conditioning strength. At inference, we can scale the learned gating parameters α_ℓ by a factor $\omega \in [0, 1]$ to smoothly interpolate between the unaltered ViT subspace and a fully text-conditioned state. Plotting this trajectory (Fig. 2, Fig. 7) reveals a clear Pareto frontier with an optimal operating point at a scaling factor of ω=0.6 for DINOv2 and SigLIP, where both slightly exceed the original ViT's representation quality while unlocking high steerability. Most strikingly, for MAE, representation quality monotonically improves as ω increases, rising from 40 at ω=0.0 to 50 points at ω=0.6. Text conditioning enriches MAE's features with semantic structure, making them more transferable.

> ***Finding 2.*** SteerViT produces the first family of visual representations that can be steered with text without sacrificing the representation quality of the underlying vision encoder.

4.4 Text Specificity Guides Semantic Granularity

Next, we explore the role of text in forming high-fidelity visual representations. For this, we choose Personal Object Discrimination Suite (PODS) [33], a benchmark evaluating the formation of instance-aware feature spaces via the ability of models to recognize particular objects (e.g., *your* mug vs. all other mugs). Given a small set of reference images, PODS computes cosine similarities on frozen global features and performs: (i) one-vs-all instance classification as well

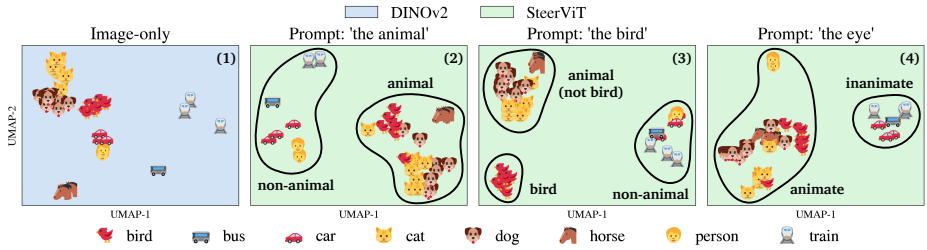


Fig. 9: SteerViT steers embedding topology via text. We show sub-sampled UMAP projections of 500 images across eight PASCAL-VOC classes. **(1)** DINOv2 clusters by object class. **(2)** Conditioning SteerViT on "animal" merges animal classes into one cluster while separating non-animals. **(3)** Conditioning on "bird" increases bird vs. non-bird separability while preserving the animal/non-animal macro-structure. **(4)** Conditioning on "eye" groups all animate classes (including "person", clustered with inanimate classes previously) together, demonstrating compositional attribute steering.

as (ii) similarity-based retrieval over a set of test images. Following [33], we report PR-AUC for classification and NDCG for retrieval.

From coarse to fine-grained steering. We vary the amount of detail provided to SteerViT via the text prompt from coarse supercategories (e.g., "mug", "shoe"), object names (e.g., "white ECCV mug"), to comprehensive MLLM-generated descriptions of each instance's visual appearance. As shown in Fig. 8, the semantics of the steered representations are sensitive to the prompt specificity. When conditioning is too coarse, the model overlooks fine-grained cues necessary for discriminating instances within an object category, yielding slightly worse performance than vanilla DINOv2 (27.9% vs. 29.6% PR-AUC). Enriching the prompt with instance-level descriptions substantially boosts performance to 58.1% PR-AUC, surpassing custom DINOv2 variants fine-tuned on synthetic task-specific data (48.0% PR-AUC) and nearly closing the gap between zero-shot and fine-tuned DINOv2 on retrieval (77.3% vs. 79.6% NDCG). This result is particularly noteworthy given that DINOv2 fine-tuning is object-specific, requiring a separate model per object class: 100 models in this setting compared to a single SteerViT model. These results demonstrate that SteerViT does not simply *add* information to the visual encoder but precisely controls the granularity of its visual features through the level of detail in the text prompt.

> ***Finding 3.*** The level of detail in text conditioning directly controls the granularity of the *steerable visual representations*.

4.5 Visualizing and Analyzing the Steered Embedding Space

The results on PODS (Sec. 4.4) indicate that query specificity dictates the level of semantic abstraction of *steerable visual representations*. Next, we explore how text conditioning shapes the embedding space topology of SteerViT using two complementary modes: steering along the *semantic hierarchy* and steering by *compositional attributes*.

For this, we select 500 images containing exactly one out of eight curated classes from PASCAL-VOC [7]. The encoded global image features are then reduced to 2D using UMAP [26].

Steering Along the Semantic Hierarchy. As shown in Figure 9 (1), DINOv2 embeddings cluster rigidly by object class. Conditioned on "animal" (2), two macro-clusters emerge for SteerViT, containing all animal and non-animal classes, respectively. Crucially, fine-grained structure is preserved (e.g., dogs and cats remain closer than birds), confirming that text controls clustering granularity without destroying object-level semantics. When conditioned on "bird" (3) separability between bird and non-bird images increases while preserving the higher-level animal versus non-animal macro-separation. This confirms that text can steer clustering at multiple levels of the semantic hierarchy, an emergent behavior not actively encouraged during training.

Steering by Compositional Attributes. Beyond semantic categories, text conditioning enables arbitrary grouping criteria, e.g., by shared object parts, defining a grouping principle orthogonal to semantic categories. Conditioning on "eye" (Figure 9, (4)) produces two macro clusters: animate classes that possess eyes versus inanimate objects that do not. Notably, *person* images, which were farther from animals under previous conditioning, now cluster together with them as the shared attribute "has eyes" overrides the semantic category boundary. This demonstrates that we can steer representations by compositional properties, not just taxonomy.

Finding 4. SteerViT can reorganize its embedding space using text, controlling both the level of semantic abstraction and clustering criteria.

4.6 Text Facilitates Zero-Shot Domain Transfer

The previous sections showed that SteerViT uses text to steer global features and route attention to queried objects, while preserving the representation quality of the underlying ViT. We now ask: *Does this language-conditioned flexibility translate into generalization to novel downstream tasks and unseen domains?*

In addition to generalizing to novel tasks like PODS (shown in Sec. 4.4), we showcase SteerViT's adaptability to extreme out-of-distribution settings by performing anomaly segmentation (AS) on the industrial MVTec AD [4] dataset. Here, models are conditioned on prompts in the style of "the anomaly in the <object>" and anomaly maps are derived by upsampling the heatmaps produced by the learned linear segmentation head. We compare SteerViT to other segmentation models and dedicated AS methods using *Per-Region-Overlap* (PRO). More details and extended results are provided in Sec. B.3. As shown in Table 3, SteerViT again matches dedicated zero-shot methods despite the extreme OOD setting. These results reinforce the pattern of text-driven steering unlocking capabilities already present in the frozen ViT and transferring them to domains beyond the training distribution.

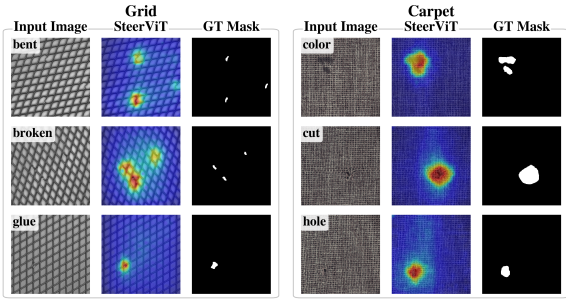


Fig. 10: Anomaly segmentation heatmap produced by SteerViT. Language-conditioning enables robust zero-shot generalization to this OOD task.

Method	PRO
MaskCLIP [42]	40.5
CLIPseg [25]	34.6
SAM3 [6]	54.5
WinCLIP [14]	64.6
DIVAD [12]	73.3
FADE [21]	84.5
SteerViT	82.1

Table 3: ZS anomaly segmentation (MVTec). Dedicated methods in gray.

Finding 5. SteerViT can use natural language to transfer rich unimodal vision encoders to OOD domains without task-specific training.

4.7 Analysis of SteerViT

We ablate the key architectural and training decisions in SteerViT to isolate the contribution of each component. Unless stated otherwise, all experiments use the DINOv2 ViT-B/14 backbone and the RoBERTa-Large as the text encoder and are trained at 336×336 resolution with a batch size 12 for 500k iterations (∼84 H100 GPU-hours) on the full data mixture (Sec. 3.3), using AdamW with a cosine schedule that warms up to 3×10^{-4} over 5k steps, decays to 3×10^{-5} by 40k steps, and remains constant thereafter. We evaluate in three domains: fine-grained classification probe accuracy (FG-CLS; Sec. 4.3), text steerability via CORE retrieval (Sec. 4.1), and personalized object discrimination PR-AUC on PODS with descriptive prompts (Sec. 4.4).

Table 4: Architecture ablations on DINOv2 ViT-B/14. SteerViT combines three principal design choices; each ablation row toggles exactly one (marked ✗).

Early Fusion	Tanh Gate	MLP Proj.	FG-CLS ↑	CORE ↑	PODS ↑
–	–	–	89.0	43.7	29.6
✓	✓	✓	87.7	**96.0**	**58.1**
✗	✓	✓	**91.8**	93.3	36.6
✓	✗	✓	83.5	94.6	47.1
✓	✓	✗	86.7	95.2	56.4

✗ in Early Fusion means late fusion. ✗ in Tanh Gate means ungated cross-attention. ✗ in MLP Proj. means single linear layer. Vanilla DINOv2 baseline (top row, gray) does not consider text.

Architecture Choices. SteerViT has three principal design choices: early fusion of text within the ViT, tanh-gated CA layers, and MLP text projector. Tab. 4 ablates each component.

Early vs. late fusion. We interleave gated cross-attention within ViT layers (early fusion). Late fusion (Tab. 4, row 3) adds text only after the final ViT layer. While late fusion (row 3) also achieves high steerability (93.3) and higher FG-CLS (91.8 vs. 87.7), it reduces PODS performance dramatically (36.6 vs. 58.1).

This illustrates the importance of early fusion for fine-grained vision–language modeling, with the gap vanishing (26.5 vs. 27.9) when conditioning on coarse supercategory prompts.

Role of gating. Removing the zero-initialized tanh gate (row 4) reduces FG-CLS, CORE, and PODS by 4.2, 1.4, and 11.0 points below SteerViT (row 2), showing that ungated cross-attention disrupts frozen features.

Language projector. A two-layer MLP projects RoBERTa features to the ViT space (Sec. 3.1). Replacing it with a linear projector (row 5) lowers FG-CLS by 1.0 and PODS by 1.7 points, indicating that the MLP better aligns modalities.

Generalization Across Pretrained Backbones. To validate that our approach generalizes beyond DINOv2, we apply SteerViT to two additional ViTs: SigLIP [39] and MAE [10]. Tab. 5 compares vanilla, late fusion, and early fusion (SteerViT) variants.

Table 5: Steering different ViT pretraining approaches. SteerViT consistently outperforms late fusion, with the largest gains on weaker backbones.

Backbone	CORE ↑		
	Base	Late	SteerViT
DINOv2	43.7	93.3	**96.0**
SigLIP	38.3	75.4	**91.3**
MAE	21.8	41.0	**74.9**

Overall, DINOv2 produces the strongest steerable representations, consistent with its richer self-supervised features. SigLIP and MAE similarly benefit from text conditioning but start from weaker baselines. Early fusion consistently outperforms late fusion across all three ViT families. The advantage is more pronounced for MAE and SigLIP, where early fusion boosts steerability by 33.9 and 15.9 points, respectively. These larger early-fusion gains compared to DINOv2 (2.7) align with our later layer-wise analysis (cf. Fig. 13), where SteerViT representations diverge earlier from their base ViTs for MAE and SigLIP backbones, suggesting that language injection is especially beneficial when the underlying visual features are less semantically mature.

Additional ablations for backbones, model scaling, proxy-task are in Sec. C.

5 Conclusion

We introduce Steerable Visual Representations (SteerViT), a new class of visual representations that equip *any* pretrained visual encoder with the ability to steer its features with natural language. SteerViT inverts the MLLM paradigm by conditioning the visual encoder on language. By interleaving lightweight cross-attention layers into frozen ViT blocks, our method steers both global and local features with text while preserving the base ViT's representation quality, achieving a Pareto improvement over prior approaches with only ∼21M trainable parameters. Across personalized object discrimination and industrial anomaly segmentation, our method matches or surpasses dedicated methods without task-specific training, generalizing across multiple ViT backbones. These results suggest that text can serve as a lightweight, post-hoc steering mechanism that extends the capabilities of rich vision encoders to new domains without fine-tuning.

References

1. Alayrac, J.B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., et al.: Flamingo: a Visual Language Model for Few-Shot Learning. In: Advances in Neural Information Processing Systems (NeurIPS) (2022)
2. Amini, A., Banaszak, A., Benoit, H., Böök, A., et al.: Lfm2 technical report. arXiv preprint arXiv:2511.23404 (2025)
3. Bai, S., Cai, Y., Chen, R., Chen, K., et al.: Qwen3-vl technical report. arXiv preprint arXiv:2511.21631 (2025)
4. Bergmann, P., Fauser, M., Sattlegger, D., Steger, C.: MVTec AD — A Comprehensive Real-World Dataset for Unsupervised Anomaly Detection. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
5. Buswell, G.T.: How people look at pictures: a study of the psychology and perception in art. (1935)
6. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., et al.: SAM 3: Segment Anything with Concepts. In: International Conference on Learning Representations (ICLR) (2026)
7. Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: "The Pascal Visual Object Classes (VOC) Challenge". International Journal of Computer Vision **88**(2), 303–338 (Jun 2010)
8. Ghiasi, A., Kazemi, H., Borgnia, E., Reich, S., Shu, M., Goldblum, M., Wilson, A.G., Goldstein, T.: What do vision transformers learn? a visual exploration. arXiv preprint arXiv:2212.06727 (2022)
9. Gupta, A., Dollar, P., Girshick, R.: LVIS: A dataset for large vocabulary instance segmentation. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
10. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked Autoencoders Are Scalable Vision Learners. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
11. Herranz, L., Jiang, S., Li, X.: Scene Recognition with CNNs: Objects, Scales and Dataset Bias. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
12. Hicsonmez, S., Shabayek, A.E.R., Aouada, D.: Training Free Zero-Shot Visual Anomaly Localization via Diffusion Inversion. arXiv preprint arXiv:2601.08022 (2026)
13. Howard, J.: Imagewoof: a subset of 10 classes from Imagenet that aren't so easy to classify (2019), `https://github.com/fastai/imagenette#imagewoof`
14. Jeong, J., Zou, Y., Kim, T., Zhang, D., Ravichandran, A., Dabeer, O.: WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
15. Jiang, T., Song, M., Zhang, Z., Huang, H., Deng, W., Sun, F., et al.: E5-V: Universal Embeddings with Multimodal Large Language Models. arXiv preprint arXiv:2407.12580 (2024)
16. Kamath, A., Singh, M., LeCun, Y., Synnaeve, G., Misra, I., Carion, N.: MDETR: Modulated detection for end-to-end multi-modal understanding. In: International Conference on Computer Vision (ICCV) (2021)
17. Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.: ReferItGame: Referring to objects in photographs of natural scenes. In: Empirical Methods in Natural Language Processing (EMNLP). pp. 787–798 (2014)

18. Krause, J., Stark, M., Deng, J., Fei-Fei, L.: 3D Object Representations for Fine-Grained Categorization. In: International Conference on Computer Vision Workshops (ICCVW) (2013)
19. Krishna, R., Zhu, Y., Groth, O., Johnson, J., Hata, K., Kravitz, J., et al.: Visual Genome: Connecting Language and Vision Using Crowdsourced Dense Image Annotations. International Journal of Computer Vision (IJCV) **123**(1), 32–73 (2017)
20. Labs, B.F.: FLUX.2: Frontier Visual Intelligence. `https://bfl.ai/blog/flux-2` (2025)
21. Li, Y., Ivanova, E., Bruveris, M.: FADE: Few-shot/zero-shot Anomaly Detection Engine using Large Vision-Language Model. In: British Machine Vision Conference (BMVC) (2024)
22. Lian, L., Ding, Y., Ge, Y., Liu, S., Mao, H., Li, B., et al.: Describe Anything: Detailed Localized Image and Video Captioning. In: International Conference on Computer Vision (ICCV) (2025)
23. Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J., Zhang, L.: Grounding DINO: Marrying DINO with Grounded Pre-Training for Open-Set Object Detection. In: European Conference on Computer Vision (ECCV) (2024)
24. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., et al.: RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692 (2019)
25. Lüddecke, T., Ecker, A.: Image Segmentation Using Text and Image Prompts. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
26. McInnes, L., Healy, J., Saul, N., Großberger, L.: UMAP: Uniform Manifold Approximation and Projection. Journal of Open Source Software **3**(29), 861 (2018)
27. Neuhold, G., Ollmann, T., Bulò, S.R., Kontschieder, P.: The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes. In: International Conference on Computer Vision (ICCV) (2017)
28. Oquab, M., Darcet, T., Moutakanni, T., Vo, H.V., Szafraniec, M., Khalidov, V., Fernandez, P., et al.: DINOv2: Learning Robust Visual Features without Supervision (2024)
29. Pariza, V., Salehi, M., Asano, Y.: Hummingbird Evaluation for Vision Encoders (4 2024), `https://github.com/vpariza/open-hummingbird-eval`
30. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., et al.: Learning Transferable Visual Models From Natural Language Supervision. In: International Conference on Machine Learning (ICML) (2021)
31. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., et al.: SAM 2: Segment anything in images and videos. In: International Conference on Learning Representations (ICLR) (2025)
32. Sagawa, S., Koh, P.W., Hashimoto, T.B., Liang, P.: Distributionally Robust Neural Networks. In: International Conference on Learning Representations (ICLR) (2020)
33. Sundaram, S., Chae, J., Tian, Y., Beery, S., Isola, P.: Personalized Representation from Personalized Generation. In: International Conference on Learning Representations (ICLR) (2025)
34. Thirukovalluru, R., Han, X., Dhingra, B., Dinan, E., Elbayad, M.: Text-Guided Semantic Image Encoder. arXiv preprint arXiv:2511.20770 (2025)
35. Torralba, A., Efros, A.A.: Unbiased look at dataset bias. In: Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1521–1528 (2011)
36. Vaze, S., Carion, N., Misra, I.: Genecis: A benchmark for general conditional image similarity. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2023)

37. Xiao, R., Kim, S., Georgescu, M.I., Akata, Z., Alaniz, S.: FLAIR: VLM with Fine-grained Language-informed Image Representations. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
38. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling Context in Referring Expressions. In: European Conference on Computer Vision (ECCV) (2016)
39. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid Loss for Language Image Pre-Training. In: International Conference on Computer Vision (ICCV) (2023)
40. Zhan, G., Liu, Y., Han, K., Xie, W., Zisserman, A.: ELIP: Enhanced Visual-Language Foundation Models for Image Retrieval. In: IEEE International Conference on Content-Based Multimedia Indexing (2025)
41. Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., Torralba, A.: Scene Parsing through ADE20K Dataset. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2017)
42. Zhou, C., Loy, C.C., Dai, B.: Extract Free Dense Labels from CLIP. In: European Conference on Computer Vision (ECCV) (2022)
43. Zhou, K., Yang, J., Loy, C.C., Liu, Z.: Conditional Prompt Learning for Vision-Language Models. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
44. Zhu, J., Wang, W., Chen, Z., Liu, Z., Ye, S., Gu, L., Tian, H., Duan, Y., Su, W., Shao, J., et al.: Internvl3: Exploring advanced training and test-time recipes for open-source multimodal models. arXiv preprint arXiv:2504.10479 (2025)
45. Zou, Y., Jeong, J., Pemula, L., Zhang, D., Dabeer, O.: SPot-the-Difference Self-Supervised Pre-training for Anomaly Detection and Segmentation. In: European Conference on Computer Vision (ECCV) (2022)

Steerable Visual Representations

Supplementary Material

In the following, we discuss the inspiration of our work from human image perception (Sec. A), additional qualitative and quantitative results on CORE, MOSAIC, and anomaly segmentation (Sec. B), additional ablations (Sec. C), and details of experimental setup such as training dataset and feature extraction process (Sec. D).

A Motivation: Parallels to Human Image Perception

Fig. 11 illustrates how human gaze patterns over the same image shift depending on a task-relevant textual prompt [5], providing inspiration from neuroscience for SteerViT's early fusion of language into the visual encoding process.

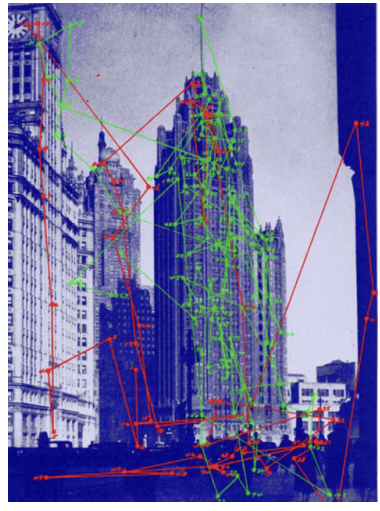


Fig. 11: Human gaze patterns. Eye movement patterns differ when observers are instructed to locate a person looking out of a window in the tower (green) versus when no contextual instruction is provided (red). Adapted from [5].

B Additional Qualitative and Quantitative Results

This section provides further details about the employed evaluations, provides qualitative examples and reports additional results.

Table 6: Overview of scene and object types used in CORE benchmark. Each of the three indoor and outdoor environments is paired with five corresponding objects, that are inpainted into images of that scene using an image editing model. The prompts provide contextual guidance, indicating the object the model should focus on.

Indoor		
Kitchen	**Living Room**	**Bathroom**
"the black blender with a transparent glass cup."	"the small black TV remote."	"the small yellow rubber duck."
"the fruit bowl with fresh red apples."	"the small bouquet of red roses in a white and gray vase."	"the wet mop with a red handle."
"the green glass bottle of olive oil."	"the stack of multiple books."	"the stack of white toilet paper rolls on the floor."
"the dark green kitchen towel."	"the black and white soccer ball."	"the small pink carpet on the floor."
"the gray plate with a slice of pepperoni pizza."	"the bottle of red wine."	"the hanging winter coat."
Outdoor		
Street	**Suburb**	**Park**
"the apple tree with ripe apples."	"the coiled green garden hose."	"the baby stroller."
"the small metal trash can."	"the blue curbside mailbox."	"the orange traffic cone."
"the American flag."	"the bicycle leaning against something."	"the small birdhouse on a post."
"the zebra walking on the street."	"the orange basketball."	"the dark blue hiking backpack."
"the puddle with water on the street."	"the red fire hydrant."	"the colorful kite lying on the grass."

B.1 Assessing Representational Steerability

B.1.1 Controlled Prompt-Conditional Retrieval on CORE

For our *COnditional REtrieval* (CORE) benchmark introduced in Sec. 4.1, we select 100 base images for each of six scenes (three indoor and three outdoor environments) from SUN397 [11]. For each image, we create five separate instances by inpainting a scene-relevant background object (e.g., an "olive oil bottle" in "kitchen") using FLUX.2 [dev] [20], resulting in a total of 500 images per scene. The selected scenes, the corresponding five objects added to each scene and the text-conditioning prompts are reported in Table 6.

To test the ability of visual encoders to steer their global representations in order to retrieve images relevant to a specified prompt, we frame the problem as one-vs-all retrieval. For each image X_v^{Ω} containing object Ω, the objective is to retrieve other instances including Ω. Here, all images (query and gallery instances) are encoded with the same object-specific prompt X_t^{Ω} (cf. Tab. 6). For a given query image, the gallery is comprised of 495 variations with non-identical base images of the same scene. The steerability score is calculated using top-1 retrieval accuracy and averaged across query images and object types.

We report detailed quantitative results on a per-scene basis in Table 7. SteerViT (green) consistently outperforms query-agnostic ViTs (blue), MLLMs (red), and OV localization models (yellow) across all indoor and outdoor scenes, surpassing the underlying DINOv2 by an average of 52.5 points. When conditioned on a randomly chosen incorrect object (✗), SteerViT's accuracy drops

Table 7: Per-scene performance on the CORE benchmark. SteerViT performs consistently best across all indoor and outdoor scenes when using the correct prompt (✓). When text-conditioning on an object other than the one to retrieve (✗; incorrect prompt), performance drops significantly.

Method	Bathroom		Kitchen		Liv. Room		Park		Suburb		Street	
	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗	✓	✗
MAE	22.0	n/a	20.0	n/a	19.4	n/a	25.6	n/a	21.7	n/a	22.4	n/a
DINOv2	54.8	n/a	38.0	n/a	30.8	n/a	56.7	n/a	34.3	n/a	47.6	n/a
SigLIP	61.2	61.6	31.0	29.4	35.2	35.0	31.6	32.6	30.3	30.9	40.4	39.8
CLIP	64.2	64.2	37.6	37.6	47.6	47.6	44.6	44.2	26.3	26.3	44.8	44.8
FLAIR	95.6	69.0	79.6	42.2	82.0	48.0	74.4	55.8	84.6	54.3	71.4	42.2
InternVL3-1B	61.8	54.4	35.6	28.4	43.0	32.6	50.2	42.3	42.9	33.1	48.6	45.6
InternVL3-2B	89.4	70.2	70.2	42.2	60.6	36.4	75.8	57.7	81.1	50.3	78.8	57.2
Qwen3-VL-2B	81.6	57.6	48.8	30.8	63.8	33.6	66.5	46.1	64.0	34.3	64.8	45.4
LFM2.5-VL-1.6B	85.0	73.8	56.4	42.0	64.6	49.0	66.5	48.8	72.0	51.4	73.4	64.0
GroundingDINO	89.4	30.2	89.6	24.8	91.0	25.2	86.5	33.5	93.1	29.7	81.6	28.6
SAM3	97.8	35.6	95.4	27.0	91.4	26.6	94.0	33.0	95.4	21.7	85.6	26.0
SteerViT	99.2	44.4	98.6	43.6	96.6	49.4	93.0	58.1	97.7	56.0	90.8	34.4

drastically, whereas cross-modal encoders (CLIP and SigLIP) see negligible changes. This confirms that text actively shapes the visual representation in SteerViT, unlike cross-modal encoders where no cross-modal interaction influencing the nature of the vision features occurs.

Figure 12 illustrates qualitatively how retrievals differ for a prompt-agnostic generic vision model like DINOv2 (blue) and our prompt-aware SteerViT (green) for one indoor and outdoor scene each. Whereas DINOv2 mostly encodes scene-level appearance and retrieves visually similar images, SteerViT focuses on the objects specified in the prompt and retrieves images containing those. Despite many of the inpainted objects being very small and in the background, the global features incorporate them sufficiently to allow for highly accurate retrievals.

B.1.2 Layer-wise Effects of Text Conditioning

To further analyze how text conditioning reshapes the global image representation, we measure the divergence $(1 - \cos(Z_v^{(\ell)}, \tilde{Z}_v^{(\ell)}))$ between intermediate representations produced by SteerViT (Z_v) and its underlying vanilla ViT ($\tilde{Z}_v$) after each transformer block ℓ. Divergence is averaged across all COCO validation images conditioned on a randomly selected object present in the scene. As illustrated in Figure 13, divergence is already non-zero in the early-to-mid layers, indicating that the text signal is incorporated throughout the encoding process. It grows most strongly in later blocks, consistent with the view that ear-

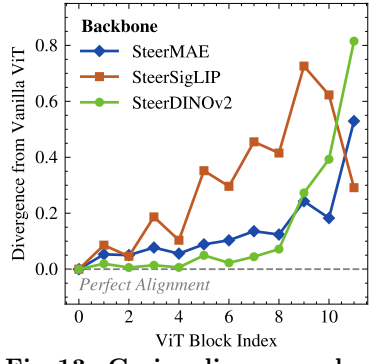


Fig. 13: Cosine divergence between SteerViT and base ViT intermediate features. Text-conditioning progressively steers internal features across layers.

Fig. 12: Qualitative image retrieval results on CORE benchmark. White circles highlight the inpainted object. Green / red border indicates whether image contains queried object of interest. DINOv2 retrieves images mostly based on global scene similarity with little regard for the object of interest. In contrast, SteerViT effectively considers the task prompt and ranks images with the correct object highest, despite their innocuous background placement and comparably small size.

lier ViT layers primarily encode lower-level visual structure while later layers capture higher-level semantics [8]. However, the divergence profile is backbone-dependent: SteerDINOv2 remains close to its base model until the final blocks and then departs sharply; SteerSigLIP diverges slowly and gradually throughout; while SteerMAE diverges strongly in intermediate layers but converges back toward the original MAE space in the final blocks, suggesting that text conditioning enriches MAE's features with semantic structure before returning them to the original representation space.

B.2 MOSAIC Benchmark

The MOSAIC benchmark introduced in Sec. 4.2 evaluates whether text conditioning can steer the self-attention of the `[CLS]` token toward patch tokens corresponding to a prompted object. We stitch four PASCAL-VOC [7] images (padded to 1:1 aspect ratio) into a single 2×2 mosaic to prevent a single dominant salient object. For a given text prompt specifying an object class, the ground-truth is a binary segmentation mask over the mosaic's patch grid, constructed from instances of that class across all four sub-images.

Figure 14 provides more qualitative examples of attention maps on such multi-image mosaic. Again, we see a strong propensity of DINOv2 to focus on the most dominant object(s) within the collage. SteerViT selectively attends to the objects or entities specified in the prompt (Fig. 14, top). Despite being trained with single-instance data (i.e., only one object corresponding to prompt), we find that our model can localize multiple instances of an object class across images in mosaic (Fig. 14, SteerViT attends to all instances of "chair" in middle row). We discuss this emergent multi-localization property of our method in Sec. C.1. SteerViT also considers described attributes and routes global attention accordingly (e.g., "black sheep" vs. "white sheep" in Fig. 14 (bottom)).

Fig. 14: Effective prompt-based attention steering. DINOv2 focuses on dominant image regions. **Top:** SteerViT can attend to specific objects-of-interest. **Middle:** Despite single-instance training, it can consider multiple occurrences of the specified concept. **Bottom:** specifying attributes (e.g., color) places focus on specific entities.

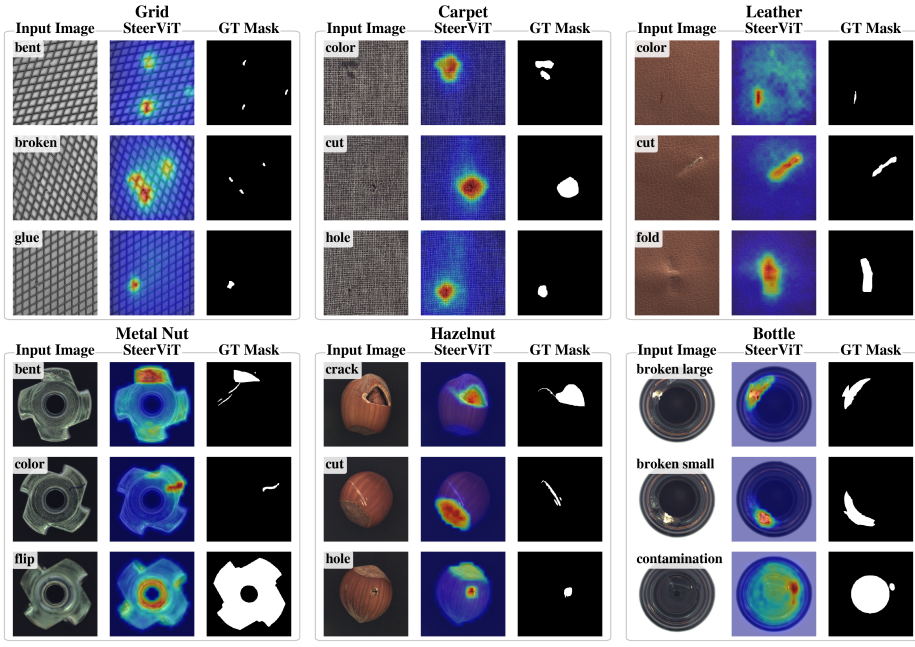


Fig. 15: Anomaly segmentation heatmaps predicted by SteerViT. We use the linear classification head optimized for patch-wise referential segmentation. Despite the stark task and domain gap, performance is especially good on texture-based classes (top) but also allows SteerViT to highlight small defects in objects (bottom).

B.3 Anomaly Segmentation

We perform zero-shot anomaly segmentation (AS) on the MVTec AD [4] and VisA [45] datasets. We derive the anomaly map by upsampling the continuous heatmap produced by the learned linear segmentation head. As for all text-steerable methods, we use an ensemble of ten anomaly prompts (e.g., "`the anomaly in the <object>.`") and average the resulting heatmaps across prompts. Following standard practice in the AD literature [14], we measure the pixel-wise *area under the ROC curve* (ROC_P), the *Per-Region-Overlap* (PRO), and threshold-optimal F_1 score ($F_1^{\max}$). Performance of existing training- and adaptation-free AS baselines is reported as stated in the original publication.

SteerViT again matches dedicated methods on this extreme OOD setting. As shown in Table 8, SteerViT reaches 82.1 PRO on MVTec AD, substantially outperforming off-the-shelf segmentation-based approaches (SAM3 at 54.5, CLIPseg at 34.6) and closing much of the gap to specialist methods such as FADE (84.5). On VisA, SteerViT surpasses FADE in ROC_P (92.1 vs. 91.5) and PRO (82.0 vs. 79.3), indicating robust transfer to a harder, more diverse inspection setting. Qualitative examples of SteerViT's predictions on a subset of object types and defect categories are provided in Fig. 15. While the predicted heatmaps are especially accurate for texture-based inputs, the zero-shot setting makes it impos-

Table 8: Zero-shot segmentation performance on industrial AD data.

Method			MVTec AD			VisA		
Name	Ref.	AS	ROC_P	PRO	F_1^{max}	ROC_P	PRO	F_1^{max}
MaskCLIP	[42]	✗	63.7	40.5	18.5	60.9	27.3	7.3
CLIPseg	[25]	✗	69.0	34.6	12.5	89.5	62.4	13.9
SAM3	[6]	✗	79.9	54.5	24.1	89.8	65.9	15.5
WinCLIP	[14]	✓	85.1	64.6	31.7	79.6	59.8	14.8
DIVAD	[12]	✓	88.0	73.3	35.5	**93.4**	78.2	**24.0**
FADE	[21]	✓	**89.6**	**84.5**	**39.8**	91.5	79.3	16.7
SteerViT	**Ours**	✗	87.8	82.1	35.6	92.1	**82.0**	18.3

AS indicates dedicated anomaly segmentation methods.

sible to accurately predict certain defects (e.g., flipped metal nut), as no training on non-anomalous instances occurs and no visible defects (e.g., scratches) are present.

C Additional Analysis on Training Design Choices

Below, we further analyze important design choices involved in the training of SteerViT. To gauge model performance, we report the preservation of representation quality, as measured by fine-grained linear classification probes (FG-CLS) and binary object of interest segmentation (ADE20k), using the previously introduced methodology. Models' steerability and textual understanding are assessed using the CORE and PODS benchmarks. We also evaluate referential grounding on RefCOCOg by computing IoU between the predicted patch-level segmentation mask and the ground-truth mask.

C.1 Training Objective: Pointing vs. Segmenting

Our goal is to instill language understanding into a frozen ViT via a simple, scalable proxy task. Referential localization serves this purpose: given a text prompt, the model must ground the described object. To avoid the complexity of a pixel-level decoder, we operate entirely in the ViT's $n \times n$ patch-token grid.

Pointing. The simplest formulation assigns a hard target (1 at the center patch, 0 elsewhere), but we find this unstable to optimize in practice for both classification and regression. Instead, we pass a Gaussian kernel (σ=1.1) at the bounding-box center and train with soft cross-entropy, which provides a smooth gradient towards the object center. However, this target is invariant to object shape and size, collapsing supervision to a single spatial location.

Segmenting. Consequently, we generate segmentation masks using SAM2 [31] conditioned on ground-truth bounding boxes and project them onto the patch grid. Unlike pointing, this target activates all tokens overlapping the object,

Table 9: Role of supervision signal. Segmentation is a superior training objective compared to pointing (Gaussian kernel at bounding box center) across all downstream evaluations, resulting in improved visual and multimodal understanding.

Train Objective	FG-CLS ↑	ADE20k ↑	CORE ↑	PODS ↑
Pointing	80.4	47.4	95.2	45.7
Segmentation	**87.7**	**55.4**	**96.0**	**58.1**

Table 10: Scaling visual and text encoders. Larger ViT backbones consistently improve performance across all tasks. Reducing RoBERTa-Large to Base modestly degrades visual quality (FG-CLS, ADE20k) but preserves multimodal understanding (CORE, PODS).

ViT	RoBERTa	FG-CLS ↑	ADE20k ↑	CORE ↑	PODS ↑
Small	Large	80.0	50.8	93.6	44.1
Base	Large	**87.7**	55.4	96.0	58.1
Large	Large	85.8	**55.5**	**96.8**	**62.8**
Base	Base	**87.9**	53.6	95.7	57.4
Base	Large	87.7	**55.4**	**96.0**	**58.1**

teaching content-matching: each patch must determine whether it depicts the described object, not merely how close it is to the center.

As shown in Tab. 9, segmentation consistently outperforms pointing across all metrics. The largest gains appear on feature quality preservation (FG-CLS: +7.3; ADE20k: +8.0) and PODS (+12.4). Pointing teaches the model *where* an object is, whereas segmentation also teaches *what* the object looks like, producing features that better encode object extent and appearance.

C.2 Training Duration

Steerability emerges rapidly: within 50k iterations of training, CORE accuracy reaches 95.3% (vs. 43.5 for frozen DINOv2), while FG-CLS remains nearly constant at 89.6 (vs. 89.2 for frozen DINOv2). However, tasks that require deeper language understanding continue improving with longer training: between 50k and 450k iterations, PODS improves from 49.9 to 58.1 and RefCOCOg from 63.4 to 70.6, suggesting that longer training instills richer multimodal representations rather than merely teaching the model to route attention. We adopt 500k iterations for all final models.

C.3 Scaling

As shown in Table 10, scaling the vision backbone from ViT-S through ViT-B to ViT-L leads to improvements in both representational quality as well as

textual understanding. Interestingly, scaling the text encoder from RoBERTa-Base (125M parameters) to RoBERTa-Large (355M) also improves performance on visual tasks. This suggests that rich embeddings of the larger text encoder add more semantic structure to the ViT's residual stream (row 4 vs row 5).

C.4 Role of FFN

Table 11: Role of the FFN. Removing the gated FFN from each cross-attention block yields comparable or better performance. The effect is prominent for MAE, where adding FFN degrades steerability and zero-shot transfer substantially.

Backbone	FFN	FG-CLS ↑	ADE20k ↑	CORE ↑	PODS ↑	MVTec ↑
DINOv2	✗	**87.7**	**55.4**	**96.0**	**58.1**	**82.1**
DINOv2	✓	86.0	54.3	95.0	55.2	79.3
SigLIP	✗	**82.6**	47.7	**91.3**	**27.4**	74.8
SigLIP	✓	80.5	**48.5**	89.1	26.9	**80.8**
MAE	✗	67.3	**35.9**	**74.9**	**23.8**	**77.3**
MAE	✓	**67.8**	35.5	67.7	21.8	73.9

SteerViT inverts the gated cross-attention formulation introduced by the Flamingo MLLM [1]. However, it forgoes secondary gated feed-forward networks (FFN) following the cross-attention operations in the original architecture. As shown in Tab. 11, including the FFN brings little to no benefit in representation quality, while consistently hurting steerability and out-of-distribution transfer. The effect is especially pronounced for MAE, where adding the FFN reduces CORE by 7.2 points and MVTec PRO by 3.4 points (row 5 vs. row 6). At the same time, the FFN substantially increases the parameter count of the adapter, growing the cross-attention module from 21.2M additional parameters without a FFN to 35.4M with FFN (+67%). Since the FFN is both expensive and empirically dispensable, we omit it in the final architecture.

D Further Details on Experimental Setup

D.1 Training Data

We train on a mixture of referential segmentation and grounding datasets to ensure diversity in both visual domains and textual expression styles (see Fig. 16): **RefCOCO, RefCOCO+, RefCOCOg** [17, 38] provide referring expressions grounded in COCO images. RefCOCO+ excludes spatial language (e.g., "left of"), encouraging the model to rely on appearance cues, while RefCOCOg contains longer, more descriptive expressions that exercise the model's capacity for detailed textual understanding.

LVIS [9] uses the same underlying COCO images but also considers fine-grained and long-tail object categories.

Visual Genome [19] (preprocessed following MDeTr [16]) contributes region descriptions paired with bounding boxes across densely annotated scenes. These descriptions span a broader vocabulary and more complex spatial relationships than the RefCOCO family, increasing the diversity of text conditioning signals. We use SAM2 [31] to convert bounding boxes to binary segmentation masks.

Mapillary Vistas [27] provides street-level imagery with fine-grained panoptic annotations. Including this dataset expands the visual domain beyond COCO, improving out-of-distribution generalization. We adopt Describe Anything's [22] synthetic referential expressions along with accompanying segmentation masks.

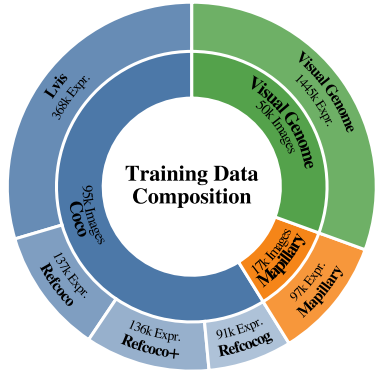


Fig. 16: Training data composition. The inner ring shows unique images per source while the outer ring shows associated referring expressions. The mixture is intended to combine fine-grained object supervision, diverse referring expressions, and broad visual domain coverage.

Overall, our combined dataset of 162k unique images and 2.28M image-text pairs exposes the model to varied scene complexities (from single objects to dense urban panoramas), expression lengths (from two-word labels to multi-sentence descriptions), and visual domains (indoor, outdoor, street-level), encouraging robust steered representations.

D.2 Baseline Feature Extraction

This section provides more details on how we extract text-conditioned (or text-fused) visual features from each baseline family evaluated in Sec. 4. For all baselines, global feature aggregation follows the convention of the base method (e.g., `[CLS]` token or mean pooling) unless stated otherwise.

Cross-Modal Encoders (CLIP [30], SigLIP [39]). Images and language inputs are embedded independently by their respective encoders. We combine the resulting visual and text feature vectors via element-wise addition (late fusion), the simplest form of post-hoc multimodal interaction.

MLLMs (InternVL3 [44], Qwen3-VL [3]). We feed the text prompt X_t and image X_v as a single multimodal input sequence and extract hidden states from the last LLM layer. For dense features, we take the hidden state at each image-token position. For global features, following E5-V [15], we append an instruction asking the model to summarize the image in one word in addition to the initial text prompt X_t and extract the last-token hidden state.

Open-Vocabulary Localization (SAM3 [6], GroundingDINO [23]). We ignore decoder outputs and extract intermediate features from the multimodal

encoder. For SAM3, we take the patch-level outputs of the *Fusion Encoder* and apply mean pooling to obtain global image features. For GroundingDINO, we use the outputs of the cross-modal *Feature Enhancer* layer. Because features are available at multiple spatial scales, we first interpolate them to a common intermediate resolution before averaging to obtain dense pseudo-patch-level features which are optionally mean-pooled to yield global image-level embeddings.

D.3 Assessing Visual Representation Quality

In addition to the steerability of vision representations, we also evaluate their quality for common computer vision downstream tasks.

Fine-grained image classification tests whether representations capture sufficient details to perform accurate categorization within specialized domains. Namely, we train a linear probe (300 epochs, LR: $1e^{-3}$, BS: 128) on frozen features extracted from the ImageWoof [13], Waterbirds [32], and StanfordCars [18] datasets. Here, prompt-aware models are conditioned on ‘‘`the dog`’’, ‘‘`the bird`’’, and ‘‘`the car`’’, respectively.

Binary object-of-interest segmentation tests the semantic quality of patch-level features using the ADE20k [41] dataset. We apply the training-free OpenHummingBird [29] evaluation methodology where patch-class pairs are precomputed for the training split and patch-wise segmentation is performed at inference via nearest-neighbor retrieval from the reference bank. Each class is evaluated separately (i.e., binary class/non-class segmentation objective) and features are conditioned on the object of interest's original class name (e.g., ‘‘`the chair`’’).