Heads That Write, Not Just Point: Image Retrieval Heads in Vision-Language Models

Junsung Park[1] **Sangwon Yu**[2] **Donghun Kang**[1] **Yeongtak Oh**[1] **Jaewon Jeong**[3]
Gyeongtae Yoo[1] **Han Cheol Moon**[4] **Jungbeom Lee**[5*] **Sungroh Yoon**[1,3,6*]

[1]Department of Electrical and Computer Engineering, Seoul National University [2]Upstage AI
[3]Interdisciplinary Program in AI, Seoul National University [4]Samsung Electronics
[5]Korea University [6]AIIS, ASRI, INMC, and ISRC, Seoul National University

Abstract

Large vision-language models (LVLMs) can process interleaved text and multiple images in a single context, but how they internally identify the image relevant to a question remains poorly understood. We study this grounding step as *in-context image retrieval* and introduce ICIR-MCQ, a controlled probe for head-level analysis of image retrieval. Using each head's attention mass over image-token spans, we first identify sparse *pointer heads* that reliably point to the target image. These heads transfer beyond the probe: a single pointer head is competitive with external vision-language retrievers on a naturalistic multi-image retrieval benchmark, showing that LVLMs contain a strong internal image retrieval signal. Attention alone, however, shows where a head reads from, not what its output passes on to later layers, so we introduce a complementary output-side analysis. For each head, we train a lightweight image retriever on its value-weighted output and use it to identify *writer heads*, whose outputs contain enough information to identify the target image. Surprisingly, the pointer and writer criteria select substantially different head sets, with only 35 to 47 heads overlapping among the top-100 heads under each criterion. Targeted set-partition ablations show that the heads most important for standard multi-image inference are concentrated in this pointer-writer intersection, while heads selected by only one criterion show less consistent separation from random-head controls. The most consistent causal effects arise from heads selected by both the attention-based pointer and output-side writer criteria. Code and data: `https://github.com/parkquasar/Image-Retrieval-Heads`.

1 Introduction

Modern large vision-language models (LVLMs) [3, 6, 21, 40] can process interleaved text and multiple images in a single context, enabling applications such as document understanding and multi-image question answering. In multi-image contexts, answering an image-grounded question requires more than visual recognition: the model must first identify which image the question refers to. We refer to this image-grounding component as *in-context image retrieval*. This retrieval step gates downstream computation, since grounding the question in the wrong image conditions the answer on the wrong visual evidence. Recent multi-image benchmarks show that model accuracy drops as the number of candidate images increases, exposing in-context image retrieval as a central behavioral bottleneck [39, 41]. However, they leave open how LVLMs internally perform this image selection, and which components in the model carry the image retrieval signal.

*Corresponding authors: Jungbeom Lee (`jbeomlee@korea.ac.kr`), Sungroh Yoon (`sryoon@snu.ac.kr`).

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Attention-pattern analysis has been a standard tool for identifying specialized heads in transformers [10, 27, 37]. Recent retrieval-head analyses and attention-based VLM head analyses likewise identify heads by where they attend in the input context [4, 8, 42]. This approach is intuitive and has been effective in practice: attention patterns expose which tokens or spans a head points to. However, pointing is only one side of the attention operation. Attention patterns do not by themselves determine what information the head writes into the residual stream. A head's output is the value-weighted vector produced by mixing value vectors according to its attention weights, and this output, rather than the attention pattern itself, is what downstream layers consume [20].

We hypothesize that this gap between pointing and writing is especially important for in-context image retrieval in LVLMs, where retrieval is cross-modal: attention points to visual-token spans, but the value-weighted output must carry target-image information in a hidden representation used by later language-generation layers. This creates a mismatch between pointing and writing. A head may point cleanly to the target image without writing target-image information in a form recoverable from its output. Conversely, output-side recoverability and attention-mass strength need not induce the same ranking. Attention-only head discovery may fail to distinguish heads that merely point to the target image from heads that also write target-image information into the residual stream.

We test this hypothesis with a two-sided analysis of head-level retrieval in LVLMs. On the input side, we identify *pointer heads* using attention mass over image-token spans (§4). On the output side, we identify *writer heads* by training a lightweight head-specific image retriever on each head's value-weighted output, asking whether the target image can be identified from the information carried by that output (§5). We then use targeted ablations to test which head population the model's answer-generation pathway depends on. Identifying image retrieval heads in LVLMs therefore requires combining where a head points with what it writes.

To carry out this analysis, we introduce ICIR-MCQ (Figure 1; §3.2), a controlled probe that gives each sample a clear target image while keeping the task solvable by standard LVLMs. ICIR-MCQ is designed for controlled head-level discovery rather than as a standalone downstream benchmark; we evaluate transfer and causal relevance on Visual Haystacks [41], a naturalistic multi-image retrieval benchmark. Across four LVLMs spanning two families and scales, only 35–47 of the top-100 heads under the pointer and writer criteria overlap. Targeted set-partition ablations further show that standard multi-image inference depends primarily on this pointer-writer intersection, while heads selected by only one criterion show less consistent separation from random-head controls (§6).

Our contributions are as follows.

1. **A controlled probe for head-level image retrieval.** We introduce ICIR-MCQ, a controlled probe with clear target-image labels for head-level retrieval analysis.
2. **Pointer heads as strong implicit image retrievers.** We identify sparse *pointer heads* that reproduce at identical (layer, head) indices within shared LLM backbones. A single pointer head is competitive with external vision-language retrievers like CLIP and SigLIP.
3. **Writer heads via a complementary output-side criterion.** We introduce an output-side head discovery method that trains a lightweight image retriever on each head's output, identifying *writer heads* whose outputs retain recoverable target-image information. This retriever-based ranking differs substantially from the attention-based pointer ranking.
4. **Causal validation: retrieval heads are pointers that write.** Set-partition ablations show that standard multi-image inference is most sensitive to heads selected by both criteria.

2 Related Work

Specialized attention heads and head-level interventions in transformers. Prior work has identified specialized transformer heads for syntax [10, 37], induction [27], and long-context factual retrieval [42]. Many of these analyses identify heads primarily from attention patterns over textual contexts. However, attention weights alone do not determine the output of an attention layer [20]. Our output-side analysis is aligned with broader mechanistic work that studies activations, circuits, and interventions beyond raw attention patterns [5, 11, 14, 15, 22, 35]. We extend this perspective to cross-modal image retrieval in LVLMs, where information about the target image must be written to the residual stream rather than directly emitted as an output token.

Mechanistic interpretability of vision-language models. Prior work studies how VLMs integrate visual and textual information across tokens, layers, and attention patterns [13, 17, 26, 29, 32, 44]. Head-level VLM studies have identified visual, OCR, and broad functional heads in single-image, rendered-text, or multimodal VQA settings [4, 8, 9, 16]. These works either focus on single-image or rendered-text settings, or assign broad functional labels rather than isolating the image-selection mechanism in natural multi-image contexts. We study natural multi-image retrieval, and our two-sided analysis exposes a discrepancy that attention-based identification alone cannot capture: heads selected by the pointer criterion are not necessarily selected by the writer criterion.

Multi-image vision-language models and benchmarks. Recent LVLMs support interleaved text and multi-image inputs [3, 6, 21, 40], enabling evaluation over many images in context. Multi-image benchmarks such as Visual Haystacks and MM-NIAH show that performance degrades as the number of images grows [33, 34, 39, 41], suggesting that target-image grounding is a central bottleneck. However, they evaluate output behavior rather than the internal retrieval mechanism. Our work addresses this gap by providing a head-level mechanistic account of in-context image retrieval, using a controlled probe (§3.2) for discovery and Visual Haystacks for transfer evaluation (§4.3).

3 Setup

3.1 Preliminaries

In-context image retrieval. Consider an LVLM processing an interleaved image-text context

$$C = (\tau_0, I_1, \tau_1, I_2, \ldots, I_N, \tau_N, u),$$

where $I_1, \ldots, I_N$ are candidate images, $\tau_0, \ldots, \tau_N$ are intervening text passages, and u denotes task-specific textual instructions, including the question and answer options when applicable. Answering an image-grounded question in this setting requires grounding the question in the target image before performing the downstream visual computation. We refer to this image-grounding component, which selects the relevant image within the context, as *in-context image retrieval*. This retrieval step gates multi-image inference: grounding the query in the wrong image conditions the answer on the wrong visual evidence. This paper studies the head-level mechanism underlying this retrieval step.

Attention notation. After tokenization, the interleaved context C becomes a sequence of length T containing both visual and text tokens. Each image I_i occupies a contiguous visual-token span $[s_i, e_i]$, while different image spans are separated by text-token spans induced by $\tau_0, \ldots, \tau_N$. For an LVLM with L transformer blocks and H attention heads per block, we denote the post-softmax attention matrix of head (l, h) by $A^{(l,h)} \in \mathbb{R}^{T \times T}$. For a token position p, the attention mass assigned by head (l, h) to image i is

$$m_i^{(l,h)}(p) \;=\; \sum_{j=s_i}^{e_i} A_{p,j}^{(l,h)}. \tag{1}$$

Both the input-side and output-side criteria define a per-head, per-sample score vector $s^{(l,h)} \in \mathbb{R}^N$ over candidate images, and differ in how they compute it (§4.1 and §5.1). The score is computed at a task-specified query position p, which we instantiate for ICIR-MCQ in §3.2. For a sample with target image $i^\star$, we define the head-level retrieval correctness as

$$r^{(l,h)} \;=\; \mathbb{1}\left[\operatorname*{argmax}_i \, s_i^{(l,h)} = i^\star\right]. \tag{2}$$

We report head-level R@1 by averaging this indicator over samples.

Models. We study four LVLMs spanning two model families and two parameter scales: InternVL3.5-{4B, 8B} [40] and Qwen3-VL-{4B, 8B} [6]. The two model families differ in vision encoder and training data, but the 4B variants share an LLM backbone, as do the 8B variants. This design lets us examine head-level stability across LLM backbones and, within shared backbones, head-index reproducibility. We choose these families because they are recent open-weight LVLMs that support interleaved multi-image inputs and allow head-level attention extraction and intervention. Exact architectural dimensions and implementation details are provided in Appendix A.

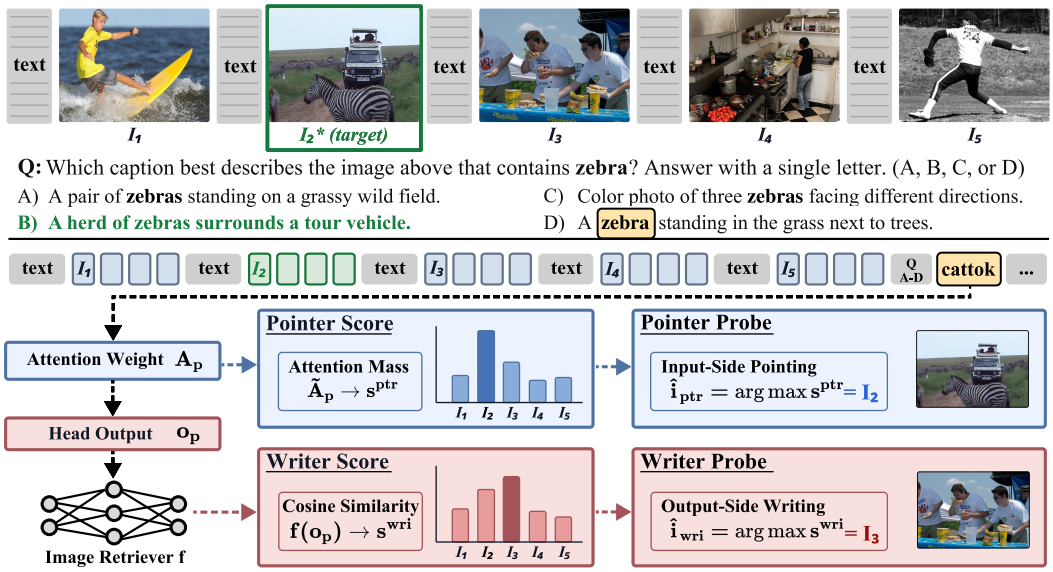


Figure 1: An ICIR-MCQ sample. The model selects the caption matching the image containing the queried category (*zebra*). At the *cattok*, the *pointer* criterion ranks images by projection-based attention mass (Eq. 3), while the *writer* criterion ranks images using a retriever trained on the head output. The two criteria can disagree in the same head.

3.2 ICIR-MCQ: A Controlled Probe for Head-Level Image Retrieval

We design ICIR-MCQ as a controlled probe for analyzing image retrieval at the attention head level. Each sample consists of an interleaved image-text context with N candidate images, a textual query specifying an object category, and four candidate captions. The model is asked to select the caption that best describes the image containing the queried category (Figure 1, top). The target image $i^\star$ is the unique image in the context annotated with the queried category; in Figure 1, this is the second image, which contains a *zebra* and matches the caption (B).

ICIR-MCQ is constructed to satisfy three properties. First, the input is an interleaved image-text context, matching the general form of multi-image inputs to LVLMs. Second, the intervening text passages do not reveal the target image or the correct caption. Third, the candidate images are natural images, so head-level findings are not tied to a synthetic visual construction. To satisfy these requirements, we construct ICIR-MCQ from MS-COCO [24], inheriting the interleaved structure from MM-NIAH's image-retrieval task [39] and substituting natural images for the original images.

More precisely, we keep the inherited text passages and reformulate the task as multiple-choice caption matching. Because the original MM-NIAH passages are kept fixed while the images and captions are newly substituted from COCO, the passages are independent of the target image and answer options by construction. We use COCO category annotations to control the per-image attribution target: the queried category is annotated in the target image but not in any distractor image, making the target image unambiguous given the category cue. Distractor captions are sampled from other COCO images containing the same queried category, so the model cannot solve the task from the category cue alone; it must ground the query in the correct image.

For head-level analysis, we score each head at the last token of the final mention of the queried category, which we call the *cattok*. This token is a query position at which the category cue has just appeared in text and all candidate image spans are in the causal past. In Figure 1, the cattok is the last token of *zebra* in option (D). Scoring at the cattok asks which image each head ranks highest when conditioned on the queried category.

The target image position and correct answer letter are balanced uniformly across options; further sampling and filtering details are in Appendix B. We compute head-level retrieval R@1 under each criterion's score, defined in §4.1 and §5.1. ICIR-MCQ serves as our discovery probe and supplies the data for training and evaluating the head-specific image retrievers (§5.1).

Table 1: Top-5 pointer heads by head-level attention-mass R@1 on ICIR-MCQ. The strongest heads cluster in the same mid-block layer band, and within each LLM-backbone group (8B above; 4B below), top heads reproduce at identical (layer, head) indices. Cells show `LxxHyy` (R@1).

Model	Top-1	Top-2	Top-3	Top-4	Top-5	Median
InternVL3.5-8B	`L21H10` (.890)	`L21H11` (.850)	`L21H08` (.842)	`L20H17` (.825)	`L21H09` (.823)	.206
Qwen3-VL-8B	`L21H10` (.922)	`L21H08` (.883)	`L21H11` (.878)	`L23H10` (.873)	`L21H09` (.869)	.198
InternVL3.5-4B	`L21H10` (.866)	`L21H11` (.847)	`L23H10` (.835)	`L21H08` (.826)	`L21H09` (.824)	.217
Qwen3-VL-4B	`L21H10` (.905)	`L23H10` (.878)	`L21H11` (.870)	`L21H08` (.859)	`L21H09` (.856)	.206

4 Input-Side Retrieval: Pointer Heads

4.1 Pointer Criterion: Retrieval by Per-Image Attention Mass

A natural input-side criterion for image retrieval is whether a head attends to the target image. We instantiate this as the *pointer criterion*. For head (l, h) and the cattok position p, the score assigned to image i is the attention mass over its visual-token span:

$$s_i^{(l,h),\mathrm{ptr}} = \sum_{j=s_i}^{e_i} \tilde{A}_{p,j}^{(l,h)}, \qquad \tilde{A}_{p,j}^{(l,h)} = \mathrm{softmax}\left(\frac{\mathbf{q}_p^{(l,h)} \cdot \mathbf{k}_j^{(l,h)\top}}{\sqrt{d_h}}\right)_j . \tag{3}$$

Here $\mathbf{q}_p^{(l,h)}$ and $\mathbf{k}_j^{(l,h)}$ are the direct outputs of the query and key projection layers and the softmax runs over all prefill positions j; $\tilde{A}$ is a proxy for the forward-pass attention A. Plugging $s^{(l,h),\mathrm{ptr}}$ into Eq. 2 yields the head's R@1 under the pointer criterion. We call high-R@1 heads *pointer heads*: they rank the target image first by concentrating attention mass on its visual-token span. The criterion follows the attention-pattern convention used to identify retrieval heads in language-only models [42], adapted here from token-level lookup to span-level image retrieval.

4.2 Pointer Heads Are Sparse and Localized

Pointer ability is sharply non-uniform: the best pointer head reaches R@1 between .866 and .922 across the four LVLMs (Table 1; chance R@1 is .183), while the median head remains near chance. High-R@1 heads concentrate in a narrow mid-block layer band, with the same band visible on every LVLM; the per-model layer×head R@1 heatmap is reported in Appendix C (Figure 4). We denote by $\mathcal{H}_K^{\mathrm{ptr}}$ the top-K heads under this pointer ranking. Despite differing in vision encoder and training data, LVLMs sharing an LLM backbone show sharp reproducibility at specific (layer, head) indices. The top-3 pointer heads coincide as a set within each backbone group, and 86 of the top-100 are shared between the 8B models, 80 between the 4B models. Per-head pointer R@1 grids and within-backbone consistency statistics are reported in Appendix C.

4.3 Pointer Heads Transfer Beyond the Controlled Probe

We next ask whether pointer heads discovered on ICIR-MCQ transfer to a naturalistic multi-image benchmark. We use Visual Haystacks [41]: each example contains multiple images and a Yes/No question with an *anchor object* that identifies the target image and a *target object* whose presence in it is queried. For example, a question may ask, "In the image that contains a *sofa*, is there a *bottle*?", where *sofa* is the anchor object and *bottle* is the target object. For pointer scoring, we use the last token of the anchor-object phrase as the cattok, because this is where the model has enough information to determine which image should be retrieved. We evaluate at haystack size $N{=}10$, where N denotes the number of candidate images, using each head's per-image attention mass to rank candidate images and measuring whether the top-ranked image is the target image.

Transfer from ICIR-MCQ to a naturalistic benchmark. For each LVLM, we use its top pointer head identified on ICIR-MCQ. Per-image retrieval R@1 ranges from .787 to .836 at haystack size $N{=}10$, far above the chance level of .1 for selecting among ten images. At this haystack size the 4B variants match or slightly exceed their 8B counterparts, suggesting that strong pointer signals are not confined to the larger models.

Table 2: Image retrieval R@1 on Visual Haystacks (haystack size N=10, chance R@1 is $1/10$). Internal results use the single pointer head `L21H10` in each LVLM; external retrievers are evaluated on the same samples. **IV** = InternVL3.5, **QV** = Qwen3-VL, and **QV-Emb** = Qwen3-VL-Embedding-8B.

Method	Single Pointer Head				Retriever Model		
	IV-8B	QV-8B	IV-4B	QV-4B	CLIP	SigLIP	QV-Emb
R@1	.787	.832	.794	.836	.573	.649	.751

A single pointer head is competitive with external retrievers. A single attention head from an LVLM not trained with an explicit retrieval objective achieves higher per-image retrieval R@1 (Table 2) than standard external vision-language retrievers—CLIP [31], SigLIP [43], and Qwen3-VL-Embedding-8B [23]. This comparison should not be read as a controlled training-objective comparison: the internal pointer head and the external retrievers operate on different representations and conditioning signals. Rather, it shows that the LVLM contains a highly effective context-conditioned image-selection signal in a single head. This advantage is consistent with the internal head's access to richer signal: external retrievers compare global image embeddings with context-free text embeddings, whereas the pointer head attends over patch-level keys in the LVLM's own representational space using a query conditioned on the context through the cattok. Retriever-side work continues to improve both components [19, 30]. Additional retriever details are provided in Appendix A.

5 Output-Side Retrieval: Writer Heads

The pointer criterion asks where a head points, whereas downstream computation uses what it writes to the residual stream. At the cattok position p, the head output is

$$\mathbf{o}_p^{(l,h)} \;=\; \sum_{j=1}^{T} A_{p,j}^{(l,h)}\, \mathbf{v}_j^{(l,h)} \;\in\; \mathbb{R}^{d_h}, \qquad \mathbf{v}_j^{(l,h)} \;=\; W_v^{(l,h)} \mathbf{x}_j, \tag{4}$$

where $\mathbf{x}_j$ is the input to the value projection at position j and $W_v^{(l,h)}$ is the head's value projection. A head's value projection determines which information from the attended tokens enters the output: even when attention concentrates on the target image, there is no guarantee that the head output contains information sufficient to identify that image. We therefore introduce a complementary output-side criterion that asks whether the target image can be identified from the head output itself. This lets us compare attention-based pointing with output-side writing at the same head granularity.

5.1 Writer Criterion: Retrieval from Head Outputs

We introduce a retriever-based output-side head discovery method. For each head (l, h), we train a lightweight head-specific *image retriever* on the head output $\mathbf{o}_p^{(l,h)}$. Following the standard dual-encoder retrieval formulation [18, 31], the retriever maps the head output and each candidate image embedding into a shared space and scores candidates by cosine similarity,

$$s_i^{(l,h),\mathrm{wri}} \;=\; \tilde{f}_{\theta_{l,h}}(\mathbf{o}_p^{(l,h)}) \cdot \tilde{g}_{\phi_{l,h}}(\mathbf{e}_i), \tag{5}$$

where $\mathbf{e}_i$ is obtained by mean-pooling candidate image i's projected visual tokens in the LVLM's language-model embedding space, so the writer probe uses the LVLM's own image representation rather than an external encoder. We train $f_{\theta_{l,h}}$ and $g_{\phi_{l,h}}$ jointly with InfoNCE [28] for each head.

We split ICIR-MCQ into disjoint train/evaluation subsets at the sample level. For writer-head discovery, we evaluate Eq. 5 over a held-out image pool of 1,200 candidates per LVLM and compute R@1 by whether the target image ranks first. We call this metric *writer R@1*, and call high-R@1 heads under this criterion *writer heads*. Operationally, a writer head is one whose output lets the trained retriever identify the target image, in the spirit of probing intermediate representations [7]. Whether these outputs are used by the model's answer-generation pathway is tested separately in §6. Architecture details and hyperparameters are provided in Appendix D.

Table 3: Top-5 writer heads by writer R@1 on the ICIR-MCQ held-out image pool. Within each LLM-backbone group (8B above; 4B below), top heads largely overlap. Writer R@1 is evaluated on a held-out pool of 1,200 candidate images, whereas pointer R@1 in Table 1 is evaluated on the images in each context; the two R@1 are not directly comparable in absolute scale. Cells show LxxHyy (R@1).

Model	Top-1	Top-2	Top-3	Top-4	Top-5	Median
InternVL3.5-8B	L24H23 (.489)	L20H28 (.435)	L20H15 (.426)	L21H31 (.421)	L18H19 (.418)	.077
Qwen3-VL-8B	L24H23 (.595)	L21H17 (.470)	L26H26 (.460)	L21H31 (.459)	L20H28 (.458)	.095
InternVL3.5-4B	L21H31 (.472)	L20H28 (.439)	L24H23 (.433)	L20H15 (.428)	L21H17 (.428)	.080
Qwen3-VL-4B	L24H23 (.509)	L20H28 (.487)	L21H31 (.486)	L21H17 (.483)	L18H19 (.463)	.101

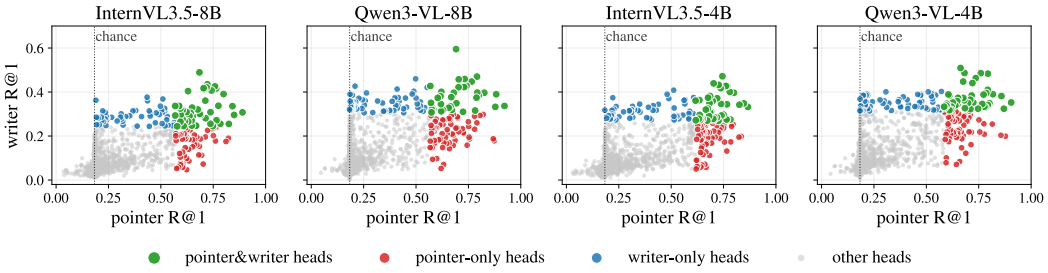


Figure 2: Head-level pointer R@1 versus writer R@1 on ICIR-MCQ. Each point is one attention head: colors denote *pointer&writer*, *pointer-only*, *writer-only*, and other heads, where membership is defined by the top-100 heads under each ranking. The vertical dotted line marks pointer-axis chance (R@1 = .183). The two criteria select substantially different head sets across all four LVLMs.

The writer criterion is parallel to the pointer criterion: both produce per-image scores and are evaluated by top-1 retrieval accuracy. However, pointer R@1 is computed over the images in each prompt, whereas writer R@1 is computed over a held-out image pool. We therefore use writer R@1 as an output-side ranking criterion, rather than comparing its absolute value directly against pointer R@1. We denote the top-K pointer and writer heads by $\mathcal{H}_K^{\mathrm{ptr}}$ and $\mathcal{H}_K^{\mathrm{wri}}$.

5.2 Writer Heads Encode Recoverable Target-Image Information

The trained retrievers give us a concrete way to test whether a head output carries recoverable target-image information. Table 3 shows that the strongest writer heads reach R@1 of .472–.595 on the held-out pool of 1,200 candidate images, compared with median values of .077–.101.

Writer ability is sparse and mid-block. Writer R@1 is sharply non-uniform across heads. The strongest writer heads concentrate in mid-block layers 18–27 (Appendix D). This suggests from the output side that image-retrieval-relevant signals are localized rather than uniformly distributed across the network. Per-head writer R@1 grids are provided in Appendix D.

Top writer heads reproduce within shared LLM backbones. As with pointer heads, writer heads reproduce within shared LLM backbones despite differences in vision encoder and training data. The 8B models share three of the top-5 writer heads and the 4B models share four, with three heads common to all four models (Table 3). Although writer heads occupy a similar mid-block region as pointer heads, their top-ranked identities differ, motivating a direct comparison between the two rankings.

5.3 Pointing Is Not Writing

Pointing and writing identify substantially different heads. Both rankings concentrate in the same mid-block region, yet at top-100 under each criterion, only 35–47 heads are selected by both.

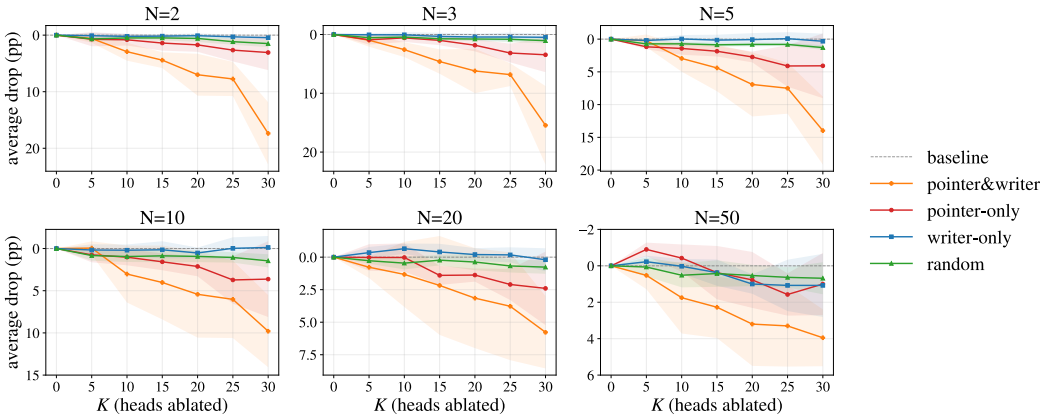


Figure 3: Set-partition ablation on Visual Haystacks. Each panel corresponds to a haystack size N, the number of candidate images. Curves show average accuracy drop relative to the unablated baseline as K heads are ablated. The pointer-writer overlap shows the most consistent effect. Shaded bands show ± 1 standard deviation across the four LVLMs.

The off-diagonal heads in Figure 2 make the dissociation concrete. Pointer-only heads rank in the pointer top-100 but outside the writer top-100. Conversely, writer-only heads in $\mathcal{H}^{\text{wri}}_{100} \setminus \mathcal{H}^{\text{ptr}}_{100}$ score highly under the writer criterion despite not being among the strongest pointer heads. Thus, pointing and writing provide complementary, non-interchangeable rankings.

This dissociation has direct methodological implications: attention-based identification alone cannot distinguish heads that merely point to the target image from heads that also write target-image information into the residual stream. The remaining question is which population—pointer-only, writer-only, or their intersection—the model's answer-generation pathway actually depends on. Section 6 addresses this with targeted set-partition ablations over these disjoint head sets.

6 Causal Validation: Retrieval Heads Are Pointers That Write

Sections 4 and 5 establish that pointing and writing identify substantially different heads, even though both criteria localize to the same mid-block region. The key question is which population the standard single-pass forward actually depends on. If the causally relevant retrieval heads are those that both point to the target image and write recoverable information into the residual stream, the dominant causal contribution should lie in the overlap between the two criteria. We test this hypothesis with targeted head ablations on Visual Haystacks across multiple haystack sizes, where each example has a ground-truth target image while the model still performs standard multi-image question answering.

6.1 Ablation Setup

Intervention. For each selected head (l, h), we zero the head's output contribution before the output projection, thereby removing its additive write to the residual stream. We use this output-side ablation because it is a stricter test of a head's causal contribution than query-side masking used in prior retrieval-head analysis [42]: query masking perturbs where a head attends but can still leave a value-weighted output, whereas our intervention removes the selected head's write itself. This output-side, head-level zero-ablation is in line with the broader causal-mediation and activation-patching literature for localizing functional roles in transformers [25, 36, 38].

We apply this intervention only over the image-retrieval span of the question. Following the Visual Haystacks format in Section 4.3, this span starts at the first token of the anchor object and runs to the last prefill position (the target object ends the question, so only a few format tokens follow it). The intervention is not applied during the answer generation.

Set partition. Directly comparing the pointer top-K and writer top-K sets would confound causal contribution with their partial overlap. We therefore form the union of the top-100 pointer heads and top-100 writer heads, and partition it into three disjoint subsets. The *pointer-writer overlap* contains heads selected by both criteria. The *pointer-only tail* contains heads in the pointer top-100 but not the writer top-100. The *writer-only tail* contains heads in the writer top-100 but not the pointer top-100. Across models, the pointer-writer overlap contains 35–47 heads, and each single-criterion tail contains 53 to 65 heads. We ablate the top-K heads from each subset for $K \in \{5, 10, 15, 20, 25, 30\}$, with matched-budget random-head controls at the same budgets. We evaluate at haystack sizes $N \in \{2, 3, 5, 10, 20, 50\}$. As N increases, baseline accuracy approaches the Yes/No chance level, reducing the dynamic range available for additional accuracy drops under ablation.

6.2 The Pointer-Writer Overlap Carries the Causal Effect

Ablating the pointer-writer overlap produces the most consistent accuracy drops among the tested head populations (Figure 3). The effect generally grows with the ablation budget and is most visible at haystack sizes 2–20, where the pointer-writer curve separates from both single-criterion tails and matched random-head controls. At larger haystack sizes, the unablated baseline is already closer to the Yes/No chance level, leaving less room for further accuracy drops, so ablation effects are smaller. Writer-only ablations show little consistent separation from random controls, suggesting that output-side recoverability alone does not identify the heads most relevant to standard inference in this intervention window. Pointer-only ablations have a modest and less consistent effect, but remain weaker than the pointer-writer overlap in the averaged results for $K \geq 10$. Overall, the strongest causal evidence points to heads selected by both criteria rather than to heads selected by either criterion alone. Per-model and per-haystack-size results are provided in Appendix I.

6.3 What the Intervention Isolates

The separation from matched random controls, especially at moderate haystack sizes, suggests that the drop is not merely due to removing heads in this sequence region or reducing head capacity in general. Rather, standard multi-image inference is most sensitive to the particular heads that both point to the target image and write recoverable target-image information.

Together, the pointer criterion reveals strong internal image-selection signals, while the writer criterion shows that attention mass alone does not fully characterize them. Targeted ablations further show that standard single-pass inference is most sensitive to heads selected by both criteria.

7 Discussion

Why the pointer-writer overlap matters. Pointing and writing are controlled by different parts of the attention: Q-K interactions determine where a head points, whereas the value pathway determines what information is written to the residual stream. Our set-partition ablations show that standard multi-image inference is most sensitive to heads where both properties coincide. Thus, image retrieval in LVLMs is not fully characterized by where a head attends; the relevant signal must also be written in a form available to downstream computation.

From analysis to intervention. Our results suggest that interventions based only on amplifying attention to the target image may be insufficient when the selected heads do not write target-image information. A possible future direction is to modify the value-side pathway of heads that already point well, for example by retraining or steering only a small set of value/output projections while leaving the rest of the model unchanged. Such targeted interventions could test whether improving the writer side of existing pointer heads can improve multi-image inference without full-model retraining.

Limitations. Our analysis focuses on four recent open-weight LVLMs from two model families, and broader model coverage would further clarify how general the pointer-writer structure is. Our naturalistic evaluation isolates single-target image retrieval in Visual Haystacks; extending the framework to cross-image comparison, multi-target grounding, and compositional reasoning is a natural direction for future work. Complementary generation-time interventions could further characterize how retrieved visual information is converted into the final textual answer.

8 Conclusion

We studied in-context image retrieval in LVLMs at the attention head level. Using ICIR-MCQ, we identified sparse *pointer heads* whose per-image attention mass ranks the target image first. These heads localize to a consistent mid-block region and transfer beyond the controlled probe, providing a strong input-side retrieval signal. However, attention mass alone is incomplete: a head selected by the pointer criterion is not necessarily selected by the writer criterion. We therefore introduced a retriever-based output-side *writer* criterion and found that pointer and writer rankings identify related but distinct head sets. Targeted set-partition ablations show that standard single-pass inference is most sensitive to heads selected by both criteria rather than to either single-criterion tail. Together, these results support a two-sided account of image retrieval in LVLMs: our strongest causal evidence favors heads selected by both the attention-based pointer and output-side writer criteria.

Acknowledgments and Disclosure of Funding

This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) (Nos. 2022R1A3B1077720, 2022R1A5A7083908, and RS-2026-25488668); by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government (MSIT) (No. RS-2021-II211343, Artificial Intelligence Graduate School Program, Seoul National University; No. RS-2022-II220959; No. IITP-2026-RS-2024-00397085, Leading Generative AI Human Resources Development; and No. IITP-2026-RS-2020-II201819, IITP-ICT Creative Consilience Program); and by the BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2026. This research was also supported by the ANCHOR through the Seoul ANCHOR Center, funded by the Ministry of Education (MOE) and the Seoul Metropolitan Government (2026-ANCHOR-01-003-09), and by Samsung Electronics Co., Ltd (IO260506-16457-01).

References

[1] Pravesh Agrawal, Szymon Antoniak, Emma Bou Hanna, Baptiste Bout, Devendra Chaplot, Jessica Chudnovsky, Diogo Costa, Baudouin De Monicault, Saurabh Garg, et al. Pixtral 12B. *arXiv preprint arXiv:2410.07073*, 2024.

[2] Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebrón, and Sumit Sanghai. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4895–4901, 2023.

[3] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35: 23716–23736, 2022.

[4] Ingeol Baek, Hwan Chang, Sunghyun Ryu, and Hwanhee Lee. How do large vision-language models see text in image? unveiling the distinctive role of OCR heads. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20452–20464, 2025.

[5] Kanghyun Baek, Jaihyun Lew, Chaehun Shin, Jungbeom Lee, and Sungroh Yoon. Diagnosing and correcting concept omission in multimodal diffusion transformers. In *International Conference on Machine Learning*, volume 306, pages 5107–5124, 2026.

[6] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

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

[8] Jing Bi, Junjia Guo, Yunlong Tang, Lianggong Bruce Wen, Zhang Liu, Bingjie Wang, and Chenliang Xu. Unveiling visual perception in language models: An attention head analysis approach. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 4135–4144, 2025.

[9] Miso Choi, Seonga Choi, Mincheol Kwon, Woosung Joung, Jinkyu Kim, and Jungbeom Lee. The truth stays in the family: Enhancing contextual truthfulness via inherited heads in model lineages. In *International Conference on Machine Learning*, volume 306, pages 20021–20044, 2026.

[10] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL workshop BlackboxNLP: analyzing and interpreting neural networks for NLP*, pages 276–286, 2019.

[11] Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352, 2023.

[12] Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, volume 2024, pages 35549–35562, 2024.

[13] Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations*, volume 2024, pages 2632–2652, 2024.

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

[15] Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, 2023.

[16] Yanbei Jiang, Xueqi Ma, Shu Liu, Sarah Monazam Erfani, Tongliang Liu, James Bailey, Jey Han Lau, and Krista A Ehinger. Investigating the functional roles of attention heads in vision language models: Evidence for reasoning modules. *arXiv preprint arXiv:2512.10300*, 2025.

[17] Seil Kang, Jinyeong Kim, Junhyeok Kim, and Seong Jae Hwang. See what you are told: Visual attention sink in large multimodal models. In *International Conference on Learning Representations*, volume 2025, pages 87676–87703, 2025.

[18] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 6769–6781, 2020.

[19] Yongjoo Kim, Mincheol Kwon, Seonga Choi, Minseung Lee, Kyeong-Jin Oh, Hyunyoung Lee, Yunsu Choi, and Jungbeom Lee. SAGE: Semantic attribute graphs for multi-entity visual retrieval. In *Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing*, 2026.

[20] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7057–7075, 2020.

[21] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. LLaVA-OneVision: Easy visual task transfer. *Transactions on Machine Learning Research*, 2025.

[22] Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36:41451–41530, 2023.

[23] Mingxin Li, Yanzhao Zhang, Dingkun Long, Keqin Chen, Sibo Song, Shuai Bai, Zhibo Yang, Pengjun Xie, An Yang, Dayiheng Liu, et al. Qwen3-VL-Embedding and Qwen3-VL-Reranker: A unified framework for state-of-the-art multimodal retrieval and ranking. *arXiv preprint arXiv:2601.04720*, 2026.

[24] 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 *European conference on computer vision*, pages 740–755. Springer, 2014.

[25] Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. *Advances in neural information processing systems*, 35:17359–17372, 2022.

[26] Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. Towards interpreting visual information processing in vision-language models. In *International Conference on Learning Representations*, volume 2025, pages 57172–57189, 2025.

[27] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.

[28] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[29] Vedant Palit, Rohan Pandey, Aryaman Arora, and Paul Pu Liang. Towards vision-language mechanistic interpretability: A causal tracing tool for BLIP. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 2856–2861, 2023.

[30] Junsung Park, Jungbeom Lee, Jongyoon Song, Sangwon Yu, Dahuin Jung, and Sungroh Yoon. Know "no" better: A data-driven approach for enhancing negation awareness in CLIP. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2825–2835, 2025.

[31] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[32] Sarah Schwettmann, Neil Chowdhury, Samuel Klein, David Bau, and Antonio Torralba. Multimodal neurons in pretrained text-only transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*, pages 2862–2867, 2023.

[33] Aditya Sharma, Michael Saxon, and William Yang Wang. Losing visual needles in image haystacks: Vision language models are easily distracted in short and long contexts. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 5429–5451, 2024.

[34] Xinyu Tian, Shu Zou, Zhaoyuan Yang, and Jing Zhang. Identifying and mitigating position bias of multi-image vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10599–10609, 2025.

[35] Eric Todd, Millicent Li, Arnab Sen Sharma, Aaron Mueller, Byron Wallace, and David Bau. Function vectors in large language models. In *International conference on learning representations*, volume 2024, pages 17282–17333, 2024.

[36] Jesse Vig, Sebastian Gehrmann, Yonatan Belinkov, Sharon Qian, Daniel Nevo, Yaron Singer, and Stuart Shieber. Investigating gender bias in language models using causal mediation analysis. *Advances in neural information processing systems*, 33:12388–12401, 2020.

[37] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 5797–5808, 2019.

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

[39] Weiyun Wang, Shuibo Zhang, Yiming Ren, Yuchen Duan, Tiantong Li, Shuo Liu, Mengkang Hu, Zhe Chen, Kaipeng Zhang, Lewei Lu, et al. Needle in a multimodal haystack. *Advances in Neural Information Processing Systems*, 37:20540–20565, 2024.

[40] Weiyun Wang, Zhangwei Gao, Lixin Gu, Hengjun Pu, Long Cui, Xingguang Wei, Zhaoyang Liu, Linglin Jing, Shenglong Ye, Jie Shao, et al. InternVL3.5: Advancing open-source multimodal models in versatility, reasoning, and efficiency. *arXiv preprint arXiv:2508.18265*, 2025.

[41] Tsung-Han Wu, Giscard Biamby, Jerome Quenum, Ritwik Gupta, Joseph E Gonzalez, Trevor Darrell, and David M Chan. Visual haystacks: A vision-centric needle-in-a-haystack benchmark. In *International Conference on Learning Representations*, volume 2025, pages 64592–64615, 2025.

[42] Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. In *International Conference on Learning Representations*, volume 2025, pages 62143–62156, 2025.

[43] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986, 2023.

[44] Zhi Zhang, Srishti Yadav, Fengze Han, and Ekaterina Shutova. Cross-modal information flow in multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19781–19791, 2025.

A Implementation Details

Model shape. The four LVLMs share the same language-model shape: 36 transformer blocks with 32 attention heads each, $L \cdot H = 1{,}152$ heads in total, and head dimension $d_h = 128$. Each model is loaded from its HuggingFace snapshot in bfloat16 and run with FlashAttention-2 [12].

Attention score and head-output capture. We register forward hooks on `q_proj` and `k_proj` of each transformer block to capture the query and key projections needed to recompute per-head attention scores during prefill, and a forward-pre-hook on `o_proj` to capture each head's output (the value-weighted sum entering the output projection) for the writer probe. For models with Grouped Query Attention [2], we expand KV heads to match the query heads before computing per-head attention. The pointer score (Eq. 3) is computed from these captured projections at the designated readout row only; the forward pass itself is unchanged.

Visual Haystacks setup. For each haystack size, a fixed random seed (42) controls filler-image selection and haystack shuffling, so that every model sees the same haystack composition and order.

Image preprocessing and per-image input budget. We fix the per-image input budget at 448×448 pixels for all four LVLMs we study so that cross-model comparisons are not confounded by input size. **InternVL3.5** uses a 448×448 resize (BICUBIC) followed by ImageNet normalization, yielding 256 visual tokens per image (the architecture's native per-tile budget). **Qwen3-VL** uses an aspect-ratio-preserving resize within a configurable pixel budget by default; we override the processor with `min_pixels = max_pixels =` 448×448, which fixes the pixel budget rather than the resolution: the aspect ratio is preserved and each side is rounded to a multiple of 32, giving roughly 192 visual tokens per image, about three quarters of InternVL3.5's 256. The same transforms are used in baseline and ablation runs, so within-model comparisons are unaffected by the token-count difference between the two families.

Generation configuration. All forward passes run in bfloat16. Decoding is greedy (`do_sample=False`, `num_beams=1`) with `max_new_tokens=128`, so any randomness in answers is removed. For VHs (Yes/No format), generation stops at the first "yes" or "no" word, with no effect on parsed accuracy.

External retriever evaluation. We evaluate CLIP ViT-L/14, SigLIP so400m-patch14-384, and Qwen3-VL-Embedding-8B as external retrievers on Visual Haystacks. For CLIP and SigLIP, we follow the standard OpenAI 80-template prompt ensemble [31]: the text embedding for the anchor object is the mean of the ℓ_2-normalized embeddings of the 80 templated prompts, and the per-image score is the cosine similarity between this ensembled text embedding and the image embedding. The anchor noun is inserted without its article, and SigLIP prompts are padded to the 64-token maximum length used in its pretraining, which its text encoder requires. For Qwen3-VL-Embedding-8B we use a natural retrieval phrasing query, "an image with {anchor}", and score images by cosine similarity to this query embedding. In all three retrievers we select the argmax image as the predicted target.

Ablation procedure. For the targeted ablation experiments (§6), a forward pre-hook on `o_proj` zeroes the selected head's slice of the `o_proj` input over the image-retrieval span: from the first token of the anchor-object phrase to the last prefill position (the target-object phrase ends the question, so the remaining covered tokens are the fixed answer-format suffix). The first answer token is decoded from the last prefill position, where the ablation is already in effect (Appendix I.1 checks that continuing the intervention through generation does not change the conclusions). Random selection draws K heads uniformly from the $1{,}152$ heads at the matched budget (averaged over three seeds in the set-partition ablation of §6.1); the unablated baseline runs the same pipeline with no head zeroed.

Computational resources. All main-text experiments run on NVIDIA L40 GPUs (48GB VRAM). Single-GPU inference suffices for all four LVLMs at the haystack sizes evaluated in this paper ($N \in \{2, 3, 5, 10, 20, 50\}$). A Visual Haystacks pass at $N{=}10$ takes about 1.5 s per sample for InternVL3.5-8B and about 4 s for Qwen3-VL-8B, whose answers are longer, and head-output extraction on ICIR-MCQ about 13 s per sample; the set-partition grid of §6 comprises about 220 cells of 1,000 samples per LVLM, on the order of a few hundred GPU-hours per LVLM.

B ICIR-MCQ: Additional Construction Details

ICIR-MCQ is described in the main text (§3.2) and used as the head-discovery benchmark throughout §4 and §5. This appendix gives the design rationale, the sampling procedure, the split sizes, and one abridged prompt.

Why this design (extended). Three properties matter for head-level attribution. First, the answer is a single discrete token whose correctness is unambiguous. Second, all four caption options mention the queried category (the three distractor options are captions of other images of that category), so the answer cannot be identified from the category cue alone; the distractor images are drawn from other categories, so the target image is unambiguous given the cue. Third, the small candidate set ($N \leq 6$) keeps the head-level R@1 estimator stable at the 6,000-sample scale, allowing a full sweep of all $L \cdot H$=1,152 heads without budgeted subsetting.

Sampling procedure. We draw each sample by (i) selecting a target image and one of its own COCO captions that mentions the queried category (sampling from the union of the MS-COCO 2017 train and validation splits), (ii) selecting $N_{\text{choices}} - 1$ distractor images from other COCO categories, excluding any image annotated with the target's category, so that the queried category is annotated in the target image alone, and (iii) constructing a four-way multiple choice over caption options A–D, in which one option is the target's own caption and the other three are captions of other images of the queried category, so that all four options mention the category. We constrain (a) the target image's position among the N_{choices} images to be uniform across the N_{choices} slots, and (b) the answer letter (the slot of the target's caption among the four options A–D) to be uniformly distributed across {A, B, C, D}. The number of MCQ option letters (always four) is independent of N_{choices}; the candidate images themselves are unlabeled and identified by their position in the prompt rather than by a letter.

Split sizes. We release two splits: N_{choices}=5 with 3,000 samples and N_{choices}=6 with 3,000 samples, for a total of 6,000. The released file, its data card, and the builder are available at `https://huggingface.co/datasets/jerryray/ICIR-MCQ`; the analysis code is at `https://github.com/parkquasar/Image-Retrieval-Heads`.

Robustness of head rankings. The top head is invariant between the two N_{choices} splits for both 8B models (L21H10 in both cases), and the top-5 sets coincide across the splits, with only the ordering of ranks 2–5 differing.

B.1 An abridged ICIR-MCQ prompt

Figure 1 is a shortened schematic of another sample of the same category. Below is the prompt of one actual sample (sample 4860 of the released file; six candidate images), with each text passage cut to its first sentence or so. Passages are the interleaved documents inherited from MM-NIAH (§3.2) and are unrelated to the images; the images are inserted at the `<image>` markers in order. The category is named in the question and again in each of the four options, which describe different images of that category, so the last mention of the category (the cattok) falls inside the options (option D here).

```
What the real geeks soon realize, however, is that it's fun to brew with the
  Aeropress. You can literally try thousands of different recipes and still
  [...]
<image 1>
"You said an interesting word, 'rumor.' I've been in the business [...]
<image 2>
Here is the equivalent circuit diagram:The below code will demonstrate how it
  works by turning the motor on and off. [...]
<image 3>
The below circuit can be used to demonstrate. Note that there are no external
  components at all. [...]
<image 4>
It includes one one-bedroom unit, and 11 two-bedroom units. The property also
  has parking, controlled access entry and a laundry facility. [...]
<image 5>
Weirick will specialize in working with landlords and tenants regarding their
```

```
  industrial real estate needs in the San Fernando Valley. [...]
<image 6>
CDC's latest plan, which was accepted, was to remove the toxic soil and pile
  it up on three separate berms on the property. [...]
Which caption best describes the image above that contains zebra? Answer
  with a single letter (A, B, C, or D).
A) A group of zebras and wildebeest grazing in a field.
B) The two zebra stand next to each other in the wilderness.
C) A group of zebras and other animals grazing in a field with a rainbow in
  the background.
D) The zebra eats the grass near a building.
```

Target image: position 4 of 6. Correct option: A.

C Pointer Heads: Extended Results

This appendix collects pointer-head results that are summarized in the main text: the per-LVLM head R@1 grid, pairwise ranking similarity across the four LVLMs, the per-haystack R@1 of the top pointer head on Visual Haystacks, and that head's R@1 next to the model's own accuracy.

C.1 Per-LVLM head R@1 grid

Figure 4 shows the pointer R@1 of every head in each LVLM.

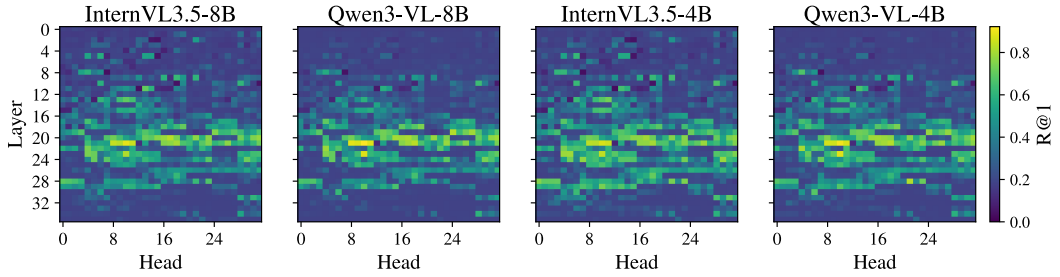


Figure 4: Head-level attention-mass R@1 on ICIR-MCQ for all four LVLMs, shown as layers $\times$ heads with a shared color scale. Pointer ability concentrates in a narrow mid-block layer band, and the same band appears across both model families and both parameter scales. The strongest pointer head per model is `L21H10` on every panel (Table 1).

C.2 Pairwise ranking similarity across the four LVLMs

Table 4 reports pairwise statistics on the per-head R@1 ranking across the four LVLMs. For every pair of models it reports four statistics on the per-head R@1 ranking, defined in the caption. Pairs are grouped into shared-LM-backbone (same LM scale across the two families) and cross-LM-backbone (different LM scale).

Rankings are highly aligned for every pair (Spearman ≥ 0.90, top-100 overlap ≥ 0.75); the top-3 set is identical within each shared-LM-backbone pair and shares 2 of 3 heads across backbones.

C.3 Per-haystack R@1 decay

Table 5 reports the per-haystack R@1 of `L21H10`, the top ICIR-MCQ pointer head on every model, on Visual Haystacks. Ranked by VHs R@1 at N=10, it is also the top head on both 8B models and ranks second on InternVL3.5-4B and third on Qwen3-VL-4B. R@1 generally falls with haystack size, and the two scales within a family are not consistently ordered: the InternVL3.5 pair stays within 0.8 pp at every haystack size, while Qwen3-VL-4B trails Qwen3-VL-8B by up to 6.5 pp at $N \in \{2, 3, 20\}$ and matches it at $N \in \{5, 10\}$. The pointer head is strong at both scales, in line with the cross-scale stability noted in §4.2, but R@1 itself is not ordered by model scale.

Table 4: Pairwise per-head R@1 similarity across the four LVLMs. Pearson and Spearman are computed on the full $1{,}152$-d head ranking; layer-max correlation is on the per-layer maximum R@1 vector (length 36); top-K overlap is $|\mathcal{H}_K^{\mathrm{ptr}}(\text{model 1}) \cap \mathcal{H}_K^{\mathrm{ptr}}(\text{model 2})|/K$. The top-1 pointer head is `L21H10` on every model.

	Pearson	Spearman	LayerMax	Top-3	Top-10	Top-100
Shared-LM-backbone pairs						
InternVL3.5-8B ↔ Qwen3-VL-8B	0.965	0.922	0.921	1.00	0.70	0.86
InternVL3.5-4B ↔ Qwen3-VL-4B	0.966	0.927	0.933	1.00	0.60	0.80
Cross-LM-backbone pairs						
InternVL3.5-8B ↔ InternVL3.5-4B	0.977	0.946	0.962	0.67	0.70	0.85
Qwen3-VL-8B ↔ Qwen3-VL-4B	0.971	0.939	0.980	0.67	0.60	0.85
InternVL3.5-8B ↔ Qwen3-VL-4B	0.964	0.916	0.919	0.67	0.70	0.82
InternVL3.5-4B ↔ Qwen3-VL-8B	0.937	0.900	0.932	0.67	0.70	0.75

Table 5: R@1 of the shared top pointer head `L21H10` on VHs across haystack sizes $\in \{2, 3, 5, 10, 20\}$ (n=1,000 per cell). Cattok-position attention argmax accuracy. N=50 is not included in this readout.

Haystack size	2	3	5	10	20
InternVL3.5-8B	.947	.914	.860	.787	.677
Qwen3-VL-8B	.959	.931	.901	.832	.782
InternVL3.5-4B	.939	.907	.859	.794	.682
Qwen3-VL-4B	.894	.901	.895	.836	.745

C.4 Single-head retrieval versus the model's own accuracy

Table 6 places the top pointer head's retrieval R@1 next to the accuracy of the unmodified model on the same inputs, for both benchmarks. On ICIR-MCQ the model answers with a letter and we score the restricted argmax over the four option tokens at the first answer position, on all 6,000 samples. On Visual Haystacks ($N = 10$) accuracy is the Yes/No accuracy of the unablated baseline used in §6. On ICIR-MCQ the head's R@1 and the model's accuracy are within a few points of each other, whereas on Visual Haystacks the single head identifies the target image 11 to 16 points more often than the model answers the question correctly.

Table 6: Top pointer head (L21H10) retrieval R@1 and base-model accuracy on ICIR-MCQ (6,000 samples) and Visual Haystacks ($N = 10$, 1,000 samples).

	ICIR-MCQ		Visual Haystacks	
Model	Head R@1	Model acc. (%)	Head R@1	Model acc. (%)
InternVL3.5-4B	.866	85.5	.794	68.1
InternVL3.5-8B	.890	88.9	.787	65.9
Qwen3-VL-4B	.905	90.1	.836	67.5
Qwen3-VL-8B	.922	91.3	.832	67.6

D Writer probe: training details and extended agreement analysis

This appendix gives the training details of the writer probe (§5) and compares the writer and pointer rankings; Appendices E and G report control experiments on the candidate pool, on object regions, and on the readout position.

Head-level training setup. For each attention head, we treat its attention-value-weighted output at the query position as a d_h-dimensional image embedding and train a small head-level projection with a contrastive (InfoNCE-style) objective using ICIR-MCQ as positive-negative pairs. Reading head-level representations through a small trained probe follows the inference-time-intervention [22] and function-vector [35] setups; we adapt the construction to a cross-modal retrieval objective. The

probe projects the head's output and the candidate image embedding into a shared 256-dimensional space through one linear layer each, followed by L2 normalization, and scores candidates by dot product. We selected this configuration (one projector layer, L2 normalization, batch 256, learning rate 1×10^{-3}) from a sweep over architecture, projector depth, batch size, learning rate, and normalization on the 1,200-example evaluation split used for checkpoint selection and the reported ranking; the Qwen3-VL-8B probe uses batch 128. The LVLM weights are not modified; only the head-level probe is trained.

Agreement with the pointer ranking. The two criteria both concentrate in mid-block layers, but not on the same layers: the pointer ranking peaks at layer 21 on all four LVLMs, whereas the strongest writer heads sit at layer 24 on three of the four (layer 21 on InternVL3.5-4B). The writer top-100 is also more layer-dispersed than the pointer top-100 (47–60 versus 71–82 of 100 heads inside layers 18–27). The two rankings likewise disagree on which specific heads are top-ranked; the writer top-5 per LVLM are reported in Table 3 and do not match the pointer top-5 of Table 1. Figure 5 shows the full per-head writer R@1 grid, mirroring the pointer-side grid in Figure 4: the bright band sits in the same mid-block region in both rankings, with the writer band extending slightly further toward later layers. Ranks diverge more in the mid-tail than at the top.

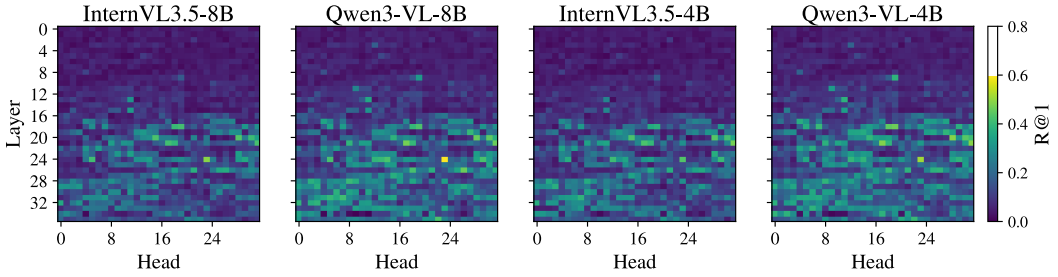


Figure 5: Head-level writer R@1 on the ICIR-MCQ held-out evaluation pool (1,200 candidate images; chance R@1 $\approx$ .001) for all four LVLMs, shown as layers $\times$ heads with a shared color scale. Writer R@1 concentrates in a narrow mid-block layer band, the same region picked out by the pointer ranking (Figure 4).

Writer-ranked vs. pointer-ranked ablation. Whether the heads that only one criterion selects are causally used is tested by the set-partition ablation (§6.1, Appendix I): at $K \geq 10$ the overlap of the two top-100 sets produces the largest drops on average across models, the writer-only tail leaves accuracy near the random control on all four LVLMs, and the pointer-only tail is model-dependent (on InternVL3.5-8B it drops as much as or more than the overlap at $N \geq 5$, $K \geq 25$; near the control on Qwen3-VL-8B).

E Writer Criterion: Object-Region and In-Context Controls

E.1 Object regions instead of full images

The writer probe (§5) retrieves the embedding of the whole target image, so a head could in principle rank well by carrying global image identity rather than information about the queried object. We therefore retrained the per-head retriever with the target represented by the embedding of the object region, the COCO box of the queried category inside the target image, with the probe configuration of Appendix D. As a control for training variance we also retrained the full-image probe once more under the same settings (matched control). Table 7 compares the resulting head rankings with the full-image writer ranking of the main text, and reports a same-category retrieval test in which the candidate pool contains only object regions of the queried category taken from other images, so that category identity alone cannot solve the task; for a query with m_i same-category distractor regions chance is $1/(m_i + 1)$, averaged over queries.

The object-region ranking agrees with the full-image writer ranking at $\rho = 0.970$ to 0.971, somewhat below the agreement between two full-image trainings ($\rho \geq 0.993$, 93 to 96 shared top-100 heads), so object information changes the ranking by more than training noise but leaves it largely intact. On

Table 7: Object-region writer probes. Spearman ρ of the object-region head ranking against the full-image writer ranking (all 1,152 heads); ρ and top-100 overlap of a retrained full-image control against the same ranking; same-category retrieval with the mean number m of same-category distractor regions, the per-query chance $1/(m_i + 1)$ averaged over queries, and mean R@1 over the top-100 object-region heads and over the top-10.

	Ranking agreement		Same-category retrieval			
Model	object vs. full	control ρ / overlap	m	chance	top-100 R@1	top-10 R@1
InternVL3.5-4B	0.970	0.993 / 93	12.1	0.091	0.466	0.592
InternVL3.5-8B	0.970	0.993 / 96	14.2	0.077	0.424	0.578
Qwen3-VL-4B	0.971	0.996 / 96	16.4	0.064	0.485	0.633
Qwen3-VL-8B	0.970	0.996 / 96	16.4	0.064	0.429	0.590

the same-category test the top-100 object-region heads retrieve the correct region in 42 to 48% of queries against a chance of 6 to 9%, and the top-10 heads in 58 to 63%. Writer heads therefore carry information about the queried object itself, not only about which image it came from.

E.2 In-context candidates instead of a held-out pool

The writer criterion scores each head against a held-out pool of 1,200 candidate images. A natural alternative scores it only against the images of the same context, five or six candidates, which is closer to the retrieval the model actually performs. Table 8 compares this in-context ranking with the held-out writer ranking and with the pointer ranking, using a paired head bootstrap (5,000 resamples) for the difference of the two correlations, and a split-half diagnostic in which the test samples are divided into two halves, heads are ranked on each half, and we count the top-100 heads shared by the two halves.

Table 8: In-context writer ranking versus the held-out writer and pointer rankings (Spearman ρ over 1,152 heads; $\Delta = \rho_{\text{writer}} - \rho_{\text{pointer}}$ with its 95% bootstrap CI), and split-half stability of the two writer scorings: top-100 heads shared between halves and the R@1 gap between the rank-1 and rank-100 head (points).

	Agreement of the in-context ranking				Split-half stability			
					shared top-100		rank 1 vs. 100 gap	
Model	vs. writer	vs. pointer	Δ	95% CI	in-ctx	held-out	in-ctx	held-out
InternVL3.5-4B	0.758	0.480	0.277	[0.231, 0.325]	65	85	4.3	22.0
InternVL3.5-8B	0.716	0.530	0.186	[0.143, 0.231]	71	87	4.0	24.4
Qwen3-VL-4B	0.573	0.403	0.170	[0.124, 0.217]	61	86	1.8	19.5
Qwen3-VL-8B	0.657	0.406	0.251	[0.202, 0.302]	63	84	2.2	28.9

The in-context ranking is closer to the held-out writer ranking than to the pointer ranking in every model, with the difference bounded away from zero. It is, however, a poorly resolved criterion: with only a handful of candidates most heads score near the ceiling, the rank-1 and rank-100 heads differ by only 1.8 to 4.3 points, and the two halves of the test set agree on just 61 to 71 of their top-100 heads, against 84 to 87 for the held-out scoring whose rank-1 to rank-100 gap is about 20 to 29 points. The held-out pool is used in the main text for this reason.

F Where Writer-Only Heads Get Their Information

Writer-only heads are a relative partition: heads in the writer top-100 but outside the pointer top-100 (§6.1). The probe recovers the target image from their outputs although they fall outside the pointer top-100, which raises the question of where that information comes from: from the target-image tokens themselves, or relayed through text positions. Following Eq. 4, we decompose each writer-only head's output at the cattok into the contributions of four source groups: target-image tokens, distractor-image tokens, interleaved filler passages, and the remaining question and option tokens. We then apply the writer probe of §5, unchanged, to the output reconstructed from one source group alone or with one group removed, and report the mean writer R@1 over the writer-only heads of each model

(Table 9). Because the probe and the cosine ranking are nonlinear, a single source can score slightly above the full output; the rows are sufficiency and removal tests, not an additive decomposition. The two removal rows carry 95% confidence intervals from 5,000 paired head-bootstrap resamples within each model's writer-only set; both exclude zero in every model.

Table 9: Writer R@1 of writer-only heads (mean over the set) when the head output is reconstructed from one source group or with one group removed. The two removal rows give the change relative to the full output, with its 95% bootstrap CI beneath. Text tokens are the interleaved filler passages plus the question and option tokens (including special tokens).

Source intervention	InternVL3.5-4B	InternVL3.5-8B	Qwen3-VL-4B	Qwen3-VL-8B
Full head output	0.331	0.285	0.352	0.346
Target-image tokens only	0.338	0.341	0.322	0.326
Remove target-image tokens (Δ)	-0.175	-0.160	-0.163	-0.167
95% CI	$[-0.198, -0.150]$	$[-0.182, -0.138]$	$[-0.185, -0.140]$	$[-0.189, -0.144]$
Text tokens only	0.155	0.126	0.190	0.178
Remove all text tokens (Δ)	-0.049	-0.025	-0.085	-0.075
95% CI	$[-0.075, -0.025]$	$[-0.050, -0.001]$	$[-0.112, -0.061]$	$[-0.101, -0.051]$
Interleaved filler only	0.003	0.003	0.002	0.003
Distractor-image tokens only	0.004	0.003	0.002	0.002

The target-image tokens alone retain a writer R@1 comparable to the full output, and removing them causes the largest drop in every model. Text tokens alone stay well below the full output, and removing all text costs much less than removing the target-image tokens; interleaved filler and distractor-image tokens alone carry almost nothing. Text therefore provides secondary information, but a relay through text positions is not required for the target-image information that the probe recovers.

Attention patterns and pointer ranks. The attention patterns point in the same direction. At the cattok and at positions $+1$ to $+4$, writer-only heads place 2.1 to 4.0 times as much attention mass on the target image as layer-matched random heads, while their attention to the interleaved filler is below that of random heads (0.74 to 0.89 times). Nor do they become strong pointers at a neighboring position: across $+1$ to $+4$ the median pointer rank of writer-only heads stays above 200 of 1,152, and at none of these positions in any model do more than 12 of them enter the pointer top-100. Attention magnitude alone also does not determine output content: among heads with similar pointer R@1 (0.30 to 0.40), writer R@1 differs by 4.0 to 4.2 times between the 10th and 90th percentiles, consistent with Eq. 4, in which attention weights set how sources are mixed while the value pathway sets what the mixture carries [20].

Scope. The decomposition localizes the source positions of the current head's output; it does not say how the target-image-token representations formed in earlier layers. Writer R@1 measures probe recoverability, not downstream use: writer-only ablations remain near the matched-budget random-head controls (§6.1, Appendix I).

G Readout Position: Both Signals Are Anchored at the Cattok

The pointer and writer criteria are read at the cattok, the last token of the queried-category mention (§3.2). This appendix checks how much of each signal survives when the readout moves to later positions: the four tokens following the cattok ($+1$ to $+4$), the token midway between the cattok and the end of the prompt (mid), and the last prefill token (last). The pointer ranking is recomputed at every position from the attention mass of that position over each image span (Eq. 3) on the 6,000 ICIR-MCQ samples. The writer probes are retrained independently at each position ($+1$, $+2$, $+4$, mid, last) with the configuration of Appendix D; a probe retrained at the cattok itself reproduces the cattok writer ranking at Spearman $\rho = 0.993$ on every model, which serves as the reference for retraining variability. Table 10 reports, for each position, the Spearman correlation of the full 1,152-head ranking with the cattok ranking together with the top-100 overlap, and the retained signal of the head sets fixed at the cattok: the mean R@1 of the cattok pointer top-100 (writer top-100) at that position divided by its value at the cattok.

Table 10: Readout-position control. First row per model: Spearman ρ of the pointer ranking at each position against the cattok pointer ranking, and the top-100 overlap. Second row: mean pointer R@1 of the pointer top-100 of this sweep's cattok ranking, at each position divided by its value at the cattok. Third row: the same correlation and overlap for the writer ranking retrained at each position; the cattok entry is the retrained cattok probe against the original cattok ranking. Fourth row: mean writer R@1 of the fixed cattok writer top-100 under the probe retrained at each position, divided by its value under the retrained cattok probe. Writer probes were not retrained at $+3$.

	cattok	$+1$	$+2$	$+3$	$+4$	mid	last
InternVL3.5-4B	–	0.960 / 79	0.946 / 77	0.945 / 75	0.948 / 76	0.957 / 78	0.683 / 52
pointer set	1.00	0.86	0.81	0.80	0.86	0.82	0.34
writer ranking ρ / overlap	0.993 / –	0.850 / 72	0.812 / 67	–	0.847 / 70	0.810 / 65	0.536 / 16
writer set	1.00	0.73	0.57	–	0.68	0.51	0.32
InternVL3.5-8B	–	0.957 / 77	0.949 / 74	0.948 / 73	0.943 / 72	0.959 / 73	0.682 / 50
pointer set	1.00	0.85	0.81	0.80	0.85	0.84	0.41
writer ranking ρ / overlap	0.993 / –	0.847 / 63	0.792 / 62	–	0.823 / 59	0.818 / 57	0.558 / 17
writer set	1.00	0.69	0.54	–	0.62	0.51	0.46
Qwen3-VL-4B	–	0.961 / 87	0.949 / 81	0.946 / 82	0.945 / 85	0.959 / 82	0.739 / 51
pointer set	1.00	0.84	0.77	0.78	0.85	0.79	0.60
writer ranking ρ / overlap	0.993 / –	0.878 / 67	0.820 / 59	–	0.857 / 60	0.803 / 53	0.713 / 26
writer set	1.00	0.63	0.44	–	0.55	0.36	0.82
Qwen3-VL-8B	–	0.959 / 87	0.943 / 85	0.942 / 82	0.939 / 86	0.954 / 80	0.727 / 60
pointer set	1.00	0.85	0.77	0.77	0.85	0.78	0.61
writer ranking ρ / overlap	0.993 / –	0.875 / 67	0.829 / 64	–	0.851 / 68	0.815 / 57	0.730 / 33
writer set	1.00	0.67	0.46	–	0.56	0.39	0.98

Both rankings stay close to their cattok version at the local positions and at mid: pointer ρ 0.939–0.961 with top-100 overlap 72–87, writer ρ 0.79–0.88, and the top-1 pointer head is unchanged at every local position in every model. The fixed sets keep a substantial fraction of their signal there: at $+1$ to $+4$ the pointer set retains 0.77–0.86 of its cattok R@1 and the writer set 0.44–0.73. At the last prefill token both rankings move further from the cattok ranking (pointer ρ 0.68–0.74, writer ρ 0.54–0.73), but both signals remain measurable. The discovered head populations therefore persist beyond the single scoring position, while the cattok remains the position at which both signals are strongest.

H Majority-vote retrieval with the writer ranking

This appendix asks how the writer ranking transfers to Visual Haystacks [41], and whether ensembling the top-K writer-ranked heads recovers the gap to the single best pointer head. We pair each ranking with the scoring rule it was ranked under, so the comparison is between matched ranking–metric pairs rather than between two metrics applied to a fixed head set.

Head-level vote and majority-vote scoring. Let $\mathcal{H}_K^{\mathrm{ptr}}$ and $\mathcal{H}_K^{\mathrm{wri}}$ denote the top-K heads under the pointer and writer rankings. For a VHs sample with image set $\{e_i\}_{i=1}^N$:

- Pointer-ranking, attention-score voting (A): each $h \in \mathcal{H}_K^{\mathrm{ptr}}$ casts a vote $v_h^A = \arg\max_i \, a_{h,i}$, where $a_{h,i}$ is the cattok attention sum from head h to image i's patches.
- Writer-ranking, trained-probe voting (B): each $h \in \mathcal{H}_K^{\mathrm{wri}}$ casts a vote

$$v_h^B = \arg\max_i \left\langle \tilde{f}_h(o_h),\, \tilde{g}_h(e_i) \right\rangle,$$

  where f_h and g_h are the trained head-side and image-side projectors of head h (Appendix D), and $\tilde{\cdot}$ denotes ℓ_2 normalization.

The sample-level prediction is the majority vote over the K votes (with ties broken by smallest image index); $K = 1$ recovers the standalone single-head predictor in each case. We report R@1 against the ground-truth target image position on $1{,}000$ samples per haystack size for all four LVLMs; Figure 6 plots the per-LVLM dose-response.

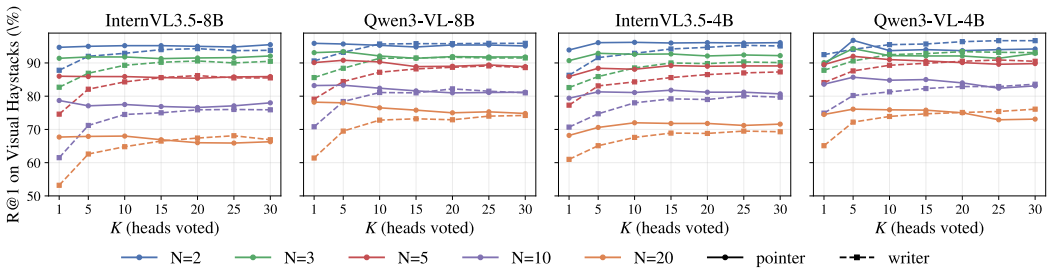


Figure 6: Top-K majority-vote R@1 on Visual Haystacks under the two matched ranking–metric pairs, for all four LVLMs (n=1,000 per haystack size). Solid lines: pointer ranking, attention-score voting (A). Dashed lines: writer ranking, trained-probe voting (B). Each color is one haystack size $\in \{2, 3, 5, 10, 20\}$.

Single head: the pointer ranking is stronger in 19 of 20 cells. At K=1 the pointer top head outperforms the writer top head in 19 of 20 (model, N) cells, by 2.3 to 17.2 pp; the exception is Qwen3-VL-4B at N=2 (89.4 vs. 92.5). The gap widens with haystack size, to about 17 pp on the 8B models and about 9 pp on the 4B models. The pointer ranking thus identifies the stronger individual head, even though the writer probe was trained for retrieval.

Ensembling closes most of the single-head gap. Majority vote over the top-K writer-probe heads improves with K and saturates by $K \approx$ 20–30 on all four models; at saturation it is within 4.0 pp of the best pointer configuration in every (model, N) cell, within 1.2 pp at N=2, and matches or marginally exceeds it in four cells. The pointer ranking behaves differently, and the difference depends on scale: on the 8B models a single head is already at or near the best (adding heads gains at most 0.8 pp), while on the 4B models a small ensemble helps (up to +7.4 pp at K=5 on Qwen3-VL-4B). The pointer ranking thus identifies a few heads whose raw attention pattern is by itself an accurate retriever, whereas the writer ranking identifies heads whose outputs carry retrieval information that is realized in aggregate rather than head by head.

I Set-Partition Ablation: overlap vs. pointer-only vs. writer-only

To isolate which part of the pointer top-100 and writer top-100 drives the causal effect, we partition their union into three disjoint subsets and run the targeted output-zero ablation of §6 (Appendix A) on each at $K \in \{5, 10, 15, 20, 25, 30\}$ and $N \in \{2, 3, 5, 10, 20, 50\}$. Within each subset heads are ordered by pointer rank (overlap, pointer-only) or writer rank (writer-only), and the first K heads of that order are ablated; the overlap subset has 35–47 heads and pointer-only and writer-only 53–65 each, depending on the model. A random control of the same K uniformly sampled heads serves as the reference. Figure 7 plots the per-model dose-response curves.

Table 11 lists the overlap heads of each model in ablation order. The pointer top-1 head `L21H10` is in the overlap of every model (writer rank 35/33/55/50 on InternVL3.5-8B / InternVL3.5-4B / Qwen3-VL-8B / Qwen3-VL-4B), and 23 heads are in the overlap of all four models.

I.1 Prefill-only versus full-generation intervention

The ablations above stop at the last prefill position, so the first Yes/No logit already reflects the intervention while the generated tokens are untouched (Appendix A). To check that stopping at the prefill does not change the conclusions, we re-ran the three arms and the three random controls at $N = 10$, $K = 30$ with the zeroing continued through every generated token, and compared the two runs sample by sample (Table 12). For both InternVL models, which answer with a single word, no prediction changes. For the Qwen3-VL models, whose answers can run to a full sentence (mostly on the 8B model), continuing the intervention touches tokens after the decision and changes at most 5.1% of parsed answers, moving accuracy by at most 0.9 points without changing the ordering of the arms.

Table 11: Overlap heads (in both pointer top-100 and writer top-100, cattok query) per LVLM, listed by pointer rank, the order in which the overlap subset is ablated (the first K heads form the budget-K overlap arm). Format LxxHyy(ax/ry) gives the head's 1-indexed pointer rank x and writer rank y.

Model	Overlap heads (pointer-rank order)
InternVL3.5-8B (47 heads)	L21H10(a1/r35), L21H11(a2/r46), L21H08(a3/r20), L20H17(a4/r83), L21H18(a7/r10), L20H16(a8/r81), L20H31(a9/r16), L20H21(a10/r68), L21H17(a11/r6), L20H15(a14/r3), L19H24(a17/r97), L20H30(a18/r8), L22H04(a20/r34), L23H04(a21/r82), L20H28(a23/r2), L21H31(a24/r4), L20H13(a25/r63), L24H29(a27/r14), L24H05(a28/r7), L18H19(a29/r5), L20H14(a32/r45), L19H27(a34/r42), L24H23(a36/r1), L19H13(a38/r25), L28H23(a41/r91), L19H03(a42/r94), L17H04(a46/r51), L22H06(a49/r22), L23H31(a50/r90), L28H02(a51/r69), L13H11(a52/r38), L19H28(a54/r49), L19H14(a58/r74), L18H18(a59/r9), L24H13(a64/r33), L19H25(a69/r75), L31H30(a73/r96), L23H09(a78/r27), L29H15(a79/r64), L19H12(a82/r44), L29H11(a85/r77), L15H11(a87/r24), L23H11(a89/r17), L23H28(a91/r18), L20H29(a94/r99), L18H05(a99/r47), L24H14(a100/r19)
Qwen3-VL-8B (35 heads)	L21H10(a1/r55), L21H08(a2/r21), L21H11(a3/r60), L21H18(a6/r18), L20H15(a9/r24), L19H27(a10/r16), L21H17(a11/r2), L20H31(a13/r28), L22H04(a14/r86), L20H30(a17/r6), L18H19(a23/r7), L21H31(a24/r4), L24H29(a25/r10), L19H13(a28/r96), L20H28(a30/r5), L24H05(a32/r11), L17H04(a36/r45), L24H23(a37/r1), L18H18(a40/r17), L22H06(a45/r80), L28H02(a46/r26), L24H30(a49/r58), L23H30(a54/r68), L25H09(a62/r81), L24H13(a67/r15), L19H25(a68/r91), L18H05(a70/r49), L30H06(a71/r25), L26H31(a75/r29), L25H11(a84/r76), L23H28(a90/r62), L17H27(a91/r79), L23H12(a93/r97), L28H03(a97/r43), L24H14(a98/r8)
InternVL3.5-4B (42 heads)	L21H10(a1/r33), L21H11(a2/r23), L21H08(a4/r29), L21H18(a6/r9), L23H04(a10/r92), L24H29(a11/r8), L21H17(a12/r5), L20H31(a13/r27), L21H16(a14/r89), L20H16(a16/r91), L20H15(a17/r4), L21H23(a21/r59), L21H31(a22/r1), L28H23(a23/r63), L22H04(a25/r24), L20H30(a26/r6), L24H23(a29/r3), L13H11(a30/r100), L24H05(a32/r15), L20H18(a33/r74), L20H14(a36/r73), L20H28(a38/r2), L28H03(a39/r97), L31H30(a40/r85), L28H02(a41/r21), L19H27(a45/r50), L24H13(a46/r53), L18H19(a47/r11), L24H14(a49/r22), L17H04(a51/r66), L23H28(a52/r13), L19H13(a53/r36), L29H13(a55/r99), L22H06(a61/r38), L21H30(a69/r84), L19H14(a79/r88), L15H11(a84/r96), L19H28(a86/r60), L18H18(a87/r18), L19H12(a88/r82), L28H16(a99/r98), L26H18(a100/r20)
Qwen3-VL-4B (43 heads)	L21H10(a1/r50), L21H11(a3/r85), L21H08(a4/r28), L21H18(a6/r15), L22H04(a8/r55), L21H17(a9/r4), L20H16(a10/r62), L20H31(a11/r8), L20H15(a12/r12), L20H17(a13/r77), L19H24(a14/r69), L19H27(a17/r19), L20H18(a18/r58), L21H31(a19/r3), L21H23(a22/r86), L18H19(a23/r5), L19H13(a24/r48), L20H30(a25/r7), L24H29(a26/r26), L28H23(a29/r87), L19H14(a30/r36), L19H12(a35/r53), L21H30(a38/r82), L20H28(a40/r2), L30H06(a42/r40), L22H06(a45/r32), L18H18(a46/r6), L24H23(a51/r1), L18H05(a56/r33), L17H04(a58/r31), L24H13(a61/r59), L24H05(a62/r43), L19H28(a67/r63), L19H25(a72/r49), L28H02(a76/r72), L24H30(a77/r92), L13H11(a83/r25), L17H05(a84/r95), L24H14(a87/r16), L17H27(a91/r57), L23H12(a92/r64), L23H14(a98/r80), L26H31(a100/r71)

Table 12: Full-generation minus prefill-only intervention at $N = 10$, $K = 30$ (VHs, $n = 1{,}000$): change in accuracy (points) and fraction of samples whose parsed answer changed.

	overlap	pointer-only	writer-only	random (3 seeds)
InternVL3.5-4B	0.0 (0%)	0.0 (0%)	0.0 (0%)	0.0 (0%)
InternVL3.5-8B	0.0 (0%)	0.0 (0%)	0.0 (0%)	0.0 (0%)
Qwen3-VL-4B	0.0 (0%)	0.0 (0%)	0.0 (0%)	$+0.1$ / $+0.1$ / -0.1 (0.1–0.3%)
Qwen3-VL-8B	-0.4 (4.1%)	-0.8 (5.1%)	-0.9 (2.2%)	-0.2 / $+0.8$ / 0.0 (0–1.7%)

J Replication on Additional LVLMs

J.1 Discovery on three further models

We applied the pointer and writer criteria, unchanged, to three additional LVLMs: LLaVA-OneVision-7B [21] (Qwen2 backbone, SigLIP vision tower), Pixtral-12B [1] (Mistral-Nemo backbone, Pixtral vision encoder), and InternVL3.5-14B [40] (a larger member of the InternVL family). Probes use the configuration of Appendix D. Images are processed at a fixed size per model so that candidate images contribute equal numbers of visual tokens: 384-pixel single-tile inputs for LLaVA-OneVision, whose any-resolution default would otherwise give images different token counts, and 448 pixels for Pixtral. Table 13 reports the pointer top-1 head R@1 on ICIR-MCQ and the pointer-writer overlap among the

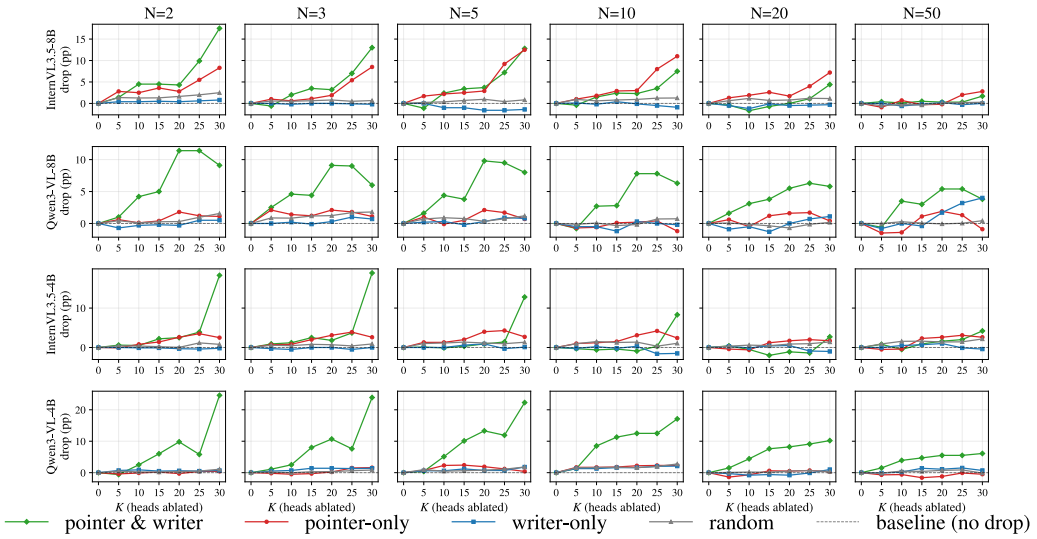


Figure 7: Set-partition dose-response per LVLM (top-100 partition, output-zero). Rows: LVLMs (top to bottom: InternVL3.5-8B, Qwen3-VL-8B, InternVL3.5-4B, Qwen3-VL-4B). Columns: haystack size $\in \{2, 3, 5, 10, 20, 50\}$. Random is the mean over 3 seeds.

top-100 heads, together with the overlap expected if the two rankings were unrelated ($100^2/H$ for H heads).

Table 13: Pointer-writer discovery on additional LVLMs (ICIR-MCQ, 6,000 samples). Overlap counts heads in both top-100 lists; chance is $100^2/H$.

Model	LLM / vision	H	Top-1 R@1	Overlap	Chance	Ratio
Main-text models (4)	InternVL3.5, Qwen3-VL	1,152	0.866–0.922	35–47	8.7	4.0–5.4×
LLaVA-OneVision-7B	Qwen2 / SigLIP	784	0.962	46	12.8	3.6×
Pixtral-12B	Mistral-Nemo / Pixtral	1,280	0.874	40	7.8	5.1×
InternVL3.5-14B	Qwen3-14B / InternViT	1,600	0.844	31	6.2	5.0×

All three models reproduce the discovery-side pattern: a single pointer head retrieves the target image with high R@1, and the pointer and writer top-100 lists overlap several times more than chance while remaining far from identical.

J.2 Set-partition ablation on LLaVA-OneVision-7B

We ran the complete set-partition protocol of §6 on LLaVA-OneVision-7B under the same conditions as the four main-text models: Visual Haystacks, $n = 1{,}000$ per haystack size, the output-zero intervention of §6, top-100 partition, $K \in \{5, 10, 15, 20, 25, 30\}$, and matched-budget random controls over three seeds. Table 14 reports accuracy relative to the random control at each (N, K). Haystack size 50 is omitted for this model: with about 1,485 visual tokens per image the N=50 prompt runs to about 74k tokens, more than twice the model's 32,768-token training context, and the unablated baseline itself falls below chance, so no intervention effect can be measured there. The four main-text models use at most 256 tokens per image (Appendix A) and stay within context.

The overlap set is the most damaging arm in all 30 cells, and its drop is significant (McNemar, $p < 0.05$) against all three random seeds in 24 of the 30 cells and against at least two seeds in all 30. The sparsity of the effect is visible at $K = 5$: removing five overlap heads already costs 0.07 to 0.14 accuracy at $N \leq 10$, while five pointer-only or writer-only heads stay within 0.02 of the random control. Writer-only ablations stay within ± 0.02 of random throughout.

Table 14: LLaVA-OneVision-7B set-partition ablation: accuracy minus the matched random control (mean of three seeds) for the pointer-and-writer overlap (first row per N), pointer-only, and writer-only sets. Baseline is the unablated accuracy at each haystack size.

N	Baseline	K=5	K=10	K=15	K=20	K=25	K=30
2	0.761	−0.123	−0.180	−0.198	−0.209	−0.211	−0.213
pointer-only		−0.011	−0.094	−0.098	−0.100	−0.106	−0.132
writer-only		−0.003	−0.005	−0.002	−0.007	−0.009	−0.017
3	0.741	−0.144	−0.193	−0.207	−0.213	−0.214	−0.210
pointer-only		−0.007	−0.083	−0.114	−0.113	−0.113	−0.136
writer-only		−0.004	−0.003	−0.007	−0.007	−0.015	−0.017
5	0.685	−0.100	−0.165	−0.167	−0.160	−0.149	−0.156
pointer-only		−0.005	−0.055	−0.080	−0.063	−0.054	−0.075
writer-only		+0.003	−0.002	−0.007	−0.005	+0.005	−0.005
10	0.637	−0.069	−0.105	−0.112	−0.094	−0.083	−0.088
pointer-only		−0.013	−0.028	−0.041	−0.017	−0.010	−0.017
writer-only		+0.013	+0.014	+0.006	+0.013	+0.015	+0.010
20	0.577	−0.041	−0.067	−0.066	−0.056	−0.047	−0.047
pointer-only		−0.016	−0.015	−0.003	+0.011	+0.008	−0.014
writer-only		+0.006	+0.008	+0.003	+0.003	+0.020	+0.018

K Visual vs. Text Retrieval Heads

We examined whether the image retrieval heads of Qwen3-VL-8B also serve text retrieval, using a text needle-in-a-haystack task: a numeric needle is embedded in a long interleaved image-text context (the images are filler and none is a retrieval target), and the model is asked to return the number. Baseline accuracy is 100% on the 291 samples, so head-level differences reflect attention behavior rather than task difficulty; we rank all 1,152 heads by their mean attention mass on the needle tokens at the position where the answer is generated. The two probes are not matched (see below), so we read the result as suggestive only. Table 15 shows selected heads and their ranks in image versus text retrieval.

Table 15: Image vs. text retrieval rankings (Qwen3-VL-8B).

Head	Image rank	Text rank	Character
L21H10	1	118	Image-specific
L21H08	2	67	Image-specific
L21H11	3	66	Image-specific
L23H04	20	10	Mixed
L29H09	137	1	Text-specific
L29H11	221	2	Text-specific

The top image retrieval head ranks 118th for text retrieval and the top text retrieval head ranks 137th for image retrieval; no head is in the top 10 for both. The pointer cluster sits in layers 19–23, whereas the two strongest text retrieval heads sit at layer 29 and the text top-10 spans layers 23–32.

Why the comparison is not precise. The two probes differ in candidate granularity (image spans of hundreds of patch tokens versus a few-token needle), in the scoring rule (argmax of attention mass over image spans versus attention to the needle position), in task format, and in query position (the cattok versus the answer position), any of which can move a head's rank for reasons unrelated to modality. We therefore take the result as evidence that the two retrieval functions are not served by the same top-ranked heads in this network, and leave a matched probe to future work.

L Extended Discussion

Cross-model convergence. Head-level roles that recur across independently trained models are more plausibly a property of the architecture and of multi-image training than a per-model artifact:

InternVL3.5 and Qwen3-VL are trained by different organizations on different data mixtures, yet the pointer cluster occupies the same mid-block layers of the same 36-block shape on all four models (Appendix C). This mid-block position matches where language-only transformers implement retrieval and factual lookup [15, 42], which suggests that the pointer-writer cluster is the multimodal analogue at the same structural position; we do not distinguish between these explanations.

Relation to text retrieval heads. Wu et al. [42] identified retrieval heads in text-only LLMs by an attention-pattern criterion. We expect input-side selection vs. output-side encoding to be a useful axis for revisiting text retrieval heads in long-context QA, but leave that to future work.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the four contributions (ICIR-MCQ probe, pointer heads, writer criterion, set-partition causal validation), each covered by a dedicated results section, and the claims are scoped to the LVLMs and benchmarks actually evaluated.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A "Limitations" paragraph in the Discussion covers the four-LVLM, two-family scope, the single-target Visual Haystacks evaluation, and the open directions (cross-image comparison, multi-target grounding, compositional reasoning, generation-time interventions).

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [N/A]

Justification: The paper is empirical and mechanistic; it does not state or prove formal theorems.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Probe construction, the pointer and writer scoring rules, ICIR-MCQ split sizes, the writer-probe architecture and sweep, and the ablation hook are described in the paper and appendices; all LVLMs and benchmarks used are public.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in

some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code (MIT) is released at `https://github.com/parkquasar/Image-Retrieval-Heads` and the ICIR-MCQ benchmark (CC BY 4.0) at `https://huggingface.co/datasets/jerryray/ICIR-MCQ`. The repository contains the head rankings used in the paper, the extraction, probe-training, ablation, and baseline scripts, and instructions that reproduce the main-text tables; all other assets are public.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Appendices A and D report image preprocessing, attention implementation, dtype, generation settings, fixed seeds, the writer-probe configuration and sweep, ablation budgets, and the three-seed random-head control.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The set-partition ablation reports a matched-budget random-head control averaged over three seeds, and Figure 3 shades ± 1 standard deviation across the four LVLMs; retrieval R@1 is computed on a fixed 1,000-sample set so all conditions see identical inputs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All main-text experiments run on NVIDIA L40 (48GB) GPUs; single-GPU inference suffices for all four LVLMs up to $N{=}50$. A Visual Haystacks pass at $N{=}10$ takes 1.5 to 4 s per sample, and the set-partition grid is a few hundred GPU-hours per LVLM (Appendix A).

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work is mechanistic-interpretability analysis of publicly released LVLM checkpoints on public benchmarks, with no human subjects and no release of new generative models. The interleaved text passages are reused from MM-NIAH and originate from the web-crawled OBELICS corpus.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: The paper is foundational interpretability research on already-released LVLMs; it adds no generative capability and opens no path to negative applications beyond those of the underlying models.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: No model is released; the ICIR-MCQ benchmark is assembled from public COCO and MM-NIAH data and poses no meaningful misuse risk.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All assets are cited and used under their posted licenses: InternVL3.5-{4B,8B,14B}, Qwen3-VL-{4B,8B}, LLaVA-OneVision-7B, Pixtral-12B, SigLIP, and Qwen3-VL-Embedding-8B (Apache-2.0); Visual Haystacks, MM-NIAH, and CLIP (MIT); MS-COCO, the source of the ICIR-MCQ images (annotations CC BY 4.0); the ICIR-MCQ passages are reused from MM-NIAH and originate from OBELICS (CC BY 4.0).

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [Yes]

Justification: ICIR-MCQ is released with a data card (format, sources, license, rebuild instructions) and checksums at `https://huggingface.co/datasets/jerryray/ICIR-MCQ`; the code is documented in the README of `https://github.com/parkquasar/Image-Retrieval-Heads`.

Guidelines:

- The answer [N/A] means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [N/A]

Justification: The paper does not involve crowdsourcing or human-subjects research; all annotations come from existing public datasets.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.

- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human subjects, so IRB review is not applicable.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LVLMs are the object of study, not a methodological aid. The seven models analyzed are named in §3 and Appendix J, with loading and attention-implementation details in Appendix A.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.