Semantic Head Specialization Guides Hybrid ViT Attention for Multimodal LLMs

Anonymous EMNLP submission

Abstract

Hybrid attention dominates frontier LLMs, yet Vision Transformers (ViTs) in multimodal LLMs lack a satisfactory hybrid design, with no consensus on why certain attention patterns work better. To fill this gap, we study ViT attention heads and find they differentiate into object- and background-specialist roles, a pattern most pronounced under full attention; we call this Semantic Head Specialization (SHS). We propose SHS-Index to quantify this specialization, show that it distinguishes full-attention from chunk-window ViTs, and find that it strongly tracks downstream benchmark performance. We then identify three structural factors that shape SHS—window interaction, token serialization, and local softmax allocation—and use them as design principles for hybrid attention. Guided by these factors, we design Ariadne Attention, a hybrid that matches full attention on 22 image and video tasks at $6.5\times$ less attention compute. Our findings establish head specialization as a measurable property for diagnosing and designing principled hybrid ViT attention at the multimodal-LLM scale.

1 Introduction

As Vision Transformers (ViTs) (Dosovitskiy et al., 2021) scale to higher resolutions, full self-attention becomes increasingly expensive, making efficient attention a central design problem. Hybrid attention is a natural candidate: frontier LLMs such as GPT-OSS (OpenAI, 2025) and Gemma 3 (Gemma Team, 2025) reduce cost by interleaving local and full-attention layers. In ViTs, however, existing hybrids have not yet matched the quality of full attention. Widely used efficient ViT designs such as Qwen2.5-VL (Bai et al., 2025b) rely on chunk-window locality, where tokens are grouped into non-overlapping windows, while recent high-quality VLM encoders such as Qwen3-VL (Bai et al., 2025a) return to full attention. This quality gap raises a concrete question: what does full attention preserve that current chunk-window ViT hybrids lose?

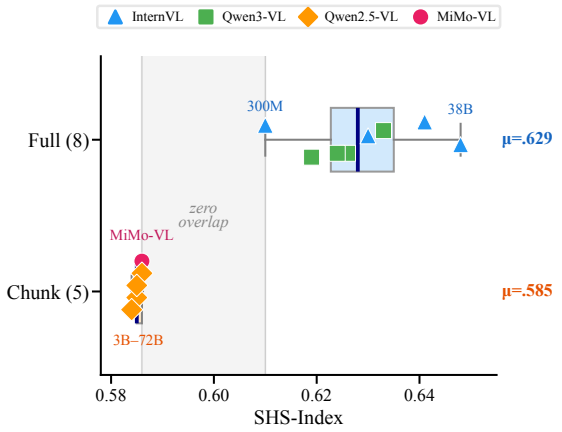

Figure 1: **SHS-Index separates ViT attention designs.** Across 13 open-source VLMs, SHS-Index cleanly separates full-attention and chunk-window ViTs. Within Qwen2.5-VL, changing the LLM size from 3B to 72B leaves SHS-Index nearly unchanged, indicating that the signal comes from the ViT attention design.

We study this question with a controlled full-vs.-chunk comparison, choosing chunk windows because they are the main deployed ViT hybrid baseline in current VLMs. We train two ViTs from scratch under identical conditions, ensuring that the attention operator is the only difference: one model uses full attention, while the other uses Qwen-style chunk windows. We then inspect per-head attention maps to see how the two operators organize visual information internally. The maps reveal a clear qualitative contrast: under full attention, heads differentiate into semantic roles, with some focusing on foreground objects and others on the background; under chunk windows, heads instead show more uniform grid-like patterns. We name this phenomenon **Semantic Head Specialization (SHS)**. To make this observation measur-

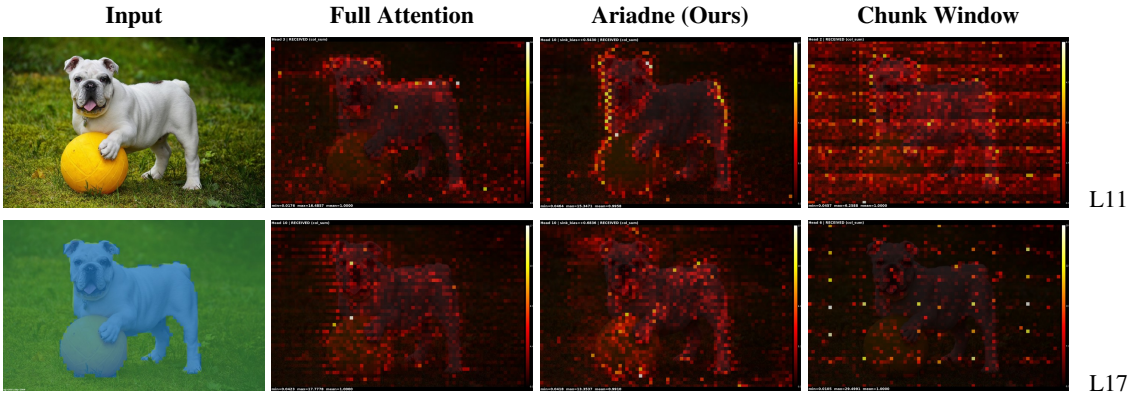

Figure 2: **Semantic Head Specialization (SHS) in ViT attention heads.** *First column:* input image (top) and a foreground (blue) / background (green) segmentation (bottom); each ViT token is labeled foreground or background by the dominant region within its image patch. *Columns 2–4:* received-attention heatmaps for a representative head at two ViT layers (L11, L17) across three attention designs. Under Full Attention, heads specialize: bright regions concentrate on the foreground object, revealing clear object/background differentiation. Our Ariadne attention preserves this specialization. Under Chunk Window attention (Qwen2.5-VL style), heads show grid-like window-boundary artifacts and fail to differentiate foreground from background.

able beyond individual heatmaps, we propose **SHS-Index**, an AUROC-based metric of how strongly each head separates foreground from background tokens. Across 13 open-source VLMs, SHS-Index separates full-attention and chunk-window ViTs with no overlap in score ranges, showing that the full-vs.-chunk specialization gap generalizes beyond our controlled training pair. Within Qwen2.5-VL, changing only the LLM backbone size from 3B to 72B changes SHS-Index by less than 0.002, ruling out LLM scale as the main driver of this signal (Figure 1). These results justify using SHS-Index as a ViT-side diagnostic for attention design.

We next use SHS-Index to diagnose why chunk-window attention weakens specialization and how a better hybrid can recover it. We focus on three structural factors—window interaction, token serialization, and local softmax allocation—and measure how each one changes SHS-Index. After this structural analysis, we evaluate the sufficiently trained diagnostic configurations on 22 downstream benchmarks. SHS-Index strongly tracks benchmark performance (Pearson r=0.79, p=0.021), making it a benchmark-aligned signal for attention design. Guided by the validated factors, we design **Ariadne Attention**[1], an SHS-informed hybrid combining sliding windows, per-head sink bias, and row-column alternating serialization. Ariadne matches full attention on 22 image and video tasks at $6.5\times$ less attention compute (Appendix B). These results establish head specialization as a measurable property linking attention design to ViT quality, opening a path toward principled hybrid ViT attention design.

Our contributions are: **(1)** We discover Semantic Head Specialization (SHS), the phenomenon that ViT heads differentiate into object- and background-specialist roles, and propose SHS-Index, a diagnostic metric that separates attention families across 13 open-source VLMs. **(2)** We identify three architectural factors that shape SHS—window interaction, token serialization, and local softmax allocation—through targeted structural comparisons. **(3)** We show that SHS-Index tracks benchmark performance on 22 downstream tasks and design Ariadne Attention, an SHS-guided hybrid that matches full attention on the full image+video suite at $6.5\times$ less attention compute.

2 Semantic Head Specialization in ViT Attention

To compare how full-attention and chunk-window ViTs differ in attention behavior and downstream quality, we first study their per-head attention patterns under a controlled full-vs.-chunk comparison (§2.1). We quantify this phenomenon with **SHS-Index** (§2.2) and analyze whether the specialization gap is an intrinsic property of ViT attention rather than merely a side effect of LLM scale.

[1]Named after Ariadne's thread in Greek mythology: as the thread guided Theseus through the labyrinth, sliding-window attention traces a continuous path through the token sequence.

2.1 Full-vs.-Chunk Comparison of Attention Patterns

We aim to understand how different ViT attention operators shape per-head attention patterns. Rather than starting from downstream scores, we inspect the attention maps induced by two specific operators: **full attention**, where every token can attend to every other token, and **chunk-window attention**, where non-overlapping chunks restrict each token to attend only within its own chunk. To compare these operators, we start from the attention computation itself. For head h over a sequence of N visual tokens, let $\mathcal{N}(i)$ be the tokens that query i attends to, and define $\mathcal{I}(a, b) = \{a, a+1, \ldots, b\}$ as the set of token indices from a to b. For full attention, $\mathcal{N}(i) = \mathcal{I}(1, N)$. For chunk attention with chunk size C (e.g., $\ll N$), $\mathcal{N}(i) = \mathcal{I}(b_i C{+}1, \min(N, (b_i{+}1)C))$, where $b_i = \lfloor (i-1)/C \rfloor$. The head-specific attention weight from query i to key j is:

$$a_{ij}^{(h)} = \frac{\exp(\mathbf{q}_i^\top \mathbf{k}_j / \sqrt{d})}{\sum_{j' \in \mathcal{N}(i)} \exp(\mathbf{q}_i^\top \mathbf{k}_{j'} / \sqrt{d}) + \exp(s_{\text{aux}}^{(h)})} \quad (1)$$

where $\mathbf{q}_i$ and $\mathbf{k}_j$ are the query and key vectors for tokens i and j, d is the per-head dimension, and $s_{\text{aux}}^{(h)}$ is a per-head learnable sink bias (Xiao et al., 2024; Darcet et al., 2024) (set to $-\infty$ when absent). We use *received attention* as each token's importance measure: the total attention mass that token collects from all queries that can see it. Received attention for token j is:

$$r_h(j) = \sum_{i:\, j \in \mathcal{N}(i)} a_{ij}^{(h)} \quad (2)$$

Intuitively, $r_h(j)$ is token j's head-h importance: total votes from queries that can see it.

A fair comparison must not conflate attention type with layer depth or training conditions. We therefore train a matched pair of 32-layer ViT encoders. One model uses full self-attention in all layers. The other follows Qwen2.5-VL-style: layers 7, 15, 23, and 31 use full attention, and the remaining 28 layers replace full attention with chunk-window attention (Bai et al., 2025b) (architecture details in Appendix A.1). We then visualize per-head received-attention maps on COCO val2017 images (Lin et al., 2014). The results reveal a striking difference between the two models. Under full attention (Figure 2, column 2), heads do not merely produce different-looking maps; they organize into consistent semantic roles. Some heads assign high received attention to foreground objects, with bright regions aligned with the dog and ball, while others assign high received attention to background regions. We call this phenomenon **Semantic Head Specialization (SHS)**: attention heads specialize by separating semantic regions of the image, here most clearly visible as distinct object- and background-oriented roles.

The chunk-window baseline shows much weaker SHS (Figure 2, column 4). Its heatmaps exhibit grid-like artifacts from window boundaries, and different heads look qualitatively similar rather than forming clear semantic roles. This contrast strongly suggests that the quality gap between full and chunk-window attention is reflected not only in downstream benchmark scores, but also in how attention heads organize internally and assign each token to a foreground- or background-oriented role.

SHS is also not a single-image artifact. The same head maintains a consistent foreground-or-background preference across all analysis images (per-image heatmaps in Appendix D; iteration-aggregated object dominance in Appendix E), indicating that SHS is an *intrinsic property* of the trained model rather than a stimulus-dependent visualization effect. The pattern further generalizes across diverse visual scenes—including single objects, multiple objects, complex backgrounds, and scenes with overlapping or cluttered foreground entities—and consistently maintains object/background head specialization (Appendix D).

2.2 Quantifying Head Specialization

The heatmaps above suggest that a specialized head should consistently prefer one semantic side of the image: either object tokens or background tokens. We therefore turn SHS into a foreground–background separation problem. For this quantitative analysis, we use COCO segmentation masks to assign each visual token to foreground or background according to the dominant region in its image patch (details in Appendix C). Given the received-attention score $r_h(j)$ (Eq. 2), a foreground-specialist head should give larger scores to foreground tokens than to background tokens, while a background-specialist head should consistently do the opposite.

For each head h and image, we compute the AUROC of received attention $r_h(j)$ for ranking

foreground tokens above background tokens:

$$\begin{aligned}\mathrm{AUROC}_h = \; & \Pr(r_h(j^+) > r_h(j^-)) \\ & + \tfrac{1}{2}\Pr(r_h(j^+) = r_h(j^-)),\end{aligned} \tag{3}$$

where j^+ and j^- denote foreground and background tokens. AUROC values near 1 or 0 indicate foreground- or background-oriented heads, while values near 0.5 indicate no separation. Since either direction (FG/BG) indicates specialization, we define the head-level score as

$$\mathrm{SHS}_h = \max(\mathrm{AUROC}_h,\, 1 - \mathrm{AUROC}_h). \tag{4}$$

which ranges from 0.5 (no separation) to 1 (strong specialization). The final **SHS-Index** averages this score over images, layers, and heads:

$$\text{SHS-Index} = \mathbb{E}_{\mathrm{img},\ell,h}[\mathrm{SHS}_h]. \tag{5}$$

On our controlled full-vs.-chunk pair, SHS-Index is .606 for Full Attention and .577 for Chunk Window, quantitatively matching the specialization gap observed in the heatmaps.

SHS-Index separates attention families in open-source VLMs. The controlled pair above shows that SHS-Index captures the specialization gap between full and chunk-window attention under matched training conditions. We next ask whether the same signal appears in existing VLMs rather than only in our controlled models. Across 13 open-source VLMs, SHS-Index cleanly separates the 8 full-attention ViTs (mean SHS = 0.629) from the 5 chunk-window ViTs (mean SHS = 0.585), with *zero overlap* (Figure 1; full data in Appendix E). Moreover, the identical Qwen2.5-VL ViT changes by only < 0.002 when paired with 3B–72B LLM backbones, showing that SHS-Index is determined by the ViT attention design rather than by the language-model scale. Thus, SHS-Index is not merely a diagnostic for our trained pair; it identifies a consistent full-vs.-chunk attention-family difference across existing VLMs.

3 Structural Sources of the SHS Gap

The evidence above points to a clear SHS gap: full attention specializes heads much more strongly than chunk-window attention. We now ask which structural choices create this gap, with the goal of turning them into design principles for a better hybrid attention. We study three factors in turn—window interaction, token serialization, and local softmax allocation—using controlled attention-only comparisons. As in §2.1, all comparisons keep the 32-layer ViT, data, training recipe, and full-attention anchor layers fixed; only the attention structure in the remaining 28 layers changes.

Factor 1: Window isolation restricts spatial scope. Chunk-window attention differs from full attention most directly in spatial reach. In full attention, every token can collect evidence from the whole image. In chunk-window attention, image tokens are partitioned into non-overlapping chunks, so a local layer can only mix information inside each chunk. If SHS requires heads to compare object and background evidence across regions, then the first question is whether relaxing this chunk-level isolation improves SHS. To answer this question, we keep Qwen2.5-VL's original 2D block layout, where image patches are grouped into local 2×2 blocks and then ordered as chunk windows (Figure 3b), and change only the attention operator from chunk-window attention to sliding-window attention (SWA) (Jiang et al., 2023). Here, SWA denotes a token-wise local attention pattern: with window (l, r), query i attends to $\mathcal{N}_{\mathrm{SWA}}(i) = \mathcal{I}(\max(1, i-l), \min(N, i+r))$, i.e., a window of preceding and following tokens rather than a fixed chunk. Unlike fixed chunks, this window overlaps across chunk boundaries, so tokens are no longer confined to their original chunk; across layers, information can propagate through successive windows to much farther spatial regions. Under this controlled change, SHS-Index increases from .577 (chunk-2D) to .588 (SWA-2D). Attention distance provides a direct physical check: using the attention distance of Dosovitskiy et al. (2021), $D = \mathbb{E}_{h,i}[\sum_j a_{ij}^{(h)} \|p_i - p_j\|_2]$, where p_i is the 2D image position of token i, the average span rises from $D = 10.69$ to $D = 11.80$. **This identifies window isolation as one source of the SHS gap: chunk windows restrict spatial scope, and a better hybrid ViT should propagate spatial context by allowing token interaction to flow freely across window boundaries.**

Factor 2: Serialization order determines specialization quality. SWA increases cross-window interaction, but it still operates over a 1D token sequence. Therefore, applying SWA to image tokens requires choosing how the 2D patch grid is serialized into that sequence. The SWA-2D variant above keeps Qwen2.5-VL's chunk-window ordering: it groups nearby patches into local blocks, then places blocks from the same chunk next to one an-

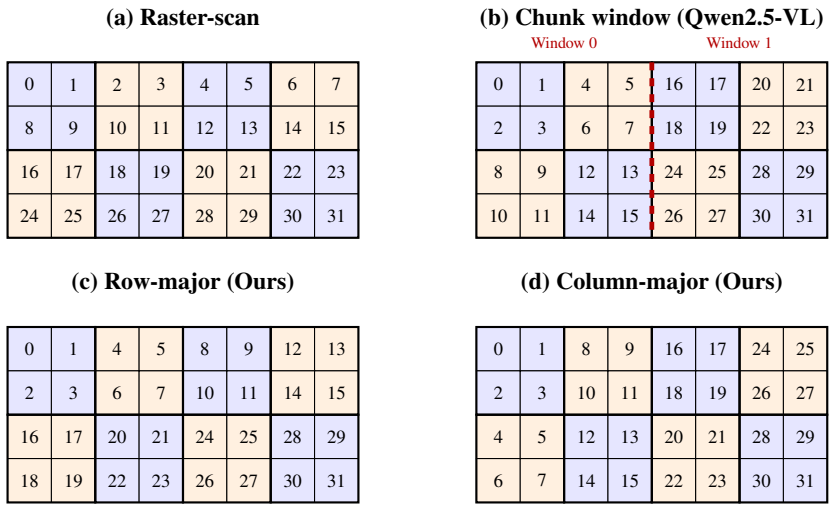

Figure 3: Token arrangements on a 4×8 patch grid (2×4 blocks). Numbers indicate position in the 1D sequence; bold borders delimit 2×2 blocks. **(a)** Raster-scan. **(b)** Qwen2.5-VL chunk window: blocks grouped into non-overlapping windows (dashed red line); token 15 and token 16 are spatially adjacent but in separate windows. **(c)** Row-major: all blocks serialized left-to-right, top-to-bottom for SWA. **(d)** Column-major: blocks serialized top-to-bottom, left-to-right. Ariadne alternates (c) and (d) across layers (§4.1).

other in the 1D sequence. This ordering is appropriate when attention is computed separately inside each chunk, but it becomes problematic when a sliding window moves along the sequence. As shown in Figure 4a, when SWA slides across a chunk boundary under this 2D layout, neighboring sequence positions can jump to distant image regions, so a single local attention window mixes spatially near and far tokens. We therefore test whether SWA should use a serialization that preserves spatial continuity. To test this, we keep the SWA operator fixed and introduce *SWA-1D*: a row-major 1D serialization that flattens image blocks left-to-right and top-to-bottom, making consecutive sequence positions correspond more closely to neighboring image regions (Figure 3b–c). Under this controlled change, SHS-Index increases from .588 (SWA-2D) to .600 (SWA-1D). **This identifies serialization order as a second source of the SHS gap: for sliding-window attention on the original 2D layout, token serialization should preserve spatially coherent neighborhoods.**

Factor 3: Softmax distributes attention to background. Even after increasing cross-window interaction and improving serialization, local attention still lets each query attend to only a subset of tokens. Within that subset, standard attention normalizes scores with softmax (Vaswani et al., 2017), forcing the attention weights over available keys to consume the query's full probability mass. When a

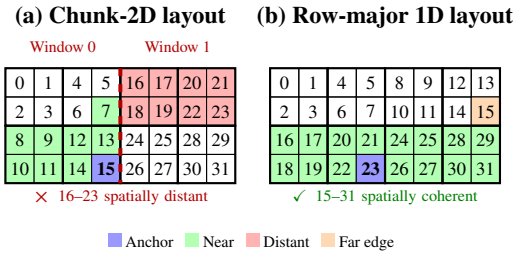

Figure 4: Why the 2D token ordering is incompatible with SWA. Numbers indicate position in the 1D sequence. **(a)** Chunk-2D: neighbors 16–23 map to the *top* of Window 1, spatially far from the anchor. **(b)** Row-major 1D: nearly all neighbors fall into surrounding rows, forming a coherent neighborhood.

local window contains many uninformative background tokens, this constraint may assign nontrivial attention to tokens that the head would otherwise prefer to ignore. We therefore test whether local attention benefits from an explicit absorber for surplus probability mass. Following recent efficient attention designs such as GPT-OSS (OpenAI, 2025), we add a per-head learnable sink bias $s_{\text{aux}}^{(h)}$: as defined in Eq. 1, it contributes an extra denominator term but has no associated image token, so it acts as a virtual sink that can absorb attention mass instead of forcing that mass onto foreground/background tokens. The SHS evidence is mixed but informative. Adding the sink slightly reduces SHS for SWA-1D (.600 → .598), which we report explicitly. However, the same intervention increases SHS in the

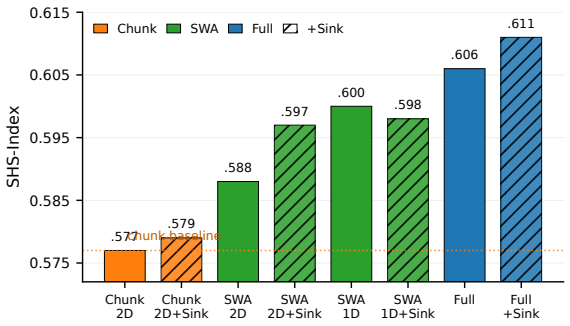

Figure 5: **SHS-Index across structural configurations.** The controlled variants differ only in the local attention structure while sharing the same backbone and training recipe. SHS-Index rises under the structure changes validated above, with Full+Sink shown as the highest-SHS diagnostic point and chunk-2D as the lowest.

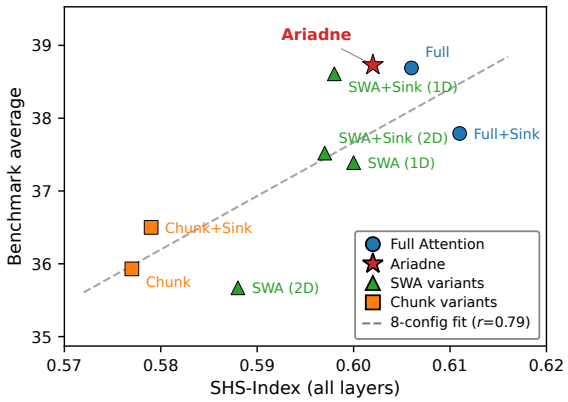

Figure 6: **SHS-Index tracks benchmark performance.** Across the controlled attention configurations, SHS-Index correlates strongly with the 22-task benchmark average (Pearson r=0.79, p=0.021; dashed line = 8-config linear fit). Adding the later design point (Ariadne) preserves the same trend (r=0.79, p=0.012).

other controlled settings: chunk-2D (.577 → .579), SWA-2D (.588 → .597), and full attention (.606 → .611). Allocation analysis further shows the sink's intended role: in SWA models, 94–97% of heads redirect more background than foreground attention to the sink (Appendix F). **This identifies local softmax allocation as a third source of the SHS gap: local-attention softmax can waste attention on uninformative tokens, and an explicit sink provides a useful absorber channel, though its SHS effect depends on the surrounding attention structure with SWA-1D as exception.**

Putting the controlled interventions together gives a compact view of how attention structure affects SHS-Index (Figure 5; full configurations in Table 3). Because the variants differ only in attention structure, the spread in SHS-Index directly reflects how these structural choices affect head specialization. This makes SHS-Index a natural bridge from mechanism to design: after showing that attention structure shapes the signal predicts architectural quality, we next check whether the same signal tracks downstream performance.

SHS-Index tracks benchmark performance. We evaluate the sufficiently trained diagnostic configurations on 22 downstream benchmarks, including 20 image tasks such as AI2D and OCRBench, and 2 video tasks including Video-MME. SHS-Index strongly tracks the 22-task benchmark average (Pearson r=0.79, p=0.021; Figure 6; full correlation table and robustness analysis in Appendix G). Thus, SHS-Index serves two roles: it separates full-vs.-chunk attention families, and it provides a benchmark-aligned signal for attention design. Full SHS-Index trajectories are reported in Appendix E (Table 11), and the complete benchmark trajectories used in the architectural comparisons are reported in Appendix B (Tables 5 and 6 for 20-image-task and 22-task averages, respectively). Collective-head evidence is in Appendix H.

4 SHS-Guided Design of Ariadne Attention

The previous section identifies structural factors that shape SHS and shows that SHS-Index tracks downstream benchmark performance. We now use these findings to design a hybrid attention pattern. Guided by the validated components—sliding-window attention, 1D serialization, and sink bias—we further increase token interaction across image dimensions by alternating row- and column-major serialization. The resulting hybrid, **Ariadne Attention**, aims to preserve the specialization and quality of full attention while retaining the efficiency of local attention without sacrificing either.

4.1 Ariadne Attention

We now instantiate the design suggested by the SHS analysis. Ariadne starts from the three components that improved or stabilized SHS in the controlled study: sliding-window attention (SWA) to relax chunk isolation, spatially coherent 1D serialization to make local windows meaningful on images, and a per-head sink bias to absorb surplus softmax mass. SWA replaces non-overlapping chunk windows with overlapping token-wise neigh-

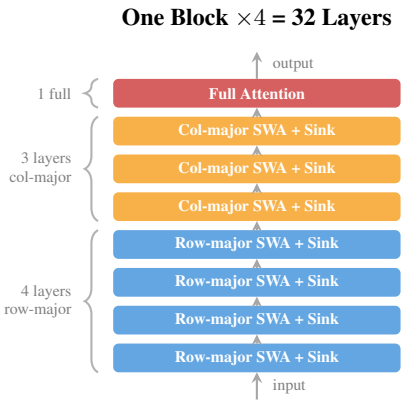

Figure 7: The **4-row-3-col** block architecture of Ariadne Attention. Each block contains 4 row-major SWA + sink layers, 3 column-major SWA + sink layers, and 1 full-attention layer; four repeats form the 32-layer ViT.

	Image tasks (20)	Video tasks (2)
Full Attention	**40.4**	22.1
Chunk-window Attention	37.2	23.3
Ariadne Attention	40.2	**24.3**

Table 1: Benchmark averages reported separately for image and video tasks. Ariadne nearly matches full attention on image tasks while outperforming it on the two video tasks.

borhoods, so adjacent windows can exchange information without returning to full attention. The sink bias uses FlashAttention-3's native `s_aux` term (Shah et al., 2024) as a zero-token absorber, serving the same role as attention sinks/register tokens (Xiao et al., 2024; Darcet et al., 2024) without adding extra visual tokens.

The remaining design question is how to serialize 2D image tokens for SWA. A single row-major order preserves horizontal continuity but propagates information less directly across columns. Ariadne therefore alternates row-major and column-major serialization across depth, so local windows connect both image dimensions over successive layers. Concretely, each 8-layer block contains 4 row-major SWA + sink layers, 3 column-major SWA + sink layers, and 1 full-attention anchor layer; repeating this block four times forms the 32-layer ViT (Figure 7). We set the SWA window to $w{=}64$ per side (128 tokens total context), matching GPT-OSS (OpenAI, 2025); an ablation over $w{\in}\{32, 64, 128\}$ selects the same value as best (SHS=.598, 22-task Bench=38.73) (Appendix A).

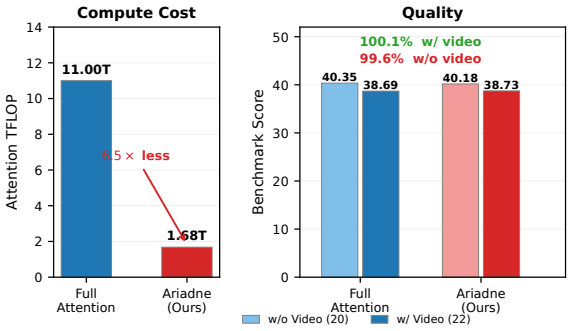

Figure 8: Attention compute vs. image-task quality at 896^2. Ariadne (★), an SHS-informed hybrid, nearly matches full attention at $6.5\times$ less compute cost.

4.2 Main Results

We first ask whether the SHS-guided design closes the quality gap to full attention. Ariadne scores 40.18 on the 20-task image benchmark, within 0.17 points of full attention, and reaches 24.25 on the two video tasks (Table 1; Figure 8). Relative to chunk-window attention, Ariadne improves the 20-image average by $+3.0$ and the 2-video average by $+1.0$, showing that the hybrid recovers nearly all of full attention's image quality while improving over the chunk-window baseline.

The largest gains occur on tasks where chunk-window isolation is most limiting: tasks that require finding evidence across distant regions or combining fine local details with broader context. On perception benchmarks, Ariadne improves over chunk-window attention by $+6.1$ on CV-Bench and $+7.2$ on VLMs-are-Blind. On high-resolution visual search, V* improves by $+11.0$, consistent with row/column serialization and sliding windows improving evidence propagation across the image. OCR and document-style tasks show the same pattern: DocVQA improves by $+9.8$, InfoVQA by $+7.7$, and OCRBench by $+8.9$, reflecting the need to connect local text evidence with page-level layout context across distant regions.

The remaining task-level differences are also informative. Grounding tasks change little (RefCOCO $+0.2$, RefCOCO+ ±0.0, RefCOCOg $+1.1$), suggesting that these benchmarks are less sensitive to the cross-region integration targeted by Ariadne and are dominated by local phrase–region matching. Chart and counting tasks regress slightly (ChartXiv-RQ -1.6, ChartQA -1.2, PixMo-Count -1.9), indicating that exact geometric reasoning and instance-level counting may require additional structure-preserving mechanisms beyond the cur-

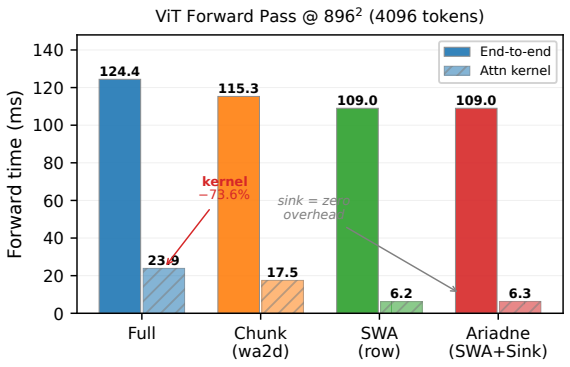

Figure 9: ViT forward-pass time at 896^2 (4,096 tokens). Solid bars: end-to-end block time; hatched bars: attention kernel only. Ariadne's attention kernel is $3.8\times$ faster than Full (6.3 ms vs. 23.9 ms), and matches SWA without sink, confirming zero overhead.

rent row/column serialization. Overall, Ariadne's strengths are most pronounced on tasks that require cross-region evidence integration, while highly geometry- or count-sensitive tasks point to a complementary direction for future hybrid designs. Full per-task results across all 22 tasks are reported in Appendix B (Table 7).

4.3 Efficiency

Finally, we test actual kernel timings, not just asymptotic attention complexity. Sink bias adds zero computational overhead: FlashAttention-3 implements `s_aux` as a single scalar addition to the softmax denominator (Shah et al., 2024). At 896^2, Ariadne's end-to-end ViT forward time is 109.0 ms vs. 124.4 ms for full attention (-12.4%); at 1344^2, the saving grows to -26.4% as the quadratic cost becomes dominant. SWA is also faster than chunk-window attention (109.0 ms vs. 115.3 ms at 896^2) because FlashAttention-3's contiguous SWA kernel is better optimized than the varlen interface (Figure 9). Thus Ariadne improves the measured quality–compute frontier; Appendix I reports comprehensive timing data across six resolutions, four attention operators, and input token counts.

5 Related Work

Attention head behavior. In NLP, prior work has identified specialized attention heads for syntactic and positional roles (Voita et al., 2019) and shown most heads are individually dispensable (Michel et al., 2019). In vision, Caron et al. (2021) show that *self-supervised* ViTs (DINO) develop CLS-attention maps that segment objects, but this analysis is limited to the last layer's CLS attention under self-supervised training and does not quantify per-head specialization across architectures. We extend these observations to *supervised* multimodal ViTs, formalize them with SHS-Index (continuous, threshold-free, direction-agnostic, computed across all heads and layers), and show that the degree of specialization predicts benchmark quality.

Hybrid attention in Vision Transformers. Efficient ViTs use local or hybrid attention, from Swin shifted windows (Liu et al., 2021) to Qwen2.5-VL chunk windows (Bai et al., 2025b), while recent VLM encoders such as InternVL 2.5 (Chen et al., 2024) and Qwen3-VL (Bai et al., 2025a) use full attention. Prior work also studies absorbers such as ViT registers (Darcet et al., 2024) and LLM attention sinks (Xiao et al., 2024). However, it does not isolate why full and hybrid ViTs differ or provide a hybrid ViT with comparable quality at lower compute. We address this with controlled attention-only comparisons and SHS, then instantiate the findings in Ariadne: a sliding-window hybrid that is within 0.2 points of full attention on image tasks with much lower attention compute, using sink bias (a per-head scalar) as a simpler, more elegant alternative to extra register tokens.

6 Conclusion

This paper studies why current hybrid ViT attention designs lag behind full attention in multimodal LLMs and how to close this gap. From a controlled full-vs.-chunk comparison, we discover Semantic Head Specialization (SHS): full-attention heads separate into object- and background-specialist roles, while chunk-window heads show much weaker separation. We quantify SHS with SHS-Index, show that it separates full-attention and chunk-window families across 13 open-source VLMs, and identify three structural factors that shape it: window interaction, token serialization, and local softmax allocation. Across controlled attention configurations, SHS-Index strongly tracks benchmark performance. Guided by these factors, we design Ariadne Attention, a hybrid that matches full attention on the 22-task image+video benchmark at $6.5\times$ less attention compute. Overall, SHS-Index provides a simple and interpretable signal for diagnosing existing ViT attention and guiding future hybrid architecture design.

Limitations

Scale of the controlled study. All nine architectures share the same training configuration (random-init ViT, 0.5B LLM, 10k steps). While this controlled design isolates the effect of attention operators, and our open-source VLM analysis confirms the SHS separation replicates at production scale (300M–72B), whether the quantitative SHS-to-benchmark correlation holds under different data mixtures, learning rates, or longer schedules remains to be verified in future work.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. 2025a. Qwen3-VL technical report. *ArXiv preprint*, abs/2511.21631.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025b. Qwen2.5-VL technical report. *ArXiv preprint*, abs/2502.13923.

Federico Barbero, Álvaro Arroyo, Xiangming Gu, Christos Perivolaropoulos, Michael Bronstein, Petar Veličković, and Razvan Pascanu. 2025. Why do LLMs attend to the first token? *ArXiv preprint*, abs/2504.02732.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. 2021. Emerging properties in self-supervised vision transformers. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9630–9640. IEEE.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, et al. 2024. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *ArXiv preprint*, abs/2412.05271.

Tri Dao. 2024. Flashattention-2: Faster attention with better parallelism and work partitioning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. 2024. Vision transformers need registers. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. 2024. Molmo and PixMo: Open weights and open data for state-of-the-art vision-language models. *ArXiv preprint*, abs/2409.17146.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Chaoyou Fu, Yuhan Dai, Yongdong Luo, Lei Li, Shuhuai Ren, Renrui Zhang, Zihan Wang, Chenyu Zhou, Yunhang Shen, Mengdan Zhang, Peixian Chen, Yanwei Li, Shaohui Lin, Sirui Zhao, Ke Li, Tong Xu, Xiawu Zheng, Enhong Chen, Caifeng Shan, et al. 2025. Video-MME: The first-ever comprehensive evaluation benchmark of multi-modal LLMs in video analysis. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 24108–24118.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A. Smith, Wei-Chiu Ma, and Ranjay Krishna. 2024. BLINK: Multimodal large language models can see but not perceive. In *Computer Vision – ECCV 2024*, pages 148–166. Springer.

Gemma Team. 2025. Gemma 3 technical report. *ArXiv preprint*, abs/2503.19786.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. 2025. When attention sink emerges in language models: An empirical view. In *The Thirteenth International Conference on Learning Representations (ICLR)*.

Tianyu Guo, Druv Pai, Yu Bai, Jiantao Jiao, Michael I. Jordan, and Song Mei. 2024. Active-dormant attention heads: Mechanistically demystifying extreme-token phenomena in LLMs. *ArXiv preprint*, abs/2410.13835.

Kairui Hu, Penghao Wu, Fanyi Pu, Wang Xiao, Yuanhan Zhang, Xiang Yue, Bo Li, and Ziwei Liu. 2025. Video-MMMU: Evaluating knowledge acquisition from multi-discipline professional videos. *ArXiv preprint*, abs/2501.13826.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego

de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *ArXiv preprint*, abs/2310.06825.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. 2024. MANTIS: Interleaved multi-image instruction tuning. *Transactions on Machine Learning Research (TMLR)*.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. 2016. A diagram is worth a dozen images. In *European Conference on Computer Vision (ECCV)*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755.

Yuan Liu, Haodong Duan, Yuanhan Zhang, Bo Li, Songyang Zhang, Wangbo Zhao, Yike Yuan, Jiaqi Wang, Conghui He, Ziwei Liu, Kai Chen, and Dahua Lin. 2024a. MMBench: Is your multi-modal model an all-around player? In *Computer Vision – ECCV 2024*, pages 216–233. Springer.

Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. 2024b. OCRBench: On the hidden mystery of OCR in large multimodal models. *Science China Information Sciences*, 67.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 9992–10002. IEEE.

Ahmed Masry, Xuan Long Do, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. 2022. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2263–2279, Dublin, Ireland. Association for Computational Linguistics.

Minesh Mathew, Viraj Bagal, Rubèn Pérez Tito, Dimosthenis Karatzas, Ernest Valveny, and C. V. Jawahar. 2022. InfographicVQA. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. 2021. DocVQA: A dataset for VQA on document images. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2199–2208.

Paul Michel, Omer Levy, and Graham Neubig. 2019. Are sixteen heads really better than one? In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 14014–14024.

OpenAI. 2025. gpt-oss-120b & gpt-oss-20b model card. *ArXiv preprint*, abs/2508.10925.

Pooyan Rahmanzadehgervi, Logan Bolton, Mohammad Reza Taesiri, and Anh Totti Nguyen. 2024. Vision language models are blind. In *Computer Vision – ACCV 2024*, pages 293–309. Springer.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. 2024. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, Austin Wang, Rob Fergus, Yann LeCun, and Saining Xie. 2024. Cambrian-1: A fully open, vision-centric exploration of multimodal LLMs. In *Advances in Neural Information Processing Systems*, volume 37, pages 87310–87356. Curran Associates, Inc.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. 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, Florence, Italy. Association for Computational Linguistics.

Zirui Wang, Mengzhou Xia, Luxi He, Howard Chen, Yitao Liu, Richard Zhu, Kaiqu Liang, Xindi Wu, Haotian Liu, Sadhika Malladi, Alexis Chevalier, Sanjeev Arora, and Danqi Chen. 2024. Charxiv: Charting gaps in realistic chart understanding in multimodal llms. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Penghao Wu and Saining Xie. 2024. V*: Guided visual search as a core mechanism in multimodal llms. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 13084–13094. IEEE.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. 2024. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Xiaomi LLM-Core Team. 2026. Mimo-v2-flash technical report. *ArXiv preprint*, abs/2601.02780.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. 2024. Qwen2 technical report. *ArXiv preprint*, abs/2407.10671.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. 2016. Modeling context in referring expressions. In *European Conference on Computer Vision (ECCV)*.

Xiang Yue, Yuansheng Ni, Tianyu Zheng, Kai Zhang, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, Cong Wei, Botao Yu, Ruibin Yuan, Renliang Sun, Ming Yin, Boyuan Zheng, Zhenzhu Yang, Yibo Liu, Wenhao Huang, et al. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert AGI. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2024, Seattle, WA, USA, June 16-22, 2024*, pages 9556–9567. IEEE.

Yi-Fan Zhang, Huanyu Zhang, Haochen Tian, Chaoyou Fu, Shuangqing Zhang, Junfei Wu, Feng Li, Kun Wang, Qingsong Wen, Zhang Zhang, Liang Wang, Rong Jin, and Tieniu Tan. 2025. MME-RealWorld: Could your multimodal LLM challenge high-resolution real-world scenarios that are difficult for humans? In *The Thirteenth International Conference on Learning Representations (ICLR)*.

Orr Zohar, Xiaohan Wang, Yann Dubois, Nikhil Mehta, Tong Xiao, Philippe Hansen-Estruch, Licheng Yu, Xiaofang Wang, Felix Juefei-Xu, Ning Zhang, Serena Yeung-Levy, and Xide Xia. 2024. Apollo: An exploration of video understanding in large multimodal models. *ArXiv preprint*, abs/2412.10360.

A Model Architecture and Configurations

A.1 ViT Architecture

All nine models use a 32-layer ViT encoder following the Qwen2.5-VL (Bai et al., 2025b) design. Of the 32 layers, 4 (at 0-indexed positions 7, 15, 23, 31) use full self-attention; the remaining 28 use the local attention operator under study. The ViT extracts non-overlapping 14×14 patches and groups every 2×2 spatially adjacent patches into an atomic block of 4 sub-tokens, the basic unit of both chunk-window grouping and 1D serialization. Architectural hyperparameters are listed in Table 2.

A.2 Nine Model Configurations

Table 3 lists all nine attention configurations explored in this work. They span three design axes: token layout (2D vs. 1D), attention type (chunk vs. sliding window vs. full), and sink bias (present or absent).

Parameter	Value
Layers	32
Full-attention layers	positions 7, 15, 23, 31
Local-attention layers	28
Hidden dimension	1,280
MLP dimension	3,456
Attention heads	16
Head dimension	80
Patch size	14×14
Block size	2×2 patches (4 sub-tokens)
Position encoding	2D RoPE
Initialization	Random

Table 2: ViT encoder hyperparameters, shared across all 9 configurations.

A.3 Attention Design Space

Chunk window attention. Blocks are grouped into 4×4-block windows (W=64 tokens per window), similar to the shifted-window mechanism of Swin Transformer (Liu et al., 2021). Self-attention is computed independently within each window. Tokens in different windows can only exchange information through the 4 full-attention layers.

Sliding window attention (SWA). Each token attends to a window of (w_L, w_R) tokens to its left and right in the 1D sequence, as popularized by Mistral (Jiang et al., 2023). We use w_L=w_R=64 (128 tokens total context). Unlike chunk windows, SWA windows overlap, enabling cross-window information flow. Implemented via FlashAttention-3 (Shah et al., 2024) varlen interface.

Sink bias. A single learnable scalar $s_{\text{aux}}^{(h)}$ per head is added to the softmax denominator, acting as a virtual sink token that absorbs surplus attention mass (Xiao et al., 2024; Darcet et al., 2024). Natively supported by FlashAttention-3's `s_aux` parameter at zero additional cost.

Token serialization. For 1D-layout models, blocks are serialized into a flat sequence. Row-major: left-to-right, top-to-bottom. Column-major: top-to-bottom, left-to-right. The Ariadne configuration alternates 4 row-major and 3 column-major layers within each block of 7 local layers (4-row-3-col pattern). See Figure 3 for an illustration.

A.4 Window Size Ablation

Table 4 compares three window sizes on the Ariadne configuration.

Short Name	Layout	Attention	Sink	Window Size	SHS-Index	Performance (Avg. Acc)
wa2d	2D	Chunk	No	64 tok	.577	37.19
wa2d_sink	2D	Chunk	Yes	64 tok	.579	37.72
swa2d	2D	SWA	No	128 tok	.588	36.73
swa2d_sink	2D	SWA	Yes	128 tok	.597	38.87
swa_row	1D row	SWA	No	128 tok	.600	38.67
swa_sink_row	1D row	SWA	Yes	128 tok	.598	40.09
swa_sink_4r3c	1D row+col	SWA	Yes	128 tok	.602	40.18
full	N/A	Full	No	Global	.606	40.35
full_sink	N/A	Full	Yes	Global	.611	39.52

Table 3: Complete model configurations. All models share the same ViT architecture (32 layers, 16 heads, d_h=80) and training setup (Table 8); only the local attention operator differs. Bench = macro-average over 20 image tasks.

Window	Context	20-image	22-task
w=32	64 tok	37.87	36.73
w=64	128 tok	**40.18**	**38.73**
w=128	256 tok	38.46	37.09

Table 4: Window size ablation (Ariadne, swa_sink_4r3c) at the primary checkpoint. w=64 is optimal under both 20-image and 22-task averages, matching the window size used by GPT-OSS (OpenAI, 2025) and MiMo-V2-Flash (Xiaomi LLM-Core Team, 2026).

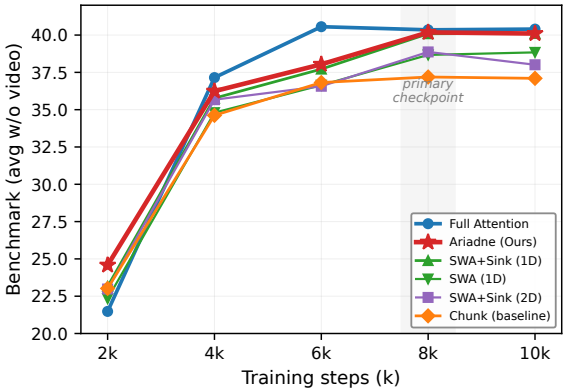

Figure 10: Benchmark trajectory across training for six key models. Architectural rankings are stable across late checkpoints, while SWA+Sink (2D) begins to overfit in the late stage.

B Complete Benchmark Results

This appendix reports full benchmark numbers across all iterations (Tables 5 and 6, for the 20-image-task and 22-task averages, respectively) and per-task breakdowns for the three key models at the primary checkpoint (Table 7).

B.1 Benchmark Across Iterations

Table 5 reports the macro-average benchmark (20 image tasks, excluding video) at each checkpoint; Table 6 reports the 22-task counterpart used in the SHS-quality correlation (§3). The ranking remains stable across late checkpoints under both metrics, motivating our choice of primary checkpoint; see Figure 10 for a visualization.

B.2 Key Models Per-Task Results

Table 7 breaks down the benchmark into all 22 individual tasks for the three key models (Full, Ariadne, Chunk). Ariadne recovers most of the gap on vision-centric tasks while grounding tasks show near-parity.

C Experimental Setup Details

C.1 Training Configuration

Table 8 summarizes the training setup shared across all nine models; only the attention operator differs.

Training data. We train on **Cambrian-10M** (Tong et al., 2024), a publicly available multimodal instruction-tuning dataset of approximately 10M examples spanning OCR, general vision-language understanding, math, code, science, and multi-turn conversation. We use the released category-balanced mixture without modification.

Design rationale. We deliberately choose a small LLM backend. As noted by Tong et al. (2024), a strong language model can act as a shortcut that compensates for deficiencies in visual representations, masking real differences between vision encoder designs. A weaker LLM forces the ViT to develop genuinely better representations, making architectural differences more visible. This rationale is further supported by the scaling consistency findings of Zohar et al. (2024).

C.2 Benchmark Suite

We evaluate on 22 tasks spanning 5 capability dimensions (Table 9). The primary metric is the macro-average over the 20 image tasks ("Avg w/o

Iter	full_s†	full	4r3c	sink_r	swa_r	s2d_s	swa2d	w_s	wa2d
2k	22.25	21.48	24.58	23.15	22.34	22.98	22.67	23.22	23.02
4k	34.44	37.15	36.24	35.77	34.80	35.67	33.30	35.48	34.62
6k	37.70	40.56	38.04	37.72	36.64	36.57	37.64	38.19	36.82
8k	39.52	40.35	40.18	40.09	38.67	38.87	36.73	37.72	37.19
10k	39.94	40.40	40.10	40.11	38.84	38.01	35.21	38.27	37.10

Table 5: Benchmark (avg w/o video, 20 tasks) across training iterations. †full_sink 2k–6k based on general tasks only (10 tasks); 8k/10k are complete (20 tasks).

Iter	full_s†	full	4r3c	sink_r	swa_r	s2d_s	swa2d	w_s	wa2d
2k	22.25	21.42	24.41	23.04	22.35	22.85	22.58	23.13	22.92
4k	33.25	35.77	35.16	34.67	33.73	34.67	32.41	34.49	33.61
6k	36.26	38.93	36.85	36.53	35.55	35.45	36.54	37.01	35.62
8k	37.79	38.69	38.73	38.61	37.39	37.52	35.67	36.50	35.93
10k	38.20	38.73	38.66	38.63	37.61	36.84	34.27	37.02	35.91

Table 6: Benchmark (avg w/ video, 22 tasks) across training iterations. Includes Video-MME and Video-MMMU. This is the metric used for the SHS-quality Pearson correlation in §3. †full_sink 2k–6k based on general tasks only (10 tasks); 8k/10k are complete (22 tasks).

video"); we additionally report the 22-task average including video ("Avg w/ video").

C.3 Analysis Data

Per-head attention analysis uses images sampled from COCO val2017, paired with a foreground/background segmentation that distinguishes salient objects from the background. Each ViT token is labeled foreground or background by the dominant region within its image patch; patches crossing the boundary (edge patches) are excluded from analysis.

D Additional Attention Heatmaps

This appendix provides additional received-attention heatmaps beyond the dog image in Figure 2. Figure 11 shows per-head box plots for the dog image, and Figure 12 presents a compact 2×2 grid of four additional images (bird, two_cats, coco_537964, coco_043816), each comparing Full Attention, Ariadne, and Chunk Window at layers L11 and L17.

E SHS Supplementary Data

E.1 SHS-Index on Open-Source VLMs

Table 10 reports SHS-Index for 13 production-scale open-source VLMs spanning 4 model families, diverse attention designs, and model sizes from 300M standalone ViTs to 72B-parameter VLMs. SHS is computed on the same analysis images and foreground/background masks used for our controlled experiments.

E.2 Iteration-Level SHS Metrics

This subsection provides iteration-level SHS metrics for all nine controlled architectures. Figures 13 and 14 visualize the trends from Tables 11, 12, and 13. SHS-Index declines gradually for all models as training progresses (specialization sharpens in early training then plateaus), but the relative ordering is stable from 4k onward. Obj-dominant percentage shows a similar downward trend, reflecting that heads increasingly split between object and background roles.

E.3 SHS-Index Across Iterations (All Layers)

Table 11 reports the all-layer SHS-Index at each checkpoint. The high/low group separation identified in Appendix G.2 is visible from 4k onward; see Figure 13 for a visualization.

E.4 SHS-Index Across Iterations (Core L9–21)

Table 12 restricts the computation to core layers L9–21, where head specialization is strongest. The gap between high and low groups widens compared to the all-layer average.

E.5 Obj-Dominant Percentage Across Iterations

Table 13 tracks the fraction of heads classified as object-dominant (majority vote across analysis images, core L9–21). Full attention models maintain the highest obj% throughout; SWA 1D without sink develops the lowest, consistent with the sink-

Cat.	Task	Full	Chunk	Ariadne	Δ
General	AI2D	37.2	35.3	**37.5**	+2.3
	BLINK	37.0	**39.4**	38.3	−1.1
	CV-Bench	47.7	43.4	**49.5**	+6.1
	MMMU	29.1	28.7	**30.2**	+1.5
	MMBench-EN	25.0	33.8	**34.1**	+0.4
	MMBench-CN	30.8	27.2	**29.9**	+2.7
	Mantis	16.1	15.2	**22.1**	+6.9
	MME-RW-CN	26.8	27.2	**27.5**	+0.3
	V*	48.7	28.8	**39.8**	+11.0
	VLMs-are-Blind	31.5	28.2	**35.4**	+7.2
OCR	ChartXiv-RQ	19.1	**21.0**	19.4	−1.6
	ChartXiv-DQ	52.1	**50.2**	49.7	−0.5
	DocVQA	53.5	39.8	**49.5**	+9.8
	InfoVQA	32.4	23.6	**31.3**	+7.7
	ChartQA	19.4	**18.8**	17.7	−1.2
	OCRBench	68.0	54.7	**63.6**	+8.9
Ground.	RefCOCO	76.2	75.0	**75.2**	+0.2
	RefCOCO+	63.4	**62.8**	62.8	±0.0
	RefCOCOg	67.7	68.5	**69.5**	+1.1
Count	PixMo-Count	25.1	**22.3**	20.4	−1.9
Video	Video-MME	27.5	30.9	**32.3**	+1.4
	Video-MMMU	16.7	15.8	**16.2**	+0.4
Avg		**38.7**	35.9	**38.7**	+2.8

Table 7: Per-task benchmark results for the three key models. Δ = Ariadne − Chunk. Bold = best between Chunk and Ariadne. Ariadne recovers most of the full-hybrid gap on vision-centric tasks (V* +11.0, DocVQA +9.8, OCRBench +8.9); grounding tasks, which rely on local spatial reasoning within the sliding window, show near-parity.

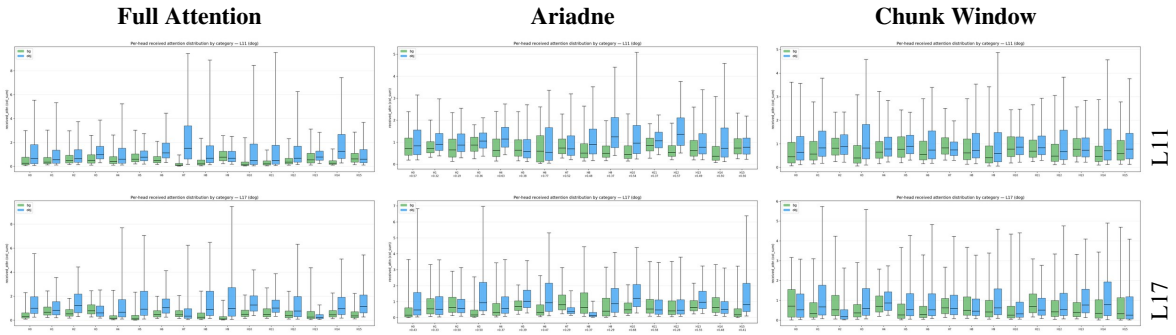

Figure 11: **Per-head received-attention box plots** corresponding to Figure 2. Each subplot shows the distribution for all 16 heads, split by foreground (blue) and background (green). Full Attention and Ariadne show clear obj/bg separation; Chunk Window distributions are uniform, indicating suppressed specialization.

as-background-suppressor analysis in Appendix F. See Figure 14 for a visualization.

E.6 Mean rcv_auroc_obj Across Iterations

Table 14 reports the mean per-head AUROC for the object class (before the direction-agnostic max operation used in SHS-Index). This raw metric confirms that full attention models develop the strongest foreground discrimination, while chunk-window models remain the weakest at every checkpoint.

F Mechanism Analysis Details

F.1 SHS vs. Entropy vs. Distance

Definitions. *Attention entropy*: $H = -\sum_j a_j \log a_j$, measuring how peaked or diffuse a head's attention distribution is. *Attention distance*: the attention-weighted mean spatial distance between each query and its attended keys, measuring how far each token "looks." Table 15 reports all three metrics alongside benchmark for all 9 models at the primary checkpoint.

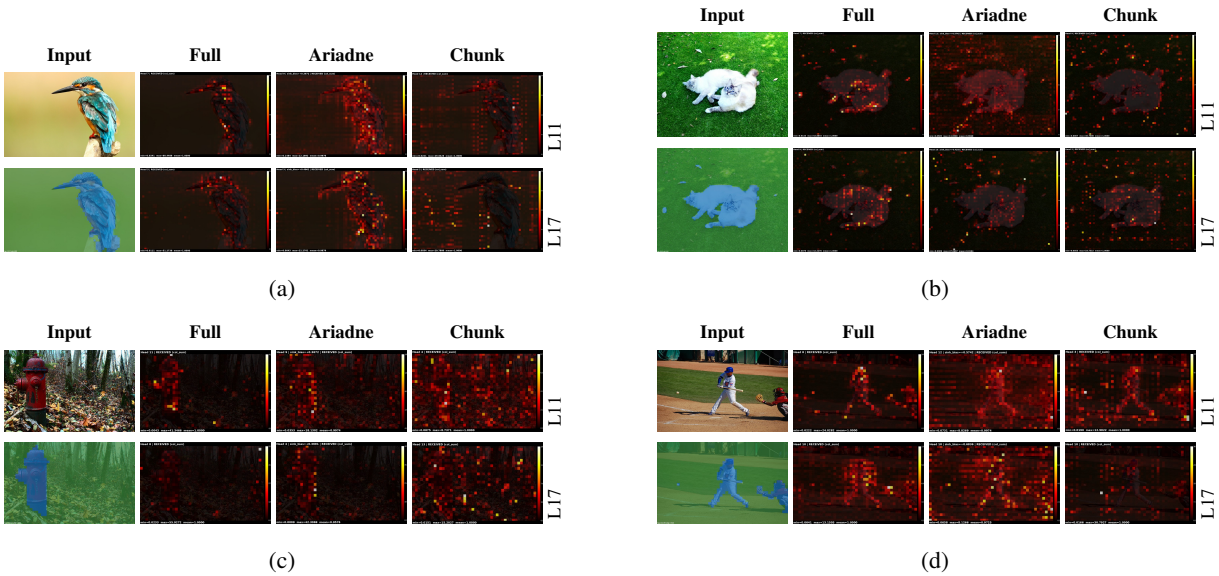

Figure 12: **Additional received-attention heatmaps** across four images, laid out as Figure 2. Each block: first column = input image (top) and foreground mask (bottom); columns 2–4 = representative head heatmaps at L11 and L17 (layer labels on the right). Full Attention and Ariadne consistently show clear object/background differentiation; Chunk Window shows suppressed specialization across all images.

Parameter	Value
ViT encoder	Random-init, 32 layers
LLM backbone	Qwen2-0.5B (Yang et al., 2024)
Projector	Random-init
ViT / LLM lr	5×10^{-4} / 1×10^{-5}
LR schedule	Constant w/ 1k warmup
Global batch size	2,048
Max seq length	32,768 tokens
Seq packing	Yes (Chen et al., 2024)
Steps	10,000
Precision	bfloat16
Compute budget	∼100k GPU-hours (NVIDIA Hopper)
Framework	Megatron-based (in-house)
Inference	SGLang 0.4 (modified)

Table 8: Training configuration shared across all nine models.

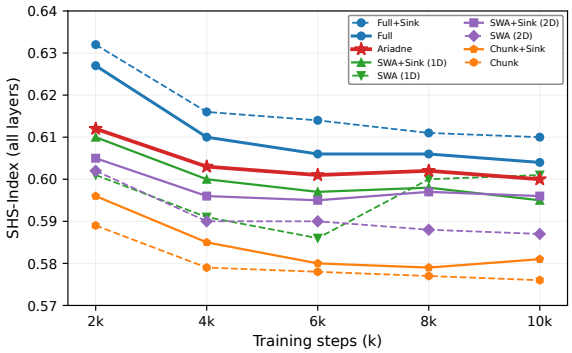

Figure 13: SHS-Index (all layers) across training iterations for all 9 models. The high/low SHS separation is visible from 4k onward and remains stable.

Cat.	Tasks	N
General	AI2D, BLINK, CV-Bench, MMMU, MMBench-EN/CN, Mantis, MME-RealWorld-CN, V*, VLMs-are-Blind	10
OCR	ChartXiv-RQ/DQ, DocVQA, InfoVQA, ChartQA, OCRBench	6
Ground.	RefCOCO, RefCOCO+, RefCOCOg	3
Count	PixMo-Count	1
Video	Video-MME, Video-MMMU	2

Table 9: Benchmark suite: 20 image + 2 video tasks. Video tasks excluded from primary "Avg w/o video"; included in "Avg w/ video." Citations: General (Kembhavi et al., 2016; Fu et al., 2024; Tong et al., 2024; Yue et al., 2024; Liu et al., 2024a; Jiang et al., 2024; Zhang et al., 2025; Wu and Xie, 2024; Rahmanzadehgervi et al., 2024); OCR (Wang et al., 2024; Mathew et al., 2021, 2022; Masry et al., 2022; Liu et al., 2024b); Grounding (Yu et al., 2016); Count (Deitke et al., 2024); Video (Fu et al., 2025; Hu et al., 2025).

Cross-model Spearman correlation with benchmark (N=8, excl. full_sink). Table 16 compares the three metrics' correlation with benchmark performance.

Pairwise deltas. Table 17 isolates each architectural change to show which metrics track benchmark improvements.

#	Model	Family	Attention	ViT Spec	SHS-all	SHS-core
1	InternVL3.5-38B	InternVL	Full	6B (45L, 3200d, 25h)	**.648**	.627
2	InternVL3.5-8B	InternVL	Full	300M (24L, 1024d, 16h)	.641	**.651**
3	Qwen3-VL-32B	Qwen3-VL	Full	675M (27L, 1152d, 16h)	.633	.644
4	InternVL3-8B	InternVL	Full	300M (24L, 1024d, 16h)	.630	.649
5	Qwen3-VL-8B	Qwen3-VL	Full	675M (27L, 1152d, 16h)	.626	.639
6	Qwen3-VL-4B	Qwen3-VL	Full	400M (24L, 1024d, 16h)	.624	.639
7	Qwen3-VL-2B	Qwen3-VL	Full	400M (24L, 1024d, 16h)	.619	.636
8	InternViT-300M	InternVL	Full	300M (24L, 1024d, 16h)	.610	.625
9	MiMo-VL-7B	MiMo	Chunk+Full	675M (32L, 1280d, 16h)	.586	.587
10	Qwen2.5-VL-3B	Qwen2.5-VL	Chunk+Full	675M (32L, 1280d, 16h)	.586	.586
11	Qwen2.5-VL-7B	Qwen2.5-VL	Chunk+Full	675M (32L, 1280d, 16h)	.585	.584
12	Qwen2.5-VL-72B	Qwen2.5-VL	Chunk+Full	675M (32L, 1280d, 16h)	.585	.585
13	Qwen2.5-VL-32B	Qwen2.5-VL	Chunk+Full	675M (32L, 1280d, 16h)	.584	.584

Table 10: SHS-Index on 13 open-source VLMs. Full-attention models (mean SHS $= 0.629$) and chunk-window models (mean SHS $= 0.585$) show zero overlap, consistent with our controlled experiments. Qwen2.5-VL uses the identical ViT across 3B–72B LLM sizes (SHS varies by < 0.002), confirming SHS is a ViT-intrinsic property.

Iter	full_s	full	4r3c	sink_r	swa_r	s2d_s	swa2d	w_s	wa2d
2k	.632	.627	.612	.610	.601	.605	.602	.596	.589
4k	.616	.610	.603	.600	.591	.596	.590	.585	.579
6k	.614	.606	.601	.597	.586	.595	.590	.580	.578
8k	.611	.606	.602	.598	.600	.597	.588	.579	.577
10k	.610	.604	.600	.595	.601	.596	.587	.581	.576

Table 11: SHS-Index (mean over all 32 layers) across training iterations for all 9 controlled architectures. This is the primary aggregation used throughout the paper (Table 15, Pearson correlation in §3, and Figure 6). Column headers abbreviated: full_s = full_sink, sink_r = swa_sink_row, swa_r = swa_row, s2d_s = swa2d_sink, w_s = wa2d_sink.

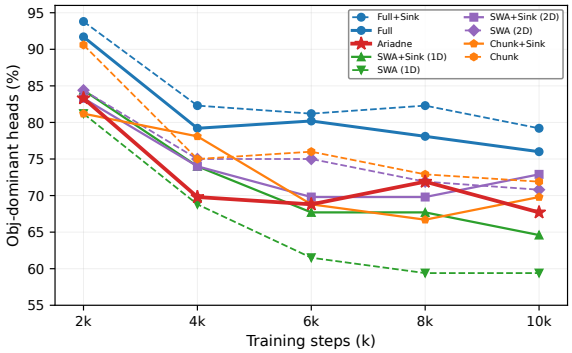

Figure 14: Obj-dominant head percentage (core layers L9–21) across iterations. SWA 1D models (without sink) develop the lowest obj%, while full attention models maintain the highest, consistent with the sink-as-background-suppressor analysis in Appendix F.

F.2 Sink 2×2 Analysis

We analyze how sink bias (related to the attention sink phenomenon (Xiao et al., 2024) and register tokens (Darcet et al., 2024)) differentially absorbs attention from background vs. object tokens across 5 sink-equipped models (Table 18). The sink steal ratio is defined as: $\text{steal}(\text{cat}) = \text{absorbed_sum}(\text{cat})/(\text{rcv_sum}(\text{cat}) + \text{absorbed_sum}(\text{cat}))$, computed over core layers L9–L21 at the primary checkpoint.

F.3 2D Incompatibility Hypothesis

Under 2D block layout, the sliding window's left and right neighbors in 1D sequence map to spatially disjoint regions of the image at every window boundary. This forces each head to attend to a mixture of spatially near and spatially distant tokens within the same window, diluting the spatial coherence that enables meaningful object/background discrimination.

Evidence (Table 17): the 2D→1D transition yields the largest single-step benchmark improvement (+1.94 without sink, +1.22 with sink), yet Entropy and Distance are essentially unchanged (ΔEntropy $= +0.10$, ΔDistance $= -0.09$). Only SHS increases (+0.013), suggesting that the improvement comes from spatial coherence enabling more meaningful head specialization, not from changes in attention sharpness or receptive field size.

Iter	full_s	full	4r3c	sink_r	swa_r	s2d_s	swa2d	w_s	wa2d
2k	.651	.644	.612	.615	.604	.605	.606	.597	.586
4k	.636	.629	.602	.603	.593	.598	.596	.583	.577
6k	.635	.624	.604	.600	.588	.597	.599	.578	.576
8k	.632	.623	.604	.599	.603	.601	.592	.580	.576
10k	.630	.622	.605	.594	.601	.598	.593	.580	.574

Table 12: SHS-Index (core layers L9–21) across training iterations.

Iter	full_s	full	4r3c	sink_r	swa_r	s2d_s	swa2d	w_s	wa2d
2k	93.8	91.7	83.3	84.4	81.2	83.3	84.4	81.2	90.6
4k	82.3	79.2	69.8	74.0	68.8	74.0	75.0	78.1	75.0
6k	81.2	80.2	68.8	67.7	61.5	69.8	75.0	68.8	76.0
8k	82.3	78.1	71.9	67.7	59.4	69.8	71.9	66.7	72.9
10k	79.2	76.0	67.7	64.6	59.4	72.9	70.8	69.8	71.9

Table 13: Percentage of obj-dominant heads (core L9–21, majority vote across analysis images).

G SHS-Benchmark Correlation Details

G.1 Statistical Methods

Our primary analysis is a Pearson correlation between SHS-Index and the 22-task benchmark average across the eight diagnostic configurations (§3). We report the correlation both *before* introducing Ariadne (the eight diagnostic configurations alone) and *after* including it (all nine controlled architectures), at the primary and final checkpoints, and we also report the 20-image-task counterpart for completeness. As a robustness check, we additionally apply K-means clustering (k=2) on SHS-Index followed by a Mann-Whitney U test on the resulting groups (§G.2); the two analyses agree on the qualitative conclusion.

G.2 Robustness: K-means Clustering and Mann-Whitney U Test

As a non-parametric robustness check, K-means (k=2) on SHS-Index automatically separates the eight diagnostic configurations into a high-SHS group (5 models: full, full_sink, swa_sink_row, swa_row, swa2d_sink; SHS ≥ 0.597) and a low-SHS group (3 models: swa2d, wa2d_sink, wa2d; SHS ≤ 0.588). Adding Ariadne keeps the low-SHS group unchanged and expands the high-SHS group to 6 models. In both settings, the groups show *zero benchmark overlap* at the primary checkpoint: the lowest-scoring high-SHS model (38.67) exceeds the highest-scoring low-SHS model (37.72). A Mann-Whitney U test on the 20-image-task average confirms significance (Table 19).

H SHS Collective Property Experiments

H.1 Head Elimination

We test whether individual high-SHS or low-SHS heads are more important by replacing specified heads' V outputs with their calibration-set mean (mean ablation), following the head importance analysis methodology of Michel et al. (2019). For each model, we select the 8 heads with highest and 8 with lowest core-layer AUROC, eliminate them separately, and compare the benchmark drop.

High vs. low SHS heads and all-16 elimination. Table 20 combines both experiments on the full model. The upper panel shows that eliminating 8 high-SHS heads causes the same benchmark drop as eliminating 8 low-SHS heads (avg Δ=-0.18); individual heads are interchangeable regardless of specialization direction. The lower panel shows that eliminating all 16 core-layer heads simultaneously causes dramatic drops on vision-dependent tasks, confirming that SHS is a collective property.

Cross-model consistency. Table 21 extends the high-vs-low analysis to three architectures.

H.2 QK Weight Norm vs. SHS

We compute the per-head Frobenius norm of Q and K weight matrices and correlate with per-head SHS (AUROC). Table 22 shows Spearman ρ between QK norm product and per-head AUROC for each model.

H.3 QK Equalization Experiment

We artificially equalize all heads' QK norms to the per-layer maximum, then recompute SHS. If QK norm causes SHS, equalization should increase

Iter	full_s	full	4r3c	sink_r	swa_r	s2d_s	swa2d	w_s	wa2d
2k	.618	.608	.566	.576	.552	.567	.570	.566	.557
4k	.596	.584	.546	.554	.533	.547	.548	.545	.543
6k	.589	.574	.545	.540	.524	.539	.546	.529	.537
8k	.587	.574	.545	.543	.522	.540	.539	.531	.534
10k	.581	.566	.543	.532	.519	.537	.537	.530	.531

Table 14: Mean rcv_auroc_obj (core L9–21) across training iterations.

Model	Layout	Attn	Sink	SHS	Entropy	Distance	Bench
full	N/A	Full	No	.606	4.584	14.922	40.35
full_sink	N/A	Full	Yes	.611	4.350	14.824	39.52
4r3c	1D rc	SWA	Yes	.602	2.768	13.950	40.18
swa_sink_row	1D r	SWA	Yes	.598	2.969	12.002	40.09
swa_row	1D r	SWA	No	.600	3.010	11.713	38.67
swa2d_sink	2D	SWA	Yes	.597	2.975	12.802	38.87
swa2d	2D	SWA	No	.588	2.914	11.802	36.73
wa2d_sink	2D	Chunk	Yes	.579	2.428	11.146	37.72
wa2d	2D	Chunk	No	.577	2.476	10.690	37.19

Table 15: SHS, Entropy, and Distance for all 9 models at the primary checkpoint. SHS-Index is the all-32-layer mean (matching Table 11; same convention used for the Pearson correlation in §3). Entropy and Distance are computed only over the 28 non-full-attn layers, since the 4 full-attn layers are identical across all 9 architectures and would dominate these distance/entropy averages.

Metric	ρ	p
SHS	**+0.857**	**0.007**
Distance	+0.833	0.010
Entropy	+0.524	0.183

Table 16: Cross-model Spearman ρ vs. benchmark (N=8). SHS is the strongest correlate; Entropy is not significant.

Change	ΔSHS	ΔEnt.	ΔDist.	ΔBench
+Sink (Chunk 2D)	+.002	−0.05	+0.46	+0.53
+Sink (SWA 2D)	+.009	+0.06	+1.00	+2.14
+Sink (SWA 1D)	−.002	−0.04	+0.29	+1.42
+Sink (Full)	+.005	−0.23	−0.10	−0.83
Chunk→SWA (2D)	+.011	+0.44	+1.11	−0.46
Chunk→SWA (2D+s)	+.018	+0.55	+1.66	+1.15
2D→1D (SWA)	**+.012**	**+0.10**	−0.09	**+1.94**
2D→1D (SWA+s)	+.001	−0.01	−0.80	+1.22
+RowCol	+.004	−0.20	+1.95	+0.09

Table 17: Pairwise deltas for each architectural change. The critical 2D→1D transition (bold) shows +1.94 bench improvement while Entropy and Distance are flat; only SHS moves. SHS is the only metric that tracks every step. (ΔSHS uses all-32-layer mean as in Table 15; ΔEntropy/Distance use the non-full-attn layers as in Table 15.)

SHS. Instead, SHS drops in all 5 models tested (Table 23):

Model	bg stl.	obj stl.	Δ	bg>obj
full_sink	0.38%	0.23%	+0.14%	97.9%
swa_sink_row	4.61%	2.86%	+1.74%	96.9%
swa_sink_4r3c	3.79%	2.52%	+1.26%	93.8%
swa2d_sink	2.84%	1.68%	+1.16%	96.9%
wa2d_sink	4.32%	3.69%	+0.64%	78.1%

Table 18: Sink steal analysis (all heads, core L9–21 at the primary checkpoint). bg/obj stl. = fraction of received attention redirected to sink. In SWA models, 94–97% of heads steal more from bg than obj, i.e., sink acts as a *background suppressor*. In wa2d_sink, directionality drops to 78%, explaining why chunk+sink fails.

Setting	High	Low	p (U)	d
8 diagnostic @8k	38.67–40.35	36.73–37.72	**.018**	3.65
9 all @8k	38.67–40.35	36.73–37.72	**.012**	**3.91**
9 all @10k	38.01–40.40	35.21–38.27	**.024**	2.37

Table 19: Robustness check: K-means (k=2) on SHS-Index partitions the controlled configurations into high-SHS and low-SHS groups (benchmark ranges shown). Mann-Whitney U test on the 20-image-task average confirms significant separation. Cohen's d uses unweighted pooled SD. The 9-model grouping remains significant at 10k.

I Efficiency Analysis

Table 24 reports ViT forward pass timing across six resolutions. All timings measured on NVIDIA Hopper GPU, FlashAttention-3 (Shah et al., 2024)

Panel A: High-8 vs. Low-8 (avg over 23 tasks)		
Condition	**Δ vs. base**	
Remove 8 hi-SHS	−1.25	
Remove 8 lo-SHS	−1.07	
hi − lo	−0.18	(14/23 tasks hi<lo)
Panel B: All 16 heads removed		
Task	**Base**	**Δ**
V*	48.69	−13.1
OCRBench	68.00	−16.7
DocVQA	53.46	−13.8
MMMU	29.11	+0.1

Table 20: Head elimination on the full model (core layers). **Panel A**: removing 8 high-SHS vs. 8 low-SHS heads yields negligible difference; individual heads are interchangeable. **Panel B**: removing all 16 heads collapses vision-dependent tasks; MMMU (reasoning-heavy) is unaffected. The collective specialization pattern is indispensable. The head-elimination suite (23 tasks) is a superset of the 22-task primary benchmark, with sub-scores (e.g., MME-cognition vs. MME-perception) reported separately for finer-grained per-task analysis.

Model	**Tasks**	**avg (hi − lo)**
full	23	−0.18
wa2d	23	+0.20
swa_sink_row	25	−0.27

Table 21: High-low SHS head elimination across models. No model shows a significant difference. Task counts vary slightly (23–25) across models because evaluation suites differ in which sub-scores are reported separately at this checkpoint; the conclusion (no high/low gap) is robust to this choice.

	Q	K	V	O	Q×K
full	+.09	+.11	−.06	+.05	+.10
4r3c	+.15	+.16	+.01	+.05	+.16
sink_row	+.00	+.02	−.03	+.13	+.01
swa_row	+.14	+.14	+.06	+.16	+.14
s2d_sink	+.33	+.32	+.05	+.19	+.33
swa2d	+.19	+.20	+.01	+.12	+.20
w_sink	+.12	+.12	−.10	+.07	+.12
wa2d	+.27	+.28	−.01	+.14	+.28
Mean	**+.16**	**+.17**	−.01	**+.11**	**+.17**

Table 22: Spearman ρ between per-head weight Frobenius norms and SHS. Q/K show weak positive correlations (8/8 positive); V shows none (mean −.01).

with `s_aux=yes`, bfloat16, 4 images batched (varlen contiguous batching), warmup 10, iterations 10. ViT: 32 layers, hidden 1280, 16 heads, head_dim 80, MLP 3456. Full-attention layers at positions [7, 15, 23, 31]. TFLOP counts theoretical attention FLOPs ($QK^\top$ + softmax$\cdot V$, multiply-add = 2 ops). `attn_op` is the FlashAttention kernel only (no QKV/proj linear; wa2d includes reorder overhead). `e2e` is the full transformer block (norm + QKV + RoPE + attn + proj + norm + MLP).

Model	**Base SHS**	**Eq. SHS**	**Δ**
full	.6237	.6178	−0.0059
4r3c	.6044	.6001	−0.0044
swa_sink_row	.5993	.5958	−0.0035
swa_row	.6036	.5997	−0.0038
wa2d	.5763	.5751	−0.0013

Table 23: QK equalization: SHS drops in all 5 models. The positive QK-SHS correlation is a consequence, not a cause, of head specialization.

Key conclusions.

- **Sink has zero overhead**: SWA-sink-row and SWA-row produce identical `attn_op` and e2e timings at every resolution, confirming that FlashAttention-3's (Shah et al., 2024) native `s_aux` parameter incurs no additional computation.
- **SWA faster than chunk**: Despite a larger effective window (128 vs. 64 tokens), SWA's kernel time is dramatically lower than wa2d (6.3 ms vs. 17.5 ms at 896^2). FlashAttention-3's contiguous SWA kernel is better optimized than the varlen interface used for chunk windows (Dao et al., 2022; Dao, 2024). The wa2d kernel time also includes the overhead of 2D window reordering.
- **Savings grow with resolution**: At 896^2, Ariadne saves 12.4% e2e vs. full attention; at 1344^2, the saving grows to 26.4%. Full attention TFLOP scales quadratically (0.04T → 55.7T), while SWA scales linearly (0.02T → 7.64T).

J Why Sink Bias Hurts Full Attention

The Full+Sink model has the highest SHS-Index (0.611) yet ranks only fourth on benchmarks (39.5 vs. 40.4 for Full), a −0.9 drop. In contrast, sink bias consistently *improves* SWA models (+0.5 to +2.1). Three lines of evidence explain this asymmetry.

Full attention already has implicit sinks. Guo et al. (2024) show that full-attention heads naturally develop an *active-dormant* mechanism: for each

		Full			wa2d (Chunk)			SWA-row			SWA-sink-row		
Res	**Tok**	**FLOP**	**op**	**e2e**	**FLOP**	**op**	**e2e**	**FLOP**	**op**	**e2e**	**FLOP**	**op**	**e2e**
224^2	256	0.04	1.2	22.1	0.01	2.8	24.4	0.02	1.2	22.1	0.02	1.2	22.3
448^2	1024	0.69	2.2	32.1	0.12	4.5	32.9	0.16	1.4	31.4	0.16	1.4	31.3
672^2	2304	3.48	7.6	68.7	0.52	9.7	68.4	0.60	2.9	64.6	0.60	2.9	64.6
$\mathbf{896^2}$	**4096**	**11.0**	**23.9**	**124.4**	1.52	17.5	115.3	1.68	6.2	109.0	**1.68**	**6.3**	**109.0**
1120^2	6400	26.8	55.2	211.2	3.59	28.8	178.1	3.83	12.1	168.5	3.83	12.0	168.6
$\mathbf{1344^2}$	**9216**	**55.7**	**110.3**	**330.0**	7.30	44.3	255.5	7.64	21.2	242.2	**7.64**	**21.1**	**242.9**

Table 24: ViT forward pass profiling. FLOP = attention TFLOP (tera floating-point operations); op = FlashAttention kernel time (ms); e2e = full block time (ms). SWA-sink-row ≡ SWA-row: sink adds *zero* overhead (FA3 native s_aux support). At 896^2: Ariadne 109.0 ms vs. Full 124.4 ms (-12.4%). At 1344^2: 242.9 ms vs. 330.0 ms (-26.4%). The attention kernel difference is even more dramatic: at 896^2, SWA-sink 6.3 ms vs. Full 23.9 ms (-73.6%). SWA is faster than chunk (wa2d) despite a larger window because FA3 optimizes contiguous SWA better than varlen chunk windows.

input, some heads route attention to implicit sink tokens whose value-state norms approach zero, effectively implementing a learned, input-dependent no-op. Barbero et al. (2025) further demonstrate that sinks function as anti-over-mixing regulators, and that larger context windows develop stronger implicit sinks. A learnable sink bias $s_{\text{aux}}^{(h)}$ is a fixed, input-independent scalar; it competes with the pre-existing input-dependent mechanism rather than complementing it.

SWA lacks this implicit mechanism. With only 128 tokens per window, SWA models cannot develop effective implicit sinks; the local context is too limited. The softmax constraint forces each query to distribute its full attention budget within the window, wasting mass on uninformative background tokens. Here, an explicit sink bias fills a genuine gap: it provides the no-op channel that the architecture cannot learn on its own.

Over-specialization reduces diversity. Full+Sink's obj-dominant head fraction is 82.3% (the highest among all 9 models), meaning sink bias pushes too many heads toward foreground. This reduces the collective role diversity that §H identifies as indispensable, explaining why SHS-Index rises (heads are individually more specialized) while benchmark falls (the ensemble loses complementary background expertise).

This analysis is consistent with Gu et al. (2025), who show that attention sinks arise from the softmax normalization constraint and function as key biases rather than informative signals; adding a second explicit bias atop an already-effective implicit one yields diminishing or negative returns.

K Pearson Correlation

Table 25 reports the full Pearson r matrix. The relationship is strong and significant at the primary checkpoint under both task averages (20 image and 22 with video) and under both $N{=}8$ (eight diagnostic configurations) and $N{=}9$ (adding Ariadne). At the final checkpoint the correlation weakens but the same direction holds, consistent with the wider variance in late-checkpoint benchmark scores reported in Table 6.

Step	Setting	22-task		20 image-task	
		r	p	r	p
8k	$N{=}8$ (diagnostic)	**.79**	**.021**	.81	.015
8k	$N{=}9$ (incl. Ariadne)	**.79**	**.012**	.81	.008
10k	$N{=}8$	.66	.077	.67	.067
10k	$N{=}9$	.67	.048	.69	.041

Table 25: Pearson correlation between SHS-Index and benchmark across controlled architectures. The same trend holds across both benchmark aggregations; the final-checkpoint correlation is weaker because late-checkpoint benchmark variance grows (Table 6).