HyMoR-Attn: Granularity-Matched Repair of Omitted Attention Support for Video Diffusion

Anonymous authors
Paper under double-blind review

Abstract

Block-sparse attention has become the backbone of efficient video diffusion. A growing body of work seeks better fidelity and efficiency trade-offs by improving the accuracy of coarse block selection. Intuitively, a more accurate selector should yield a better frontier. However, we find that coarse-only selection can itself limit the attainable frontier. Video-diffusion attention is inherently two-scale. Coarse blocks efficiently capture a locally dense body while a scattered tail of high-value interactions remains inside omitted blocks. We call this failure the *Omitted-Micro Problem*. We further find that omission risk is hierarchically predictable. Risk is stable at the layer and head level and stable enough for offline per-model calibration while the coordinates of valuable Microtiles remain content-dependent and require online selection. This observation motivates a rethink of selection-centric sparse attention. Improving coarse rankings alone cannot recover fine interactions that the execution granularity makes invisible. We introduce **H**ybrid **M**icro **O**mission **R**epair (HyMoR-Attn), a training-free support-repair method built on a coarse Macro backbone with a fine-grained Micro residual. It combines offline risk calibration to locate vulnerable layers and heads, online peak-preserving selection to recover the omitted tail, and granularity-matched Macro and Micro execution with exact LSE merging. Across four settings spanning two models, two tasks and 480p and 720p resolutions, HyMoR-Attn is Pareto-optimal among the compared methods. It achieves up to 2.06× and 1.72× speedup over Full Attention while reaching up to 33.01 and 28.46 dB PSNR on HunyuanVideo and Wan 2.1 respectively. The quality ceiling of coarse block sparsity is set not by the chosen budget but by the high-value support it discards. Matching repair granularity to local interaction density raises fidelity and efficiency together.

1 Introduction

Text-to-video diffusion models generate high-quality videos at a large computational cost (Kong et al., 2024; Wan Team, 2025). Attention dominates this cost because it scales quadratically with the number of tokens (Wen et al., 2025). Block-sparse attention is the current standard solution for efficient video generation (Kong et al., 2024; Chen et al., 2025). It divides the attention map into regular large blocks, computes only the blocks with the highest scores, and skips the rest (Hu et al., 2026; Zhang et al., 2025). This design reduces computation while keeping GPU tensor cores busy (Hu et al., 2026; Durvasula et al., 2025). The intuition is straightforward. The blocks we keep contain the important interactions, the gold. The blocks we discard are mostly low-value noise, the sand. But what if the discarded sand still contains grains of gold?

Prior work has focused on making block selection more accurate. Fixed-budget Top-k selection keeps the workload regular (Lin et al., 2025), while mass-based Top-p selection adapts the budget to the attention distribution (Zhang et al., 2026). Recent hybrids further introduce safety floors and per-head budgets (Lin et al., 2026; Liu et al., 2026b; Zhao et al., 2026). All these methods share one unexamined assumption. Once a block is discarded, nothing inside it is worth computing. Existing selectors ask which blocks to keep, but they never look inside the blocks they throw away.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

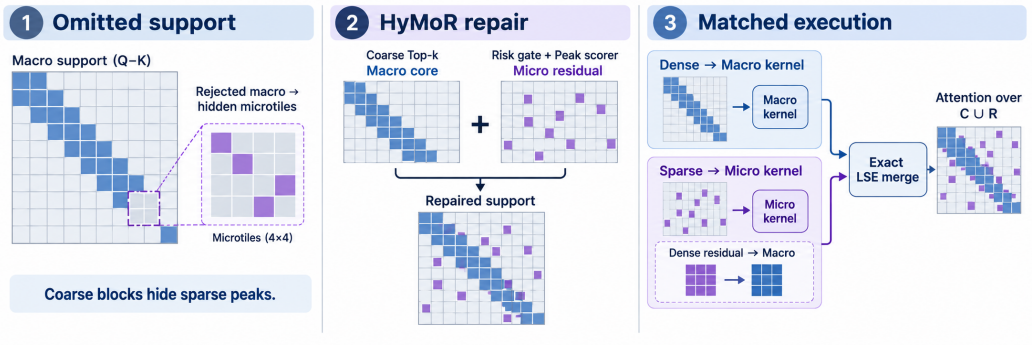


Figure 1: Matching execution granularity to local support density recovers omitted high-value interactions at low cost. Coarse block-sparse attention leaves high-value Microtiles inside rejected Macro blocks (1). HyMoR-Attn repairs this gap with a Macro core plus a peak-selected Micro residual (2), and computes each region at a granularity matched to its local density (3).

We look inside, and we find something surprising. Most of the recoverable quality in rejected blocks is often concentrated in just a few small Microtiles. In a representative case, a single K16 Microtile recovers 93.5% of the quality lost by discarding its parent Macro block, even though the block's pooled score placed it far below the selection threshold. Coarse block scores dilute these high-value peaks across many low-value interactions, so the entire block looks like sand and gets thrown away. Increasing the block budget does not solve this problem efficiently. It restores every interaction in a rejected block, even when only a small fraction matters. We call this failure the *Omitted-Micro Problem*. This two-scale structure reveals a deeper limitation. No single execution granularity fits both the dense body of important interactions and the scattered tail of high-value peaks. Coarse execution wastes compute on low-value interactions inside kept blocks, while it discards isolated high-value interactions outside them. Fine execution can preserve these scattered peaks, but using it everywhere sacrifices hardware efficiency in dense regions. Quality and latency both become tied to one block size. We call this limitation the *granularity trap*.

Omission risk follows a clear hierarchy. At the layer and head level, which parts of the model are most likely to lose valuable interactions stays consistent across different videos and random seeds. This stability means we can identify vulnerable layers and heads once, offline. At the level of individual token coordinates, however, which Microtiles contain valuable interactions depends on the input content. These coordinates must be selected online for each video.

We address this gap with HyMoR-Attn, a training-free support repair method that recovers the omitted high-value tail. HyMoR-Attn separates two decisions that single-granularity methods combine into one. *Quality routing* decides which interactions to keep. A hardware-aligned Macro core handles the locally dense support, the gold blocks. A peak-preserving Micro residual recovers scattered high-value interactions, the gold grains hidden in the sand. *Granularity matching* decides how to compute each region. Dense regions run at the Macro level for regular, efficient execution. Scattered residuals run at the Micro level so we do not pay for low-value interactions. An exact LSE merge then computes attention over the union of both regions. Offline calibration decides where Micro repair is worthwhile. An online scorer decides which Microtiles to recover for each input. Figure 1 summarizes this design.

This design matches execution granularity to the local structure of the attention support. It keeps the efficiency of coarse block-sparse attention in dense regions, and pays fine-grained execution cost only where omitted interactions offer high recovery value. The quality ceiling of coarse block sparsity is not set by how many blocks we keep. It is set by how much gold we throw away with the sand. Matching repair granularity to local density raises fidelity and efficiency together. Our contributions are:

- **The Omitted-Micro Problem.** We identify the *Omitted-Micro Problem* in sparse video attention: a coarse block rejected as low-value can still contain a few token interactions with exceptionally high recovery value. Restoring only a handful of Microtiles recovers most of the benefit of restoring the full block.

- **Hierarchically Predictable Omission Risk.** We show that omission risk follows a two-level structure: it is stable at the layer and head level, suitable for offline per-model calibration, but content-dependent at token coordinates and requires online selection.
- **Granularity-Matched Support Repair (HyMoR-Attn).** Motivated by these findings, we introduce HyMoR-Attn, a training-free method that preserves an efficient Macro backbone and pays fine-grained execution cost only for the recovered high-value tail. It assigns Macro or Micro execution according to local support density and uses an exact LSE merge to compute attention over their union. HyMoR-Attnoutperforms every evaluated sparse baseline in fidelity at comparable or lower end-to-end latency, improving PSNR from 31.98 to 33.01 dB over DFSAttn on HyVideo t2v at 720p.

2 Related Work

Block-sparse attention for video diffusion. Block-sparse attention is the standard approach for efficient video diffusion. The field has evolved from static predefined patterns (Xi et al., 2025; Li et al., 2025c; Chen et al., 2025) to dynamic input-adaptive selection, with DFS-Attn (Hu et al., 2026) as a representative method, and further to adaptive budgeting with per-head safeguards (Lin et al., 2025; Zhang et al., 2026; Lin et al., 2026; Liu et al., 2026b). All these methods share a common default. Once a coarse block falls below the selection threshold, the entire block is discarded. This line of work has focused on making the selector more accurate, but it has not asked whether discarded blocks still contain interactions worth recovering at finer granularity.

Fine-grained structure and our positioning. A separate line of work explores fine-grained attention structure. FG-Attn, PSA, VSA, and HEART study fine-grained structure within or above the coarse selected support (Durvasula et al., 2025; Li et al., 2025b; Zhang et al.; Zheng et al., 2026). Two recent findings are particularly relevant. Veda (Han et al., 2026) shows that extrema-aware statistics improve tile selection over mean pooling. MonarchRT (Agarwal et al., 2026) reports that even oracle Top-k selection can fail because blockwise averaging hides sparse high-value patterns. Both show that coarse aggregation loses information, but neither asks whether that information survives in blocks that were already discarded. HyMoR-Attn addresses this gap by recovering fine-grained interactions from within discarded coarse blocks, at a granularity matched to the recovered support. Prior structural analyses support this two-scale view (Wen et al., 2025; Bolya et al., 2023). To our knowledge, HyMoR-Attn is the first method that repairs omitted fine-grained support from within discarded coarse blocks. A broader discussion is provided in Appendix A.1.

3 Preliminaries

We consider one attention layer of a video diffusion transformer. After a Hilbert3D permutation of the video tokens, the layer takes queries, keys, and values $\boldsymbol{Q}, \boldsymbol{K}, \boldsymbol{V} \in \mathbb{R}^{H\times N\times d}$ and computes $\mathrm{softmax}(\boldsymbol{Q}\boldsymbol{K}^\top/\sqrt{d})\boldsymbol{V}$. The cost grows quadratically with N, and sparse attention restricts the computation to a selected support set.

We work with two logical granularities. A **Microtile** is a $Q16 \times K16$ block. A **Macro tile** is a $Q128 \times K96$ block, composed of $8 \times 6 = 48$ Microtiles. The Macro shape aligns with the native work-block size of modern GPU tensor cores, so the dense Macro core runs with high hardware utilization. The Micro tile is finer-grained and used only where the support is too sparse to fill a Macro efficiently.

4 Diagnosing the Granularity Trap

Before presenting the method, we establish four diagnostic facts about coarse block-sparse attention in video diffusion. The facts form a chain: we first show what value is lost inside omitted blocks, then explain why coarse selectors cannot see it, then show that omission risk is predictable at the layer and head level but not at the token level, and finally demonstrate that no single execution granularity can repair both the dense body and the scattered tail. Each fact motivates one component of HyMoR-Attn.

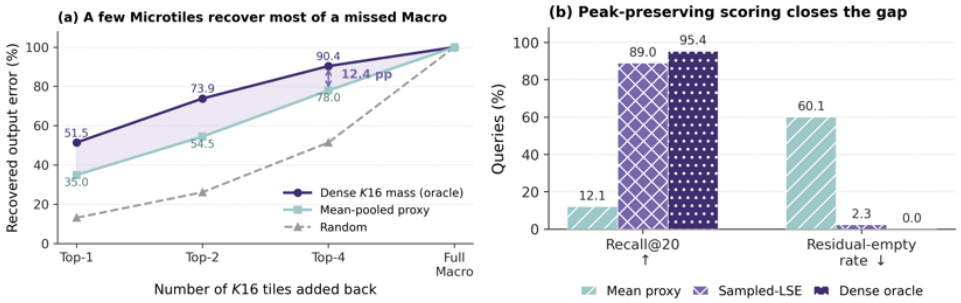


Figure 2: Two faces of the Omitted-Micro Problem, measured at different scopes. **(a)** *Within-block recovery*, averaged over all omitted Macros: cumulative recovered output error as Top-k $K16$ tiles are added back inside a given omitted block, for the dense-mass oracle, the mean-pooled proxy, and a random baseline. The oracle reaches 90.4% with four tiles while the mean-pooled proxy lags by 12.4 percentage points. Concentration is much sharper in the most severe omissions. **(b)** *Cross-block ranking within each Q16 row* ($\rho_C = 0.20$): Recall@k is the percentage of high-error Q16 rows where the scorer ranks the highest-addback omitted $K16$ tile of that row in its top k. The mean-pooled proxy reaches only 12.1% at Top-20, the sampled-LSE scorer reaches 89.0%, and the dense-mass oracle reaches 95.4%.

4.1 The Omitted-Micro Problem

Missed Macro blocks are dominated by a few high-value $K16$ tiles. Figure 2(a) shows cumulative add-back recovery averaged over all omitted Macros: the dense oracle recovers 90.4% of the full-block benefit with four $K16$ tiles, and even the mean-pooled proxy reaches 78.0%. Concentration is much sharper in the most severe omissions, where a single $K16$ tile recovers 93.5% of the full-block benefit and two tiles recover 99.4%. This pattern holds at both the Macro level and the finer $Q16 \times K16$ level. We measure recovered error ΔE as the reduction in relative L2 distance to the dense output; the full protocol and all recovery numbers are in Appendix A.4 (Tables 5, 6, and 7).

The value is also long-tailed across blocks. Half of the omitted Macros have benefit below 0.00037, while the single most severe Macro reaches $\Delta E = 0.7603$, accounting for 77.5% of the total error. Most omitted blocks contribute little, and the few that matter concentrate their recoverable quality in a handful of $K16$ tiles. Uniformly enlarging the Macro budget is therefore uneconomical. Targeted fine repair restricted to severe omissions is the right response.

4.2 Why Coarse Scores Miss Them

Mean pooling fails at cross-block ranking within a row, not just within-block sorting. The two panels of Figure 2 measure selection quality at two very different scopes. Comparing them reveals where mean pooling breaks. Panel (a) asks how well the mean proxy ranks $K16$ tiles inside one omitted Macro. It works reasonably well. It recovers 78.0% of the full-block benefit with four tiles in the all-Macro average and 92.9% inside the most severe omissions. Panel (b) asks a harder question. Within each high-error Q16 row, across all its omitted Macros, does the scorer rank the row's highest-addback $K16$ tile near the top? Here the mean proxy collapses to 12.1% Recall@20, far below the 95.4% oracle.

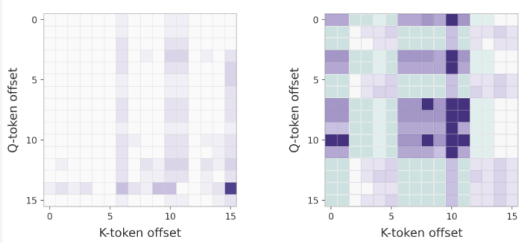


Figure 3: The mean-pooling blind spot. Pooling a $K16$ tile dilutes a few extreme positive logits into a sea of near-zero values. It also lets large positive and negative logits cancel. The pooled score bears little relation to true tile mass.

The cause is baseline shift. Tiles in the same Macro share roughly the same baseline. Their pooled order reflects real differences in mass. Across blocks the baseline shifts. A mediocre tile in a high-

benefit Macro can have a higher mean score than the best tile in a low-benefit one. Pooling washes out the peaks that distinguish high-value tiles from noise. This makes scores incomparable across blocks (Figure 3). The sampled-LSE scorer closes most of this gap. It reaches 89.0% at Top-20. Full numbers across Core ratios are in Appendix A.5. This cross-block ranking failure motivates the peak-preserving Micro scorer described in §5.1.1. That scorer approximates tile mass without paying for a full dense computation.

4.3 Omission Risk Is Hierarchically Predictable

Omission risk follows a two-level structure that directly motivates our design: it is *stable at the (layer, head) level* (so it can be calibrated offline) but *content-dependent at the token-coordinate level* (so it must be selected online).

Layer/head risk is stable across videos. We define omission risk per (layer, head) as the total output-error recovery achievable by restoring the best omitted $K16$ tile in each high-error $Q16$ row, so a higher score means that position leaks more quality to omitted Microtiles (full definition in Appendix A.11). The per-(layer, head) risk ranking transfers well across videos, with Spearman correlation $\rho = 0.71$ and Top-600 set overlap $= 0.86$ (Figure 4(a)). The ranking is also stable across Core ratios. A single per-model calibration therefore suffices: the risk set is computed once and reused across inputs. Full stability data and dense attention maps are in Appendices A.6 and A.7.

Token coordinates are content-dependent. While high-risk head *positions* transfer across videos, the *coordinates* of valuable Microtiles within them do not. A fixed-position prior reaches only 12.7% Recall@20, while the online sampled-LSE scorer reaches 89.0% (Figure 4(b,c)).

Together with the finding from §4.1 that severe omissions concentrate most of their recoverable value in one or two $K16$ tiles, this two-level split makes targeted fine repair efficient. The split localizes where the repair problem lives: which positions need repair is stable enough to calibrate offline, and which tiles within them must be selected online. The offline risk ranking decides where to invest the repair budget, and the online scorer decides which tiles to keep within those positions.

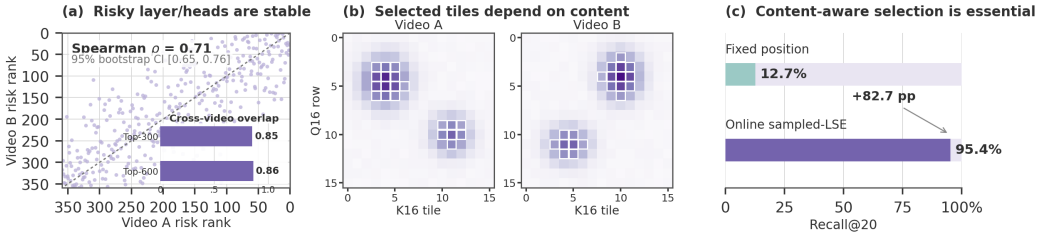


Figure 4: Two-level risk structure. **(a)** The per-(layer, head) omission-risk ranking r_{lh} is stable across videos: rank-rank scatter of two videos, with Spearman $\rho = 0.71$ and top-set overlap growing with set size (Top-600 $= 0.86$; full sweep in Appendix A.6). **(b)** Within a fixed risk (layer, head), the high-value $K16$ coordinates are content-dependent: Q16×K16 heatmaps for Video A and Video B, with white boxes marking each video's own Top-20 tiles, and the two sets do not coincide. **(c)** Offline fixed positions transfer at only 12.7% Recall@20, while the online sampled-LSE scorer reaches 89.0%.

4.4 The Single-Granularity Dilemma

At the same realized density (0.20 of Full Attention) on the primary setting, coarse block-sparse attention (DFSAttn, selecting and executing at its native $Q128 \times K128$ block) reaches 28.23 dB PSNR in 211 seconds, while an all-fine configuration that selects and executes every interaction at $Q16 \times K16$ (Tiled-16) reaches 28.98 dB but takes 401 seconds (Table 1). The all-fine endpoint recovers 0.75 dB of quality by repairing the scattered tail, but it costs 90% more latency because locally dense regions are fragmented into many small kernels, losing tensor-core utilization and memory locality. This quantifies the granularity trap introduced in §1: coarse execution is fast but leaves the tail unrepaired, and fine execution repairs the tail but wastes the dense body. Sparse attention must preserve a regular coarse backbone for the dense body while repairing the scattered tail at a finer granularity, executing each region at a granularity matched to its local density.

Table 1: Single- vs mixed-granularity endpoints on the primary setting (HyVideo, t2v, 480p). All sparse rows realize density 0.20. Block shapes are $Q \times K$ tokens; E2E is end-to-end GPU time.

Method	Granularity	Density	PSNR↑	E2E(s)↓
Full Attention	dense	1.00	–	315
DFSAttn	128×128	0.20	28.23	211
Tiled-16 (ablation)	16×16	0.20	28.98	401
HyMoR-Attn (Top-600)	$128 \times 96 + 16 \times 16$	0.20	**28.89**	**198**

5 HyMoR-Attn

5.1 Quality Routing

Building on the diagnosis above, HyMoR-Attn is a training-free support repair method that separates two decisions single-granularity methods conflate. **Quality routing** decides which interactions are worth keeping, and **granularity matching** decides at what granularity to compute them. Figure 1 summarizes the full pipeline, and throughout this section *Residual* refers to the Micro repair branch.

5.1.1 Peak-Preserving Micro Scorer

The Macro Core uses standard top-k selection with a downsampled mean-pooled score. This coarse selector is efficient, but it underestimates tiles that contain a few extreme logits, as shown in §4.2. The Residual scorer therefore needs to approximate both the total attention mass of a $K16$ tile and the presence of high-value peaks inside it. We use a *sampled log-mean-exp* over a small set of representative vectors.

For a 16-token block $\boldsymbol{X}$, we take the mean $\mu_{\boldsymbol{X}}$ and the three tokens with the largest residual norm

$$r_t = \|\boldsymbol{x}_t - \mu_{\boldsymbol{X}}\|_2, \tag{1}$$

giving $\mathcal{R}(\boldsymbol{X}) = \{\mu_{\boldsymbol{X}}, \boldsymbol{x}_{t_1}, \boldsymbol{x}_{t_2}, \boldsymbol{x}_{t_3}\}$. The mean captures the overall trend; the outliers capture the peaks that pooling discards.

For $Q16$ block i and $K16$ block j, the 4×4 representative pairs yield logits $L_{ij}^{ab} = \mathcal{R}(\boldsymbol{Q}_i)_a \mathcal{R}(\boldsymbol{K}_j)_b^\top / \sqrt{d}$, and the score is

$$Z_{ij} = \log\left(\frac{1}{16}\sum_{a,b} \exp L_{ij}^{ab}\right). \tag{2}$$

Log-mean-exp accumulates mass when the distribution is flat and approaches the max when a single extreme logit dominates, so no manual weighting is needed. The estimated mass is

$$\hat{P}_{ij} = \mathrm{softmax}_j\left(Z_{ij}\right), \tag{3}$$

used as the tile-level proxy for complement Top-p selection below.

5.1.2 Complement Top-p with Risk-Ranked Budget Allocation

Top-p alone misallocates the residual budget. Because mass concentrates heavily inside the Core, most rows reach $p_{total} = 0.90$ (selection rationale in Appendix A.11) with zero additional tiles, leaving severe omissions unrepaired. We therefore use a *complement Top-p* with a per-row budget clamp.

For each risk $(head, Q16)$ row, let $\mathcal{C}_i$ be the $K16$ tiles covered by the Core, with mass

$$m_i^C = \sum_{j \in \mathcal{C}_i} \hat{P}_{ij}. \tag{4}$$

We select the smallest complement prefix P_i such that $m_i^C + \sum_{j \in P_i} \hat{P}_{ij} \geq p_{total}$ ($p_{total} = 0.90$). The Residual fills the mass missing from the original distribution; it is not renormalized over the uncovered region.

To ensure every risk row receives at least some repair, we clamp the per-row tile count to a moderate range (values in Appendix A.11). The final Residual is

$$R_i = \mathrm{TopK}_{k_i}\left(\hat{P}_{i,\overline{\mathcal{C}}_i}\right). \tag{5}$$

Risk-head routing. The scorer, selection, and Residual execution run *only* on the top N_{risk} (layer, head) positions, ranked by offline omission-risk calibration (Appendix A.11). All other positions use the Macro core alone. This follows directly from the mechanism analysis (§4.3): severe omissions concentrate in a small fraction of positions, so paying the scorer cost everywhere is uneconomical. The ranking is calibrated once per model and reused across inputs; N_{risk} is a quality–speed parameter (operating points in Appendix A.11). The resulting tile placements are visualized in Appendix A.8.

5.2 Granularity Matching

The diagnosis in §4.4 shows that no single granularity serves both the dense Core body and the scattered residual tail, so this section gives each branch an execution path matched to its local structure.

5.2.1 Dual-Granularity Computation

The Macro branch computes the block-sparse core using warp-group level WGMMA instructions. These instructions deliver high throughput for regular large-block matrix operations.

The Micro branch handles scattered residual interactions at finer granularity. Each program processes one head and one Q16 row. It iterates over selected K16/V16 tiles, executes small Tensor-Core MMA via `tl.dot`, and maintains online softmax state.

The Macro branch exploits WGMMA for high throughput on regular blocks. The Micro branch uses small MMA tiles to retain execution flexibility for irregular sparse regions.

To further improve hardware efficiency, regions with sufficiently high microtile occupancy are promoted to Macro execution. This is a form of hardware-aware support rounding. It delivers better data reuse and higher arithmetic intensity through one regular block computation.

5.2.2 Exact LSE Merge of Dual Supports

The Core and Residual are each computed separately at their own granularity. The Core is computed at the Macro level as a standard block-sparse attention; the Residual is computed tile by tile at the Micro level. Both return their output and log-sum-exp (LSE) vector.

The two branches are each locally normalized and cannot be added directly. With $m = \max(L_C, L_R)$, $w_C = \exp(L_C - m)$, and $w_R = \exp(L_R - m)$, the merge is

$$\boldsymbol{O} = \frac{w_C \boldsymbol{O}_C + w_R \boldsymbol{O}_R}{w_C + w_R}, \tag{6}$$

which is exactly equivalent to one softmax over $C \cup R$: each branch is an exact softmax over its own support, and the weighted merge recovers the full normalization. Rows with an empty Residual keep the Core output directly.

6 Experiments

6.1 Experimental Setup

Models and tasks. We evaluate on four settings spanning two models: HunyuanVideo-13B text-to-video at 480×720 and 720×1280, and Wan2.1-14B Wan Team (2025) text-to-video and image-to-video at 720×1280. All methods share the same prompts and seeds with a 12-step dense warmup.

Baselines. We compare against Full Attention, DFSAttn Hu et al. (2026), RAPID Lin et al. (2026), AdaSpa Xia et al. (2025) and SVG-2 Yang et al. (2026). All baselines are training-free and run on the same hardware at identical resolution and steps.

Table 2: Quality and efficiency results across the evaluated HunyuanVideo-13B and Wan2.1-14B settings. Density is the realized fraction of QK interactions relative to Full Attention. Top-N denotes the top-N risk (layer, head) positions.

Method	Quality				Efficiency		
	PSNR↑	SSIM↑	LPIPS↓	VBench↑	Density	E2E(s)↓	Speedup↑
HunyuanVideo-13B t2v, 480p							
Full Attention	–	–	–	81.12	1.00	315	–
AdaSpa	25.86	0.838	0.133	80.06	0.21	242.31	1.30
SVG-2	26.50	0.863	0.113	80.22	0.28	202.00	1.56
RAPID	25.75	0.819	0.172	80.03	0.25	238.64	1.32
DFSAttn	28.23	0.881	0.096	80.65	0.20	211.00	1.49
HyMoR-Attn (Top-600)	**28.89**	**0.892**	**0.087**	80.82	**0.20**	**198**	**1.59**
HyMoR-Attn (Top-300)	**28.48**	**0.887**	**0.094**	80.71	**0.20**	**192**	**1.64**
HunyuanVideo-13B t2v, 720p							
Full Attention	–	–	–	80.80	1.00	1715	–
AdaSpa	29.87	0.892	0.103	79.95	0.21	1054.49	1.63
SVG-2	30.51	0.917	0.083	80.11	0.29	878.00	1.95
RAPID	29.76	0.873	0.142	79.93	0.25	1025.68	1.67
DFSAttn	31.98	0.934	0.076	80.48	0.20	902.00	1.90
HyMoR-Attn (Top-600)	**33.01**	**0.947**	**0.054**	80.74	**0.20**	**868.00**	**1.98**
HyMoR-Attn (Top-300)	**32.64**	**0.940**	**0.057**	80.65	**0.20**	**831.00**	**2.06**
Wan2.1-14B t2v, 720p							
Full Attention	–	–	–	80.40	1.00	1801	–
SVG-2	23.61	0.783	0.192	78.99	0.29	1060.85	1.70
RAPID	24.32	0.751	0.189	79.16	0.23	1215.15	1.48
DFSAttn	26.47	0.804	0.178	79.70	0.23	1083.78	1.66
HyMoR-Attn	**27.39**	**0.813**	**0.163**	79.93	**0.20**	**1049.74**	**1.72**
Wan2.1-14B i2v, 720p							
Full Attention	–	–	–	79.90	1.00	1441	–
SVG-2	26.59	0.783	0.186	79.14	0.29	863.00	1.67
DFSAttn	27.65	0.816	0.172	79.40	0.20	894.00	1.61
HyMoR-Attn	**28.46**	**0.823**	**0.167**	79.60	**0.20**	**851.00**	**1.69**

Data splits. The omission-risk ranking is calibrated on a held-out 30% subset of the Penguin Video Benchmark (Tencent-Hunyuan, 2025). Validation prompts select p_{total} and $\tau_{promote}$. Test results use the full VBench prompt suite Huang et al. (2024).

Metrics. Quality is measured against the Full Attention reference using PSNR, SSIM, and LPIPS Zhang et al. (2018), with VBench scores also reported. Efficiency is evaluated by steady-state end-to-end GPU time, retained QK interaction density, and speedup over Full Attention. All experiments run on a single NVIDIA GH200 GPU with CUDA 12.4.

6.2 Main Results

We present the main results in two parts, each answering one question about the method. We first verify that HyMoR-Attn is Pareto-optimal across settings and robust to the risk-head budget, and then isolate the mechanism of the quality gain through component ablations.

6.2.1 Consistent Pareto Improvement across Settings

Table 2 shows that HyMoR-Attn consistently moves the quality–efficiency frontier across all four settings. Compared with the strongest baseline, DFSAttn, it improves PSNR by 0.66–1.03 dB at comparable or lower end-to-end latency and at equal or lower realized density, with every quality improvement being significant under paired tests ($p < 0.05$). DFSAttn reaches a quality ceiling because its uniform coarse block size cannot repair the scattered tail; Full Attention is the dense upper bound but costs 1.5–$2.1\times$ more latency. HyMoR-Attn narrows the VBench gap to Full Attention by 0.07–0.11 points across settings, getting closer to dense quality without dense cost. This advantage holds for both HunyuanVideo and Wan2.1, text-to-video and image-to-video generation, and the evaluated 480p and 720p resolutions. Although the high-risk layer–head sets differ between the two models, the same calibration procedure identifies them without model-specific redesign. These results show that the gain is not tied to one architecture or evaluation setting, but reflects a transferable

Table 3: Component ablations on the VBench 10% subset (HyVideo, t2v, 480p). Each row removes or randomizes one element of the full method.

Configuration	Density	PSNR↑	SSIM↑	LPIPS↓
Full method	**0.1984**	**28.89**	0.892	**0.087**
w/o Micro residual	0.1961	28.32	0.882	0.095
random tile selection	0.1984	28.55	0.885	0.094
random head selection	0.1984	28.64	0.887	0.093
mean-pooling scorer	0.1983	28.72	0.889	0.091

advantage of granularity-matched support repair. Qualitative examples across all VBench categories are in Appendix A.2.

HyMoR-Attn is also robust to the risk-head budget. As the budget grows from Top-50 to Top-800, PSNR improves monotonically from 32.23 to 33.43 dB while realized density changes by only 0.0020. Top-300 and Top-600 are representative speed- and quality-oriented operating points, improving PSNR by 0.73 dB and 1.05 dB over DFSAttn. The budget is therefore a smooth quality–speed control knob, not a hyperparameter that requires careful tuning. The full sweep is in Appendix A.9.

6.3 Ablations

Table 3 isolates each design choice on the VBench 10% subset, all at matched density. A p_{total} sensitivity sweep is in Appendix Table 4.

The gain comes from repairing omitted Micro interactions, not from denser attention. Removing the Micro residual costs 0.57 dB, the largest drop. Randomizing *which* tiles to keep (at fixed risk heads and matched density) costs 0.34 dB, showing that content-dependent coordinate selection matters; yet it remains above the no-Micro baseline, confirming that putting budget on the right heads recovers some value even with random placement. Randomizing *which* heads to repair (at fixed scorer and tile budget) costs 0.25 dB, so stable offline risk ranking is necessary to place the budget where omissions are most severe. Replacing the peak-preserving scorer with mean pooling costs 0.17 dB, the smallest drop: once the right heads are targeted, peak-aware scoring still adds measurable value. Together the gains decompose cleanly: offline calibration decides *where* to repair (which heads), and online selection decides *which* tiles to keep and how to score them. Both are necessary, and neither relies on increasing density. Detailed analysis is in Appendix A.10.

7 Conclusion

We studied why coarse block selection reaches a quality ceiling in sparse video diffusion attention. The central finding is the Omitted-Micro Problem: blocks discarded by a coarse selector can still contain a small number of high-value fine interactions. The value is long-tailed both across blocks and inside each block, so omitted support is fundamentally a fine-scale repair problem. We further showed that omission risk is hierarchically predictable: layer/head risk is stable enough for per-model offline calibration, while the coordinates of high-value Microtiles remain content-dependent and require online selection. This offline/online split localizes where the repair problem lives.

Across four evaluated settings, HyMoR-Attn is Pareto-optimal among the compared methods, improving PSNR over the strongest baseline at comparable generation cost. The evidence exposes a structural point: the quality ceiling of coarse block sparsity is set not by the chosen budget but by the high-value support it discards, and a repair at a granularity matched to local density raises fidelity without paying a proportionate cost. These results motivate treating omitted support as a fine-scale repair problem, and we hope they encourage sparse-attention designs that match repair granularity to local interaction structure.

AI USE STATEMENT

During the preparation of this work, we used AI-assisted tools solely to polish the writing (e.g., improving grammar and wording). All ideas, methodology, analyses, and experimental results were conceived, carried out, and verified by the authors. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of this publication.

REPRODUCIBILITY STATEMENT

All code and experimental artifacts are publicly released at https://anonymous.4open.science/status/TT8F429. Section 6 specifies the models, prompt splits, metrics, and hardware used for evaluation. Appendix A.11 describes the risk calibration and main hyperparameters.

REFERENCES

Krish Agarwal, Zhuoming Chen, Cheng Luo, Yongqi Chen, Haizhong Zheng, Xun Huang, Atri Rudra, and Beidi Chen. Monarchrt: Efficient attention for real-time video generation. *arXiv preprint arXiv:2602.12271*, 2026.

Daniel Bolya, Cheng-Yang Fu, Christopher Savides, Tao Xiao, and Judy Hoffman. Token merging: Your vit but faster. In *International Conference on Learning Representations (ICLR)*, 2023.

Jiayu Chen, Zhikun Jiang, Maoliang Li, Jiayi Luo, Jiawei Yang, Zihao Zheng, Hengyi Zhang, Guojie Luo, and Xiang Chen. Token radius attention for efficient video generation. *arXiv preprint arXiv:2608.02504*, 2026a.

Junsong Chen, Jincheng Yu, Yitong Li, Shuchen Xue, Haozhe Liu, Jingyu Xin, Yuyang Zhao, Tian Ye, Zhangjie Wu, Zian Wang, Daquan Zhou, Ping Luo, Song Han, and Enze Xie. Sana-video 2.0: Hybrid linear attention with attention residuals for efficient video generation. *arXiv preprint arXiv:2607.21553*, 2026b.

Pengtao Chen, Xianfang Zeng, Maosen Zhao, Peng Ye, Mingzhu Shen, Wei Cheng, Gang Yu, and Tao Chen. Sparse-vdit: Unleashing the power of sparse attention to accelerate video diffusion transformers. *arXiv preprint arXiv:2506.03065*, 2025.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Hangliang Ding, Dacheng Li, Runlong Su, Peiyuan Zhang, Zhijie Deng, Ion Stoica, and Hao Zhang. Efficient-vdit: Efficient video diffusion transformers with attention tile. *arXiv preprint arXiv:2502.06155*, 2025.

Sankeerth Durvasula, Kavya Sreedhar, Zain Moustafa, Suraj Kothawade, Tianlei Pang, Ashish Gondimalla, Suvinay Subramanian, Narges Shahidi, and Nandita Vijaykumar. Fg-attn: Leveraging fine-grained sparse attention in video diffusion models. *arXiv preprint arXiv:2509.16518*, 2025.

Gael Glorian, Ioannis Lamprou, Zhen Zhang, Yujie Yuan, and Hongsheng Liu. Lvsa: Training-free sparse attention for long video diffusion. *arXiv preprint arXiv:2605.31057*, 2026.

Shihao Han, Hao Yang, Xiaofeng Mei, Xinting Hu, Yi Jiang, and Xiaojuan Qi. Veda: Scalable video diffusion via distilled sparse attention. *arXiv preprint arXiv:2605.30325*, 2026.

Jie Hu, Zixiang Gao, Yutong He, and Kun Yuan. Dfsattn: Dynamic fine-grained sparse attention for efficient video generation. *arXiv preprint arXiv:2605.23445*, 2026.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, and Ziwei Liu. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Dongyeun Lee, Amir Zandieh, Vahab Mirrokni, Junmo Kim, and Insu Han. Hypervattention: Efficient sparse attention with spatio-temporal clustering for video diffusion. *arXiv preprint arXiv:2607.03012*, 2026a.

Jongbeom Lee, Hyunwoo Yu, Jincheol Yang, Jaemin Choi, and Suk-Ju Kang. Sparstar: Sparse attention for spacetime autoregressive video synthesis. *arXiv preprint arXiv:2608.10519*, 2026b.

Chunlu Li, Yixuan Pan, Bai Du, Zhenyuan Chen, Yanzhao Li, Hui Dong, Hui Wang, and Zhiqiang Zou. Training-free sparse attention based on cumulative energy filtering. *arXiv preprint arXiv:2606.16317*, 2026a.

Haopeng Li, Yitong Li, Junsong Chen, Tian Ye, Haozhe Liu, Jincheng Yu, Duomin Wang, Ruihua Zhang, Zeke Xie, Enze Xie, and Song Han. Sol-attn: Accelerating video generation inference via on-the-fly attention sparsification. *arXiv preprint arXiv:2607.24027*, 2026b.

Haopeng Li, Shitong Shao, Wenliang Zhong, Zikai Zhou, Lichen Bai, Hui Xiong, and Zeke Xie. Pisa: Piecewise sparse attention is wiser for efficient diffusion transformers. *arXiv preprint arXiv:2602.01077*, 2026c.

Qirui Li, Guangcong Zheng, Qi Zhao, Jie Li, Bin Dong, Yiwu Yao, and Xi Li. Compact attention: Exploiting structured spatio-temporal sparsity for fast video generation. *arXiv preprint arXiv:2508.12969*, 2025a.

Xiaolong Li, Youping Gu, Xi Lin, Weijie Wang, and Bohan Zhuang. Psa: Pyramid sparse attention for efficient video understanding and generation. *arXiv preprint arXiv:2512.04025*, 2025b.

Xingyang Li, Muyang Li, Tianle Cai, Haocheng Xi, Shuo Yang, Yujun Lin, Lvmin Zhang, Songlin Yang, Jinbo Hu, Kelly Peng, et al. Radial attention: O(n log n) sparse attention with energy decay for long video generation. *arXiv preprint arXiv:2506.19852*, 2025c.

Chaofan Lin, Jiaming Tang, Shuo Yang, Hanshuo Wang, Tian Tang, Boyu Tian, Ion Stoica, Song Han, and Mingyu Gao. Twilight: Adaptive attention sparsity with hierarchical top-p pruning. *arXiv preprint arXiv:2502.02770*, 2025. NeurIPS 2025.

Shangran Lin, Lu Lu, Jian Chen, and Qiang Liu. Rapid: Reusing attention sparsity with inter-step adaptation for efficient video diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Akide Liu, Zeyu Zhang, Zhexin Li, Xuehai Bai, Yizeng Han, Jiasheng Tang, Yuanjie Xing, Jichao Wu, Mingyang Yang, Weihua Chen, Jiahao He, Yuanyu He, Fan Wang, Gholamreza Haffari, and Bohan Zhuang. Fpsattention: Training-aware fp8 and sparsity co-design for fast video diffusion. *arXiv preprint arXiv:2506.04648*, 2025a.

Enhuai Liu, Yunke Wang, Yutong Wang, Changming Sun, and Chang Xu. Losa: Near-lossless sparse attention for training-free video diffusion acceleration. *arXiv preprint arXiv:2608.12032*, 2026a.

Hao Liu, Chenghuan Huang, Ye Huang, Zhiying Wen, Hao Liu, Mohan Zhang, Chen Li, Ziyang Ma, Jing Lyu, and Jiangsu Du. Fvattn: Adaptive sparse attention with runtime load balancing for video generation. *arXiv preprint arXiv:2607.16190*, 2026b.

Shanghao Liu, Renze Chen, Size Zheng, Yuanqiang Liu, Yun Liang, and Hailong Yang. Spade: An input-adaptive sparse attention engine for fast video diffusion models inference. *arXiv preprint arXiv:2608.03335*, 2026c. DAC 2026.

Siran Liu, Zane Cao, and Yongchao He. A unified sparse attention via multi-granularity compression. *arXiv preprint arXiv:2512.14082*, 2025b.

Yuxi Liu, Haoyu Li, Zekun Zhang, Tengxu Sun, Yixiang Cai, Jiayong Li, Yifei Xia, Tianle Liu, Baole Ai, Ang Wang, Jiamang Wang, Lin Qu, Kai Zhang, Kun Yuan, and Bin Cui. Sparkdiffusion: Mitigating the high-sparsity trap — a unified framework for up to 265x single-gpu acceleration of visual generation. *arXiv preprint arXiv:2609.23153*, 2026d.

Jiayi Luo, Jiayu Chen, Jiankun Wang, Cong Wang, Hanxin Zhu, Qingyun Sun, Chen Gao, Zhibo Chen, and Jianxin Li. Attention sparsity is input-stable: Training-free sparse attention for video generation via offline sparsity profiling and online qk co-clustering. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2026.

Wentao Ni, Kangqi Zhang, Zhongming Yu, Oren Nelson, Mingu Lee, Hong Cai, Fatih Porikli, Jongryool Kim, Zhijian Liu, and Jishen Zhao. Double-p: Hierarchical top-p sparse attention for long-context llms. *arXiv preprint arXiv:2602.05191*, 2026.

Jayesh Shah, Ganesh Bikshandi, Ying Zhang, Vijay Geng, Felipe Cunha, Felipe Aberger, Marisa Kirisame, et al. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. *arXiv preprint arXiv:2407.08608*, 2024.

Xuan Shen, Chenxia Han, Yufa Zhou, Yanyue Xie, Yifan Gong, Quanyi Wang, Yiwei Wang, Yanzhi Wang, Pu Zhao, and Jiuxiang Gu. Draftattention: Fast video diffusion via low-resolution attention guidance. *arXiv preprint arXiv:2505.14708*, 2025.

Dor Shmilovich, Tony Wu, Aviad Dahan, and Yuval Domb. Liteattention: A temporal sparse attention for diffusion transformers. *arXiv preprint arXiv:2511.11062*, 2025.

Wenhao Sun, Rong-Cheng Tu, Yifu Ding, Zhao Jin, Jingyi Liao, Shunyu Liu, and Dacheng Tao. Vorta: Efficient video diffusion via routing sparse attention. *arXiv preprint arXiv:2505.18809*, 2025.

Tencent-Hunyuan. Penguin Video Benchmark. `https://github.com/Tencent-Hunyuan/HunyuanVideo/blob/main/assets/PenguinVideoBenchmark.csv`, 2025.

Wan Team. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Yuxin Wen, Jim Wu, Ajay Jain, Tom Goldstein, and Ashwinee Panda. Analysis of attention in video diffusion transformers. *arXiv preprint arXiv:2504.10317*, 2025.

Jianzong Wu, Liang Hou, Haotian Yang, Xin Tao, Ye Tian, Pengfei Wan, Di Zhang, and Yunhai Tong. Vmoba: Mixture-of-block attention for video diffusion models. *arXiv preprint arXiv:2506.23858*, 2025.

Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, Jianfei Chen, Ion Stoica, Kurt Keutzer, and Song Han. Sparse videogen: Accelerating video diffusion transformers with spatial-temporal sparsity. *arXiv preprint arXiv:2502.01776*, 2025.

Yifei Xia, Suhan Ling, Fangcheng Fu, Yujie Wang, Huixia Li, Xuefeng Xiao, and Bin Cui. Training-free and adaptive sparse attention for efficient long video generation. *arXiv preprint arXiv:2502.21079*, 2025. ICCV 2025.

Yan Xie, Tiansheng Wen, Tangda Huang, Bo Chen, Chenyu You, Stefanie Jegelka, and Yifei Wang. Scaling attention via feature sparsity. *arXiv preprint arXiv:2603.22300*, 2026. ICLR 2026.

Ruyi Xu, Guangxuan Xiao, Haofeng Huang, Junxian Guo, and Song Han. Xattention: Block sparse attention with antidiagonal scoring. *arXiv preprint arXiv:2503.16428*, 2025.

Shuo Yang, Haocheng Xi, Yilong Zhao, Muyang Li, Jintao Zhang, Han Cai, Yujun Lin, Xiuyu Li, Chenfeng Xu, Kelly Peng, et al. Sparse videogen2: Accelerate video generation with sparse attention via semantic-aware permutation. *Advances in Neural Information Processing Systems*, 38:96965–96991, 2026.

Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, Y. X. Wei, Lean Wang, Zhiping Xiao, Yuqing Wang, Chong Ruan, Ming Zhang, Wenfeng Liang, and Wangding Zeng. Native sparse attention: Hardware-aligned and natively trainable sparse attention. *arXiv preprint arXiv:2502.11089*, 2025.

Chenlu Zhan, Wen Li, Chuyu Shen, Jun Zhang, Suhui Wu, and Hao Zhang. Bidirectional sparse attention for faster video diffusion training. *arXiv preprint arXiv:2509.01085*, 2025.

Jintao Zhang, Chendong Xiang, Haofeng Huang, Jia Wei, Haocheng Xi, Jun Zhu, and Jianfei Chen. Spargeattention: Accurate and training-free sparse attention accelerating any model inference. *arXiv preprint arXiv:2502.18137*, 2025.

Jintao Zhang, Kai Jiang, Chendong Xiang, Weiqi Feng, Yuezhou Hu, Haocheng Xi, Jianfei Chen, and Jun Zhu. Spargeattention2: Trainable sparse attention via hybrid top-k+top-p masking and distillation fine-tuning. *arXiv preprint arXiv:2602.13515*, 2026.

Peiyuan Zhang, Haofeng Huang, Yongqi Chen, Will Lin, Zhengzhong Liu, Ion Stoica, Eric Xing, and Hao Zhang. Vsa: Faster video diffusion with trainable sparse attention, 2025c. *URL https://arxiv. org/abs/2505.13389*.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Qi Zhao, Qirui Li, Hanlin Tang, Yiduo Li, Zhen Guo, Cuifeng Shen, Chao Xu, Zhaosheng Chi, Xiaojin Lu, Kan Liu, Tao Lan, Lin Qu, and Xi Li. Scope: Subspace clustering with online per-head top-k estimation for sparse video attention. *arXiv preprint arXiv:2608.12780*, 2026.

Xuzhe Zheng, Yuexiao Ma, Jing Xu, Xiawu Zheng, Rongrong Ji, and Fei Chao. Heart: Exploiting head heterogeneity in sparse attention for video diffusion. *arXiv preprint arXiv:2605.14513*, 2026.

Kan Zhu, Tian Tang, Qinyu Xu, Yile Gu, Zhichen Zeng, Rohan Kadekodi, Liangyu Zhao, Ang Li, Arvind Krishnamurthy, and Baris Kasikci. Tactic: Adaptive sparse attention with clustering and distribution fitting for long-context llms. *arXiv preprint arXiv:2502.12216*, 2025.

A Appendix

A.1 Extended Related Work

A.1.1 Sparse Support Selection in Video Diffusion

Efficient video diffusion attention commonly selects a sparse support and then executes it with optimized kernels. Static methods impose predefined structure: SVG assigns heads to spatial or temporal patterns Xi et al. (2025), Radial Attention exploits spatiotemporal energy decay Li et al. (2025c), Sparse-vDiT builds kernels for diagonal and vertical-stripe patterns Chen et al. (2025), Efficient-vDiT uses tile-level repetition Ding et al. (2025), Compact Attention exploits structured spatio-temporal sparsity Li et al. (2025a), and HyperVAttention applies spatio-temporal clustering Lee et al. (2026a). Dynamic methods adapt the mask to the input. DFSAttn combines Hilbert3D ordering, hierarchical block scoring, and mask caching Hu et al. (2026); SpargeAttention provides training-free sparse and quantized attention Zhang et al. (2025); AdaSpa uses hierarchical blocks with LSE caching Xia et al. (2025); VORTA routes among attention variants Sun et al. (2025); BSA jointly sparsifies queries and key-values Zhan et al. (2025); Sol-Attn performs on-the-fly attention sparsification Li et al. (2026b); PISA introduces piecewise sparse attention Li et al. (2026c); LVSA targets long-video diffusion with training-free sparse attention Glorian et al. (2026); XAttention uses antidiagonal scoring for block-sparse selection Xu et al. (2025); LoSA fixes a near-lossless mass threshold for training-free acceleration Liu et al. (2026a); Token Radius Attention maps query entropy to token-level budgets Chen et al. (2026a); SparSTAR adapts block-sparse attention to autoregressive video synthesis Lee et al. (2026b); and SparkDiffusion addresses the high-sparsity trap through staged training Liu et al. (2026d). Budgeting has also become adaptive. Twilight analyzes

the complementary failure modes of Top-k and Top-p Lin et al. (2025); Tactic and Double-P use cumulative or hierarchical mass constraints Zhu et al. (2025); Ni et al. (2026); SpargeAttention2, RAPID, FVAttn, and SCOPE combine adaptive mass with fixed or head-specific safeguards Zhang et al. (2026); Lin et al. (2026); Liu et al. (2026b); Zhao et al. (2026); and CEF decouples compute allocation from per-step scheduling Li et al. (2026a). Draft attention and scaling attention further exploit temporal redundancy across steps Shen et al. (2025); Xie et al. (2026), and FPS-Attention adapts sparsity across frames Liu et al. (2025a).

Several systems reduce online selection cost by exploiting persistent structure. InputStable profiles input-stable sparsity offline Luo et al. (2026), LiteAttention reuses cross-step temporal coherence Shmilovich et al. (2025), and VSA learns a coarse stage that predicts critical fine cubes Zhang et al.. These methods establish that prior structure can guide sparse selection. Our finding resolves predictability at two different levels: layer/head omission risk is stable enough for per-model calibration, while high-value token coordinates remain content-dependent. The stable level allocates where online search is worthwhile; the current input still determines which Microtiles are repaired.

A.1.2 Fine-Grained and Multi-Granularity Attention

Granularity is already an explicit design dimension in sparse attention. FG-Attn exposes fine-grained sparsity inside blocks and provides $M \times N$ kernels that skip near-zero work Durvasula et al. (2025). PSA decouples logical blocks from hardware tiles and assigns multi-level pooled representations according to block importance Li et al. (2025b). SPADE selects among 3D blocking schemes at runtime Liu et al. (2026c). VSA uses learned coarse-to-fine selection Zhang et al., VMoBA applies mixture-of-block attention to video diffusion Wu et al. (2025), HEART exploits head-level heterogeneity in sparse video attention Zheng et al. (2026), Token Radius Attention computes query-specific token-level budgets Chen et al. (2026a), SANA-Video 2.0 combines linear attention with attention residuals for hybrid efficient video diffusion Chen et al. (2026b), while NSA and UniSparse apply hierarchical compression in long-context language models Yuan et al. (2025); Liu et al. (2025b). These approaches show that finer structure can improve mask construction, representation allocation, or execution efficiency. HyMoR-Attn studies the complementary case in which valuable fine structure lies inside support already omitted by a coarse core. It recomputes those interactions from raw keys and values, preserves the Macro backbone, and merges the two supports exactly.

Structural analyses provide closely related evidence. Attention in video diffusion is highly non-uniform, and token merging methods exploit substantial redundancy Wen et al. (2025); Bolya et al. (2023). Veda shows that extrema-aware statistics improve tile selection over mean pooling Han et al. (2026), while MonarchRT reports that even oracle Top-k selection can fail on video attention containing periodic, sparse, and dense patterns Agarwal et al. (2026). We localize the consequence of this heterogeneity: the value remaining inside omitted coarse blocks concentrates in a few Microtiles. This omitted micro structure motivates both peak-preserving online selection and execution at a granularity matched to the recovered support.

A.1.3 Hardware-Aligned Sparse Execution

FlashAttention introduced IO-aware tiling and recomputation Dao et al. (2022), and FlashAttention-3 improves Hopper utilization with asynchronous tensor cores Shah et al. (2024). These systems optimize execution for a given support and layout. HyMoR-Attn makes layout assignment part of support repair: a regular Macro path handles the locally dense body, a Micro path handles the scattered recovered tail, and an exact LSE merge produces attention over their union.

Taken together, prior work addresses three neighboring questions: how to select a sparse support, how to expose or represent fine-grained structure, and how to execute a chosen layout efficiently. We connect them through the omitted support itself. We identify its concentrated Micro-scale value, separate stable layer/head risk from content-dependent token coordinates, and repair the recovered interactions at a granularity matched to local density. HyMoR-Attn is the resulting end-to-end design.

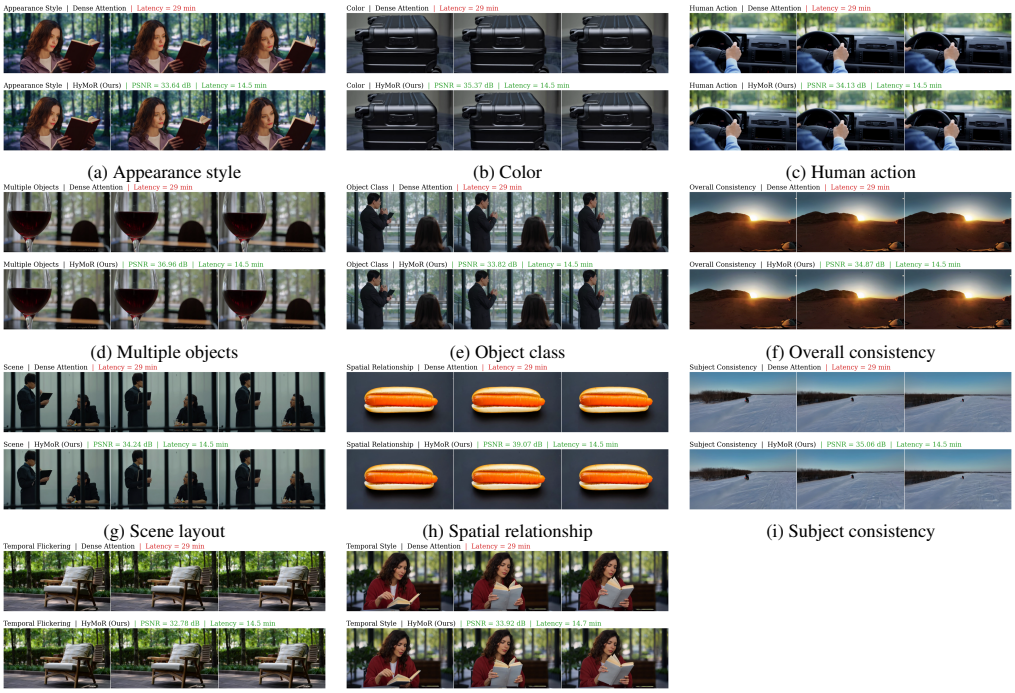


Figure 5: Qualitative examples across the eleven VBench evaluation dimensions. Each panel shows a representative frame generated by HyMoR-Attn on one prompt from the corresponding category.

A.2 Qualitative Examples on VBench Categories

The main text reports VBench scores as a quantitative summary. Here we complement those numbers with qualitative examples across all eleven VBench evaluation dimensions. Each panel shows a representative frame generated by HyMoR-Attn on one prompt from the corresponding category, covering appearance style, color, human action, multiple objects, object class, overall consistency, scene layout, spatial relationship, subject consistency, temporal flickering, and temporal style.

A.3 Sensitivity of p_{total}

Table 4 sweeps the complement Top-p mass target p_{total} on a VBench subset (HyVideo, t2v, 480p). PSNR improves monotonically with p_{total} but saturates above 0.85, while E2E time starts increasing again at 0.95 because the per-row budget cap is reached for more rows. We use $p_{total} = 0.90$ as the main setting.

Table 4: Sensitivity of the complement Top-p mass target p_{total} on a VBench subset (HyVideo, t2v, 480p).

p_{total}	Density	E2E (s)↓	PSNR↑
0.60	0.1973	186.0	28.23
0.70	0.1977	192.0	28.35
0.80	0.1982	199.0	28.46
0.85	0.1983	196.3	28.48
0.90	**0.1985**	**196.4**	**28.48**
0.95	0.1987	200.0	28.49

A.4 ADD-BACK RECOVERY DETAILS

Measurement protocol. For each omitted Macro block we add back its $K16$ tiles one by one and measure the recovered error ΔE, defined as the reduction in relative L2 distance to the dense output. The relative L2 distance is computed per $Q16$ query row across its 16 tokens and d channels, normalized by the dense output norm of that row. Blocks whose full-Macro benefit is zero or negative are excluded; they account for less than 3% of all omitted Macros. Section 4.1 reports three add-back experiments at different scopes and granularities. This section provides the full numerical tables for all three.

All omitted Macros (per-block mean). The first experiment averages over every omitted Macro block. Each value is the fraction of the full-Macro add-back benefit recovered by adding back the Top-k $K16$ tiles, measured in the early $Q128 \times K96$ configuration.

Table 5: Add-back recovery averaged over all omitted Macro blocks (early $Q128 \times K96$ configuration). Each row shows the fraction of the full-Macro benefit recovered by Top-k $K16$ tiles selected by the dense-mass oracle, the mean-pooled proxy, or a random baseline.

Selection	Top-1 $K16$	Top-2 $K16$	Top-4 $K16$	Full Macro
Dense oracle	51.5%	73.9%	90.4%	100%
Mean proxy	35.0%	54.5%	78.0%	100%
Random	13.2%	26.2%	51.5%	100%

$Q16 \times K16$ **row-regions (all).** At the finer $Q16 \times K16$ granularity, we measure add-back recovery over all 1,285,632 row-regions where a full $K96$ add-back reduces error (96.9% of all regions; the remaining 3.1% still contribute to the output-error mean but are excluded from the ratio). Top-k tiles are added back jointly and softmax is recomputed. Random values are the exact mean over all same-size combinations per region.

Table 6: Add-back recovery at $Q16 \times K16$ granularity, averaged over all row-regions where full $K96$ add-back reduces error. Concentration is similar to the Macro-level average, confirming the within-block skew is not an artifact of the coarser $K16$-column analysis.

Selection	Top-1 $K16$	Top-2 $K16$	Top-4 $K16$	Full $K96$ (6 tiles)
Dense oracle	48.0%	67.2%	89.3%	100%
Mean proxy	31.3%	52.5%	82.4%	100%
Random	16.6%	33.2%	66.6%	100%

Severe omissions only. The second experiment restricts to the highest-severity omitted $Q16$ rows on HyVideo at the main operating density $\rho_C = 0.20$. Severity is defined per $Q16$ row as the maximum ΔE of any omitted $K96$ Macro in that row, using the relative-L2 definition above. We take the highest-severity rows and, for each row, select the omitted Macro with the largest full-block benefit. Recovery is measured among its six $K16$ tiles. Random values are the exact mean over all same-size combinations. Blocks with non-positive full-Macro benefit are excluded, as in the all-Macro average.

Table 7: Add-back recovery inside the most severe omissions (highest-severity $Q16$ rows, highest-benefit omitted $K96$ Macro each, $\rho_C = 0.20$). Concentration is substantially sharper than the all-Macro average.

Selection	Top-1 $K16$	Top-2 $K16$	Top-4 $K16$	Full $K96$ (6 tiles)
Dense oracle	93.5%	99.4%	99.9%	100%
Mean proxy	55.4%	72.9%	92.9%	100%
Random	18.4%	35.9%	68.8%	100%

A.5 Recall across Core Ratios

Section 4.2 reports the selection gap between the mean-pooled proxy, the sampled-LSE scorer, and the dense-mass oracle at the main operating density $\rho_C = 0.20$. Table 8 extends the comparison to three Core ratios, $\rho_C \in \{0.25, 0.20, 0.16\}$. The metric is Recall@$k$, the percentage of high-error Q16 rows where the scorer ranks the highest-addback omitted $K16$ tile of that row in its top k, using the three-video K96 Core snapshot and the same add-back measurement protocol.

Sampled-LSE consistently outperforms the mean proxy at every ratio and every k. The oracle improves monotonically as the Core shrinks, while the mean proxy shows no consistent trend, so the gap between cheap proxy and true mass grows at sparser densities. Sampled-LSE tracks the oracle much more closely and captures most of the recoverable tail at every evaluated density.

Table 8: Recall@k over the high-error Q16 rows (each row's highest-addback omitted $K16$ tile ranked within that row's top k), for three Core ratios ρ_C. Sampled-LSE consistently outperforms the mean proxy, and the gap between mean proxy and oracle widens at sparser densities.

ρ_C	Selection	Top-1	Top-5	Top-10	Top-20
0.25	Mean proxy K16	2.3%	5.8%	6.9%	9.2%
	Sampled-LSE	55.5%	76.9%	79.2%	81.5%
	Dense mass oracle	66.5%	83.2%	86.7%	89.0%
0.20	Mean proxy K16	1.2%	5.2%	6.9%	12.1%
	Sampled-LSE	54.9%	83.2%	87.3%	89.0%
	Dense mass oracle	72.3%	90.8%	94.2%	95.4%
0.16	Mean proxy K16	2.9%	5.2%	7.5%	9.8%
	Sampled-LSE	57.8%	89.6%	94.2%	95.4%
	Dense mass oracle	76.3%	97.7%	99.4%	99.4%

A.6 Stability of Risk Ranking

The main text states that the (layer, head) risk ranking is stable. This section provides supporting measurements across both videos and Core ratios.

Cross-video stability. The per-(layer, head) risk ranking transfers well across videos. Full-rank Spearman correlation is $\rho = 0.71$ (bootstrapped range 0.65–0.76), and top-set overlap grows with set size: Top-50 $= 0.78$, Top-100 $= 0.82$, Top-200 $= 0.84$, Top-300 $= 0.85$, Top-600 $= 0.86$. A single per-model calibration therefore suffices.

Cross-density stability. We compute the per-(layer, head) risk score r_{lh} (defined in Appendix A.11) on the HyVideo calibration set at five densities: $\rho_C \in \{0.10, 0.15, 0.20, 0.25, 0.30\}$. Table 9 reports pairwise correlations computed over the top 50% of heads ranked at the sparser density of each pair. The truncated values are higher because the bottom half of the ranking is dominated by near-zero noise. Adjacent pairs correlate at 0.86–0.90, and the most extreme pair (0.10 vs 0.30) still reaches 0.67. The correlation drops as the density gap widens, which is expected: ranks near the bottom of the list are dominated by near-zero noise.

Table 9: Pairwise Spearman rank correlation of r_{lh} across Core ratios ρ_C, computed over the top 50% of heads ranked at the sparser density of each pair. The ranking is most stable between adjacent densities and degrades gracefully as the density gap widens.

ρ_C	0.10	0.15	0.20	0.25	0.30
0.10	1.00	0.87	0.82	0.74	0.67
0.15	0.87	1.00	0.90	0.80	0.71
0.20	0.82	0.90	1.00	0.86	0.75
0.25	0.74	0.80	0.86	1.00	0.87
0.30	0.67	0.71	0.75	0.87	1.00

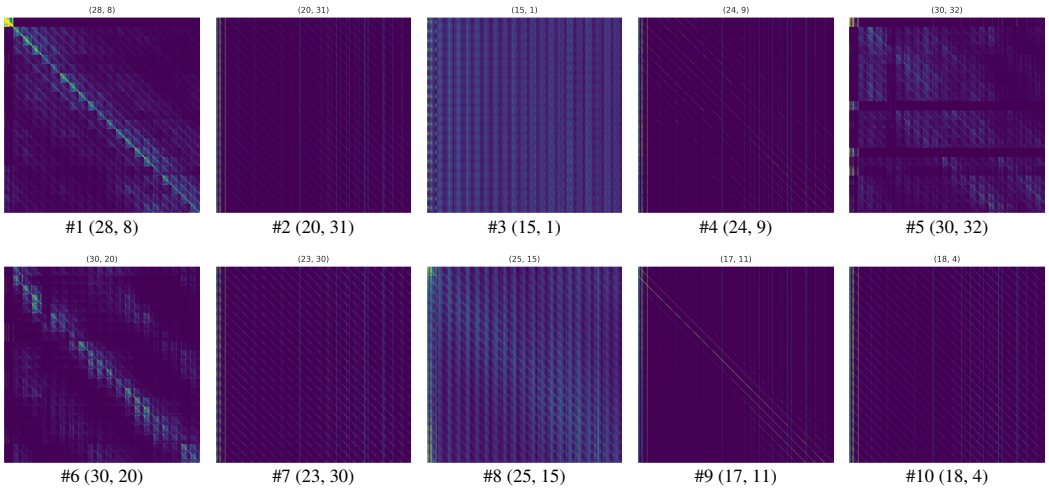


Figure 6: Dense attention heatmaps of the ten highest-risk (layer, head) positions on Wan2.1-14B at step 36. Heads are ranked by K16 add-back recovery risk, with risk sums shown in each sub-title. Per-head color limits are clipped at the 99.8-th percentile for visibility. The maps use the original token order and 16×16 display bins. High-risk heads do not follow one fixed geometry. They include diagonal bands, vertical key stripes, and mixed local structures.

For the method, what matters is not the full ranking but the overlap of the *Top-*N_{risk} set actually used for Micro repair. With $N_{risk} = 600$, adjacent densities overlap at 0.90 on average, and the most dissimilar pair ($\rho_C = 0.10$ vs $\rho_C = 0.30$) still overlaps at 0.76. Mirroring the pattern observed across videos, the noisy tail below the top few hundred positions contributes little to the actual repair set, so calibrating at the target operating density is sufficient; the risk set does not shift enough to warrant re-calibration per density.

A.7 Dense Attention Patterns of Top-Risk Heads

The main text shows that omission risk is concentrated at the layer and head level. We next ask what dense attention looks like inside these high-risk heads. Figures 6 and 7 visualize the ten highest-risk heads of Wan2.1-14B and HyVideo-13B at representative denoising steps. Heads are selected only by the offline $K16$ add-back recovery risk. The heatmaps are therefore a qualitative view of the structures associated with the measured risk, not an additional selection criterion.

Three observations are consistent across the two models. First, high-risk heads are structurally heterogeneous. Some are dominated by a narrow diagonal or repeated offset diagonals, while others contain vertical key stripes or compact off-diagonal clusters. Thus, a single hand-crafted sparse pattern would not cover all high-risk cases. Second, many heads combine a broad, regular body with much thinner structures. Averaging over a large Macro can weaken these narrow signals even when they remain clear at the finer display scale. This provides a visual explanation for the spike dilution measured in §4.2. The figures alone do not identify which tiles are omitted by a particular Core mask; that link is established by the add-back recovery risk used to select these heads. Third, the same pattern families recur in both models, but their high-risk layer–head identities differ. This supports the division used by HyMoR-Attn: calibrate vulnerable positions once for each model, then select Microtiles online from the current input without imposing a universal geometric prior.

A.8 Micro Residual Placement on a High-Risk Head

The previous section shows that high-risk heads exhibit diagonal bands, vertical hub stripes, and local clusters, all of which leak mass across Macro block boundaries. We now examine one representative head in detail to answer a direct question: when the Micro residual adds tiles back, where exactly do they go?

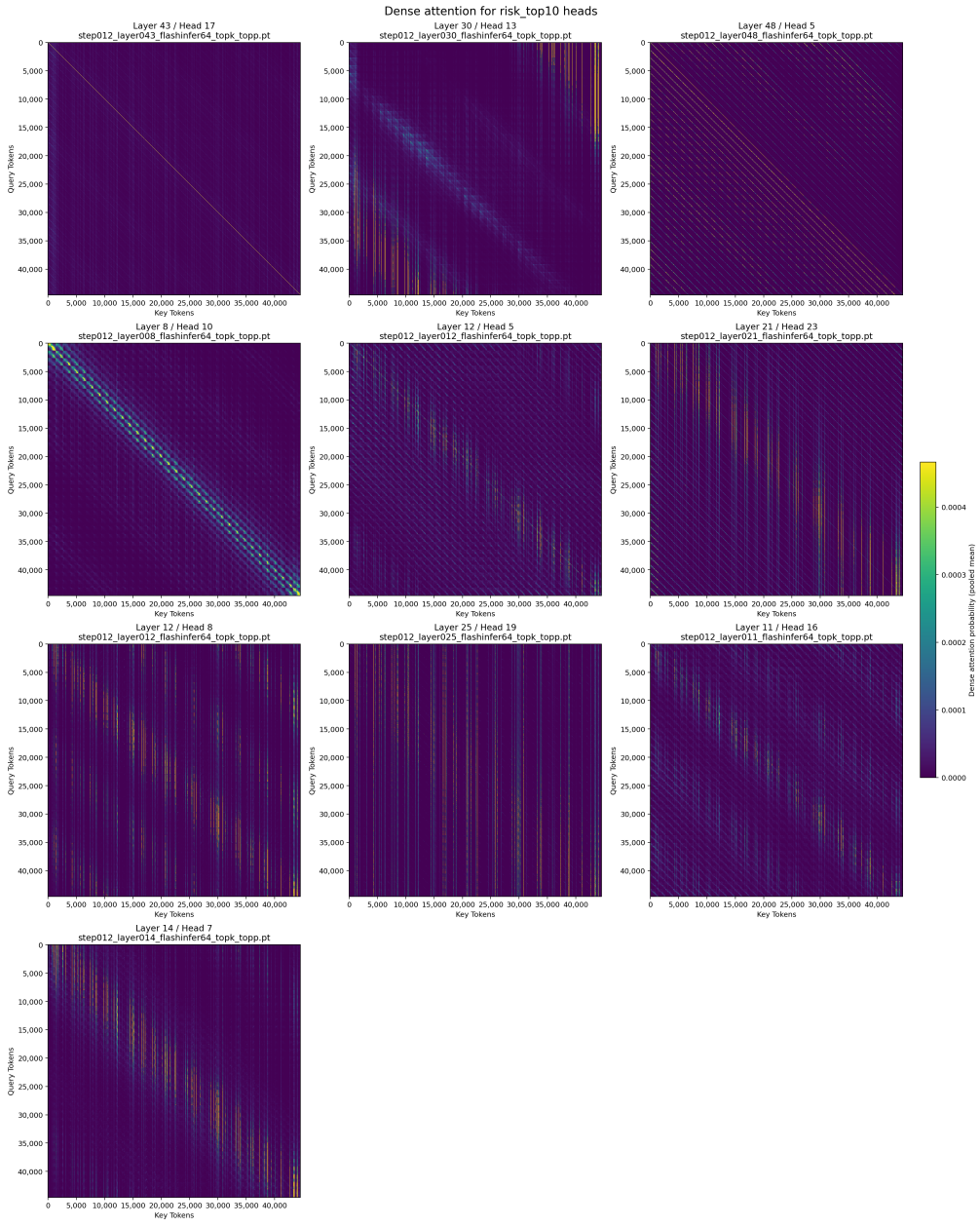


Figure 7: Dense attention heatmaps of the ten highest-risk (layer, head) positions on HyVideo-13B at step 12. Attention probabilities are mean-pooled only for display. Diagonal bands, repeated offset diagonals, vertical key stripes, and local clusters all occur among the top-risk heads. The pattern families recur across models, while the responsible layer–head positions differ.

Figure 8 visualizes a single head (HunyuanVideo Layer 12 / Head 5 / Step 12) in three layers. The background color shows the exact dense attention truth at Q16/K16 resolution. White outlines mark Macro blocks retained by the Core. Red outlines mark the Micro residual tiles selected by the complement Top-p scorer. The right panel zooms in around the highest-mass residual tile (marked with a yellow star) to show the fine structure.

Quantitatively, the Macro Core covers 63.81% of the total dense attention mass on this head. The Micro residual recovers an additional 17.88%, which is 49.42% of the mass that the Core leaves uncovered. The average true attention mass of selected Micro tiles is 62.1 times higher than that

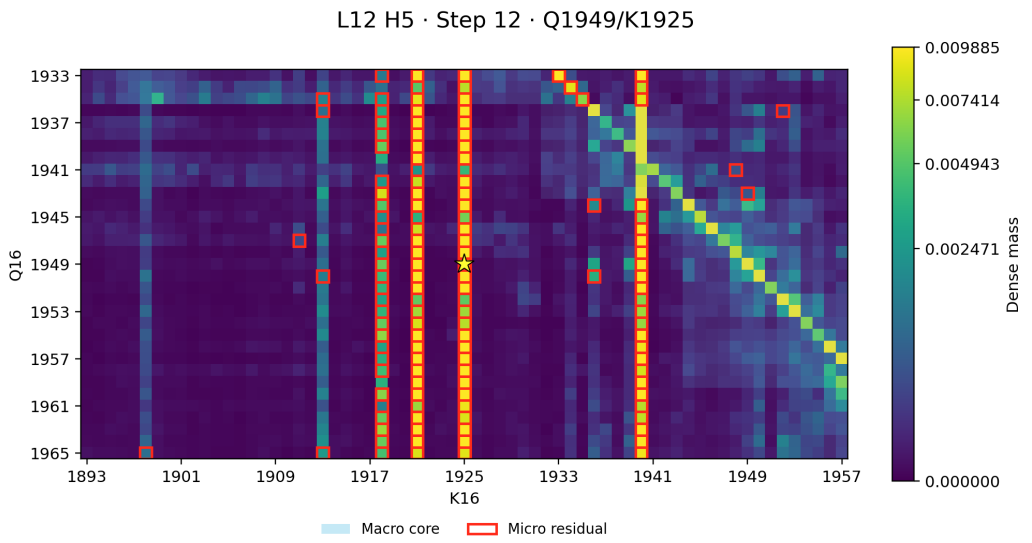


Figure 8: Dense attention with Macro Core and Micro residual overlay on a representative high-risk head (HunyuanVideo Layer 12 / Head 5 / Step 12). The background heatmap shows exact dense attention probabilities at $Q16 \times K16$ resolution. White outlines mark Macro blocks retained by the Core; red outlines mark Micro residual tiles selected by the complement Top-p scorer. The Micro residual concentrates at diagonal-band edges, vertical hub stripes, and local bright spots that the Macro Core leaves uncovered.

of unselected non-Core tiles, and 95.18% of the residual tiles fall in the top 5% of non-Core tiles ranked by exact dense mass. The residual is therefore not adding mass uniformly. It concentrates on the small fraction of omitted tiles that actually matter.

Qualitatively, three placement patterns are visible, and each corresponds to one of the structural families identified in the previous section.

Diagonal-band edge fill-in. The zoomed panel on the right is the clearest example. A diagonal band of high attention runs from upper-left to lower-right. The Macro Core (white boxes) covers the center of the band, but its block-aligned edges cut diagonally across the band. The Micro residual (red outlines) picks up tiles along both sides of this cut: tiles that belong to the band but sit just outside the nearest retained Macro block. This is exactly the edge-leakage failure described for diagonal-band heads: a coarse block can cover the band center without capturing its full width.

Vertical hub-stripe capture. Several near-vertical bright columns are visible in the global view and more clearly in the zoom. These are hub key positions: a single K16 column receives elevated attention from many Q16 rows. When such a column falls outside the Core, every query row loses signal from that reference point. The Micro residual selects these columns row by row, recovering the hub signal without paying the cost of retaining the entire Macro block that contains it. This matches the vertical-stripe failure mode: mean-pooled scores spread the hub value across 48 Microtiles and under-rate the whole block, but the peak-preserving scorer finds the high-value columns inside it.

Local cluster hit. Scattered bright spots appear away from the main diagonal, particularly in the upper half of the global view. These are content-specific associations between distant regions of the video. Each such spot is small enough to fit inside a single omitted Macro block, and its mass is diluted to near-zero when averaged over the 48 Microtiles of that block. The Micro residual selects these spots individually, recovering content-specific signal that coarse selection would discard entirely.

Taken together, the visual and quantitative evidence shows that the Micro residual acts as a targeted repair, not a uniform densification. It finds the same structural features that the previous section

identified as high-risk, and it places its budget exactly on the tiles where Macro selection leaks the most mass. This validates the two design choices at the core of the method: offline risk ranking locates the right heads to repair, and the online peak-preserving scorer locates the right tiles within those heads.

A.8.1 Two Modes of High-Risk Heads

The single-head example above is representative of one type of high-risk head, but high-risk positions are structurally diverse. We examine three additional heads that all rank in the top of the offline risk ranking but exhibit very different patterns of attention mass and Micro enrichment. Comparing them reveals why omission risk must be measured by output-error recovery rather than by attention-mass statistics, and why the offline calibration in Appendix A.11 uses a high-error-row ΔE sum instead of a pooled mass proxy.

Table 10 summarizes four high-risk heads on HunyuanVideo at step 12, ordered by their offline risk rank. Two metrics characterize residual placement quality: the enrichment ratio (average true mass of selected residual tiles divided by average true mass of unselected non-Core tiles) and the top-5% hit rate (fraction of residual tiles that fall in the top 5% of non-Core tiles by exact dense mass).

Table 10: Four high-risk heads on HunyuanVideo at step 12, ordered by offline risk rank (sum of best $K16$ add-back recovery over high-error $Q16$ rows). All four are in the top tier of the 1440-head ranking, yet their enrichment ratios span more than an order of magnitude. Low enrichment does not imply low risk—it can indicate a different failure mode where omissions are extreme but concentrated in few rows.

Head	Risk rank	Enrichment ratio	Top-5% hit rate
Layer 43 / Head 17	#1	6.74×	53.3%
Layer 30 / Head 13	–	139.53×	99.3%
Layer 48 / Head 5	–	122.73×	88.4%
Layer 12 / Head 5	–	62.1×	95.2%

The enrichment gradient spans more than an order of magnitude, and it does *not* align with the risk ranking. The highest-risk head (Layer 43 / Head 17) has the lowest enrichment (6.74×), while vertical-hub heads like Layer 30 / Head 13 reach nearly 140×. This mismatch is not an error in either metric; it reflects two distinct failure modes of coarse block selection, each with a different relationship between attention mass and output-error recovery.

Figures 9, 10, and 11 show the three additional heads in the same format as Figure 8. Each head exhibits a different dominant structure, and the following paragraphs characterize each mode.

Layer 30 / Head 13: the broad-distribution mode. This head is the strongest example of the vertical-stripe failure mode. Several bright vertical key columns run across nearly the full query range, and most of these columns fall outside the Macro Core. The Micro residual selects these columns with near-perfect precision: 99.3% of selected tiles are in the top 5% of non-Core true mass. The zoomed panel makes the mechanism clear: each selected tile sits directly on a bright vertical stripe, one tile per query row, recovering the hub signal without the cost of retaining the surrounding Macro blocks. Risk in this head is *broad*: many Q16 rows each lose a moderate amount, and the scorer finds the lost mass easily because the hub columns are conspicuous even at coarse resolution.

Layer 48 / Head 5: diagonal plus isolated clusters. Here the dominant structure is a diagonal band, but several compact off-diagonal bright spots also appear. The Macro Core retains blocks along the diagonal center, leaving both the band edges and the isolated clusters uncovered. The Micro residual picks up both: it follows the diagonal edge to fill band-width leakage, and it separately selects the isolated off-diagonal spots. The enrichment (122.73×) is nearly as high as on the vertical-hub head, because the off-diagonal spots are especially concentrated and especially well separated from the Core.

Layer 43 / Head 17: the concentrated mode. This head is the most severe case in the entire model, ranking #1 in total add-back recovery across all 1440 (layer, head) positions. Its risk profile

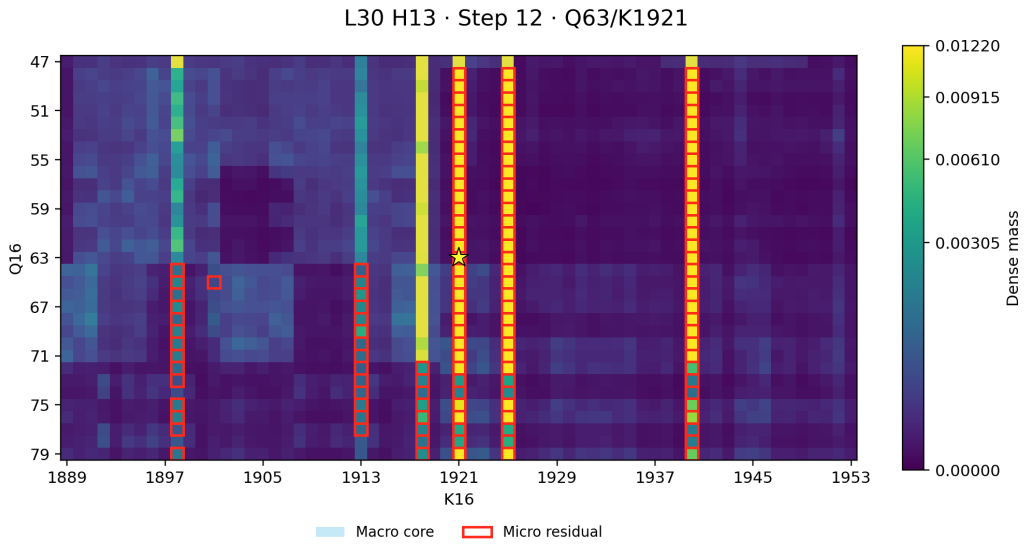


Figure 9: Dense attention with Macro Core and Micro residual overlay on Layer 30 / Head 13 (step 12). This head is dominated by vertical hub stripes: a few key columns receive strong attention from nearly all query rows. The Micro residual picks these high-value columns row by row with 139.53× enrichment and 99.3% top-5% hit rate, nearly perfectly recovering the omitted hub signal.

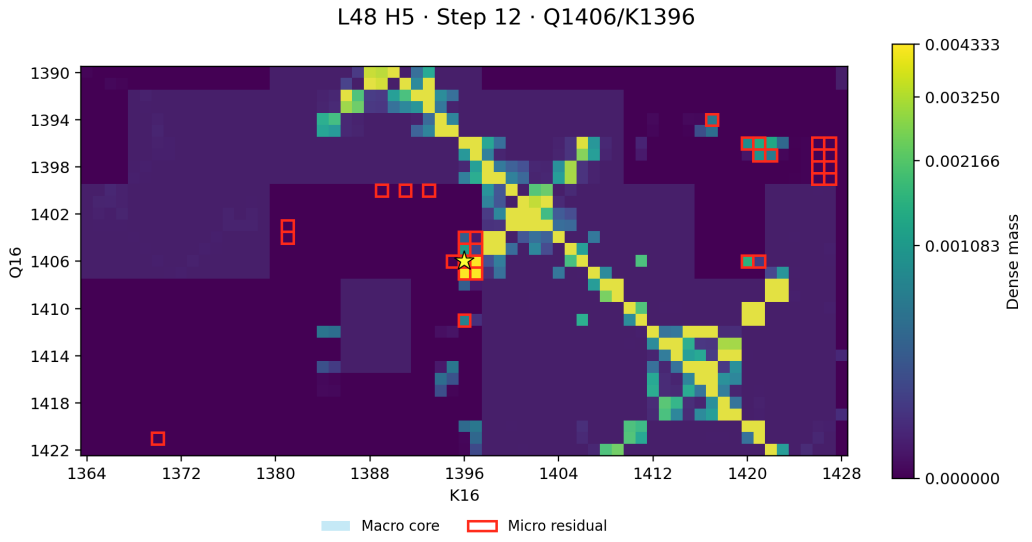


Figure 10: Dense attention with Macro Core and Micro residual overlay on Layer 48 / Head 5 (step 12). This head combines a diagonal band with off-diagonal bright clusters. The Micro residual concentrates at the diagonal edge and on isolated off-diagonal spots, achieving 122.73× enrichment and 88.4% top-5% hit rate.

is qualitatively different from the vertical-hub and diagonal-edge cases. Instead of being spread across many rows, the risk is highly concentrated: the 64 highest-error Q16 rows in Layer 43 contain 40 where Head 17 is the dominant source of error (62.5%). In all 40 of those rows, the single best omitted K16 tile lies exactly on the diagonal (best $K16$ index == query $Q16$ index), indicating a strong local self-alignment pattern. The median recovery from restoring that one diagonal K16 is approximately 100% of the recovery from restoring the entire omitted K96 Macro block—one Microtile contributes nearly the full benefit, and the other five tiles in the block contribute almost

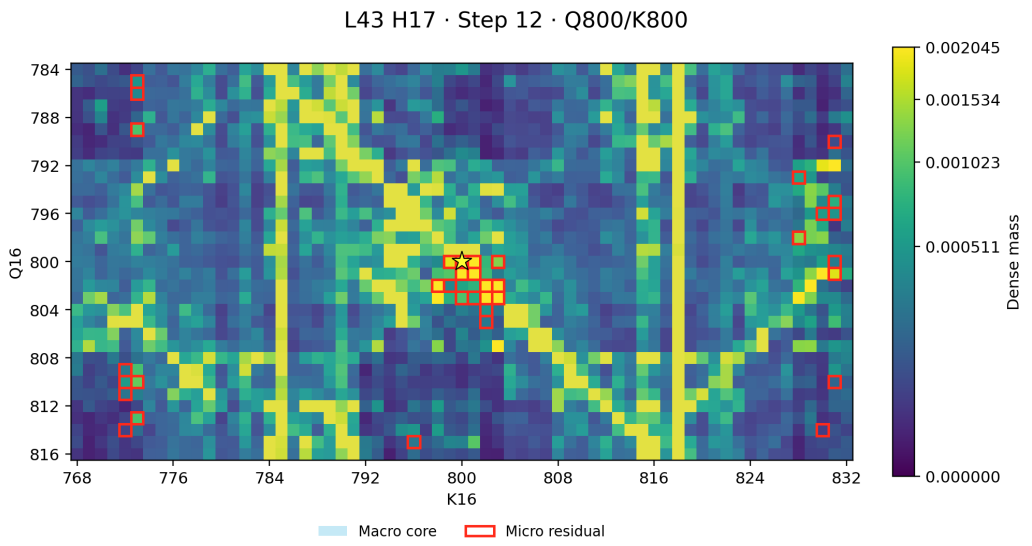


Figure 11: Dense attention with Macro Core and Micro residual overlay on Layer 43 / Head 17 (step 12). This head has the highest overall omission risk of all 1440 heads, but the lowest enrichment ratio among the four shown. Risk concentrates on the diagonal: among its 64 high-error Q16 rows, Head 17 contributes 40 of them (62.5%), and in every one of those 40 rows the best-recovery K16 falls exactly on the diagonal (best $K16$ == query $Q16$). A single diagonal K16 tile recovers nearly the full benefit of the entire omitted Macro block. The low overall enrichment reflects the fact that this extreme recovery is confined to a small number of rows and is diluted when averaged across the full Q16 grid.

nothing. This is the canonical Omitted-Micro Problem: a coarse block is discarded because its pooled score is low, yet a single tile inside it carries essentially all of the block's recoverable value.

The low enrichment ratio (6.74×) is consistent with this picture. Enrichment averages over all Q16 rows in the head, and for this head the vast majority of rows have little recoverable mass outside the Core. Only a small set of diagonal-miss rows produce extreme recovery values, and those rows are too few to move the head-wide average. Measuring risk by pooled attention mass or average enrichment would therefore *under*-rank this head substantially, even though it is the single largest contributor to total output-error loss from omitted Microtiles.

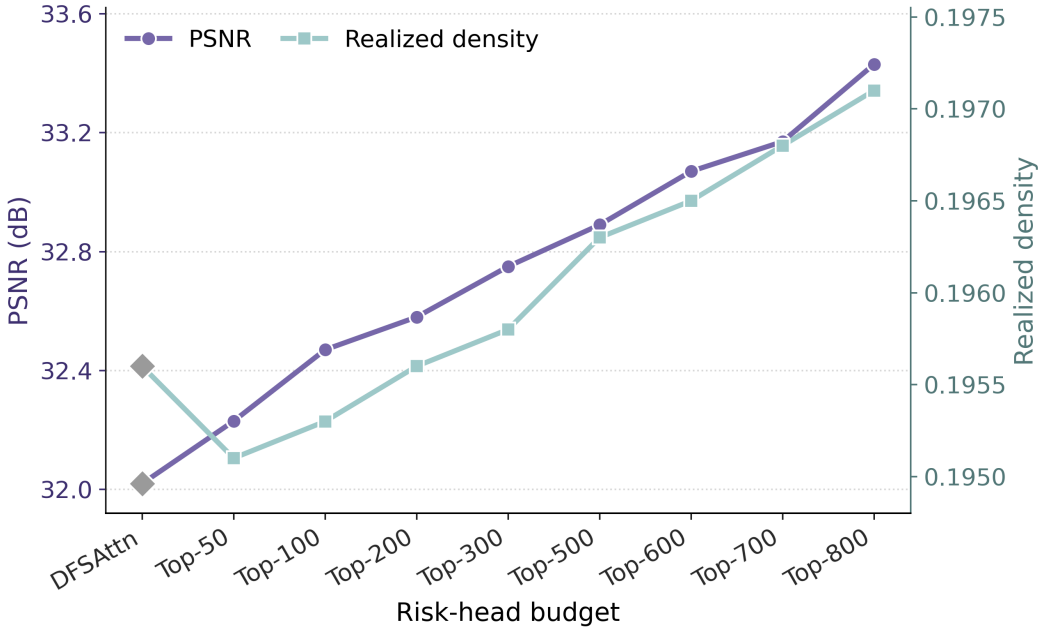
Why risk is defined by output-error recovery, not by mass. The contrast between the broad-distribution mode (Layer 30 / Head 13) and the concentrated mode (Layer 43 / Head 17) makes a simple but important point: omission risk and enrichment ratio are not the same thing. Enrichment measures how efficiently the scorer finds high-attention-mass tiles in the complement, averaged over the full head. Risk measures total output-error recovery across the high-error tail of rows. A head can be very high-risk while having low average enrichment, if its risk concentrates in a small number of rows with extreme per-row recovery.

This is precisely why the offline calibration uses sum of ΔE over high-error rows as the primary ranking key (Appendix A.11), rather than a mass-based or enrichment-based proxy. Mass-based proxies dilute extreme but sparse recoverable value across the many low-value interactions that surround it, and heads like Layer 43 / Head 17 would be under-ranked as a result. The output-error-based risk definition correctly surfaces both modes of high-risk head, and the Micro residual budget is allocated to both. The two-mode structure also explains why the method needs a dedicated peak-preserving scorer (§5.1.1) rather than relying on mean-pooled scores: concentrated extreme values like those in Layer 43 / Head 17 are exactly the pattern that mean pooling erases.

A.9 Risk-Head Budget Sweep

Table 11 examines whether HyMoR-Attn depends on a carefully tuned risk-head budget. It does not. Every configuration improves PSNR over DFSAttn at comparable density. As the budget increases from Top-50 to Top-800, PSNR improves monotonically from 32.23 to 33.43 dB, while realized density changes by only 0.0020. This shows that the quality gains come from extending Micro repair to more risk-ranked heads, not from substantially densifying attention. Top-300 improves PSNR by 0.73 dB over DFSAttn, while Top-600 improves PSNR by 1.05 dB. These are representative speed-oriented and quality-oriented operating points on the same broad trade-off curve, not isolated optima obtained through careful tuning. The risk-head budget therefore serves as a direct and robust quality–cost control knob, not a hyperparameter burden.

Table 11: Risk-head count sweep on HyVideo t2v 720p (VBench 10% subset). The Micro residual is applied to the top-N risk-ranked (layer, head) positions; all other settings are fixed. DFSAttn is the block-sparse baseline on the same subset.



A.10 Detailed Ablation Analysis

The Micro residual is the largest single contributor. Removing it drops PSNR from 28.89 to 28.32 dB, a loss of 0.57 dB, at only 0.002 lower density. The Macro-only baseline reaches 28.32 dB (Table 3), close to DFSAttn's 28.23 dB in Table 2, as expected: both perform coarse Top-k selection. This gap directly supports the paper's central claim: coarse selection has a quality ceiling set by the high-value support it discards, and repairing omitted Micro interactions removes that ceiling. The gain is therefore not from denser attention or more compute in general—it comes from recovering specific fine-grained interactions that coarse scores miss.

Content-dependent tile selection matters, but the right heads alone recover some value. When we keep the same Top-600 risk heads and the same per-row tile budget but randomize *which* Micrtiles are selected within each row, density stays identical (0.1984) but PSNR drops by 0.34 dB. Importantly, this random-tile configuration still sits 0.23 dB above the no-Micro baseline (28.55 vs 28.32). That residual gap is informative: once the budget is placed on the right heads, even random tile selection recovers roughly 40% of the full Micro benefit, simply because high-risk heads contain more high-value tiles overall and random sampling hits some of them. The remaining 0.34 dB gap shows that content-dependent coordinate selection is responsible for the other 60%.

Stable offline risk ranking is necessary to place the budget. Replacing the calibrated Top-600 risk set with 600 uniformly random (layer, head) positions—keeping the same sampled-LSE scorer and the same per-row tile budget—costs 0.25 dB PSNR. Density is unchanged. This rules out the possibility that any 600 heads would suffice, and isolates the offline ranking as the mechanism that concentrates the residual budget where omissions are most severe. The 0.25 dB gap is smaller than the 0.34 dB tile-selection gap, meaning that once a broad set of vulnerable heads is identified (offline), the online scorer still has substantial work to do finding the right coordinates within them.

Peak-preserving scoring adds measurable value on top of correct placement. Replacing the sampled-LSE scorer with the mean-pooled proxy, at the same Top-600 risk set and the same per-row tile budget, costs 0.17 dB PSNR. Density is unchanged. This is the smallest component, consistent with the diagnosis in §4.2: mean pooling is reasonably reliable within a single block (hence it still recovers most of the benefit once the right heads are targeted), but it fails at cross-block ranking within a row, so it mis-prioritizes tiles from different Macro parents. The 0.17 dB gap quantifies precisely how much the peak-preserving scorer recovers beyond what a pooled proxy can do, once the offline ranking has narrowed the search to the right heads.

Summary. The ablation pattern decomposes cleanly into a hierarchy of decisions. Offline calibration decides *where* to invest repair budget (which heads); online selection decides *which* tiles to keep within those positions and *how* to score them. The two levels are complementary and both necessary: removing either the head ranking or the tile ranking causes a substantial drop. Critically, none of the gain comes from increasing density—all four ablation rows operate at essentially the same realized density (0.196–0.198), so the quality differences reflect how the budget is allocated, not how large it is.

A.11 Implementation Details and Hyperparameters

Risk-head budget. N_{risk} is a quality–speed parameter. Top-300 is the speed-oriented operating point and Top-600 is the quality-oriented point. Both are reported in the main results. The full sweep from Top-50 to Top-800 is in Appendix A.9.

Complement mass target. $p_{total} = 0.90$ is the complement Top-p mass target. A sensitivity sweep is in Appendix Table 4, where PSNR saturates above 0.85.

Offline omission-risk calibration. The risk set used by the Micro residual is obtained by a per-model offline calibration on a held-out video set. The goal is to rank (layer, head) positions by how much output quality is lost to omitted fine-grained support, so that the repair budget can be concentrated where omissions hurt the most.

Concretely, for each high-error $Q16$ row q in head (l, h) (rows where the Macro core output deviates most from dense attention), we find the single omitted $K16$ tile with the largest add-back recovery

$$\Delta E^*_{lhq} = \max_{j \notin C} \Delta E_{lhqj}, \tag{7}$$

where ΔE_{lhqj} is the reduction in relative L2 distance to the dense output from adding back $K16$ tile j in $Q16$ row q, following the definition in §4.1. Heads are ranked by the *total* recoverable error summed over all high-error rows—the total amount of output quality that the Macro core leaks to omitted Microtiles in that head. The top N_{risk} positions form the risk set; the scorer, complement Top-p, and Residual execution run only on these positions, and every other position relies on the Macro core alone.

The calibration is model-specific and runs independently for each model. The resulting ranking is stable across videos and seeds (see §4.3), so it is calibrated once per model and reused at inference. N_{risk} is a quality–speed parameter, not a fixed constant.