Beyond Coarse Blocks: Fine-Grained Sparse Attention for Video Diffusion

Satya Chivatam
chivatam@amazon.com

Abstract

Video Sparse Attention (VSA) accelerates video diffusion by applying exact attention to selected regular blocks, but its 64-token support granularity can discard locally important interactions. We introduce a training-free refinement that scores 8-token sub-blocks while preserving the selected valid-token count of the released 80%-sparse VSA configuration for every query block. Across a matched, fixed-seed evaluation of all 946 VBench prompts and 16 dimensions, fine-grained support improves normalized Total by +1.38 points to 81.59 and improves 11 of 16 dimensions at the same sparsity. In an isolated same-prompt run, the current packed executor adds +4.27% end-to-end latency relative to coarse VSA. Profiling localizes its 2.48× attention-path gap to the unfused execution of short, irregular segments, leaving custom fusion and scheduling as a well-scoped next systems step.

1 Introduction

Diffusion Transformers (DiTs) make attention a central cost in generative inference [Peebles and Xie, 2023]. Video models magnify that cost because each denoising step processes a long spatiotemporal token sequence. Recent systems therefore exploit local, structured, or learned sparsity [Zhang et al., 2025c, Xi et al., 2025, Zhang et al., 2025b]. Efficient masks, however, must satisfy two objectives: retain interactions that determine video quality and expose a regular structure that GPUs can execute efficiently.

VSA learns a two-stage approximation: pooled coarse attention identifies important regions, and exact attention processes selected 64-token key/value blocks [Zhang et al., 2025b]. We call its released 80%-sparse operating point *coarse VSA*. Regular blocks provide efficient memory access, but they also couple hardware regularity to support granularity. A selected block may contain only a few useful tokens, while a discarded block may still contain a small object, color cue, or motion detail.

We ask whether finer support improves quality without spending more attention. Our method, *fine-grained VSA*, preserves the trained checkpoint, query blocks, coarse residual path, and selected valid-token count, but redistributes exact attention over 8-token sub-blocks. The comparison therefore changes where the sparse budget is spent, not its size.

The experiments establish three findings. First, fine-grained VSA improves the full-prompt normalized VBench Total by +1.38 points over coarse VSA and improves 11/16 dimensions. Second, a frozen diagnostic repairs 17 of 24 previously identified coarse-VSA failures. Third, packed execution limits the isolated same-prompt end-to-end overhead to +4.27%, but the irregular attention path remains slower. The contribution is thus a controlled quality result at a fixed sparse budget and a precise systems target for hardware-efficient fine-grained sparsity.

The 10th Machine Learning for Systems Workshop at NeurIPS 2026, Sydney, Australia

2 Methods

Coarse VSA and the fixed budget. Let q index a 64-token query block. Coarse VSA selects a set of 64-token key/value blocks P_q. Because edge blocks may contain padding, we record the selected valid-token total

$$T_q = \sum_{p \in P_q} n_p, \tag{1}$$

where n_p is the valid size of block p. This definition makes the baseline budget explicit even when sequence edges are padded.

Fine-grained support refinement. Fine-grained VSA divides the key/value sequence into groups of eight tokens. It mean-pools each group, scores it against the existing coarse query descriptor, masks padding, and selects groups whose valid-token total is exactly T_q. In the tested geometry, the nominal budget is 8,000 key/value tokens per query block; the maximum measured budget error is zero. Exact attention is computed on the selected groups and combined with the unchanged gated coarse residual. No retraining or additional predictor is introduced. Algorithm 1 summarizes the inference procedure.

Algorithm 1 Fine-grained VSA at a fixed valid-token budget

Require: coarse support $\mathcal{P}_q$ and query descriptor $\bar{q}$
Require: key/value sequence (K, V) and coarse residual r_q
$T_q \leftarrow \sum_{p \in \mathcal{P}_q} n_p$
$\mathcal{C} \leftarrow \textsc{PartitionValid}(K, V, 8)$
for all $c_j \in \mathcal{C}$ **do**
  $s_j \leftarrow \langle \bar{q}, \mathrm{mean}(K_{c_j}) \rangle$
end for
$\mathcal{S}_q \leftarrow \textsc{SelectBudget}(\{(c_j, s_j, m_j)\}, T_q)$
return $\textsc{ExactAttention}(q, K_{\mathcal{S}_q}, V_{\mathcal{S}_q}) + r_q$

Packed execution. Naively materializing many short gathers makes metadata handling, address generation, and under-filled matrix operations dominate runtime. We therefore coalesce adjacent selected sub-blocks into wider segments and perform online softmax with FP32 partial accumulation. Packing changes only the execution representation; the selected support and generated-video policy remain fixed (Figure 1).

3 Experiments

Setup. We evaluate a FastVideo implementation of a 1.3B Wan2.1 text-to-video model [Wan Team et al., 2025, Zhang et al., 2025c] at 480×832, 81 frames, 16 fps, three denoising steps, guidance 3.0, and seed 1024. We compare dense BF16 attention, coarse VSA with 64-token blocks at 80% sparsity, and fine-grained VSA with 8-token sub-blocks. Generation runs on eight NVIDIA B200 GPUs. All methods share prompts, checkpoint, seed, scheduler, and decoder. Attention and DiT values are

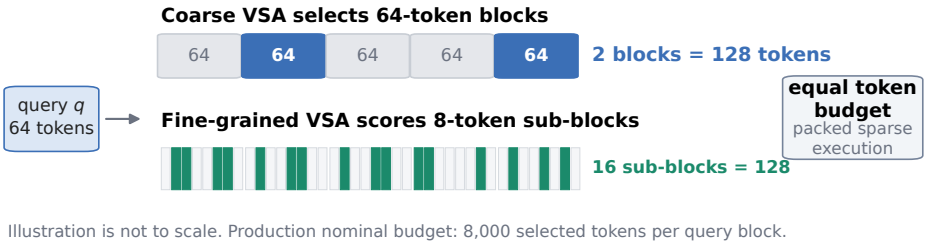


Figure 1: Fine-grained VSA changes support granularity, not the selected valid-token budget. Adjacent selected sub-blocks are packed into wider execution segments; the diagram uses a small illustrative budget.

Table 1: Matched full-prompt evaluation with a fixed seed shared by all methods. Scores use VBench's published normalization inside the FastVideo harness. Fine-grained VSA has zero selected-token budget error relative to coarse VSA; its measured sparsity reflects valid edge tokens. Attention and DiT are full-prompt medians. E2E is one isolated same-prompt run per method after warm-up on the same B200.

Method	Sparsity (%)	Quality ↑	Semantic ↑	Total ↑	Attn. (ms) ↓	DiT / E2E (s) ↓
Dense BF16	0.00	85.41	63.02	80.93	591.71	1.53 / 12.48
Coarse VSA (64)	80.00	84.61	62.62	80.22	**274.82**	**1.44 / 9.33**
Fine VSA (8)	79.97	**85.69**	**65.19**	**81.59**	680.80	1.82 / 9.73

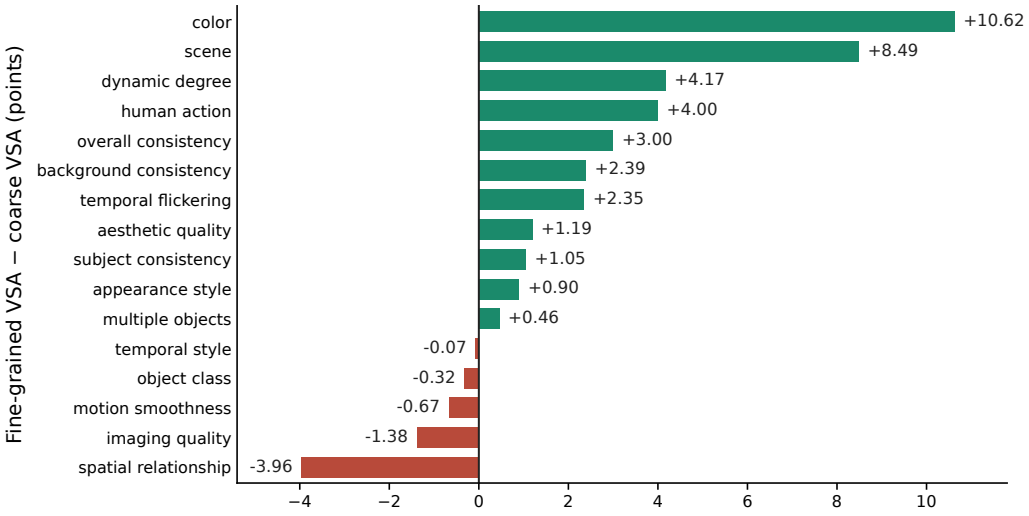


Figure 2: Fine-grained VSA minus coarse VSA for every normalized VBench dimension. 11 of 16 changes are positive; the aggregate Total change is +1.38 points. Exact three-method values are reported in Appendix Table 2.

full-prompt medians. For E2E latency, we separately load each method in a fresh process on the same B200, warm it once, and time one saved generation of "a bird and a cat" with the same seed and video settings.

Quality is measured with the integrated VBench 1.0 harness [Huang et al., 2024]. We generate one matched video for every one of the 946 benchmark prompts and each method, totaling 2,838 videos across 16 dimensions, with seed 1024 held fixed across methods. This paired design compares attention methods along identical stochastic trajectories. A similar VBench setup appears in DraftAttention [Shen et al., 2025a]; its public evaluation artifact fixes seed 42 [Shen et al., 2025b]. We use VBench's published normalization and aggregate definitions. Because our scene and temporal-flickering wrappers differ from the official pipeline, we treat the scores as controlled within-harness comparisons rather than leaderboard entries.

Aggregate quality. Table 1 is the primary result. At effectively the same 80% sparsity, fine-grained VSA raises Quality by +1.08 points, Semantic by +2.57 points, and Total by +1.38 points over coarse VSA. It also exceeds dense BF16 by +0.66 points in Total, although the matched fixed-seed design isolates method differences rather than estimating repeated-sample variance.

Where quality changes. Figure 2 shows that the aggregate improvement is not confined to the frozen stress set used during development. The largest gains are color (+10.62 points) and scene (+8.49), followed by dynamic degree (+4.17) and human action (+4.00). The largest regressions are spatial relationship (-3.96 points) and imaging quality (-1.38). This pattern is consistent with finer support recovering localized appearance and action cues while leaving some relational cases unresolved.

Systems cost. Coarse VSA behaves as expected inside the denoiser: median attention falls from 591.71 to 274.82 ms and DiT time from 1.53 to 1.44 s; both paired quantities are lower on all 946 prompts. In the isolated same-prompt run, E2E latency is 9.33 s versus 12.48 s for dense, a -25.21% change. This single-prompt result confirms the expected systems direction but is not a latency distribution.

Fine-grained VSA takes 9.73 s versus 9.33 s for coarse VSA in the same isolated run, an overhead of +4.27%. Full-prompt median attention is 680.80 ms versus 274.82 ms, or 2.48× slower. Packing reduces the isolated sparse attention call from 7.245 to 5.423 ms, but coarse VSA remains faster at 2.460 ms. Profiling attributes the remaining gap to short, irregular segments traversing an unfused executor rather than to support selection itself. A custom fused kernel with persistent scheduling is therefore the next systems target; the selected support, attention budget, and output definition need not change.

A frozen 72-prompt diagnostic provides supporting evidence. It contains 24 previously identified coarse-VSA failures and 48 matched safe prompts. Fine-grained VSA repairs 17 of the 24 failures and creates 7 failures among the other 48. The complete 946-prompt evaluation remains the basis for the broad quality claim.

4 Related Work

Video diffusion systems reduce attention cost through structured locality, learned masks, or layer-specific routing. FastVideo uses sliding spatial and temporal tiles [Zhang et al., 2025c]; Sparse VideoGen identifies head-specific spatial and temporal patterns [Xi et al., 2025]; VORTA routes layers among sparse attention variants [Sun et al., 2025]; and Radial Attention imposes distance-dependent sparse structure for long videos [Li et al., 2025]. VSA instead trains a coarse-to-fine selector whose regular exact-attention blocks are directly executable [Zhang et al., 2025b]. Our work keeps that trained selector and budget fixed, isolating block granularity rather than introducing a new routing policy.

Training-free mask construction and hardware co-design provide complementary routes to sparse attention. SpargeAttention builds dynamic sparse masks without retraining [Zhang et al., 2025a], while SpargeAttention2 adds trainable hybrid masking and distillation [Zhang et al., 2026]. FPSAttention jointly designs sparsity and FP8 execution [Liu et al., 2025]. Recent video-specific methods exploit head heterogeneity, spatiotemporal clustering, or runtime load balancing [Zheng et al., 2026, Lee et al., 2026, Liu et al., 2026]. Fine-grained VSA asks a narrower counterfactual: with the checkpoint and selected-token count unchanged, how much quality is lost solely because support is allocated in 64-token units?

5 Conclusion

Redistributing a fixed sparse attention budget from blocks of 64 tokens to groups of eight improves the aggregate VBench score and most measured dimensions without retraining. The result identifies coarse support allocation as a quality bottleneck in VSA and separates that finding from the remaining implementation problem: executing irregular fine-grained support with the efficiency of regular blocks.

6 Limitations

The study covers one 1.3B checkpoint, one video geometry, and one B200 system. The fixed seed provides a controlled paired comparison but does not estimate repeated-sample variance. Automatic metrics do not replace human preference evaluation, and two integrated metric paths differ from official VBench, preventing leaderboard interpretation. The E2E comparison uses one warmed prompt per method and therefore does not estimate a latency distribution. The current unfused executor does not yet match coarse-VSA attention or E2E latency. We also do not release the implementation with this submission, which limits independent reproduction beyond the algorithmic and experimental details reported here.

References

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, Yu Qiao, and Ziwei Liu. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

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

Xingyang Li, Muyang Li, Tianle Cai, Haocheng Xi, Shuo Yang, Yujun Lin, Lvmin Zhang, Songlin Yang, Jinbo Hu, Kelly Peng, Maneesh Agrawala, Ion Stoica, Kurt Keutzer, and Song Han. Radial attention: O(n log n) sparse attention with energy decay for long video generation, 2025.

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, 2025.

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, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

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, 2025a.

Xuan Shen, Chenxia Han, Yufa Zhou, Yanyue Xie, Yifan Gong, Quanyi Wang, Yiwei Wang, Yanzhi Wang, Pu Zhao, and Jiuxiang Gu. Draftattention official repository. Official project repository, 2025b. Public VBench evaluation configuration; fixed seed 42.

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

Wan Team, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models, 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, 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, 2025a.

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, 2026.

Peiyuan Zhang, Yongqi Chen, Haofeng Huang, Will Lin, Zhengzhong Liu, Ion Stoica, Eric Xing, and Hao Zhang. Vsa: Faster video diffusion with trainable sparse attention. In *Advances in Neural Information Processing Systems*, 2025b.

Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhengzhong Liu, and Hao Zhang. Fast video generation with sliding tile attention. In *Proceedings of the International Conference on Machine Learning*, 2025c.

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

A Exact dimension scores

Table 2: All 16 normalized dimensions from the matched fixed-seed evaluation. Rows are ordered by fine-grained VSA minus coarse VSA.

Dimension	Dense	Coarse VSA	Fine VSA	Δ
Color	81.01	74.19	84.81	+10.62
Scene	33.94	28.28	36.77	+8.49
Dynamic degree	90.28	86.11	90.28	+4.17
Human action	75.00	70.00	74.00	+4.00
Overall consistency	61.58	59.61	62.61	+3.00
Background consistency	91.84	90.56	92.95	+2.39
Temporal flickering	96.35	94.26	96.61	+2.35
Aesthetic quality	59.86	59.94	61.14	+1.19
Subject consistency	96.71	95.46	96.52	+1.05
Appearance style	71.08	71.18	72.08	+0.90
Multiple objects	52.67	56.48	56.94	+0.46
Temporal style	60.29	56.69	56.61	−0.07
Object class	74.21	78.80	78.48	−0.32
Motion smoothness	95.32	95.48	94.81	−0.67
Imaging quality	69.92	71.22	69.84	−1.38
Spatial relationship	57.37	68.36	64.40	−3.96

B Specification of fine-grained selection

Token layout. VSA tiles the latent video into $4 \times 4 \times 4$ cubes (4 latent frames $\times$ 4×4 patches) of 64 tokens, stored time-major with valid tokens first in partial edge tiles. Fine-grained VSA splits each cube into eight contiguous 8-token groups, so every group is a $1 \times 2 \times 4$ slab: one latent frame and a 2×4 patch area (about 32×64 pixels). In this geometry every partial tile holds a multiple of eight valid tokens, so each group is either fully valid or fully padded.

Symbols in Algorithm 1. $m_j \in \{0, 8\}$ is the valid-token count of group c_j, and $s_j = \langle \bar{q}, \mathrm{mean}(K_{c_j}) \rangle$ uses the same 64-token block-mean query $\bar{q}$ as coarse VSA. SELECTBUDGET masks groups with $m_j = 0$ and returns the $T_q/8$ highest-scoring valid groups. Since T_q in Eq. (1) is a multiple of eight here, the selected support contains exactly T_q valid tokens.

Scope. P_q, T_q, and the fine selection are computed independently for each denoising step, layer, head, and 64-token query block; all queries in a block share one selection, and nothing is shared across heads. Wan2.1 uses joint spatiotemporal self-attention, so there are no separate spatial and temporal stages: the same rule runs in every self-attention call, and a selected group may come from any frame. Text cross-attention and the gated coarse residual r_q are unchanged.

C Mechanistic analysis: why finer support helps

At a fixed valid-token budget, fine-grained selection captures more dense attention mass because attention inside a 64-token block concentrates in one or two 8-token slabs, which block-mean scoring averages away. The effect is concentrated in a small, stable set of content-retrieval heads.

Method. We capture queries, keys, and values along the coarse-VSA trajectory for 8 prompts (3 steps $\times$ 30 layers $\times$ 12 heads) and evaluate each selection rule on identical inputs against exact dense attention. We report *mass recall*, the fraction of each query's dense attention probability that falls inside the selected support, and the relative L_2 error of the exact-branch output. The gated coarse branch contributes 0.4% of the attention-output norm, so the exact branch determines the approximation. Layers and heads are 0-indexed (L18H7 is layer 18, head 7).

Granularity, not ranking. Mass recall rises monotonically as groups shrink (Table 3). Groups of 32 and 16 tokens retain 22% and 52% of the 8-token recall gain, and 8-token groups achieve 91% of the error reduction available at token resolution. A control that keeps 64-token support but ranks blocks by their best 8-token child ("coarse-max") reaches 0.806 recall and 0.135 error, recovering only 26% of the recall gain, so most of the benefit requires spending the budget at 8-token resolution.

Table 3: Granularity sweep at a fixed per-query token budget. Score-selected groups use the VSA score; the oracle selects groups by true dense mass under the same budget. Group size 64 is coarse VSA and 8 is fine VSA.

Group size (tokens)	64	32	16	8	4	1
Mass recall, score-selected ↑	0.799	0.805	0.813	0.826	0.841	0.870
Mass recall, oracle ↑	0.833	–	–	0.864	–	0.894
Exact-branch error ↓	0.152	–	–	0.123	–	0.120

The oracle gap at 64 tokens (0.833 versus 0.799) shows that scoring error also limits coarse VSA. The improvement holds on each of the 8 prompts (recall $+0.025$ to $+0.030$) and replicates at a second seed (recall $0.782 \to 0.813$, versus $0.778 \to 0.807$ for the same prompt at seed 1024). We did not evaluate VBench quality or latency at intermediate granularities.

Block-mean scoring underestimates concentrated blocks. Within a selected block, the hottest 8-token slab holds 30% of the block's mass on average (12.5% under uniformity). VSA ranks a block by its mean logit, whereas its softmax mass is governed by the log-mean-exp of a few hot keys. In content heads such as L18H7 and L22H8, this Jensen gap lowers the rank correlation between score and block mass from 0.91–0.92 (log-mean-exp) to 0.57–0.59 (mean). Slab-level scores track mass better than block-level scores (Spearman 0.89 versus 0.77), even though the descriptor was trained at 64-token granularity. Fine VSA retains 79% of coarse VSA's tokens and moves the remaining 21% from cold slabs of selected blocks to hot slabs of near-miss blocks (85% ranked 126–250 under the coarse score), roughly doubling the mass those tokens carry (0.026 to 0.053).

Head taxonomy. Per-head effects are stable, with correlations of 0.97–1.00 across prompt pairs and 0.983 across seeds, and they fall into three groups.

- *Content-retrieval heads.* Twenty of 360 heads, all in layers 17–24, account for 61% of the net error reduction; their error falls from 0.50 to 0.18. They attend to a few distinctive keys across frames, which is the case block means dilute most.
- *Same-frame heads* (e.g., L1H1, L3H4, L28H10). Per-token logits oscillate with the key–query frame offset with a period of about six latent frames, carried by the temporal RoPE band. The 4-frame block-mean query therefore aliases single-frame slabs, and fine selection drops slabs in the block's own outer frames; coarse selection keeps whole tiles and is immune. Error in the 19 most degraded heads rises from 0.25 to 0.36.
- *Near-uniform heads* in layer 0 (H2, H3, H8, H9; entropy close to $\log N$). No sparse support approximates them well.

Refining the query side, by scoring each slab with the log-sum-exp over the block's eight 8-token query sub-blocks, lowers error in the degraded heads to 0.29 at the same budget, which suggests a direction for future selectors.

D Causal analysis of the color and scene gains

Much of the color ($+10.62$) and scene ($+8.49$) gain in Table 2 is plausibly a seed-specific artifact. Under seed 1024, coarse selection in four near-uniform layer-0 heads at the first denoising step pushes generations toward a saturated cyan template, and the shared-seed protocol replicates this single event across all 946 prompts.

Pixel distance is uninformative. Three-step sampling is highly sensitive to small systematic perturbations: adding only the gated coarse branch to dense attention (0.4% of the output norm) already moves the video to 18.2 dB PSNR from the dense video, and every sparse method lands near 10 dB. We therefore compare distribution-level statistics rather than distance to the dense output.

Observation. On 28 prompts at seed 1024, coarse VSA produces a recurring saturated cyan/teal backdrop and recolors objects (Figure 3). The median saturated-cyan pixel share[1] is 0.31 for coarse VSA, 0.02 for fine VSA, and 0.05 for dense attention (means 0.30, 0.16, and 0.17). Coarse VSA exceeds dense in cyan share on 19 of 28 prompts ($p = 0.007$, Wilcoxon signed-rank) and in saturation on 25 of 28 ($p = 1.5 \times 10^{-5}$); fine VSA is indistinguishable from dense ($p = 0.53$ and 0.25).

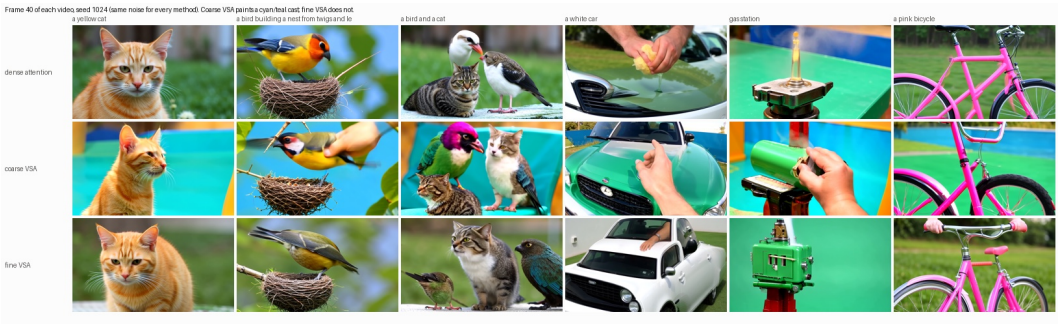


Figure 3: Frame 40 of six prompts at seed 1024, with identical initial noise for every method. Coarse VSA paints a saturated cyan backdrop and recolors objects (the white car turns green); fine VSA does not.

Activation patching. We substitute fine selection into a subset of steps, layers, or heads of an otherwise coarse run and measure the fraction of the coarse-to-fine pixel change it reproduces, together with the cyan share (Table 4, 12 prompts). Patching only step 0 reproduces 93% of the change and removes the cast, whereas steps 1 and 2 reproduce none of it; layers 0–9 account for 80%. Patching only the 20 content-retrieval heads, where attention error improves most, reproduces 4% and leaves the cast, while patching every head except them reproduces 93%. The visible difference is therefore not a consequence of lower attention error in general.

Table 4: Activation patching (12 prompts, seed 1024). Fine selection is used only where stated and coarse selection elsewhere. "Reproduced" is $1 - \|x_{\text{mix}} - x_{\text{fine}}\|^2 / \|x_{\text{coarse}} - x_{\text{fine}}\|^2$ (norms over pixels), averaged over prompts. Median cyan share: coarse 0.273, fine 0.047, dense 0.086.

Fine selection in	Reproduced	Cyan share
Step 0 only	0.93	0.057
Step 1 only	0.00	0.268
Step 2 only	0.00	0.274
Layers 0–9 only	0.80	0.062
Layers 10–19 only	0.08	0.285
Layers 20–29 only	0.01	0.270
20 content-retrieval heads only	0.04	0.294
All heads except those 20	0.93	0.043

Localization. Restricting interventions to step 0 on the 8 strongest-cast prompts (Table 5) isolates the four near-uniform layer-0 heads. Reverting only H2, H3, H8, and H9 to coarse selection in an otherwise fine run restores the cast, and fine selection in layer 0 alone removes it. Decomposing the coarse-minus-fine output difference in layers 0–9, the token-varying component reproduces the cast and the token-mean component does not, so the effect is not a shared bias shift. These are 4 of the 1,080 head-calls in a generation.

What these heads compute. With near-uniform attention, the dense outputs of these heads are smooth absolute-position maps (vertical, center-to-edge, horizontal, and last-frame). Fine selection preserves their shape (cosine to dense 0.83–0.97); coarse selection distorts it (0.62–0.84). Because cross-attention follows self-attention within each block, layer-0 self-attention at step 0 depends only

[1] Hue 160–200°, saturation above 0.45, value above 0.35. The measure was defined after observing the artifact and then tested on held-out seeds (Table 6).

Table 5: Localization at step 0 (8 strongest-cast prompts). Each row changes the selection only at step 0; the base method named in the row is used everywhere else. Median cyan share.

Intervention at step 0	Cyan share
Reference: coarse / fine / dense	0.448 / 0.000 / 0.001
Fine everywhere, coarse in L0 heads H2, H3, H8, H9	0.494
Fine everywhere, coarse in the 7 other degraded heads of layers 0–9	0.000
Coarse everywhere, fine in layer 0 only	0.040
Coarse everywhere, fine in layers 1–9 only	0.442
Fine, plus token-mean part of the coarse−fine difference (layers 0–9)	0.000
Fine, plus token-varying part of that difference (layers 0–9)	0.444

on the initial noise and the timestep. Its statistics are bit-identical across all 8 captured prompts, while layer 1 already differs. One seed therefore yields one distortion for every prompt.

Seed robustness. We regenerated six cast prompts at four further seeds (Table 6). Across the 24 prompt–seed pairs outside seed 1024, coarse and fine VSA do not differ significantly (cyan share $+0.001$, $p = 0.056$; saturation -0.011, $p = 0.70$), and both are more saturated than dense attention ($p \leq 0.004$). The position-map distortion occurs at every seed we measured (0, 7, 42, 1024; H2 cosine 0.62–0.70 for coarse, 0.81–0.83 for fine), but only some initial noises fall into the cyan template.

Table 6: Seed robustness. A prompt counts as cast when its cyan share exceeds the same-seed dense video's by more than 0.10. The six prompts were selected for showing the cast at seed 1024, so that row is 6/6 by construction.

	Cast prompts		Median cyan share		
Seed	Coarse	Fine	Coarse	Fine	Dense
1024 (paper)	6/6	0/6	0.465	0.000	0.005
0	0/6	0/6	0.001	0.000	0.000
7	0/6	0/6	0.003	0.000	0.000
42	2/6	1/6	0.012	0.002	0.000
2025	2/6	1/6	0.051	0.002	0.000

Implications for Table 2. The attention-level improvement in Appendix C is robust across prompts and seeds, but the output-level color effect is seed-specific. The color and scene gains should therefore be re-estimated with multiple samples per prompt (e.g., VBench's five) before being attributed to finer support. Table 7 recomputes the VBench Total from Table 2 with VBench's standard aggregation (which reproduces Table 1), excluding these dimensions or the two dimensions evaluated with non-standard wrappers. Fine VSA retains an advantage of $+0.80$ to $+1.05$ points. These re-aggregations do not correct for seed effects that may reach other dimensions.

Table 7: VBench Total recomputed from Table 2 with dimensions excluded (Quality: weighted mean with dynamic degree at 0.5; Semantic: mean; Total $= (4\,\mathrm{Quality} + \mathrm{Semantic})/5$; ± 0.01 rounding).

Excluded	Dense	Coarse	Fine	Fine−coarse	Fine−dense
None (Table 1)	80.93	80.22	81.59	$+1.38$	$+0.66$
Scene, temporal flickering	80.06	79.67	80.72	$+1.05$	$+0.65$
Color, scene	81.24	80.86	81.84	$+0.98$	$+0.60$
Color, scene, temporal flickering	79.65	79.46	80.26	$+0.80$	$+0.60$

Limitations. These analyses use one checkpoint and prompt subsets (8 prompts for attention capture, 12 and 8 for patching, 28 for color statistics, 6 for seeds), and the color measure was defined post hoc. They do not include VBench at intermediate granularities, an equal-budget random-selection control, or higher step counts.