COFIRA: COMPUTE–FIDELITY REGION ALLOCATION FOR EFFICIENT VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion transformers have substantially advanced video generation, but their attention cost grows rapidly with the large spatiotemporal token sequences involved. Sparse attention is a promising way to alleviate this bottleneck, yet evaluated video attention remains broadly distributed even under oracle selection, suggesting diminishing returns from selector-only sparsification. Existing compensation methods recover non-exact regions with a single summary, but this binary choice between a fixed surrogate and exact attention limits the representations available for regional allocation. We introduce **CoFiRA**, a training-free framework for heterogeneous region-wise compute–fidelity allocation that couples regional representation design with compute allocation. Using query-conditioned contribution and within-cluster K/V variation, CoFiRA assigns each region a calibrated mean, a variable-capacity nested set of real K/V representatives, or exact attention, creating a scalable representation space matched to regional demand. A fidelity-aware runtime packs variable-capacity computations and schedules their irregular routes on device. On HunyuanVideo T2V 13B, CoFiRA achieves a 2.27× end-to-end speedup while reaching 33.09 dB PSNR. Code and demo samples are available at Anonymous/CoFiRA.

1 INTRODUCTION

Diffusion transformers have driven remarkable progress in video generation (Peebles & Xie, 2023; Wan Team, 2025; Kong et al., 2024), enabled in large part by self-attention's ability to capture global dependencies across space and time (Vaswani et al., 2017). The same spatiotemporal scope, however, also makes attention expensive. Its cost grows quadratically with sequence length, while video sequence length scales with both spatial resolution and temporal duration. Attention therefore becomes a major inference bottleneck, making quality-preserving acceleration an important problem. Many recent video sparse-attention methods address this bottleneck through structured block-wise selection (Xi et al., 2025; Yang et al., 2025; Luo et al., 2026). Semantic-clustering methods organize query and key tokens into executable regions, estimate each region's importance, and evaluate selected regions exactly over their original token interactions.

To achieve lower density while retaining dominant attention contributions, mainstream video sparse-attention methods improve the selector so that fewer attention regions capture more attention mass (Yang et al., 2025; Luo et al., 2026), following a strategy widely used in other modalities (Tang et al., 2024; Khaki et al., 2025). This paradigm is effective when attention is concentrated within a small support, such that retaining only a small fraction of tokens preserves most of the mass. Under token-level oracle ranking in the evaluated workloads, the video-generation setting exhibits substantially broader attention support than the text-decoding and image-conditioned-decoding settings. This pattern is consistent with redundancy along both axes of video: repeated spatial structures and temporally persistent content produce many semantically similar tokens with comparable attention scores, distributing non-negligible mass across a broad set of regions. As shown in Figure 1, oracle ranking in the other evaluated workloads reaches high retained mass at low density, whereas video attention requires substantially higher density to retain the same mass. At a fixed low density, even this oracle ranking therefore leaves more mass outside the selected set. Although retained mass does not determine output error, a keep/drop policy cannot represent this residual.

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.

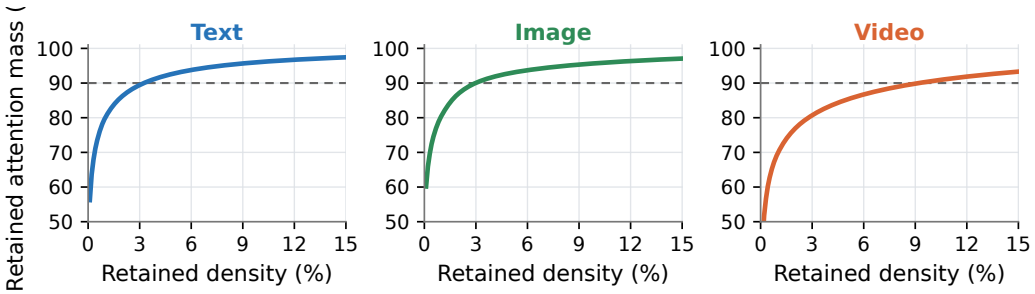


Figure 1: Token-level oracle attention-mass concentration across text, image, and video. In the evaluated settings, video attention requires substantially higher retained density to preserve the same attention mass.

The problem therefore shifts from selecting exact regions alone to representing non-exact contributions accurately and efficiently. A recent method takes this direction by computing exact attention for selected regions and approximating the remainder with compact summaries (Zhou et al., 2026). Centroid-based compensation, in particular, compresses each remaining region into a single first-order K/V summary (Zhou et al., 2026). This leaves only two representational endpoints: one mean per region or all original tokens. For clusters containing several correlated but distinct spatiotemporal modes, a single summary has insufficient capacity to represent their heterogeneous K/V structure; the resulting information loss can propagate as localized structural errors. Exact attention avoids this under-representation but expends full token-level computation even on regions that require only intermediate representative capacity. Within this compensation paradigm, the central limitation is therefore a mismatch between continuously varying region complexity and a binary representation policy.

Motivated by this observation, we introduce **CoFiRA**, a training-free framework for heterogeneous region-wise compute–fidelity allocation that couples regional representation design with compute allocation. CoFiRA estimates where fidelity matters from query-conditioned contribution and within-cluster K/V variation, combining them into a regional sensitivity score. It then assigns progressively higher-capacity representations as sensitivity increases: a compact mean representation for low-sensitivity regions, nested sets of real K/V representatives for intermediate regions, and exact attention for eligible high-sensitivity regions under a size-aware token budget. A lightweight residual first-moment procedure constructs one nested representative sequence whose prefixes expose all intermediate capacities without rate-specific fitting. CoFiRA thereby provides additional representational capacity where a single summary is insufficient, at substantially lower effective density than exact attention.

However, this richer representation space creates an irregular workload: executing each fidelity separately increases launch overhead, while padding every region to the largest capacity substantially reduces the computational savings. We therefore develop a two-stream fidelity-aware runtime with one bias-augmented packed approximate stream for mean and representative regions and a second variable-length stream for exact regions. It constructs irregular route plans on device and combines both streams through global log-sum-exp normalization. Across three text-to-video and image-to-video settings spanning the Wan2.1 and HunyuanVideo model families, CoFiRA achieves the best paired-to-dense fidelity and fastest end-to-end inference among the evaluated sparse methods, while keeping reference-free VBench quality close to dense generation. On HunyuanVideo T2V 13B, CoFiRA operates at 12.5% sparse-step representation density, reduces total denoising-transformer computation to 46% of dense inference, and reaches 33.09 dB paired-to-dense PSNR with a 2.27$\times$ end-to-end speedup.

Our main contributions are summarized as follows:

- We formulate heterogeneous region-wise compute–fidelity allocation by coupling regional representation design with sensitivity-guided compute assignment.
- We develop construction-efficient multi-capacity representations: one nested real K/V sequence provides all intermediate operating points without rate-specific fitting or full exact-attention cost.

- We develop a two-stream runtime with packed approximate execution and device-side routing; across three settings, CoFiRA achieves the best paired-to-dense fidelity and fastest inference among the evaluated sparse methods.

2 Related Work

2.1 Efficient Video Generation

Video diffusion models repeatedly apply large Diffusion Transformers to long spatiotemporal token sequences (Wan Team, 2025; Kong et al., 2024), making inference expensive in both the number of denoising steps and the cost of each model evaluation. Existing studies improve efficiency from several directions. DOLLAR reduces the sampling trajectory through distillation (Ding et al., 2025); PAB and TeaCache exploit redundancy across neighboring timesteps by caching and reusing attention outputs or intermediate features (Zhao et al., 2025b; Liu et al., 2025); efficient architectures such as LinGen replace standard attention with linear-complexity modules (Wang et al., 2025); and ViDiT-Q and Q-VDiT reduce arithmetic and memory costs through low-bit quantization (Zhao et al., 2025a; Feng et al., 2025). IO-aware exact-attention kernels provide an orthogonal systems-level acceleration path (Dao et al., 2022; Dao, 2024; Shah et al., 2024). Another line of work employs sparse attention to reduce the token interactions evaluated in each model forward pass. These approaches target different sources of overhead, and several of them can be combined as complementary acceleration techniques.

2.2 Sparse and Approximate Attention

Sparse attention has been widely studied for efficient LLM and VLM inference. Quest retrieves critical KV-cache pages conditioned on the current query (Tang et al., 2024), while SparseVILA filters and retrieves visual tokens during VLM prefill and decoding (Khaki et al., 2025). Both rely on concentrated attention that allows each query to retain few keys. Video-generation methods adapt this principle to diffusion transformers, where many parallel queries favor structured cluster- or block-level sparsity. SVG constructs sparse layouts from spatial and temporal patterns (Xi et al., 2025); SVG2 groups important cluster pairs through semantic clustering and token permutation (Yang et al., 2025); and SVOO combines offline layer-wise sparsity profiling with online bidirectional Q/K co-clustering (Luo et al., 2026). PSA assigns each regular query–key block an input-dependent resolution within a pooled K/V pyramid, providing a multi-level quality–compute trade-off within one pooling family (Li et al., 2025). Compensation-based methods instead introduce a second regional representation. SVG-EAR evaluates high-risk regions exactly and represents the remainder with K/V centroids, while PISA computes selected blocks exactly and approximates the remainder through a fixed Taylor-expansion path (Zhou et al., 2026; Li et al., 2026). Both retain a binary choice between exact attention and a fixed approximation. These approaches recover discarded contributions, but a single surrogate can underrepresent structurally complex regions, while exact attention can overcompute regions requiring only intermediate capacity. Existing formulations therefore leave open how to move beyond fixed representational endpoints by constructing a scalable regional representation space and jointly allocating representation mechanism and capacity according to regional demands under a compute budget.

3 Method

CoFiRA formulates video sparse attention as **region-wise compute–fidelity allocation** by coupling regional representation design with compute allocation. Figure 2 illustrates its fidelity-specific computation. Unlike pooled multi-resolution allocation (Li et al., 2025), which selects input resolutions within one pooling path, CoFiRA assigns each semantic region both a representation mechanism and capacity under a compute budget. Each region operates at one point on a discrete fidelity ladder, ranging from a compact mean through progressively richer, nested sets of real K/V representatives to exact attention. A lightweight score based on query-conditioned contribution and within-cluster K/V variation assigns these operating points. Concretely, score ranks map intermediate regions to progressively larger nested representative prefixes, while a separate size-aware budget promotes the most sensitive eligible regions to exact attention. We first formulate this allocation problem, then

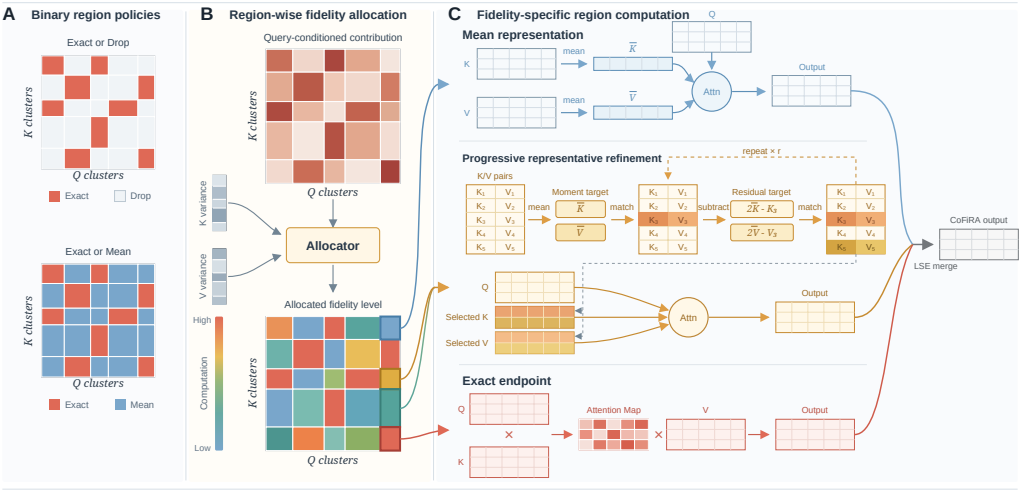


Figure 2: **Overview of CoFiRA.** (A) Binary baselines use exact-or-drop or exact-or-mean decisions. (B) CoFiRA allocates a fidelity level to each region from its contribution and K/V variation. (C) Low-fidelity regions use a compact mean representation, intermediate regions use progressively refined real K/V representatives, and high-fidelity regions use exact attention. Their regional outputs are combined through a globally normalized LSE merge.

describe the fidelity ladder, noise-aware exact-budget scheduling, and the heterogeneous execution backend.

3.1 Problem Formulation and Compute–Fidelity Allocation

Consider one attention head with $Q, K, V \in \mathbb{R}^{N\times d}$. Its dense output is

$$O = PV, \qquad P = \mathrm{softmax}\left(\frac{QK^\top}{\sqrt{d}}\right), \tag{1}$$

To expose regular blocks that can be executed efficiently on GPUs, we follow SVG2's semantic permutation (Yang et al., 2025) and independently partition the query and key tokens into $\{\mathcal{Q}_i\}_{i=1}^{C_q}$ and $\{\mathcal{K}_j\}_{j=1}^{C_k}$. Their Cartesian product $\mathcal{R}_{ij} = \mathcal{Q}_i \times \mathcal{K}_j$ defines an **attention region**. Existing methods typically compute selected regions using all original tokens and either discard or summarize the rest with one fixed approximation. These two endpoints fail in complementary ways: a single mean can under-represent a complex cluster, whereas exact attention pays the full token cost for regions that require only partial refinement.

CoFiRA instead assigns each region a compute–fidelity level. The two endpoints are a compact mean representation and exact attention over all original tokens. Between them, a variable representative capacity r specifies how many real K/V pairs from key cluster $\mathcal{K}_j$ are used to represent the region. A larger r retains more real K/V pairs and therefore provides greater capacity to capture the cluster's internal structure. These intermediate representations are nested, so increasing r extends an existing representative prefix rather than constructing an unrelated approximation. The resulting representation set determines the operating points available to the allocator: each region receives an executable fidelity level according to its estimated demand under a compute budget.

3.2 Regional Sensitivity and Fidelity Assignment

CoFiRA prioritizes richer representations for regions that both affect the current queries and are difficult to compress. It estimates these two factors separately. For query centroid $\bar{q}_i$, the query-conditioned contribution of key cluster j is approximated by the size-weighted centroid attention

$$p_{ij} = \mathrm{softmax}_j\left(\frac{\bar{q}_i^\top \bar{k}_j}{\sqrt{d}} + \log n_j\right), \qquad n_j = |\mathcal{K}_j|. \tag{2}$$

where the softmax is taken over key clusters. The $\log n_j$ term accounts for cluster size, and p_{ij} estimates regional attention mass. SVG-EAR similarly observes that within-cluster coherence affects mean compensation, but estimates this risk through costly per-key probing for binary Exact/Mean routing (Zhou et al., 2026). CoFiRA instead uses normalized mean-squared deviations u_j^K and u_j^V within aligned key and value clusters as lightweight compressibility signals. Key and value variation capture uncertainty in attention strength and diversity of returned content, respectively. We combine them into the regional sensitivity score

$$R_{ij} = p_{ij}\sqrt{u_j^K u_j^V}. \tag{3}$$

Thus, a region receives a high score only when its contribution is non-negligible and a compact summary is likely to erase meaningful K/V structure. We use R_{ij} as an inexpensive surrogate for fidelity assignment, not as a formal error bound.

For each query cluster, a fixed representative fraction selects the highest-scoring regions. Their ordering under R_{ij} determines progressively larger capacities across logarithmically spaced operating points, while the remaining regions use the mean. Candidate regions for exact attention are ranked by the size-aware score $R_{ij}/\sqrt{n_j}$ and promoted under a key-token budget, so a region containing n_j keys consumes n_j units rather than counting as one block. This accounting prevents a large key cluster from receiving exact attention at the same nominal cost as a small one. Under a fixed capacity histogram, descending error-scale–capacity matching minimizes a separable regional-error surrogate. Appendix A provides the exact assignment rule and the corresponding proxy analysis.

3.3 Progressive Regional Fidelity

The assignment above determines how each attention region is represented during computation. Regions assigned the lowest fidelity use a compact mean representation, intermediate regions retain a variable number of representative K/V pairs, and regions requiring the highest fidelity preserve all original tokens for exact attention. These regional contributions are evaluated separately at their assigned fidelities. Each representation returns an unnormalized attention numerator and partition mass, enabling contributions computed at different fidelities to be merged into a globally normalized attention output.

Mean representation. Let $\bar{k}_j$ denote the mean key in $\mathcal{K}_j$, and let $\bar{v}_j$ denote the mean of its aligned values. Existing compensation methods such as SVG-EAR summarize each non-exact region using these means (Zhou et al., 2026). Although inexpensive, this replaces an average of exponentiated scores with the exponential of their average. Jensen's inequality therefore exposes a systematic underestimation of the regional partition mass:

$$Z_j^{\text{mean}}(q) = n_j e^{q^\top \bar{k}_j/\sqrt{d}} \le \sum_{x\in\mathcal{K}_j} e^{q^\top k_x/\sqrt{d}} = Z_j(q). \tag{4}$$

A second-order log-moment approximation identifies score variance as the leading missing term. With the stable query norms induced by QK-RMSNorm, we estimate it from scalar within-cluster key variation and calibrate the regional partition mass as follows:

$$\delta_j = \text{clip}\Big(\frac{\alpha}{2}\sigma_{K,j}^2, 0, \tau\Big), \qquad \widehat{Z}_j = n_j e^{q^\top \bar{k}_j/\sqrt{d}+\delta_j}, \qquad \widehat{A}_j = \widehat{Z}_j \bar{v}_j, \tag{5}$$

where α controls the calibration scale and clipping prevents extreme adjustments. This calibration changes a region's global weight but not its returned content: after regional normalization, the mean path equals $\bar{v}_j$ for every query. By contrast, the exact regional output $f_j(q)$ satisfies $\nabla_q f_j(q) = d^{-1/2}\,\text{Cov}_{w(q)}(V, K)$; query-dependent content therefore depends on aligned K–V variation that a single mean cannot express. More generally, outputs formed from a shared bank of r representatives lie in an affine value subspace of dimension at most $r-1$. Their average regional error is therefore lower-bounded by the exact response covariance outside its leading $r-1$ directions, linking capacity demand to query-dependent response structure.

Progressive representative refinement. CoFiRA restores this query-dependent response at controllable cost through progressively larger sets of aligned K/V pairs.

Because representatives are constructed online for each routed cluster, their construction must be lightweight and deterministic while providing inexpensive control over the information retained by each prefix. We therefore use residual first-moment matching to build one ordered sequence whose nested prefixes support multiple capacities from a single construction. Write $z_x = (k_x, v_x)$ for an aligned K/V pair and $\bar{z}_j = (\bar{k}_j, \bar{v}_j)$ for the cluster mean. Starting from an empty set, step m forms the remaining moment target t_j^m and selects the closest unused real pair under a variation-normalized K/V distance:

$$t_j^m = m\bar{z}_j - \sum_{\ell=1}^{m-1} z_{x_\ell}, \qquad x_m = \arg\min_{x \notin \mathcal{S}_j^{m-1}} \left\| z_x - t_j^m \right\|_{\mathrm{KV}}^2 . \tag{6}$$

Appending x_m yields a herding-style (Rebuffi et al., 2017) ordered sequence whose larger capacities extend the same prefix rather than constructing independent approximations.

For capacity r, let $\mathcal{S}_j^r$ contain the first $\hat{r} = \min(r, n_j)$ representatives. Each retained pair receives the cardinality-corrected attention weight $w_x(q)$, from which its regional partition mass and numerator are computed as

$$w_x(q) = \frac{n_j}{\hat{r}} e^{q^\top k_x/\sqrt{d}}, \qquad \widehat{Z}_j^{(r)} = \sum_{x \in \mathcal{S}_j^r} w_x, \qquad \widehat{A}_j^{(r)} = \sum_{x \in \mathcal{S}_j^r} w_x v_x. \tag{7}$$

The factor $n_j/\hat{r}$ preserves the represented cluster cardinality. Supporting selection, capacity–error, and construction analyses are consolidated in Appendices A and B; they show that larger nested prefixes reduce aggregate output error, while residual matching instantiates all capacities without rate-specific fitting.

Exact endpoint. Exact regions retain every original K/V pair in $\mathcal{K}_j$ and recover the corresponding dense-attention contribution. They are reserved for regions whose estimated sensitivity justifies their full token cost. Together, the mean representation, nested representatives, and exact endpoint form a discrete fidelity ladder over which regional compute can be allocated.

3.4 Noise-Aware Exact-Budget Scheduling

Diffusion denoising progressively maps noise into structured samples, with high-noise steps establishing global structure and later steps refining local details. Approximation errors introduced at high-noise steps can therefore divert the generation trajectory and remain difficult to correct. CoFiRA consequently allocates more exact computation to higher-noise sparse steps and reduces it with the scheduler noise level σ_t:

$$\rho_t^{\mathrm{exact}} = \rho_{\min} + (\rho_{\max} - \rho_{\min}) \left(\frac{\sigma_t - \sigma_{\min}}{\sigma_{\max} - \sigma_{\min}} \right)^{\gamma}, \tag{8}$$

where $\sigma_{\min}$ and $\sigma_{\max}$ span the active sparse phase, and γ is calibrated so that the discrete average matches the target exact budget. At step t, $\rho_t^{\mathrm{exact}} N$ is the exact key-token budget for each query cluster; candidate regions are admitted in size-aware score order subject to this budget. The schedule determines **when** exact computation is most valuable, while regional sensitivity determines **where** to allocate it. Intermediate representative budgets remain fixed throughout the trajectory.

3.5 Fidelity-Aware Heterogeneous Execution

CoFiRA's fidelity assignment produces a highly heterogeneous workload: different regions contain a mean, a short representative sequence, or all original tokens. Executing each fidelity level separately incurs repeated launches, while padding every region to the largest capacity substantially reduces the theoretical saving. We therefore pack mean and representative regions into one bias-augmented approximate stream, while exact regions form a second variable-length stream. The bias term preserves each region's represented token mass, so the packed computation is mathematically equivalent to evaluating the assigned approximate regions separately. Compact cluster-level route descriptors encode heterogeneous capacities without materializing a token-level sparse layout. The approximate and exact streams are then merged from their log-sum-exp statistics, globally normalizing their regional contributions without padding them to a common key extent; Appendix A.5 gives the complete formulation.

Table 1: Quality and efficiency comparison at 720p. VBench Macro averages all six official custom-input dimensions for T2V and the task-applicable VBench-I2V dimensions for I2V; Step Attn. PFLOPs is the average self-attention computation per sparse denoising step.

	Quality				Efficiency				
Method	PSNR ↑	SSIM ↑	LPIPS ↓	VBench Macro ↑	Time (s) ↓	Speedup ↑	Rep. density ↓	Step Attn. PFLOPs ↓	Total PFLOPs ↓
Wan2.1-T2V-14B									
Dense	–	–	–	77.07	1848.3	1.00×	–	–	658.5
SVG2	25.60	0.827	0.162	75.46	1193.4	1.55×	0.32	3.13	440.2
SVOO	25.29	0.843	0.144	**76.94**	1272.3	1.45×	0.37	3.60	427.9
PSA	21.95	0.711	0.241	75.77	1240.0	1.49×	0.13	**1.19**	**372.5**
SVG-EAR	26.96	0.871	0.122	76.85	1188.9	1.55×	0.25	2.51	384.5
CoFiRA	**28.06**	**0.878**	**0.121**	76.55	**1037.1**	**1.78×**	**0.11**	1.22	373.3
Wan2.1-I2V-14B									
Dense	–	–	–	80.51	1482.0	1.00×	–	–	526.8
SVG2	25.64	0.823	0.163	80.38	1008.2	1.47×	0.31	3.04	349.7
SVOO	26.92	0.860	0.131	80.25	998.4	1.48×	0.31	3.10	326.3
SVG-EAR	27.92	0.877	0.120	81.00	1022.1	1.45×	0.25	2.48	306.5
CoFiRA	**28.34**	**0.882**	**0.119**	**81.14**	**846.9**	**1.75×**	**0.11**	**1.21**	**298.5**
HunyuanVideo-T2V-13B									
Dense	–	–	–	73.77	1676.3	1.00×	–	–	612.4
SVG2	29.78	0.883	0.163	71.69	873.1	1.92×	0.27	2.99	351.4
SVOO	27.58	0.870	0.158	**73.92**	846.6	1.98×	0.26	2.85	**270.3**
SVG-EAR	32.17	0.929	0.098	73.76	926.1	1.81×	0.22	2.27	285.3
CoFiRA	**33.09**	**0.934**	**0.090**	73.81	**738.9**	**2.27×**	**0.13**	**1.72**	280.7

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Models and datasets. We evaluate CoFiRA across two video diffusion transformer families and both text-to-video (T2V) and image-to-video (I2V) generation. Specifically, we use Wan2.1-T2V-14B, Wan2.1-I2V-14B, and HunyuanVideo-T2V-13B. Following SVG2, T2V experiments use the Penguin benchmark at 720p, while I2V experiments use VBench-I2V (Huang et al., 2025) under a 720p pixel budget. Wan T2V and I2V generate 81 frames with 50 and 40 denoising steps, respectively, while HunyuanVideo generates 129 frames with 50 steps.

Baselines and metrics. We compare with four training-free sparse-attention baselines. SVG2 and SVOO retain selected regions for exact attention and discard the remainder, SVG-EAR routes high-risk regions to exact attention and represents the remainder with cluster means, while PSA assigns input-dependent pooling rates to regular query–key blocks. Each baseline follows its paper configuration and original algorithmic path. We report PSNR, SSIM, and LPIPS (Wang et al., 2004; Zhang et al., 2018) against paired dense videos, and use VBench (Huang et al., 2024) to assess reference-free video-quality attributes. For T2V, we evaluate all six dimensions supported by the official custom-input protocol and report their macro average. Efficiency is evaluated by end-to-end generation time, speedup over dense inference, sparse-step representation density, sparse-step attention PFLOPs, and full-trajectory transformer PFLOPs.

Implementation. All methods use BF16, generation seed 0, and identical task-specific resolution, frame, scheduler, and VAE protocols. We retain each baseline's released dense warm-up and sparsity control. CoFiRA uses dense attention over the first 0.3 of the denoising trajectory on Wan and the first 0.25 on HunyuanVideo; its nominal exact budgets are 4.5%, 6%, and 10% for Wan T2V, Wan I2V, and Hunyuan T2V, respectively. Efficiency is measured on NVIDIA H100 GPUs with CUDA 12.8 and exact FlashAttention-3 for dense attention. Pipeline time includes denoising and VAE decoding. Full model-specific configurations, frozen sample lists, and evaluation protocols are provided in Appendix B.

4.2 MAIN RESULTS

Table 1 shows that CoFiRA achieves the best paired-to-dense fidelity in all three settings, consistently improving PSNR, SSIM, and LPIPS. On Wan T2V, for example, it improves PSNR by 1.10

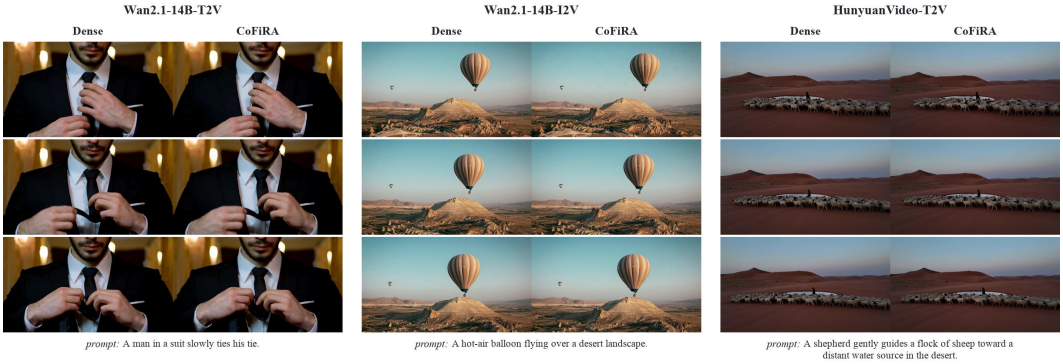


Figure 3: Qualitative comparison between dense attention and CoFiRA.

Table 2: Effect of multi-rate region representation on Wan T2V.

Metric	E/M	Fixed-6	**Multi-rate**
PSNR ↑	27.17	27.29	**27.70**
SSIM ↑	0.846	0.857	**0.870**
LPIPS ↓	0.165	0.151	**0.139**
Patch error ↓	0.115	0.110	**0.102**
Residual > 0.3 ↓	8.5%	7.1%	**5.9%**

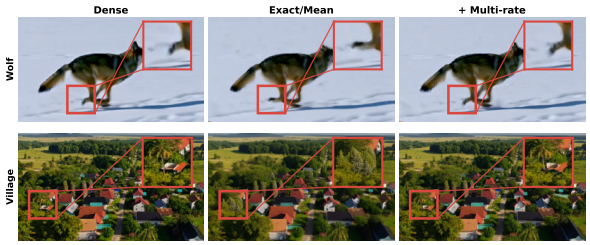


Figure 4: Localized reconstruction errors under Exact/Mean and multi-rate representations.

dB over the strongest-fidelity baseline while reducing generation time by 12.8%. Qualitative results in Figure 3 further show that CoFiRA closely reproduces the subject identity and scene layout of dense generation across sampled frames, model families, and generation tasks. VBench complements these paired-reference metrics by assessing reference-free quality attributes. CoFiRA remains close to dense generation and competitive with sparse baselines across all three settings, including the best VBench macro score of 81.14 on Wan I2V.

These fidelity gains come with the strongest end-to-end acceleration in every evaluated setting, ranging from 1.75–2.27×. Relative to the fastest baseline in each setting, it further reduces generation time by 12.7–15.2%. On HunyuanVideo, CoFiRA reaches 2.27× speedup while improving all three paired-reference metrics over the baselines. Across tasks, it reduces sparse-step representation density relative to all evaluated baselines and nearly halves total denoising-transformer computation relative to dense inference. We additionally report sparse-step PFLOPs to isolate sparse-attention cost. On Wan T2V, CoFiRA uses PFLOPs comparable to PSA per sparse step while improving PSNR by 6.11 dB. Overall, CoFiRA combines the best paired-to-dense fidelity with the lowest representation density and fastest end-to-end inference.

4.3 Localized Residuals under Binary Region Approximation

Binary region approximation forces every non-exact region into a single mean, regardless of its internal complexity. Table 2 isolates both the value of intermediate representations and the value of allocating their capacity adaptively. Exact/Mean removes the intermediate levels, whereas Fixed-6 assigns six real K/V tokens to every intermediate region; all three settings use comparable total computation. CoFiRA improves all three global fidelity metrics, gaining 0.52 dB over Exact/Mean and 0.41 dB over Fixed-6. Mean patch error falls from 0.115 and 0.110 to 0.102, respectively, while residual exceedance falls from 8.5% and 7.1% to 5.9%. Mean patch error measures average local reconstruction deviation, whereas residual exceedance reports the fraction of patches whose residual exceeds 0.3. Figure 4 illustrates the corresponding local errors: Exact/Mean introduces tree-like texture absent from the dense result, whereas CoFiRA preserves the building structure, consistent with a single mean lacking sufficient capacity for a structurally complex region. Under comparable computation, these controls identify adaptive region-wise capacity allocation as the primary source of CoFiRA's fidelity gain.

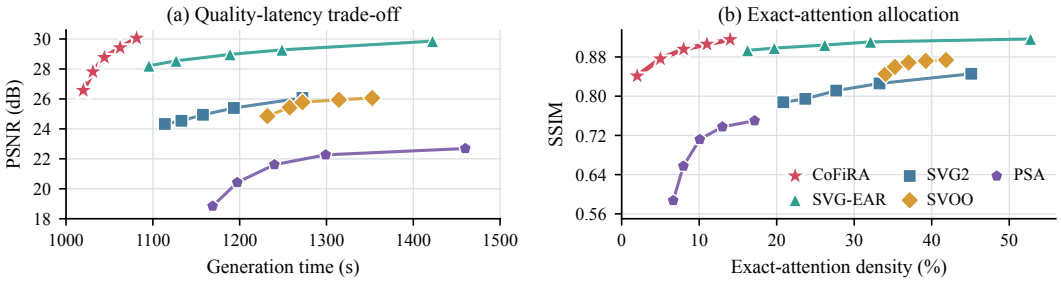


Figure 5: Compute–fidelity analysis on 20 Wan T2V samples. (a) PSNR versus H100 generation time. (b) SSIM versus exact-attention density.

Table 3: Routing signals.

Signal	PSNR	SSIM	LPIPS
Contribution	27.44	0.863	0.145
K/V variation	18.52	0.464	0.486
Joint	**27.70**	**0.870**	**0.139**

Table 4: Allocation placement.

Assignment	PSNR	SSIM	LPIPS
Full shuffle	17.70	0.429	0.509
Interm. shuffle	27.05	0.849	0.158
Adaptive	**27.70**	**0.870**	**0.139**

Table 5: Attention latency.

Execution	Latency
Separate kernels	250.47 ms
+ Packed approx. stream	140.72 ms
+ Device scheduling	**133.07 ms**

4.4 Compute–Fidelity Trade-off

Figure 5 evaluates how efficiently each method converts additional computation into paired-to-dense fidelity. Panel (a) shows a more favorable fidelity–latency trade-off and a markedly steeper improvement curve for CoFiRA. At a 14% exact budget, CoFiRA reaches 30.05 dB PSNR in 1081.3 seconds, surpassing SVG-EAR's best point of 29.86 dB in 1422.2 seconds. This steeper curve shows that CoFiRA converts incremental computation into fidelity more efficiently, consistent with assigning representative capacity where it is most useful.

Panel (b) explains where this efficiency arises. All CoFiRA operating points use lower exact-attention density than every SVG-EAR point. At 11% exact density, CoFiRA reaches 0.906 SSIM, exceeding the 0.904 achieved by SVG-EAR at 26.2% exact density. This comparison indicates that Exact/Mean can assign full exact attention to regions already represented adequately at an intermediate capacity, thereby wasting computation. By matching capacity to regional complexity, CoFiRA uses the exact-attention budget more efficiently.

4.5 Ablation Study

Table 3 shows that query-conditioned contribution is the primary routing signal, reaching 27.44 dB PSNR. Adding K/V variation consistently improves all three metrics to 27.70 dB PSNR, 0.870 SSIM, and 0.139 LPIPS, indicating that it provides complementary information about within-region complexity. Table 4 preserves the allocation histogram and computation while shuffling either the full fidelity assignment or intermediate capacities alone. Both controls degrade all three metrics. Thus, neither the amount of computation nor the proportions of fidelity levels are sufficient: matching fidelity and representative capacity to regional demand is essential to CoFiRA's gains. Finally, Table 5 evaluates the complete sparse-attention module. Packing the mean and variable-capacity representatives into one approximate stream reduces latency from 250.47 to 140.72 ms, while device-side scheduling further lowers it to 133.07 ms, a 46.9% reduction overall. Together with the lower FLOP counts in Table 1, these results show that CoFiRA's end-to-end gains arise from both reduced computation and more efficient execution.

5 Conclusion

We introduced CoFiRA, a training-free framework that jointly allocates representative capacity and computation across heterogeneous attention regions. By combining compact means, variable-capacity real K/V representatives, and exact attention, CoFiRA improves paired-to-dense fidelity and inference speed across T2V and I2V tasks and two model families, reaching up to 2.27× speedup.

AI USE STATEMENT

Generative AI tools assisted with figure creation and revision, manuscript organization and language polishing, and title and keyword development. The authors reviewed all AI-assisted outputs and take responsibility for the final paper.

REFERENCES

Shuai Bai et al. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In *International Conference on Learning Representations*, 2024.

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*, volume 35, 2022.

Zihan Ding, Chi Jin, Difan Liu, Haitian Zheng, Krishna Kumar Singh, Qiang Zhang, Yan Kang, Zhe Lin, and Yuchen Liu. DOLLAR: Few-step video generation via distillation and latent reward optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17961–17971, 2025.

Weilun Feng, Chuanguang Yang, Haotong Qin, Xiangqi Li, Yu Wang, Zhulin An, Libo Huang, Boyu Diao, Zixiang Zhao, Yongjun Xu, and Michele Magno. Q-VDiT: Towards accurate quantization and distillation of video-generation diffusion transformers. *arXiv preprint arXiv:2505.22167*, 2025.

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*, pp. 21807–21818, 2024.

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, Yaohui Wang, Xinyuan Chen, Ying-Cong Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025. doi: 10.1109/TPAMI.2025.3633890.

Samir Khaki, Junxian Guo, Jiaming Tang, Shang Yang, Yukang Chen, Konstantinos N. Plataniotis, Yao Lu, Song Han, and Zhijian Liu. Sparsevila: Decoupling visual sparsity for efficient VLM inference. *arXiv preprint arXiv:2510.17777*, 2025.

Weijie Kong et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

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, 2026. URL https://arxiv.org/abs/2602.01077.

Xiaolong Li, Youping Gu, Xi Lin, Weijie Wang, and Bohan Zhuang. PSA: Pyramid sparse attention for efficient video understanding and generation, 2025. URL https://arxiv.org/abs/2512.04025.

Feng Liu, Shiwei Zhang, Xiaofeng Wang, Yujie Wei, Haonan Qiu, Yuzhong Zhao, Yingya Zhang, Qixiang Ye, and Fang Wan. Timestep embedding tells: It's time to cache for video diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7353–7363, 2025.

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 43rd International Conference on Machine Learning*, volume 306 of *Proceedings of Machine Learning Research*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H. Lampert. iCaRL: Incremental classifier and representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Jay Shah, Ganesh Bikshandi, Ying Zhang, Vijay Thakkar, Pradeep Ramani, and Tri Dao. Flashattention-3: Fast and accurate attention with asynchrony and low-precision. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Jiaming Tang, Yilong Zhao, Kan Zhu, Guangxuan Xiao, Baris Kasikci, and Song Han. Quest: Query-aware sparsity for efficient long-context LLM inference. *arXiv preprint arXiv:2406.10774*, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

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

Hongjie Wang, Chih-Yao Ma, Yen-Cheng Liu, Ji Hou, Tao Xu, Jialiang Wang, Felix Juefei-Xu, Yaqiao Luo, Peizhao Zhang, Tingbo Hou, Peter Vajda, Niraj K. Jha, and Xiaoliang Dai. Lingen: Towards high-resolution minute-length text-to-video generation with linear computational complexity. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2578–2588, 2025.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004. doi: 10.1109/TIP.2003.819861.

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. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 68208–68224, 2025.

Shuo Yang, Haocheng Xi, Yilong Zhao, Muyang Li, Jintao Zhang, Han Cai, Yujun Lin, Xiuyu Li, Chenfeng Xu, Kelly Peng, Jianfei Chen, Song Han, Kurt Keutzer, and Ion Stoica. Sparse videogen2: Accelerate video generation with sparse attention via semantic-aware permutation. In *Advances in Neural Information Processing Systems*, 2025.

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*, pp. 586–595, 2018.

Tianchen Zhao, Tongcheng Fang, Haofeng Huang, Rui Wan, Widyadewi Soedarmadji, Enshu Liu, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, Huazhong Yang, Xuefei Ning, and Yu Wang. ViDiT-Q: Efficient and accurate quantization of diffusion transformers for image and video generation. In *International Conference on Learning Representations*, 2025a.

Xuanlei Zhao, Xiaolong Jin, Kai Wang, and Yang You. Real-time video generation with pyramid attention broadcast. In *International Conference on Learning Representations*, 2025b.

Xuanyi Zhou, Qiuyang Mang, Shuo Yang, Haocheng Xi, Jintao Zhang, Huanzhi Mao, Joseph E. Gonzalez, Kurt Keutzer, Ion Stoica, and Alvin Cheung. SVG-EAR: Parameter-free linear compensation for sparse video generation via error-aware routing. *arXiv preprint arXiv:2603.08982*, 2026.

A Additional Method Details

This section expands the four operations summarized in Section 3: computing the allocator statistics, calibrating the mean representation, constructing nested real K/V representatives, and merging the resulting fidelity levels under one global normalization.

A.1 Fidelity Allocator Statistics and Assignment

For a key cluster $\mathcal{K}_j$ containing n_j tokens and its aligned values, we compute the per-channel mean-squared deviations

$$s_j^K = \frac{1}{n_j d} \sum_{x \in \mathcal{K}_j} \|k_x - \bar{k}_j\|_2^2, \qquad s_j^V = \frac{1}{n_j d} \sum_{x \in \mathcal{K}_j} \|v_x - \bar{v}_j\|_2^2. \tag{9}$$

Within each head, they are normalized across key clusters as

$$u_j^X = \frac{\log(1 + s_j^X)}{\max\Big(C_k^{-1} \sum_{\ell=1}^{C_k} \log(1 + s_\ell^X), 10^{-8}\Big)}, \qquad X \in \{K, V\}. \tag{10}$$

The normalized quantities u_j^K and u_j^V enter the regional sensitivity R_{ij} in Equation 3. The raw quantities are retained for scale-sensitive operations below; in particular, $\sigma_{K,j}^2 = s_j^K$ in the mean-mass calibration. As a diagnostic of the $u_j^V = 0$ edge case, we inspected a complete Wan T2V-14B sparse-attention snapshot containing 98,075 nonempty key clusters across all 40 heads. Every zero-u_j^V cluster was a singleton and also had $u_j^K = 0$; no zero-u_j^V, nonzero-u_j^K case occurred in this snapshot. This empirical check complements, rather than changes, our treatment of R_{ij} as a heuristic surrogate instead of an error bound.

Let C_k be the number of key clusters and ρ_{rep} the fraction eligible for representative refinement. When $\rho_{\text{rep}} > 0$, we set $M_{\text{rep}} = \max(4, \min(C_k, \text{round}(\rho_{\text{rep}} C_k)))$; otherwise, $M_{\text{rep}} = 0$ (all deployed configurations have $C_k \geq 4$). For each query cluster i, we sort the M_{rep} highest-scoring regions by R_{ij} and divide this ordered list into four contiguous groups. From highest to lowest sensitivity, the groups receive 16, 8, 4, and 2 real K/V representatives, respectively. Regions outside this list use the mean representation. Group g occupies indices $[\lfloor g M_{\text{rep}}/4 \rfloor, \lfloor (g+1) M_{\text{rep}}/4 \rfloor)$ for $g = 0, \ldots, 3$.

Exact promotion uses a separate key-token budget. Here ρ_{exact} denotes the current step's scheduled exact fraction ρ_t^{exact} from Equation 8, and $B_{\text{exact}} = \max(1, \text{round}(\rho_{\text{exact}} N))$, where N is the sequence length. For each query cluster, eligible regions with $n_j \leq 256$ are ranked by the size-aware score $R_{ij}/\sqrt{n_j}$. We inspect the highest-ranked M_{cand} regions, where

$$M_{\text{cand}} = \begin{cases} \min(C_k, \max(M_{\text{rep}}, \text{round}(2\rho_{\text{rep}} C_k))), & M_{\text{rep}} > 0, \\ \min(C_k, \text{round}(2\rho_{\text{exact}} C_k)), & M_{\text{rep}} = 0. \end{cases} \tag{11}$$

The longest prefix whose cumulative key count does not exceed B_{exact} is evaluated exactly. Exact assignment overrides any representative level previously assigned to the same region. Because regions are indivisible, the realized exact-token count is at most, but need not equal, B_{exact}. The candidate-pool multiplier and the 256-token size cap are fixed implementation hyperparameters that bound routing and exact-kernel overhead.

Queries are clustered by cosine distance and keys by Euclidean distance. On the first sparse invocation, fixed random samples (seeds 123 and 456 for Q and K) initialize the centroids, followed by 20 Q updates and 10 K updates. Later sparse invocations warm-start from the centroids produced by the preceding invocation of the same attention layer and perform one update. The model-specific cluster counts and budgets are listed in Table 7.

A.2 Derivation of the Mean-Mass Calibration

Consider a fixed query q and the scores $s_x = q^\top k_x/\sqrt{d}$ within key cluster $\mathcal{K}_j$. Writing $s_x = \mu_j + \epsilon_x$, where $\mu_j = q^\top \bar{k}_j/\sqrt{d}$ and $\mathbb{E}_x[\epsilon_x] = 0$, the exact regional partition mass is

$$Z_j(q) = \sum_{x \in \mathcal{K}_j} e^{s_x} = n_j e^{\mu_j} \mathbb{E}_x[e^{\epsilon_x}]. \tag{12}$$

Averages and covariances in this derivation are empirical quantities under the uniform distribution over tokens in $\mathcal{K}_j$. A direct mean representation drops the final expectation and uses $n_j e^{\mu_j}$. Jensen's inequality shows that this is a lower estimate: $e^{\mathbb{E}[s_x]} \le \mathbb{E}[e^{s_x}]$, with strict inequality whenever the scores have nonzero variation.

The cumulant expansion of the omitted term gives

$$\log \mathbb{E}_x[e^{\epsilon_x}] = \frac{1}{2}\operatorname{Var}_x(s_x) + \frac{1}{6}\kappa_3(s_x) + \cdots, \tag{13}$$

so the leading correction to the mean score is one half of its variance. The exact score variance is query dependent,

$$\operatorname{Var}_x(s_x) = \frac{1}{d} q^\top \operatorname{Cov}(K_j) q. \tag{14}$$

Evaluating this quadratic form for every query–cluster pair would undermine the purpose of the lowest-cost representation. We therefore use the scalar within-cluster key variation $\sigma^2_{K,j}$ as an isotropic proxy. QK-RMSNorm keeps query norms approximately stable, while the scale α absorbs the remaining normalization and anisotropy, yielding

$$\delta_j = \operatorname{clip}\Big(\frac{\alpha}{2}\sigma^2_{K,j}, 0, \tau\Big). \tag{15}$$

We use $\alpha = 1$ and $\tau = 2$. The corrected approximation $\widehat{Z}_j = n_j e^{\mu_j + \delta_j}$ compensates for the leading missing term in the log-partition mass, and $\widehat{A}_j = \widehat{Z}_j \bar{v}_j$ supplies the corresponding unnormalized value contribution. This approximation does not model score–value correlation or multiple modes within the cluster; those cases motivate the progressively richer representative sets used by CoFiRA.

A.3 Residual First-Moment Representative Selection

For each key cluster $\mathcal{K}_j$, let $\phi(\cdot)$ extract a fixed d_ϕ-dimensional channel sketch. We write $\tilde{k}_x = \phi(k_x)$ and $\tilde{v}_x = \phi(v_x)$, with corresponding cluster means $\widetilde{\bar{k}}_j$ and $\widetilde{\bar{v}}_j$. For a nonempty selected set $\mathcal{S}$, define its variation-normalized K/V first-moment error as

$$\mathcal{E}_{\mathrm{KV}}(\mathcal{S}; \mathcal{K}_j) = \frac{\Big\| \frac{1}{|\mathcal{S}|}\sum_{x\in\mathcal{S}} \tilde{k}_x - \widetilde{\bar{k}}_j \Big\|_2^2}{d_\phi s_j^K} + \frac{\Big\| \frac{1}{|\mathcal{S}|}\sum_{x\in\mathcal{S}} \tilde{v}_x - \widetilde{\bar{v}}_j \Big\|_2^2}{d_\phi s_j^V}, \tag{16}$$

where s_j^K and s_j^V are clamped below by 10^{-8}. This normalization prevents the numerical scale of either K or V from dominating selection.

Suppose $m-1$ representatives $x_1, \ldots, x_{m-1}$ have already been selected. The residual targets for the next pair are

$$t^K_{j,m} = m\widetilde{\bar{k}}_j - \sum_{\ell<m} \tilde{k}_{x_\ell}, \qquad t^V_{j,m} = m\widetilde{\bar{v}}_j - \sum_{\ell<m} \tilde{v}_{x_\ell}. \tag{17}$$

The implementation chooses the unused aligned K/V pair

$$x_m = \arg\min_{x\in\mathcal{K}_j\setminus\mathcal{S}_j^{m-1}} \left[\frac{\|\tilde{k}_x - t^K_{j,m}\|_2^2}{d_\phi s_j^K} + \frac{\|\tilde{v}_x - t^V_{j,m}\|_2^2}{d_\phi s_j^V} \right]. \tag{18}$$

For fixed m, this is equivalent to greedily minimizing $\mathcal{E}_{\mathrm{KV}}(\mathcal{S}_j^{m-1} \cup \{x\}; \mathcal{K}_j)$ up to a constant scale. It therefore selects the pair that best repairs the first-moment discrepancy left by the current prefix, rather than repeatedly choosing the token nearest to the cluster mean.

Why a single mean is insufficient. For a fixed cluster, define the regionally normalized exact output and token weights as

$$f_j(q) = \sum_{x\in\mathcal{K}_j} w_x(q) v_x, \qquad w_x(q) = \frac{e^{q^\top k_x/\sqrt{d}}}{\sum_{y\in\mathcal{K}_j} e^{q^\top k_y/\sqrt{d}}}. \tag{19}$$

Direct differentiation gives the response identity

$$\nabla_q f_j(q) = \frac{1}{\sqrt{d}} \operatorname{Cov}_{w(q)}(V, K), \tag{20}$$

where $\mathrm{Cov}_{w(q)}(V,K) = \sum_x w_x(q)(v_x - f_j(q))(k_x - \sum_y w_y(q)k_y)^\top$. Mass calibration changes the weight of this region in the global output, but the regionally normalized mean path remains $\bar{v}_j$ for every query and has zero Jacobian. By contrast, the output of r aligned representatives lies in the affine hull of their values, and its Jacobian has rank at most $r-1$.

A balanced two-mode cluster makes the separation explicit. Suppose half of its tokens equal $(\bar{k}_j + a, \bar{v}_j + b)$ and half equal $(\bar{k}_j - a, \bar{v}_j - b)$. With $t = q^\top a/\sqrt{d}$,

$$f_j(q) = \bar{v}_j + b\tanh(t), \qquad f_j^{\mathrm{mean}}(q) = \bar{v}_j. \tag{21}$$

Thus even an oracle correction of the mean path's partition mass leaves content error $\|b\|_2|\tanh(t)|$. Retaining one real pair from each mode with the $n_j/2$ cardinality correction exactly recovers both the regional numerator and mass. Moreover, if the joint sketch distinguishes the two modes, residual matching selects the opposite mode on its second step after either mode is selected first. This example isolates the representational limitation of a single mean without claiming universal dominance over other multi-representative constructions.

The selection rule admits a precise one-step guarantee. Define the variation-normalized joint sketch

$$y_x = \left[\frac{\tilde{k}_x}{\sqrt{d_\phi s_j^K}}; \frac{\tilde{v}_x}{\sqrt{d_\phi s_j^V}}\right], \qquad \bar{y}_j = \frac{1}{n_j}\sum_{x\in\mathcal{K}_j} y_x, \tag{22}$$

and let $u_x = y_x - \bar{y}_j$ and $r_m = \sum_{\ell=1}^m u_{x_\ell}$. Equation 16 then becomes $\|r_m\|_2^2/m^2$, and Equation 18 selects the unused token minimizing $\|r_{m-1} + u_x\|_2^2$.

One-step optimality. Let $\mathcal{R}_m$ be the unused tokens before step m and $M_m = |\mathcal{R}_m| = n_j - m + 1$. Since $\sum_{x\in\mathcal{R}_m} u_x = -r_{m-1}$, the greedy extension satisfies

$$\begin{aligned}\|r_m\|_2^2 &\le \frac{1}{M_m}\sum_{x\in\mathcal{R}_m}\|r_{m-1}+u_x\|_2^2 \\ &= \left(1 - \frac{2}{M_m}\right)\|r_{m-1}\|_2^2 + \frac{1}{M_m}\sum_{x\in\mathcal{R}_m}\|u_x\|_2^2.\end{aligned} \tag{23}$$

The right-hand side is the average next-prefix error numerator over all feasible extensions of the current prefix. Thus, at every capacity increment, residual matching selects an extension whose first-moment error is no larger than the average feasible extension. The universal lower bound is zero, attained when the selected prefix exactly matches the cluster K/V first moments.

Finite-population capacity bound. For $n_j > 1$, let $H = \max_{x\in\mathcal{K}_j}\|u_x\|_2$; a singleton is represented exactly by its only token. Bounding the final term in Equation 23 by H^2 and applying induction gives

$$\|r_m\|_2^2 \le H^2\frac{m(n_j - m)}{n_j - 1}, \qquad 1 \le m \le n_j, \tag{24}$$

where the endpoint $m = n_j$ follows from $\sum_x u_x = 0$. Consequently, the deployed greedy prefix obeys the unconditional bound

$$\mathcal{E}_{\mathrm{KV}}(\mathcal{S}_j^m; \mathcal{K}_j) \le H^2\frac{n_j - m}{m(n_j - 1)}. \tag{25}$$

This controls the sketched first moment for every finite cluster without requiring the coverage condition used below.

Sketch-controlled response and residual information. The finite-population result controls the statistic optimized by the deployed selector, but a first moment alone cannot determine nonlinear attention. We therefore separate the attention response represented by the sketch from the information it omits. Fix a query q, let $s_x = q^\top k_x/\sqrt{d}$, and define the shift-invariant weights $w_x = \exp(s_x - \min_y s_y) \ge 1$. The exact and representative regional outputs are

$$f = \frac{n_j^{-1}\sum_x w_x v_x}{n_j^{-1}\sum_x w_x}, \qquad \widehat{f}_m = \frac{m^{-1}\sum_{x\in\mathcal{S}_j^m} w_x v_x}{m^{-1}\sum_{x\in\mathcal{S}_j^m} w_x}. \tag{26}$$

Write $g_x(q) = w_x(v_x - f)$, so $n_j^{-1}\sum_x g_x = 0$, and decompose its query-dependent response for any matrix B as

$$g_x = Bu_x + r_x(B), \qquad \eta_B^2 = \frac{1}{n_j}\sum_x \|r_x(B)\|_2^2. \tag{27}$$

Here Bu_x is the component linearly represented by the actual joint sketch, whereas $r_x(B)$ contains nonlinear response and information discarded by that sketch. If $\epsilon_m = \|m^{-1}\sum_{x\in\mathcal{S}_j^m} u_x\|_2$ and $\bar{w}_m = m^{-1}\sum_{x\in\mathcal{S}_j^m} w_x$, then

$$\|\widehat{f}_m - f\|_2 \le \frac{\|B\|_{\mathrm{op}}\epsilon_m + \sqrt{(n_j-m)/m}\,\eta_B}{\bar{w}_m}. \tag{28}$$

To prove this bound, set $d_x = \mathbf{1}\{x \in \mathcal{S}_j^m\}/m - 1/n_j$. Then $\sum_x d_x = 0$, $\sum_x d_x^2 = (n_j - m)/(mn_j)$, and

$$\widehat{f}_m - f = \frac{\sum_x d_x g_x}{\bar{w}_m} = \frac{B\sum_x d_x u_x + \sum_x d_x r_x(B)}{\bar{w}_m}. \tag{29}$$

The first term is bounded by $\|B\|_{\mathrm{op}}\epsilon_m$ and the second by Cauchy–Schwarz. Combining this result with Equation 25 yields the unconditional bound

$$\|\widehat{f}_m - f\|_2 \le \frac{1}{\bar{w}_m}\inf_B\left[\|B\|_{\mathrm{op}}H\sqrt{\frac{n_j-m}{m(n_j-1)}} + \sqrt{\frac{n_j-m}{m}}\,\eta_B\right]. \tag{30}$$

Both terms vanish when all tokens are retained. For smaller capacities, the bound deliberately exposes the residual rather than assuming that matching a low-dimensional first moment recovers every attention response.

This separation is also exact in an information-theoretic sense. Let U have rows $u_x^\top$ and let $G(q)$ have rows $g_x(q)^\top$. For a fixed query, all probability reweightings of the original support that preserve the sketch first moment return the same regional output if and only if

$$\mathrm{col}(G(q)) \subseteq \mathrm{col}(U). \tag{31}$$

Indeed, the inclusion gives $G = UB^\top$ and hence preserves the centered response whenever the sketch moment is preserved. If it fails, there exists a direction h with $\mathbf{1}^\top h = 0$, $U^\top h = 0$, but $G^\top h \neq 0$; a sufficiently small probability perturbation along h preserves the moment while changing the output. Equation 31 therefore characterizes when the residual in Equation 28 can vanish, rather than assuming that it is always small.

Rate under cluster coverage. The usual coverage condition for herding also yields a stronger deterministic rate. Suppose that $\|u_x\|_2 \le R$ and, for every nonzero prefix residual up to capacity m, an unused token satisfies $\langle r_{\ell-1}, u_x\rangle \le -\gamma\|r_{\ell-1}\|_2$ for some $\gamma > 0$. Greedy selection is no worse than this token, so

$$\|r_\ell\|_2^2 \le \|r_{\ell-1}\|_2^2 - 2\gamma\|r_{\ell-1}\|_2 + R^2. \tag{32}$$

If $\|r_{\ell-1}\|_2 > R^2/(2\gamma)$, the residual cannot increase; otherwise the triangle inequality gives $\|r_\ell\|_2 \le R + R^2/(2\gamma)$. Induction therefore yields

$$\mathcal{E}_{\mathrm{KV}}(\mathcal{S}_j^m; \mathcal{K}_j) \le \frac{\big(R + R^2/(2\gamma)\big)^2}{m^2}. \tag{33}$$

Hence, whenever the remaining cluster continues to cover a direction opposite to the current residual, the first-moment error decreases at rate $O(m^{-2})$.

A spectral lower bound on finite capacity. The preceding bounds explain what the deployed selector controls. A complementary question is why a region may require more than one representative in the first place. Consider any finite query set $q_1, \ldots, q_T$ for one region, with exact regionally normalized outputs $f_t = f_j(q_t)$, mean $\bar{f}$, and covariance

$$C_f = \frac{1}{T}\sum_{t=1}^{T}(f_t - \bar{f})(f_t - \bar{f})^\top, \qquad \lambda_1 \ge \lambda_2 \ge \cdots \ge 0. \tag{34}$$

Any approximation sharing the same r representative values across these queries returns outputs in their affine hull, whose dimension is at most $r-1$. Consequently, irrespective of whether the representatives are real tokens or synthesized centers,

$$\frac{1}{T}\sum_{t=1}^{T}\|f_t-\widehat{f}_t\|_2^2 \geq \sum_{\ell=r}^{d_v}\lambda_\ell(C_f). \tag{35}$$

The proof follows from the optimal rank-$(r-1)$ affine approximation: every feasible output set lies in such a subspace, whose minimum average projection error is the covariance energy outside its leading $r-1$ directions. For $r=1$, the lower bound is $\mathrm{tr}(C_f)$; changing only the region's partition mass cannot remove this query-dependent content error. More generally, the spectral tail formalizes why regions with different response structure require different capacities. Under an ideal relaxed allocation, each additional representative should be assigned to the largest remaining weighted response eigenvalue across regions. This provides a capacity-allocation reference, not an optimality claim for the lightweight score used by CoFiRA.

The selector itself operates on a lightweight 32-channel projection. We use 32 uniformly spaced channels for ϕ, and the explicit residual in Equation 30 accounts for response information outside this sketch.

The resulting ordered prefix supports the representative capacities used by the fidelity allocator. If a cluster contains fewer tokens than the requested capacity, all of its tokens are retained. Constructing all prefixes costs $O(n_j m d_\phi)$ for maximum capacity $m \leq 16$, whereas constructing rate-specific synthetic subcluster centroids by I K-means iterations costs $O(I n_j m d)$ before the associated value reductions.

A.4 Empirical Capacity-to-Output Fidelity

The analysis above controls the sketched first-moment residual, but does not imply that every capacity increment must reduce every query's softmax output error. We therefore directly evaluate the deployed nested prefixes on two real Wan T2V-14B samples. Starting from the first sparse denoising step, we use ten consecutive timesteps, eight Transformer layers, all 40 attention heads, and eight real queries per head. For each of the resulting 160 attention calls, we hold Q/K/V tensors, Q/K clusters, exact regions, mean regions, and the representative sequence fixed, and set every intermediate region to capacity 2, 4, 8, or 16. Dense attention over the same Q/K/V tensors is the reference, yielding 51,200 matched output vectors.

Table 6: Attention-output error versus nested representative capacity. Mean MSE is pooled over matched output vectors; P95 and P99 are macro averages of the corresponding within-call quantiles.

Capacity	Mean MSE ↓	Macro P95 ↓	Macro P99 ↓
2	8.43×10^{-4}	1.40×10^{-3}	7.29×10^{-3}
4	6.43×10^{-4}	8.93×10^{-4}	5.43×10^{-3}
8	4.54×10^{-4}	5.48×10^{-4}	3.10×10^{-3}
16	3.23×10^{-4}	3.89×10^{-4}	1.85×10^{-3}

The three upgrades reduce error for 93.7%, 95.7%, and 63.0% of individual output vectors, respectively. When each attention call is treated as one unit, mean MSE improves in 149/160, 159/160, and 150/160 calls. Average MSE decreases monotonically for both samples, every evaluated timestep, and every evaluated layer. These results establish a strong aggregate capacity–fidelity trend without claiming per-query monotonicity, which softmax coupling does not guarantee.

A.5 Globally Normalized Fusion Across Fidelity Levels

Each regional representation returns an unnormalized attention numerator $\widehat{A}_j(q)$ and partition mass $\widehat{Z}_j(q)$. These quantities, rather than independently normalized regional outputs, are additive. If all

regions were materialized together, the result would be

$$A(q) = \sum_j \widehat{A}_j(q), \qquad Z(q) = \sum_j \widehat{Z}_j(q), \qquad o(q) = \frac{A(q)}{Z(q)}. \tag{36}$$

Thus, averaging independently normalized regional outputs would assign the wrong mass to each region.

This global normalization also yields an exact error decomposition. Let Z_j and $f_j = A_j/Z_j$ be the true regional mass and content, define $\pi_j = Z_j / \sum_k Z_k$, and write

$$\zeta_j = \frac{\widehat{Z}_j}{Z_j} - 1, \qquad \xi_j = \widehat{f}_j - f_j. \tag{37}$$

If $o = \sum_j \pi_j f_j$ and $\widehat{o}$ is the globally normalized approximate output, direct expansion gives

$$\widehat{o} - o = \frac{\sum_j \pi_j \left[(1+\zeta_j)\xi_j + \zeta_j(f_j - o)\right]}{1 + \sum_j \pi_j \zeta_j}. \tag{38}$$

The denominator is positive because it equals $\sum_j \widehat{Z}_j / \sum_j Z_j$. Equation 38 separates regional content error from cross-region reweighting caused by mass error: mean calibration primarily reduces ζ_j, intermediate representatives affect both ζ_j and ξ_j, and exact regions set both to zero.

The same decomposition motivates the variation term in the lightweight allocator. Within one region, let $t_x = q^\top(k_x - \bar{k}_j)/\sqrt{d}$, $\rho = \max_x |t_x|$, and define $\sigma_K^2 = \mathbb{E}\|K - \bar{k}_j\|_2^2$ and $\sigma_V^2 = \mathbb{E}\|V - \bar{v}_j\|_2^2$. Since $\mathbb{E}t = 0$, Jensen's inequality, $|e^t - 1| \le e^\rho |t|$, and Cauchy–Schwarz imply

$$\|f_j(q) - \bar{v}_j\|_2 \le e^\rho \sqrt{\mathbb{E}t^2}\, \sigma_V \le e^\rho \frac{\|q\|_2}{\sqrt{d}} \sigma_K \sigma_V. \tag{39}$$

Thus regional contribution and joint K/V variation naturally enter a proxy for the risk of using a mean. CoFiRA uses centroid contribution and normalized log-variations to keep this signal inexpensive; it remains a routing surrogate, not a formal error bound or a solution to global rate–distortion optimization.

Fixed-histogram allocation principle. To connect regional error scales to capacity placement, fix one query cluster, its mean and exact routes, and the multiset of capacities assigned to its J intermediate regions. Consider the separable surrogate

$$F(\mathbf{r}) = \sum_{j=1}^{J} a_j h(r_j), \qquad a_j \ge 0, \tag{40}$$

where a_j collects the region's contribution and response-error scale and $h(r)$ is non-increasing in capacity. This is a controlled surrogate motivated by Equations 38 and 39, not an assertion that the exact softmax error is separable. Under a fixed capacity multiset, F is minimized by matching larger capacities to larger a_j. Indeed, if $a_i \ge a_j$ but $r_i < r_j$, swapping the two assignments changes the objective by

$$F_{\text{before}} - F_{\text{after}} = (a_i - a_j)\big[h(r_i) - h(r_j)\big] \ge 0. \tag{41}$$

Repeatedly removing such inversions yields the sorted assignment.

The same argument quantifies the effect of using a proxy. Let $s_j = R_{ij}$, let $\widehat{\mathbf{r}}$ be the descending score–capacity matching, and let $\mathbf{r}^\star$ be the assignment sorted by the unknown a_j. For any scale $c > 0$,

$$F(\widehat{\mathbf{r}}) - F(\mathbf{r}^\star) \le \Delta h \sum_{j=1}^{J} |a_j - c s_j|, \qquad \Delta h = \max_r h(r) - \min_r h(r). \tag{42}$$

To see this, decompose $a_j = c s_j + (a_j - c s_j)$: score sorting is optimal for the first term, while each coefficient residual changes by at most Δh. Thus, sorting is exact when the proxy recovers regional error scales up to a positive factor, and its surrogate loss degrades continuously with score mismatch. This result concerns only placement under a fixed capacity histogram; it does not optimize the

histogram, exact promotion, or final-video error. The intermediate-only shuffle in Table 9 tests this placement principle while holding the nominal capacity histogram fixed.

The packed kernels realize the same regional quantities through additive logit biases. A mean token receives $\log n_j + \delta_j$, each of the $\hat{r}$ representatives receives $\log(n_j/\hat{r})$, and each exact token receives zero bias. These choices reproduce Equations 5 and 7 without explicitly materializing their cardinality factors.

The runtime packs mean and representative regions into an approximate stream and places exact regions in a separate variable-length stream. For stream $b \in \{\text{approx}, \text{exact}\}$, let $o_b = A_b/Z_b$ be its normalized output and $L_b = \log Z_b$ its log-partition value. The two streams are merged as

$$L = \log \sum_b e^{L_b}, \qquad o = \sum_b e^{L_b - L} o_b. \tag{43}$$

An empty stream is assigned $L_b = -\infty$ and contributes zero weight. Because $e^{L_b - L} = Z_b / \sum_{b'} Z_{b'}$, this expression is algebraically identical to summing all regional numerators and partition masses before normalization. It therefore applies one global normalization across the exact and surrogate regional contributions while allowing them to be evaluated by separate heterogeneous kernels.

For conditional models, CoFiRA approximates only video-to-video attention; image-conditioning branches in Wan I2V and text interactions in HunyuanVideo remain exact.

B Experimental Setup and Diagnostic Protocols

B.1 Cross-Modal Oracle Protocol

Figure 1 measures the cumulative exact attention mass retained after ranking candidate units by their dense-attention contribution. For text, we run Qwen2.5-7B (Qwen Team, 2024) on a 4,096-token context and collect final-query attention distributions from all heads and layers over 16 decoding steps. For image-conditioned decoding, we similarly evaluate the Qwen2.5-VL-7B-Instruct vision-language model (Bai et al., 2025) using an image represented by at most 1,024 visual tokens. Individual keys serve as candidate units in both settings.

For video, we evaluate ten Wan2.1-T2V-14B generations at 720×1280 resolution with 81 frames. Across all 50 denoising timesteps, we sample 32 queries from each of the 40 transformer layers and aggregate all attention heads. Individual keys are ranked by their exact dense-attention probabilities for the token-level curve in Figure 1. Each curve reports cumulative retained attention mass averaged over all collected query–head distributions.

This comparison characterizes oracle attention concentration in three representative inference settings targeted by sparse attention: text decoding, image-conditioned decoding, and video generation.

Practical video attention kernels operate on executable regions rather than independent tokens. We therefore additionally partition queries and keys into 100 and 500 clusters, aggregate exact dense-attention probabilities within each key cluster, and rank the resulting cluster masses by an oracle. As shown in Figure 6, expanding token selections to executable clusters further increases the density required to retain the same attention mass.

B.2 Evaluation Sets and Frozen Sample Lists

Wan text-to-video. We evaluate Wan2.1-T2V on the complete 600-prompt Penguin benchmark; therefore, no subset selection is required.

HunyuanVideo text-to-video. Because Penguin does not provide category annotations for stratified sampling, we uniformly sample 100 prompts without replacement using selection seed 0. The sampled row indices are sorted to ensure a deterministic execution order and frozen before evaluation. All methods use generation seed 0. The selected zero-based Penguin row indices are:

1, 14, 33, 39, 41, 42, 59, 63, 64, 70, 75, 82, 91, 93, 95, 97, 101, 103, 111, 127, 133, 142, 143, 145, 150, 153, 166, 182, 188, 191, 193, 195, 208, 209, 220, 223, 227, 241, 244, 247, 249, 256, 265, 266, 281, 282, 288, 294, 297, 298,

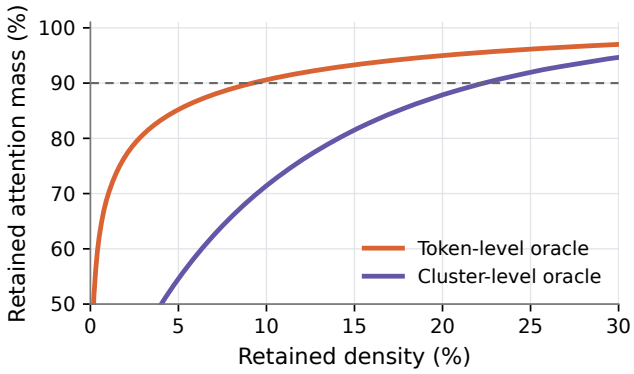


Figure 6: Token- and cluster-level oracle concentration for video attention. Executable clustering further increases the density required to retain the same attention mass.

308, 310, 313, 317, 323, 324, 327, 328, 333, 338, 340, 341, 358, 362, 366, 394, 400, 406, 408, 414, 429, 430, 444, 453, 455, 458, 461, 469, 483, 487, 488, 497, 501, 503, 504, 505, 516, 520, 523, 533, 545, 547, 548, 561, 565, 569, 573, 577, 591, 599.

Wan image-to-video. The VBench-I2V metadata contains 1,118 entries organized by task family, making the first 100 entries an unrepresentative subset. We therefore construct a fixed 100-sample subset by stratifying over subject, background, and seven camera- motion categories, sampling without replacement within each stratum, and retaining 100 distinct source images. The resulting subset contains 22 subject, 10 background, and 68 camera-motion examples. We use selection seed 20260812 and generation seed 0. The selected zero-based metadata rows are:

5, 30, 34, 42, 52, 58, 71, 73, 89, 106, 112, 126, 147, 154, 163, 191, 199, 213, 217, 229, 247, 249, 260, 265, 276, 293, 302, 311, 317, 321, 332, 346, 352, 371, 381, 388, 402, 408, 417, 436, 447, 452, 466, 485, 496, 507, 526, 533, 543, 548, 552, 561, 568, 578, 595, 608, 639, 644, 654, 659, 671, 675, 702, 704, 726, 740, 746, 765, 769, 790, 795, 803, 808, 822, 837, 847, 849, 856, 888, 896, 913, 917, 929, 931, 946, 949, 951, 964, 967, 996, 1000, 1004, 1018, 1022, 1024, 1038, 1041, 1053, 1078, 1096.

Each source ID follows the format `vbench_i2v_XXXX`. We use the official 16:9-cropped VBench-I2V image suite. During inference, each input image retains its aspect ratio under a 720×1280 pixel-area budget, with both dimensions rounded to multiples of 16 as required by the model. Dense attention and all sparse methods use identical input geometry and native, non-tiled VAE decoding.

B.3 Method Configurations

Table 7 records the operating points used in the main comparison. Baselines follow their paper settings and official implementations. PISA is not included in the quantitative video comparison because, at the time of submission, its official repository had not released inference code for Wan or HunyuanVideo. For context, under its own HunyuanVideo protocol, PISA reports nearly the same latency as SVG2 (641 vs. 649 seconds), with lower PSNR and SSIM (26.17/0.840 vs. 26.40/0.848) and marginally better LPIPS (0.106 vs. 0.109). These results place PISA and SVG2 at broadly comparable quality–latency operating points under the reported protocol.

For SVG-EAR, p controls the computation budget of its error-aware exact-region route. PSA uses its official `psa_balanced` configuration with Gilbert reordering and threshold-based allocation among its pooling levels; the optional similarity mask is disabled by default. We additionally evaluate this option at the same 50% level-1 budget. On the 20-sample sweep, it improves PSNR/SSIM/LPIPS from 22.27/0.738/0.228 to 23.17/0.784/0.197, while increasing H100 runtime by approximately 7%, confirming a fidelity–latency trade-off rather than a free improvement. Even CoFiRA's slowest evaluated point (14% exact budget) remains 22% faster on H100 (1081.3 versus approximately 1390 seconds) while achieving 6.88 dB higher PSNR (30.05 versus 23.17 dB). CoFiRA uses nested 2/4/8/16 representative rates, a 32-channel moment sketch, and mean-calibration coefficient 1.0. Following the baseline convention, all clustering-based methods keep the first 3% of Transformer layers dense throughout denoising, corresponding to one layer in every evaluated model. All methods use BF16. Wan T2V and I2V generate 81 frames with 50 and 40

Table 7: Main experimental configurations. Exact ranges denote the noise-aware low-to-high schedule; the value in parentheses is its nominal mean.

Task	Method	Dense warm-up	C_q	C_k	Selector / rate	Principal setting
Wan T2V	SVG2	15/50	100	500	top-p	$p = 0.90$
	SVOO	10/50	256	1024	dynamic	layer budget 15–20%, flow shift 5.0
	PSA	15/50	–	–	pyramid	1/2/4/8/skip pooling, 128×128 blocks
	SVG-EAR	10/50	300	1000	error-aware	$p = 0.85$
	CoFiRA	15/50	550	2500	multi-rate	exact 3–6% (4.5%), routed 24%
Wan I2V	SVG2	12/40	100	500	top-p	$p = 0.90$
	SVOO	8/40	256	1024	dynamic	layer budget 10–15%
	SVG-EAR	8/40	300	1000	error-aware	$p = 0.85$
	CoFiRA	12/40	600	2500	multi-rate	exact 4–8% (6%), routed 24%
Hunyuan T2V	SVG2	15/50	100	500	top-p	$p = 0.90$
	SVOO	5/50	256	1024	dynamic	layer budget 5–10%
	SVG-EAR	10/50	400	1000	error-aware	$p = 0.85$
	CoFiRA	12/50	700	2500	multi-rate	exact 7–13% (10%), routed 21%

Table 8: Prompt-level paired-bootstrap uncertainty for CoFiRA's PSNR gain over SVG-EAR in the main experiments.

Task	Prompts	Mean PSNR gain (dB)	95% CI (dB)
Wan T2V	600	1.09	[0.98, 1.21]
Wan I2V	100	0.42	[0.15, 0.69]
HunyuanVideo T2V	100	0.92	[0.67, 1.19]

denoising steps, respectively; HunyuanVideo generates 129 frames with 50 steps. Wan uses UniPC with flow prediction, a flow shift of 5.0, and classifier-free guidance scale 5.0. HunyuanVideo uses flow-matching Euler sampling with shift 7.0 and guidance scale 6.0.

B.4 Prompt-Level Uncertainty

We quantify uncertainty in the main paired-reference comparison by resampling paired prompt-level PSNR differences between CoFiRA and SVG-EAR, the strongest paired-reference baseline in all three settings. Table 8 reports 95% paired-bootstrap confidence intervals computed from the unrounded per-prompt metrics. All three intervals exclude zero.

The consistently positive intervals show that CoFiRA's average gains are stable under prompt resampling rather than tied to one favorable evaluation subset.

B.5 Capacity-Allocation Controls

Table 9 isolates SVG-EAR's binary exact-or-mean representation within the otherwise unchanged CoFiRA pipeline. Because representative capacity and exact budget take discrete values, their PFLOPs cannot in general be matched exactly. We therefore choose the nearest feasible integer-capacity and exact-budget settings and report full-trajectory computation. The resulting operating points have comparable total computation. Adaptive allocation improves mean PSNR over fixed capacities 5 and 6 by 0.48 and 0.41 dB, respectively, while also improving SSIM and LPIPS; the fixed-5 and fixed-6 PSNR gains have 95% paired-bootstrap CIs of 0.32–0.64 and 0.24–0.56 dB. Thus, the gain does not arise solely from adding one intermediate operating point; assigning different representative capacities according to regional sensitivity provides an additional benefit.

We next shuffle capacity assignments while preserving their nominal histograms and near-matched executed PFLOPs. The unrounded measured totals differ slightly because requested representative capacities can be clipped by realized key-cluster sizes and, for the full shuffle, exact routes inherit different key counts; all three totals round to 373.3 PFLOPs at the reported one-decimal precision. Shuffling only the intermediate $2/4/8/16$ capacities, while leaving mean and exact regions unchanged, reduces PSNR from 27.70 to 27.05 dB (95% paired-bootstrap CI for the drop: 0.46–0.84 dB) and degrades both SSIM and LPIPS. Shuffling the full allocation causes a larger drop to 17.70

Table 9: Representative-capacity and routing controls on Wan T2V. All rows share Q/K partitions, warm-up, and execution. Exact budgets are adjusted only where needed to obtain the nearest feasible compute match; Total PFLOPs are measured from the routes actually executed across the full denoising trajectory. Fixed policies use the indicated number of real K/V tokens per intermediate region.

Representation policy	PSNR ↑	SSIM ↑	LPIPS ↓	Total PFLOPs ↓
Exact/Mean	27.17	0.846	0.165	368.0
Fixed capacity 5	27.22	0.852	0.156	371.7
Fixed capacity 6	27.29	0.857	0.151	373.8
Intermediate-only shuffle	27.05	0.849	0.158	373.3
Full-allocation shuffle	17.70	0.429	0.509	373.3
Adaptive multi-rate	**27.70**	**0.870**	**0.139**	373.3

dB (95% paired-bootstrap CI for the 9.99 dB PSNR drop: 9.04–10.95 dB), because it also removes the placement of exact attention from the most sensitive regions. Together, these controls show that neither the amount of compute nor the proportions of fidelity levels are sufficient: sensitivity-guided placement improves over fixed-capacity and shuffled assignments at matched computation.

Across the 20 paired samples, multi-rate representation improves PSNR by 0.52 dB over Exact/Mean (95% paired-bootstrap CI: 0.20–0.91 dB).

Noise-aware scheduling. The schedule introduced in Section 3.4 redistributes the exact-attention budget across denoising timesteps according to noise level. We compare it with a fixed per-step budget while holding the allocator, fidelity ladder, clustering, and execution backend unchanged. Both schedules require 373.3 PFLOPs of actual total computation. Noise-aware scheduling improves PSNR/SSIM/LPIPS from 27.62/0.868/0.141 to 27.70/0.870/0.139 across the 20 paired samples. It therefore provides a modest 0.08 dB mean PSNR gain by refining the temporal distribution of a fixed computation budget.

B.6 Representative-Construction Accuracy and Cost

CoFiRA constructs nested $2/4/8/16$ real-K/V representatives through one lightweight residual first-moment selection. A natural alternative is rate-specific subclustering: for an allocated capacity r, rerun r-means inside the routed key cluster and use the resulting centroids as synthetic K/V representatives. Such centroids summarize subgroups rather than selecting existing tokens and may reduce quantization error, but they do not guarantee lower attention-output error because the final approximation also depends on the queries, softmax weights, and joint K/V structure. We therefore compare the accuracy and construction cost of these choices under matched inference states, including uniformly spaced real-token sampling as a low-cost baseline.

The diagnostic uses two Wan T2V-14B samples and covers all 35 sparse denoising steps, all 39 sparse Transformer layers, both classifier-free-guidance branches, all 75,600 query positions, and all 40 attention heads. This gives 5,460 matched attention calls. Within each call, we hold Q/K/V tensors, Q/K clusters, regional fidelity assignments, exact and mean regions, and the allocated $2/4/8/16$ capacities fixed. Only the intermediate representative construction changes. Dense attention over the same Q/K/V tensors provides the reference, and we aggregate squared error over every evaluated output element.

We profile a standard Wan T2V-14B execution point with 40 heads and 98,353 nonempty outer key clusters. All constructors start from identical outer-clustered K/V tensors, cluster statistics, and regional assignments, and materialize the K/V banks and multiplicities required by the allocated $2/4/8/16$ capacities. CoFiRA constructs one maximum-capacity nested sequence whose prefixes serve all rates, whereas the synthetic baseline independently fits the required rate-specific centroid banks. Construction latency excludes outer Q/K clustering, routing, attention, partition merging, and data loading. After three warm-up runs, we report the median of 20 CUDA-event measurements.

Uniform sampling reduces Wan T2V generation time from 1037.1 to 1024.4 seconds (1.2%), but its output MSE is 2.2% higher than CoFiRA's. Its selection complexity is zero because evenly spaced

Table 10: Attention-output error and representative-construction cost. Here n_j is cluster size, m is maximum capacity, d is head dimension, d_ϕ is sketch dimension, and I is the number of inner K-means iterations. Selection complexity excludes the mandatory materialization of the output representative bank. Attention latency measures the complete profiled sparse-attention call. Generation time follows the same end-to-end H100 timing protocol as Table 1.

Intermediate representation	Output MSE ↓	Construction latency ↓	Attention latency ↓	Time (s) ↓	Selection complexity ↓
Uniform real K/V	9.11×10^{-4}	**5.42 ms**	**128.43 ms**	**1024.4 s**	0
CoFiRA nested real K/V	8.91×10^{-4}	10.06 ms	133.07 ms	1037.1 s	$O(n_j m d_\phi)$
Synthetic subcluster centroids	$\mathbf{7.51 \times 10^{-4}}$	235.30 ms	358.31 ms	1652.0 s	$O(I n_j m d)$

indices are available in closed form; the measured 5.42 ms is the unavoidable cost of allocating, gathering, and writing the K/V bank and multiplicities. It is deterministic for a fixed token order, but is not permutation-invariant: its representatives depend on the incidental within-cluster ordering rather than the K/V distribution, providing no control over moment discrepancy or mode coverage. Synthetic centroids reduce local attention-output MSE by 15.8% in this diagnostic, indicating that additional within-cluster structure can improve the approximation. However, they require rerunning K-means inside every routed key cluster for each capacity, followed by additional K/V reductions to form the centroids, with complexity $O(I n_j m d)$. CoFiRA instead constructs one nested real-token prefix with a 32-channel sketch, reuses its prefixes across all capacities, and costs $O(n_j m d_\phi)$. Measured construction latency is 10.06 ms for CoFiRA and 235.30 ms for synthetic centroids, making the latter 23.4× slower. The corresponding complete sparse-attention latencies are 133.07 and 358.31 ms, respectively, a 2.69× difference. Thus, although synthetic centroids achieve lower local output error, their construction cost is unsuitable for the online acceleration path. Real K/V representatives provide a practical accuracy–cost trade-off without rate-specific inner clustering. The fixed-capacity results in Table 9 attribute the principal fidelity gain to the multi-level representation space and sensitivity-guided regional allocation, while residual matching provides a lightweight, deterministic construction for the assigned capacities.

B.7 Comparison and Sweep Protocols

The reconstruction metrics in Table 2 and the ablations in Tables 3 and 9, together with the operating-point curves, use 20 Penguin samples with dataset IDs 1, 14, 33, 62, 86, 87, 104, 111, 136, 145, 193, 226, 311, 316, 336, 339, 352, 354, 365, and 383, with generation seed 0. Runtime latency uses the saved workload specified below.

Figure 5 evaluates five operating points for each method on this 20-sample set. CoFiRA uses exact budgets of 2%, 5%, 8%, 11%, and 14%; noise-aware scheduling is disabled during this sweep so that the specified budget remains fixed across sparse steps. SVG2 uses top-p values of 0.75, 0.80, 0.85, 0.90, and 0.95, while SVG-EAR uses 0.65, 0.75, 0.85, 0.90, and 0.95. SVOO varies its layer-budget range over 5–10%, 10–15%, 15–20%, 20–25%, and 25–30%. PSA varies its cumulative level-1 importance budget over 20%, 30%, 40%, 50%, and 60%; the plotted exact density is the resulting unpooled level-1 interaction share rather than this control value.

Figure 5(a) reports the H100 generation time at each operating point. Figure 5(b) reports the measured exact- attention density at the same operating points.

Figure 7 complements the exact-density analysis by plotting the same quality measurements against representation density, which also counts the mean and real K/V representatives used by approximate regions. CoFiRA retains the strongest quality–density trade-off under this complete representation accounting.

B.8 Router-to-Error Correlation

We further examine whether CoFiRA's lightweight routing score reflects actual regional approximation risk rather than merely reproducing the attention- contribution ranking. Under the paper's 550×2500 clustering configuration, we analyze two real Wan T2V-14B samples over all 35 sparse denoising steps, 39 sparse Transformer layers, both classifier-free-guidance branches, and all 40 at-

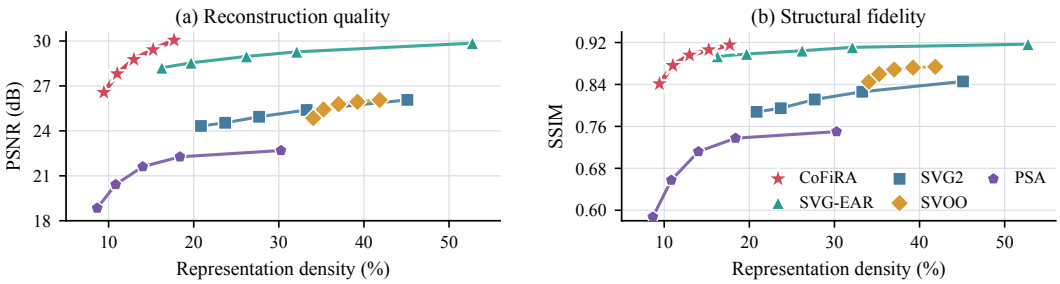


Figure 7: PSNR and SSIM versus sparse-step representation density on the same 20 Wan T2V samples.

Table 11: Correlation between router signals and measured regional error or adjacent-rate refinement benefit, with H100 score-formation latency. Higher is better for correlation and lower is better for latency.

Router signal	Mean error	Adjacent-rate gain	Latency
CoFiRA	<u>0.9358</u>	**0.6508**	**1.73 ms**
Contribution only	0.9186	<u>0.6456</u>	–
Complexity only	0.1618	0.0816	–
SVG-EAR error score	**0.9443**	0.6156	8.19 ms

tention heads. For each sampled query-cluster row, we measure the relative attention-output error induced by replacing each query-cluster/key-cluster region with its calibrated mean. We also measure the error reduction from each adjacent representative-capacity upgrade ($r2 \to r4$, $r4 \to r8$, and $r8 \to r16$). Spearman rank correlation is computed independently within each row and then averaged over 218,400 head-level observations. The adjacent-rate result is the mean correlation across the three upgrades.

We first isolate the roles of CoFiRA's two routing signals. Query-conditioned contribution alone already predicts mean-approximation error and adjacent-rate benefit accurately, whereas K/V variation alone is insufficient to decide whether a region warrants additional computation. Their combination raises the two correlations from 0.9186 and 0.6456 to 0.9358 and 0.6508. This agrees with the final-video ablation in Table 3: contribution-only, variation-only, and joint routing obtain 27.44, 18.52, and 27.70 dB PSNR, respectively. Together, the two diagnostics identify query-conditioned contribution as the primary routing signal and K/V variation as a lightweight refinement for regions with similar contribution but different compressibility.

SVG-EAR likewise recognizes that within-cluster coherence affects the accuracy of mean compensation, but operationalizes this observation differently. It uses a query centroid to probe individual keys against their key centroid and ranks regions by an error-to-cost score for binary Exact/Mean routing. Its score formation therefore costs $O(C_q N_k d)$ because the per-key probe is repeated for every query cluster. CoFiRA instead combines cluster-level contribution with linearly accumulated K/V variation, requiring only $O(N_k d + C_q C_k d)$ score formation. This avoids repeated per-key probing while retaining both query-conditioned importance and within-cluster compressibility, and uses the resulting score to allocate progressively larger real- representative capacities rather than only choosing between Exact and Mean.

On the same Wan T2V-14B router shape, complete score formation takes 1.73 ms for CoFiRA and 8.19 ms for SVG-EAR, making CoFiRA's score formation 4.7× faster. This comparison excludes clustering and downstream route construction for both methods. Despite this substantially lower cost, CoFiRA remains close to SVG-EAR's direct estimator in correlation with measured mean-approximation error (0.9358 versus 0.9443). Adjacent-rate gain instead measures whether a score ranks regions by the actual error reduction obtained from each capacity upgrade, averaged over $r2 \to r4$, $r4 \to r8$, and $r8 \to r16$. This criterion is more closely aligned with CoFiRA's multi-rate objective, on which CoFiRA reaches 0.6508 correlation versus 0.6156 for SVG-EAR. Thus, the

Table 12: Attention-level ablation of the mean-mass correction. Improved fractions pair regions or output vectors, respectively, against the uncorrected mean.

Metric	Uncorrected	Corrected	Reduction	Improved
Mean abs. log-partition error	0.6571	**0.3299**	49.8%	97.29%
Routed attention-output MSE	7.217×10^{-4}	$\mathbf{6.750\times10^{-4}}$	6.47%	83.01%

lightweight score preserves the mean-risk signal at much lower cost while aligning more closely with incremental capacity allocation.

Mean-mass calibration and proxy limits. We directly switch the mean-mass correction on and off while holding the real Q/K/V tensors, clustering, routes, and all other approximations fixed. The diagnostic covers 240 Wan T2V-14B attention calls from two videos, spanning the first ten sparse denoising timesteps, 12 Transformer layers, all 40 heads, and 12 real queries per head. This yields 115,200 paired output vectors and 213.3 million mean-region observations. We evaluate both its target quantity, absolute regional log-partition error, and routed attention-output MSE against exact attention under otherwise identical routes.

The correction nearly halves its target error and also modestly reduces routed attention-output MSE. At the attention-call level, 96.25% of calls improve; the mean relative MSE reduction is 9.72% (paired-bootstrap 95% CI: 8.75–10.71%). In a separate proxy diagnostic, scalar K variation has a mean within-row Spearman correlation of 0.391 with exact query-dependent logit variance, characterizing it as a lightweight signal for routing. Among non-singleton query–key regions, those in the lowest decile of V variation and highest decile of K variation comprise 0.61%. Their mean single-region output error is 0.70× the overall mean. This subset is somewhat enriched in high-error regions (17.9% fall in the top error decile, versus 10% overall), but accounts for only 1.06% of that decile. Together, these attention-level measurements show that calibration consistently improves its target quantity and routed attention accuracy, while the variation score provides a low-cost routing signal.

C Additional Evaluation and Analysis

C.1 Metric and Timing Protocols

PSNR, SSIM, and LPIPS compare each sparse output with its paired dense video under the same prompt, conditioning image when applicable, and generation seed. Metrics are aggregated over frames and then over samples. VBench evaluates reference-free quality attributes independently of dense-reference fidelity. All methods within a task share the same resize, crop, frame, and VAE protocol. For T2V, the official custom-input protocol supports subject consistency, background consistency, motion smoothness, dynamic degree, aesthetic quality, and imaging quality. We evaluate this complete six-dimension set and report its unweighted macro average; for I2V, we report the macro average over the task-applicable VBench-I2V dimensions. Tables 13 and 14 provide the complete dimension-wise results.

Localized residual metric. Removing multi-rate representations reduces the computation of the Exact/Mean variant. To compensate for this reduction, we increase its exact-attention budget from the 4.5% used by CoFiRA to 6%. The resulting full-trajectory computation is 368.0 PFLOPs for Exact/Mean and 373.3 PFLOPs for CoFiRA. These are comparable operating points under discrete exact budgets and representative capacities.

Global reconstruction metrics can obscure severe errors confined to a small spatial region. We therefore evaluate Exact/Mean and CoFiRA on the same 20 fixed Wan T2V samples using a patch-level residual. Each sparse video is paired with the dense video generated from the same prompt and seed. RGB frames are scaled to $[0,1]$, aligned to a common frame count, and divided into non-overlapping 32×32 patches. For patch p in frame t, we define

$$e_{t,p} = \mathrm{L1}_{t,p} + 0.5\big(1-\mathrm{SSIM}_{t,p}\big) + \mathrm{L1}_p\Big[(Y_t - Y_{t-1}) - (\widehat{Y}_t - \widehat{Y}_{t-1})\Big], \tag{44}$$

Table 13: Dimension-wise VBench results for T2V. Bold and underlined entries mark the best and second-best sparse methods.

Model	Method	Subject	Background	Motion	Dynamic	Aesthetic	Imaging	Macro
Hunyuan T2V	Dense	94.77	95.54	99.52	30.00	57.37	65.42	73.77
	SVG2	94.26	95.30	99.48	24.00	55.51	61.63	71.69
	SVOO	94.59	95.22	99.49	**32.00**	**57.19**	**65.05**	**73.92**
	SVG-EAR	<u>94.62</u>	<u>95.51</u>	<u>99.51</u>	<u>31.00</u>	56.96	64.95	73.76
	CoFiRA	**94.80**	**95.54**	**99.52**	<u>31.00</u>	<u>56.97</u>	<u>65.01</u>	<u>73.81</u>
Wan T2V	Dense	95.81	96.29	98.72	43.67	59.73	68.19	77.07
	SVG2	95.36	95.91	98.64	36.83	58.78	67.23	75.46
	SVOO	<u>95.70</u>	**96.21**	<u>98.71</u>	**43.17**	<u>59.69</u>	<u>68.14</u>	**76.94**
	PSA	90.90	93.41	**98.74**	<u>42.83</u>	**59.75**	**69.01**	75.77
	SVG-EAR	<u>95.70</u>	96.18	<u>98.71</u>	<u>42.83</u>	59.65	68.02	<u>76.85</u>
	CoFiRA	**95.77**	<u>96.19</u>	<u>98.71</u>	41.33	59.53	67.74	76.55

Table 14: Dimension-wise VBench-I2V results. Parentheses give the number of samples to which each dimension applies. I2V-Subj./Back. denote input-image subject/background fidelity; Cam. denotes camera motion and Flicker denotes temporal flickering. Bold and underlined entries mark the best and second-best sparse methods.

Method	I2V-Subj. (22)	I2V-Back. (10)	Cam. (68)	Subject (22)	Back. (10)	Motion (22)	Dynamic (22)	Flicker (32)	Aesthetic (32)	Imaging (32)	Macro
Dense	97.36	99.27	29.41	93.03	96.93	98.19	59.09	97.16	61.77	72.86	80.51
SVG2	96.54	98.98	**32.35**	92.31	96.67	98.10	59.09	97.13	60.27	72.39	80.38
SVOO	**97.29**	**99.24**	27.94	<u>92.80</u>	**96.76**	98.16	59.09	97.14	61.33	<u>72.72</u>	80.25
SVG-EAR	<u>97.22</u>	<u>99.23</u>	<u>30.88</u>	**92.89**	<u>96.70</u>	<u>98.19</u>	**63.64**	<u>97.17</u>	<u>61.45</u>	72.66	<u>81.00</u>
CoFiRA	97.15	99.20	**32.35**	<u>92.80</u>	96.58	**98.20**	**63.64**	**97.20**	**61.54**	**72.76**	**81.14**

where the three terms measure local RGB reconstruction error, structural distortion, and temporal-change error, respectively. SSIM is computed on grayscale patches with a 7×7 window, and the temporal term is set to zero for the first frame. Mean patch error averages $e_{t,p}$ over all valid patches, frames, and samples, reflecting the overall degree of localized distortion. Residual exceedance rate is the fraction of patches with $e_{t,p} > 0.3$, measuring the frequency of large localized deviations.

All latency measurements use one NVIDIA H100. Pipeline time includes denoising and VAE decoding and excludes model loading, video export, and metric evaluation. We synchronize the GPU at both timing boundaries and report the median of three complete runs. Dense attention uses exact FlashAttention-3.

Exact-attention density is the fraction of dense query–key interactions computed using the original K/V tokens. Representation density, reported as Density in the main table, measures the equivalent query–key interactions actually represented during sparse attention. For a query–key cluster pair, it counts one representative for the mean level, the selected number of real K/V representatives for an intermediate level, and every original K/V token for exact attention, weighted by the corresponding query-cluster size. Thus, exact and representation density coincide for selection methods that discard all non-exact regions, whereas CoFiRA's representation density also includes its approximate representations. For PSA, a block assigned pooling factor h contributes $1/h$ of its dense interactions, while a skipped block contributes zero.

Table 5 replays a saved input from a real Wan T2V-14B sparse-attention call under the paper's 550×2500 clustering configuration. All rows use the same routing decisions and mathematically equivalent attention computation; only their execution organization differs. Latency covers the complete sparse-attention module, including QKV projection, Q/K normalization, RoPE, sparse attention, and output projection. We run six forwards, discard the first as warm-up, and report the median of the remaining runs after averaging over all 39 sparse layers.

Table 15: H100 latency profile of one complete CoFiRA sparse-attention module.

Stage	Latency
Shared projections and transforms	26.85 ms
Clustering and routing	31.92 ms
Representative construction	10.06 ms
Exact stream	21.99 ms
Approximate stream	31.51 ms
Partition merge	8.59 ms
Other	2.15 ms
Total	**133.07 ms**

Table 16: Per-GPU peak memory at the Wan T2V-14B operating points. Bold marks the lowest sparse-method peak.

Method	Peak memory (GiB) ↓
Dense	50.04
SVG2	51.68
SVOO	53.38
PSA	53.70
SVG-EAR	53.55
CoFiRA	**50.51**

Table 17: CoFiRA component-memory profile. Values are component-local maxima over all calls in the two stable profiled sparse steps.

Component	Transient peak (MiB)	Retained (MiB)
Clustering	2970	24
QKV permutation	2238	2238
Proxy scores	840	420
K/V reliability (each)	3	1
Route assignment	1081	105
Cluster means	788	49
Representative selection	9	6
Exact stream	752	750
Mean stream	804	750
Representative stream	752	750
Partition merge	750	750
Inverse permutation	738	738

C.2 Runtime and Memory Profiles

We additionally profile per-GPU memory at the Wan T2V-14B operating points in Table 7. All methods use BF16, 720×1280 resolution, 81 frames, and their main-table configurations. We restore the matched dense prefix and profile two consecutive sparse steps for each baseline, including the first sparse step and its one-time initialization; CoFiRA is profiled for three consecutive sparse steps to verify the stable regime. Dense attention is measured over the two corresponding denoising steps. Total device memory is sampled every 0.1 seconds on one NVIDIA RTX 6000D, and Table 16 reports the maximum over each complete profiling window.

Table 15 shows that clustering and routing and the approximate stream are the largest method-specific components. The runtime ablation therefore targets the two corresponding execution costs: packed evaluation avoids separate launches across fidelity levels, and device-side scheduling avoids repeated host synchronization for irregular route plans.

CoFiRA peaks only 0.48 GiB (0.96%) above dense inference and uses 1.17–3.18 GiB less peak memory than the other sparse methods. To identify where its additional memory is used, we place synchronized allocation boundaries around the components of every CoFiRA sparse-attention call. Each denoising step contains 78 measured calls: 39 sparse layers in each of the two classifier-free-guidance branches. Table 17 reports the maximum component-local allocation over stable sparse steps two and three. "Transient peak" is the highest live tensor allocation reached by a component minus the allocation already live at its entry; "retained" is the corresponding increase still live when that component returns. These quantities describe overlapping tensor lifetimes and therefore must not be summed.

The exact, mean, and representative rows are logical profiling categories: mean and representative computations share the single packed approximate execution stream, while exact attention uses the second stream. The low end-to-end peak follows from CoFiRA's allocation and execution layout, rather than from eliminating all temporary storage. Clustering has the largest local scratch requirement, but only 24 MiB remains live when it returns. QKV permutation and these attention components retain their outputs only within the current layer, so these tensors do not accumulate across layers or denoising steps. Routing is encoded by compact cluster-level integer rate codes and packed indices instead of a token-level attention layout, while the 10.7% representation density

Table 18: Sparse-phase efficiency decomposition across models and tasks.

Task	Method	Dense warm-up	Step Attn. PFLOPs	Density	Total PFLOPs
Wan T2V	SVG2	15/50	3.1269	0.317	440.2
	SVOO	10/50	3.5989	0.369	427.9
	PSA	15/50	**1.1927**	0.127	**372.5**
	SVG-EAR	10/50	2.5140	0.250	384.5
	CoFiRA	15/50	1.2154	**0.107**	373.3
Wan I2V	SVG2	12/40	3.0400	0.307	349.7
	SVOO	8/40	3.1008	0.314	326.3
	SVG-EAR	8/40	2.4806	0.246	306.5
	CoFiRA	12/40	**1.2114**	**0.107**	**298.5**
Hunyuan T2V	SVG2	15/50	2.9929	0.274	351.4
	SVOO	5/50	2.8484	0.260	**270.3**
	SVG-EAR	10/50	2.2725	0.224	285.3
	CoFiRA	12/50	**1.7211**	**0.125**	280.7

limits the K/V data and workspace consumed by the exact, mean, and representative components. Their outputs are combined through the log-sum-exp merge and released before the next layer. Consequently, common model weights and activations dominate the process footprint, leaving CoFiRA close to dense inference despite its clustering and multi-rate routing machinery.

Step Attn. PFLOPs measure the executed $QK^\top$ and AV operations in an average sparse denoising step. For CoFiRA, the value is averaged over the complete noise-aware sparse trajectory.

C.3 Controlled Analysis of Operating-Point Factors

The fidelity–efficiency operating point of a sparse-attention method is principally shaped by four algorithmic choices: its sparse-compute budget, regional representation and allocation policy, semantic-cluster granularity, and dense warm-up length. Because these choices are jointly designed and interact with one another, the following controlled comparisons examine their effects at explicitly reported operating points; they do not replace the complete method configurations selected by the respective authors. We therefore treat the main-table results, obtained using each method's prescribed configuration under a shared evaluation protocol, as the primary method-level comparison. Section 4.3 isolates representation and allocation design, while Figure 5 examines the response to compute budget. We next analyze cluster granularity and dense warm-up. Execution-backend gains are evaluated separately in the runtime ablation of Section 4.5.

C.3.1 Cluster Granularity

SVG-EAR is the strongest competing method in paired-reference fidelity and therefore provides the most informative baseline for examining whether CoFiRA's fidelity advantage arises merely from its finer semantic partition. We rerun SVG-EAR using both its original 300×1000 Q/K clustering and the finer 550×2500 configuration used by CoFiRA. Figure 8 reports the low-, middle-, and high-budget operating points from each sweep: exact budgets of 2%, 8%, and 14% for CoFiRA, and top-p values of 0.65, 0.85, and 0.95 for both SVG-EAR configurations. Increasing cluster granularity shifts the SVG-EAR curve upward and to the right. Finer clusters reduce within-cluster heterogeneity and make mean compensation more faithful, thereby improving reconstruction quality. However, they also introduce two sources of additional latency: the larger number of clusters increases clustering and routing overhead, while the resulting shorter and less regular attention workloads reduce sparse-kernel efficiency. At a fixed number of represented query–key interactions, the reported $QK^\top$ and AV FLOPs change little, so the rightward shift reflects these execution overheads, all of which are included in the reported latency.

This experiment isolates cluster granularity rather than equalizing the complete compute budgets of the two methods. For context, at the main-table operating points, CoFiRA uses 10.7% representation density and 1.22 attention PFLOPs per sparse step, compared with 25.0% and 2.51 PFLOPs for SVG-EAR. Even after adopting CoFiRA's finer partition, SVG-EAR requires 1316.6 seconds to reach 30.05 dB, versus 1081.3 seconds for CoFiRA at the same PSNR, reducing latency by 235.3 seconds, or 17.9%. At its highest budget, the finer SVG-EAR configuration attains 32.06 dB but requires 1938.5 seconds, corresponding to only $0.95\times$ the dense speed. Although this point achieves

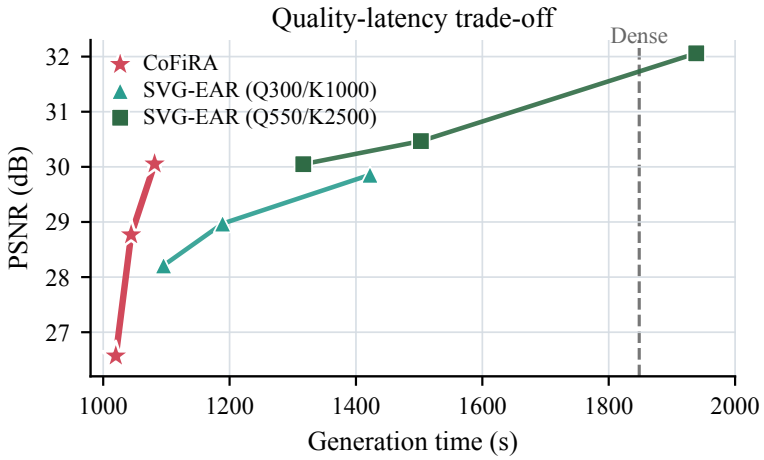


Figure 8: Quality–latency trade-off under original and finer SVG-EAR clustering. The dashed line marks dense generation latency.

Table 19: Zero-warmup operating-point comparison on HunyuanVideo-T2V-13B. CoFiRA uses a fixed 13% exact-attention budget.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	Time (s) ↓	Speedup ↑
Dense	–	–	–	1676.3	1.00×
SVG2	19.64	0.657	0.354	655.8	2.56×
SVOO	20.40	0.686	0.311	835.5	2.01×
SVG-EAR	22.68	0.798	**0.225**	740.9	2.26×
CoFiRA	**22.71**	**0.812**	0.227	**650.2**	**2.58**×

high fidelity, it no longer constitutes a viable operating point for an acceleration method. Finer clustering can therefore trade substantial additional runtime for higher SVG-EAR fidelity, but does not account for CoFiRA's more efficient fidelity–latency operating points.

C.3.2 Zero-Warmup Comparison

Dense warm-up and sparse-attention budgets jointly shape each method's fidelity–efficiency operating point. We additionally evaluate all methods with zero dense warm-up on HunyuanVideo-T2V-13B, applying sparse attention from the first denoising step. In this comparison, CoFiRA uses a fixed 13% exact-attention budget; the main comparison retains the prescribed warm-up and complete configuration of each method.

Under these zero-warmup operating points, CoFiRA and SVG-EAR achieve comparable paired-to-dense fidelity: 22.71 versus 22.68 dB PSNR, 0.812 versus 0.798 SSIM, and 0.227 versus 0.225 LPIPS. CoFiRA requires 650.2 rather than 740.9 seconds, reducing latency by 90.7 seconds, or 12.2%. Thus, at comparable fidelity, CoFiRA provides faster inference when sparse attention is applied throughout the complete denoising trajectory.

C.4 Robustness to the Method Seed

CoFiRA uses randomized initialization when forming its regional token groups. To isolate the resulting algorithmic randomness from video-generation stochasticity, we fix the prompt, generation seed, and all inference settings, and vary only the method seed over five runs. Against the same dense-attention output, PSNR, SSIM, and LPIPS are 24.42 ± 0.23 dB, 0.823 ± 0.004, and 0.151 ± 0.002, respectively. The small variation in this controlled case indicates that CoFiRA is stable to its internal random initialization when the generated content is held fixed.

D ADDITIONAL QUALITATIVE RESULTS

Figures 9– 11 compare representative frames produced from the same condition and generation seed. The examples span people, animals, indoor scenes, landscapes, and stylized motion rather than being selected for a single visual category.

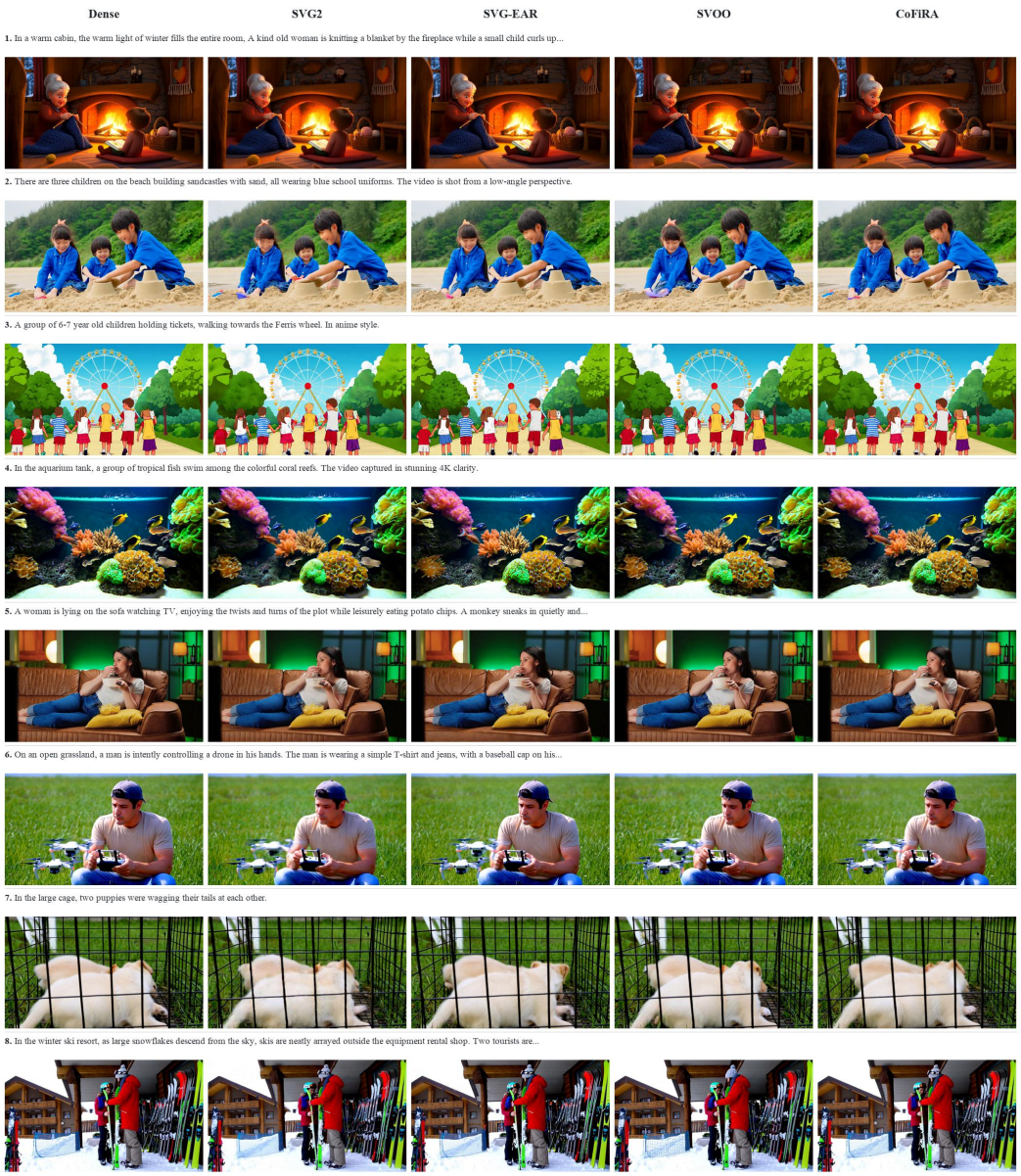


Figure 9: Additional qualitative comparisons on Wan2.1-T2V-14B. Each row uses one prompt and displays an aligned frame from dense attention, SVG2, SVG-EAR, SVOO, and CoFiRA.

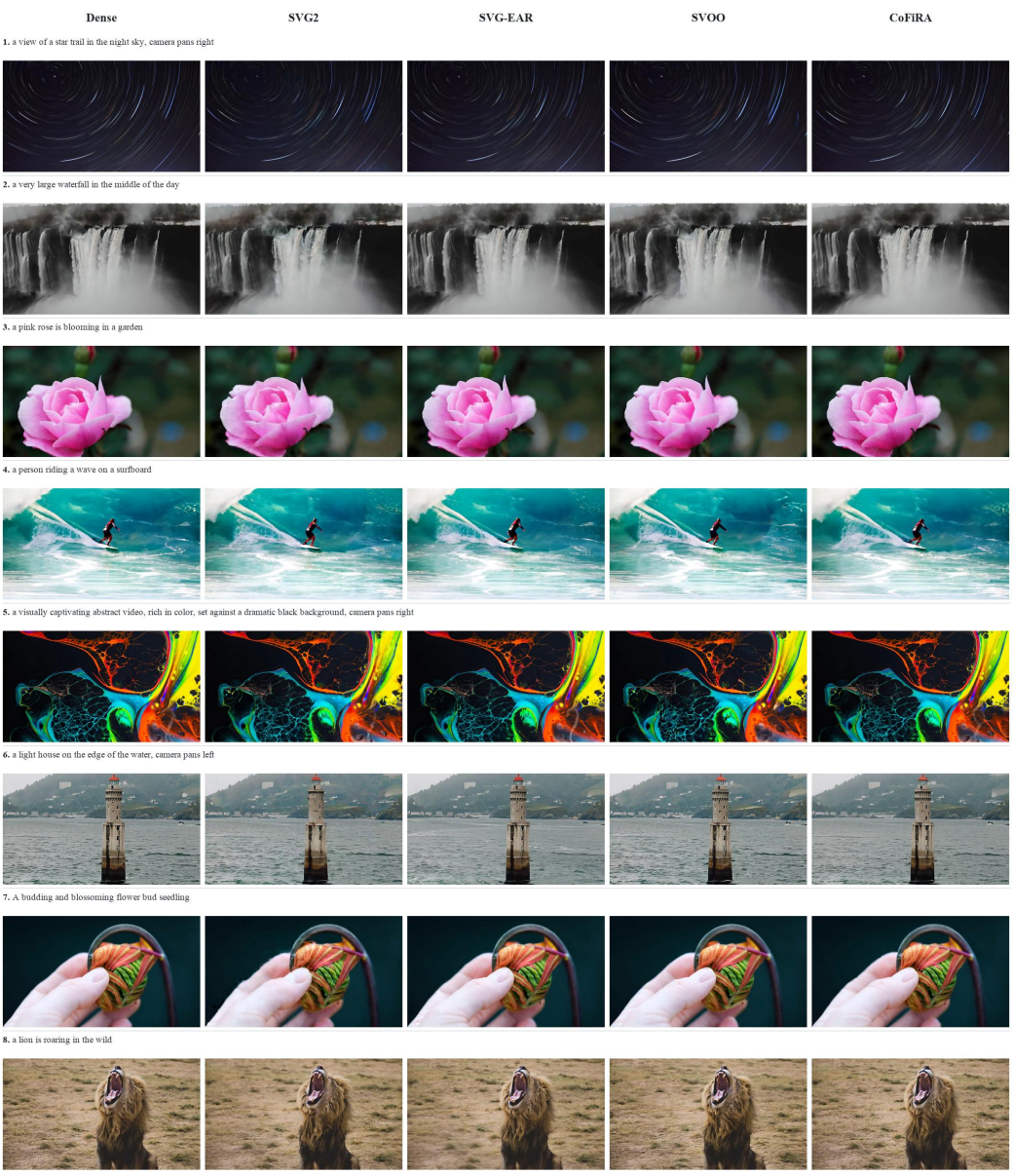


Figure 10: Additional qualitative comparisons on Wan2.1-I2V-14B under matched input images, prompts, and generation seeds.

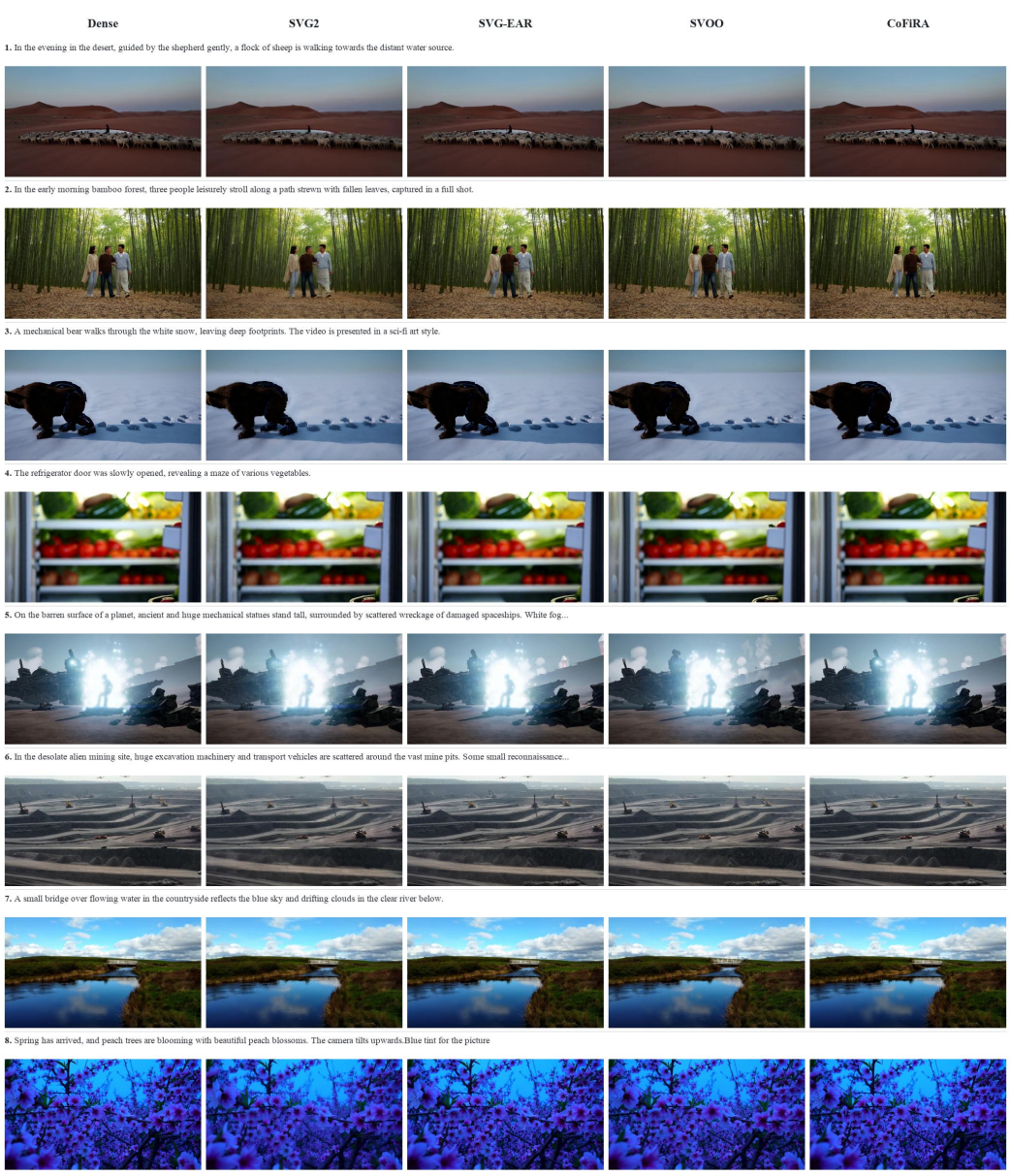


Figure 11: Additional qualitative comparisons on HunyuanVideo-T2V-13B under matched prompts and generation seeds.