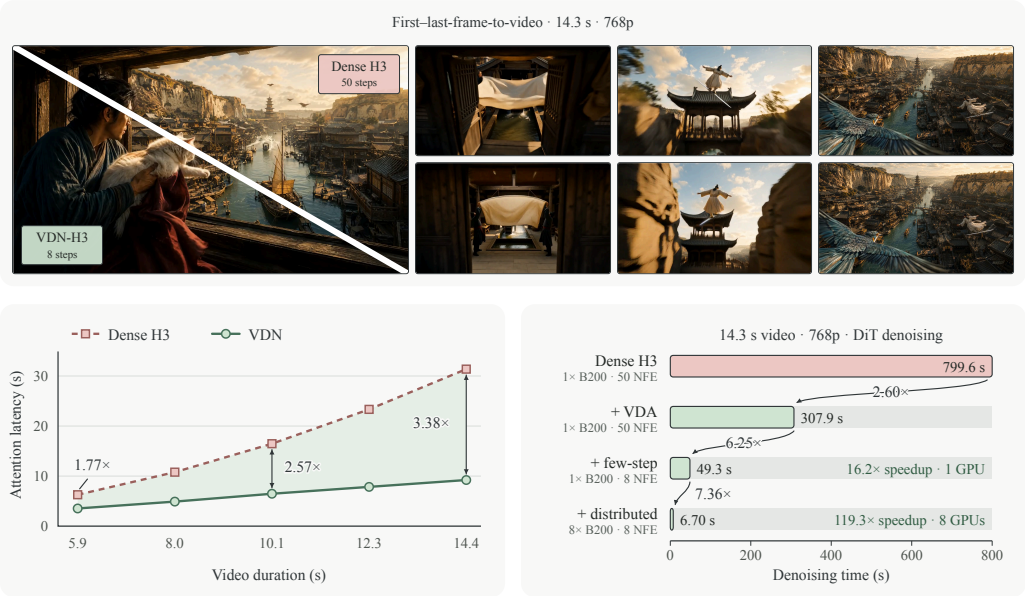
VIDEO DELTANET: A VIDEO-NATIVE HYBRID ATTENTION FOR EFFICIENT VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Video diffusion models repeatedly process long spatiotemporal token sequences during denoising, making attention a major computational bottleneck. Linear attention offers an appealing alternative and has been widely adopted in recent large language models, but directly applying it to video models often fails to preserve the fine-grained interactions required for high-quality generation. We present **Video DeltaNet (VDN)**, which combines local Softmax attention with bidirectional linear memory for long-range video context. Its linear branch introduces **Video Delta Attention (VDA)**, which jointly resolves correlated within-frame writes and yields a non-expansive inherited-state transition without additional frame-size key scaling. Separate output projections and learnable gates calibrate the two branches, while a staged teacher-alignment recipe progressively introduces the new pathway into pretrained models. We instantiate VDN on MiniMax H3, applying the hybrid to video-to-video interactions while retaining Softmax for interactions involving text or audio. Applying VDN directly yields 2.6× and 3.2× backbone speedups on B200 and H200, respectively. With 8-NFE distillation and optimized inference, VDN-H3 completes DiT denoising for a 14.3-second, 768p video in 6.70 seconds on eight NVIDIA B200 GPUs, corresponding to a 14.5× speedup over the 50-NFE Dense H3 baseline on the same GPU count.



(a) Speedup grows with video duration.

(b) Progressive acceleration of VDN-H3.

Figure 1: Quality and efficiency of VDN-H3. (Top) VDN-H3 matches Dense H3 in visual quality and temporal consistency. (a) Attention speedup grows to 3.38× with video duration. (b) VDN-H3 denoises a 14.3-second, 768p video in 6.70 seconds, achieving 2.1× real-time throughput.

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.

1 INTRODUCTION

Long-sequence video generation is dominated by attention (Polyak et al., 2024; MiniMax, 2026). In MiniMax H3, Softmax accounts for more than 85% of denoiser runtime. Dense Softmax preserves complete query–key access but scales quadratically with sequence length. Recurrent linear attention instead compresses distant context into a fixed-size state with linear cost (Yang et al., 2025a; Kimi Team, 2025). Recent video models explore linear and hybrid designs (Chen et al., 2025; Zhu et al., 2026), yet replacing Softmax outright degrades quality because this state cannot preserve every fine-grained interaction.

Closing this quality gap requires addressing three mismatches. First, a fixed-size state must preserve global properties such as subject identity, scene layout, appearance, and long-range motion, whereas Softmax retains an expanding set of keys and values. Second, delta-rule linear attention updates its recurrent state token by token, following autoregressive language-model decoding (Yang et al., 2024b; 2025a). Video diffusion instead processes all spatial tokens in a frame together; imposing an arbitrary patch order is unnatural, while independent writes from correlated patches can interfere and require frame-size-dependent stabilization. Third, inserting a new linear pathway changes the information flow and residual statistics of a Softmax-pretrained model; without careful adaptation, it can disrupt pretrained capabilities before becoming useful.

We introduce **Video DeltaNet (VDN)**, a hybrid attention architecture designed around these challenges. VDN retains Softmax for local video interactions and global boundary anchors, while bidirectional linear memory represents distant video context. The anchors keep the beginning and end of the clip directly accessible as global references. The linear branch uses **Video Delta Attention (VDA)**, a video-native delta rule that jointly resolves correlated writes within each frame. Its inherited-state transition is non-expansive without scaling keys by the number of spatial tokens per frame. RMS normalization stabilizes the scale of the linear readout, while separate gates and output projections calibrate each branch before their outputs are combined. A staged adaptation recipe first aligns the new linear pathway with the pretrained teacher, then uses low-rank updates to co-adapt the hybrid layer while preserving the pretrained backbone.

We instantiate the approach on MiniMax H3 to obtain **VDN-H3**. Applying VDN directly yields 2.6× and 3.2× backbone speedups on B200 and H200, respectively. With 8-NFE distillation, VDN-H3 completes DiT denoising for a 14.3-second, 768p video in 6.70 seconds on eight NVIDIA B200 GPUs. This corresponds to a 14.5× reduction relative to the 50-NFE Dense H3 baseline on the same GPU count. On a fixed third-party benchmark, 8-NFE VDN-H3 matches or slightly exceeds 50-NFE Dense H3 across video-quality metrics while preserving comparable motion magnitude.

Our main contributions are:

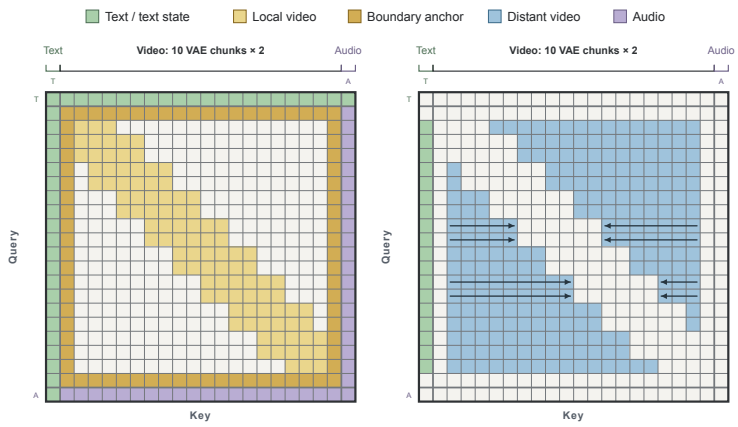
1. **A hybrid video attention architecture** that combines local Softmax and global boundary anchors with bidirectional linear memory, together with branch-specific normalization, gates, and output projections.
2. **A frame-wise delta operator** that resolves correlated spatial writes and has a non-expansive inherited-state transition without additional frame-size key scaling, together with an efficient batched implementation.
3. **A complete adaptation recipe for pretrained video models** that combines staged teacher alignment, low-rank refinement, few-step distillation, and optimized inference without training a new foundation model from scratch.

2 VIDEO DELTANET: HYBRID ATTENTION ARCHITECTURE

Video DeltaNet allocates video-to-video interactions by temporal role: exact Softmax handles nearby tokens and two boundary frames, while bidirectional linear memory carries the remaining context.

2.1 SLIDING-WINDOW SOFTMAX ATTENTION

Nearby frames contain local correspondences that determine texture, object boundaries, and short-term motion. These interactions benefit from direct token-to-token matching across neighboring



(a) Softmax (b) Linear

Figure 2: Softmax and Linear attention allocation. The schematic uses one text token, twenty video tokens in ten two-frame chunks, and one audio token.

frames. VDN therefore retains exact Softmax attention within a bidirectional temporal window to preserve quality, while assigning distant interactions to linear attention.

The window follows the video tokenizer. H3's VAE decodes five consecutive latent frames as one temporal chunk, so each query chunk attends to itself and its immediately preceding and following chunks. This prevents the Softmax boundary from cutting through the tokenizer's natural temporal unit, producing a 15-frame window except at sequence boundaries.

VDN also adds **two boundary anchors with four-way connectivity**. Every video frame attends to all tokens in the first and last latent frames, and the first and last frames attend to the complete sequence. Thus every query can read both endpoints, while both endpoints can aggregate clip-wide content; only two frame rows and columns receive dense connectivity. In image-to-video, the supplied first frame is directly visible throughout the clip and the generated last frame remains a global anchor. In first–last-frame-to-video, both supplied endpoints are directly visible. Anchor entries already inside a local window are included only once.

2.2 Bidirectional Linear Attention

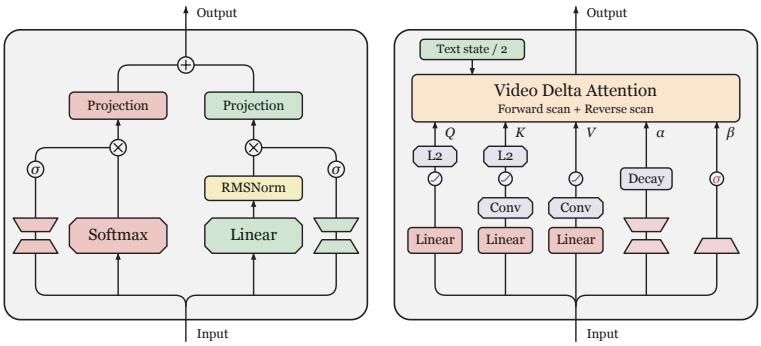
Outside the local window and boundary anchors, a forward state $S_t^{\rightarrow}$ summarizes earlier frames and a reverse state $S_t^{\leftarrow}$ summarizes later frames. The two regions are disjoint, so their readouts add without double counting. VDN gathers the state adjacent to each window boundary and bridges the skipped local span with channel-wise decay; Appendix A.3 gives the exact indexing. The linear branch also summarizes the prompt as S_T and splits it evenly between the two scan initializations:

$$\begin{aligned} o_t^L &= S_t^{\rightarrow} q_t + S_t^{\leftarrow} q_t, \\ S_0^{\rightarrow} = S_0^{\leftarrow} &= \tfrac{1}{2} S_T, \qquad \big(S_0^{\rightarrow} + S_0^{\leftarrow}\big) q = S_T q. \end{aligned} \tag{1a}$$

Thus the summed readout counts prompt memory exactly once, while text remains directly visible to the Softmax branch.

2.3 Combining Outputs from Two Branches

Both branches reuse the pretrained QKV projections. Softmax keeps H3's QK normalization and rotary processing, while the linear branch applies its own feature preparation following Gated DeltaNet and Kimi Delta Attention (Yang et al., 2025a; Kimi Team, 2025); Appendix D.2 gives the released configuration. Because the restricted Softmax and recurrent readouts have different scales, each receives a content-dependent sigmoid gate, and the linear output is RMS-normalized. Their independently projected outputs are



(a) Hybrid attention (b) Linear branch

Figure 3: Video DeltaNet architecture. Softmax and Linear outputs are independently gated and projected; the Linear branch uses bidirectional Video Delta Attention.

$$Y = \big(G^S \odot O^S\big) W_O^S + \big[G^L \odot \mathrm{RMSNorm}(O^L)\big] W_O^L. \tag{1b}$$

Separate projections let the branches contribute in different residual-stream directions. Their sum enters the pretrained residual stream, and the original feed-forward sublayer is unchanged.

3 Video Delta Attention: Frame-wise Delta Rule

Video Delta Attention (VDA) extends the delta rule from individual tokens to entire video frames. It jointly updates the recurrent state from all key-value pairs in a frame, so their correlations shape the write while the inherited-state transition remains non-expansive without extra frame-size normalization. We first review the standard token-wise rule and then derive the frame-wise update.

3.1 Preliminaries of Linear Attention

Delta-rule memory reads a value associated with a key and writes a correction proportional to the prediction residual. Gated DeltaNet combines this mechanism with memory decay (Yang et al., 2025a), while Kimi Delta Attention introduces finer-grained decay (Kimi Team, 2025). At step t, for a state $S_{t-1} \in \mathbb{R}^{d_v \times d_k}$, a key $k_t \in \mathbb{R}^{d_k}$, and a value $v_t \in \mathbb{R}^{d_v}$, the familiar one-token update is

$$\begin{aligned} \bar{S}_t &= S_{t-1}\,\mathrm{Diag}(\alpha_t), \\ S_t &= \bar{S}_t + \beta_t (v_t - \bar{S}_t k_t) k_t^\top. \end{aligned} \tag{2}$$

The decay gate α_t controls how much inherited memory remains, while β_t controls the strength of the erase-and-write correction. This token-wise recurrence is well matched to autoregressive decoding, where one new token arrives at each step.

Video diffusion presents a different unit of computation: all spatial tokens of a latent frame are available together. Sequentially imposing a patch order is unnecessary, so a natural first adaptation is to compute their delta corrections in parallel from the same decayed state. SANA-WM follows this batched construction but, beyond the token-level $1/\sqrt{d_k}$ key normalization, scales keys by another $1/\sqrt{U}$ to stabilize the additive transition (Zhu et al., 2026).

Let a video contain F latent frames with $U = H_\ell W_\ell$ tokens each. For token u of frame t, denote its key, value, and write gate by $k_{t,u} \in \mathbb{R}^{d_k}$, $v_{t,u} \in \mathbb{R}^{d_v}$, and $\beta_{t,u} \geq 0$. A frozen-state update adds all token corrections evaluated at $\bar{S}_t = S_{t-1}\,\mathrm{Diag}(\alpha_t)$:

$$S_t^{\mathrm{batch}} = \bar{S}_t + \sum_{u=1}^{U} \beta_{t,u} (v_{t,u} - \bar{S}_t k_{t,u}) k_{t,u}^\top. \tag{3a}$$

Stack the keys and values as $K_t \in \mathbb{R}^{U\times d_k}$ and $V_t \in \mathbb{R}^{U\times d_v}$, and write $\beta_t = (\beta_{t,1}, \ldots, \beta_{t,U})^\top$. The two frame statistics are

$$A_t = \sum_{u=1}^{U} \beta_{t,u} k_{t,u} k_{t,u}^\top = K_t^\top \operatorname{Diag}(\beta_t) K_t,$$
$$B_t = \sum_{u=1}^{U} \beta_{t,u} v_{t,u} k_{t,u}^\top = V_t^\top \operatorname{Diag}(\beta_t) K_t. \tag{3b}$$

Here A_t summarizes key correlations and B_t value–key writes, giving

$$S_t^{\text{batch}} = \bar{S}_t(I - A_t) + B_t. \tag{3c}$$

Because every residual in Equation (3a) uses the same $\bar{S}_t$, overlapping keys can produce conflicting writes without accounting for one another (Appendix A.2).

3.2 Video Delta Attention: A Frame-wise Update

This independence becomes problematic when several patches address similar key directions. Their corrections can reinforce or conflict even though they are meant to describe one frame. VDA instead lets all spatial tokens determine a single new state together, so overlapping directions are resolved inside the update while previously accumulated memory remains a reference.

The classical one-token delta rule can be viewed as one gradient step on its prediction error. Rather than taking one such step independently for every patch, VDA defines the frame-level state as the solution to a joint objective:

$$S_t = \arg\min_S \ \frac{1}{2}\|S - \bar{S}_t\|_F^2 + \frac{1}{2}\sum_{u=1}^{U} \beta_{t,u} \|S k_{t,u} - v_{t,u}\|_2^2. \tag{4}$$

The first term keeps the new memory close to the decayed state. The second asks that same state to fit all key–value associations in the frame simultaneously. Differentiating once gives a compact normal equation and closed-form update:

$$S_t(I + A_t) = \bar{S}_t + B_t, \qquad S_t = (\bar{S}_t + B_t)(I + A_t)^{-1}. \tag{5}$$

In contrast to Equation (3c), each residual in the joint solution is evaluated at the shared updated state S_t. The inverse couples the writes: when two patch keys overlap, their inner product affects both effective corrections. The solve therefore responds to the observed key correlations rather than uniformly shrinking every key according to U. We compute this inverse in the $d_k \times d_k$ key-channel space; Appendix A.2 illustrates the effect of key correlations.

3.3 Stability and Correlation Awareness of Video Delta Attention

VDA has a stable inherited-state transition: the contribution carried from earlier frames cannot be amplified by a frame update when the prepared features and gates are fixed.

Proposition 1 — Non-expansive inherited-state transition. For fixed prepared features and gates, with $\beta_{t,u} \geq 0$ and $0 \leq \alpha_{t,j} \leq 1$, the transition $M_t = \operatorname{Diag}(\alpha_t)(I + A_t)^{-1}$ satisfies $\|M_t\|_2 \leq 1$. Thus changing the entering state by ΔS changes its carried contribution by at most $\|\Delta S\|_F$.

With only token-level $1/\sqrt{d_k}$ key normalization, additive frame-wise writes can amplify inherited state; SANA-WM adds $1/\sqrt{U}$ key scaling to control that transition (Zhu et al., 2026). VDA uses the token-level normalization without the extra $1/\sqrt{U}$: for fixed prepared features and gates, $(I + A_t)^{-1}$ makes its inherited-state transition non-expansive. Appendix A.1 proves the proposition, and Appendix A.2 compares the two rules under different key correlations.

4 END-TO-END TRAINING PIPELINE

We first describe the three-stage adaptation that integrates the Linear branch into the pretrained Softmax backbone, followed by few-step distillation from 50 to eight denoising steps.

4.1 STAGED ARCHITECTURE ADAPTATION

Architecture adaptation proceeds in three stages while the pretrained base weights remain frozen. **A1 - Per-layer alignment** initializes each Linear branch independently from frozen activations for 200 steps, with the Softmax gate fixed at 0.99 and gradients clipped per layer at 0.1. **A2 - End-to-end alignment** installs all calibrated branches and optimizes them jointly for 500 steps, correcting composition errors while keeping the pretrained weights and Softmax gates frozen. **Stage B - LoRA co-adaptation** adds QKVO LoRA adapters and trains them with the Linear pathway and Softmax gates for 2,000 steps. Appendix B.1 visualizes the trainable components, and Appendix B.2 gives the complete settings.

4.2 FEW-STEP DISTILLATION

After architecture adaptation, we distill the 50-NFE VDN-H3 into an 8-NFE sampler with LoRA, using a DMD2-style objective without the GAN term (Yin et al., 2024a). We initialize the student's LoRA from MiniMax-H3-Turbo-LoRA (larryvrh, 2026), which also uses DMD2-style distillation, and train it against VDN-H3's own 50-NFE sampler, isolating NFE reduction from architecture conversion. Generator, real-score, and fake-score roles share one FSDP backbone through separate adapters, with three fake-score updates per generator update. The released checkpoint is trained for 250 generator steps.

5 EFFICIENT INFERENCE

Fused operators. Four Triton kernels implement VDA preparation, frame statistics (A_t, B_t), boundary-state gathering with the decay bridge, and the gated RMS-normalized epilogue. QK normalization and rotary embedding are fused separately on the Softmax path.

Scans and inverses. Because VDA reads memory only at VAE-chunk boundaries, we compose each chunk into $S_{\text{out}} = S_{\text{in}} M_{\text{chunk}} + J_{\text{chunk}}$ and scan the shorter chunk sequence in both directions. A register-resident CUDA kernel computes each small $(I + A_t)^{-1}$ and directly emits the transition and injection terms, avoiding the launches and intermediate traffic of a batched factorization pipeline.

Parallel execution. Following Ulysses (Jacobs et al., 2023), VDA is sharded by attention head and runs on a side stream alongside packed-varlen Window Softmax. MXFP8 accelerates wide GEMMs, while recurrent states and inverses remain in FP32; AdaLN modulation parameters are precomputed before the block loop. Appendix D.2 gives the full kernel and execution details.

6 EXPERIMENTS

6.1 SETTINGS

Data. We curated a training set of 10,015 video clips at 1344×768 resolution, each with 345 frames at 24 fps (14.375 seconds). Each sample is pre-encoded and cached as video latents of shape $(24, 102, 48, 84)$, stereo audio latents of shape $(2, 32, 575)$, and Qwen3-VL text embeddings of shape $(L, 5120)$ with token-type tags (Bai et al., 2025). This removes the VAEs and text encoder from the training loop. For evaluation, every model renders the same 103 fixed third-party prompts at the same resolution and duration.

Baselines. The 50-NFE comparison includes full-attention Dense H3, the training-free sparse baseline Radial Attention (Li et al., 2025b), and the Stage-B VDN-H3 checkpoint. Radial Attention's density is set to match VDN-H3's B200 latency. The 8-NFE comparison includes Dense H3, distilled VDN-H3, and FastH3 v2 (FastVideo Team, 2026b), a sparse-attention baseline for video generation. These comparisons isolate architecture changes at 50 NFE and evaluate accelerated samplers at 8 NFE.

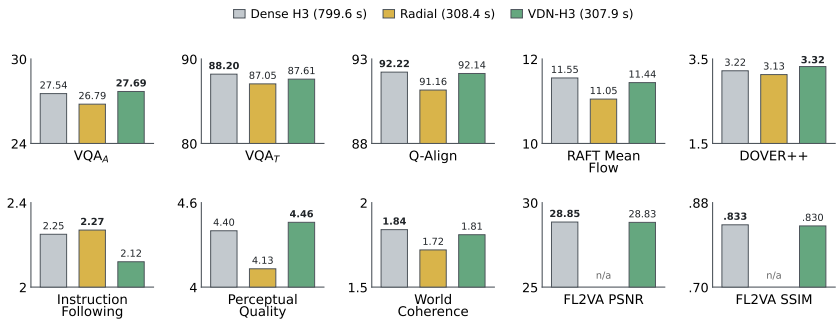


Figure 4: 50-NFE comparison across ten metrics. VDN-H3 remains close to Dense H3 and is higher than Radial on six of seven shared quality metrics while matching its B200 latency. Radial was not evaluated on FL2VA.

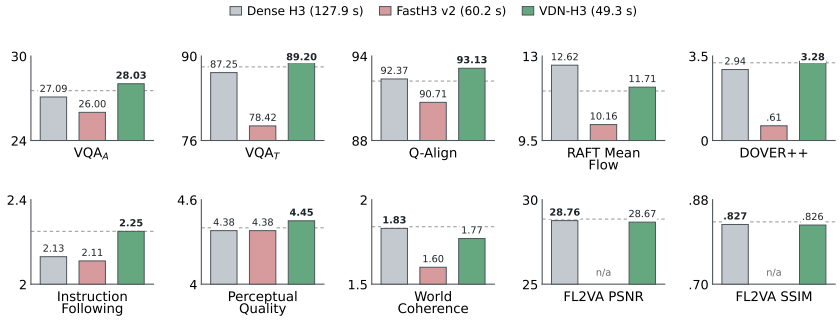


Figure 5: 8-NFE comparison across the same ten metrics. VDN-H3 is higher than FastH3 v2 on all seven shared quality metrics, remains close to Dense H3, and runs 1.2× faster than FastH3 v2 on one B200. FastH3 v2 was not evaluated on FL2VA; dashed lines mark 50-NFE Dense H3.

Metrics. For text-to-video evaluation, we report EvalCrafter VQA_A and VQA_T (Liu et al., 2024), Q-Align (Wu et al., 2024), and DOVER++ overall (Wu et al., 2023). FIRM-Video evaluates Instruction Following, Perceptual Quality, and World Coherence on a 1–5 scale (Zhang et al., 2026d). For first–last-frame-to-video evaluation (FL2VA), PSNR and SSIM measure fidelity at the two conditioning frames (Wang et al., 2004). Higher is better for all quality metrics; Q-Align and DOVER++ use a $\times 100$ scale. We additionally report RAFT mean flow as a motion diagnostic (Teed & Deng, 2020).

Environment. We use PyTorch 2.13 and CUDA 12.9 on NVIDIA H200 and B200 clusters. Training uses FSDP2/HSDP, activation checkpointing, and pinned-memory activation offload. Parameters are gathered in bf16 and gradients reduced in FP32, except that the decay-α modules remain FP32. Appendix B.2 gives the architecture-adaptation settings.

6.2 Quality Results

Figure 4 compares the attention architectures at 50 NFE across ten metrics. VDN-H3 remains close to Dense H3: it is higher on VQA_A, DOVER++, and FIRM perceptual quality, while the gaps are only 0.08 on Q-Align, 0.03 on FIRM world coherence, 0.02 dB on FL2VA PSNR, and 0.0024 on FL2VA SSIM. Across the seven quality metrics shared with Radial, VDN-H3 is higher on six, including gains of 0.90 on VQA_A, 0.98 on Q-Align, and 0.33 on FIRM perceptual quality; RAFT flow remains similar. These results support our claim that VDN-H3 maintains quality comparable to Dense H3 while outperforming Radial at similar latency.

Figure 5 compares the 8-NFE samplers on the same ten metrics. VDN-H3 exceeds FastH3 v2 on all seven shared quality metrics, with margins of 10.78 on VQA_T and 2.67 on DOVER++. It also exceeds 8-NFE Dense H3 on six of nine quality metrics; world coherence and FL2VA endpoint fidelity remain close. Appendix C reports FAST-VQA, FL2VA LPIPS, and the 4-NFE results.

Additional paired qualitative samples appear in Appendix E.

Table 2: B200 efficiency. Gains are incremental and speedups are cumulative against one-GPU Dense H3. The one-GPU duration sweep compares Dense H3 and VDN-H3 at 50 NFE.

Configuration	#GPU	NFE	s/NFE	End-to-end latency	Gain	Speedup
Dense H3	1	50	15.99 s	799.6 s	–	1.0×
VDN-H3	1	50	6.16 s	307.9 s	2.6×	2.6×
+ few-step	1	8	6.16 s	49.3 s	6.3×	16.2×
+ distributed	8	8	0.85 s	6.70 s	7.4×	119.3×

Latent frames	42	72	87	102
Attention density	42.08%	26.82%	22.85%	19.98%
Attention speedup	1.7×	2.3×	2.7×	3.0×
VDN-H3 latency	118.8 s	213.1 s	260.0 s	307.9 s
End-to-end speedup	1.6×	2.1×	2.4×	**2.6**×

Table 3: Stage-wise relative ℓ_2 parameter displacement (%). Columns abbreviate the write gate β, output-gate MLP g_o, decay MLP α, output-gate bias b_o, and output projection W_o.

Stage	Steps	β	g_o Down	g_o Up	α Down	α Up	b_o	W_o (1.93B)
A1	200	35.6	32.9	24.7	20.5	18.3	15.3	1.25
A2	500	17.3	17.6	13.9	14.1	11.3	13.7	0.82
B	2,000	14.0	14.4	11.1	12.2	9.2	9.9	0.58

6.3 Efficiency Results

Due to space limitations, we report B200 results here; all H200 results appear in Appendix Table 7.

Backbone acceleration. As shown in Table 2, for the 14.3-second, 768p workload, the system-optimized VDN-H3 backbone reduces one complete transformer evaluation from 16.0 to 6.2 seconds on one B200 (2.6×) at the same sequence length and NFE count.

Scaling with video length. The duration sweep in Table 2 shows that the advantage widens with sequence length. Dense Softmax scores every video-token pair, so its video–video workload grows quadratically with the number of latent frames. As the sequence grows from 42 to 102 latent frames, the measured Softmax attention density falls from 42.1% to 20.0%; at fixed 50 NFE, whole-backbone speedup on B200 increases from 1.6× to 2.6×.

Sampling and distributed inference. The deployment ladder in Table 2 shows that 8-NFE distillation reduces the optimized backbone's one-B200 denoising time from 307.9 to 49.3 seconds, while eight-GPU head-sharded inference brings the final latency to 6.7 seconds.

6.4 Ablation & Analysis

Stage partitioning. Table 3 measures $\|\theta_s - \theta_{s-1}\|_2 / \|\theta_{s-1}\|_2$ within each stage. Despite using only 200 steps, A1 moves every monitored module the most; the 500-step A2 and 2,000-step Stage B make successively smaller changes, for example $35.6 \to 17.3 \to 14.0\%$ for the write-gate projection and $1.25 \to 0.82 \to 0.58\%$ for W_o. Thus A1 supplies the main branch initialization, while the longer end-to-end stages refine composition and co-adaptation.

Architecture ablations. Table 4 tests the frame update, boundary anchors, and long-range Linear branch separately. We first replace VDA's joint solve with a frozen-state additive update that uses an extra $1/\sqrt{U}$ key scaling factor. VDA scores higher on five of eight quality metrics, including gains of 0.42 on VQA_A, 0.69 on VQA_T, 0.15 on FAST-VQA, and 0.07 on FIRM world coherence. The additive update scores higher by 0.19 on Q-Align and 0.14 on DOVER++, while FIRM perceptual quality ties. VDA thus avoids the frame-size-dependent scaling factor while retaining competitive quality and scoring higher on most metrics; its inherited-state transition is also non-expansive. Removing the first- and last-frame anchors while keeping VDA and the Linear branch reduces seven of eight quality metrics. The largest changes are -0.75 in VQA_A and -1.35 in VQA_T; Q-Align is the only metric that increases, by 0.24. This supports using the endpoints as explicit global references in addition to the local Softmax window. Finally, comparing the no-anchor VDN variant with Window Softmax only isolates the Linear branch because both omit the anchors. Removing the branch lowers six of eight quality metrics, including VQA_A by 1.57, VQA_T by 1.75, and FAST-VQA by 1.77, showing that local Softmax alone does not recover distant context.

Table 4: Architecture ablations on video quality and motion; bold marks each column maximum.

	Video quality					FIRM-Video			Motion
Method	**VQA$_A$**	**VQA$_T$**	**Q-Align**	**F-VQA**	**DOVER++**	**IF**	**PQ**	**WC**	**Flow**
VDN	**30.56**	**90.11**	92.69	**83.47**	3.62	2.27	**4.47**	**1.83**	11.67
− Video Delta rule	30.14	89.42	92.88	83.32	**3.76**	2.25	**4.47**	1.76	11.29
− Boundary anchors	29.81	88.76	**92.93**	83.34	3.50	2.25	4.43	1.79	11.58
− Linear branch	28.24	87.01	92.18	81.57	3.24	**2.30**	4.38	1.82	**11.99**

Table 5: B200 kernel ablations at 102 latent frames (latency in milliseconds).

Fused VDA stages				Other kernels			
Operator	**Orig.**	**Opt.**	**Speedup**	**Operator**	**Orig.**	**Opt.**	**Speedup**
VDA-Prep	17.07	3.37	5.1×	Chunk scan, 1 GPU	3.29	4.56	0.72×
VDA-Stats	11.39	3.99	2.9×	Chunk scan, 8 GPU	3.02	0.82	3.7×
VDA-Gather	1.57	0.21	7.5×	Matrix inverse	6.32	1.26	5.0×
VDA-Epilogue	6.93	0.76	9.1×	Window Softmax	69.5	56.1	1.24×

Kernel speedups. Table 5 summarizes the B200 results at 102 latent frames. Fused VDA-Prep, VDA-Stats, VDA-Gather, and VDA-Epilogue provide 5.1×, 2.9×, 7.5×, and 9.1× speedups, respectively. Chunk scans add overhead on one GPU (0.72×) but become effective after head sharding, reaching 3.7× across eight GPUs. The fused small-matrix inverse and packed Window Softmax provide another 5.0× and 1.24× speedups. Appendix D.2 gives implementation details, and Appendix D reports the H200 results.

7 Related Work

Efficient video attention. Video diffusion transformers process long spatiotemporal sequences (HaCohen et al., 2024; Jin et al., 2024; Kong et al., 2024; Polyak et al., 2024; Wan et al., 2025; Yang et al., 2025b; MiniMax, 2026). Acceleration spans exact FlashAttention kernels (Dao et al., 2022; Dao, 2023; Shah et al., 2024), low-precision kernels (Zhang et al., 2025b; 2024; 2025c), general and video-specific sparsity (Zhang et al., 2025d; 2026a; Xi et al., 2025), sparse–linear hybrids (Zhang et al., 2025a; 2026b), and quantized diffusion (Zhao et al., 2024; Li et al., 2025a; 2026; Xue et al., 2026; Xi et al., 2026).

Linear attention and recurrent memory. Linear Transformers, Performers, and selective state-space models establish linear-time sequence processing (Katharopoulos et al., 2020; Choromanski et al., 2021; Gu & Dao, 2023; Dao & Gu, 2024). Data-dependent memory updates include GLA, DeltaNet, Gated Delta Networks, Kimi Linear, and Gated DeltaNet-2 (Yang et al., 2024a;b; 2025a; Kimi Team, 2025; Hatamizadeh et al., 2026). Vision and diffusion adaptations include VideoMamba, M4V, LoGeR, and the SANA family (Li et al., 2024b; Huang et al., 2026; Zhang et al., 2026c; Xie et al., 2025; Chen et al., 2025; 2026; Zhu et al., 2026); concurrent work analyzes frame-level delta updates (Zhu, 2026).

Few-step and distributed inference. Distillation and consistency methods reduce denoising evaluations (Salimans & Ho, 2022; Song et al., 2023; Luo et al., 2023; Sauer et al., 2023; Yin et al., 2024b;a; Wang et al., 2024; Li et al., 2024a). Orthogonal multi-GPU systems shard sequences or pipeline diffusion computation (Jacobs et al., 2023; Fang & Zhao, 2024; Li et al., 2024c; Fang et al., 2024a;b); StreamDiffusionV2 targets low-latency distributed video streaming (Feng et al., 2026).

8 Conclusion

Video DeltaNet accelerates video generation while largely preserving Dense H3 quality. On the 14.3-second, 768p workload, its optimized backbone is 2.6× faster on one B200 and 3.2× faster on one H200. With 8-NFE distillation and eight-GPU inference, DiT denoising takes 6.70 seconds on B200s and 12.5 seconds on H200s. The 8-NFE VDN-H3 remains close to 50-NFE Dense H3 across the reported quality evaluations.

AI USE STATEMENT

Generative AI tools were used to assist with language editing, document formatting, and figure prototyping. The authors reviewed the AI-assisted material and remain responsible for the technical claims, experiments, and final content.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

Junsong Chen, Yuyang Zhao, Jincheng Yu, Ruihang Chu, Junyu Chen, Shuai Yang, Xianbang Wang, Yicheng Pan, Daquan Zhou, Huan Ling, Haozhe Liu, Hongwei Yi, Hao Zhang, Muyang Li, Yukang Chen, Han Cai, Sanja Fidler, Ping Luo, Song Han, and Enze Xie. SANA-Video: Efficient video generation with block linear diffusion transformer. *arXiv preprint arXiv:2509.24695*, 2025.

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

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy Colwell, and Adrian Weller. Rethinking attention with performers. In *International Conference on Learning Representations*, 2021.

Tri Dao. FlashAttention-2: Faster attention with better parallelism and work partitioning. *arXiv preprint arXiv:2307.08691*, 2023.

Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *International Conference on Machine Learning*, 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*, 2022.

Jiarui Fang and Shangchun Zhao. USP: A unified sequence parallelism approach for long context generative ai. *arXiv preprint arXiv:2405.07719*, 2024.

Jiarui Fang, Jinzhe Pan, Aoyu Li, Xibo Sun, and Jiannan Wang. PipeFusion: Patch-level pipeline parallelism for diffusion transformers inference. *arXiv preprint arXiv:2405.14430*, 2024a.

Jiarui Fang, Jinzhe Pan, Xibo Sun, Aoyu Li, and Jiannan Wang. xDiT: An inference engine for diffusion transformers with massive parallelism. *arXiv preprint arXiv:2411.01738*, 2024b.

FastVideo Team. FastH3 Preview v1: Open-weight 4-step sparse-distilled MiniMax-H3. Hugging Face model repository, 2026a. URL `https://huggingface.co/FastVideo/FastVideo-FastH3-4-step-Preview-v1-LoRA`.

FastVideo Team. FastH3 8-Step V2: Data-free distillation of MiniMax-H3. Hugging Face model repository, 2026b. URL `https://huggingface.co/FastVideo/FastVideo-FastH3-8-Step-V2`.

Tianrui Feng, Zhi Li, Shuo Yang, Haocheng Xi, Muyang Li, Xiuyu Li, Lvmin Zhang, Keting Yang, Kelly Peng, Song Han, Maneesh Agrawala, Kurt Keutzer, Akio Kodaira, and Chenfeng Xu. StreamDiffusionV2: A streaming system for dynamic and interactive video generation. In *Proceedings of Machine Learning and Systems*, 2026.

Chongjian Ge, Hanwen Jiang, Tianyu Wang, Jiuxiang Gu, Yiran Xu, Ziwen Chen, Shaoteng Liu, Jing Shi, Yicong Hong, Zefan Cai, Hailin Jin, and Hao Tan. Chimera: Designing and chinchilla-scaling hybrid visual diffusion transformers. *arXiv preprint arXiv:2607.28611*, 2026.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*, 2023.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. LTX-Video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Ali Hatamizadeh, Yejin Choi, and Jan Kautz. Gated DeltaNet-2: Decoupling erase and write in linear attention. *arXiv preprint arXiv:2605.22791*, 2026.

Jiancheng Huang, Gengwei Zhang, Zequn Jie, Siyu Jiao, Yinlong Qian, Ling Chen, Yunchao Wei, and Lin Ma. M4V: Multimodal mamba for efficient text-to-video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Sam Ade Jacobs, Masahiro Tanaka, Chengming Zhang, Minjia Zhang, Shuaiwen Leon Song, Samyam Rajbhandari, and Yuxiong He. DeepSpeed Ulysses: System optimizations for enabling training of extreme long sequence transformer models. *arXiv preprint arXiv:2309.14509*, 2023.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. *arXiv preprint arXiv:2410.05954*, 2024.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In *International Conference on Machine Learning*, 2020.

Kimi Team. Kimi linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*, 2025.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

larryvrh. MiniMax-H3-Turbo-LoRA. Hugging Face model repository, 2026. URL `https://huggingface.co/larryvrh/MiniMax-H3-Turbo-Lora`.

Jiachen Li, Weixi Feng, Tsu-Jui Fu, Xinyi Wang, Sugato Basu, Wenhu Chen, and William Yang Wang. T2V-Turbo: Breaking the quality bottleneck of video consistency model with mixed reward feedback. In *Advances in Neural Information Processing Systems*, volume 37, 2024a. arXiv:2405.18750.

Kunchang Li, Xinhao Li, Yi Wang, Yinan He, Yali Wang, Limin Wang, and Yu Qiao. VideoMamba: State space model for efficient video understanding. *arXiv preprint arXiv:2403.06977*, 2024b.

Muyang Li, Tianle Cai, Jiaxin Cao, Qinsheng Zhang, Han Cai, Junjie Bai, Yangqing Jia, Ming-Yu Liu, Kai Li, and Song Han. DistriFusion: Distributed parallel inference for high-resolution diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024c.

Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. SVDQuant: Absorbing outliers by low-rank components for 4-bit diffusion models. In *International Conference on Learning Representations*, 2025a.

Xingyang Li, Muyang Li, Tianle Cai, Haocheng Xi, Shuo Yang, Yujun Lin, Lvmin Zhang, Songlin Yang, Jinbo Hu, Kelly Peng, Maneesh Agrawala, Ion Stoica, Kurt Keutzer, and Song Han. Radial attention: $\mathcal{O}(n \log n)$ sparse attention with energy decay for long video generation. In *Advances in Neural Information Processing Systems*, 2025b.

Xingyang Li, Samuel Tesfai, Zhekai Zhang, Haocheng Xi, Shuo Yang, Lvmin Zhang, Yufei Sun, Kelly Peng, Maneesh Agrawala, Ion Stoica, Kurt Keutzer, Jun-Yan Zhu, Song Han, Yujun Lin, and Muyang Li. DeltaQuant: 4-bit video diffusion models with spatiotemporal delta smoothing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. EvalCrafter: Benchmarking and evaluating large video generation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

MiniMax. MiniMax H3: An Open Model Breaking the Boundaries Between Tasks and Modalities. Model release, 2026. URL `https://www.minimax.io/blog/minimax-h3`.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, David Yan, Dhruv Choudhary, Dingkang Wang, Geet Sethi, Guan Pang, Haoyu Ma, Ishan Misra, Ji Hou, Jialiang Wang, Kiran Jagadeesh, Kunpeng Li, Luxin Zhang, Mannat Singh, Mary Williamson, Matt Le, Matthew Yu, Mitesh Kumar Singh, Peizhao Zhang, Peter Vajda, Quentin Duval, Rohit Girdhar, Roshan Sumbaly, Sai Saketh Rambhatla, Sam Tsai, Samaneh Azadi, Samyak Datta, Sanyuan Chen, Sean Bell, Sharadh Ramaswamy, Shelly Sheynin, Siddharth Bhattacharya, Simran Motwani, Tao Xu, Tianhe Li, Tingbo Hou, Wei-Ning Hsu, Xi Yin, Xiaoliang Dai, Yaniv Taigman, Yaqiao Luo, Yen-Cheng Liu, Yi-Chiao Wu, Yue Zhao, Yuval Kirstain, Zecheng He, Zijian He, Albert Pumarola, Ali Thabet, Artsiom Sanakoyeu, Arun Mallya, Baishan Guo, Boris Araya, Breena Kerr, Carleigh Wood, Ce Liu, Cen Peng, Dimitry Vengertsev, Edgar Schonfeld, Elliot Blanchard, Felix Juefei-Xu, Fraylie Nord, Jeff Liang, John Hoffman, Jonas Kohler, Kaolin Fire, Karthik Sivakumar, Lawrence Chen, Licheng Yu, Luya Gao, Markos Georgopoulos, Rashel Moritz, Sara K. Sampson, Shikai Li, Simone Parmeggiani, Steve Fine, Tara Fowler, Vladan Petrovic, and Yuming Du. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. *arXiv preprint arXiv:2311.17042*, 2023.

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*, 2024.

Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *International Conference on Machine Learning*, 2023.

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision*, 2020.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang,

Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Fu-Yun Wang, Zhaoyang Huang, Weikang Bian, Xiaoyu Shi, Keqiang Sun, Guanglu Song, Yu Liu, and Hongsheng Li. AnimateLCM: Computation-efficient personalized style video generation without personalized video data. *arXiv preprint arXiv:2402.00769*, 2024.

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.

Haoning Wu, Chaofeng Chen, Jingwen Hou, Liang Liao, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. FAST-VQA: Efficient end-to-end video quality assessment with fragment sampling. In *European Conference on Computer Vision*, 2022.

Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023.

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, Qiong Yan, Xiongkuo Min, Guangtao Zhai, and Weisi Lin. Q-Align: Teaching LMMs for visual scoring via discrete text-defined levels. In *International Conference on Machine Learning*, 2024.

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

Haocheng Xi, Shuo Yang, Yilong Zhao, Muyang Li, Han Cai, Xingyang Li, Yujun Lin, Zhuoyang Zhang, Jintao Zhang, Xiuyu Li, Zhiying Xu, Jun Wu, Chenfeng Xu, Ion Stoica, Song Han, and Kurt Keutzer. Quant VideoGen: Auto-regressive long video generation via 2-bit KV-cache quantization. *arXiv preprint arXiv:2602.02958*, 2026.

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. SANA: Efficient high-resolution image synthesis with linear diffusion transformers. In *International Conference on Learning Representations*, 2025.

Bowen Xue, Zihan Min, Xingyang Li, Zhekai Zhang, Haocheng Xi, Lvmin Zhang, Maneesh Agrawala, Jun-Yan Zhu, Song Han, Yujun Lin, and Muyang Li. FourTune: Towards fully 4-bit efficient post-training for diffusion models. *arXiv preprint arXiv:2607.05711*, 2026.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. In *International Conference on Machine Learning*, 2024a.

Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. In *Advances in Neural Information Processing Systems*, 2024b.

Songlin Yang, Jan Kautz, and Ali Hatamizadeh. Gated delta networks: Improving Mamba2 with delta rule. In *International Conference on Learning Representations*, 2025a.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Yuxuan Zhang, Weihan Wang, Yean Cheng, Bin Xu, Xiaotao Gu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, 2025b.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T. Freeman. Improved distribution matching distillation for fast image synthesis. *arXiv preprint arXiv:2405.14867*, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T. Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b.

Jintao Zhang, Haofeng Huang, Pengle Zhang, Jia Wei, Jun Zhu, and Jianfei Chen. SageAttention2: Efficient attention with thorough outlier smoothing and per-thread INT4 quantization. *arXiv preprint arXiv:2411.10958*, 2024.

Jintao Zhang, Haoxu Wang, Kai Jiang, Shuo Yang, Kaiwen Zheng, Haocheng Xi, Ziteng Wang, Hongzhou Zhu, Min Zhao, Ion Stoica, Joseph E. Gonzalez, Jun Zhu, and Jianfei Chen. SLA: Beyond sparsity in diffusion transformers via fine-tunable sparse-linear attention. *arXiv preprint arXiv:2509.24006*, 2025a.

Jintao Zhang, Jia Wei, Haofeng Huang, Pengle Zhang, Jun Zhu, and Jianfei Chen. SageAttention: Accurate 8-bit attention for plug-and-play inference acceleration. In *International Conference on Learning Representations*, 2025b.

Jintao Zhang, Jia Wei, Pengle Zhang, Xiaoming Xu, Haofeng Huang, Haoxu Wang, Kai Jiang, Jianfei Chen, and Jun Zhu. SageAttention3: Microscaling FP4 attention for inference and an exploration of 8-bit training. *arXiv preprint arXiv:2505.11594*, 2025c.

Jintao Zhang, Chendong Xiang, Haofeng Huang, Jia Wei, Haocheng Xi, Jun Zhu, and Jianfei Chen. SpargeAttention: Accurate and training-free sparse attention accelerating any model inference. In *International Conference on Machine Learning*, 2025d.

Jintao Zhang, Kai Jiang, Chendong Xiang, Weiqi Feng, Yuezhou Hu, Haocheng Xi, Jianfei Chen, and Jun Zhu. SpargeAttention2: Trainable sparse attention via hybrid top-k+top-p masking and distillation fine-tuning. *arXiv preprint arXiv:2602.13515*, 2026a.

Jintao Zhang, Haoxu Wang, Kai Jiang, Kaiwen Zheng, Youhe Jiang, Ion Stoica, Jianfei Chen, Jun Zhu, and Joseph E. Gonzalez. SLA2: Sparse-linear attention with learnable routing and QAT. *arXiv preprint arXiv:2602.12675*, 2026b.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Chen Sun, Ming-Hsuan Yang, Forrester Cole, Trevor Darrell, and Deqing Sun. LoGeR: Long-context geometric reconstruction with hybrid memory. *arXiv preprint arXiv:2603.03269*, 2026c.

Peiyuan Zhang, Xiangyu Zhao, Hongbo Liu, Xiaoxing Hu, Mingxin Liu, Shuran Ma, Yunhang Shen, Jian Hu, Haihan Gao, Haoyu Cao, and Xue Yang. FIRM-Video: Check before you score for reliable text-to-video reward modeling. *arXiv preprint arXiv:2608.21839*, 2026d.

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*, 2018.

Tianchen Zhao, Tongcheng Fang, Haofeng Huang, Enshu Liu, Rui Wan, Widyadewi Soedarmadji, 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. *arXiv preprint arXiv:2406.02540*, 2024.

Haoyi Zhu. Reflections on Video DeltaNet. Blog post, aug 2026. URL `https://haoyizhu.site/blog/video-delta-rule/`.

Haoyi Zhu, Haozhe Liu, Yuyang Zhao, Tian Ye, Junsong Chen, Jincheng Yu, Tong He, Song Han, and Enze Xie. SANA-WM: Efficient minute-scale world modeling with hybrid linear diffusion transformer. *arXiv preprint arXiv:2605.15178*, 2026.

The appendix is organized as follows. Appendix A gives the mathematical derivations and the complete bidirectional VDA block algorithm. Appendix B documents architecture adaptation. Appendix C reports the complete quality comparison across denoising budgets. Appendix D provides kernel details and additional efficiency results. Appendix E contains paired qualitative examples.

A Mathematical and Algorithmic Details

A.1 Proof of Proposition 1

Consider two entering states with identical prepared features and gates for frame t. Substituting $\bar{S}_t = S_{t-1}\,\mathrm{Diag}(\alpha_t)$ into Equation (5) separates the inherited state from the new write:

$$\begin{aligned} S_t &= S_{t-1}M_t + J_t, \\ M_t &= \mathrm{Diag}(\alpha_t)(I + A_t)^{-1}, \qquad J_t = B_t(I + A_t)^{-1}. \end{aligned} \tag{A.1}$$

For two entering states with the same frame inputs, $\Delta S_t = \Delta S_{t-1}M_t$. Since $\beta_{t,u} \geq 0$, the Gram matrix is positive semidefinite: $x^\top A_t x = \sum_u \beta_{t,u}(k_{t,u}^\top x)^2 \geq 0$ for every x. Writing $A_t = Q\,\mathrm{Diag}(\lambda_i)Q^\top$, with $\lambda_i \geq 0$, shows that $(I + A_t)^{-1}$ has eigenvalues $1/(1+\lambda_i) \in (0, 1]$. Together with $0 \leq \alpha_{t,j} \leq 1$, this gives

$$\begin{aligned} \|\Delta S_t\|_F &\leq \|\Delta S_{t-1}\|_F \|M_t\|_2, \\ \|M_t\|_2 &\leq \|\,\mathrm{Diag}(\alpha_t)\|_2 \|(I + A_t)^{-1}\|_2 \leq 1. \end{aligned} \tag{A.2}$$

This proves Proposition 1 without requiring the decay matrix and Gram matrix to commute. The bound concerns the inherited state at fixed frame inputs; it does not bound new writes or the full nonlinear network.

A.2 Independent and Joint Frame Writes

The difference between independent and joint frame writes can be seen by expanding Equation (3c):

$$S_t^{\mathrm{batch}} = \bar{S}_t + \sum_{u=1}^{U} \beta_{t,u}(v_{t,u} - \bar{S}_t k_{t,u})k_{t,u}^\top.$$

Every correction uses the same pre-update state, so a patch's residual ignores the other current-frame patches.

By comparison, the first-order condition for VDA can be rearranged as

$$S_t = \bar{S}_t + \sum_{u=1}^{U} \beta_{t,u}(v_{t,u} - S_t k_{t,u})k_{t,u}^\top.$$

Here every residual depends on the shared updated state S_t. Overlapping keys accumulate in A_t, so Equation (5) adjusts their joint update.

With $\alpha_t = \mathbf{1}$, the additive inherited-state factor is $I - A_t$. Its eigenvalues $1 - \lambda_i(A_t)$ leave $[-1, 1]$ if any $\lambda_i(A_t) > 2$. For unit keys and sigmoid write gates, $\lambda_{\max}(A_t) \leq \mathrm{tr}(A_t) = \sum_u \beta_{t,u} \leq U$, and aligned keys can approach this bound. Thus per-token normalization alone does not ensure stability.

Both updates start from token-level $1/\sqrt{d_k}$ key normalization. SANA-WM additionally scales each key by $1/\sqrt{U}$ (Zhu et al., 2026), changing the additive erase statistic to A_t/U and the value–key write statistic to $B_t/\sqrt{U}$; the inherited-state factor $I - A_t/U$ is then non-expansive. VDA uses $(I + A_t)^{-1}$ and requires no additional frame-size scale.

The inverse also responds to the observed key correlations. If all U positions repeat a unit key k with a common gate β, writing $s = \bar{S}_t k$ and $\bar{v} = U^{-1}\sum_u v_u$ gives

$$S_t k = \frac{1}{1 + U\beta}s + \frac{U\beta}{1 + U\beta}\bar{v}. \tag{A.3}$$

Repeated keys thus receive a saturating $U\beta/(1+U\beta)$ weight, whereas orthogonal keys each receive $\beta/(1+\beta)$. SANA-WM's scaled-key readout instead assigns β to repeated-key consensus and β/U to each orthogonal direction. The distinction is geometric: VDA adjusts the effective write to the key correlations actually present in the frame, whereas the additive rule applies the same token-count scale in every direction.

A.3 Boundary-State Readout and Decay Bridge

The linear branch represents only context outside the local Softmax window and the two global anchors. Anchor entries already visible to Softmax are therefore included only once, and anchor indices 0 and $F-1$ are excluded from the linear states. Let P_j be the forward state through interior frame j and R_j the reverse state from the last interior frame down to j. Their virtual initial states are $P_0 = R_{F-1} = S_T/2$. For query t, let $[\ell_t, h_t]$ be its local window clipped to interior frame indices. The prepared query reads

$$\begin{aligned} \widetilde{S}_t &= P_{\ell_t-1}\,\mathrm{Diag}\left(\prod_{r=\ell_t}^{t}\alpha_r\right) \\ &\quad + R_{h_t+1}\,\mathrm{Diag}\left(\prod_{r=t}^{h_t}\alpha_r\right), \\ o^L_{t,u} &= \widetilde{S}_t q_{t,u}. \end{aligned} \tag{A.4}$$

The products are channel-wise. When distant video is absent on one side, the corresponding initial text state is used instead of a fabricated frame state. The bridge applies decay across the excluded local interval without adding its frame writes. If a window covers the complete clip, the implementation bypasses the linear pathway and uses the dense attention path. Clips consisting only of anchor frames also receive no linear video output.

A.4 Bidirectional VDA Block Algorithm

One hybrid attention block executes the following steps:

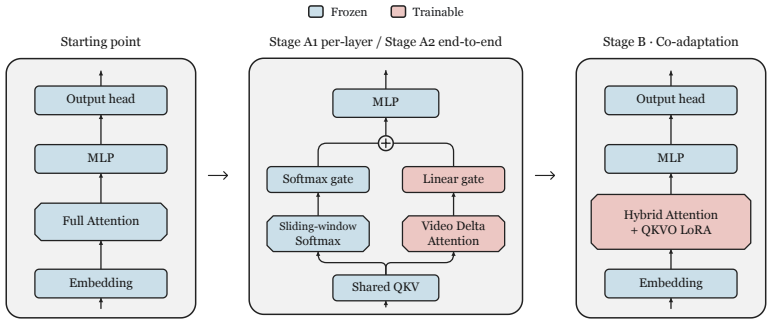
1. Compute shared QKV and retained multimodal Softmax with chunk and anchor masks.
2. Remove anchor video rows from the linear inputs; prepare Q/K/V features and frame gates.
3. Compute A_t, B_t, α_t for all interior frames. Form the text state from the prompt.
4. Compute $(I + A_t)^{-1}$ and form transition M_t and injection J_t for every frame and head.
5. Scan $S \leftarrow SM_t + J_t$ in both temporal directions, each initialized with $S_T/2$.
6. Gather outside-window states and apply the decay bridge in Equation (A.4).
7. Read each query; apply RMSNorm and the linear gate; zero anchor rows.
8. Gate and project Softmax; add the projected linear output to video rows.

B ARCHITECTURE ADAPTATION

This section documents how the pretrained dense backbone is converted into VDN-H3.

B.1 PROGRESSIVE ARCHITECTURE ADAPTATION

Figure 6 summarizes how the pretrained dense backbone is converted into VDN-H3. The diagram separates the frozen starting point, per-layer and end-to-end alignment in Stages A1 and A2, and the LoRA co-adaptation used in Stage B.



(a) MiniMax H3 (b) Hybrid attention layer (c) VDN-H3

Figure 6: Training pipeline for progressive adaptation of the pretrained H3 backbone. (a) The dense model is the frozen starting point. (b) Stage A1 aligns each Linear branch independently, then Stage A2 trains the assembled hybrid layers end to end while the pretrained weights and Softmax gates remain frozen. (c) Stage B jointly trains the Linear pathway, Softmax gates, and QKVO LoRA adapters.

B.2 OPTIMIZATION HYPERPARAMETERS

We use AdamW with linear warmup followed by cosine decay. Stages A1, A2, and B use a peak learning rate of 10^{-4} and decay to 5×10^{-6}. During few-step distillation, the generator and fake-score model use peak learning rates of 10^{-5} and 2×10^{-5}, decaying to 10^{-6} and 2×10^{-6}, respectively. Following Chimera (Ge et al., 2026), parameters are divided into learning-rate groups by module scale: large fan-in matrices use the base schedule, while smaller vector parameters use larger multipliers. In Stage B, the Linear branch uses $0.4\times$ the LoRA learning rate. Table 6 lists the remaining stage-specific settings.

Table 6: Hyperparameters for architecture adaptation. The table covers Stages A1, A2, and B; few-step distillation is described in Section 4.2.

Setting	**A1: per-layer**	**A2: end-to-end**	**B: LoRA co-adaptation**
Parameter status	**Train:** one Linear branch **Freeze:** backbone and Softmax gate	**Train:** all Linear branches **Freeze:** backbone and Softmax gates	**Train:** branches, gates, and QKVO LoRA **Freeze:** backbone FFN, embeddings, and head
Steps	200	500	2,000
Optimizer / schedule	AdamW; linear warmup → cosine	AdamW; linear warmup → cosine	AdamW; linear warmup → cosine
β_1, β_2	0.9, 0.999	0.9, 0.999	0.9, 0.999
Peak / minimum LR	10^{-4} / 5×10^{-6}	10^{-4} / 5×10^{-6}	10^{-4} / 5×10^{-6}
Warmup steps	50	50	200
Weight decay	0	0	0
Gradient clip	0.1 per layer	1.0 global	1.0 global
LR multipliers	large 1.0 / small 5.0	large 1.0 / small 5.0	LoRA 1.0 / branch 0.4 / small 2.0
LoRA rank / scale	–	–	64 / 64

C Additional Quality Results

C.1 Full Evaluation Suite

Figures 7, 8, and 9 report the complete 50-NFE, 8-NFE, and 4-NFE evaluation suites. EvalCrafter VQA_A and VQA_T measure aesthetic and technical quality (Liu et al., 2024); Q-Align assesses perceived visual quality (Wu et al., 2024); and FAST-VQA and DOVER++ provide no-reference video-quality scores (Wu et al., 2022; 2023). The expanded suite also reports RAFT mean flow as a motion-magnitude diagnostic (Teed & Deng, 2020) and FIRM-Video Instruction Following, Perceptual Quality, and World Coherence on a 1–5 scale (Zhang et al., 2026d). For first–last-frame-to-video (FL2VA), endpoint fidelity is measured by the mean PSNR, SSIM (Wang et al., 2004), and LPIPS (Zhang et al., 2018) at the two conditioning frames. Higher is better for all quality metrics except LPIPS; RAFT flow indicates motion rather than quality. Q-Align, FAST-VQA, and DOVER++ use a $\times 100$ scale. Within each metric, the figures use labeled, truncated vertical scales, and asterisks reproduce paired tests against 50-NFE Dense H3 only.

50-NFE results. Figure 7 adds FAST-VQA and FL2VA LPIPS to the ten metrics in the main-paper comparison and reports the complete significance-marked suite. VDN-H3 remains close to Dense H3 and scores higher than Radial on six of the eight quality metrics available for both models.

8-NFE results. Figure 8 likewise adds FAST-VQA and FL2VA LPIPS to the main comparison. VDN-H3 leads all five no-reference metrics and remains close to Dense H3 on FIRM and FL2VA; its VQA_T score is 10.78 points higher than FastH3 v2 (FastVideo Team, 2026b).

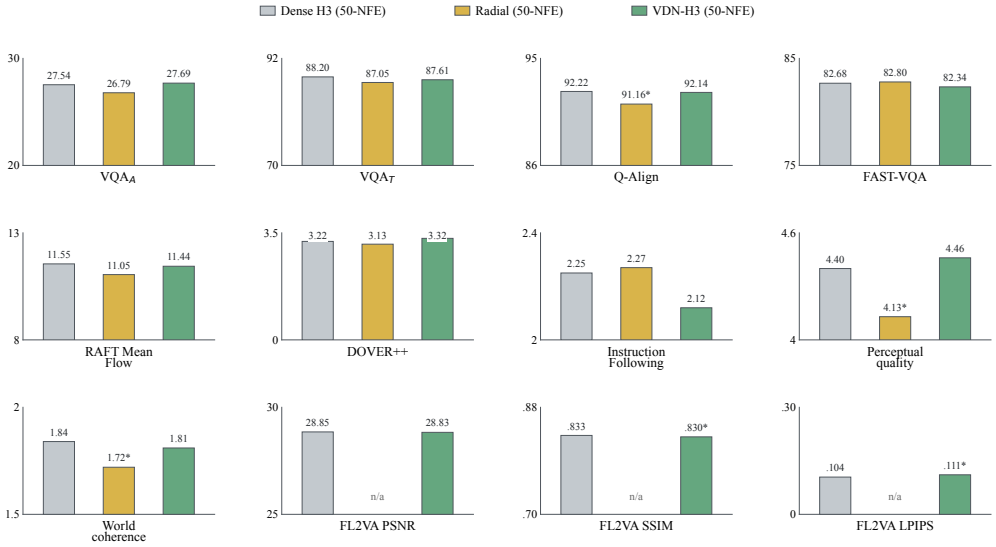


Figure 7: 50-NFE quality comparison of Dense H3, Radial, and VDN-H3 across the complete evaluation suite. VDN-H3 is the Stage-B checkpoint before few-step distillation.

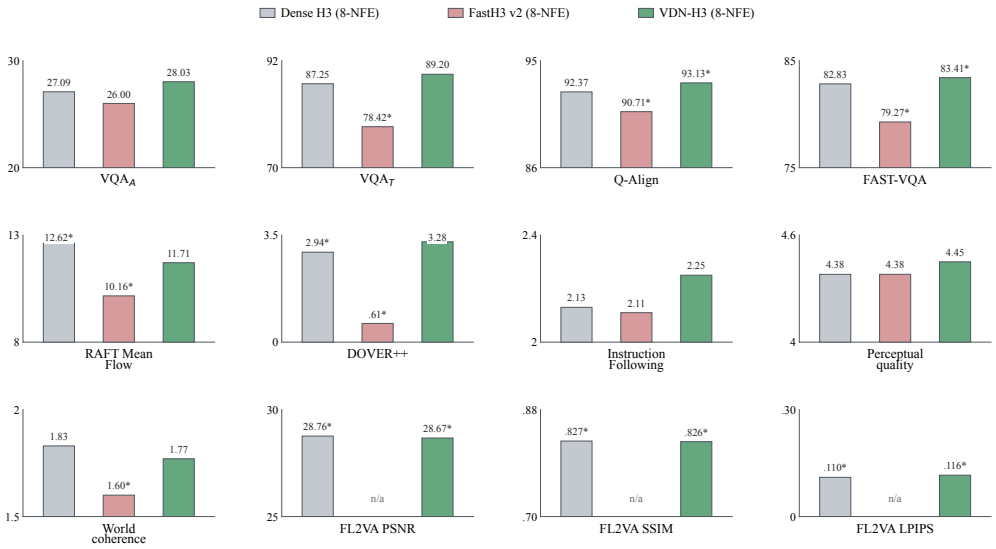


Figure 8: 8-NFE quality comparison across the complete evaluation suite.

4-NFE results. Figure 9 shows that VDN-H3 exceeds Dense H3 and FastH3 v1 (FastVideo Team, 2026a) on all five no-reference metrics. Relative to FastH3 v1, VQA_T improves by 12.64 points and FL2VA endpoint fidelity is higher.

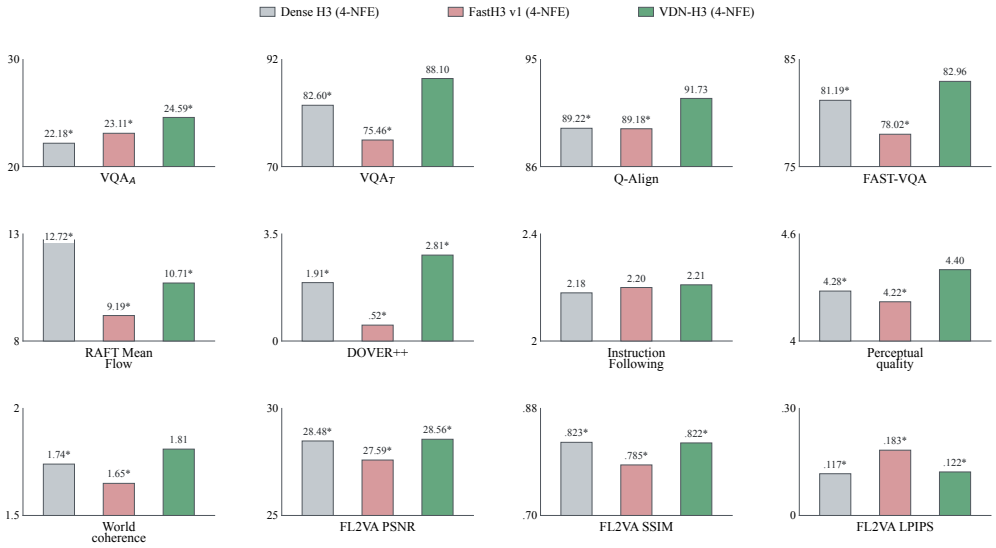


Figure 9: 4-NFE quality comparison. VDN-H3 retains a clear margin over FastH3 v1 at the same NFE count.

D SYSTEMS AND EFFICIENCY

D.1 H200 DEPLOYMENT AND DURATION SCALING

Table 7 complements the B200 deployment results in the main paper. On one H200 at 102 latent frames, VDN-H3 reduces 50-NFE denoising from 1767.3 to 557.9 seconds ($3.2\times$); 8-NFE sampling and eight-GPU head sharding reduce it further to 89.3 and 12.5 seconds. From 42 to 102 latent frames, retained attention density falls from 42.1% to 20.0%, while the 50-NFE backbone speedup grows from $1.8\times$ to $3.2\times$.

Table 7: H200 efficiency. Gains are incremental and speedups are cumulative against one-GPU Dense H3. The one-GPU duration sweep compares Dense H3 and VDN-H3 at 50 NFE.

Configuration	#GPU	NFE	s/NFE	End-to-end latency	Gain	Speedup
Dense H3	1	50	35.35 s	1767.3 s	–	$1.0\times$
VDN-H3	1	50	11.16 s	557.9 s	$3.2\times$	$3.2\times$
+ few-step	1	8	11.16 s	89.3 s	$6.3\times$	$19.8\times$
+ distributed	8	8	1.56 s	12.5 s	$7.1\times$	$141.6\times$

Latent frames	42	72	87	102
Attention density	42.08%	26.82%	22.85%	19.98%
Attention speedup	$2.0\times$	$3.0\times$	$3.5\times$	$4.0\times$
VDN-H3 latency	218.1 s	388.8 s	476.9 s	557.9 s
End-to-end speedup	$1.8\times$	$2.5\times$	$2.8\times$	$\mathbf{3.2}\times$

D.2 KERNEL OPTIMIZATION DETAILS

Feature preparation. The linear branch L2-normalizes Q and K; applies depthwise 5×5 spatial and five-tap temporal filtering plus SiLU to K and V; omits rotary embeddings; and represents the prompt state as a leading virtual frame.

Fused VDA operators. *VDA-Prep* fuses feature preparation and layout conversion; *VDA-Stats* constructs A_t and B_t in one pass; *VDA-Gather* collects boundary states and applies the decay bridge; and *VDA-Epilogue* fuses RMS normalization, output gating, and final layout conversion.

Window Softmax. Queries are grouped by visible-key pattern for FlashAttention's varlen API, avoiding materialization of a global attention mask.

Execution and precision. Head-sharded VDA overlaps packed Window Softmax on a separate stream. Wide projections use MXFP8; recurrent states and small-matrix inverses remain in FP32. Precomputing AdaLN parameters removes conditioning from the per-block critical path.

Complete kernel ablations. Figure 10 reports every operator-level ablation on H200 and B200. On H200, the fused VDA stages provide 2.1–$11.4\times$ speedups and the fused inverse provides $4.6\times$. Eight-GPU head sharding turns chunk composition from single-GPU overhead into a $4.2\times$ speedup. B200 follows the same overall trend, showing that the gains are not specific to one accelerator generation.

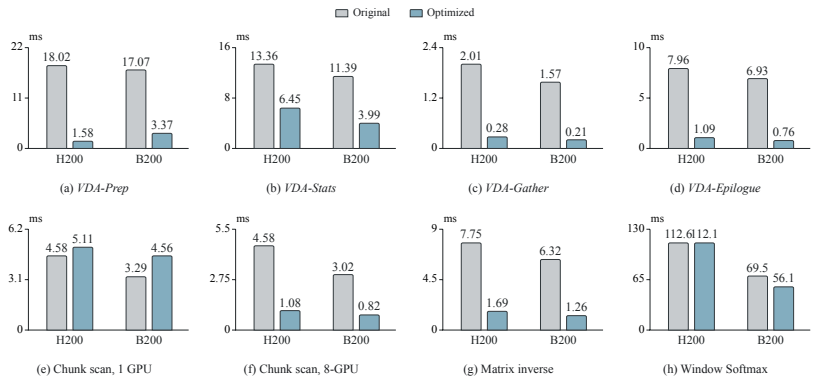
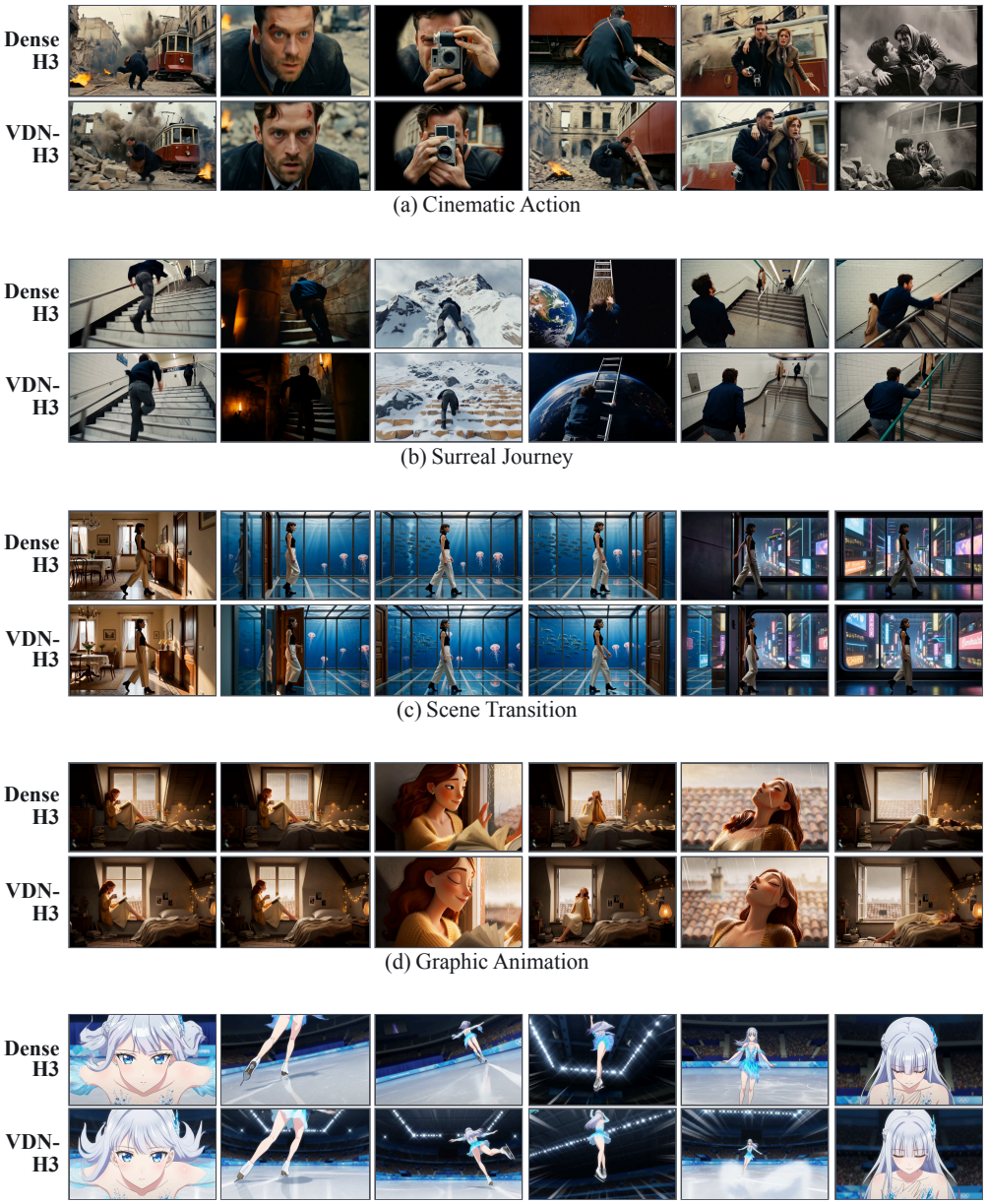


Figure 10: Complete kernel-level optimization ablations on H200 and B200.

E QUALITATIVE GALLERY

Figure 11 compares 50-NFE Dense H3 with 8-NFE VDN-H3 across cinematic action, surreal scenes, scene transitions, graphic animation, and animated sports. Each row uses synchronized frames from the same prompt to support direct comparison.



(a) Cinematic Action

(b) Surreal Journey

(c) Scene Transition

(d) Graphic Animation

(e) Animated Sports

Figure 11: Qualitative gallery. Six synchronized frames from five prompts, with 50-NFE Dense H3 above 8-NFE VDN-H3 in each pair.