# LONG-HORIZON STREAMING VIDEO GENERATION VIA HYBRID ATTENTION WITH DECOUPLED DISTILLATION

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Streaming video generation (SVG) distills a pretrained bidirectional video diffusion model into an autoregressive model equipped with sliding window attention (SWA). However, SWA inevitably loses distant history during long video generation, and its computational overhead remains a critical challenge to real-time deployment. In this work, we propose **Hybrid Forcing**, which jointly optimizes temporal information retention and computational efficiency through a hybrid attention design. First, we introduce lightweight linear temporal attention to preserve long-range dependencies beyond the sliding window. In particular, we maintain a compact key–value state to incrementally absorb evicted tokens, retaining temporal context with negligible memory and computational overhead. Second, we incorporate block-sparse attention into the local sliding window to reduce redundant computation within short-range modeling, reallocating computational capacity toward more critical dependencies. Finally, we introduce a decoupled distillation strategy tailored to the hybrid attention design. A few-step initial distillation is performed under dense attention, then the distillation of our proposed linear temporal and block-sparse attention is activated for streaming modeling, ensuring stable optimization. Extensive experiments on both short- and long-form video generation benchmarks demonstrate that Hybrid Forcing consistently achieves state-of-the-art performance. Notably, our model achieves **real-time**, **long** video generation at **29.5 FPS** on a single NVIDIA H100 GPU **without quantization or model compression**. The source code and trained models will be released.

## 1 INTRODUCTION

The rapid scaling of video diffusion models (VDMs) has substantially improved video generation quality, enabling high-fidelity synthesis from text, image, or video inputs (OpenAI, 2024; Google, 2025; Gao et al., 2025; Zhang et al., 2025d; Peng et al., 2025; Kong et al., 2024) to multimodal outputs (Wan et al., 2025; Li et al., 2025; ltx2, 2026; Gao et al., 2025; Kling, 2026; RunwayML, 2023; Bruce et al., 2024). However, most large-scale VDMs rely on bidirectional attention, whose quadratic memory and computation costs restrict generation to short clips and make real-time long-form deployment impractical (Zhang & Agrawala, 2025; Yin et al., 2025; Sand-AI, 2025; Huang et al., 2025; Yang et al., 2025; Chen et al., 2024). To improve efficiency, recent works convert pretrained bidirectional VDMs into autoregressive streaming models that generate videos chunk by chunk (Yin et al., 2025; Huang et al., 2025; Yang et al., 2025; Lu et al., 2025; Ji et al., 2025), typically using ODE-based initialization and Distribution Matching Distillation (DMD) (Dong et al., 2024; Yin et al., 2024; Wang et al., 2023; Wu et al., 2026) to obtain few-step causal students. During both distillation and inference, these models commonly rely on sliding window attention (SWA) with KV caching (Liu et al., 2021; Zhang et al., 2025c; Kodaira et al., 2025) to enforce causality, as illustrated in fig. 1(a). Nevertheless, as the window advances, earlier frames are evicted from the cache, forcing generation to depend on a recency-biased context and causing errors to accumulate over long rollouts (Yang et al., 2025; Cui et al., 2025; Dalal et al., 2025).

Existing streaming methods mitigate this issue with heuristic memory strategies (Liu et al., 2025; Yang et al., 2025; Lu et al., 2025; Ji et al., 2025). For example, attention-sink methods retain the initial frame in the KV cache to anchor generation (Yang et al., 2025; Xiao et al., 2023; Lu et al., 2025), as shown in fig. 1(b), but repeatedly attending to a static frame can restrict motion diversity and underuse informative intermediate frames. Dynamic sink updates based on token similarity (Ji



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.

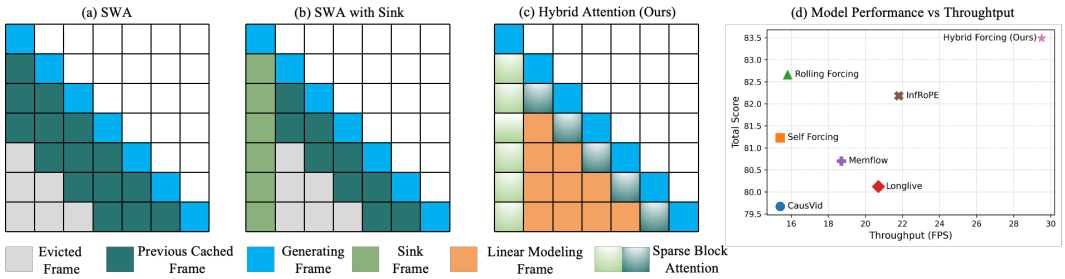


Figure 1: Illustration of our hybrid attention paradigm for SVG. (a) The standard SWA approach only caches the most recent frames, leading to significant error accumulation in long-form generation. (b) While sinking the first frame can reduce early drift, it sacrifices motion diversity and fails to capture long-horizon dependencies effectively. (c) Our hybrid attention combines long-term history retention with efficient local modeling, leveraging low-cost history linear modeling and sparse SWA to enhance performance. (d) Performance vs Throughput on VBench-Long.

et al., 2025) alleviate this limitation, but remain myopic because tokens selected for the current frame may not be useful for future frames. Moreover, these methods typically achieve only around 20 FPS on an H100 GPU (Liu et al., 2025; Yang et al., 2025; Ji et al., 2025; Chen et al., 2024; 2025a).

In this paper, we propose **Hybrid Forcing**, a streaming framework that unifies long-term history retention and efficient local modeling through hybrid attention, as illustrated in fig. 1(c). Specifically, we introduce a linear temporal attention pathway (Chen et al., 2025b; Team et al., 2025; Qiu et al., 2025) that compactly summarizes evicted tokens into a persistent KV state, enabling long-range dependency modeling without explicit sequence growth. We further incorporate block-sparse attention into SWA to prune redundant local interactions (Zhang et al., 2025b;a; Xi et al., 2025; Xia et al., 2025), reducing computation while preserving fine-grained temporal refinement. These two components are complementary: sparse local attention removes redundant short-range correlations, while the linear temporal pathway retains globally aggregated history, reallocating attention capacity toward informative temporal dependencies. To train the hybrid design stably, we adopt a decoupled distillation strategy: the model first undergoes a short dense-attention DMD phase to consolidate semantic representations, and then activates hybrid attention for streaming training.

Adapting pretrained VDMs to long-horizon streaming further introduces two stability challenges. First, pretrained models are exposed to limited temporal RoPE indices (*e.g.*, 21 frames) (Su et al., 2024; Wan et al., 2025), making naive extrapolation unstable during long rollouts (Yesiltepe et al., 2025; Cui et al., 2025); we therefore enforce a consistent relative temporal RoPE bound during both training and inference. Second, DMD-only long-horizon training can suffer from mode collapse (Wu et al., 2026; Zheng et al., 2025; Lu & Song, 2024), such as frozen motion or persistent camera drift. We mitigate this issue with a teacher-guided regularization loss that anchors the student to teacher rollouts from identical initial noise, preserving temporal diversity and motion dynamics. Extensive experiments show that Hybrid Forcing achieves state-of-the-art short- and long-horizon performance while enabling real-time streaming generation at **29.5 FPS** on a single H100 GPU. As illustrated in fig. 1(d), our method not only achieves the best overall *Total Score* on Vbench-Long, but also attains the highest throughput among all compared methods, highlighting its favorable balance between generation quality and efficiency.

## 2 Related Work

**Long-Duration Video Generation:** Much effort has been devoted to extending pretrained video diffusion models (VDMs) for longer video generation (Jin et al., 2024; Zhang & Agrawala, 2025; Henschel et al., 2025; Chen et al., 2025a). Representative methods, such as NUWA-XL (Yin et al., 2023), LaVie (Wang et al., 2025), extend temporal coverage through hierarchical generation, cascaded refinement, but often suffer from substantial computational costs and quality degradation over extended durations. Recently, more attention has been paid to causal autoregressive (AR)-based models for long video generation (Yin et al., 2025; Huang et al., 2025; Liu et al., 2025; Yang et al., 2025; Lu et al., 2025; Ji et al., 2025; Cui et al., 2025; Yesiltepe et al., 2025; Zhu et al., 2026). CausVid (Yin et al., 2025) reformulates bidirectional video diffusion into a causal model and leverages

DMD (Yin et al., 2024) to compress multi-step models into a few-step one. Self-Forcing (Huang et al., 2025) addresses the train-test mismatch by conditioning on self-generated frames to enhance temporal stability. Rolling Forcing (Liu et al., 2025) denoises consecutive frames with progressively increased noise. LongLive (Yang et al., 2025) introduces frame-level autoregression and KV recache tuning, further enhancing interactive generation. Despite the efficiency of these AR-based models, they fall short in **capturing long-term context due to the use of sliding window attention (SWA)**. Errors introduced in earlier frames accumulate over time, leading to degraded temporal coherence and unstable long-form generation.

**Sparse Attention:** Recent works have explored the sparsity in attention maps for video generation (Zhang et al., 2025a;b; Xi et al., 2025; Xia et al., 2025). Leveraging this property, Sparse Video Generation (Xi et al., 2025) methods define a fixed sparse attention mask to reduce the computational cost of attention. AdaSpa (Zhang et al., 2025a) introduces a blockified pattern to efficiently capture the hierarchical sparsity in DiTs, significantly reducing attention complexity. VSA (Zhang et al., 2025b) adapts the DeepSeek NSA (Yuan et al., 2025) to video DiTs, finetuning the model with newly introduced VSA modules to achieve better performance compared to training-free methods. SLA (Zhang et al., 2025a) introduces a learnable Sparse-Linear Attention mechanism into pretrained VDMs to accelerate inference speed. While these methods have demonstrated efficiency improvements in bidirectional video generation, their effectiveness in SVG remains unproven. Moreover, many of these approaches **fail to account for the streaming attention paradigm**, which can lead to kernel errors when applying to SVG models.

# 3 METHODS

## 3.1 PRELIMINARY

**Video Diffusion Model (VDM)**: Current VDMs usually adopt the flow-matching paradigm (Lipman et al., 2022; Liu et al., 2022), which builds a trajectory from clean video latents $\boldsymbol{x}_0 \sim p_{\text{data}}(\boldsymbol{x})$ to Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I})$. A point along this trajectory is interpolated as $\boldsymbol{x}_t = \alpha_t \boldsymbol{x}_0 + \beta_t \boldsymbol{\epsilon}$, with $\alpha_t$ and $\beta_t$ controlled by a noise scheduler (Song et al., 2020a; Karras et al., 2022; Zhao et al., 2023) and $t \in [0, 1]$. The model predicts the velocity field $\boldsymbol{v}_\theta$ at $\boldsymbol{x}_t$. The target velocity is defined as the difference between the noise and the clean data. The loss function is $\mathcal{L}_{FM} = \mathbb{E}_{\boldsymbol{x},\boldsymbol{\epsilon},t}\left[\|\boldsymbol{v}_\theta(\boldsymbol{x}_t, t) - (\boldsymbol{\epsilon} - \boldsymbol{x_0})\|^2\right]$. New video samples are generated by solving the ODE with the learned velocity field (Lipman et al., 2022; Song et al., 2020b).

**Streaming Video Generation**: Current VDMs are limited to generating short video clips due to the complexity for long video generation (Huang et al., 2025). SVG generates videos frame-by-frame, factorizing the distribution as $p(\boldsymbol{x}_0^{1:N}) = \prod_{i=1}^{N} p(\boldsymbol{x}_0^i \mid \boldsymbol{x}_0^{<i})$, where each $\boldsymbol{x}_0^i$ represents a video frame or a chunk of frames. Inference typically uses an SWA mechanism with a fixed window size $w$ to replace the original vanilla attention, limiting the scope to recent frames as $p(\boldsymbol{x}_0^{1:N}) = \prod_{i=1}^{N} p(\boldsymbol{x}_0^i \mid \boldsymbol{x}_0^1, \boldsymbol{x}_0^{i-w-1:i-1})$, where $\boldsymbol{x}_0^1$ is the sink frame (Yang et al., 2025). SVGs are often trained through a two-stage distillation pipeline. The first stage uses a pretrained bidirectional diffusion model to generate ODE trajectories, which are then mimicked by a student AR generator $g_\theta$ using a causal attention mask. The second stage applies Distribution Matching Distillation (DMD) (Yin et al., 2024) to compress the student model into a few-step generator. The DMD loss is:

$$\nabla_\theta D_{\text{KL}} = \mathbb{E}_{\epsilon\sim\mathcal{N}(0;\mathbf{I})}\big[-(s_{\text{real}}(x_t) - s_{\text{fake}}(x_t))\frac{\partial g_\theta}{\partial \theta}\big], \tag{1}$$

where $s_{\text{real}}$ and $s_{\text{fake}}$ are score functions (Song et al., 2020b). Despite improvements, two challenges remain: long-range context is lost as the sliding window advances, leading to temporal drift, and the attention computation within windows remains expensive, hindering real-time deployment.

## 3.2 HYBRID ATTENTION

SVG requires modeling both long-range temporal dependencies and fine-grained local interactions. However, standard scaled dot-product attention incurs $\mathcal{O}(N^2 d)$ complexity, making full-history attention infeasible in both memory and computation. This challenge motivates us to develop a hybrid attention mechanism as illustrated in fig. 2(b). Linear attention compresses distant history into a compact representation for modeling global temporal dependencies, while sparse attention

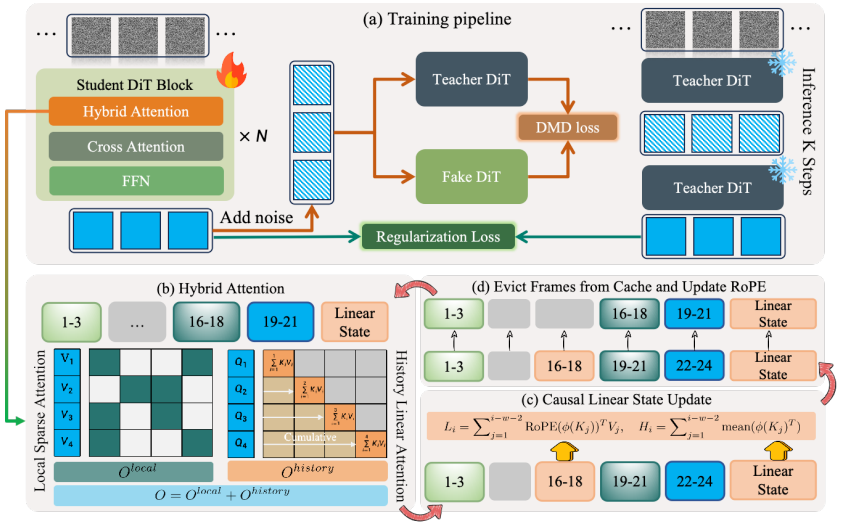


Figure 2: Framework of our training and inference pipeline. (a) An auxiliary regularization loss is introduced to mitigate training mode collapse. (b) The output of our hybrid attention is computed as the sum of sparse local window attention and linear history attention. (c) The linear state is updated prior to evicting outdated frames from the KV cache to preserve long-term information. (d) The temporal RoPE index is constrained within the maximum temporal index of pretrained model.

preserves fine-grained dependencies within a sliding window for capturing local window interactions. This hybrid approach enables efficient modeling of both long-range and local dependencies in a complementary manner, minimizes memory and computational overhead.

*Linear attention for long-range history modeling.* Instead of storing all historical frames $\boldsymbol{x}_0^{1:i-w-2}$, we maintain a dynamically updated KV state that aggregates evicted frames. The linear state update is performed when a frame is evicted from the cache, as shown in fig. 2(c). The state update rule is:

$$L_i = \sum_{j=1}^{i-w-2} \text{RoPE}(\phi(K_j))^T V_j, \quad H_i = \sum_{j=1}^{i-w-2} \text{mean}(\phi(K_j)^T), \tag{2}$$

where $\phi(\cdot)$ is the activation function. The history linear attention output for frame $i$ is computed as:

$$\boldsymbol{O}_i^{\text{history}} = \text{Linear}\left(\frac{\text{RoPE}(\phi(Q_i))L_i}{\phi(Q_i)H_i}\right). \tag{3}$$

Unlike standard linear attention that attends to all frames, we restrict the receptive field to evicted frames only. As the sliding window advances, $L_i$ and $H_i$ are updated online with newly evicted features, resulting in constant memory cost independent of video length. This mechanism enables effective long-horizon dependency modeling with negligible overhead. Importantly, because distant information is compressed into a compact state, we can safely reduce the local window size (*e.g.*, from $w = 21$ to $w = 9$), further lowering inference latency.

*Sparse SWA for efficient local modeling.* While linear attention captures global consistency, local temporal interactions still require careful modeling. We therefore retain SWA within the recent $w$ frames but accelerate it using block-sparse computation. Building upon FlashAttention (Dao et al., 2022), which improves memory efficiency via block-wise tiling, we further exploit the sparsity of attention maps in VDMs (Xi et al., 2025; Zhang et al., 2025a) to reduce redundant computation. Specifically, we compute a block importance score as:

$$\boldsymbol{A}_{ij} = \text{Mean}(\boldsymbol{q}_i)\,\text{Mean}(\boldsymbol{k}_j)^T, \tag{4}$$

where $\{q_i\}$, $\{k_j\}$ are blocks from $Q_i$ and $K_{1,i-w:i}$. We retain only the top-$K$ most activated blocks, forming a binary mask $M_{ij}$, whose value is 1 if $\boldsymbol{A}_{ij} \in$ TopK and 0 otherwise. Attention is then computed only over active blocks as:

$$\boldsymbol{S}_{ij} = \boldsymbol{q}_i \boldsymbol{k}_j^T / \sqrt{d}, \quad \boldsymbol{P}_{ij} = \text{OnlineSoftmax}(\boldsymbol{S}_{ij}), \quad \boldsymbol{O}_i^{\text{local}} = \sum_{j:M_{ij}=1} \boldsymbol{P}_{ij} \boldsymbol{v}_j. \tag{5}$$

*Hybrid attention.* Finally, the output of our hybrid attention is the sum of compressed global history and sparse local attention:

$$\boldsymbol{O}_i = \boldsymbol{O}_i^{\text{history}} + \boldsymbol{O}_i^{\text{local}}. \tag{6}$$

This decomposition naturally separates long-term dependency modeling from short-term interaction modeling, achieving both temporal coherence and real-time efficiency without increasing memory complexity. Specially, the sparse local attention mechanism effectively eliminates redundant short-range correlations and unnecessary local computations, while the linear temporal pathway ensures the preservation of globally aggregated temporal history. This dual approach reallocates attention capacity towards the most informative dependencies over time.

### 3.3 Relative RoPE

Previous work clamps the temporal RoPE index at inference time (Yesiltepe et al., 2025), While this strategy can alleviate extrapolation errors in simple cases and provide unbounded generation capacity, it cannot provide stable long-duration streaming. To better address these issues, we propose to incorporate *relative RoPE directly into the training process*. Specifically, we impose a cap on the maximum temporal RoPE index during both training and inference, and represent temporal positions as relative offsets, as illustrated in fig. 2(d). The modified temporal RoPE formulation is defined as:

$$RoPE_{\text{temp}}(i) = \begin{cases} [i-2,\, i-1,\, i], & \text{if } i \leq \mathcal{T}_0, \\ [\mathcal{T}_0-2,\, \mathcal{T}_0-1,\, \mathcal{T}_0], & \text{otherwise}, \end{cases} \tag{7}$$

where $\mathcal{T}_0$ denotes the maximum temporal RoPE index used in the pretrained VDM. All spatial coordinates remain unchanged, and the temporal RoPE indices in the KV cache are updated based on the offsets to the current chunk. Meanwhile, we perform long-horizon rollouts during distillation. This exposes the model to realistic streaming settings under relative RoPE, allowing it to adapt to the capped temporal indexing mechanism. As a result, the train–test discrepancy is substantially reduced, leading to improved temporal coherence and stability in extended video generation.

### 3.4 Regularization Loss

While DMD (eq. (1)) is effective for accelerating inference, we observe that long-term DMD training in streaming settings often leads to mode collapse (Wu et al., 2026; Zheng et al., 2025). In practice, this manifests as reduced motion diversity and systematic artifacts, such as uncontrollable camera drift (e.g., persistent pan-left). To mitigate this, we introduce a regularization term, which anchors the student generator to the teacher's denoising trajectory during early timesteps, as shown in fig. 2(a). Specifically, given a distilled $N$-step student model, we supervise its first generation step using the teacher's full rollout result from the same initial noise:

$$\mathcal{L}_{\text{reg}} = \mathbb{E}_{\epsilon} \left\| f\big(g_\theta(\epsilon, 1)\big) - \text{rollout}(g_\phi, \epsilon, 1) \right\|, \tag{8}$$

where $f(\cdot)$ maps the student output at timestep 1 to the $\boldsymbol{x}_0$ space, and $\text{rollout}(\cdot)$ denotes the complete denoising process of the frozen teacher model $g_\phi$.

This formulation is motivated by the observation that the initial noise latent $\epsilon$ lies within the training distribution of both the few-step student and the multi-step teacher. As a result, the teacher's rollout provides a stable and high-quality reference for the desired generation trajectory, effectively constraining the student against degenerate modes. This supervision is applied entirely in latent space, avoiding the overhead of pixel-space decoding. For the remaining steps, we retain the standard DMD objective to refine visual fidelity. (Note that the rollout noise–video pairs can be precomputed to eliminate rollout latency during training.) Thus, the final loss function is defined as $\mathcal{L}_{distill} = \mathcal{L}_{DMD} + \lambda \mathcal{L}_{Reg}$ when $t = 1$, and $\mathcal{L}_{distill} = \mathcal{L}_{DMD}$ for all subsequent steps.

### 3.5 Decoupled Distillation

We observe that at the early stage of training, tokens projected from ODE-based initialization checkpoints contain limited semantic information. During this phase, dense attention is essential for refining spatial details and suppressing artifacts, as it enables unrestricted information exchange across tokens. Introducing hybrid attention too early at this stage can restrict information flow and compress noisy, semantically immature states, trapping optimization in suboptimal minima.

Motivated by this observation, we adopt a decoupled distillation strategy that explicitly separates semantic representation learning from structured historical modeling. Instead of jointly learning both objectives from the outset, we first conduct a DMD-only distillation phase under dense attention, without any architectural or attention modifications. This stage focuses purely on forming information-rich token representations, yielding a reliable checkpoint capable of coherent short-horizon generation under reduced sampling steps. After semantic representations are sufficiently structured, we activate the proposed linear temporal attention and block-sparse attention modules. At this point, the model can effectively learn structured long-range dependency modeling and reallocate attention capacity toward informative temporal interactions without suffering from premature distillation. By disentangling representation formation from temporal structuring, the proposed decoupled training paradigm stabilizes optimization and consistently improves performance. A complete description of the training procedure and algorithmic details is provided in the **Appendix**.

# 4 EXPERIMENT

## 4.1 EXPERIMENT SETTING

**Implementation Details.** Following Self-Forcing (Huang et al., 2025), we build Hybrid Forcing upon Wan2.1-T2V-1.3B (Wan et al., 2025), a pretrained bidirectional text-to-video diffusion model that generates 81-frame videos at a resolution of $832 \times 480$. We initialize our model from the publicly released ODE checkpoint of Self-Forcing to ensure fair comparison with prior streaming baselines (Yang et al., 2025; Ji et al., 2025; Liu et al., 2025). During DMD distillation, text prompts are sampled from a filtered and LLM-augmented version of VidProM (Wang & Yang, 2024), with a global batch size of 32. As described in Sec. 3.5, training proceeds in two phases: we first conduct 1000 steps under the original Self-Forcing paradigm to establish semantic knowledge, followed by 1000 additional steps with hybrid attention enabled. All methods operate in a chunk-wise streaming manner, where each chunk contains three latent frames. The sliding window size is set to 9. We retain the first three frames as sink frames in the KV cache. Block-sparse attention is implemented using Triton (Pytorch, 2026), with a top-20% sparsity ratio. Increasing the sparsity ratio may lead to performance degradation, while yielding only marginal speed improvements in our preliminary experiments. The maximum relative temporal RoPE index is capped at 21 during both training and inference. The regularization loss weight is set to 0.05. We use AdamW to optimize both the student generator $g_\theta$ (learning rate $1.5 \times 10^{-6}$) and the fake score model (learning rate $2 \times 10^{-7}$). The generator is updated once for every five fake score updates.

**Evaluation Metrics.** For short-form evaluation, we adopt VBench (Huang et al., 2025). Following prior work (Huang et al., 2025; Yang et al., 2025), we expand the test prompts using Qwen2.5-7B-Instruct (Bai et al., 2023) for comprehensive evaluation. For long-form evaluation, we use VBench-Long and generate 30-second videos for quantitative comparison. We also conduct a user study where users rank the overall video quality across all methods and provide the results in the **appendix**. To ensure fairness, all quantitative comparisons are conducted under identical settings: 30-second duration at 16 FPS. Following Self-Forcing, we additionally report model throughput measured on a single NVIDIA H100 GPU.

## 4.2 RESULTS

**Short-Duration Evaluation.** We compare Hybrid Forcing against a broad range of state-of-the-art models, including bidirectional diffusion models Wan2.1-1.3B (Wan et al., 2025), multi-step autoregressive models SkyReels-V2 (Chen et al., 2025a), MAGI-1 (Sand-AI, 2025), NOVA (Deng et al., 2024), Pyramid Flow (Jin et al., 2024), as well as distilled few-step autoregressive models CausVid (Yin et al., 2025), Self-Forcing (Huang et al., 2025), Long Live (Yang et al., 2025), Rolling Forcing (Liu et al., 2025), and MemFlow (Ji et al., 2025).

As shown in Table 1, Hybrid Forcing consistently achieves the best performance across all metrics, including Total, Quality, and Semantic scores. Compared with multi-step autoregressive models, our method improves the Dynamic metric by 38.4% and achieves a $+1.22$ gain in overall semantic and style consistency, even outperforming the strongest multi-step baseline ($+0.58$ for Pyramid Flow). Compared with few-step autoregressive approaches, Hybrid Forcing delivers consistently superior overall performance while achieving the highest throughput, improving the throughput by 35% over

Table 1: **Comparison with state-of-the-arts** open-source video generation models of similar parameter sizes and resolutions. Evaluation scores are calculated on the extended prompt suite of VBench (Huang et al., 2025). The FPS is reported on a single H100 GPU. The best and 2nd-best results are highlighted in **bold** and underline, respectively.

| Model | #Params | Resolution | Throughput (FPS) ↑ | Evaluation scores ↑ | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Total | Quality | Semantic | Dynamic | Overall | Consistency |
| *Diffusion models* | | | | | | | | | |
| Wan2.1 (Wan et al., 2025) | 1.3B | 832×480 | 0.78 | 83.37 | 84.30 | 79.65 | 61 | 25.50 | |
| *Multiple-Step Autoregressive Models* | | | | | | | | | |
| SkyReels-V2 (Chen et al., 2025a) | 1.3B | 960×540 | 0.49 | 81.97 | 83.96 | 74.01 | 37 | 25.31 | |
| MAGI-1 (Sand-AI, 2025) | 4.5B | 832×480 | 0.19 | 78.88 | 81.67 | 67.72 | 42 | 25.56 | |
| NOVA (Deng et al., 2024) | 0.6B | 768×480 | 0.88 | 80.31 | 80.66 | 78.92 | 46 | 25.42 | |
| Pyramid (Jin et al., 2024) | 2B | 640×384 | 6.7 | 80.98 | 83.37 | 71.42 | 52 | 26.08 | |
| *Few-Step Autoregressive Models* | | | | | | | | | |
| CausVid (Yin et al., 2025) | 1.3B | 832×480 | 15.4 | 82.65 | 84.08 | 76.91 | **72** | 25.72 | |
| Self Forcing (Huang et al., 2025) | 1.3B | 832×480 | 15.4 | 82.89 | 83.90 | 78.83 | 69 | 26.57 | |
| Rolling Forcing (Liu et al., 2025) | 1.3B | 832×480 | 15.8 | 82.95 | 83.89 | 79.22 | 52 | 26.46 | |
| Long Live (Yang et al., 2025) | 1.3B | 832×480 | 20.7 | 83.05 | 83.79 | 80.10 | 50 | 26.29 | |
| MemFlow (Ji et al., 2025) | 1.3B | 832×480 | 18.7 | 83.01 | 84.22 | 78.15 | 59 | 26.19 | |
| InfRoPE (Ji et al., 2025) | 1.3B | 832×480 | 21.8 | 81.18 | 83.47 | 74.02 | 69 | 26.28 | |
| Hybrid Forcing | 1.3B | 832×480 | **29.5** | **83.60** | **84.45** | **80.19** | **72** | **26.72** | |

Table 2: **Comparison with state-of-the-art** open-source few-step video generation models on long-duration inference setting. *Consis.* means Overall consistency. The best and 2nd-best results are highlighted in **bold** and underline, respectively.

| Model | #Params | Resolution | Throughput (FPS) ↑ | Evaluation scores ↑ | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | Total | Quality | Semantic | Dynamic | Consis. |
| CausVid (Yin et al., 2025) | 1.3B | 832×480 | 15.4 | 79.67 | 83.11 | 65.89 | 51 | 22.69 |
| Self Forcing (Huang et al., 2025) | 1.3B | 832×480 | 15.4 | 81.23 | 82.74 | 75.20 | 52 | 25.23 |
| Rolling Forcing (Liu et al., 2025) | 1.3B | 832×480 | 15.8 | 82.66 | 83.81 | 78.03 | 53 | 25.77 |
| Long Live (Yang et al., 2025) | 1.3B | 832×480 | 20.7 | 80.12 | 82.26 | 71.56 | 34 | 24.20 |
| MemFlow (Ji et al., 2025) | 1.3B | 832×480 | 18.7 | 80.70 | 83.40 | 69.92 | 61 | 23.91 |
| InfRoPE (Yesiltepe et al., 2025) | 1.3B | 832×480 | 21.8 | 82.18 | 83.47 | 77.02 | 62 | 25.62 |
| Hybrid Forcing | 1.3B | 832×480 | **29.5** | **83.49** | **84.67** | **78.78** | **74** | **26.02** |

the second-best streaming baseline. Notably, methods such as Long Live and MemFlow increase throughput by shrinking the local window size (*e.g.*, from 21 to 12), which significantly compromises long-range consistency. In contrast, even with a smaller window size of 9, Hybrid Forcing preserves global coherence through the linear temporal pathway, demonstrating the effectiveness of hybrid attention. Remarkably, Hybrid Forcing surpasses the bidirectional Wan2.1 model in overall performance while achieving approximately 30× higher throughput under identical hardware settings. Other detailed metrics and qualitative comparisons are provided in the **Appendix**.

**Long-Duration Evaluation.** To evaluate long-horizon streaming performance, we extend generation to 126 latent frames.The evaluation is conducted on VBench-Long (Huang et al., 2025), and the results are summarized in Table 2. We see that Hybrid Forcing achieves the best overall performance and the highest throughput among all few-step autoregressive baselines. While competing methods typically exhibit noticeable degradation in Quality or Dynamic scores under long-duration generation, our approach maintains stable performance and achieves even improved Quality compared to short-form results. In particular, methods such as CausVid and Self-Forcing show strong dynamics in 5-second generation, but experience significant drops in 30-second settings. Our Hybrid Forcing shows state-of-the-art Dynamic performance and exceeds the second-best method by 19.3%. Furthermore, our hybrid attention design (short local window + linear history aggregation) consistently outperforms both long-window dense attention models and short-window streaming baselines, demonstrating superior temporal consistency and semantic alignment in extended generation. Qualitative long-form results are shown in Fig. 3. We can see that CausVid exhibits significant error accumulation, leading to progressive saturation drift over time. While Self-Forcing and InfRoPE perform reasonably well in short-duration generation, they experience noticeable degradation in sky regions for longer sequences. LongLive captures minimal scene dynamics, whereas Rolling Forcing and MemFlow show moderate performance. In contrast, our method faithfully reproduces the full "house melt" process with high motion dynamics and visual fidelity. Moreover, our model maintains temporal stability over unbounded horizons, successfully generating coherent videos of more than 10 minutes without noticeable degradation. Visual evidences are provided in the **Appendix**.




Figure 3: Frame-by-frame qualitative results on 30-second videos. More comparison results and video demos are provided in the **Appendix**.

Table 3: Ablation studies on our design. The results are evaluated in the VBench-long suite with 30-seconds inference setting.

| Hybrid Attention | | Relative RoPE | Reg Loss | Decoupled Distillation | Throughput (FPS) ↑ | Evaluation scores ↑ | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Sparse | Linear | | | | | Total | Quality | Semantic | Dynamic | Consis. |
| ✗ | ✗ | ✗ | ✗ | ✗ | 23.1 | 80.25 | 81.65 | 74.64 | 53 | 25.70 |
| ✓ | ✗ | ✗ | ✗ | ✗ | 30.4 | 80.23 | 81.60 | 74.77 | 55 | 25.78 |
| ✗ | ✓ | ✗ | ✗ | ✗ | 22.3 | 81.43 | 82.58 | 76.85 | 66 | 25.65 |
| ✓ | ✓ | ✗ | ✗ | ✗ | 29.5 | 81.56 | 82.95 | 76.01 | 67 | 25.62 |
| ✓ | ✓ | ✓ | ✗ | ✗ | 29.5 | 81.85 | 83.11 | 76.82 | 64 | 25.97 |
| ✓ | ✓ | ✓ | ✓ | ✗ | 29.5 | 82.38 | 83.87 | 76.41 | 65 | 25.68 |
| ✓ | ✓ | ✓ | ✓ | ✓ | 29.5 | **83.49** | **84.67** | **78.78** | **74** | **26.02** |

### 4.3 Ablation Study

We conduct ablation studies to disentangle the structural and optimization contributions of Hybrid Forcing. In particular, we examine how the proposed hybrid attention interacts with the decoupled distillation strategy, relative temporal RoPE, and auxiliary regularization. Results are summarized in Table 3. All variants are distilled for 2000 steps on the same dataset, with a fixed local window size of 9 and 3 sink frames retained in the KV cache .

**Hybrid Attention and Decoupled Distillation.** Introducing hybrid attention improves both efficiency and long-horizon generation quality over the native short-window baseline, increasing throughput from 23.1 to 29.5 FPS and the total score from 80.25 to 81.56. The component-wise ablations further clarify the role of each branch: sparse local attention alone mainly improves efficiency, achieving the highest throughput of 30.4 FPS without noticeable quality degradation, while the linear history branch is primarily responsible for long-range temporal modeling, improving the Dynamic score from 53 to 66 even when paired with full local attention. Combining both branches achieves the best speed–quality trade-off, validating our decomposition of streaming attention into efficient local refinement and compact long-term memory.

Specially, the architectural benefit can be fully unlocked when paired with the proposed decoupled distillation strategy. Incorporating decoupled training further boosts the total score to 83.49, with consistent improvements across all metrics, particularly Dynamic (74) and Semantic (78.78). By explicitly separating semantic representation learning from structured historical modeling, the first DMD-only phase consolidates forms information-rich token representations under dense attention. Then subsequent distillation phase can focus on learning structured long-range history compression and sparsification over already coherent tokens. This separation of objectives prevents premature compression of noisy historical states and leads to more stable optimization.




Figure 4: Visual results with and without the regularization loss. Removing the regularization loss can lead to persistent camera drift and character duplication.

Table 4: **Causal interventions on the linear history state.** Segment scores are disjoint 10-second Total scores, and *Decay* denotes the first-to-last segment difference. Evaluated on VBench-Long under the 30-second setting.

| Intervention | Evaluation scores ↑ | | | Segment Total ↑ | | | Decay |
|---|---|---|---|---|---|---|---|
| | Total | Quality | Semantic | 0–10s | 10–20s | 20–30s | |
| Normal (full history) | **83.39** | **84.52** | **78.88** | **83.93** | **83.54** | **82.93** | **−1.00** |
| Frozen@10s (stale) | 83.27 | 84.46 | 78.51 | 83.89 | 83.36 | 82.60 | −1.29 |
| Reset@15s (rebuilt) | 83.05 | 84.25 | 78.26 | 83.87 | 82.94 | 82.57 | −1.30 |
| Zero (no history) | 82.73 | 83.89 | 78.11 | 83.79 | 82.77 | 82.15 | −1.64 |
| Shuffle (wrong history) | 82.05 | 83.25 | 77.26 | 83.87 | 82.14 | 81.77 | −2.10 |

**Stability Enhancements.** The relative temporal RoPE constraint and auxiliary regularization further strengthen robustness in extended streaming scenarios. Constraining temporal RoPE extrapolation mitigates positional distribution shift during long inference, yielding consistent quality gains. Meanwhile, regularization loss stabilizes motion dynamics and reduces long-horizon drift, as illustrated in fig. 4. Without this supervision, models exhibit gradual camera bias and occasional mode collapse.

**The Linear History State Provides Effective Long-Range Memory.** To verify that the trained model actively uses the linear history state during inference, we causally intervene on this state while keeping the local KV cache, sink frames, prompts, and initial noise fixed. As shown in table 4, the intact state achieves the best overall and final-segment performance, while progressively removing or corrupting its historical information consistently degrades the results: *Normal* (83.39) outperforms *Frozen@10s* (83.27), *Reset@15s* (83.05), and *Zero* (82.73), whereas injecting mismatched history with *Shuffle* performs worst (82.05). The degradation becomes more pronounced in later segments, after earlier content has left the local attention window, and the fact that an incorrect state is more harmful than no state further shows that the model relies on its history content rather than treating it as a generic bias. These results provide direct evidence that the linear state serves as an effective long-range memory and positively contributes to inference.

## 5 Conclusion

We presented Hybrid Forcing, a unified architecture–optimization framework for efficient long-horizon streaming video generation. By integrating linear temporal attention for persistent global history aggregation with block-sparse sliding-window attention for efficient local refinement, our method systematically reallocates attention toward informative dependencies across time. Crucially, we further introduce a decoupled distillation strategy to fully unlock the potential of the hybrid attention mechanism, ensuring stable and effective long-horizon modeling. In addition, to mitigate temporal distribution shift and long-horizon mode collapse, we incorporate a consistent relative RoPE constraint and teacher-guided regularization for enhanced stability. Hybrid Forcing achieves state-of-the-art performance on both short- and long-form benchmarks and enables real-time, unbounded video generation at 29.5 FPS on a single H100 GPU.

**Limitations**. Similar to prior streaming distillation approaches, our framework relies on the ODE model provided by Self-forcing (Huang et al., 2025). Consequently, the student's performance is inherently bounded by the provided model and the teacher's generation quality. In addition, our throughput evaluation is conducted on high-performance GPUs, where the real-time performance may not be observed on low-end GPUs. We anticipate that integrating quantization and model compression techniques could narrow this gap, and we are actively exploring strategies to further reduce inference latency for broader deployment.

### AI USE STATEMENT

Generative AI tools were used solely to improve the readability of author-written text, including grammar correction, wording refinement, and language polishing. They were not used to generate research ideas or hypotheses, develop theoretical or conceptual frameworks, formulate mathematical claims or proofs, design or implement methods or experiments, process datasets, or analyze or interpret experimental results; other tasks requiring disclosure are not applicable to this work. All AI-assisted edits were carefully reviewed by the authors to ensure that they preserved the intended technical meaning and introduced no unsupported claims, results, or citations. The authors take full responsibility for the final content of this work, including all text, claims, and artifacts produced with the assistance of generative AI.

### ETHICS STATEMENT

This work focuses on methodological research in long-horizon video generation and evaluates generated content solely in a research setting. It does not target individuals or demographic groups and is not intended for high-stakes decision-making, surveillance, impersonation, or other harmful applications. The user study is limited to subjective comparisons of generated video quality and does not involve sensitive personal information or consequential decisions. Nevertheless, as with other video generation technologies, the proposed method could potentially be misused to produce misleading or inappropriate content. We therefore encourage its responsible use for research purposes in accordance with applicable safeguards and regulations. Beyond these general risks associated with generative video models, we are not aware of additional ethical concerns specific to this work.

### REPRODUCIBILITY STATEMENT

We provide the training budget, implementation settings, optimization hyperparameters, and other experimental details required to reproduce our results in Sec. 4.1 and Appendix A. Our evaluation follows the established VBench and VBench-Long protocols, using their standard test prompt suites and evaluation metrics. We also include the **anonymous source code** used in our experiments as part of the supplementary material. Additional algorithmic descriptions, experimental configurations, and qualitative evaluation details are provided in the main paper and supplementary material to facilitate independent reproduction and verification of our results.

## REFERENCES

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Jake Bruce, Michael D Dennis, Ashley Edwards, Jack Parker-Holder, Yuge Shi, Edward Hughes, Matthew Lai, Aditi Mavalankar, Richie Steigerwald, Chris Apps, et al. Genie: Generative interactive environments. In *Forty-first International Conference on Machine Learning*, 2024.

Boyuan Chen, Diego Martí Monsó, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *Advances in Neural Information Processing Systems*, 37:24081–24125, 2024.

Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengcheng Ma, et al. Skyreels-v2: Infinite-length film generative model. *arXiv preprint arXiv:2504.13074*, 2025a.

Junsong Chen, Yuyang Zhao, Jincheng Yu, Ruihang Chu, Junyu Chen, Shuai Yang, Xianbang Wang, Yicheng Pan, Daquan Zhou, Huan Ling, et al. Sana-video: Efficient video generation with block linear diffusion transformer. *arXiv preprint arXiv:2509.24695*, 2025b.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Self-forcing++: Towards minute-scale high-quality video generation. *arXiv preprint arXiv:2510.02283*, 2025.

Karan Dalal, Daniel Koceja, Jiarui Xu, Yue Zhao, Shihao Han, Ka Chun Cheung, Jan Kautz, Yejin Choi, Yu Sun, and Xiaolong Wang. One-minute video generation with test-time training. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17702–17711, 2025.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in neural information processing systems*, 35: 16344–16359, 2022.

Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. *arXiv preprint arXiv:2412.14169*, 2024.

Juechu Dong, Boyuan Feng, Driss Guessous, Yanbo Liang, and Horace He. Flex attention: A programming model for generating optimized attention kernels. *arXiv preprint arXiv:2412.05496*, 2(3):4, 2024.

Yu Gao, Haoyuan Guo, Tuyen Hoang, Weilin Huang, Lu Jiang, Fangyuan Kong, Huixia Li, Jiashi Li, Liang Li, Xiaojie Li, et al. Seedance 1.0: Exploring the boundaries of video generation models. *arXiv preprint arXiv:2506.09113*, 2025.

Google. Veo: a text-to-video generation system, 2025. URL `https://storage.googleapis.com/deepmind-media/veo/Veo-3-Tech-Report.pdf`.

Roberto Henschel, Levon Khachatryan, Hayk Poghosyan, Daniil Hayrapetyan, Vahram Tadevosyan, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Streamingt2v: Consistent, dynamic, and extendable long video generation from text. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 2568–2577, 2025.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.

Sihui Ji, Xi Chen, Shuai Yang, Xin Tao, Pengfei Wan, and Hengshuang Zhao. Memflow: Flowing adaptive memory for consistent and efficient long video narratives. *arXiv preprint arXiv:2512.14699*, 2025.

Yang Jin, Zhicheng Sun, Ningyuan Li, 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.

Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022.

Kling. Kling ai: Next-generation ai creative studio. `https://klingai.com/cn/`, 2026.

Akio Kodaira, Tingbo Hou, Ji Hou, Markos Georgopoulos, Felix Juefei-Xu, Masayoshi Tomizuka, and Yue Zhao. Streamdit: Real-time streaming text-to-video generation. *arXiv preprint arXiv:2507.03745*, 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, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Junkun Yuan, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yanxin Long, 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*, 2024.

Ruibin Li, Tao Yang, Yangming Shi, Yuqi Zhang, Qide Dong, Haoran Cheng, Weiguo Feng, Shilei Wen, Bingyue Peng, and Lei Zhang. Many-for-many: Unify the training of multiple video and image generation and manipulation tasks. *arXiv preprint arXiv:2506.01758*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. *arXiv preprint arXiv:2509.25161*, 2025.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021.

ltx2. ltx2. `https://ltx.io/model/ltx-2`, 2026.

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. *arXiv preprint arXiv:2410.11081*, 2024.

Yunhong Lu, Yanhong Zeng, Haobo Li, Hao Ouyang, Qiuyu Wang, Ka Leong Cheng, Jiapeng Zhu, Hengyuan Cao, Zhipeng Zhang, Xing Zhu, et al. Reward forcing: Efficient streaming video generation with rewarded distribution matching distillation. *arXiv preprint arXiv:2512.04678*, 2025.

OpenAI. Video generation models as world simulators, 2024. URL `https://openai.com/sora`.

Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, Yuhui Wang, Anbang Ye, Gang Ren, Qianran Ma, Wanying Liang, Xiang Lian, Xiwen Wu, Yuting Zhong, Zhuangyan Li, Chaoyu Gong, Guojun Lei, Leijun Cheng, Limin Zhang, Minghao Li, Ruijie Zhang, Silan Hu, Shijie Huang, Xiaokang Wang, Yuanheng Zhao, Yuqi Wang, Ziang Wei, and Yang You. Open-sora 2.0: Training a commercial-level video generation model in 200k. *arXiv preprint arXiv:2503.09642*, 2025.

Pytorch. Triton. `https://pytorch.org/blog/triton-kernel-compilation-stages`, 2026.

Zihan Qiu, Zekun Wang, Bo Zheng, Zeyu Huang, Kaiyue Wen, Songlin Yang, Rui Men, Le Yu, Fei Huang, Suozhi Huang, et al. Gated attention for large language models: Non-linearity, sparsity, and attention-sink-free. *arXiv preprint arXiv:2505.06708*, 2025.

RunwayML. Gen-3 by runway, 2023. URL `https://research.runwayml.com/gen3`.

Sand-AI. Autoregressive video generation at scale, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020a.

Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020b.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Kimi Team, Yu Zhang, Zongyu Lin, Xingcheng Yao, Jiaxi Hu, Fanqing Meng, Chengyin Liu, Xin Men, Songlin Yang, Zhiyuan Li, et al. Kimi linear: An expressive, efficient attention architecture. *arXiv preprint arXiv:2510.26692*, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Wenhao Wang and Yi Yang. Vidprom: A million-scale real prompt-gallery dataset for text-to-video diffusion models. *Advances in Neural Information Processing Systems*, 37:65618–65642, 2024.

Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, et al. Lavie: High-quality video generation with cascaded latent diffusion models. *International Journal of Computer Vision*, 133(5):3059–3078, 2025.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in neural information processing systems*, 36:8406–8441, 2023.

Tianhe Wu, Ruibin Li, Lei Zhang, and Kede Ma. Diversity-preserved distribution matching distillation for fast visual synthesis. *arXiv preprint arXiv:2602.03139*, 2026.

Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, et al. Sparse videogen: Accelerating video diffusion transformers with spatial-temporal sparsity. *arXiv preprint arXiv:2502.01776*, 2025.

Yifei Xia, Suhan Ling, Fangcheng Fu, Yujie Wang, Huixia Li, Xuefeng Xiao, and Bin Cui. Training-free and adaptive sparse attention for efficient long video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 15982–15993, 2025.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. *arXiv preprint arXiv:2309.17453*, 2023.

Shuai Yang, Wei Huang, Ruihang Chu, Yicheng Xiao, Yuyang Zhao, Xianbang Wang, Muyang Li, Enze Xie, Yingcong Chen, Yao Lu, et al. Longlive: Real-time interactive long video generation. *arXiv preprint arXiv:2509.22622*, 2025.

Hidir Yesiltepe, Tuna Han Salih Meral, Adil Kaan Akan, Kaan Oktay, and Pinar Yanardag. Infinity-rope: Action-controllable infinite video generation emerges from autoregressive self-rollout. *arXiv preprint arXiv:2511.20649*, 2025.

Shengming Yin, Chenfei Wu, Huan Yang, Jianfeng Wang, Xiaodong Wang, Minheng Ni, Zhengyuan Yang, Linjie Li, Shuguang Liu, Fan Yang, et al. Nuwa-xl: Diffusion over diffusion for extremely long video generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1309–1320, 2023.

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*, pp. 6613–6623, 2024.

Tianwei Yin, Qiang Zhang, Richard Zhang, William T Freeman, Fredo Durand, Eli Shechtman, and Xun Huang. From slow bidirectional to fast autoregressive video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22963–22974, 2025.

Jingyang Yuan, Huazuo Gao, Damai Dai, Junyu Luo, Liang Zhao, Zhengyan Zhang, Zhenda Xie, Yuxing Wei, Lean Wang, Zhiping Xiao, et al. Native sparse attention: Hardware-aligned and natively trainable sparse attention. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 23078–23097, 2025.

Jintao Zhang, Haoxu Wang, Kai Jiang, Shuo Yang, Kaiwen Zheng, Haocheng Xi, Ziteng Wang, Hongzhou Zhu, Min Zhao, Ion Stoica, et al. Sla: Beyond sparsity in diffusion transformers via fine-tunable sparse-linear attention. *arXiv preprint arXiv:2509.24006*, 2025a.

Lvmin Zhang and Maneesh Agrawala. Packing input frame contexts in next-frame prediction models for video generation. *Arxiv*, 2025.

Peiyuan Zhang, Yongqi Chen, Haofeng Huang, Will Lin, Zhengzhong Liu, Ion Stoica, Eric Xing, and Hao Zhang. Vsa: Faster video diffusion with trainable sparse attention. *arXiv preprint arXiv:2505.13389*, 2025b.

Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhengzhong Liu, and Hao Zhang. Fast video generation with sliding tile attention. *arXiv preprint arXiv:2502.04507*, 2025c.

Yifu Zhang, Hao Yang, Yuqi Zhang, Yifei Hu, Fengda Zhu, Chuang Lin, Xiaofeng Mei, Yi Jiang, Zehuan Yuan, and Bingyue Peng. Waver: Wave your way to lifelike video generation. *arXiv preprint arXiv:2508.15761*, 2025d.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. Unipc: A unified predictor-corrector framework for fast sampling of diffusion models. *Advances in Neural Information Processing Systems*, 36:49842–49869, 2023.

Kaiwen Zheng, Yuji Wang, Qianli Ma, Huayu Chen, Jintao Zhang, Yogesh Balaji, Jianfei Chen, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. Large scale diffusion distillation via score-regularized continuous-time consistency. *arXiv preprint arXiv:2510.08431*, 2025.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*, 2026.

# APPENDIX

In this appendix, we provide the following materials:

- **A.** A complete description of our algorithm (see Sec. 3.5 in the main paper);
- **B.** The template used for enhancing VBench prompts (see Sec. 4.1 in the main paper);
- **C.** Runtime comparison between dense self attention block vs our hybrid self attention block;
- **D.** User study of long video generation result;
- **E.** Detailed evaluation results on the short VBench benchmark and VBench-Long Benchmark (see Sec. 4.2 in the main paper);
- **F.** More qualitative comparisons under the 5-second and 30-second inference setting (see Sec. 4.2 in the main paper);
- **G.** Cases of 10-minute videos (see Sec. 4.2 in the main paper).

We compile all visual comparison videos into a single MP4 file named comparison_video.mp4 in the supplementary material. In addition, we provide a any complete video with all clips generated by our hybrid forcing model. For better viewing experience, we uploaded the video demo and comparison videos to a dedicated anonymous website https://anonymous.4open.science/w/hybrid_page-BD98/, where the videos can be played directly in the browser. Due to the size constraints of the supplementary materials and website, the videos are compressed at a lower bitrate. Note that, due to the significant number of video files included in our demonstrations, initial page loading may require several minutes to complete. We appreciate your patience during this process, as the complete visual experience is essential to understand the capabilities and performance of our approach.

## A ALGORITHM DETAILS

The details of training, inference, and hybrid attention are shown in algorithm 1, algorithm 2 and algorithm 3, respectively.

## B TEMPLATE FOR VBENCH PROMPTS ENHANCEMENTS

Following previous works (Huang et al., 2025; Yang et al., 2025; Ji et al., 2025), we enhance the text prompts in VBench using **Qwen/Qwen2.5-7B-Instruct** (Bai et al., 2023) together with the official prompt enhancement template. The prompt used to guide the enhancement is shown as follows:

Prompt Enhancement Instruction

```
You are a prompt engineer, aiming to rewrite user inputs into high-
    quality prompts for better video generation without affecting
    the original meaning.
Task requirements:
1. For overly concise user inputs, reasonably infer and add details
    ...
2. Enhance the main features in user descriptions...
3. Output the entire prompt in English...
4. Prompts should match the user's intent...
5. Emphasize motion information...
6. Your output should have natural motion attributes...
7. The revised prompt should be around 80-100 words long.
Revised prompt examples:
1. Japanese-style fresh film photography, ...
2. Anime thick-coated illustration, ...
3. CG game concept digital art, ...
4. American TV series poster style, ...
I will now provide the prompt for you to rewrite...
```

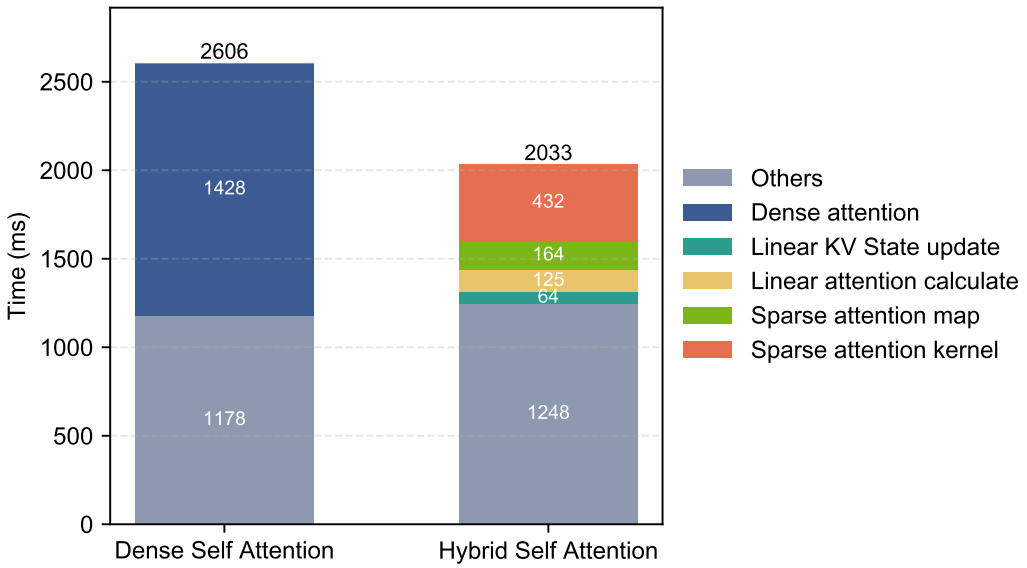


Figure 5: Runtime breakdown of dense self-attention and the proposed hybrid self-attention.

## C RUNTIME COMPARISION

As illustrated in fig. 5, Dense Self Attention takes 2606 ms in total, where the dense attention computation itself contributes 1428 ms, about 54.8% of the total runtime. In contrast, Hybrid Self Attention reduces the total runtime to 2033 ms, giving about 22.0% overall latency reduction and roughly 1.28x speedup. Within the hybrid design, the sparse attention kernel costs 432 ms, while Sparse map calculation costs 164 ms; together, the sparse attention part accounts for about 29.3% of the hybrid runtime. The linear-attention related cost, including KV state update and linear attention calculation, is 189 ms in total, about 9.3%. Compared with the original dense attention computation alone, the hybrid attention-specific computation drops from 1428 ms to 785 ms, showing that the proposed hybrid attention substantially reduces the core attention cost. Note that the "Others" category includes several auxiliary operations, including the Q/K/V linear projections, Q/K normalization, query RoPE application, KV cache update, and final output projection.

## D USER STUDIES

To further evaluate the perceptual quality of our method in long-duration video generation, we conducted a user study comparing Hybrid Forcing with six baseline methods: CausVid, Self Forcing, Rolling Forcing, Long Live, MemFlow, and InfRoPE. We recruited 10 participants most of whom are researchers in computer vision and selected 20 prompt groups for evaluation. For each prompt, participants were shown seven videos generated by different methods under the same resolution and duration setting. The videos were randomly shuffled and anonymized before presentation to avoid method-specific bias. Participants were asked to rank the videos in each group according to overall visual preference, considering temporal consistency, motion quality, visual fidelity, and semantic alignment. This resulted in 200 complete rankings and 1,400 method-level judgments in total.

The user study results are consistent with the automatic evaluation in Table 2. Hybrid Forcing achieves the best average rank of 2.29 and receives the highest Top-1 preference rate of 36.2%, substantially outperforming the strongest baselines, Rolling Forcing. In particular, Hybrid Forcing is ranked within the top three in 75.2% of all comparisons, suggesting that users consistently prefer its long-range temporal consistency, motion quality, and visual fidelity. Rolling Forcing remains competitive among baselines, which aligns with its automatic evaluation results, while methods with weaker consistency scores receive lower human preference rankings. Overall, the user study indicates that the improvements of Hybrid Forcing are perceptually meaningful and translate into stronger human preference in long video generation.

Table 5: **User study results.** Participants ranked videos generated by different methods for each prompt. Avg. Rank is lower better, while Top-1 Rate and Top-3 Rate are higher better.

| Model | Avg. Rank ↓ | Top-1 Rate ↑ | Top-3 Rate ↑ |
|---|---|---|---|
| CausVid | 5.61 | 4.5% | 18.8% |
| Self Forcing | 4.45 | 9.8% | 34.5% |
| Rolling Forcing | <u>3.18</u> | <u>18.8%</u> | <u>62.0%</u> |
| Long Live | 5.18 | 6.2% | 24.0% |
| MemFlow | 4.77 | 7.5% | 29.8% |
| InfRoPE | 3.52 | 17.0% | 55.8% |
| Hybrid Forcing | **2.29** | **36.2%** | **75.2%** |

**Algorithm 1** Hybrid Forcing Training

**Require:** Denoise timesteps $\{\mathbf{t}\} = \{t_1, \ldots, t_T\}$
**Require:** Number of video chunks $N$
**Require:** AR diffusion model $g_\theta$, Teacher diffusion model $g_\phi$
**Require:** KV cache $\mathbf{KV}$ and Linear State $\mathbf{LN}$

**loop**
  Initialize model output $\mathbf{X}_\theta \leftarrow []$
  Initialize noise input $\mathbf{X}_T \leftarrow []$
  Initialize KV cache $\mathbf{KV} \leftarrow []$
  Initialize Linear State cache $\mathbf{LN} \leftarrow []$
  Uniformly sample $s \sim len(\{\mathbf{t}\})$
  **for** $i = 1, \ldots, N$ **do**
    Initialize $\boldsymbol{x}^i_{t_T} \sim \mathcal{N}(0, I)$
    $\mathbf{X}_T$ **.** `append`$(\boldsymbol{x}^i_{t_T})$
    **for** $j = T, \ldots, s$ **do**
      **if** $t_j = t_s$ **then**
        Enable gradient computation
        Set $\hat{\boldsymbol{x}}^i_0 \leftarrow g_\theta(\boldsymbol{x}^i_{t_j}; t_j, \mathbf{KV}, \mathbf{LN})$ ▷ Hybrid attention
        $\mathbf{X}_\theta$ **.** `append`$(\hat{\boldsymbol{x}}^i_0)$
        Disable gradient computation
        Cache $\mathbf{kv}^i \leftarrow g^{\mathrm{KV}}_\theta(\hat{\boldsymbol{x}}^i_0; 0, \mathbf{KV}, \mathbf{LN})$ ▷ Hybrid attention
        **if** $\mathbf{KV}$ is full **then**
          $\mathbf{kv}^{evited} \leftarrow \mathbf{KV}.\mathrm{pop}(0)$
          Update Linear State $\mathbf{LN} \leftarrow \mathbf{kv}^{evited}$ ▷ Eq. (3) in the main paper
        **end if**
        $\mathbf{KV}$ **.** `append`$(\mathbf{kv}^i)$
      **else**
        Disable gradient computation
        Set $\hat{\boldsymbol{x}}^i_0 \leftarrow g_\theta(\boldsymbol{x}^i_{t_j}; t_j, \mathbf{KV}, \mathbf{LN})$
        Sample $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$
        Set $\boldsymbol{x}^i_{t_{j-1}} \leftarrow \alpha_{t_{j-1}} \hat{\boldsymbol{x}}^i_0 + \beta_{t_{j-1}} \boldsymbol{\epsilon}$
      **end if**
    **end for**
  **end for**
  **if** $s = T$ **then**
    $\mathbf{X}_\phi \leftarrow$ `Rollout`$(\mathbf{X}_T, g_\phi, T)$
    $\mathcal{L}_{distill} = \mathcal{L}_{DMD} + \lambda$`MSE`$(\mathbf{X}_\phi, \mathbf{X}_\theta)$
  **else**
    $\mathcal{L}_{distill} = \mathcal{L}_{DMD}$
  **end if**
  Update $\theta$ via $\mathcal{L}_{distill}$
**end loop**

**Algorithm 2** Autoregressive Diffusion Inference with Rolling KV Cache

**Require:** Denoise timesteps $\{t_1, \ldots, t_T\}$
**Require:** Number of generated chunks $M$
**Require:** AR diffusion model $g_\theta$
Initialize model output $\mathbf{X}_\theta \leftarrow []$
Initialize KV cache $\mathbf{KV} \leftarrow []$
Initialize Linear State cache $\mathbf{LN} \leftarrow []$
**for** $i = 1, \ldots, M$ **do**
  Initialize $\boldsymbol{x}^i_{t_T} \sim \mathcal{N}(0, I)$
  **for** $j = T, \ldots, 1$ **do**
    Set $\hat{\boldsymbol{x}}^i_0 \leftarrow g_\theta(\boldsymbol{x}^i_{t_j}; t_j, \mathbf{KV}, \mathbf{LN})$
    **if** $j = 1$ **then**
      $\mathbf{X}_\theta$ . `append`$(\hat{\boldsymbol{x}}^i_0)$
      Cache $\mathbf{kv}^i \leftarrow g^{\text{KV}}_\theta(\hat{\boldsymbol{x}}^i_0; 0, \mathbf{KV}, \mathbf{LN})$
      **if** $\mathbf{KV}$ is full **then**
        $\mathbf{kv}^{evited} \leftarrow \mathbf{KV}.\text{pop}(0)$ ▷ Cache eviction
        Update Linear State $\mathbf{LN} \leftarrow \mathbf{kv}^{evited}$ ▷ Eq. (3) in the main paper
      **end if**
      $\mathbf{KV}$ . `append`$(\mathbf{kv}^i)$
    **else**
      Sample $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$
      Set $\boldsymbol{x}^i_{t_{j-1}} \leftarrow \alpha_{t_{j-1}} \hat{\boldsymbol{x}}^i_0 + \beta_{t_{j-1}} \boldsymbol{\epsilon}$
    **end if**
  **end for**
**end for**
**return** $\mathbf{X}_\theta$

**Algorithm 3** Hybrid Attention

**Require:** Matrices $Q, K, V \in \mathbb{R}^{N \times d}$.
**Require:** Linear State Matrics $L \in \mathbb{R}^{d \times d}, H \in \mathbb{R}^{1 \times d}$.
**Require:** Block sizes $b_q, b_{kv}$, hyper-parameters sparse Top $K$.
// Sparse Attention Part
Divide $Q$ to $T_m = N/b_q$ blocks $\{\mathbf{Q}_i\}$ ;
Divide $K, V$ to $T_n = N/b_{kv}$ blocks $\{\mathbf{K}_i\}, \{\mathbf{V}_i\}$ ;
$P_c = \text{Softmax}(\text{pool}(Q)\text{pool}(K)^\top/\sqrt{d})$ ;  Initialize $M_c = 0$ ;
$M_c[i, j] = 1$ if $P_c[i, j] \in \texttt{TopK}(P_c[i, :], K)$ ;
**for** $i = 1$ **to** $T_m$ **do**
  **for** $j = 1$ **to** $T_n$ **do**
    **if** $M_c[i, j] = 1$ **then**
      $\mathbf{S}_{ij} = \mathbf{Q}_i \mathbf{K}_j^\top / \sqrt{d}$ ;
      $m_{ij} = \max(m_{i,j-1}, \text{rowmax}(\mathbf{S}_{ij}))$ ;
      $\mathbf{P}_{ij} = \exp(\mathbf{S}_{ij} - m_{ij})$ ;
      $l_{ij} = e^{m_{i,j-1} - m_{ij}} l_{i,j-1} + \text{rowsum}(\mathbf{P}_{ij})$ ;
      $\mathbf{O}^s_{ij} = \text{diag}(e^{m_{i,j-1} - m_{ij}})\mathbf{O}^s_{i,j-1} + \mathbf{P}_{ij}\mathbf{V}_j$ ;
    **end if**
  **end for**
  $\mathbf{O}^s_i = \text{diag}(l_i^{T_n})^{-1}\mathbf{O}^s_{i,T_n}$ ;
  $O^{local} = \{\mathbf{O}^s_i\}$
**end for**
// History Linear Attention Part
$O^{\text{history}} = \text{Linear}\left(\frac{\text{RoPE}(\phi(Q))L}{\phi(Q)H}\right)$
**return** $O = O^{local} + O^{hitory}$ ;

## E Detailed evaluation results on the short VBench benchmark and VBench-Long Benchmark

In fig. 6, we present the detailed evaluation results on VBench and VBench-Long for representative few-step autoregressive models across all 16 VBench metrics. Overall, our hybrid forcing method

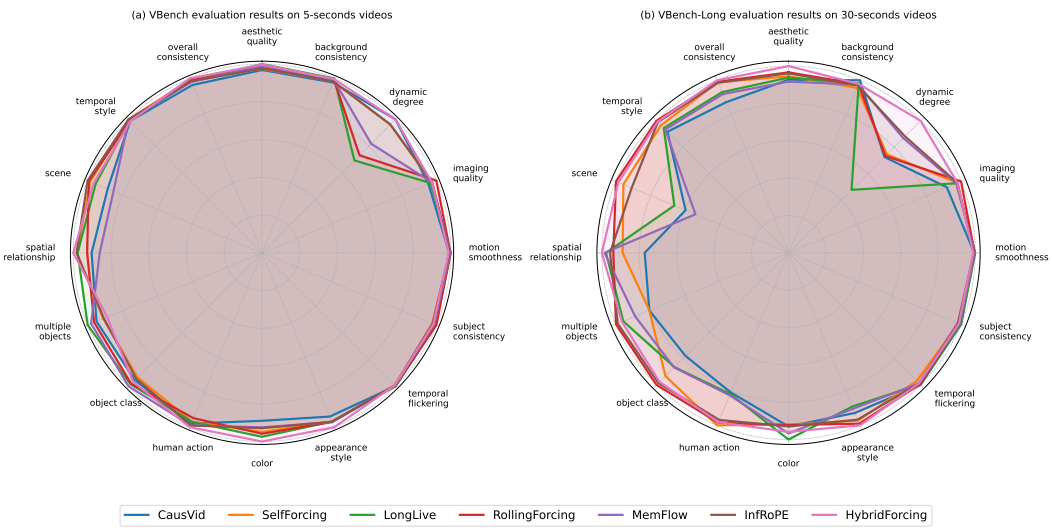


Figure 6: (a) VBench evaluation results over all 16 metrics on 5-seconds videos. (b) VBench-long evaluation results over all 16 metrics on 30-seconds videos. Values are normalized by the max value for better visualization effect.

consistently outperforms its competitors, with particularly notable improvements in semantic-related metrics. Specifically, under the 5-second evaluation setting, our hybrid forcing achieves leading performance on several metrics, including dynamic degree, appearance style, color, spatial relationship, and overall consistency, while maintaining comparable results on the remaining metrics. Under the more challenging 30-second generation setting, hybrid forcing demonstrates even clearer advantages over existing autoregressive methods. Our method ranks first or second across most evaluation metrics, with particularly pronounced improvements in aesthetic quality, dynamic degree, and spatial relationship consistency. These results demonstrate that Hybrid Forcing more effectively preserves semantic coherence, visual consistency, and natural temporal dynamics over extended generation horizons, especially when producing longer videos.

## F More qualitative comparisons under the 5-second and 30-second inference setting

In fig. 7, fig. 8 and fig. 9, we present qualitative comparisons for 5-second video generation. As shown in the figures, *LongLive* and *MemFlow* exhibit limited motion dynamics, particularly in the bike-riding and motorcycle-riding cases. This issue becomes more apparent when inspecting the MP4 file in the 00:02–00:32 segment of comparison_video.mp4 in the supplementary materials, where the bicycle and the motorcycle nearly remain stationary throughout the sequence. In addition, the motion ranges of objects such as the stormtrooper and the shark evolve more slowly compared to other methods, indicating weaker temporal dynamics. In contrast, our hybrid forcing approach produces more natural and temporally coherent motion patterns while maintaining stronger visual fidelity, appearance consistency, and long-term temporal stability throughout the generated sequence. Please refer to the corresponding comparison_video.mp4 in our supplementary materials for complete qualitative video comparisons across different methods.

In fig. 10, fig. 11, and fig. 12, we provide qualitative comparisons for 30-second video generation. As illustrated in the figures, *CausVid*, *Self-Forcing*, and *InfRoPE* struggle to maintain stable generation over long durations, with noticeable visual degradation emerging after approximately 10 seconds. In extreme cases, severe structural artifacts appear; for example, in the turtle video, the turtle's tail gradually deforms and visually merges with the head in the middle of the sequence. While other methods demonstrate relatively stable performance, our hybrid forcing method achieves comparable or better visual quality while achieving the highest generation throughput. Specifically, our method achieves $\times 1.42$, $\times 1.57$, and $\times 1.86$ higher throughput than LongLive, MemFlow, and Rolling Forcing, respectively, demonstrating a favorable balance between long-horizon generation quality and

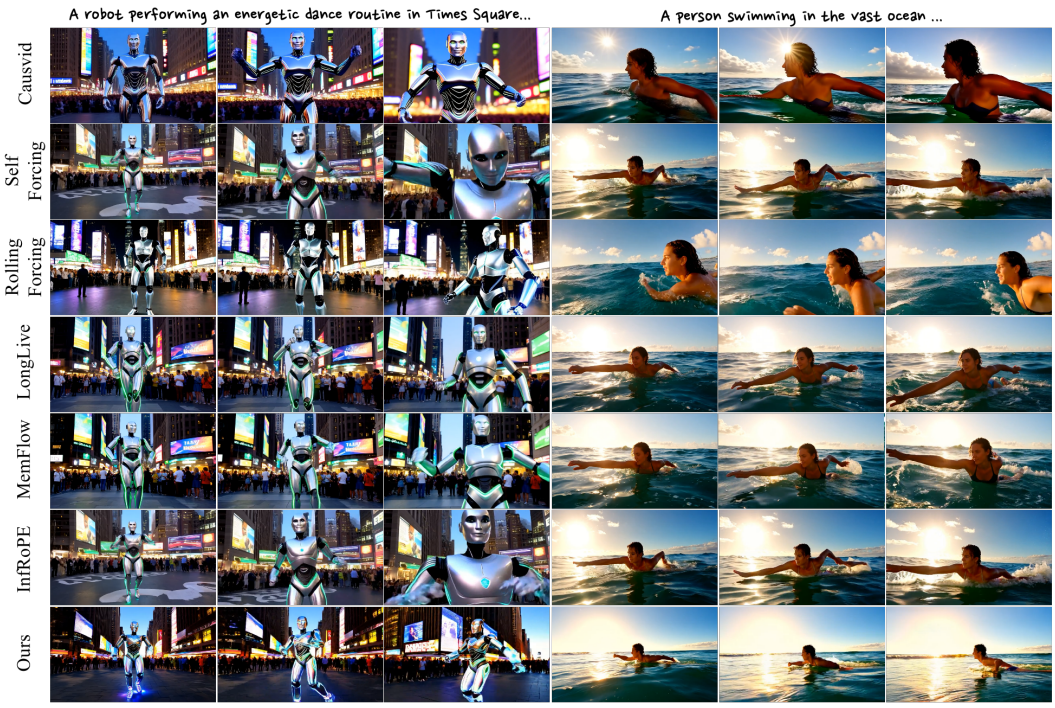


Figure 7: Qualitative comparison on 5-seconds video.

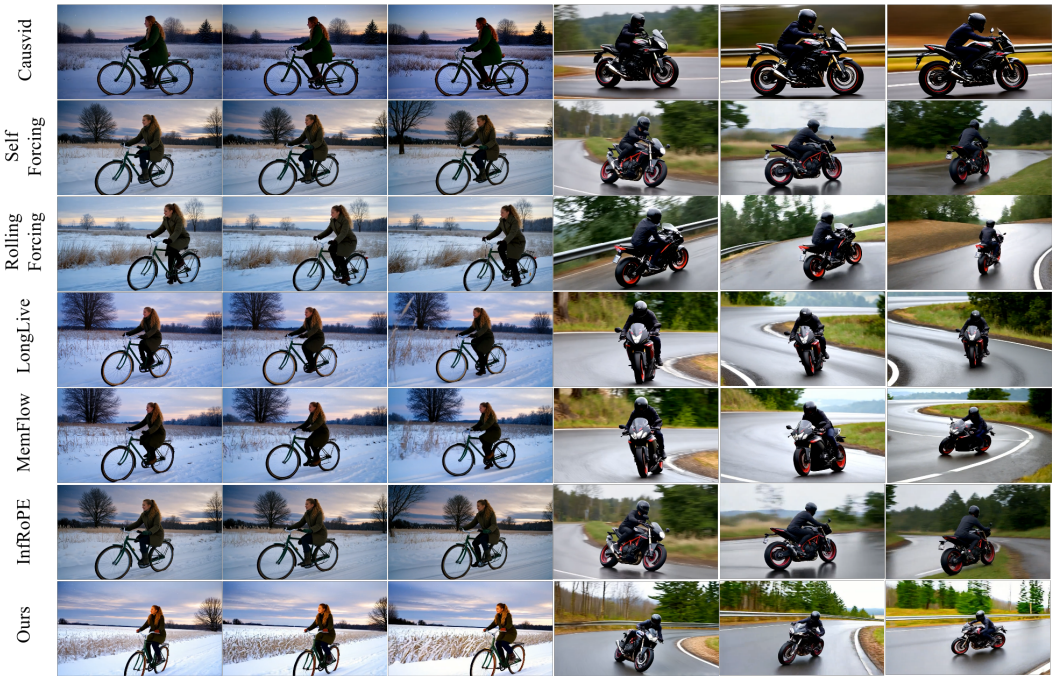


Figure 8: Qualitative comparison on 5-seconds video.

computational efficiency. Please refer to the 00:34–02:22 segment of comparison_video.mp4 in the supplementary materials for full video comparisons.

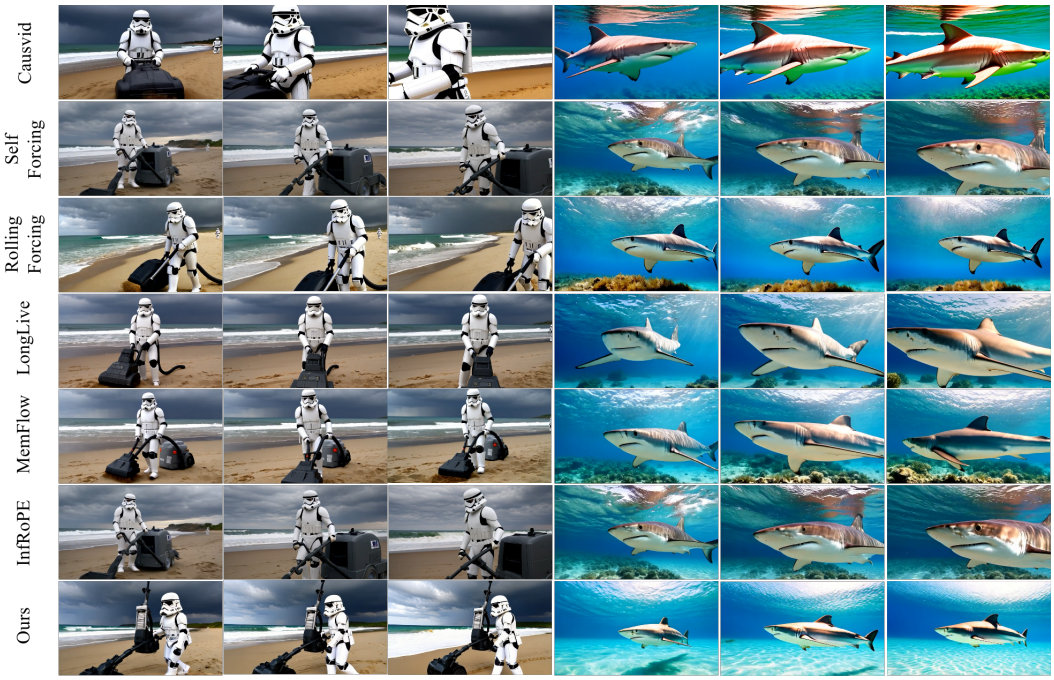


Figure 9: Qualitative comparison on 5-seconds video.

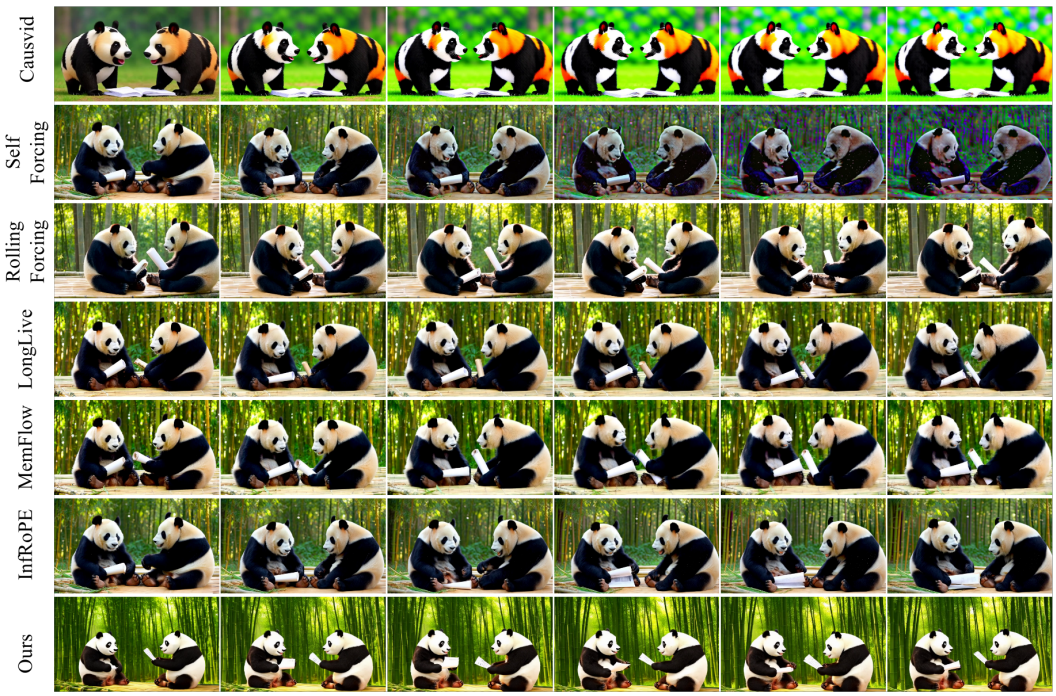


Figure 10: Qualitative comparison on 30-seconds video.

## G CASES OF 10-MINUTE VIDEOS

We present examples of single-prompt 10-minute video generation in fig. 13. As illustrated in the figure, both character identity and background appearance remain largely consistent throughout the entire sequence, even after 10 minutes of continuous generation. This demonstrates the



Figure 11: Qualitative comparison on 30-seconds video.




Figure 12: Qualitative comparison on 30-seconds video.

strong long-term temporal stability of our approach. Please refer to the 02:25–02:42 segment of comparison_video.mp4 in the supplementary materials for videos with 32× speed-up.

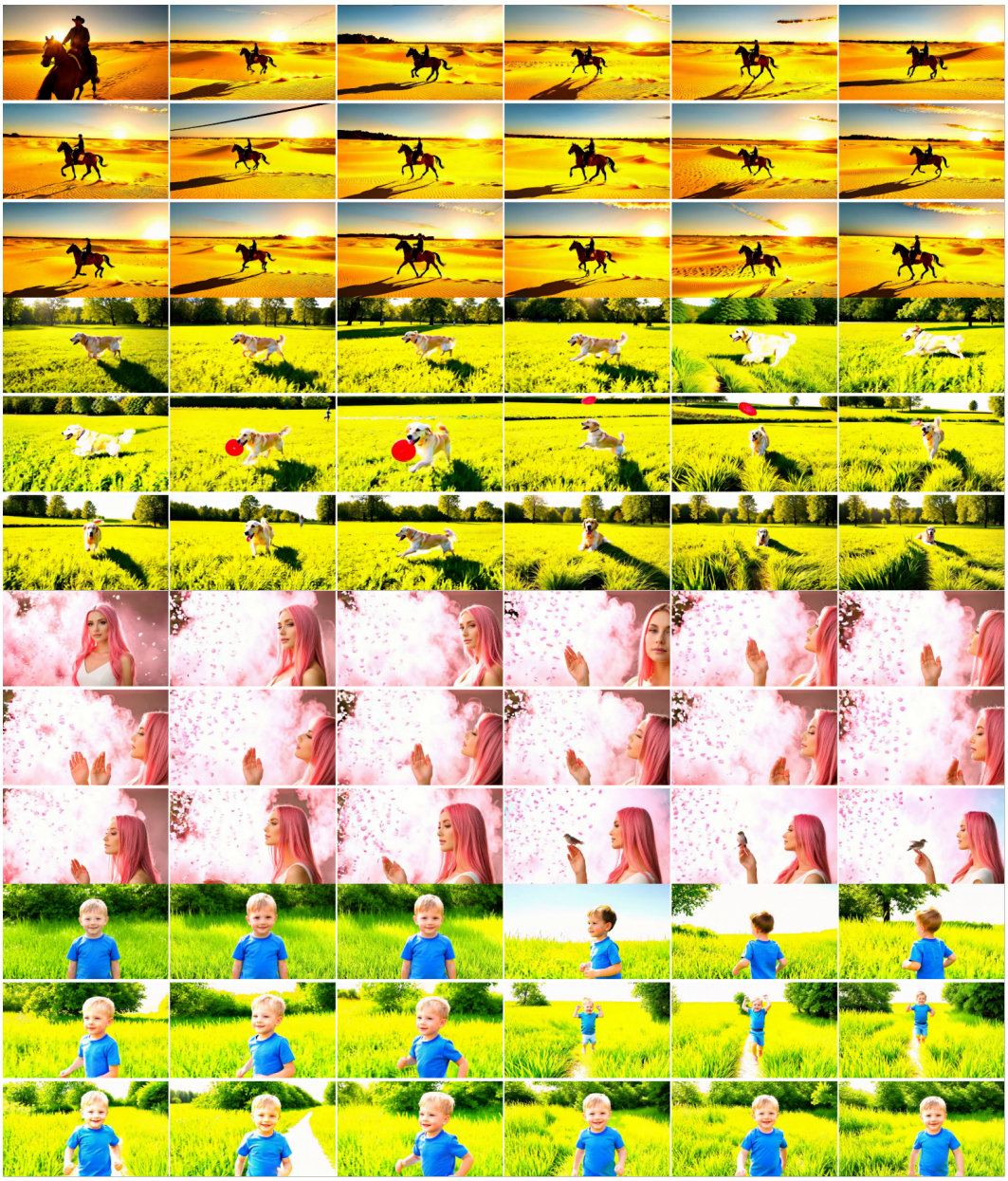

Figure 13: Visual result of 10 minutes video.

Combined with our high generation throughput of **29.5 FPS**, our proposed method is well suited for latency-sensitive application scenarios, including real-time livestreaming, virtual avatar systems, and interactive world models, where both long-duration consistency and high inference speed are essential.