OPSD-V: ON-POLICY SELF-DISTILLATION FOR POST-TRAINING FEW-STEP AUTOREGRESSIVE VIDEO GENERATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

We propose OPSD-V, an on-policy self-distillation paradigm for post-training few-step autoregressive (AR) video diffusion models. Existing few-step AR video generators, often obtained through DMD-style distillation from short-clip video teachers, achieve low latency but suffer from error accumulation and weakened motion dynamics during long rollouts. OPSD-V incorporates real long-video data into post-training, rather than relying on short-clip video teachers, to further reduce long-horizon degradation and improve motion dynamics while preserving the original few-step inference path. Specifically, we use the DMD-distilled few-step model for both the student and teacher branches. The student follows the exact inference-time rollout, generating each chunk conditioned on its own previously generated KV cache. The teacher is evaluated at the same student-visited denoising states, but uses a cleaner, AR-consistent temporal cache in which older history is replaced with real-video context. The teacher then provides dense trajectory-level supervision to the student. To maintain autoregressive consistency and prevent the teacher from becoming a fully teacher-forced oracle, both branches share an initial real-video prefix, and the teacher retains its most recent model-generated cache chunk. Experiments on Self-Forcing and LongLive demonstrate improved visual quality and motion dynamics, with higher VBench-based Quality scores. A user study further shows that OPSD-V is preferred over the base models in 66.0% of overall-preference judgments (82.5% excluding ties). Visual results, code, and model checkpoints are available through our anonymous project page, code repository, and Hugging Face repository, respectively.

1 INTRODUCTION

Video generation has rapidly evolved from short, offline clip synthesis to large-scale video foundation models capable of producing high-resolution, text-aligned, and temporally coherent videos. Modern video generation models are largely built upon diffusion transformers (DiTs) (Peebles & Xie, 2022), whose scalability has enabled substantial progress in visual fidelity, motion quality, prompt following, and long-video generation (Yang et al., 2024; Kong et al., 2024; Wan Team, 2025; Meituan LongCat Team et al., 2025; HaCohen et al., 2026; Brooks et al., 2024). Despite these advances, most large-scale video foundation models are still primarily designed for offline generation, where an entire video is synthesized after sampling rather than produced continuously during user interaction.

For real-time video generation with low viewing latency, recent methods combine autoregressive (AR) modeling and KV-cache reuse with few-step distillation (Yin et al., 2024b;a). CausVid distills a bidirectional video DiT into a causal AR student for fast streaming (Yin et al., 2025), while Self-Forcing aligns training with inference through rollouts on its own generated frames and rolling KV cache (Huang et al., 2025a). These techniques support streaming avatars (Yang et al., 2025a; Meituan LongCat Team et al., 2026; Wang et al., 2026), interactive media (Shin et al., 2025; Sand.ai et al., 2025), world-model-like applications (Ball et al., 2025; Robbyant Team et al., 2026), and embodied intelligence (Li et al., 2026b; Ye et al., 2026), which require continuous generation with stable identities, motions, and scene structures over long horizons.

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.

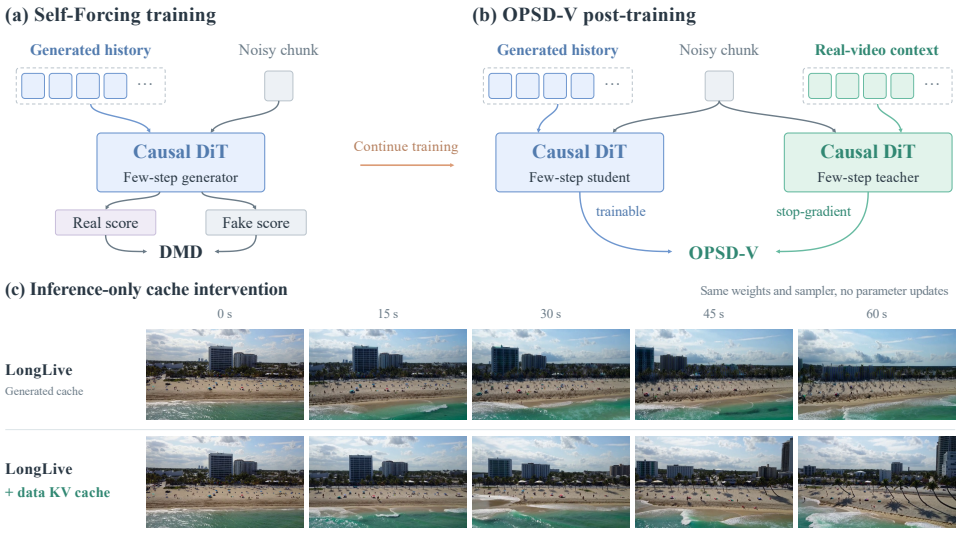

Figure 1: **Overview and inference-time cache intervention.** *Top:* While Self-Forcing applies DMD distribution supervision after self-rollout, OPSD-V supervises an on-policy student against an EMA teacher equipped with a data-assisted KV cache, performing dense velocity matching at identical student-visited denoising states. *Bottom:* A diagnostic comparison using the original LongLive model. Both rollouts start from the same real-video first chunk. The upper row relies entirely on model-generated history, whereas the lower row replaces older historical KV entries with corresponding real-video features while keeping the latest model-generated chunk. Without updating parameters or modifying the sampler, this simple cache intervention significantly stabilizes long-horizon rollout, motivating our teacher cache design. The protocol is detailed in Appendix C.5.

Despite the success of self-forcing training and DMD-style few-step distillation, existing AR video generators still suffer from long-horizon degradation.A fundamental limitation is that DMD-style training relies on a bidirectional short-clip teacher, whose limited temporal range constrains supervision for long-horizon autoregressive generation. While recent works Yang et al. (2025b); Cui et al. (2025); Zhang et al. (2026a); Li et al. (2026a) extend training rollouts and leverage attention sinks Xiao et al. (2023) to mitigate error accumulation, their supervision remains constrained by the temporal range and capacity of the underlying clip-level teacher. Moreover, DMD often induces weakened dynamics and color drift, providing chunk- or clip-level distribution matching rather than dense, corrective targets along the entire denoising trajectory.

This motivates us to seek a better supervisory source than traditional bidirectional teacher models to reduce long-horizon degradation in few-step AR generators. Real long-video data offers a natural alternative: real sequences inherently exhibit higher visual realism and more coherent motion than synthetic model outputs. Driven by this intuition, we first experimented with supervised fine-tuning (SFT) of the DMD-distilled model on real long videos. However, this direct approach leads to severe quality degradation during few-step sampling (Fig. 3), as our SFT baseline trains on noisy states constructed from real videos rather than those visited during the model's few-step AR rollout (Yin et al., 2024a; Jiang et al., 2026). This clear training–inference mismatch leads us to the central question:

Can real long-video data serve as stronger training supervision while preserving the original few-step AR generation capability?

Since using real data as a direct training target is too harsh for the few-step trajectory, an intuitive solution is to take a softer approach. Instead of forcing the model to reproduce real-world latents, real data can simply serve as an informative backdrop to provide better guidance. This idea leads naturally to On-Policy Self-Distillation (OPSD). Recently studied in autoregressive large language models (Zhao et al., 2026; Shenfeld et al., 2026; Hübotter et al., 2026), OPSD adopts an on-policy learning paradigm in which the model samples from its current policy as a student, while a stronger

teacher distribution is obtained by conditioning the same model on richer in-context information, helping preserve the model's existing capabilities. This formulation naturally provides dense supervision along the model's own trajectory. Applying this intuition to autoregressive long-video generation, real long videos play a similar role: by adding real-video features into the teacher's temporal KV cache, the teacher can provide cleaner, more reliable velocity targets directly at the denoising states visited by the student during its rollout. A simple inference-time diagnostic strongly supports this design: real-video cache context yields a visibly more stable rollout from the original LongLive model without parameter updates or sampler changes (Fig. 1, bottom).

In this paper, we propose OPSD-V, a post-training framework for few-step autoregressive (AR) video diffusion models (Fig. 1). Continuing from a DMD-distilled few-step AR model, we use a teacher branch to provide dense trajectory-level supervision for the student. Specifically, the student is the few-step AR generator itself, running strictly on-policy: it rolls out chunks conditioned on its own generated history in the KV cache. Meanwhile, the teacher uses real long videos as privileged context: it replaces older history in its cache with real-video features while keeping its latest model prediction, and predicts target velocities at the student's denoising states. Matching these velocities along the student's few-step rollout significantly improves long-horizon video quality without changing the original inference pipeline.

We validate OPSD-V on Self-Forcing (Huang et al., 2025a), a standard few-step AR generation pipeline, and LongLive (Yang et al., 2025b), a baseline tuned for long autoregressive rollouts. Across both backbones, OPSD-V improves motion dynamics and reduces long-horizon degradation while preserving the original few-step inference procedure (Fig. 3).

2 Related Work

Video diffusion foundation models. Diffusion and flow matching (Song et al., 2020; Lipman et al., 2023; Ho et al., 2020), combined with scalable diffusion transformers (DiTs) (Peebles & Xie, 2022), form the foundation of modern video generation systems (Yang et al., 2024; Kong et al., 2024; Wan Team, 2025; Meituan LongCat Team et al., 2025; Brooks et al., 2024; Zheng et al., 2024). While these models achieve strong offline generation quality, they typically rely on bidirectional temporal attention and multi-step sampling, making them impractical for low-latency streaming applications. This limitation motivates the development of causal autoregressive video generation.

Autoregressive video generation and forcing-based training. AR video models generate frames or chunks sequentially using KV caches. Early methods use teacher or diffusion forcing with clean or partially noised context (Jin et al., 2025; Chen et al., 2024; 2025). CausVid distills bidirectional DiTs into few-step causal generators via DMD (Yin et al., 2025; 2024b); Causal Forcing uses an AR teacher for ODE initialization (Zhu et al., 2026). Self-Forcing reduces exposure bias through self-generated rollouts (Huang et al., 2025a), with subsequent work extending rollouts (Cui et al., 2025), refining cache management and attention sinks (Li et al., 2026a; Mao et al., 2026; Yi et al., 2025; Yesiltepe et al., 2026), and enabling long-video streaming (Yang et al., 2025b; Liu et al., 2025b). Self Gradient Forcing restores gradients to KV-cache construction through two-pass training (Zhuang et al., 2026). Reward-Forcing introduces reward feedback (Zhang et al., 2026a), while Astrolabe uses forward-process RL to align distilled AR video generators with visual preferences (Zhang et al., 2026b). Real long videos have also been used to supervise non-causal multi-head DiTs (Cai et al., 2026); OPSD-V instead post-trains few-step causal AR generators through cache-aware on-policy self-distillation while preserving their inference trajectory.

On-policy self-distillation. In autoregressive language models, on-policy self-distillation (OPSD) trains the model on its own sampled trajectories using a stronger teacher distribution obtained by conditioning the same model on richer context (Zhao et al., 2026; Shenfeld et al., 2026; Hübotter et al., 2026). Vision-OPD applies this principle to multimodal language models, using evidence-centered image crops as privileged teacher context to improve fine-grained visual understanding (Yuan et al., 2026). D-OPSD applies this principle to few-step text-to-image diffusion, using paired real images as extra teacher context to supervise the student's on-policy rollouts without degrading few-step inference (Jiang et al., 2026). Other related works further study trajectory-level distillation for continuous diffusion and flow models (Fang et al., 2026; Li et al., 2026c; Gu et al., 2026). OPSD-V applies this paradigm to post-training few-step AR video generators, improving long-horizon visual quality and motion dynamics.

3 PRELIMINARIES

3.1 FEW-STEP AUTOREGRESSIVE VIDEO GENERATION

Autoregressive (AR) video generation models decompose a long video sequence $x_{1:N} = \{x_1, \ldots, x_N\}$ into sequential latent chunks. Given a text condition c, the joint distribution naturally factorizes as

$$p_\theta(x_{1:N} \mid c) = \prod_{i=1}^{N} p_\theta(x_i \mid x_{<i}, c).$$

In causal video transformers, the prefix condition $x_{<i}$ is maintained via a key-value (KV) cache h_i. Upon denoising chunk i, the KV entries of the clean latent are appended to form h_{i+1}, serving as temporal context for subsequent chunks. To achieve real-time streaming, works such as CausVid and Self-Forcing (Yin et al., 2025; Huang et al., 2025a) employ distribution matching distillation (DMD) (Yin et al., 2024b; Liu et al., 2025a) to distill pretrained diffusion models into few-step AR generators.

3.2 ON-POLICY SELF-DISTILLATION

In autoregressive language models, on-policy self-distillation (OPSD) trains a model on its own sampled rollouts using a stronger teacher distribution formed by richer context (Zhao et al., 2026; Shenfeld et al., 2026). Given a query q, the student model samples a full output sequence $\hat{o}_{1:T} \sim \pi_\theta(\cdot \mid q)$. The teacher shares the same architecture (often parameterized by an EMA or frozen copy $\bar{\theta}$), but is conditioned on privileged auxiliary information r, such as ground-truth answers, task demonstrations, or reasoning rationale. At each step m, the student next-token distribution is aligned with the teacher distribution conditioned on the student's own sampled prefix $\hat{o}_{<m}$:

$$\mathcal{L}_{\text{OPSD}} = \mathbb{E}_{\hat{o}_{1:T} \sim \pi_\theta(\cdot \mid q)} \left[\sum_{m=1}^{T} D\left(\pi_{\bar{\theta}}(\cdot \mid q, r, \hat{o}_{<m}) \,\|\, \pi_\theta(\cdot \mid q, \hat{o}_{<m})\right) \right], \tag{1}$$

where $D(\cdot\|\cdot)$ denotes a divergence measure. In this formulation, the student remains strictly on-policy along its own generation path, while the teacher provides corrective supervision enabled by the privileged context r. D-OPSD extends this principle to few-step text-to-image diffusion, using paired real images as additional teacher context to supervise the student's own denoising states (Jiang et al., 2026). In AR video, these states additionally include the temporal KV cache. OPSD-V uses real long videos as privileged teacher context while preserving the student's own rollout and cache updates.

4 METHOD

4.1 OVERVIEW

OPSD-V is a post-training framework that optimizes few-step AR video generators using real long videos as privileged teacher context (Fig. 2). Both student and teacher branches share a frozen base transformer, equipped with trainable student LoRA weights θ and their exponential-moving-average (EMA) teacher copy $\bar{\theta}$. Given a text prompt c and a ground-truth video divided into latent chunks $x_{1:N}^{\text{data}}$, both branches initialize their KV caches from the real first chunk x_1^{data}. The student then rolls out subsequent chunks on-policy, conditioning solely on its own generated history. At each few-step denoising transition, the teacher predicts a target velocity at the exact student-visited state, but conditions on a temporal cache enhanced with real-video features. In this manner, real-world data indirectly guides the student's sampling trajectory through privileged teacher context, avoiding the issue of direct reconstruction fitting.

4.2 STUDENT ON-POLICY ROLLOUT

Let h_i^s denote the student KV cache prior to synthesizing chunk i. Conditioned on the ground-truth first chunk x_1^{data}, we initialize the cache for chunk 2 via

$$h_2^s = \text{KV}_\theta(x_1^{\text{data}}), \tag{2}$$

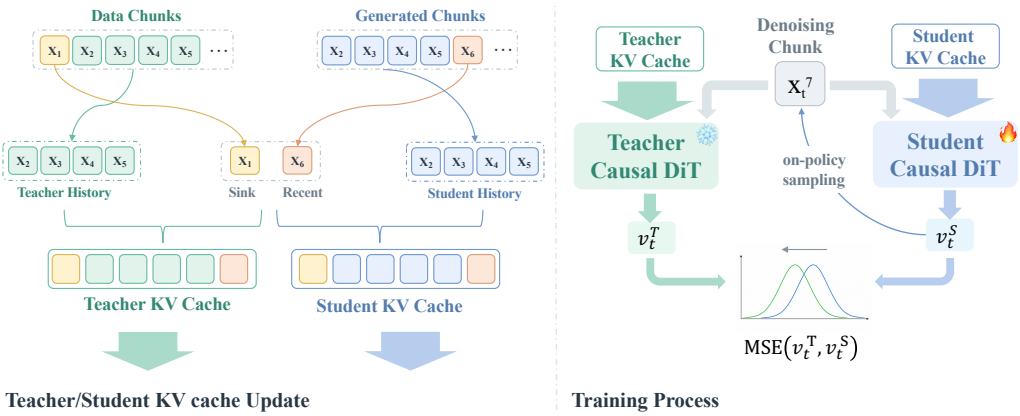

Figure 2: **Overview of OPSD-V, using chunk 7 as an example.** *Left:* Both branches share the real first chunk as an attention sink. Older history comes from student rollouts in the student cache and real videos in the teacher cache. Each branch retains its own latest clean prediction (chunk 6, orange); the teacher's is evaluated at the student's final noisy state. The diagram shows history sources before rolling-window truncation. *Right:* Both branches predict velocities at the same student-visited noisy latent, timestep, and prompt, conditioned on their respective caches. The teacher supplies a stop-gradient velocity target while the student follows its original few-step sampler. The training loss is evaluated only for chunks $i > M$.

where $\mathrm{KV}_\theta(\cdot)$ extracts the temporal key–value states computed by the causal DiT. Serving as a persistent attention sink across the entire sequence, this real prefix is shared between both student and teacher branches as foundational context, and is excluded from generation and loss computation.

Starting from chunk $i = 2$, the student draws standard Gaussian noise $z^s_{i,K} \sim \mathcal{N}(0, \mathbf{I})$ and unrolls the native K-step discrete schedule $t_K > \cdots > t_1 > t_0 = 0$. At step k, it predicts the velocity field conditioned on the current state and cache:

$$\hat{v}^s_{i,k} = f_\theta(z^s_{i,k}, t_k, c, h^s_i), \tag{3}$$

and advances to the next state using the deterministic or stochastic solver Φ:

$$z^s_{i,k-1} = \mathrm{sg}\big(\Phi(z^s_{i,k}, t_k, t_{k-1}, \hat{v}^s_{i,k}; \xi_{i,k})\big), \qquad k = K, \ldots, 1, \tag{4}$$

where $\xi_{i,k}$ denotes sampler-specific stochasticity (e.g., intermediate re-noising), and $\mathrm{sg}(\cdot)$ stops gradients through rollout transitions. Upon reaching the final step, the solver yields the predicted clean latent chunk $\hat{x}^s_i = z^s_{i,0}$. Following the causal context-encoding protocol, we forward this clean prediction through the causal DiT at temporal index i and update the student KV cache detached from gradients:

$$h^s_{i+1} = h^s_i \oplus \mathrm{sg}(\mathrm{KV}_\theta(\hat{x}^s_i)), \tag{5}$$

where $\oplus$ represents the rolling cache update: the chunk-1 attention sink is permanently retained, while older entries beyond the temporal horizon are progressively evicted. To ensure consistent positional encoding during rolling maintenance, relative RoPE embeddings for the attention sink are repositioned as the temporal window slides. Because all post-prefix context is derived solely from the student's self-generated latents, the entire rollout trajectory remains strictly on-policy.

4.3 AR-Consistent Real-Video Teacher Cache

The teacher uses its own cache h^t_i while sharing the student's noisy latent, timestep, text condition, and temporal position:

$$\hat{v}^t_{i,k} = f_{\bar{\theta}}(z^s_{i,k}, t_k, c, h^t_i). \tag{6}$$

It does not sample a separate trajectory. To obtain the latest clean chunk for updating its cache, we compute the teacher's prediction at the last noisy state $z^s_{i,1}$, immediately before the student's final step:

$$\hat{x}^t_i = \mathrm{sg}\big(z^s_{i,1} - \sigma_1 \hat{v}^t_{i,1}\big), \tag{7}$$

where σ_1 is the noise level at t_1 defined by the scheduler.

After completing chunk i, the teacher's clean estimate $\hat{x}_i^t$ is encoded via the causal DiT to update the teacher KV cache. To construct a privileged yet causal context for subsequent generation, we introduce a one-chunk-delayed replacement: for $i \geq 3$, the KV entries of the previous chunk $\hat{x}_{i-1}^t$ are replaced with the ground-truth features extracted from x_{i-1}^{data} at temporal position $i-1$, while all other cache positions and relative RoPE indices remain intact. Consequently, prior to synthesizing chunk $i \geq 3$, the student and teacher branches operate on asymmetric history sequences:

$$\begin{aligned} h_i^s &: [x_1^{\text{data}}, \hat{x}_2^s, \ldots, \hat{x}_{i-1}^s], \\ h_i^t &: [x_1^{\text{data}}, \ldots, x_{i-2}^{\text{data}}, \hat{x}_{i-1}^t]. \end{aligned} \tag{8}$$

Both caches maintain the chunk-1 sink and recent temporal context under the identical rolling policy, simplifying to the shared real prefix $[x_1^{\text{data}}]$ when $i = 2$. This asymmetric cache design balances two vital objectives: substituting older rollout history with ground-truth video features reduces accumulated errors in the teacher context, while retaining the latest generated chunk preserves autoregressive continuity without shortcutting the temporal transition. Crucially, real-video context is restricted strictly to historical chunks, and all teacher targets are evaluated directly at the student's on-policy denoising states.

4.4 Dense Denoising-Level Objective

To supervise the student along its rollout trajectory, we align the velocity predictions of the student and teacher across all discrete denoising steps for chunks $i > M$:

$$\mathcal{L}_{\text{OPSD-V}} = \frac{1}{|\mathcal{S}|} \sum_{(i,k)\in\mathcal{S}} \text{MSE}\big(\hat{v}_{i,k}^s, \text{sg}\big(\hat{v}_{i,k}^t\big)\big)\,, \tag{9}$$

where MSE averages squared errors over all temporal, spatial, and channel latent elements. The set of supervised chunk-step pairs $\mathcal{S}$ is

$$\mathcal{S} = \{(i,k) \mid M < i \leq N,\ 1 \leq k \leq K\}, \qquad |\mathcal{S}| = K(N-M). \tag{10}$$

We exclude the first $M = 7$ chunks (21 latent frames) from the loss. Specifically, after the shared real-prefix chunk 1, chunks 2–7 are generated without loss, while the student and teacher caches are updated normally. This range corresponds to the short-clip horizon already well-modeled by the base model's initial DMD pre-training (Huang et al., 2025a; Yang et al., 2025b), allowing us to focus additional supervision on later autoregressive states prone to error accumulation. From chunk 8 through chunk N, we apply uniform weighting across all $K = 4$ denoising steps.

4.5 Training Procedure

Dense supervision over long AR rollouts causes activation memory to scale with both rollout length and denoising steps if the full computation graph is retained. To eliminate this overhead, OPSD-V adopts a streaming schedule that interleaves teacher evaluation, student prediction, and immediate step-wise backpropagation (see Algorithm 1 in Appendix A.1), discarding intermediate activations after each step rather than holding them across the sequence. Furthermore, the two branches share a single frozen backbone with separate LoRA adapters and KV caches, avoiding duplicate model weights in GPU memory.

Memory-efficient per-step backward.

At each student-visited state $(z_{i,k}^s, t_k, c)$, the teacher computes a velocity target without gradients, followed by the student prediction with gradients enabled only for chunks $i > M$. For each supervised pair $(i,k) \in \mathcal{S}$, we divide the loss by $|\mathcal{S}| = K(N-M)$ and backpropagate immediately. Under Fully Sharded Data Parallel (FSDP) training, we accumulate the resulting gradients in dedicated detached buffers to preserve them across subsequent forward passes. Before advancing the rollout, we detach both branches' KV caches; sampler transitions and cache writes also run without gradients. This preserves temporal context while preventing backpropagation through earlier steps or chunks, allowing saved activations to be released after each supervised step.

Both student and teacher LoRA parameters remain fixed throughout the rollout. With rollout states, cached context, and teacher targets treated as stop-gradient quantities, per-step gradient accumulation yields the gradient of the objective in Eq. (9). Only after completing the rollout do we restore

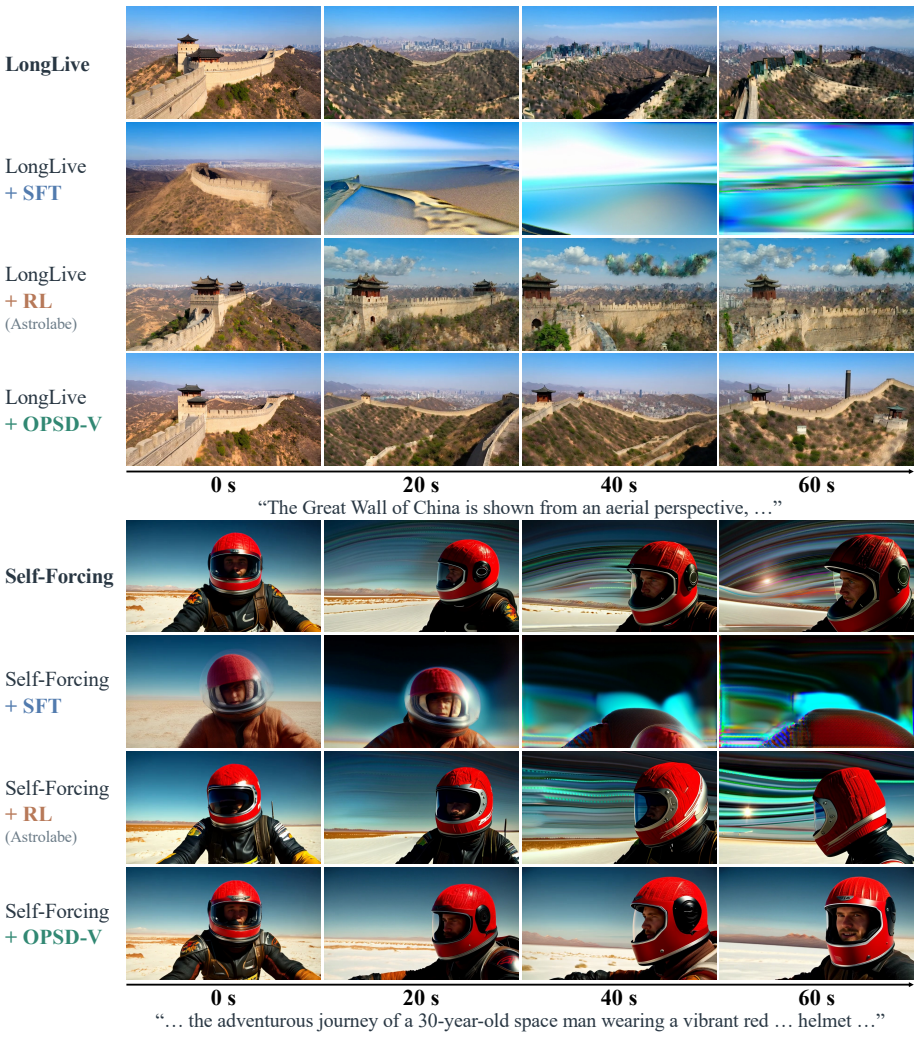

Figure 3: Qualitative comparison of long autoregressive video generation with LongLive (Yang et al., 2025b) (top) and Self-Forcing (Huang et al., 2025a) (bottom). Within each group, the rows show the base model, +SFT, +RL (Astrolabe), and +OPSD-V, from top to bottom. Each row shows four selected frames from the corresponding autoregressive rollout. All methods within each group use the same prompt, random seed, four-step sampler, and attention-sink inference setting.

the accumulated gradients, apply gradient clipping, take one optimizer step to update the student LoRA, and then update the EMA teacher.

5 Experiments

5.1 Experimental Setup

Training setup. We post-train each model with LoRA on 3,800 one-minute videos at 480p for 200 iterations using 24 H800 GPUs. Training uses only the velocity-matching loss in Eq. (9), an EMA teacher, and the original four-step sampler. Data preprocessing and implementation details are provided in Appendix A.

Base models and baselines. We validate OPSD-V across two state-of-the-art few-step AR video generators, Self-Forcing (Huang et al., 2025a) and LongLive (Yang et al., 2025b). On each base architecture, we benchmark OPSD-V against the original baseline as well as two representative post-training paradigms: supervised fine-tuning (SFT) using standard flow-matching loss and reinforce-

Table 1: One-minute video generation, averaged equally over MovieGenBench and MeiBench (120 prompts each). All methods use the Wan2.1-T2V-1.3B backbone and a four-step AR sampler. Quality and Semantic average six raw dimension scores each; Dynamic Degree uses 5-second windows. Red: best. Blue: second-best, within each backbone. ↑ Higher is better.

Method	Quality ↑	Dynamic ↑	Semantic ↑
LongLive	0.7867	0.5156	0.5123
LongLive + SFT	0.6448	0.2076	0.4132
LongLive + RL (Astrolabe)	0.7222	0.1804	0.5124
LongLive + OPSD-V	0.7992	0.6077	0.5068
Self-Forcing	0.8020	0.6421	0.5041
Self-Forcing + SFT	0.6219	0.1750	0.3844
Self-Forcing + RL (Astrolabe)	0.7702	0.5247	0.5085
Self-Forcing + OPSD-V	0.8158	0.6715	0.5046

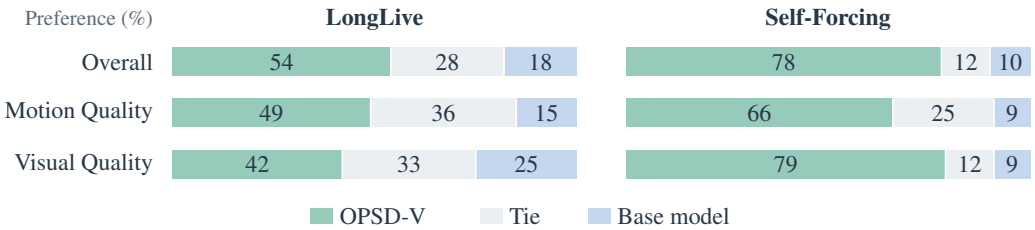

Figure 4: **User study preference results.** Preferences between OPSD-V and the corresponding base models for overall, motion, and visual quality. Percentages include ties; the protocol and aggregation rules are provided in Appendix B.3.

ment learning (RL) via Astrolabe (Zhang et al., 2026b). We also qualitatively compare Self-Forcing + OPSD-V with Self Gradient Forcing (Zhuang et al., 2026).

Evaluation protocol. Within each backbone, the base, SFT, Astrolabe, and OPSD-V comparisons use identical prompts, random seeds, solvers, and attention-sink settings. For quantitative evaluation, we compile a 240-prompt benchmark comprising the first 120 prompts from MovieGenBench (Polyak et al., 2024) and a complementary 120-prompt set termed MeiBench, synthesizing one video per prompt for assessment via VBench-Long (Huang et al., 2025b). We report balanced averages across both evaluation sets; detailed metric breakdowns and inference settings are documented in Appendix B.

5.2 Qualitative Comparison

Fig. 3 compares the base models, SFT, Astrolabe, and OPSD-V on both backbones. SFT exhibits severe visual degradation, while the base models and Astrolabe show structural distortions or background artifacts in later frames. OPSD-V preserves clearer Great Wall structures in LongLive and substantially reduces background streaking in the Self-Forcing astronaut sequence, while retaining changes in viewpoint and subject pose. These examples illustrate improved visual stability over long rollouts without changing the original four-step inference path. Additional comparisons with Self Gradient Forcing (Zhuang et al., 2026) are provided in Appendix E.

5.3 Quantitative Comparison

Tab. 1 shows that SFT lowers all three scores relative to the base models, while RL-based Astrolabe reduces quality and dynamics despite slightly higher semantic scores. OPSD-V achieves the highest Quality and Dynamic Degree on both backbones, with Semantic slightly below the LongLive baseline and slightly above the Self-Forcing baseline, retaining the original four-step sampler. In our paired user study (Fig. 4), participants favor OPSD-V over the base models for overall and motion quality on both backbones, with stronger visual-quality preference on Self-Forcing. Study details are in Appendix B.3.

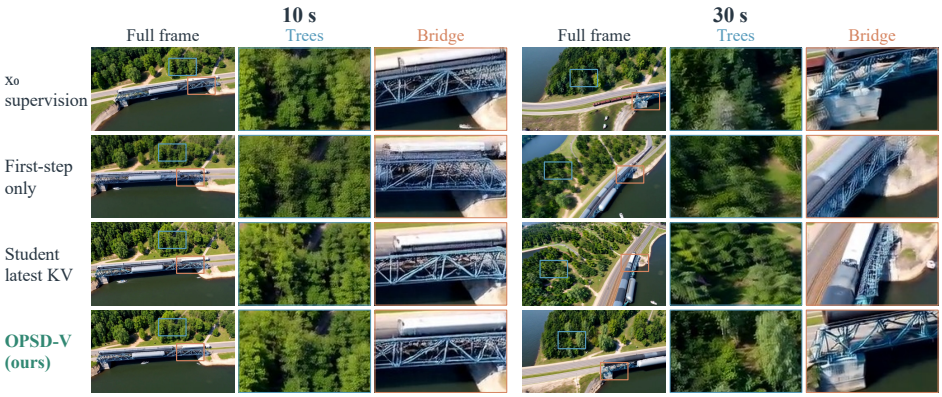

Figure 5: **Qualitative ablation on LongLive.** Rows compare x_0 supervision, first-step only, student-latest KV, and the full OPSD-V. At 10 s and 30 s, blue and orange boxes mark tree and bridge regions, respectively, enlarged beside each full frame. The 30 s crops highlight clearer foliage and bridge details with the full method, while the ablations show varying degrees of blur or local distortion.

Table 2: Ablation of OPSD-V on LongLive, averaged equally over MovieGenBench and MeiBench. Metric definitions follow Tab. 1. Red: best. Blue: second-best, among the LongLive variants. ↑ Higher is better.

Variant	Quality ↑	Dynamic ↑	Semantic ↑
x_0 supervision	0.7923	0.5646	0.5124
First-step only	0.7585	0.4547	0.5109
Student-latest KV	0.7731	0.5228	0.5092
OPSD-V (full)	0.7992	0.6077	0.5068

5.4 Ablation Study

We ablate three design choices on LongLive, retaining four-step inference. Appendix C provides detailed settings and an additional trajectory-alignment diagnostic.

x_0 supervision. We replace velocity matching with MSE on student and teacher clean-latent predictions at identical on-policy states, which implicitly downweights later denoising steps. Compared to the full method, Quality and Dynamic Degree decrease (Tab. 2), and fine details on trees and bridges degrade at 30 s (Fig. 5). Appendix C.1 derives the relationship between the two objectives.

First-step only. We supervise only the initial, highest-noise step of each supervised chunk while executing all four steps for rollout and inference. This yields lower Quality and Dynamic Degree (Tab. 2) with blurrier foliage and distorted bridge structures at 30 s (Fig. 5), supporting dense denoising-level supervision. Loss normalization and implementation details are provided in Appendix C.2.

Student-latest KV. We use the student's completed chunk rather than the teacher's clean prediction to update the latest teacher-cache slot, while preserving older real-video context. This hurts Quality and Dynamic Degree (Tab. 2) and degrades local details at 30 s (Fig. 5). Appendix C.3 details the cache update and its semantic trade-off.

6 Conclusion

We presented OPSD-V, a cache-aware on-policy self-distillation framework for post-training few-step autoregressive video generators. By using real long videos as privileged temporal context, OPSD-V constructs cleaner teacher caches and provides dense velocity supervision at the denoising states visited by the student. Experiments on Self-Forcing and LongLive show improved motion dynamics and reduced long-horizon degradation, without changing the original few-step sampler or inference-time cache mechanism. These results support on-policy self-distillation as an effective way to incorporate real long-video data while preserving few-step autoregressive generation.

AI USE STATEMENT

We used generative AI tools for manuscript drafting and translation, literature review, feedback on experimental design, implementation and debugging, result analysis, and visualization. AI-assisted work was reviewed and revised under author supervision. The authors take responsibility for the accuracy, originality, and integrity of the final submission.

ETHICS STATEMENT

OPSD-V improves long-video synthesis and shares the potential risks of video generation, including deceptive media, impersonation, and biases inherited from pretrained models and training data. Generated videos should be clearly identified as synthetic. The collection and use of real videos, as well as model deployment, should respect privacy, consent where applicable, and the licenses of source materials and pretrained models. Our user study evaluates preferences between generated videos; its protocol and aggregate results are described in Appendix B.3.

REPRODUCIBILITY STATEMENT

Appendix A provides training pseudocode, data processing, model initialization, hyperparameters, and cache-update details. Appendix B describes generation settings, metric aggregation, and the user-study protocol, while Appendix C specifies the ablation variants. Code and model checkpoints are available through our anonymous code repository and model repository, respectively, with additional visual comparisons on our anonymous project page.

REFERENCES

Philip J. Ball, Jakob Bauer, Frank Belletti, Bethanie Brownfield, Ariel Ephrat, Shlomi Fruchter, Agrim Gupta, Kristian Holsheimer, Aleksander Holynski, Jiri Hron, Christos Kaplanis, Marjorie Limont, Matt McGill, Yanko Oliveira, Jack Parker-Holder, Frank Perbet, Guy Scully, Jeremy Shar, Stephen Spencer, Omer Tov, Ruben Villegas, Emma Wang, Jessica Yung, Cip Baetu, Jordi Berbel, David Bridson, Jake Bruce, Gavin Buttimore, Sarah Chakera, Bilva Chandra, Paul Collins, Alex Cullum, Bogdan Damoc, Vibha Dasagi, Maxime Gazeau, Charles Gbadamosi, Woohyun Han, Ed Hirst, Ashyana Kachra, Lucie Kerley, Kristian Kjems, Eva Knoepfel, Vika Koriakin, Jessica Lo, Cong Lu, Zeb Mehring, Alex Moufarek, Henna Nandwani, Valeria Oliveira, Fabio Pardo, Jane Park, Andrew Pierson, Ben Poole, Helen Ran, Tim Salimans, Manuel Sanchez, Igor Saprykin, Amy Shen, Sailesh Sidhwani, Duncan Smith, Joe Stanton, Hamish Tomlinson, Dimple Vijaykumar, Luyu Wang, Piers Wingfield, Nat Wong, Keyang Xu, Christopher Yew, Nick Young, Vadim Zubov, Douglas Eck, Dumitru Erhan, Koray Kavukcuoglu, Demis Hassabis, Zoubin Gharamani, Raia Hadsell, Aäron van den Oord, Inbar Mosseri, Adrian Bolton, Satinder Singh, and Tim Rocktäschel. Genie 3: A new frontier for world models. 2025.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. URL https://openai.com/research/video-generation-models-as-world-simulators.

Shengqu Cai, Weili Nie, Chao Liu, Julius Berner, Lvmin Zhang, Nanye Ma, Hansheng Chen, Maneesh Agrawala, Leonidas Guibas, Gordon Wetzstein, and Arash Vahdat. Mode seeking meets mean seeking for fast long video generation. In *ICML*, 2026.

Boyuan Chen, Diego Marti Monso, Yilun Du, Max Simchowitz, Russ Tedrake, and Vincent Sitzmann. Diffusion forcing: Next-token prediction meets full-sequence diffusion. *arXiv preprint arXiv:2407.01392*, 2024.

Guibin Chen, Dixuan Lin, Jiangping Yang, Chunze Lin, Junchen Zhu, Mingyuan Fan, Hao Zhang, Sheng Chen, Zheng Chen, Chengcheng Ma, Weiming Xiong, Wei Wang, Nuo Pang, Kang Kang, Zhiheng Xu, Yuzhe Jin, Yupeng Liang, Yubing Song, Peng Zhao, Boyuan Xu, Di Qiu, Debang Li, Zhengcong Fei, Yang Li, and Yahui Zhou. Skyreels-v2: Infinite-length film generative model, 2025. URL https://arxiv.org/abs/2504.13074.

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.

Zhen Fang, Wenxuan Huang, Yu Zeng, Yiming Zhao, Shuang Chen, Kaituo Feng, Yunlong Lin, Lin Chen, Zehui Chen, Shaosheng Cao, and Feng Zhao. Flow-opd: On-policy distillation for flow matching models. *arXiv preprint arXiv:2605.08063*, 2026.

Yuchao Gu, Guian Fang, Yuxin Jiang, Weijia Mao, Song Han, Han Cai, and Mike Zheng Shou. Anyflow: Any-step video diffusion model with on-policy flow map distillation. *arXiv preprint arXiv:2605.13724*, 2026.

Yoav HaCohen, Benny Brazowski, Nisan Chiprut, Yaki Bitterman, Andrew Kvochko, Avishai Berkowitz, Daniel Shalem, Daphna Lifschitz, Dudu Moshe, Eitan Porat, et al. Ltx-2: Efficient joint audio-visual foundation model. *arXiv preprint arXiv:2601.03233*, 2026.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

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*, 2025a.

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

Ziqi Huang, Fan Zhang, Xiaojie Xu, Yinan He, Jiashuo Yu, Ziyue Dong, Qianli Ma, Nattapol Chanpaisit, Chenyang Si, Yuming Jiang, et al. Vbench++: Comprehensive and versatile benchmark suite for video generative models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025b.

Jonas Hübotter, Frederike Lübeck, Lejs Behric, Anton Baumann, Marco Bagatella, Daniel Marta, Ido Hakimi, Idan Shenfeld, Thomas Kleine Buening, Carlos Guestrin, and Andreas Krause. Reinforcement learning via self-distillation. *arXiv preprint arXiv:2601.20802*, 2026.

Dengyang Jiang, Xin Jin, Dongyang Liu, Zanyi Wang, Mingzhe Zheng, Ruoyi Du, Xiangpeng Yang, Qilong Wu, Zhen Li, Peng Gao, Harry Yang, and Steven Hoi. D-opsd: On-policy self-distillation for continuously tuning step-distilled diffusion models. *arXiv preprint arXiv:2605.05204*, 2026.

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. In *International Conference on Learning Representations*, volume 2025, pp. 23378–23402, 2025.

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

Haodong Li, Shaoteng Liu, Zhe Lin, and Manmohan Chandraker. Rolling sink: Bridging limited-horizon training and open-ended testing in autoregressive video diffusion. *arXiv preprint arXiv:2602.07775*, 2026a.

Lin Li, Qihang Zhang, Yiming Luo, Shuai Yang, Ruilin Wang, Fei Han, Mingrui Yu, Zelin Gao, Nan Xue, Xing Zhu, Yujun Shen, and Yinghao Xu. Causal world modeling for robot control. *arXiv preprint arXiv:2601.21998*, 2026b.

Quanhao Li, Junqiu Yu, Kaixun Jiang, Yujie Wei, Zhen Xing, Pandeng Li, Ruihang Chu, Shiwei Zhang, Yu Liu, and Zuxuan Wu. Diffusionopd: A unified perspective of on-policy distillation in diffusion models. *arXiv preprint arXiv:2605.15055*, 2026c.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. *International Conference on Learning Representations*, 2023.

Dongyang Liu, Peng Gao, David Liu, Ruoyi Du, Zhen Li, Qilong Wu, Xin Jin, Sihan Cao, Shifeng Zhang, Hongsheng Li, and Steven Hoi. Decoupled dmd: Cfg augmentation as the spear, distribution matching as the shield. *arXiv preprint arXiv:2511.22677*, 2025a.

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*, 2025b.

Xiaofeng Mao, Shaohao Rui, Kaining Ying, Bo Zheng, Chuanhao Li, Mingmin Chi, and Kaipeng Zhang. Packforcing: Short video training suffices for long video sampling and long context inference. *arXiv preprint arXiv:2603.25730*, 2026.

Meituan LongCat Team, Xunliang Cai, Qilong Huang, Zhuoliang Kang, Hongyu Li, Shijun Liang, Liya Ma, Siyu Ren, Xiaoming Wei, Rixu Xie, and Tong Zhang. Longcat-video technical report. *arXiv preprint arXiv:2510.22200*, 2025.

Meituan LongCat Team, Xunliang Cai, Meng Cheng, Feng Gao, Zhe Kong, Jiamu Li, Le Li, Weiheng Li, Hongyu Liu, Shuai Tan, et al. Longcat-video-avatar 1.5 technical report. *arXiv preprint arXiv:2605.26486*, 2026.

William Peebles and Saining Xie. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022.

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, Yihang Chen, Jie Liu, Yansong Cheng, Yao Yao, Jiayi Zhu, Yihao Meng, Kecheng Zheng, Qingyan Bai, Jingye Chen, Zehong Shen, Yue Yu, Xing Zhu, Yujun Shen, and Hao Ouyang. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026.

Sand. ai, Hansi Teng, Hongyu Jia, Lei Sun, Lingzhi Li, Maolin Li, Mingqiu Tang, Shuai Han, Tianning Zhang, W. Q. Zhang, Weifeng Luo, Xiaoyang Kang, Yuchen Sun, Yue Cao, Yunpeng Huang, Yutong Lin, Yuxin Fang, Zewei Tao, Zheng Zhang, Zhongshu Wang, Zixun Liu, Dai Shi, Guoli Su, Hanwen Sun, Hong Pan, Jie Wang, Jiexin Sheng, Min Cui, Min Hu, Ming Yan, Shucheng Yin, Siran Zhang, Tingting Liu, Xianping Yin, Xiaoyu Yang, Xin Song, Xuan Hu, Yankai Zhang, and Yuqiao Li. Magi-1: Autoregressive video generation at scale, 2025. URL https://arxiv.org/abs/2505.13211.

Idan Shenfeld, Mehul Damani, Jonas Hübotter, and Pulkit Agrawal. Self-distillation enables continual learning. *arXiv preprint arXiv:2601.19897*, 2026.

Joonghyuk Shin, Zhengqi Li, Richard Zhang, Jun-Yan Zhu, Jaesik Park, Eli Shechtman, and Xun Huang. Motionstream: Real-time video generation with interactive motion controls. *arXiv preprint arXiv:2511.01266*, 2025.

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

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

Lizhen Wang, Yongming Zhu, Zhipeng Ge, Youwei Zheng, Longhao Zhang, Tianshu Hu, Shiyang Qin, Mingshuang Luo, Jiaxu Zhang, Xin Chen, et al. Flowact-r1: Towards interactive humanoid video generation. *arXiv preprint arXiv:2601.10103*, 2026.

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

Shaoshu Yang, Zhe Kong, Feng Gao, Meng Cheng, Xiangyu Liu, Yong Zhang, Zhuoliang Kang, Wenhan Luo, Xunliang Cai, Ran He, and Xiaoming Wei. Infinitetalk: Audio-driven video generation for sparse-frame video dubbing, 2025a. URL https://arxiv.org/abs/2508.14033.

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

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. *arXiv preprint arXiv:2408.06072*, 2024.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, Ayaan Malik, Kyungmin Lee, William Liang, Nadun Ranawaka, Jiasheng Gu, Yinzhen Xu, Guanzhi Wang, Fengyuan Hu, Avnish Narayan, Johan Bjorck, Jing Wang, Gwanghyun Kim, Dantong Niu, Ruijie Zheng, Yuqi Xie, Jimmy Wu, Qi Wang, Ryan Julian, Danfei Xu, Yilun Du, Yevgen Chebotar, Scott Reed, Jan Kautz, Yuke Zhu, Linxi "Jim" Fan, and Joel Jang. World action models are zero-shot policies, 2026. URL https://arxiv.org/abs/2602.15922.

Hidir Yesiltepe, Tuna Meral, Adil Kaan Akan, Kaan Oktay, and Pinar Yanardag. Infinity-rope: Action-controllable infinite video generation emerges from autoregressive self-rollout. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 40256–40265, 2026.

Jung Yi, Wooseok Jang, Paul Hyunbin Cho, Jisu Nam, Heeji Yoon, and Seungryong Kim. Deep forcing: Training-free long video generation with deep sink and participative compression. *arXiv preprint arXiv:2512.05081*, 2025.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. *Advances in neural information processing systems*, 37:47455–47487, 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*, pp. 6613–6623, 2024b.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Qianhao Yuan, Jie Lou, Xing Yu, Hongyu Lin, Le Sun, Xianpei Han, and Yaojie Lu. Vision-OPD: Learning to see fine details for multimodal llms via on-policy self-distillation. *arXiv preprint arXiv:2605.18740*, 2026.

Jingran Zhang, Ning Li, Yuanhao Ban, Andrew Bai, and Justin Cui. Reward-forcing: Autoregressive video generation with reward feedback. *arXiv preprint arXiv:2601.16933*, 2026a.

Songchun Zhang, Zeyue Xue, Siming Fu, Jie Huang, Xianghao Kong, Yue Ma, Haoyang Huang, Nan Duan, and Anyi Rao. Astrolabe: Steering forward-process reinforcement learning for distilled autoregressive video models. *arXiv preprint arXiv:2603.17051*, 2026b.

Siyan Zhao, Zhihui Xie, Mengchen Liu, Jing Huang, Guan Pang, Feiyu Chen, and Aditya Grover. Self-distilled reasoner: On-policy self-distillation for large language models. *arXiv preprint arXiv:2601.18734*, 2026.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

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.

Junhao Zhuang, Shiyi Zhang, Yuxuan Bian, Yaowei Li, Yawen Luo, Yijun Liu, Weiyang Jin, Songchun Zhang, Xianglong He, Xuying Zhang, Haoran Li, Haoyang Huang, Zeyue Xue, and Nan Duan. Self gradient forcing: Native long video extrapolation. *arXiv preprint arXiv:2607.20368*, 2026.

A Training and Implementation Details

A.1 Training Algorithm

Algorithm 1 Cache-Aware On-Policy Self-Distillation (OPSD-V)

Require: Dataset $\mathcal{D}$; pretrained student f_θ; original sampler Φ and schedule $\{t_k\}_{k=0}^{K}$; number of initial chunks excluded from the loss $M < N$; EMA decay β.

Initialize teacher LoRA: $\bar{\theta} \leftarrow \theta$; freeze the shared backbone.
Perform all cache operations and sampler transitions without gradients.
while training **do**
  Sample $(x_{1:N}^{\text{data}}, c) \sim \mathcal{D}$; zero accumulated gradients $g \leftarrow 0$.
  Initialize h_2^s and h_2^t from x_1^{data} using θ and $\bar{\theta}$, respectively.
  for $i = 2, \ldots, N$ **do**
    Sample $z_{i,K}^s \sim \mathcal{N}(0, I)$.
    for $k = K, \ldots, 1$ **do**
      Teacher: $\hat{v}_{i,k}^t \leftarrow f_{\bar{\theta}}(z_{i,k}^s, t_k, c, h_i^t)$ **(no gradients)**.
      Student: $\hat{v}_{i,k}^s \leftarrow f_\theta(z_{i,k}^s, t_k, c, h_i^s)$ **(gradients only for** $i > M$**)**.
      if $i > M$ **then**
        $\ell_{i,k} \leftarrow \text{MSE}(\hat{v}_{i,k}^s, \text{sg}(\hat{v}_{i,k}^t))/[K(N-M)]$.
        Backpropagate: $g \leftarrow g + \nabla_\theta \ell_{i,k}$.
        Detach cache tensors and release the activation graph.
      end if
      if $k = 1$ **then** save teacher chunk $\hat{x}_i^t \leftarrow \text{sg}(z_{i,1}^s - \sigma_1 \hat{v}_{i,1}^t)$.
      Advance the detached student state $z_{i,k-1}^s$ with Φ (Eq. (4)).
    end for
    Student cache: $\hat{x}_i^s \leftarrow z_{i,0}^s$; $h_{i+1}^s \leftarrow \text{CacheWrite}_\theta(h_i^s, \hat{x}_i^s, i)$.
    Teacher cache: $h_{i+1}^t \leftarrow \text{CacheWrite}_{\bar{\theta}}(h_i^t, \hat{x}_i^t, i)$.
    if $i \geq 3$ **then**
      Re-encode x_{i-1}^{data} with $f_{\bar{\theta}}$ at temporal position $i-1$.
      Overwrite only chunk $i-1$ in h_{i+1}^t; preserve other KV features and cache positions.
    end if
  end for
  Take one optimizer step on θ with clipped g; EMA: $\bar{\theta} \leftarrow \beta\bar{\theta} + (1-\beta)\theta$.
end while

A.2 Data Representation and Optimization

Dataset and Preprocessing. We curate 3,800 one-minute video clips at 480p resolution, spanning natural landscapes, dynamic camera trajectories, and human-centric actions. Automated filtering based on optical flow and perceptual quality heuristics removes samples with negligible motion, visual instability, or severe artifacts. Each training instance pairs a video with a detailed prompt describing its visual content over time. Video latents are precomputed using the frozen 3D causal VAE from Wan2.1 (Wan Team, 2025), and text embeddings are extracted via the pre-trained text encoder. Both modalities are persisted within an LMDB database to enable high-throughput synchronized loading during training. At 480×832 pixel resolution, the resulting video latent tensor has dimension $T \times 16 \times 60 \times 104$ (corresponding to temporal length, channels, height, and width). After the shared real-video prefix initializes both caches, subsequent real-video latents supply teacher context; the student continues from its own generated history.

Architecture and Parameterization. We insert trainable LoRA adapters into all linear projections within the causal DiT transformer blocks, covering self-attention, cross-attention, and feed-forward layers. For LongLive, training resumes directly from its officially released LoRA weights, whereas for Self-Forcing, we train a freshly initialized adapter on top of the released base model. The student and EMA teacher share the identical frozen backbone weights while maintaining distinct LoRA parameters and KV caches. The native four-step sampling process strictly preserves the base model's timestep warping schedule, clean-latent estimation, and intermediate stochastic re-noising steps. Complete hyperparameters are summarized in Tab. 3.

Optimization and System Schedule. We optimize exclusively via the velocity-matching loss in Eq. (9), omitting auxiliary DMD objectives, reward models, or adversarial losses. The EMA teacher is maintained solely during training and completely discarded prior to inference. Training runs on 24 NVIDIA H800 GPUs with a per-device batch size of 1 in BF16 mixed precision, leveraging FSDP, activation checkpointing, and our streaming per-step backward schedule (Appendix A.4).

Table 3: Training configuration for OPSD-V on both backbones. Timesteps are listed before the backbone's timestep warping.

Setting	Value
LoRA rank / scaling parameter / dropout	256 / 256 / 0
Optimizer / learning rate	AdamW / 10^{-5}
Adam coefficients / weight decay	$(0, 0.999)$ / 0.01
Gradient clipping	Global norm 10
Teacher EMA decay	0.9999
Denoising timesteps / timestep shift	$[1000, 750, 500, 250]$ / 5
Training rollout / chunk size	180 / 3 latent frames
Attention window / persistent sink	12 / 3 latent frames
Loss starts at / supervised steps per chunk	Chunk 8 / all 4
Updates / per-GPU batch size	200 / 1
Hardware / arithmetic	24 H800 GPUs / BF16
Memory management	FSDP and activation checkpointing

A.3 Rollout and Cache Construction

For full-length samples, each training rollout uses a contiguous segment of 180 latent frames, divided into 60 chunks of 3 frames each. Shorter samples are clipped to the available number of complete chunks. Both the student and teacher initialize their KV caches using the ground-truth first chunk, which acts as a permanent attention sink. In subsequent steps, the student simply caches its own generated chunks. In contrast, after each chunk is generated, the teacher temporarily saves its clean prediction and then replaces the preceding generated chunk with the corresponding ground-truth video features. This replacement is performed at the original temporal index and does not advance the cache position. We only use past real-video chunks, ensuring no future frames are leaked when generating the current chunk.

The local attention budget is set to 12 latent frames, which accounts for the 3-frame sink, the current 3-frame chunk, and 6 frames for the two most recent historical chunks once the window is filled. Therefore, the sequences in Eq. (8) describe the source of historical context rather than an expanding attention window. Following the shared rolling relative-RoPE strategy, the sink keys are repositioned relative to the retained recent window as time advances.

The first $M = 7$ chunks (21 latent frames) are excluded from the distillation loss, matching the short-clip duration covered during the base models' initial DMD pre-training. Specifically, chunk 1 serves as the real prefix, while chunks 2–7 are generated with normal cache updates but no loss. All four denoising steps of chunks 8–60 are fully supervised, resulting in $53 \times 4 = 212$ equally weighted chunk-step pairs for each full rollout.

A.4 Streaming Gradient Accumulation

For each supervised pair, we evaluate the teacher without gradients, restore the student adapter, and backpropagate the student loss immediately. Detached parameter gradients are accumulated in separate buffers across FSDP forward/backward calls. Both adapters remain fixed throughout a rollout. We detach cached tensors and sampler transitions before continuing, so each subsequent loss requires only its own forward graph. At the end of the rollout, we restore the summed gradients, clip their global norm, update the student once, and then update the EMA teacher.

This computes the gradient of the normalized stop-gradient objective in Eq. (9); it does not differentiate through the process that sampled the rollout states. Saved activations are released after each supervised step, while rolling caches and parameter-gradient buffers persist. Backward activations

therefore do not accumulate over all 212 supervised predictions. Immediate backward calls do not change the optimizer or EMA update frequency.

A.5 Post-Training Baselines

Supervised fine-tuning. We fine-tune LoRA adapters with a flow-matching objective for 200 optimizer updates. LongLive starts from its released base and adapter; Self-Forcing starts from its official EMA base checkpoint with a fresh adapter. The SFT inputs are noisy real-video latents, and historical KV features are computed from real video. There is no distillation teacher or student-generated training rollout. We evaluate the final online adapter for each backbone.

Reward-based post-training. Our Astrolabe baseline uses the released HPSv3-only presets (Zhang et al., 2026b). We use rank-256 LoRA, AdamW with learning rate 10^{-5}, coefficients $(0.9, 0.999)$ and weight decay 10^{-4}, gradient clipping at norm 1, and EMA decay 0.9. The NFT coefficient is 0.1 and the KL weight is 10^{-4}. Training samples 81-frame clips from VidProM prompts, with 24 candidates per prompt and 384 candidates per update across eight GPUs.

B Evaluation Details

B.1 Generation Protocol

The benchmark evaluations use text-only generation: the first chunk is sampled from noise, and subsequent chunks are conditioned on generated history. No real-video prefix or real-video cache replacement is used. The base, SFT, Astrolabe, and OPSD-V comparisons within a backbone use the same four-step sampler and rolling attention-sink setting, with a 12-latent-frame window and a 3-frame sink. Thus, the real first chunk in Appendix A.3 is a training condition; the separate cache-intervention diagnostic in Appendix C.5 also uses a real prefix to control its initial context.

For each benchmark prompt, inference generates 243 latent frames in 81 chunks. Wan's temporal decoder yields 969 RGB frames, exported at 832×480 and 16 FPS, or approximately one minute. We evaluate one video per prompt on the first 120 MovieGenBench prompts and the 120 MeiBench prompts. Results are averaged within each benchmark and then combined with equal benchmark weights. The base, SFT, Astrolabe, and OPSD-V comparisons use the same prompt lists and seed setting.

B.2 Metrics and Detailed Scores

We report twelve VBench-based dimensions (Huang et al., 2024; 2025b) using the following feature extractors, scoring rules, and temporal settings.

Subject and background consistency. Subject consistency uses DINO ViT-B/16 features from 16 frames sampled within each two-second clip. It measures cosine similarity between normalized frame features, using both inter-frame comparisons and the first frame as a reference. The long-video score combines within-clip and cross-clip consistency. Background consistency uses CLIP ViT-B/32 image features, with the same two-second, 16-frame sampling, and aggregates inter-frame and cross-clip cosine similarities. The aggregation of within-clip and cross-clip scores follows the evaluation metadata.

Aesthetic and imaging quality. Aesthetic quality uses the LAION aesthetic predictor, comprising CLIP ViT-L/14 and an MLP, to score sampled frames. Imaging quality uses MUSIQ in evaluation mode. Both frame-quality scores are normalized to $[0, 1]$ by their respective metric implementations.

Motion smoothness and dynamic degree. Motion smoothness uses AMT-S frame interpolation and measures the agreement between interpolated frames and the corresponding observed video frames. Dynamic degree uses RAFT optical flow to identify dynamic windows and reports their proportion. We use its 5-second-window version throughout the reported results. A higher dynamic-degree score indicates more motion, rather than necessarily better perceptual quality. It contributes exactly once to Quality (6) and is also reported separately; the 2-second-window results are not used in our aggregates.

Object, color, and scene alignment. These metrics read explicit target annotations from each prompt's auxiliary metadata, matched by prompt ID; targets are not inferred from video filenames. Object class uses GroundingDINO Swin-B and measures the fraction of frames containing the annotated target. Multiple objects uses the same detector and counts frames in which all annotated targets, separated by *and*, appear together. Color uses GRiT dense captions to locate the target object and measure whether its annotated color is present. Scene uses Tag2Text captions and measures the fraction of frames matching all annotated scene keywords.

Appearance style and overall consistency. Appearance style computes normalized cosine similarity between CLIP ViT-B/32 frame features and the explicit style annotation. It uses 16 sampled frames per two-second clip, aggregates clips into prompt scores, and averages prompts equally. Style annotations cover 80 MovieGenBench prompts and three MeiBench prompts. Overall consistency uses ViCLIP and six complete ten-second clips per one-minute video. To preserve the full prompt, text is split into BPE-token chunks, and video–text matching measures both clip-to-text and text-to-clip coverage. The resulting prompt scores are averaged equally. This is a semantic-coverage score, not a classification accuracy.

Benchmark averaging. For each dimension d, let s_d^{Movie} and s_d^{Mei} denote the scores computed on MovieGenBench and MeiBench, respectively. We report their arithmetic mean, $\bar{s}_d = (s_d^{\text{Movie}} + s_d^{\text{Mei}})/2$. Each benchmark therefore receives equal weight, even when the number of evaluated prompts differs for a particular dimension. A dimension score is not assumed to cover all 120 prompts: evaluation coverage depends on the dimension. Both benchmark scores must be available; missing values are never replaced with zero.

Quality and Semantic aggregates. Quality (6) is the unweighted mean of subject consistency, background consistency, aesthetic quality, imaging quality, motion smoothness, and dynamic degree. Semantic (6) averages object class, multiple objects, color, scene, appearance style, and overall consistency. Each aggregate is computed as $\frac{1}{6}\sum_{d\in\mathcal{D}} \bar{s}_d$, where $\mathcal{D}$ contains its six dimensions. We use the raw dimension scores without applying the normalization or weighting of the official VBench total score. Since both levels use arithmetic means, averaging the six-dimensional aggregate of each benchmark gives the same result.

Detailed scores. Tabs. 4, 5, and 6 list the 12 component dimensions and the two aggregate scores for LongLive, Self-Forcing, and the LongLive ablations, respectively. We retain the available source precision during averaging and round only the final reported scores to four decimal places. Red and blue mark the best and second-best scores within each table, ranked before rounding.

Table 4: Detailed VBench scores for LongLive, averaged equally over MovieGenBench and MeiBench. Red: best. Blue: second-best, within each row. ↑ Higher is better.

Dimension	Base	+SFT	+RL (Astrolabe)	+OPSD-V
Quality components				
Subject consistency	0.9736	0.8662	0.9346	0.9713
Background consistency	0.9602	0.9082	0.9364	0.9558
Aesthetic quality	0.5738	0.4363	0.5886	0.5668
Imaging quality	0.7093	0.4601	0.7008	0.7060
Motion smoothness	0.9880	0.9905	0.9923	0.9875
Dynamic	0.5156	0.2076	0.1804	0.6077
Semantic components				
Object class	0.9903	0.9423	0.9935	0.9883
Multiple objects	0.9087	0.7465	0.9334	0.9133
Color	0.3567	0.1751	0.3316	0.3409
Scene	0.3694	0.1886	0.3725	0.3508
Appearance style	0.2270	0.2351	0.2221	0.2272
Overall consistency	0.2215	0.1915	0.2212	0.2205
Aggregate scores				
Quality	0.7867	0.6448	0.7222	0.7992
Semantic	0.5123	0.4132	0.5124	0.5068

Table 5: Detailed VBench scores for Self-Forcing, averaged equally over MovieGenBench and MeiBench. Red: best. Blue: second-best, within each row. ↑ Higher is better.

Dimension	Base	+SFT	+RL (Astrolabe)	+OPSD-V
Quality components				
Subject consistency	0.9669	0.8435	0.9145	0.9633
Background consistency	0.9544	0.9017	0.9270	0.9491
Aesthetic quality	0.5570	0.4049	0.5594	0.5970
Imaging quality	0.7081	0.4163	0.7085	0.7301
Motion smoothness	0.9834	0.9898	0.9870	0.9839
Dynamic	0.6421	0.1750	0.5247	0.6715
Semantic components				
Object class	0.9771	0.9448	0.9858	0.9775
Multiple objects	0.8973	0.7101	0.9051	0.9074
Color	0.3477	0.1203	0.3552	0.3389
Scene	0.3537	0.1223	0.3532	0.3549
Appearance style	0.2255	0.2298	0.2300	0.2253
Overall consistency	0.2230	0.1791	0.2217	0.2238
Aggregate scores				
Quality	0.8020	0.6219	0.7702	0.8158
Semantic	0.5041	0.3844	0.5085	0.5046

Table 6: Detailed VBench scores for the LongLive ablations, averaged equally over MovieGenBench and MeiBench. Red: best. Blue: second-best, within each row. ↑ Higher is better.

Dimension	x_0 supervision	First-step only	Student-latest KV	OPSD-V (full)
Quality components				
Subject consistency	0.9725	0.9242	0.9183	0.9713
Background consistency	0.9576	0.9299	0.9276	0.9558
Aesthetic quality	0.5665	0.5657	0.5699	0.5668
Imaging quality	0.7050	0.6869	0.7119	0.7060
Motion smoothness	0.9879	0.9893	0.9878	0.9875
Dynamic	0.5646	0.4547	0.5228	0.6077
Semantic components				
Object class	0.9864	0.9886	0.9887	0.9883
Multiple objects	0.9072	0.9183	0.9082	0.9133
Color	0.3581	0.3484	0.3465	0.3409
Scene	0.3724	0.3603	0.3639	0.3508
Appearance style	0.2285	0.2297	0.2261	0.2272
Overall consistency	0.2217	0.2199	0.2218	0.2205
Aggregate scores				
Quality	0.7923	0.7585	0.7731	0.7992
Semantic	0.5124	0.5109	0.5092	0.5068

B.3 User Study

Protocol. We conduct a user study with 10 participants, each comparing 20 video pairs (10 per backbone). Each pair presents the corresponding prompt, one base-model video, and one OPSD-V video generated from the same prompt. Participants choose Model A, Model B, or Same for three criteria: *Overall*, considering all factors; *Motion Quality*, considering smoothness, naturalness, jitter, and discontinuities; and *Visual Quality*, considering clarity, detail, and aesthetics. This gives 100 judgments per backbone per criterion. Fig. 4 reports the distributions including ties.

Preference aggregation. Across both backbones, OPSD-V is preferred in 66.0%, 57.5%, and 60.5% of judgments for overall, motion, and visual quality, respectively. Excluding ties, the corresponding rates are 82.5%, 82.7%, and 78.1%, computed by dividing the judgments favoring OPSD-V by all non-tied judgments.

C Detailed Ablation Studies

C.1 x_0 Supervision

This variant matches student and teacher clean-latent estimates at the same student-visited noisy state. Under the backbone's flow parameterization, $\hat{x}^{s/t}_{i,k} = z^s_{i,k} - \sigma_k \hat{v}^{s/t}_{i,k}$, so the clean-latent MSE equals $\sigma_k^2 \,\mathrm{MSE}(\hat{v}^s_{i,k}, \mathrm{sg}(\hat{v}^t_{i,k}))$. We do not divide out σ_k^2; consequently, later, lower-noise steps receive less relative weight than under velocity matching. Targets remain teacher predictions, with real videos used only as cache context. This differs from SFT, which directly fits real-data/noise interpolations.

C.2 First-Step Only

We apply velocity MSE only at the initial, highest-noise step of each supervised chunk. All four denoising steps still execute, and both caches are updated normally. The loss is averaged over the supervised pairs: for $N = 60$, $M = 7$, and $K = 4$, this gives a denominator of $N - M = 53$, compared with $K(N - M) = 212$ for the full objective. Unsupervised steps are excluded from the average, rather than counted as zero-valued losses.

C.3 Student-Latest KV

This variant changes only the source of the latest teacher-cache chunk. The full method encodes the teacher's clean estimate $\hat{x}^t_i$, evaluated at the student's final noisy state; the ablation instead encodes the student's completed chunk $\hat{x}^s_i$. Older history still uses real-video context, and the teacher continues to supervise every student denoising step. Thus, the comparison isolates the value of the teacher's latest prediction for subsequent continuation.

Across these three ablations, the full method achieves higher Quality and Dynamic Degree, while the variants obtain higher Semantic scores (Tab. 2). Fig. 5 illustrates the corresponding differences in local detail.

C.4 On-Policy Trajectory Alignment

We also let the teacher follow its own denoising trajectory $z^t_{i,k}$ instead of evaluating it at the student's state $z^s_{i,k}$. The branches share the initial Gaussian noise and transition-noise draws, but each advances from its own clean prediction. Their noisy states can therefore diverge, so training matches outputs at the same chunk and timestep but at different latents.

In Fig. 6, the teacher retains clear structure while the student becomes blurred. This qualitative failure shows why cleaner teacher samples alone may be insufficient: they need not provide the correction required at the student's current state. It supports evaluating the teacher on student-visited states, as in Eq. (6).

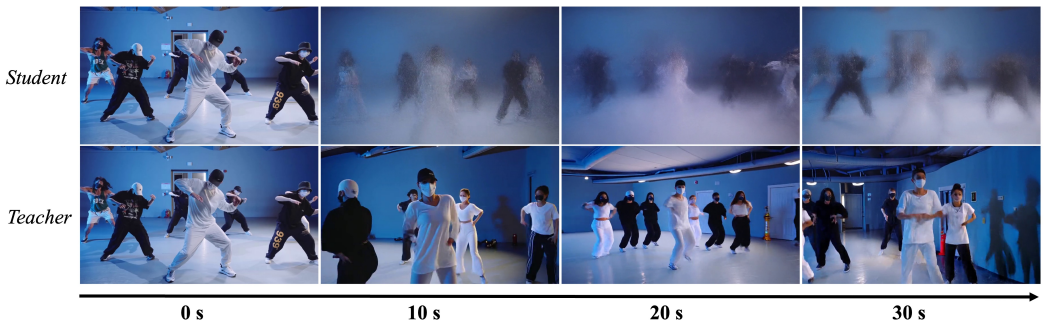

Figure 6: **Separate-trajectory diagnostic.** When the teacher follows its own denoising trajectory, its outputs remain clear (bottom), but the student rollout becomes blurred (top).

C.5 Training-Free Cache Intervention

For the original LongLive model, replacing older generated KV history with real-video features improves continuation without post-training (Fig. 1). The latest generated chunk is retained, and both runs share the same real prefix, prompt, and sampler.

Replacing the latest chunk's KV features as well can introduce ghosting and abrupt changes at chunk boundaries (Fig. 7), where real-video context may not match the preceding generated output. Keeping the latest generated chunk gives smoother transitions, supporting its role in maintaining continuity. Five matched video comparisons are available on our project page.

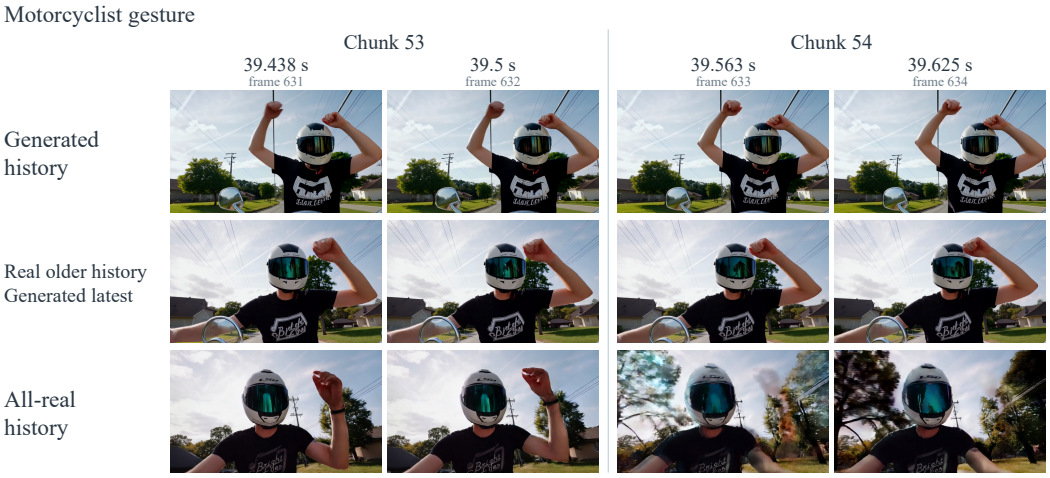

Figure 7: **Training-free ablation of the latest history chunk.** Four consecutive frames around a chunk boundary (39.56 s), with the same LongLive model, real prefix, prompt, and seed. Replacing the latest chunk's KV features with real-video context causes ghosting and abrupt changes; retaining the latest generated chunk gives a smoother transition.

D Analysis and Future Work

Our study examines the source of supervision for post-training few-step autoregressive video generators. DMD-based approaches obtain guidance from a pretrained teacher, whose temporal coverage and generation quality can limit its usefulness on longer student rollouts. Real long videos offer another source of temporal structure, but direct training on real-data/noise interpolations or fully real-video histories can introduce a mismatch with the distilled generator's sampling states. OPSD-V uses real long videos as privileged teacher context while retaining the student's own denoising trajectory and generated history. This separates the source of contextual information from the states at which corrective supervision is applied.

The current results establish this approach on two few-step AR backbones and a modest long-video dataset. Increasing the amount, diversity, and quality of training data, together with training compute, is a natural direction for further study. Better teacher-cache policies, adaptive supervision schedules, and stronger context-conditioned teachers may also improve the reliability of the targets. A remaining challenge is that a generated trajectory can diverge from its paired real video: the shared prefix and retained latest prediction help maintain continuity but do not ensure semantic or motion alignment. Understanding and controlling this mismatch could further improve data-driven post-training for causal long-video generation.

E Comparison with Self Gradient Forcing

We compare Self-Forcing + OPSD-V with the official chunkwise EMA release of Self Gradient Forcing (SGF) (Zhuang et al., 2026). We retain SGF's native four-step sampler and long-video cache policy: 3 sink frames, 6 recent historical frames, and 3 current frames in latent space, with top-aligned relative positions. We use chunkwise SGF's released checkpoint based on CausalForcing. Fig. 8 shows three paired examples. Late SGF frames contain elongated village roofs, repeated cliff textures, and an arch-like lunar background. OPSD-V retains more recognizable roofs and foliage, clearer landscape surfaces, and distinct astronaut poses in the sampled later views. Additional paired videos are available on the anonymous project page.

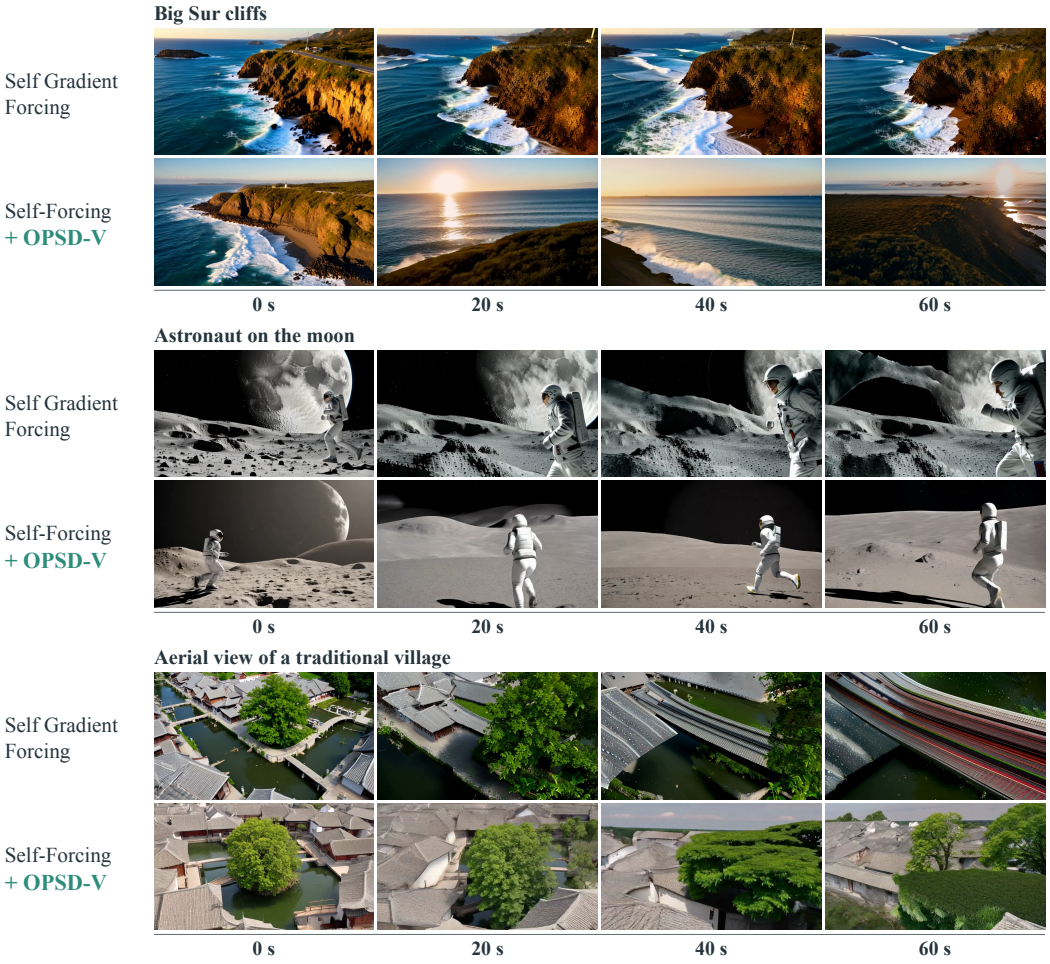

Figure 8: **Qualitative comparison with Self Gradient Forcing.** Each example shows SGF (chunkwise EMA, top) and Self-Forcing + OPSD-V (bottom) for the same prompt, at 0, 20, 40, and 60 seconds. The first two examples are from MovieGenBench and the third from MeiBench.