MEMORY-GATED DISTILLATION: MITIGATING SELF-GENERATED MEMORY DRIFT IN AUTOREGRESSIVE DIFFUSION WORLD MODEL DISTILLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Autoregressive diffusion world models enable interactive long-horizon video generation, but their sampling cost is incurred at every generated chunk, making low-latency deployment difficult. Few-step distillation from a bidirectional teacher to an autoregressive student is a promising remedy, but rollout distillation becomes unstable over long horizons. We identify self-generated memory drift as a key cause of this instability: early student errors are written into memory and reused as conditioning context, degrading subsequent distillation. We introduce Memory-Gated Distillation (MGD), which addresses this error-accumulation problem through selective memory supervision. MGD uses a tailored chunk-level reliability verifier to decide whether each student-generated chunk should be written into memory, thereby preserving reliable rollout prefixes while filtering corrupted ones. As a result, MGD retains the benefits of on-policy student rollouts while suppressing supervision from corrupted memory contexts, with no additional inference-time cost. Experiments on short- and long-horizon generation demonstrate improved few-step generation quality and a stronger quality-latency tradeoff than prior distillation baselines. Particularly in the challenging 253-frame rollout setting, 4-step MGD nearly matches the 50-step bidirectional teacher on aesthetic quality (0.564 vs. 0.569) and PSNR (13.81 vs. 13.84) with less than 1/10 of sampling time.

1 INTRODUCTION

Autoregressive diffusion models have become a compelling foundation for interactive world modeling, combining the controllability of causal generation with the visual fidelity of iterative denoising. Recent systems demonstrate near-real-time response to prompts, actions, and camera trajectories while sustaining geometric and temporal consistency over long horizons, narrowing the gap between video generators and neural simulators (Sun et al., 2025; Che et al., 2025; Yu et al., 2025; Yang et al., 2025). Realizing this potential at deployment scale, however, requires distilling such teachers into students that retain long-horizon behavior under a small number of denoising steps and a strict latency budget.

Distillation in this setting is fundamentally different from distillation of single-shot or short-horizon generators. An autoregressive world model is a closed-loop system: each generated chunk is written into a persistent memory state and fed back as conditioning for subsequent chunks, so the student determines its own future input distribution at both training and inference time. Closing the train-test gap therefore motivates training objectives that account for histories produced by the model itself. Recent autoregressive diffusion methods use next-token, rollout-aware, causal-forcing, or on-policy distillation objectives to reduce the mismatch between teacher-forced contexts and generated contexts (Chen et al., 2024; Huang et al., 2025c; Cui et al., 2026; Liu et al., 2026; Chen et al., 2026; Zhu et al., 2026; Yin et al., 2025). While effective at the chunk level, these methods implicitly assume that every generated history remains a valid context for downstream supervision once the rollout has begun, which is blind spot in the field.

We argue that this assumption breaks down precisely in long-horizon distillation. We identify a failure mode dubbed *student memory drift*: when the student emits a chunk with implausible appearance, geometry, or motion, the resulting memory state can move away from the conditioning distribution

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.

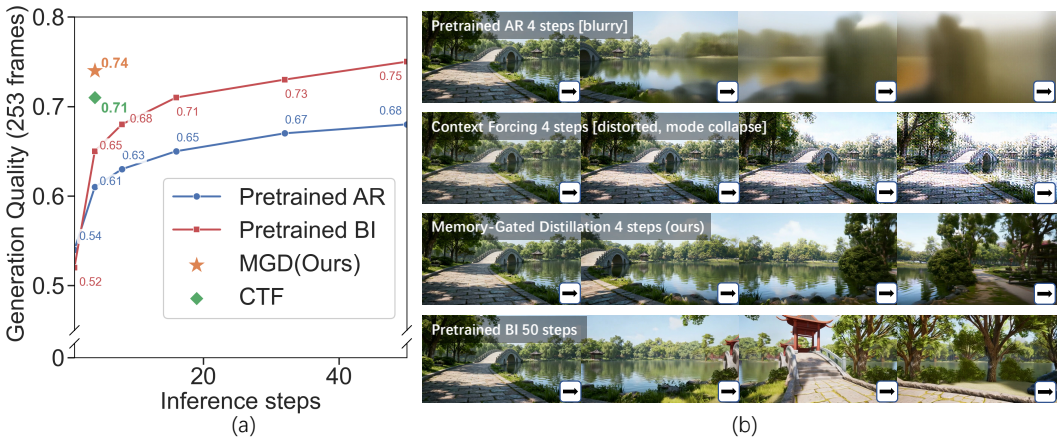

Figure 1: Memory-Gated Distillation (MGD) improves long-horizon generation under a small inference budget. **(a)** Quality and latency on 253-frame rollouts. The vertical axis reports Generation Quality, defined as the average of VBench subject consistency, background consistency, aesthetic quality, imaging quality, and $1 - \text{LPIPS}$. The pretrained bidirectional teacher improves with more denoising steps but remains expensive, while the pretrained autoregressive model gives lower quality at the same budget. With only four steps, MGD reaches a higher-quality region than Context Forcing (CTF). **(b)** Qualitative comparison on the same input image, prompt, and camera control. The pretrained autoregressive model with four steps becomes blurry over long rollouts. Context Forcing preserves the initial view but accumulates distortion and collapses toward nearly static frames. MGD maintains sharper appearance, scene geometry, and camera motion, producing a trajectory closer to the 50-step bidirectional teacher.

under which the teacher's score and the distribution-matching objective built on it remain meaningful. Subsequent gradients are then computed under a corrupted context, so the teacher score may not provide a useful direction and propagate noise rather than signal. This is distinct from teacher-side distribution mismatch (Ning et al., 2024; Yi et al., 2024; Starodubcev et al., 2026) and from generic exposure bias with unexposed self-generated memory in training (Bengio et al., 2015). Self-generated memory drift instead concerns the instability caused by exposed yet corrupted self-generated memory. To our knowledge, this insight has not been considered explicitly in prior distillation work, which uniformly weights every prefix of a student rollout.

This perspective motivates supervision that is *selective* rather than unconditional. Rather than backpropagating through every chunk of an on-policy rollout, training should determine, at each step, whether the current memory is still a faithful sample from the conditioning distribution over which the student is being optimized. We realize this principle with **Memory-Gated Distillation (MGD)**, a framework that couples an autoregressive student with a chunk-level verifier and a gated distribution-matching objective. The verifier consumes the same conditioning inputs the student uses at inference together with the candidate chunk, and predicts whether that chunk is reliable enough to be admitted into future memory. Gate decisions are applied causally from left to right and aggregated into a monotonic mask, yielding a contiguous verified prefix on which the distillation gradient is computed and beyond which gradients are suppressed. The construction adds no inference-time overhead and requires no architectural change to the student. Therefore, our method is widely applicable to existing autoregressive world models and even compatible with prior solutions for teacher distribution mismatch.

We instantiate MGD on top of a large-scale autoregressive world model, using HY-World 1.5 (Sun et al., 2025) as the bidirectional teacher and a LoRA-adapted (Hu et al., 2022) HY-World 1.5 AR student trained jointly with a LoRA-adapted fake bidirectional model that tracks the evolving student distribution. We evaluate the resulting student at both 61-frame and 253-frame horizons, using VBench metrics (Huang et al., 2024) together with geometry-aware camera measures derived from ViPE trajectories (Huang et al., 2025a). Figure 1 previews the resulting behavior at the 253-frame horizon: under a 4-step inference budget, MGD attains metrics comparable to the 50-step HY-World

1.5 baseline, corresponding to more than a $10\times$ speedup, while avoiding both the blurring of the unmodified 4-step student and the geometric drift of strong on-policy baselines. The contributions of this paper are as follows. (i) We identify and formalize student memory drift as a distinct failure mode of distillation for autoregressive world models with persistent memory, separating it from teacher-side drift and exposure bias. (ii) We propose MGD, a chunk-level verifier and gated DMD objective that supervises only the verified prefix of each student rollout, providing a principled and efficient mechanism for selective on-policy distillation. (iii) We design an evaluation protocol that probes performance across horizons, supervision ratios, and the quality–efficiency frontier, and show that MGD compares favorably with strong distillation baselines while preserving the inference speed required for interactive deployment.

2 Related work

Video generation and interactive world models. Diffusion models have driven rapid progress in image and video synthesis, evolving from pixel-space and latent denoisers (Ho et al., 2020; Song et al., 2021; Rombach et al., 2022) to large-scale text-to-video and image-to-video generators with strong motion fidelity (Ho et al., 2022b;a; Hong et al., 2023; Blattmann et al., 2023a;b). To support long horizons and online generation, a parallel line of work develops autoregressive and streaming video models that produce frames or chunks sequentially, often combining causal attention or hybrid AR-diffusion designs with streaming inference (Chen et al., 2025a; HaCohen et al., 2024; Deng et al., 2024; Gao et al., 2024; Gu et al., 2025; Xie et al., 2024; Chen et al., 2025b; Zhang et al., 2025d; Kim et al., 2025). Related autoregressive diffusion transformer designs have also been explored beyond video generation (Hu et al., 2024; Liu et al., 2024). World-modeling research extends this trajectory by coupling generation with controllable dynamics and planning (Ha & Schmidhuber, 2018; Hafner et al., 2019; 2020; 2023), and recent interactive or embodied systems respond to prompts, actions, or camera trajectories over long horizons (Yang et al., 2024; Che et al., 2025; Yu et al., 2025; Sun et al., 2025; Yang et al., 2025; Hong et al., 2025; Huang et al., 2026; Zhang et al., 2025a; Shang et al., 2025; Li et al., 2025b). Sustaining quality across such horizons requires generators to consume and update their own history reliably, motivating memory and long-context mechanisms (Li et al., 2025a; Xiao et al., 2025; Huang et al., 2025b; Hong et al., 2025; Po et al., 2025; Guo et al., 2025; Zhang et al., 2025b; Song et al., 2025).

Diffusion distillation methods. Diffusion distillation has progressed from sampler compression to distribution-level alignment. Early step-reduction and consistency approaches shorten the sampling trajectory by matching teacher trajectories or enforcing invariance along the probability-flow path (Salimans & Ho, 2022; Song et al., 2023; Luo et al., 2023a). Later score- and divergence-based objectives match the student distribution to teacher scores, while adversarial variants provide complementary alignment when score estimates are unreliable (Luo et al., 2023b; Yin et al., 2024b;a; Zhou et al., 2024; Sauer et al., 2024; Lin et al., 2024; 2025; Lu et al., 2025). For video and world models, the key challenge is not only reducing denoising steps, but also training under causal rollouts. Recent methods combine diffusion objectives with autoregressive, context-aware, causal-forcing, hierarchical-denoising, or on-policy training (Chen et al., 2024; Yin et al., 2025; Li et al., 2024; Huang et al., 2025c; Cui et al., 2026; Liu et al., 2026; Chen et al., 2026; Zhu et al., 2026; Zou et al., 2026; Li et al., 2026; Wang et al., 2025; Ye et al., 2026; Kim et al., 2025). These methods reduce train-test context mismatch or decompose long-horizon distillation, but generally do not ask which generated suffixes remain reliable enough for score matching. MGD is complementary: it keeps the backbone fixed and gates unreliable rollout supervision.

Distributional drift in distillation. Drift in diffusion distillation has mostly been studied at the sample or teacher-query level. Low-quality student samples place noised latents outside the regime with the reliable pretrained teacher score, motivating noise reweighting, timestep curricula, score-identity objectives, adversarial alignment, improved DMD variants, distribution backtracking, and scale-wise decompositions (Ning et al., 2024; Hang et al., 2023; Yi et al., 2024; Zhou et al., 2024; Sauer et al., 2024; Lin et al., 2024; Lu et al., 2025; Yin et al., 2024a; Zhang et al., 2025c; Starodubcev et al., 2026). These methods improve score supervision for individual samples. Autoregressive world models introduce an additional memory-level drift: a generated chunk is written into persistent state and changes the conditioning distribution for all later chunks. Recent memory-architecture and rollout-stabilization approaches address long-horizon degradation through explicit memory management, context-aware routing, hybrid attention, hierarchical denoising or alternative training

objectives (Hong et al., 2025; Huang et al., 2026; Shang et al., 2025; Zhu et al., 2026; Zou et al., 2026; Li et al., 2026; Wang et al., 2025). MGD instead targets memory-level drift by estimating whether each generated chunk should remain in the supervised prefix, making it orthogonal to architectural memory designs and teacher-side remedies.

3 PRELIMINARIES

Autoregressive diffusion world model. We consider a world model that generates a video as a sequence of K chunks $x_{1:K}$, each consisting of a fixed number of frames. A rollout is conditioned on an initial image y, a prompt embedding p, and a sequence of per-chunk control inputs $u_{1:K}$ (e.g., camera trajectory). Together with a memory state m_k summarizing previously generated chunks, these define the conditioning at step k,

$$c_k = (y, p, u_k, m_k). \tag{1}$$

The autoregressive student S_θ predicts the next chunk and updates memory according to

$$\hat{x}_k = S_\theta(c_k), \qquad m_{k+1} = \mathcal{M}(m_k, \hat{x}_k), \tag{2}$$

so the student determines its own future conditioning distribution: errors made at step k are written into m_{k+1} and persist throughout the remainder of the rollout.

Distribution matching distillation. We follow DMD-style distillation (Yin et al., 2024b;a; 2025), which trains a few-step student through a score-gap gradient. The framework uses a frozen *real* bidirectional denoiser B^{real} to estimate the teacher score, and a trainable *fake* denoiser B^{fake}_ψ to track the current student score. Here we adopt the noise-prediction schema, in which a denoiser predicts the residual ϵ. For a generated chunk $\hat{x}_k = S_\theta(c_k)$, sample a diffusion time t and Gaussian noise ϵ, and form

$$\tilde{x}_k = q_t(\hat{x}_k, \epsilon) = \alpha_t \hat{x}_k + \sigma_t \epsilon. \tag{3}$$

Here (α_t, σ_t) are standard signal/noise schedule coefficients. The real and fake denoisers predict:

$$\epsilon^{\text{real}}_k = B^{\text{real}}(\tilde{x}_k, t, c_k), \qquad \epsilon^{\text{fake}}_k = B^{\text{fake}}_\psi(\tilde{x}_k, t, c_k). \tag{4}$$

Under the noise-prediction parameterization, their difference is proportional to the score gap between the student and teacher distributions, yielding the surrogate gradient

$$\nabla_\theta \ell^{\text{DMD}}_k \;=\; \mathbb{E}_{t,\epsilon}\big[J_\theta(\hat{x}_k)^\top w(t)\big(\epsilon^{\text{fake}}_k - \epsilon^{\text{real}}_k\big)\big]. \tag{5}$$

Here $J_\theta(\hat{x}_k)$ is the Jacobian of the student output to θ, $w(t)$ is the timestep weight and the expectation is taken over uniform t and standard Gaussian. Thus B^{real} pulls the student toward the teacher distribution, while B^{fake}_ψ subtracts the score of the distribution already produced by the student. Since this score changes with S_θ, training alternates between fitting B^{fake}_ψ on student rollouts and updating S_θ with Eq. (5). The naive autoregressive rollout objective applies this update to every chunk,

$$\mathcal{L}_{\text{roll}} = \sum_{k=1}^{K} \ell^{\text{DMD}}_k\big(\tilde{x}_k, c_k;\, B^{\text{real}}, B^{\text{fake}}_\psi\big). \tag{6}$$

This implicitly treats every student-induced memory state m_k as a valid context for the score-gap surrogate, regardless of whether m_k drifts from the conditioning distribution.

Reference and student memory distributions. Let $p_{\text{ref}}(m)$ be the distribution of memory states the student is implicitly trained to condition on, namely those induced by teacher-generated or otherwise plausible rollouts. Let $p_\theta(m)$ be the distribution of memory states induced by the current student. Memory drift denotes the regime in which the support of p_θ moves away from that of p_{ref}. Thus, the teachers are queried by chunks that fall outside the conditioning on which they were trained or aligned. The objective developed in Section 4 replaces $\mathcal{L}_{\text{roll}}$ with a masked variant that supervises only the prefix of a rollout for which m_k remains in $\text{supp}(p_{\text{ref}})$.

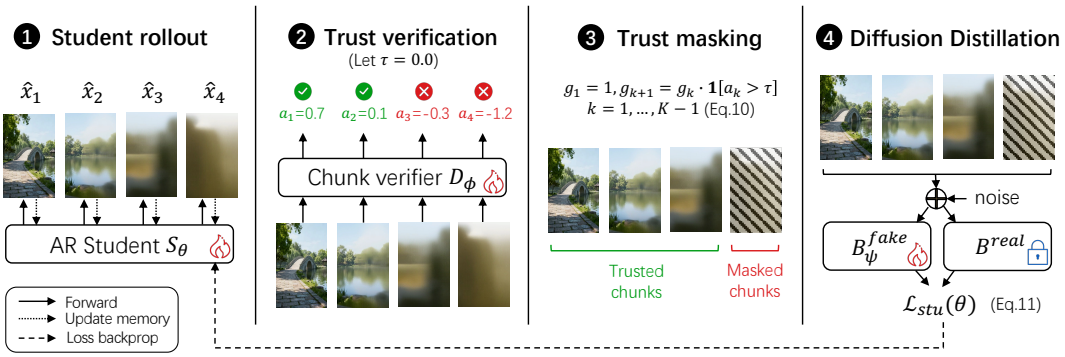

Figure 2: Overview of student update in Memory-Gated Distillation. **(1) Student rollout**: the autoregressive student S_θ generates a sequence of chunks $\hat{x}_{1:K}$, each written into the memory state used to condition the next step. **(2) Reliability verification**: the chunk verifier D_ϕ scores every generated chunk under the same conditioning the student sees at inference, producing logits a_k. Here the threshold $\tau = 0$ is used as example. **(3) Gate masking**: the per-chunk decisions are aggregated causally via Eq. (10) into a monotonic mask $g_{1:K}$, which retains a contiguous verified prefix and discards the suffix once the first rejected chunk is detected. **(4) Diffusion Distillation**: the gated distribution matching objective in Eq. (11) is computed against the frozen real teacher B^{real} and the tracking fake model B_ψ^{fake}, and gradients are backpropagated to S_θ only through the verified prefix.

4 METHOD

Autoregressive distillation differs from single-shot distillation because each student-generated chunk is written into memory and becomes part of the conditioning state for later chunks. Once an early chunk contains implausible appearance, geometry, or motion, the resulting memory can drift away from the reference distribution, so continuing to apply the same rollout loss on the corrupted suffix may amplify unreliable supervision rather than improve the student. Memory-Gated Distillation (MGD) follows a single design principle to supervise a chunk only when the rollout's memory state remains plausible under p_{ref}. Concretely, MGD replaces $\mathcal{L}_{\text{roll}}$ with a selective variant. Three components implement this principle, illustrated in Figure 2: (i) a chunk-level reliability verifier D_ϕ that estimates, from the same inputs the student sees at inference, whether the current chunk should be admitted into future memory; (ii) a causal gate mask $g_{1:K}$ that turns verifier decisions into a contiguous supervised prefix of the rollout; and (iii) an alternating training loop in which B_ψ^{fake}, D_ϕ, and S_θ are updated in turn, with B^{real} frozen. Algorithm 1 in Appendix B gives the complete training procedure. At inference, only S_θ is retained; B_ψ^{fake} and D_ϕ are discarded, so the construction adds no test-time cost.

4.1 SELECTIVE SUPERVISION UNDER MEMORY DRIFT

Recall that $m_{k+1} = \mathcal{M}(m_k, \hat{x}_k)$ is causally determined by the student's chunk at step k. Consequently, the validity of the per-chunk loss ℓ_{k+1}^{DMD} depends on whether the induced memory state lies in the support of the reference distribution p_{ref}. We capture this with the idealized indicator

$$z_k^\star = \mathbf{1}\big[\, m_{k+1} \in \text{supp}(p_{\text{ref}}) \,\big], \qquad k = 1, \ldots, K-1, \tag{7}$$

which would, if observable, certify that supervising chunk $k+1$ queries the teachers under a context drawn from the conditioning distribution they were trained on. Two consequences follow from the autoregressive structure. First, $z_k^\star$ acts on the next chunk: chunk k has been generated under memory m_k, and its quality determines whether chunk $k+1$ should inherit the updated memory m_{k+1}. Second, we enforce a monotone approximation to validity: after a chunk is rejected by the gate, later contexts are treated as unreliable because subsequent memory updates are conditioned on that chunk. The set of supervised positions is therefore constrained to be a contiguous prefix of the rollout. MGD approximates $z_k^\star$ from the local pair $(c_k, \hat{x}_k)$, which is precisely the evidence available at inference time when m_{k+1} is about to be committed. In summary, the mask acts as a learned indicator for the prefix of the rollout on which the DMD objective remains a faithful score-matching surrogate.

4.2 CHUNK-LEVEL GATING VERIFIER

The verifier is a generic gating function that scores whether a candidate chunk should be written into memory. In principle, this function can take several forms. A simple choice is a hand-designed criterion, such as a distance between the generated chunk and a ground-truth or teacher reference chunk. A more adaptive choice is a trainable discriminator that learns to separate reliable chunks from student-generated chunks under the same conditioning context. MGD only requires the verifier to return a scalar gate score. The rest of the method uses this score to form the causal mask, so any future replacement that produces a per-chunk reliability score (such as an exchangeable density-ratio estimator, a calibrated rejection sampler) can be plugged in without alternating the training loop.

In our main implementation, we use a learned verifier $D_\phi(c_k, x) \in \mathbb{R}$ that scores a candidate chunk x under the same conditioning tuple $c_k = (y, p, u_k, m_k)$ used by the student. Matching the student's interface is important: the verifier's decision is conditioned on exactly the inputs that determine m_{k+1}, and it avoids relying on future frames or other signals unavailable when the next memory state is committed. We ablate alternative verifier choices in Section 5, and find that the learned discriminator gives the best long-horizon quality, so it is used by default.

The learned verifier is trained as a binary classifier to distinguish teacher-generated chunks from student-generated chunks under shared conditioning. For each rollout step k, we form a positive sample by drawing the corresponding teacher chunk x_k^{real} from B^{real} under c_k, and a negative from the current student rollout, $\hat{x}_k = S_\theta(c_k)$. The two logits and the minimization of the verifier are:

$$a_k^{\text{real}} = D_\phi(c_k, x_k^{\text{real}}), \qquad a_k^{\text{stu}} = D_\phi(c_k, \hat{x}_k), \tag{8}$$

$$\mathcal{L}_D \;=\; -\sum_{k=1}^{K} \Big[\log \sigma(a_k^{\text{real}}) + \log\big(1 - \sigma(a_k^{\text{stu}})\big) \Big]. \tag{9}$$

Because positives and negatives share the conditioning tuple and differ only in the candidate chunk, D_ϕ learns to separate reference-aligned from student-generated continuations of the *same* memory state, which aligns its decision with the geometry of p_{ref} rather than with chunk quality in isolation. Notice that although this module is discriminator-like, it is not used as a GAN discriminator to give adversarial gradients. Its score is used only to decide the binary gate mask. We do not backpropagate an adversarial loss from D_ϕ into S_θ with stop-gradient operation; student gradients come only from the gated distillation objective in Eq. (11). Appendix C formalizes why this discriminator-style verifier promotes teacher–student distributional agreement without conflicting with the DMD objective, and Appendix D.2 provides a detailed discussion of its distinction from GAN discriminators.

4.3 MEMORY-GATED DISTILLATION OBJECTIVE

At each student update, the verifier is applied causally over the rollout. Let $a_k = a_k^{\text{stu}}$ and let $\tau \in \mathbb{R}$ be a fixed gate threshold. We define the causal gate mask $g_{1:K} \in \{0, 1\}^K$ by

$$g_1 = 1, \qquad g_{k+1} \;=\; g_k \cdot \mathbf{1}[\, a_k > \tau \,], \quad k = 1, \ldots, K-1. \tag{10}$$

Intuitively, the recursion is monotone non-increasing in k: $g_{k+1} \le g_k$ by construction, so MGD distills only on the longest reliable prefix of each rollout. Two properties of (10) mirror the structural requirements identified in Section 4.1. (a) *Contiguity*: the support of g is a prefix $\{1, \ldots, K^\star\}$ with $K^\star$ determined by the first failed test. (b) *Anchoring*: $g_1 \equiv 1$ guarantees that at least one chunk receives gradient at every step, preventing the supervision ratio from collapsing to zero before the student is competent. The student is then trained with gated DMD objective

$$\mathcal{L}_{\text{stu}}(\theta) \;=\; \sum_{k=1}^{K} g_k \cdot \ell_k^{\text{DMD}}\big(\tilde{x}_k,\, c_k;\, B^{\text{real}}, B_\psi^{\text{fake}}\big), \tag{11}$$

with $\tilde{x}_k = q_t(\hat{x}_k, \epsilon)$ as in Section 3. Comparing (11) with $\mathcal{L}_{\text{roll}}$ exposes the operative difference: a chunk is masked not because it is itself low quality, but because an earlier chunk has already pushed the memory state away from p_{ref}. The mask therefore acts as a learned indicator for the prefix of the rollout on which the DMD objective remains a faithful score-matching surrogate.

4.4 ALTERNATING OPTIMIZATION

We hold B^{real} fixed throughout training and update B_ψ^{fake}, D_ϕ, and S_θ in turn. In this design, the auxiliary B_ψ^{fake} is encouraged to cover the full student distribution (including its failure modes) for DMD gap to remain valid, while the student S_θ must avoid corrupted contexts to learn stably. Given a freshly sampled student rollout $\hat{x}_{1:K}$ and the corresponding teacher rollout $x_{1:K}^{\text{real}}$:

1. **Update** B_ψ^{fake} with the standard flow-matching loss on the entire student rollout,

$$\mathcal{L}_{\text{fake}}(\psi) \;=\; \sum_{k=1}^{K} \ell_k^{\text{FM}}\big(\tilde{x}_k,\, c_k;\, B_\psi^{\text{fake}}\big). \tag{12}$$

   The fake model must track the full distribution of S_θ, including its current failure modes, so this step is intentionally *not* gated. Applying (10) here would have two adverse effects. First, it would bias B_ψ^{fake} toward the verified prefix and degrade the score gap that drives DMD. Second, the bidirectional teachers B^{real} and B_ψ^{fake} are pretrained to denoise latents of a fixed length. So shortening the supervised window by dropping suffix chunks would change the effective input shape, drive the score estimates outside the regime they were trained on, and yield unreliable gradients. Computing $\mathcal{L}_{\text{fake}}$ over the full rollout therefore preserves both coverage of the student rollout distribution and the operating regime of the bidirectional models.
2. **Update** D_ϕ by minimizing $\mathcal{L}_D$ on $(x_k^{\text{real}}, \hat{x}_k)$ pairs sharing c_k, as in Section 4.2.
3. **Update** S_θ with B^{real}, B_ψ^{fake}, and D_ϕ frozen: compute the gate mask via (10) and minimize the masked objective (11).

This schedule exposes a clean separation of roles. B_ψ^{fake} tracks the full student rollout distribution, D_ϕ estimates whether each candidate chunk keeps the induced memory near p_{ref}, and S_θ receives DMD updates only on prefixes where the score gap is well-defined. Because B_ψ^{fake} and D_ϕ are training-time components, the deployed model has the same architecture and inference cost as the original autoregressive backbone, and MGD can be combined with existing remedies for teacher-side drift, such as timestep curricula or adversarial alignment, without modification.

5 EXPERIMENTS

5.1 SETUP

We evaluate three questions: whether MGD improves few-step autoregressive world-model distillation, whether the gains become larger at longer horizons, and how the gate threshold controls the supervised prefix. All experiments use HY-World 1.5 as the backbone (Sun et al., 2025) and train on a 7,000-sample subset of Sekai (Li et al., 2025c). The frozen real model is HY-World 1.5 BI. The fake bidirectional model is a LoRA adapter on top of HY-World 1.5 BI, and the student is a LoRA adapter on top of HY-World 1.5 AR (Hu et al., 2022). This keeps the pretrained backbones fixed and isolates the effect of the training objective. At inference, only the autoregressive student is used. Appendix A gives the training schedule and optimization details.

We compare with three baselines. HY-World 1.5 AR, 50-step represents the native high budget autoregressive model. HY-World 1.5 AR, 4-step tests whether simply reducing the number of denoising steps is sufficient. Context Forcing is the closest training baseline, since it also trains on student rollouts but keeps all subsequent supervision active once the rollout begins (Chen et al., 2026). MGD and all low budget baselines use the same 4-step inference schedule. Since the distilled student is trained as a fixed-step generator, we do not test the number of MGD inference steps in Figure 1 or in the experiments. Appendix D.1 discusses this protocol. We evaluate both 61-frame and 253-frame rollouts with identical image conditions, camera controls, prompts, and evaluation protocols.

We report the six VBench dimensions relevant to i2v generation: subject consistency, background consistency, motion smoothness, dynamic degree, aesthetic quality, and imaging quality (Huang et al., 2024). We also report PSNR, SSIM, LPIPS, R_{dist}, and T_{dist} following the HY-World 1.5 protocol, with camera distances computed from ViPE trajectories (Sun et al., 2025; Huang et al., 2025a). For the threshold ablation, Generation Quality follows the definition used in Figure 1.

Table 1: VBench results for short video generation with 61 frames and long with 253 frames.

Method	Subject consistency ↑	Background consistency ↑	Motion smoothness ↑	Dynamic degree ↑	Aesthetic quality ↑	Imaging quality ↑
61 frames						
HY-World BI, 50-step (teacher)	0.976	0.963	0.992	0.493	0.581	0.740
HY-World AR, 50-step	0.965	0.943	0.992	**0.552**	0.565	0.699
HY-World AR, 4-step	0.943	0.942	0.992	0.478	0.541	0.633
Context Forcing, 4-step	**0.989**	**0.963**	**0.996**	0.031	0.557	0.698
Ours (MGD), 4-step	0.984	0.955	**0.996**	0.357	**0.575**	**0.711**
253 frames						
HY-World BI, 50-step (teacher)	0.962	0.943	0.992	0.490	0.569	0.728
HY-World AR, 50-step	0.902	0.911	0.992	**0.610**	0.519	0.638
HY-World AR, 4-step	0.847	0.901	0.992	0.460	0.469	0.516
Context Forcing, 4-step	0.951	0.924	0.995	0.020	0.510	0.672
Ours (MGD), 4-step	**0.963**	**0.940**	**0.996**	0.350	**0.564**	**0.702**

Table 2: Reconstruction and camera-trajectory metrics for short video generation with 61 frames and long with 253 frames. Pose-based distances are computed from trajectories estimated by ViPE.

Method	61 frames					253 frames				
	PSNR ↑	SSIM ↑	LPIPS ↓	R_{dist} ↓	T_{dist} ↓	PSNR ↑	SSIM ↑	LPIPS ↓	R_{dist} ↓	T_{dist} ↓
HY-World BI, 50-step (teacher)	16.78	0.474	0.293	1.921	0.013	13.84	0.414	0.456	1.911	0.096
HY-World AR, 50-step	15.70	0.465	0.379	1.934	0.014	13.15	0.382	0.564	1.980	0.121
HY-World AR, 4-step	16.15	0.483	0.423	1.884	**0.013**	13.75	**0.409**	0.645	**1.855**	0.116
Context Forcing, 4-step	15.85	0.479	0.320	2.003	0.025	11.65	0.360	0.494	1.913	0.106
Ours (MGD), 4-step	**16.62**	**0.485**	**0.315**	**1.872**	**0.013**	**13.81**	0.385	**0.462**	1.881	**0.096**

5.2 Main results

The main results are shown in Tables 1 and 2. Table 1 reports VBench metrics, while Table 2 reports reconstruction and camera-trajectory metrics under the same 61-frame and 253-frame split.

MGD is the only 4-step method that approaches the 50-step bidirectional teacher on both short and long horizons. Its results are close to the BI teacher on most visual-quality and reconstruction dimensions, while using the same low-step budget as the accelerated baselines. The main contrast is with Context Forcing: CTF can obtain high consistency scores, but its extremely low dynamic degree indicates that this consistency often comes from suppressing motion and producing nearly static rollouts. MGD avoids this degenerate solution, preserving controlled motion while staying much closer to teacher-level appearance and geometry than the native 4-step student. The comparison with HY-World 1.5 AR further clarifies the quality-efficiency tradeoff. The 50-step AR model remains a strong high-budget baseline, especially for motion, but its cost is incompatible with the target low-latency setting. Simply reducing the AR model to 4 steps gives the desired speed but degrades long-horizon perceptual quality. MGD occupies a more useful operating point: it keeps the 4-step inference cost, tracks the BI teacher more closely than naive acceleration, and avoids the motion collapse of unconditional rollout supervision. This supports the central claim that selective supervision, rather than more aggressive rollout fitting, is needed for stable long-horizon distillation.

5.3 Ablations and analysis

We ablate two design choices: how the verifier is instantiated, and how strict the gate threshold should be. Table 3 compares verifier choices. The no-verifier setting is equivalent to Context Forcing. The reference-distance gate accepts a chunk when its distance to a ground truth or teacher reference chunk is below a fixed threshold. The learned discriminator is the default MGD verifier described in

Table 3: Verifier choice ablation. Generation Quality follows the definition of the Pareto vertical axis in Figure 1. The learned discriminator is the default verifier used in the main experiments.

Verifier	Avg. selected chunks	61f Generation Quality ↑	253f Generation Quality ↑
No verifier (Context Forcing)	4.0	<u>0.776</u>	<u>0.712</u>
Reference-distance gate	2.1	0.632	0.590
Learned discriminator	2.6	**0.782**	**0.741**

Table 4: Threshold ablation. Generation Quality follows the definition of the Pareto vertical axis in Figure 1. "Avg. selected chunks" denotes the average number of contiguous chunks that remain active for student supervision. The two limiting cases correspond to Context Forcing and supervision of a single chunk.

Threshold	Regime	Avg. selected chunks	61f Generation Quality ↑	253f Generation Quality ↑
$\tau \to -\infty$	Context Forcing limit	4.0	0.776	0.712
$\tau = -1.5$	Weak gating	3.5	<u>0.780</u>	<u>0.733</u>
$\tau = -0.5$	Moderate gating	2.6	**0.782**	**0.741**
$\tau = 0.5$	Strong gating	1.7	0.775	0.704
$\tau \to +\infty$	Single-chunk limit	1.0	0.779	0.638

Section 4.2. All variants are used only to build the gate mask and do not send adversarial gradients to the student.

The verifier ablation shows that the learned gate is most valuable when memory errors have time to accumulate. Without a verifier, training assumes that every on-policy suffix remains useful, which is reasonable for short rollouts but increasingly risky at long horizons. The reference-distance gate has the opposite problem: it rejects many plausible continuations because controllable generation is not tied to a single ground-truth future, and low-level distance is a poor proxy for whether a chunk can serve as reliable memory. The learned verifier provides a better reliability signal, removing drifted suffixes while retaining enough on-policy supervision for the student.

The threshold ablation in Table 4 tests whether the gate should be selective. When $\tau \to -\infty$, every chunk is accepted and the method reduces to Context Forcing. When $\tau \to +\infty$, only the first chunk is guaranteed to receive gradients. The sweep is therefore a controlled path between unconditional rollout supervision and almost single chunk supervision.

The threshold sweep further shows that useful gating is neither always open nor overly strict. On short rollouts, the threshold matters less because memory corruption has fewer steps to compound. On long rollouts, however, the Context Forcing limit leaves corrupted suffixes active, while the single-chunk limit discards too much rollout supervision. The best regime is an intermediate one: it removes unreliable suffixes but still trains on a multi-chunk verified prefix. This supports the central design principle of MGD, namely that long-horizon distillation needs a balance between supervision reliability and on-policy coverage.

6 Conclusion

Few-step distillation of autoregressive diffusion world models is a closed-loop problem because every generated chunk becomes context for later predictions. We identify self-generated memory drift as a central failure mode: once an unreliable chunk enters memory, subsequent DMD updates are evaluated under corrupted conditioning. MGD addresses this issue with a chunk-level verifier and a causal gate that restrict supervision to a contiguous reliable prefix while leaving the underlying score-gap objective unchanged. The auxiliary verifier and fake model are discarded at inference. Experiments with HY-World 1.5 show that this selective supervision improves the 4-step quality–efficiency frontier, especially on 253-frame rollouts, without the motion collapse observed with unconditional rollout supervision. These results highlight the importance of validating self-generated state, not only sample-level scores, when distilling stateful generative models. Future work may explore calibrated adaptive gates, richer memory representations, and combinations with teacher-side score correction.

AI USE STATEMENT

A generative AI tool was used to polish writing. All AI-assisted changes were reviewed by the authors, and no experimental results or primary research evidence were generated by the tool during this formatting conversion. The authors take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

MGD is designed to reduce the inference cost of autoregressive diffusion world models. This can have positive effects when interactive simulation and long-horizon video generation are used for research, education, prototyping, embodied-agent evaluation, and creative tools. Lower latency can make such systems more accessible and can reduce the cost of repeated inference once a model has been trained. The method also avoids adding inference-time modules, which is relevant for deployment settings with strict latency budgets. The same efficiency gains can also increase the accessibility of high-quality synthetic video generation.

Potential risks include misleading or deceptive media, impersonation, spam, and synthetic content that is difficult for viewers to distinguish from real footage. Since MGD is a distillation and training-stabilization method rather than a content-safety method, it does not by itself prevent unsafe prompts, biased generations, privacy leakage from training data, or misuse of downstream generated videos. The distilled student may inherit limitations and biases from the pretrained backbone, the training data, and the evaluation domain. Responsible use should therefore pair efficient world-model distillation with safeguards outside the scope of the present algorithm. These include dataset governance and license compliance, provenance tracking or watermarking for generated videos, prompt and output moderation where appropriate, human review for high-stakes uses, and monitoring for misuse after deployment. Applications involving people, sensitive locations, or safety-critical decision making require particular care, because visual plausibility should not be treated as evidence of physical truth or policy reliability.

REPRODUCIBILITY STATEMENT

Section 5 specifies the datasets, baselines, horizons, metrics, and evaluation protocol. Appendix A provides the training configuration, optimization hyperparameters, verifier architecture, data and license details, and distributed-training setup; Appendix B gives the complete training procedure. Together, these sections document the information needed to reproduce the reported method and evaluations.

REFERENCES

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023a.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 22563–22575, 2023b.

Haoxuan Che, Xuanhua He, Quande Liu, Cheng Jin, and Hao Chen. Gamegen-x: Interactive open-world game video generation. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=8VG8tpPZhe.

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.

Feng Chen, Bohan Zhuang, and Qi Wu. Streaming video diffusion: Online video editing with diffusion models. In *2025 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, pp. 1–9. IEEE, 2025a.

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*, 2025b.

Shuo Chen, Cong Wei, Sun Sun, Ping Nie, Kai Zhou, Ge Zhang, Ming-Hsuan Yang, and Wenhu Chen. Context forcing: Consistent autoregressive video generation with long context. *arXiv preprint arXiv:2602.06028*, 2026.

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. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=DzvPiqh23f.

Chaorui Deng, Deyao Zhu, Kunchang Li, Shi Guang, and Haoqi Fan. Causal diffusion transformers for generative modeling. *arXiv preprint arXiv:2412.12095*, 2024.

Kaifeng Gao, Jiaxin Shi, Hanwang Zhang, Chunping Wang, Jun Xiao, and Long Chen. Ca2-vdm: Efficient autoregressive video diffusion model with causal generation and cache sharing. *arXiv preprint arXiv:2411.16375*, 2024.

Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

Yuchao Gu, Weijia Mao, and Mike Zheng Shou. Long-context autoregressive video modeling with next-frame prediction. *arXiv preprint arXiv:2503.19325*, 2025.

Yuwei Guo, Ceyuan Yang, Ziyan Yang, Zhibei Ma, Zhijie Lin, Zhenheng Yang, Dahua Lin, and Lu Jiang. Long context tuning for video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17281–17291, 2025.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2(3):440, 2018.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *International conference on machine learning*, pp. 2555–2565. PMLR, 2019.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=S1lOTC4tDS.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Tiankai Hang, Shuyang Gu, Chen Li, Jianmin Bao, Dong Chen, Han Hu, Xin Geng, and Baining Guo. Efficient diffusion training via min-snr weighting strategy. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 7441–7451, 2023.

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

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022a.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022b.

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=rB6TpjAuSRy.

Yicong Hong, Yiqun Mei, Chongjian Ge, Yiran Xu, Yang Zhou, Sai Bi, Yannick Hold-Geoffroy, Mike Roberts, Matthew Fisher, Eli Shechtman, et al. Relic: Interactive video world model with long-horizon memory. *arXiv preprint arXiv:2512.04040*, 2025.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Jinyi Hu, Shengding Hu, Yuxuan Song, Yufei Huang, Mingxuan Wang, Hao Zhou, Zhiyuan Liu, Wei-Ying Ma, and Maosong Sun. Acdit: Interpolating autoregressive conditional modeling and diffusion transformer. *arXiv preprint arXiv:2412.07720*, 2024.

Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. Vipe: Video pose engine for 3d geometric perception. *arXiv preprint arXiv:2508.10934*, 2025a.

Junchao Huang, Xinting Hu, Boyao Han, Shaoshuai Shi, Zhuotao Tian, Tianyu He, and Li Jiang. Memory forcing: Spatio-temporal memory for consistent scene generation on minecraft. *arXiv preprint arXiv:2510.03198*, 2025b.

Junchao Huang, Ziyang Ye, Xinting Hu, Tianyu He, Guiyu Zhang, Shaoshuai Shi, Jiang Bian, and Li Jiang. Live: Long-horizon interactive video world modeling. *arXiv preprint arXiv:2602.03747*, 2026.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. In *Advances in Neural Information Processing Systems*, 2025c. URL https://openreview.net/forum?id=mSiN7i0BYH. Spotlight.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21807–21818, 2024.

Yeongmin Kim, Sotiris Anagnostidis, Yuming Du, Edgar Schönfeld, Jonas Kohler, Markos Georgopoulos, Albert Pumarola, Ali Thabet, and Artsiom Sanakoyeu. Autoregressive distillation of diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15745–15756, 2025.

Jiachen Li, Weixi Feng, Tsu-Jui Fu, Xinyi Wang, Sugato Basu, Wenhu Chen, and William Y Wang. T2v-turbo: Breaking the quality bottleneck of video consistency model with mixed reward feedback. *Advances in neural information processing systems*, 37:75692–75726, 2024.

Ruibin Li, Tao Yang, Fangzhou Ai, Tianhe Wu, Shilei Wen, Bingyue Peng, and Lei Zhang. Long-horizon streaming video generation via hybrid attention with decoupled distillation. *arXiv preprint arXiv:2604.10103*, 2026.

Runjia Li, Philip Torr, Andrea Vedaldi, and Tomas Jakab. Vmem: Consistent interactive video scene generation with surfel-indexed view memory. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 25690–25699, 2025a.

Shuang Li, Yihuai Gao, Dorsa Sadigh, and Shuran Song. Unified video action model. In *Robotics: Science and Systems*, 2025b. URL https://doi.org/10.15607/RSS.2025.XXI.074.

Zhen Li, Chuanhao Li, Xiaofeng Mao, Shaoheng Lin, Ming Li, Shitian Zhao, Xinyue Li, Yukang Feng, Jianwen Sun, Zizhen Li, et al. Sekai: A video dataset towards world exploration. In *Advances in Neural Information Processing Systems*, 2025c. URL https://openreview.net/forum?id=I3F7w8MvY9. Datasets and Benchmarks Track.

Shanchuan Lin, Anran Wang, and Xiao Yang. Sdxl-lightning: Progressive adversarial diffusion distillation. *arXiv preprint arXiv:2402.13929*, 2024.

Shanchuan Lin, Xin Xia, Yuxi Ren, Ceyuan Yang, Xuefeng Xiao, and Lu Jiang. Diffusion adversarial post-training for one-step video generation. In *International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=AAgzsnhc28.

Kunhao Liu, Wenbo Hu, Jiale Xu, Ying Shan, and Shijian Lu. Rolling forcing: Autoregressive long video diffusion in real time. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=IAyzXjbfwo.

Zhijun Liu, Shuai Wang, Sho Inoue, Qibing Bai, and Haizhou Li. Autoregressive diffusion transformer for text-to-speech synthesis. *arXiv preprint arXiv:2406.05551*, 2024.

Yanzuo Lu, Yuxi Ren, Xin Xia, Shanchuan Lin, Xing Wang, Xuefeng Xiao, Andy J Ma, Xiaohua Xie, and Jian-Huang Lai. Adversarial distribution matching for diffusion distillation towards efficient image and video synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 16818–16829, 2025.

Simian Luo, Yiqin Tan, Suraj Patil, Daniel Gu, Patrick Von Platen, ApolinÃĄrio Passos, Longbo Huang, Jian Li, and Hang Zhao. Lcm-lora: A universal stable-diffusion acceleration module. *arXiv preprint arXiv:2311.05556*, 2023a.

Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. *Advances in Neural Information Processing Systems*, 36:76525–76546, 2023b.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pp. 3481–3490. PMLR, 2018.

Mang Ning, Mingxiao Li, Jianlin Su, Albert Ali Salah, and Itir Onal Ertugrul. Elucidating the exposure bias in diffusion models. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=xEJMoj1SpX.

Ryan Po, Yotam Nitzan, Richard Zhang, Berlin Chen, Tri Dao, Eli Shechtman, Gordon Wetzstein, and Xun Huang. Long-context state-space video world models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8733–8744, 2025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=TIdIXIpzhoI.

Axel Sauer, Kashyap Chitta, Jens Müller, and Andreas Geiger. Projected gans converge faster. *Advances in Neural Information Processing Systems*, 34:17480–17492, 2021.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. In *European Conference on Computer Vision*, pp. 87–103. Springer, 2024.

Yu Shang, Lei Jin, Yiding Ma, Xin Zhang, Chen Gao, Wei Wu, and Yong Li. Longscape: Advancing long-horizon embodied world models with context-aware moe. *arXiv preprint arXiv:2509.21790*, 2025.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=St1giarCHLP.

Kiwhan Song, Boyuan Chen, Max Simchowitz, Yilun Du, Russ Tedrake, and Vincent Sitzmann. History-guided video diffusion. In *International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=j8Vr3E3vhy.

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

Nikita Starodubcev, Ilya Drobyshevskiy, Denis Kuznedelev, Artem Babenko, and Dmitry Baranchuk. Scale-wise distillation of diffusion models. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Z06LNjqU1g.

Wenqiang Sun, Haiyu Zhang, Haoyuan Wang, Junta Wu, Zehan Wang, Zhenwei Wang, Yunhong Wang, Jun Zhang, Tengfei Wang, and Chunchao Guo. Worldplay: Towards long-term geometric consistency for real-time interactive world modeling. *arXiv preprint arXiv:2512.14614*, 2025.

Xiaodong Wang, Zhirong Wu, and Peixi Peng. Longdwm: Cross-granularity distillation for building a long-term driving world model. *arXiv preprint arXiv:2506.01546*, 2025.

Zeqi Xiao, Yushi Lan, Yifan Zhou, Wenqi Ouyang, Shuai Yang, Yanhong Zeng, and Xingang Pan. Worldmem: Long-term consistent world simulation with memory. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=c6CAVKlKmU.

Desai Xie, Zhan Xu, Yicong Hong, Hao Tan, Difan Liu, Feng Liu, Arie E. Kaufman, and Yang Zhou. Progressive autoregressive video diffusion models. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 6312–6322, 2024. URL https://api.semanticscholar.org/CorpusID:273234156.

Sherry Yang, Yilun Du, Kamyar Ghasemipour, Jonathan Tompson, Leslie Kaelbling, Dale Schuurmans, and Pieter Abbeel. Learning interactive real-world simulators. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sFyTZEqmUY. Oral.

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.

Tianzhu Ye, Li Dong, Xun Wu, Shaohan Huang, and Furu Wei. On-policy context distillation for language models. *arXiv preprint arXiv:2602.12275*, 2026.

Xuanyu Yi, Zike Wu, Qingshan Xu, Pan Zhou, Joo-Hwee Lim, and Hanwang Zhang. Diffusion time-step curriculum for one image to 3d generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9948–9958, 2024.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22963–22974, 2025.

Jiwen Yu, Yiran Qin, Xintao Wang, Pengfei Wan, Di Zhang, and Xihui Liu. Gamefactory: Creating new games with generative interactive videos. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11590–11599, 2025.

Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, et al. Epona: Autoregressive diffusion world model for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27220–27230, 2025a.

Lvmin Zhang, Shengqu Cai, Muyang Li, Gordon Wetzstein, and Maneesh Agrawala. Frame context packing and drift prevention in next-frame-prediction video diffusion models. In *Advances in Neural Information Processing Systems*, 2025b. URL https://openreview.net/forum?id=J8JCF64aEn. Spotlight.

Shengyuan Zhang, Ling Yang, Zejian Li, An Zhao, Chenye Meng, Changyuan Yang, Guang Yang, Zhiyuan Yang, and Lingyun Sun. Distribution backtracking builds a faster convergence trajectory for diffusion distillation. In *The Thirteenth International Conference on Learning Representations*, 2025c.

Yuan Zhang, Jiacheng Jiang, Guoqing Ma, Zhiying Lu, Haoyang Huang, Jianlong Yuan, Nan Duan, and Daxin Jiang. Generative pre-trained autoregressive diffusion transformer. *arXiv preprint arXiv:2505.07344*, 2025d.

Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in neural information processing systems*, 33:7559–7570, 2020.

Mingyuan Zhou, Huangjie Zheng, Zhendong Wang, Mingzhang Yin, and Hai Huang. Score identity distillation: Exponentially fast distillation of pretrained diffusion models for one-step generation. In *Forty-first International Conference on Machine Learning*, 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.

Kai Zou, Dian Zheng, Hongbo Liu, Tiankai Hang, Bin Liu, and Nenghai Yu. Hiar: Efficient autoregressive long video generation via hierarchical denoising. *arXiv preprint arXiv:2603.08703*, 2026.

A Additional experimental details

Training data. Training uses a 7,000-sample subset of Sekai, a first-person world-exploration video dataset with camera and scene annotations (Li et al., 2025c). All training samples are resized or cropped to 480×832.

Evaluation set and metric computation. All quantitative metrics are computed on our own Sekai test sets constructed for evaluation. For 61-frame generation, we generate 500 videos for each method; for 253-frame generation, we generate 100 videos for each method. We then compute the VBench dimensions, PSNR, SSIM, LPIPS, and ViPE-derived camera distances on these generated videos and report the average values.

Existing assets and licenses. We use existing models, data, and evaluation tools only under their public licenses or terms of use. HY-World 1.5 WorldPlay, including the autoregressive and bidirectional checkpoints used as backbones, is released under the Tencent HY-WorldPlay Community License Agreement; our use is for research fine-tuning and distillation on top of HY-World 1.5, and we do not redistribute the original checkpoints (Sun et al., 2025). The Sekai subset used for training is released on Hugging Face under the `fair-noncommercial-research-license`, and its accompanying code license states that the dataset and code are for non-commercial research purposes only; we use Sekai for research training and do not redistribute the underlying source videos (Li et al., 2025c). VBench evaluation code and benchmark assets are released under the Apache License 2.0, and are used only for automatic evaluation of generated videos (Huang et al., 2024). ViPE source code is released under the Apache License 2.0; we use ViPE only to estimate camera trajectories for evaluation, not to train on the released ViPE datasets (Huang et al., 2025a).

Training schedule and trainable modules. We train MGD directly for 10,000 training steps. The student is initialized from the HY-World 1.5 autoregressive model, and the fake-score model is initialized from the HY-World 1.5 bidirectional model (Sun et al., 2025). Both models keep the original HY-World 1.5 transformer architecture unchanged and are updated only through LoRA adapters. Thus, the student and fake-score networks have the same backbone capacity as their corresponding pretrained HY-World 1.5 models, and the difference between them lies in the causal or bidirectional generation interface and the trainable LoRA parameters.

MGD optimization. Training uses 4 NVIDIA A100 GPUs. We use LoRA training with rank 64 and alpha 64 for the student and fake-score adapters (Hu et al., 2022). The global batch configuration is batch size 1, sequence-parallel batch size 1, and one gradient accumulation step. We enable full gradient checkpointing and use bfloat16 mixed precision for the transformer backbones. The student learning rate is 3×10^{-5}, while the fake-score model and verifier use learning rate 10^{-4}. We use weight decay 10^{-4} and clip gradients to norm 1.0. The gate threshold is set to $\tau = -0.5$, matching the best setting in Table 4.

Verifier architecture. The learned verifier is a lightweight 3D convolutional network trained from scratch, with approximately 25M parameters. Its input is a noised candidate latent chunk with 33 channels, and its output is a scalar logit whose positive sign indicates a reference-aligned chunk. The verifier is conditioned on prompt embeddings, ByT5 text states, camera trajectory, action trajectory, and the initial image condition. For implementation simplicity, the verifier used in our experiments does not take the autoregressive memory latents as input, which is a slight simplification of the conceptual form $D_\phi(c_k, x)$ used in the main text.

The conditioning signals are first mapped to a shared 256-dimensional representation and summed. Timestep conditioning uses sinusoidal embedding followed by an MLP. Text and ByT5 states are encoded by masked mean pooling and MLP projections. Camera extrinsics and intrinsics are projected frame by frame and passed through temporal 1D convolutions before average pooling; the action trajectory is encoded in the same temporal manner. The image condition is encoded by a small 2D convolutional encoder over 16-channel image latents, using group normalization and global pooling. The resulting conditioning vector modulates a five-stage 3D convolutional backbone through FiLM layers. The backbone uses channel widths $(64, 128, 256, 256, 256)$ and strides $(1, 2, 2)$, $(1, 2, 2)$, $(2, 2, 2)$, $(2, 2, 2)$, and $(2, 2, 2)$, followed by global 3D average pooling and a two-layer classifier head. We initialize convolutional and linear layers with Kaiming initialization, initialize FiLM modulation to the identity transformation, and use Xavier initialization for the final classifier layer.

Reference positives and parallelism. Naively drawing positive verifier samples from a 50-step bidirectional teacher at every training iteration is prohibitively expensive. In the actual training implementation, we therefore use ground-truth chunks from the training set as reference-aligned positives for verifier training, replacing the multi-step samples that would otherwise be generated by HY-World 1.5 BI. This substitution affects only the positive examples used to train the verifier; the DMD score term still uses the frozen real bidirectional model as described in the main method. Distributed training uses sequence parallel size 4, tensor parallel size 1, HSDP replicate dimension 1, and HSDP shard dimension 4.

B Full training algorithm

Algorithm 1 expands the alternating optimization described in Section 4.4. The reference-positive sampler $\mathcal{R}$ returns the positive chunks used by the verifier. In the abstract method, these positives can be sampled from B^{real} under c_k; in our implementation, $\mathcal{R}$ returns the corresponding ground-truth chunks for efficiency, as described above.

C Distributional consistency of the MGD verifier

This section gives a population-level argument showing why the discriminator-style verifier used by MGD encourages the student distribution to approach the teacher or reference distribution, while remaining compatible with the DMD objective in Eq. (5). The argument is not intended to introduce

Algorithm 1 Memory-Gated Distillation (MGD) training procedure.

Require: Frozen real score model B^{real}; trainable student S_θ, fake score model B^{fake}_ψ, and verifier D_ϕ; dataset $\mathcal{D}$; reference-positive sampler $\mathcal{R}$; horizon K; threshold τ; noising process q_t in Eq. (3); total training steps N.

Ensure: Distilled autoregressive student S_θ.

for $n = 1, \dots, N$ **do**
  Sample $(y, p, u_{1:K}, x^{\text{gt}}_{1:K}) \sim \mathcal{D}$ and initialize memory m_1.
  Student rollout.
  for $k = 1, \dots, K$ **do**
    Form $c_k \leftarrow (y, p, u_k, m_k)$ as in Eq. (1).
    Generate $\hat{x}_k \leftarrow S_\theta(c_k)$ and update $m_{k+1} \leftarrow \mathcal{M}(m_k, \hat{x}_k)$ as in Eq. (2).
  end for
  Sample diffusion times and noises $\{(t_k, \epsilon_k)\}_{k=1}^K$, then set $\tilde{x}_k \leftarrow q_{t_k}(\hat{x}_k, \epsilon_k)$ for all k using Eq. (3).
  Obtain reference positives $x^{\text{real}}_k \leftarrow \mathcal{R}(c_k, x^{\text{gt}}_k, B^{\text{real}})$ for all k.
  Update the fake score model B^{fake}_ψ.
  Treat the rollout $(\hat{x}_{1:K}, c_{1:K})$ as fixed samples from the current student distribution.
  Compute the ungated fake-model objective $\mathcal{L}_{\text{fake}}(\psi) \leftarrow \sum_{k=1}^K \ell^{\text{FM}}_k(\tilde{x}_k, c_k; B^{\text{fake}}_\psi)$, matching Eq. (12).
  Take one optimizer step on ψ to minimize $\mathcal{L}_{\text{fake}}$.
  Update the verifier D_ϕ.
  for $k = 1, \dots, K$ **do**
    Compute $a^{\text{real}}_k \leftarrow D_\phi(c_k, x^{\text{real}}_k)$ and $a^{\text{stu}}_k \leftarrow D_\phi(c_k, \hat{x}_k)$ as in Eq. (8).
  end for
  Compute $\mathcal{L}_D \leftarrow -\sum_{k=1}^K \left[\log \sigma(a^{\text{real}}_k) + \log(1 - \sigma(a^{\text{stu}}_k))\right]$ as in Eq. (9).
  Take one optimizer step on ϕ to minimize $\mathcal{L}_D$; no gradient is applied to S_θ in this step.
  Update the student S_θ.
  Freeze B^{real}, B^{fake}_ψ, and D_ϕ.
  for $k = 1, \dots, K$ **do**
    Score the student chunk $a_k \leftarrow D_\phi(c_k, \hat{x}_k)$ with gradients stopped through D_ϕ.
  end for
  Build the causal mask from Eq. (10): $g_1 \leftarrow 1$ and $g_{k+1} \leftarrow g_k \mathbf{1}[a_k > \tau]$ for $k = 1, \dots, K-1$.
  Evaluate the bidirectional denoisers once on the full noised rollout: $\epsilon^{\text{real}}_{1:K} \leftarrow B^{\text{real}}(\tilde{x}_{1:K}, t_{1:K}, c_{1:K})$, $\epsilon^{\text{fake}}_{1:K} \leftarrow B^{\text{fake}}_\psi(\tilde{x}_{1:K}, t_{1:K}, c_{1:K})$.
  for $k = 1, \dots, K$ **do**
    Form the masked DMD contribution $g_k \nabla_\theta \ell^{\text{DMD}}_k$ from the k-th residual pair $(\epsilon^{\text{real}}_k, \epsilon^{\text{fake}}_k)$ using Eq. (5).
  end for
  Take one optimizer step on θ with $\nabla_\theta \mathcal{L}_{\text{stu}} = \sum_{k=1}^K g_k \nabla_\theta \ell^{\text{DMD}}_k$, corresponding to Eq. (11).
end for
return S_θ; discard B^{fake}_ψ and D_ϕ at inference.

an additional training loss. It clarifies the role of D_ϕ under the same operational rule used in Algorithm 1: the verifier is optimized by binary classification, then frozen and used only to form the stop-gradient mask $g_{1:K}$ for the student update.

For a fixed rollout step k and conditioning tuple $c_k = (y, p, u_k, m_k)$, let $P_k(\cdot \mid c_k)$ denote the reference conditional chunk distribution, induced by the real teacher B^{real} or by the reference-positive sampler $\mathcal{R}$ used for verifier positives. Let $Q_{\theta,k}(\cdot \mid c_k)$ denote the conditional distribution of the current student chunk $\hat{x}_k = S_\theta(c_k)$. We use $P_{k,t}$ and $Q_{\theta,k,t}$ for the corresponding noised distributions after applying q_t in Eq. (3). The following standard assumptions isolate the intended behavior of the objective: (i) B^{real} and B^{fake}_ψ provide the teacher and student scores on the contexts being supervised; (ii) D_ϕ has enough capacity to reach the Bayes classifier for the binary loss in Eq. (9); and (iii) during the student update, the gate g_k is treated as a fixed stop-gradient weight, exactly as in Algorithm 1.

Bayes form of the verifier. Conditioned on c_k, the population version of Eq. (9) is

$$\mathcal{L}_{D,k}(D \mid c_k) = -\mathbb{E}_{x\sim P_k(\cdot|c_k)} \log \sigma(D(c_k, x)) - \mathbb{E}_{x\sim Q_{\theta,k}(\cdot|c_k)} \log\big(1 - \sigma(D(c_k, x))\big). \tag{13}$$

Because the two terms have equal class weights, this loss is minimized pointwise. Writing $p_k(x \mid c_k)$ and $q_{\theta,k}(x \mid c_k)$ for the densities of P_k and $Q_{\theta,k}$, the derivative with respect to the scalar logit $d = D(c_k, x)$ is

$$-p_k(x \mid c_k)\big(1 - \sigma(d)\big) + q_{\theta,k}(x \mid c_k)\sigma(d). \tag{14}$$

Setting this derivative to zero gives the Bayes-optimal verifier

$$\sigma(D^\star(c_k, x)) = \frac{p_k(x \mid c_k)}{p_k(x \mid c_k) + q_{\theta,k}(x \mid c_k)}, \qquad D^\star(c_k, x) = \log \frac{p_k(x \mid c_k)}{q_{\theta,k}(x \mid c_k)}. \tag{15}$$

Thus the learned verifier estimates a conditional teacher–student density ratio. The gate test $D^\star(c_k, \hat{x}_k) > \tau$ is therefore a likelihood-ratio test: it accepts a student chunk only when its reference density is sufficiently large relative to its current student density. In particular, if the student assigns mass to a region where $p_k(x \mid c_k) = 0$ but $q_{\theta,k}(x \mid c_k) > 0$, then $D^\star(c_k, x) = -\infty$ and that region is rejected by any finite threshold. Consequently, the supervised prefix produced by Eq. (10) removes student samples that fall outside the reference support before they can contribute suffix gradients. This is the population counterpart of the ideal validity indicator $z_k^\star$ in Eq. (7).

This density-ratio interpretation also explains why the verifier becomes non-discriminative as the student approaches the reference distribution. If $Q_{\theta,k}(\cdot \mid c_k) = P_k(\cdot \mid c_k)$, then $D^\star(c_k, x) = 0$ on their common support and the posterior probability in Eq. (15) equals $1/2$. Hence the verifier can no longer distinguish teacher and student chunks under the same conditioning. For the nonpositive thresholds used by the main MGD model, equality of the two conditional distributions is therefore accepted by the gate. More stringent thresholds may reduce coverage, as reflected by the threshold ablation, but they still do not reverse the DMD update on any accepted chunk.

Compatibility with the DMD update. Under the score interpretation stated in Section 3, the DMD gradient for chunk k is the parameter-space pullback of a teacher–student score gap. Equivalently, for the contexts on which the scores are valid, there is a local distribution-matching discrepancy $\mathcal{D}_{\mathrm{DMD},k}(\theta \mid c_k) \geq 0$ whose sample gradient is represented by Eq. (5):

$$\nabla_\theta \mathcal{D}_{\mathrm{DMD},k}(\theta \mid c_k) = \mathbb{E}_{t,\epsilon}\big[J_\theta(\hat{x}_k)^\top w(t)\big(\epsilon_k^{\mathrm{fake}} - \epsilon_k^{\mathrm{real}}\big)\big]\,. \tag{16}$$

The discrepancy is minimized when the noised student and teacher scores match, i.e., when the scores of $Q_{\theta,k,t}(\cdot \mid c_k)$ and $P_{k,t}(\cdot \mid c_k)$ agree for the timesteps being sampled. Since Gaussian noising has a non-vanishing Fourier transform, equality of the noised distributions over the sampled diffusion times implies equality of the underlying clean conditional distributions on the supervised context.

MGD replaces the unconditional rollout gradient with the masked gradient

$$\nabla_\theta \mathcal{D}_{\mathrm{MGD}}(\theta) = \sum_{k=1}^{K} \mathbb{E}[g_k\, \nabla_\theta \mathcal{D}_{\mathrm{DMD},k}(\theta \mid c_k)] = \sum_{k=1}^{K} \mathbb{E}_{t,\epsilon}\big[g_k J_\theta(\hat{x}_k)^\top w(t)\big(\epsilon_k^{\mathrm{fake}} - \epsilon_k^{\mathrm{real}}\big)\big]\,. \tag{17}$$

Because $g_k \in \{0, 1\}$ and is not differentiated through in the student step, Eq. (17) is only a nonnegative reweighting of the original DMD gradient terms. When $g_k = 1$, MGD applies exactly the same DMD direction as Eq. (5); when $g_k = 0$, it removes that term instead of replacing it with an adversarial or opposite-direction update. For a fixed mask during one student step, a sufficiently small gradient descent step on the corresponding gated discrepancy satisfies

$$\mathcal{D}_{\mathrm{MGD}}(\theta - \eta\nabla_\theta \mathcal{D}_{\mathrm{MGD}}) = \mathcal{D}_{\mathrm{MGD}}(\theta) - \eta\|\nabla_\theta \mathcal{D}_{\mathrm{MGD}}(\theta)\|_2^2 + O(\eta^2), \tag{18}$$

so the accepted-prefix discrepancy is locally decreased rather than increased. This establishes that the verifier does not conflict with the DMD objective: it selects where the DMD score gap is trusted, but it does not introduce a different student objective.

Consistency along the autoregressive memory. The same argument extends from chunks to memory states by induction over the autoregressive rollout. At $k = 1$, the mask is anchored by $g_1 = 1$, so the student is trained by the ordinary DMD direction under the initial conditioning. Under the ideal score assumptions above, its stable matching point satisfies $Q_{\theta,1}(\cdot \mid c_1) = P_1(\cdot \mid c_1)$.

Suppose that for all positions before k, the accepted student chunks match the corresponding reference conditional distributions. Since the memory update $m_{j+1} = \mathcal{M}(m_j, \hat{x}_j)$ is the same deterministic update used to define the rollout, the induced distribution of m_k on the accepted prefix is then the same pushforward memory distribution as the reference rollout. The Bayes verifier therefore receives chunks under a reference-aligned context; when $Q_{\theta,k} = P_k$ on that context, its optimal logit is zero and the nonpositive gate threshold accepts the step. The DMD term at position k is consequently active and again pushes $Q_{\theta,k}(\cdot \mid c_k)$ toward $P_k(\cdot \mid c_k)$.

By induction, the joint distribution of the verified student prefix approaches the teacher or reference rollout distribution whenever the DMD matching conditions are reached at each accepted step. The verifier is essential because, before this limit is reached, it rejects prefixes whose likelihood ratio indicates departure from the reference support and prevents DMD gradients from being computed under corrupted memory states. Therefore, MGD preserves the distribution-matching target of DMD while restricting it to the portion of the autoregressive rollout on which that target is well conditioned.

D Discussion

D.1 Fixed-step inference protocol for distilled students

All MGD results in the teaser Pareto plot and in the quantitative experiments use 4-step inference. This choice is not a missing step sweep, but follows from the nature of few-step diffusion distillation. After distillation, the student S_θ is optimized for a particular inference budget, solver, and noise-time schedule. In our experiments, this target budget is 4 steps. The DMD gradient in Eq. (5), the fake-score model B_ψ^{fake}, and the gated student objective in Eq. (11) are all trained around samples generated by this fixed student sampler. The resulting model should therefore be viewed as a 4-step generator rather than as a generic sampler whose number of denoising steps can be changed freely at test time.

Changing the number of inference steps after distillation would alter the student sampling distribution without re-optimizing the corresponding score-gap objective. Fewer steps would expose S_θ to a coarser trajectory than the one used during training, while more steps would repeatedly apply a model whose update rule was not trained to refine samples along that longer trajectory. Either change can move the generated chunks, and consequently the autoregressive memories m_k, away from the distribution on which B_ψ^{fake} tracked the student and on which the gate decisions were calibrated. Such evaluations would conflate the effect of MGD with an out-of-distribution sampler change. For this reason, the Pareto plot compares different methods at their intended operating points: high-budget pretrained teachers or baselines may be run with their native step counts, while the distilled MGD student is evaluated at its distilled 4-step budget. Obtaining MGD models for other points on the step-count axis would require training or distilling separate students for those target budgets. We therefore report only the 4-step MGD model, which is the setting optimized by the method and the setting relevant to the low-latency deployment goal.

D.2 Verifier in MGD versus discriminator in GANs

The verifier in MGD is related to the discriminator used in adversarial generative modeling, but the two modules have different operational roles. A GAN discriminator is trained in a minimax game to distinguish real samples from generated samples, and the generator is updated by gradients derived from this discriminator so that generated samples become harder to classify (Goodfellow et al., 2014). The MGD verifier uses the same basic statistical idea of separating reference-aligned samples from student-generated samples, but it applies it in a conditional, chunk-level setting. Specifically, $D_\phi(c_k, x)$ compares real and student chunks under the same autoregressive conditioning tuple $c_k = (y, p, u_k, m_k)$, so its score estimates whether the candidate chunk is reliable enough to extend the current memory.

The key difference is that MGD does not use D_ϕ as an adversarial training signal for the student. In GAN-style training, the discriminator loss induces a gradient that is backpropagated through the generated sample to update the generator. Such updates can be sensitive to discriminator design, calibration, saturation, and high-dimensional gradient geometry (Sauer et al., 2021; Mescheder et al., 2018; Zhao et al., 2020). In contrast, MGD uses the verifier only for estimation and selection. The verifier score is thresholded and converted into the causal gate mask in Eq. (10). This mask is then

treated as a training decision that selects the prefix on which Eq. (11) is applied. No adversarial loss such as $-\log \sigma(D_\phi(c_k, \hat{x}_k))$ is optimized for S_θ, and no gradient from D_ϕ is backpropagated through $\hat{x}_k$ to update the student.

This distinction is important for autoregressive distillation with persistent memory. The purpose of the verifier is not to directly push student chunks toward regions that fool a discriminator. Instead, it decides where the distribution-matching objective remains well conditioned. Once a chunk causes memory drift, later teacher and fake-score queries are conditioned on a corrupted memory state, so their score gap may no longer provide a reliable DMD update. MGD therefore keeps the DMD objective itself unchanged and restricts it to the verified prefix whose memory has not drifted. In this sense, the verifier provides discriminative estimation for supervision selection rather than an adversarial objective. In pilot experiments, directly adding an adversarial loss from the verifier to the student made long-horizon distillation unstable and sometimes divergent, which further motivated the non-adversarial design used in MGD.

E LIMITATIONS

The present empirical study has several scope boundaries. First, the experiments are conducted on a single backbone–dataset pair, namely HY-World 1.5 with a 7,000-sample subset of Sekai. This setting is representative of the target interactive world-modeling regime, but it does not establish that the same gating mechanism transfers unchanged across different autoregressive backbones, data domains, memory designs, or control interfaces. Evaluating such cross-backbone and cross-dataset transfer is an important direction for future work. Second, each reported number is computed from many generated videos but from a single training seed. The main conclusions are based on consistent long-horizon improvements, especially at 253 frames, rather than on small short-horizon margins. For example, the 61-frame Generation Quality gap between the moderate-gating MGD setting and the Context Forcing limit in Table 4 is 0.006, which is within a plausible range of seed-level variation. Multi-seed training would provide a more complete estimate of variance and should be included in future large-scale evaluations. Third, our evaluation horizon is limited to 253 frames. This already stresses the autoregressive memory more than short clips, but it does not cover substantially longer rollouts. At longer horizons, memory drift may interact with finite memory capacity, memory compression, prompt or control changes, and accumulated geometric error in ways not captured by the current benchmark. The behavior of MGD under minute-scale or open-ended interactive generation therefore remains open. Fourth, the current gate is uncalibrated and non-adaptive. The threshold τ is a single fixed scalar selected on the validation split, rather than a calibrated probability threshold or a per-step decision rule. A fixed threshold is simple and works well in our experiments, but it cannot explicitly account for step-dependent uncertainty, changing control difficulty, or different memory states. Future extensions may study calibrated verifier scores, adaptive thresholds, richer memory-conditioned verifier inputs, and combinations with teacher-side score correction.