STREAMING-AAA: STREAMING AAA GAME VIDEO GENERATION AT HIGH RESOLUTION, HIGH FRAME RATE, AND LOW LATENCY

Anonymous authors
Paper under double-blind review

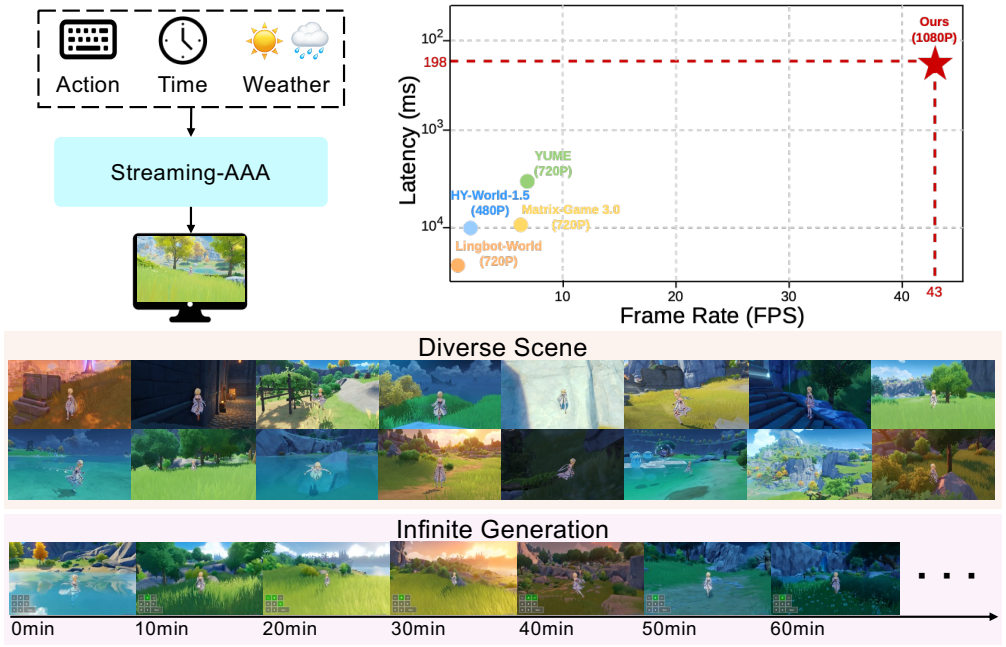

Figure 1: Overview of Streaming-AAA: Streaming AAA game video generation framework applied to *Genshin Impact*, demonstrating high-resolution, high-frame-rate, and low-latency gameplay synthesis. Speed and latency are tested on NVIDIA A800 GPUs.

ABSTRACT

Controllable streaming video generation offers a promising route toward interactive game simulation, yet existing methods rarely achieve high visual fidelity, high frame rates, and low interaction latency at once, resulting in poor responsiveness. We present Streaming-AAA, a paradigm for controllable infinite generation of AAA-quality game videos that produces **1080p** gameplay at **43 FPS** with **198 ms** end-to-end latency on merely **NVIDIA A800 GPUs**, while supporting long-horizon play without severe temporal degradation or collapse. Streaming-AAA rests on three designs: a redesigned VAE for efficient high-resolution decoding that enables real-time 1080p synthesis; diffusion-forcing pre-training with a substantially extended temporal horizon for long-term dependencies and persistent memory; and post-training distillation with memory compression that improves inference efficiency while retaining visual quality and temporal consistency. Experiments show substantial gains in fidelity, frame rate, latency, and long-term stability over prior controllable video generation approaches, yielding a significantly more playable interactive experience. Video results are available at `https://anonymous.4open.science/w/streaming-aaa/`.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

Interactive game simulation has become an increasingly important application of generative video models, where the system must synthesize future visual observations conditioned on user controls in a streaming manner Ha & Schmidhuber (2018); Ali et al. (2025); Sun et al. (2025a); Hong et al. (2025); Mao et al. (2026). Unlike offline video generation, playable game generation imposes simultaneous requirements on visual fidelity, temporal consistency, frame rate, and interaction latency—requirements that are especially stringent for AAA-level titles with high-resolution rendering, fast motion, complex scenes, and long-horizon interactions. In this work, we primarily target *Genshin Impact*, a widely played open-world action RPG whose environments are familiar enough that generation quality is easy to judge.

Despite recent progress in controllable video generation and world models Yu et al. (2025b); Ball et al. (2025); Mao et al. (2026); He et al. (2025); Tang et al. (2025); Ye et al. (2025); HunyuanWorld (2025), existing streaming systems remain far from practical gameplay: many produce plausible short clips whose quality degrades over long horizons, while others support control or interaction but run at low resolution, low frame rate, or high latency. Consequently, no current method jointly satisfies high-fidelity 1080p generation, real-time frame rates, low end-to-end latency, and stable infinite interaction.

We present **Streaming-AAA**, a controllable infinite streaming AAA game video generation framework for high-resolution, high-frame-rate, low-latency gameplay, as shown in Figure 1. We introduce an efficient high-resolution video VAE that compresses 1080p frames into compact spatio-temporal latents and reconstructs them with low decoding overhead, and builds on this representation a UNet-based latent diffusion model with factorized spatial, temporal, and cross-attention modules for efficient visual modeling, temporal reasoning, and multi-modal control. For long-horizon streaming we further adopt long-context pre-training, post-training alignment with distribution matching distillation, and a hybrid short/long-term KV-cache memory, preserving long-range consistency at low interaction latency.

Our main contributions are summarized as follows:

- We propose **Streaming-AAA**, the first controllable infinite-horizon AAA game video generation framework that simultaneously delivers high visual fidelity, real-time throughput ($43\,\mathrm{FPS}$ at $1080\,\mathrm{p}$), ultra-low end-to-end latency ($198\,\mathrm{ms}$), and stable long-rollout.
- We design a purpose-built causal video VAE with $16\times$ spatial compression and a factorized 3D-UNet diffusion backbone that decouples spatial texture, temporal dynamics, and conditional control, cutting inference cost while flexibly integrating frame-aligned conditioning signals such as agent actions and weather states.
- We establish a pre/post-training paradigm for unbounded autoregressive streaming: Diffusion Forcing pre-training encodes the inductive bias of growing uncertainty at distant horizons, and distribution matching distillation aligns the model with **2-step** generation, boosting rollout stability without sacrificing real-time performance.

2 RELATED WORKS

Video generation models. Large-scale text-to-video diffusion models Gupta et al. (2024); Gao et al. (2025); Seedance et al. (2026); Kong et al. (2024); Team et al. (2025b); Wan et al. (2025); Zhang et al. (2025); Contributors (2025); HaCohen et al. (2026); OpenAI (2024); Team et al. (2025a); Teng et al. (2025) produce photorealistic short clips with impressive quality and diversity, but typically need minutes of inference for a few seconds of video and cannot generate arbitrarily long sequences in a streaming fashion. To address this, several methods Huang et al. (2026); Cui et al. (2025); Chen et al. (2026); Yang et al. (2025); Chen et al. (2024); Cui et al. (2026); Zhu et al. (2026) retrofit pre-trained models for streaming generation: Self-Forcing Huang et al. (2026) bridges the train-test gap by aligning the training distribution with the streaming rollout, Context Forcing Chen et al. (2026) adds long-context conditioning for extended autoregressive consistency, and LongLive Yang et al. (2025) combines efficient architectures with streaming inference for real-time interactive generation. Alternatively, Diffusion Forcing Chen et al. (2024) instills streaming capability during pre-training by combining next-token prediction with full-sequence diffusion. Despite this progress, real-time streaming inference at high resolution remains a major obstacle for playable game generation.

World models. World models learn predictive environment representations for planning and interaction, and fall into two categories. Reconstruction-based approaches Yu et al. (2024; 2025a); Labs (2025b) build explicit representations such as 3D Gaussian Splatting for real-time exploration, but yield mostly static worlds with limited interactivity. Generation-based approaches instead use video models as world simulators Bar-Tal et al. (2024); Blattmann et al. (2023); Brooks et al. (2024); Decart et al. (2024), and a growing body of work Ha & Schmidhuber (2018); Assran et al. (2025); Bruce et al. (2024); Hong et al. (2025); Ali et al. (2025); Sun et al. (2025a); Xiang et al. (2025); Sun et al. (2025b); Wang et al. (2026); Chen et al. (2025); Gao et al. (2026); Hafner et al. (2025); Yang et al. (2024); Savva et al. (2026) explores interactive and controllable variants addressing long-horizon memory, action conditioning, and real-time streaming. However, none simultaneously achieves 1080p resolution, high frame rate, and low interaction latency—all essential for AAA-level playable generation. Our work closes this gap by jointly optimizing visual fidelity, streaming efficiency, and interactive responsiveness in a unified framework.

3 METHOD

3.1 PROBLEM FORMULATION AND OVERVIEW

We formulate controllable AAA game video generation as an infinite streaming prediction problem. Given previous frames $x_{1:t}$, user controls $a_{1:t+k}$, and a compact memory state m_t, the model predicts the next video segment $p_\theta(x_{t+1:t+k} \mid x_{1:t}, a_{1:t+k}, m_t)$. Unlike offline video generation, this task requires causal generation, faithful response to user inputs, and long-horizon stability under strict constraints on resolution, frame rate, and interaction latency.

Our framework, **Streaming-AAA**, meets these requirements with three components: an efficient high-resolution video VAE that compresses 1080p frames into compact spatio-temporal latents and decodes them with low overhead (Sec. 3.2); a UNet-based latent diffusion model that generates future latents conditioned on player controls and historical memory (Sec. 3.3); and long-context pre-training, post-training alignment, and hybrid KV-cache memory that jointly improve long-term consistency while reducing inference latency (Sec. 3.4–3.5).

3.2 EFFICIENT HIGH-RESOLUTION VIDEO VAE

High-resolution decoding is a major bottleneck for real-time game video generation. We therefore redesign the VAE to handle ultra-high-definition inputs (1792×1024) while keeping the latent dimension manageable and preserving high-frequency details.

Architecture Design. We use an overall temporal-spatial compression factor of $4\times16\times16$, mapping an input clip $x \in \mathbb{R}^{(4N+1)\times H\times W\times 3}$ to a latent $z \in \mathbb{R}^{(N+1)\times \frac{H}{16}\times \frac{W}{16}\times C}$ that supports both single- and multi-frame compression. The key distinction lies in the decoder: rather than relying solely on interpolation or transposed convolutions, we place three sequential Conv3D upsampling blocks followed by a **Pixel Shuffle** operation for sub-pixel upsampling at the largest stage, which mitigates the checkerboard artifacts induced by large-stride deconvolutions while preserving fine-grained texture.

Multi-Stage Curriculum Training. Training a VAE from scratch for high-resolution video is inherently unstable, so we adopt a three-stage curriculum for faster convergence: **(i) Spatial Initialization** on static 256×256 images to stabilize basic color distributions and edges; **(ii) Temporal Extension** on center-cropped 256×256 video patches to learn short-term temporal coherence; and **(iii) Native Resolution** fine-tuning on full 1792×1024 videos to adapt to the target aspect ratio and scale. For multi-frame training, we further adopt a rolling prediction mechanism that predicts frames indexed as $4n + 1$ ($n \geq 0$), applying gradients only to the last 1 or 4 predicted frames of each sequence to prevent error accumulation while maintaining temporal smoothness. The model is optimized with a composite $\mathcal{L}_2$ and LPIPS loss.

3.3 REDESIGNING UNET-BASED VIDEO DIFFUSION MODELS

Leveraging the compact latent space provided by our VAE, we construct a 3D UNet-based diffusion model following a symmetric encoder-decoder structure with three downsampling and three upsampling operations, reducing the bottleneck resolution to $\frac{1}{8}$ of the input.

Factorized Spatio-Temporal Blocks. To mitigate the overhead of dense 3D operations, each UNet stage comprises three specialized modules. The **Spatial Module**, built from convolutional layers, models intra-frame textures and structural details. The **Cross-Attention Module** serves as the condi-

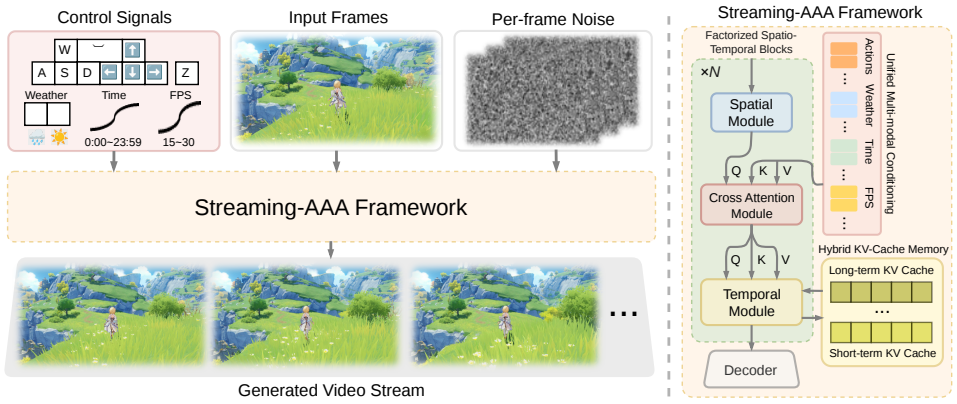

Figure 2: **Overview of the inference pipeline.** During inference, the 3D UNet-based Diffusion Model takes noise as input, injects control signals via cross-attention, and propagates historical frame information through temporal attention. After multiple denoising steps, the reconstructed latents are decoded by the VAE decoder to produce the generated video frames.

tioning interface: each condition type—*action*, *weather*, *time-of-day*, and *frame-rate*—is processed by a dedicated embedding layer, tokenized, and concatenated into a unified condition sequence that is causally injected at every stage, which standardizes the input interface across modalities and lets the model selectively attend to relevant conditions at different denoising steps. The **Temporal Module** operates causally along the temporal axis to capture the dynamic evolution of each latent pixel across frames, and caches the keys and values of finalized frames so that subsequent frames reference them instead of recomputing, reducing per-step complexity from $O(T^2)$ to $O(T)$ for sequence length T; Section 3.5 details the hybrid memory policy governing which tokens are retained. This factorization decouples spatial modeling, temporal dynamics, and conditional control, substantially reducing complexity relative to full spatio-temporal attention while improving cross-scenario generalization.

3.4 Modeling Long-horizon Uncertainty with Diffusion-forcing Pre-training

We adopt the Diffusion Forcing (DF) Chen et al. (2024) paradigm for pre-training, as its per-frame independent noise scheduling aligns naturally with long-horizon, controllable world simulation. Instead of a globally shared noise level, DF samples an independent step $k_i \sim p(k)$ over the DDPM Ho et al. (2020) range $[0, K]$ for each frame i, giving the noised latent

$$\mathbf{z}_i^{k_i} = \sqrt{\bar{\alpha}_{k_i}}\,\mathbf{z}_i^0 + \sqrt{1-\bar{\alpha}_{k_i}}\,\boldsymbol{\epsilon}_i, \qquad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{1}$$

where $\mathbf{z}_i^0$ is the clean VAE latent and $\bar{\alpha}_{k_i} = \prod_{s=1}^{k_i}(1-\beta_s)$ is the cumulative noise-retention coefficient. Stacking these per-frame latents forms a 2D noise schedule matrix that encapsulates the causal dynamics of open-world evolution. The per-frame assignment is also natively compatible with our frame-aligned metadata, enabling frame-level modeling of acute dynamic transitions.

We train with *velocity (v-)prediction* rather than $\boldsymbol{\epsilon}$-prediction, defining the target velocity of frame i as

$$v_i = \sqrt{\bar{\alpha}_{k_i}}\,\boldsymbol{\epsilon}_i - \sqrt{1-\bar{\alpha}_{k_i}}\,\mathbf{z}_i^0. \tag{2}$$

Let $\mathbf{z}^{\mathbf{k}} \triangleq \{\mathbf{z}_i^{k_i}\}_{i=1}^T$ be the noised sequence, $\mathbf{k} = \{k_i\}_{i=1}^T$ the per-frame steps, $\mathbf{c}$ the control condition sequence, and $f(\cdot)_i$ the i-th output frame. The pre-training objective is then

$$\mathcal{L}_{\text{DF}} = \mathbb{E}_{\{\mathbf{z}_i^0\},\,\mathbf{c},\,\mathbf{k},\,\{\boldsymbol{\epsilon}_i\}}\left[\frac{1}{T}\sum_{i=1}^{T}\left\|f(\mathbf{z}^{\mathbf{k}}, \mathbf{c}, \mathbf{k})_i - v_i\right\|_2^2\right]. \tag{3}$$

The per-frame independent $(\mathbf{z}_i^{k_i}, k_i, v_i)$ triplets act as an implicit curriculum that stabilizes pre-training on heterogeneous 3A data and yields continuous post-training distributions for infinite rollouts; empirically, v-prediction further mitigates error accumulation over long horizons.

Pre-training proceeds in two stages. **Stage 1** trains on 64-frame clips for 200 epochs, which permits larger batches and fast convergence on low-level textures, spatial structure, and short-term dynamics.

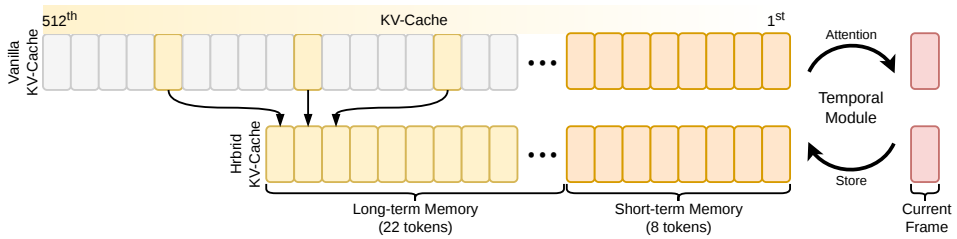

Figure 3: **Hybrid KV-Cache Memory.** The first row shows the vanilla KV-cache array; the second row shows our hybrid array, which combines sparsely sampled long-term tokens with a consecutive short-term window to extend the effective memory span to 16 s within a budget of 30 cached tokens.

Stage 2 extends the context to 640 frames (∼21 s) for 20 epochs, forcing the Temporal and Cross-Attention modules to reason over extended horizons and thereby mitigating temporal drift across distant frames.

3.5 Post-Training Alignment for Streaming Inference

As illustrated in Figure 2, streaming inference conditions on clean historical frames and control signals to generate future latents. This creates a non-negligible train-test mismatch: DF pre-training assigns each latent token an independent random noise level, so the model learns to denoise under a wide range of partially corrupted contexts, whereas teacher-forcing rollout at inference only exercises the zero-noise subset of that distribution, degrading quality and temporal consistency.

To close this gap, we align the model with its inference-time behavior via Distribution Matching Distillation (DMD) Yin et al. (2024; 2025). Inspired by the second stage of self-forcing distillation Huang et al. (2026), we optimize the generator to produce future latent tokens under fixed denoising timesteps and clean historical contexts. Since our pre-trained model already supports self-rollout, we initialize the generator, real score model, and fake score model from the same checkpoint, which stabilizes distribution matching and avoids training the rollout generator from scratch.

Randomized Score-Model Perturbation for DMD. Our key departure from conventional DMD is to perturb score-model inputs with independent per-token noise rather than a shared global level, preserving DF's stochastic training behavior. Let $\hat{\mathbf{z}}^0 = \{\hat{\mathbf{z}}_i^0\}_{i=1}^T$ be the clean latents from the student's streaming rollout, with per-frame steps $\mathbf{k} = \{k_i\}$ and draws $\boldsymbol{\epsilon} = \{\boldsymbol{\epsilon}_i\}$. We define the DF-consistent perturbation operator frame-wise as

$$\big(\mathcal{E}_{\mathbf{k}}[\hat{\mathbf{z}}^0; \boldsymbol{\epsilon}]\big)_i = \sqrt{\bar{\alpha}_{k_i}}\, \hat{\mathbf{z}}_i^0 + \sqrt{1 - \bar{\alpha}_{k_i}}\, \boldsymbol{\epsilon}_i. \tag{4}$$

Feeding the perturbed sequence $\tilde{\mathbf{z}} \triangleq \mathcal{E}_{\mathbf{k}}[\hat{\mathbf{z}}^0; \boldsymbol{\epsilon}]$ together with the condition sequence $\mathbf{c}$ to the score networks, and writing $\mu(\cdot)_i$ for the i-th output frame and $sg(\cdot)$ for the stop-gradient operator, the DMD objective on latent i becomes

$$\mathcal{L}_{\text{DMD}}^i = \mathbb{E}_{\hat{\mathbf{z}}^0,\, \mathbf{c},\, \mathbf{k},\, \boldsymbol{\epsilon}} \left[\frac{1}{2} \left\| \hat{\mathbf{z}}_i^0 - sg\Big(\hat{\mathbf{z}}_i^0 - \big(\mu_{real}(\tilde{\mathbf{z}}, \mathbf{c}, \mathbf{k})_i - \mu_{fake}^{\phi}(\tilde{\mathbf{z}}, \mathbf{c}, \mathbf{k})_i \big) \Big) \right\|_2^2 \right], \tag{5}$$

where μ_{real} is the fixed real score network and μ_{fake}^{ϕ} the fake score network trained on the student's streaming rollout with the DF loss. Empirically this outperforms shared-noise DMD: independently noised inputs match DF's pre-training distribution, so the teacher supplies more informative gradients, while decoupled teacher/student noise prevents overfitting to fixed noise patterns.

Extended-Horizon Post-Training. Long-term playability requires stable memory over extended rollouts, so we push the post-training horizon as far as hardware permits, training with 512-frame sequences (∼17 s of gameplay) on 96GB H20 GPUs. This substantially improves retention of scene context, action-dependent dynamics, and temporal consistency, reducing drift and collapse during continuous play.

Hybrid KV-Cache Memory for Efficient Infinite Streaming. Several prior methods, including HY-World1.5 HunyuanWorld (2025) and Lingbot Team et al. (2026), condition generation on camera pose, which provides an explicit positional anchor that the model can exploit to retrieve historical scene context: given the current pose, it can look up what was observed at nearby or previously visited locations. Our framework foregoes camera pose entirely and relies on raw player actions for

control, which makes the generated world directly playable without any pose estimation or camera-rig abstraction. The trade-off is that no positional index is available to guide memory retrieval, so the model must instead maintain a sufficiently long token history from which it can infer scene consistency on its own. We therefore extend the KV-cache horizon substantially and compensate for the resulting compute cost with a sparse-plus-dense memory layout described below. Extending temporal memory also lengthens the KV cache at inference, but a dense long-horizon cache inflates computation and memory, raising latency and lowering frame rate—directly conflicting with playable low-latency generation. As illustrated in Figure 3, we therefore split the cache into two complementary parts: **long-term memory** holds sparsely sampled historical tokens that retain global scene context and slow-changing state, while **short-term memory** holds a consecutive window of the most recent tokens that preserves local continuity and fine-grained motion. The model thus accesses both distant context and recent dynamics without attending to every historical token.

In our implementation the cache holds 30 tokens: the first 22 form long-term memory sampled at an interval of 5 frames, and the last 8 form consecutive short-term memory, yielding an effective memory span of up to 16 seconds. We apply the same policy during post-training distillation and inference, keeping the memory layout consistent and preventing the generator from overfitting to a dense historical context unavailable at test time.

Parallelized Teacher-Forcing Inference. For streaming generation, we adopt teacher-forcing inference rather than shifted-window denoising. Since the model always conditions on finalized clean historical frames, teacher forcing reduces the ambiguity of historical context and improves generation quality. It also avoids repeatedly denoising overlapping windows, which is important for low-latency interactive gameplay. However, a naive implementation still suffers from limited parallelism because cache writing and next-frame denoising are executed sequentially.

To improve inference efficiency, we fuse the KV-cache writing step with the initial denoising step of the next frame. Instead of performing three separate forward passes with batch size 1, we execute one forward pass with batch size 2 followed by one forward pass with batch size 1. This batching strategy improves hardware utilization and reduces per-frame latency without changing the teacher-forcing rollout formulation. As a result, Streaming-AAA retains the quality benefits of clean-history conditioning while achieving higher inference throughput for real-time play.

4 Experiments

4.1 Implementation Details

Data collection. We build an automated capture pipeline on OBS Studio. A Python script running in the OBS scripting runtime samples random actions—locomotion keys for moving forward, backward, left, and right, a jump key, together with relative mouse displacement for camera rotation—and injects them into the running game, while OBS concurrently records the rendered frames at 1920×1080. Since the same script both emits the actions and controls the recorder, every frame is annotated with the control signal in effect when it was rendered, yielding frame-aligned (frame, action) pairs without post-hoc action estimation from pixels. To avoid over-sampling the neighborhood of a single spawn point, we partition the game map into disjoint regions and run independent capture sessions in each. For *Genshin Impact* we collect 11K clips totalling 20M frames at 30 fps. Every frame is annotated with its action label and environmental context (weather and time-of-day).

Training. Our diffusion model is initialized from Stable Diffusion v1.5 Rombach et al. (2022). To reduce overhead we remove the two self-attention modules at the largest spatial resolution, and for temporal modeling we add Motion Modules Guo et al. (2023) with the number of temporal blocks reduced from 2 to 1, yielding 1.2B parameters in total. Training proceeds in three stages: diffusion forcing on 64-frame clips ($\sim$2 s); sequence extension to 640 frames ($\sim$21 s) for long-term memory; and post-training alignment via self-forcing-style distillation Huang et al. (2026) with our hybrid short-long memory at 512 frames ($\sim$17 s).

Inference. We evaluate on NVIDIA A800 GPUs, deploying the diffusion model and VAE decoder on separate GPUs for pipelined execution. We use 2 denoising steps at timesteps 999 and 666, and fuse KV-cache writing with the first denoising step into a single batched forward pass to cut per-frame latency. We further apply *torch.compile* to the diffusion model and convert the VAE decoder to TensorRT.



Figure 4: **Qualitative results of our method on *Genshin Impact*.** The top five rows show rollouts from in-domain frames, while the bottom two rows show rollouts from out-of-domain frames.

Evaluation protocol. We build the test set by randomly capturing 50 in-game screenshots as starting frames. For each starting frame we sample a fresh random action sequence spanning 60 s (1800 actions in total), and set the weather and time-of-day conditions to match those of the starting frame. All baselines are evaluated under identical settings: we run their original code and released checkpoints on the same starting frames, action sequences, and conditioning signals. For methods that additionally require a text prompt, we caption the starting frame with GPT-5 and feed the resulting description as the prompt.

4.2 Qualitative Results

In-domain visualization. The upper five rows of Figure 4 show in-domain generation results. Our method synthesizes scenes across different times of day, including daytime (first and second rows) and night (third row). It also faithfully reproduces the wild monsters that populate the game world: as shown in the third row, our model generates the cryo slimes commonly encountered in Genshin Impact. Beyond running on flat ground, the avatar further supports climbing and swimming. Together, these results demonstrate the effectiveness of our method across diverse scenarios.

Out-of-domain visualization. The bottom two rows of Figure 4 show generation results in out-of-domain scenes, where we use FLUX Labs (2025a) to replace the background with a pyramid and a busy city street, respectively. As illustrated in the second-to-last row, the pyramid retains its shape without noticeable distortion even as the avatar moves through the scene, and the content gradually transitions back into the in-domain distribution. This indicates that our method generalizes to a certain degree beyond the training distribution.

Comparison with state-of-the-art approaches. We compare our method against HY-World1.5 HunyuanWorld (2025), Lingbot Team et al. (2026), Matrix Game 3.0 Skywork AI Matrix-Game Team (2026), and YUME Mao et al. (2026) on Genshin Impact. As shown in Figure 5, our method surpasses all competing approaches in generation quality. In the first dusk scene, HY-World1.5 produces repeated shadow patterns that substantially degrade the visual result. Lingbot develops black blotchy artifacts after roughly 30 s of rollout. Matrix Game 3.0 renders both cases with a slightly overexposed appearance. YUME drifts rapidly from the cartoon style of the source domain toward photorealism, synthesizing realistic human bodies and washing the overall palette out to gray. In contrast, our method sustains high-quality generation throughout, without quality degradation as the rollout extends, which confirms the effectiveness of our design.

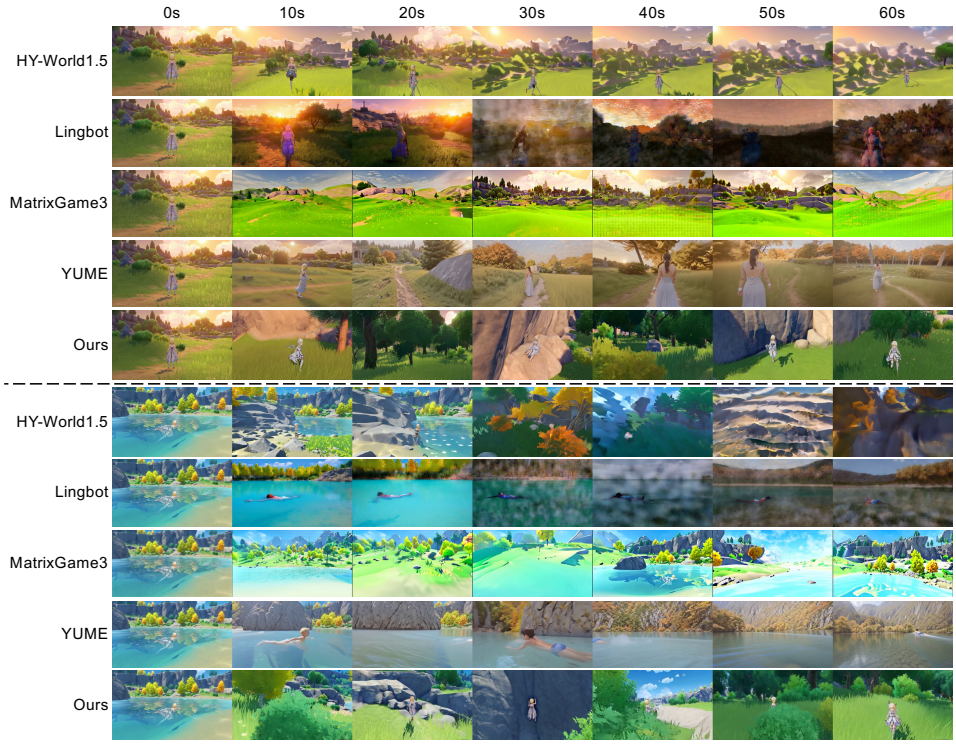

Figure 5: **Qualitative comparison with state-of-the-art approaches on *Genshin Impact*.** We show our method and the competing approaches on two cases over a one-minute rollout; rows are methods and columns are timestamps.

Table 1: **Quantitative and efficiency comparison with state-of-the-art methods.** We compare our method against existing state-of-the-art approaches on several VBench metrics, together with output resolution, frame rate, and action-to-pixel latency. Best result per column is in bold.

	Generation Quality					Efficiency		
Method	Subject Consist.	Background Consist.	Motion Smooth.	Imaging Quality	Average	Resolution	Frame Rate (↑)	Latency (↓)
Lingbot	0.689	0.876	0.983	0.438	0.747	720p	2.6 FPS (8GPUs)	8200 ms
YUME	0.716	0.860	0.985	0.639	0.800	720p	6.7 FPS	6600 ms
Matrix Game 3.0	0.690	0.887	0.987	0.666	0.807	720p	6.1 FPS	9800 ms
HY-World1.5	0.755	0.906	**0.988**	0.589	0.810	480p	2.1 FPS	11300 ms
Ours	**0.761**	**0.910**	0.982	**0.677**	**0.832**	**1080p**	**43.0 FPS**	**198 ms**

4.3 Quantitative Results

Following the protocol of Section 4, we evaluate all methods with VBench Huang et al. (2024; 2025); Zheng et al. (2025) in custom_input mode on subject consistency, background consistency, motion smoothness, and imaging quality, and report in Table 1 the scores averaged over both evaluation titles.

As shown in Table 1, our method outperforms all competing approaches—HY-World1.5 Hunyuan-World (2025), Lingbot Team et al. (2026), Matrix Game 3.0 Skywork AI Matrix-Game Team (2026), and YUME Mao et al. (2026)—on subject consistency, background consistency, and imaging quality, and leads by a clear margin on the average score. Compared with HY-World1.5, we are on par on subject consistency and background consistency, but substantially better on imaging quality (0.677 vs. 0.589). Figure 5 explains the gap: beyond roughly 30 s of rollout HY-World1.5 develops pronounced repetitive patterns, which severely hurt imaging quality. Our motion smoothness is marginally lower than that of the baselines. A plausible explanation is that the baselines move very slowly, so their rollouts consist of comparatively gentle frame-to-frame changes, whereas our rollouts contain fast locomotion and abrupt events such as jumping off a cliff, which naturally depress this metric. Our advantage is even more pronounced in efficiency: we run about $20\times$ faster than HY-World1.5 while cutting latency by more than 98%, and we do so at 1080p rather than 480p–720p.

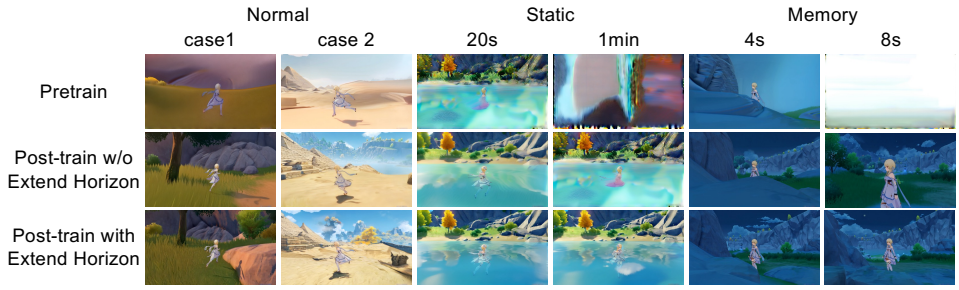

Figure 6: **Ablation study on *Genshin Impact*.** We compare the pre-trained model against post-training with a short and an extended training horizon across three scenarios: *Normal* navigation from in-domain (Case 1) and out-of-domain (Case 2) starting frames; *Static* standstill for 20 s and 60 s; and *Memory* retention after turning the camera away for 4 s/8 s and back.

Taken together, these results show that our method dominates existing state-of-the-art approaches in both generation quality and inference efficiency.

4.4 Ablation Study

Figure 6 presents our ablation study, comparing the pre-trained model against post-training with a short and an extended training horizon.

Normal scenes. Case 1 and Case 2 report results from in-domain and out-of-domain starting frames, respectively. The pre-trained model degenerates within the first 5 s of rollout, with the generated content collapsing toward a monotonous appearance, whereas both post-trained models stay stable in either case. Post-training therefore yields a substantial quality gain within our framework.

Static scenes. When the avatar stands still, the pre-trained model already breaks down after 20 s, and post-training with a short horizon becomes blurry after 60 s. Only the extended-horizon variant preserves a sharp and stable scene throughout.

Memory. We rotate the camera away and then turn back to the original viewpoint. With a short post-training horizon, the scene is already slightly altered after 4 s and differs substantially after 8 s; with the extended training horizon and longer training, it remains essentially unchanged even after 8 s. These comparisons show that both extending the post-training horizon and increasing training length are necessary to confer markedly stronger resistance to collapse in static scenes and a genuine degree of memory.

4.5 Efficiency and Latency Analysis

Frame rate and action-to-pixel latency are critical to interactive experience. End-to-end latency comprises the diffusion model (93 ms on an A800) and the VAE decoder (58 ms). Since player actions may arrive mid-execution, we budget an expected extra half-step of diffusion, giving $93 \times 1.5 + 58 \approx 198$ ms; this is the theoretical computation latency, excluding network transmission and codec overhead. Deploying the diffusion model and VAE decoder on separate GPUs for pipelined execution further yields $1000/93 \times 4 \approx 43$ fps, enabling real-time interactive gameplay. Relative to existing streaming systems, which operate at 480p–720p with single-digit frame rates and multi-second latency, our system delivers 1080p output at 43 fps with 198 ms latency; the detailed comparison is reported in Table 1.

5 Conclusion

We have presented **Streaming-AAA**, a controllable infinite AAA game video generation framework achieving 1080p, 43 fps, 198 ms-latency streaming gameplay synthesis. It integrates an efficient high-resolution video VAE with pixel-shuffle decoding, a factorized UNet latent diffusion model with unified multi-modal conditioning, a progressive pre/post-training pipeline, and a hybrid KV-cache memory with parallelized teacher-forcing inference. Experiments on *Genshin Impact* show visually rich, temporally consistent gameplay across diverse environments, weather conditions, and character actions, and applying the same pipeline unchanged to the more photorealistic and visually complex *Honor of Kings: World*—reported in full in Appendix—yields comparably strong results, indicating that our design is not tied to a particular art style.

AI USE STATEMENT

In this work, generative AI tools were used mainly to assist translating text and checking grammar. We have reviewed all AI-assisted work. We take responsibility for the final content of this work.

ETHICS STATEMENT

This work uses gameplay footage captured from two commercially released titles, *Genshin Impact* and *Honor of Kings: World*, solely for academic research. No personally identifiable information was collected; all capture was performed by automated scripts without involving human subjects. The generated gameplay is intended for research purposes and does not reproduce copyrighted assets verbatim. Like any high-fidelity generative model, the system could in principle be misused to produce deceptive or misleading game footage; we encourage downstream users to apply appropriate disclosure when sharing model-generated content.

REPRODUCIBILITY STATEMENT

Full implementation details—model architecture, training stages, hyperparameters, and inference configuration—are provided in Section 4. The data collection pipeline, including action-video synchronization and spatially uniform map coverage, is described in detail in Appendix A. The evaluation protocol, including test-set construction, VBench metric configuration, and baseline evaluation conditions, is specified in Section 4. Extended qualitative and quantitative results on *Honor of Kings: World*, together with ablation studies, are provided in Appendix C.

REFERENCES

Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*, 2025.

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.

Omer Bar-Tal, Hila Chefer, Omer Tov, Charles Herrmann, Roni Paiss, Shiran Zada, Ariel Ephrat, Junhwa Hur, Guanghui Liu, Amit Raj, et al. Lumiere: A space-time diffusion model for video generation. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

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*, 2023.

Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Leo Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, et al. Video generation models as world simulators. *OpenAI Blog*, 1(8):1, 2024.

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

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

Junyi Chen, Haoyi Zhu, Xianglong He, Yifan Wang, Jianjun Zhou, Wenzheng Chang, Yang Zhou, Zizun Li, Zhoujie Fu, Jiangmiao Pang, et al. Deepverse: 4d autoregressive video generation as a world model. *arXiv preprint arXiv:2506.01103*, 2025.

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.

LightX2V Contributors. Lightx2v: Light video generation inference framework. *GitHub repository*, 5, 2025.

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.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Lol: Longer than longer, scaling video generation to hour. *arXiv preprint arXiv:2601.16914*, 2026.

Etched Decart, Quinn McIntyre, Spruce Campbell, Xinlei Chen, and Robert Wachen. Oasis: A universe in a transformer. *URL: https://oasis-model. github. io*, 2(3):6, 2024.

Shenyuan Gao, William Liang, Kaiyuan Zheng, Ayaan Malik, Seonghyeon Ye, Sihyun Yu, Wei-Cheng Tseng, Yuzhu Dong, Kaichun Mo, Chen-Hsuan Lin, et al. Dreamdojo: A generalist robot world model from large-scale human videos. *arXiv preprint arXiv:2602.06949*, 2026.

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

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Fei-Fei Li, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. In *European Conference on Computer Vision*, pp. 393–411. Springer, 2024.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

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.

Danijar Hafner, Wilson Yan, and Timothy Lillicrap. Training agents inside of scalable world models. *arXiv preprint arXiv:2509.24527*, 2025.

Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, Fei Kang, Biao Jiang, Mengyin An, Yangyang Ren, et al. Matrix-game 2.0: An open-source real-time and streaming interactive world model. *arXiv preprint arXiv:2508.13009*, 2025.

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

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.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *Advances in Neural Information Processing Systems*, 38:167283–167308, 2026.

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

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

Team HunyuanWorld. Hy-world 1.5: A systematic framework for interactive world modeling with real-time latency and geometric consistency. *arXiv preprint*, 2025.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Black Forest Labs. FLUX.2: Frontier Visual Intelligence. https://bfl.ai/blog/flux-2, 2025a.

World Labs. Marble. https://www.worldlabs.ai/blog/marble-world-model, 2025b. Accessed: 2026-07-20.

Xiaofeng Mao, Zhen Li, Chuanhao Li, Xiaojie Xu, Kaining Ying, and Kaipeng Zhang. Yume1. 5: A text-controlled interactive world generation model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7752–7761, 2026.

OpenAI. Sora: Video generation models as world simulators. https://openai.com/index/video-generation-models-as-world-simulators/, 2024. Accessed: 2026-07-20.

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.

Georgy Savva, Oscar Michel, Daohan Lu, Suppakit Waiwitlikhit, Timothy Meehan, Dhairya Mishra, Srivats Poddar, Jack Lu, and Saining Xie. Solaris: Building a multiplayer video world model in minecraft. *arXiv preprint arXiv:2602.22208*, 2026.

Team Seedance, De Chen, Liyang Chen, Xin Chen, Ying Chen, Zhuo Chen, Zhuowei Chen, Feng Cheng, Tianheng Cheng, Yufeng Cheng, et al. Seedance 2.0: Advancing video generation for world complexity. *arXiv preprint arXiv:2604.14148*, 2026.

Skywork AI Matrix-Game Team. Matrix-game 3.0: Real-time and streaming interactive world model with long-horizon memory. Technical report, 2026. URL https://github.com/SkyworkAI/Matrix-Game/blob/main/Matrix-Game-3/assets/pdf/report.pdf.

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*, 2025a.

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 model. *arXiv preprint*, 2025b.

Junshu Tang, Jiacheng Liu, Jiaqi Li, Longhuang Wu, Haoyu Yang, Penghao Zhao, Siruis Gong, Xiang Yuan, Shuai Shao, Linfeng Zhang, et al. Hunyuan-gamecraft-2: Instruction-following interactive game world model. *arXiv preprint arXiv:2511.23429*, 2025.

Kling Team, Jialu Chen, Yuanzheng Ci, Xiangyu Du, Zipeng Feng, Kun Gai, Sainan Guo, Feng Han, Jingbin He, Kang He, et al. Kling-omni technical report. *arXiv preprint arXiv:2512.16776*, 2025a.

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

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.

Hansi Teng, Hongyu Jia, Lei Sun, Lingzhi Li, Maolin Li, Mingqiu Tang, Shuai Han, Tianning Zhang, WQ Zhang, Weifeng Luo, et al. Magi-1: Autoregressive video generation at scale. *arXiv preprint arXiv:2505.13211*, 2025.

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

Zehan Wang, Tengfei Wang, Haiyu Zhang, Xuhui Zuo, Junta Wu, Haoyuan Wang, Wenqiang Sun, Zhenwei Wang, Chenjie Cao, Hengshuang Zhao, et al. Worldcompass: Reinforcement learning for long-horizon world models. *arXiv preprint*, 2026.

Jiannan Xiang, Yi Gu, Zihan Liu, Zeyu Feng, Qiyue Gao, Yiyan Hu, Benhao Huang, Guangyi Liu, Yichi Yang, Kun Zhou, et al. Pan: A world model for general, interactable, and long-horizon world simulation. *arXiv preprint arXiv:2511.09057*, 2025.

Mingyu Yang, Junyou Li, Zhongbin Fang, Sheng Chen, Yangbin Yu, Qiang Fu, Wei Yang, and Deheng Ye. Playable game generation. *arXiv preprint arXiv:2412.00887*, 2024.

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.

Deheng Ye, Fangyun Zhou, Jiacheng Lv, Jianqi Ma, Jun Zhang, Junyan Lv, Junyou Li, Minwen Deng, Mingyu Yang, Qiang Fu, et al. Yan: Foundational interactive video generation. *arXiv preprint arXiv:2508.08601*, 2025.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6613–6623, 2024.

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

Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6658–6667, 2024.

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5916–5926, 2025a.

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, 2025b.

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

Dian Zheng, Ziqi Huang, Hongbo Liu, Kai Zou, Yinan He, Fan Zhang, Yuanhan Zhang, Jingwen He, Wei-Shi Zheng, Yu Qiao, and Ziwei Liu. VBench-2.0: Advancing video generation benchmark suite for intrinsic faithfulness. *arXiv preprint arXiv:2503.21755*, 2025.

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

A Data Collection Details

This section expands on the data collection procedure summarized in Section 4.

Automated capture with randomized control. Training a streaming world model at AAA fidelity requires large-scale gameplay video paired with frame-aligned action labels. Manual recording is labor-intensive, scales poorly, and is biased toward the habits of individual players, making it infeasible to cover the vast state space of a modern open world. We therefore drive collection with a Python script running inside the OBS Studio scripting runtime. At each control step the script samples an action from the game's interactive space—locomotion keys for moving forward, backward, left, and right, together with relative mouse displacement for camera rotation—and injects it into the running game instance as a synthetic input event. Randomizing the control sequence rather than replaying scripted trajectories exposes the model to a broad distribution of motion patterns and viewpoint changes, including transitions that a human player would rarely produce.

Action-video synchronization. Because the same script both emits the actions and controls the recorder, every action is written to a log with the timestamp at which it was dispatched, and this log is aligned against the OBS recording clock. Each captured frame is thus annotated with the control signal in effect when it was rendered, giving the frame-aligned $(\text{frame}, \text{action})$ pairs required for action-conditioned training. This removes the need for post-hoc action estimation from pixels, which is a common source of label noise in gameplay datasets scraped from video platforms.

Spatially uniform coverage. Running the randomized policy from a single spawn point concentrates samples around that location, since random walks explore slowly and the resulting data overrepresents a small neighborhood. To keep coverage uniform, we partition the game map into disjoint regions and launch independent capture sessions in each of them, so that terrain types, architectural styles, and lighting conditions across the world are represented in comparable proportion. This decouples spatial coverage from the exploration speed of the random policy and prevents the model from overfitting to a few densely sampled areas.

Corpus statistics. For *Genshin Impact* we collect 11K clips totalling 20M frames at 30 fps. For *Honor of Kings: World*, whose scenes are visually more complex and geometrically denser, we scale the corpus up to 33K clips and 54M frames. Every frame in both corpora is annotated with its action label and environmental context (weather and time-of-day).

B More Results on Genshin Impact

Comparison with state-of-the-art approaches. Figure 7 extends the comparison of Figure 5 to additional starting frames on *Genshin Impact*. Compared with HY-World1.5 HunyuanWorld (2025), our method keeps the imagery free of collapse over long horizons, whereas HY-World1.5 produces repetitive sky textures in the last two cases, which markedly degrades visual quality. Lingbot Team et al. (2026) starts to darken from around 30 s in every case. Matrix Game 3.0 Skywork AI Matrix-Game Team (2026) becomes overexposed as early as 10 s and develops grid-like artifacts (see the last case). YUME Mao et al. (2026) quickly switches to photorealistic live-action content that deviates substantially from the given starting frame. Across all of them our method outperforms existing state-of-the-art approaches by a large margin.

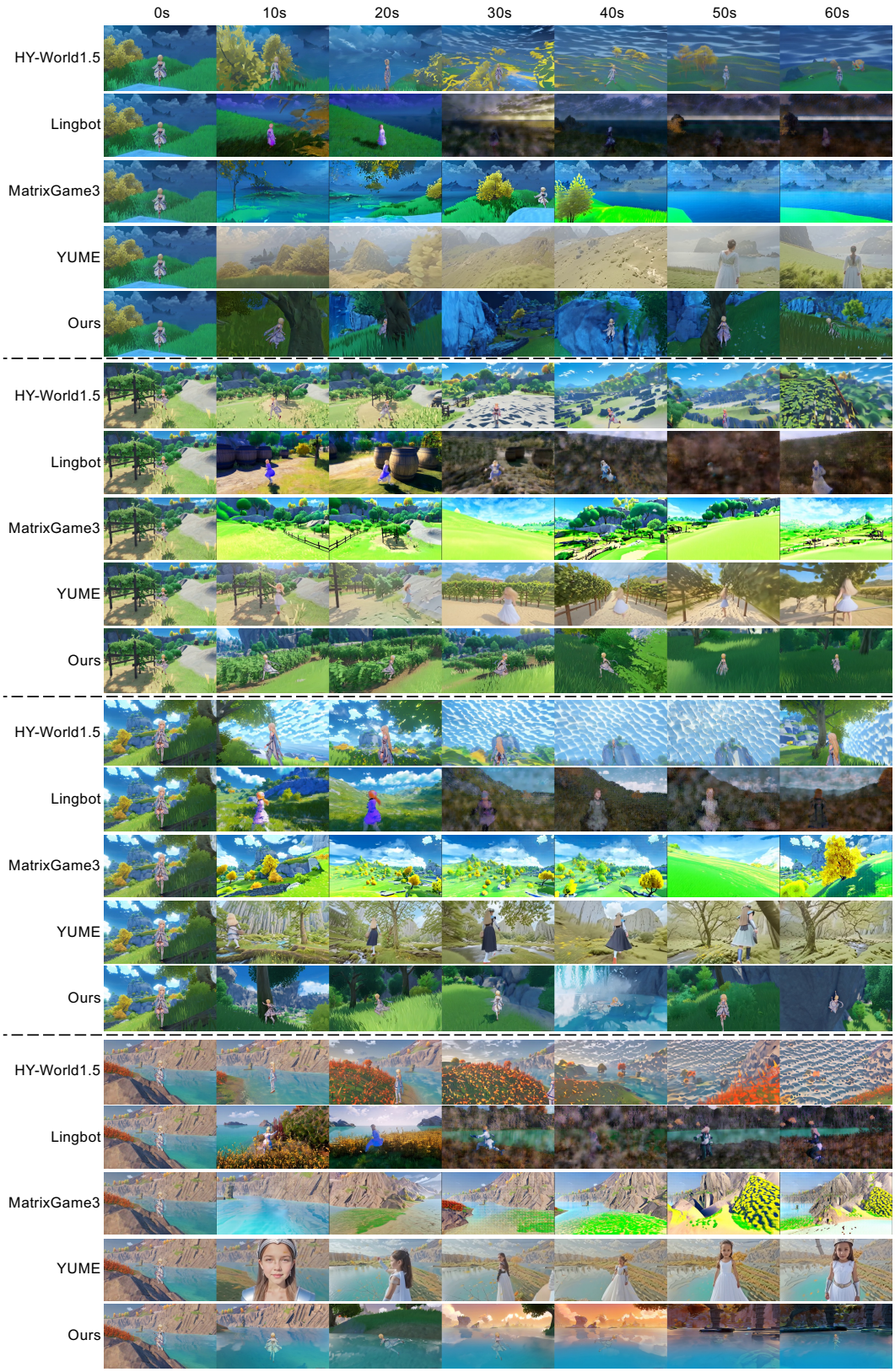

Figure 7: **Qualitative comparison with state-of-the-art approaches on *Genshin Impact*.** Extended version of Figure 5 in the main paper, covering additional starting frames. Each block is one starting frame; rows are methods and columns are timestamps over a one-minute rollout.



Figure 8: One hour of continuous play on *Genshin Impact*

Infinite-length gameplay. Most existing methods collapse within the first 30 seconds of rollout, whereas ours shows no visual collapse after one minute. To stress-test this capability, we play continuously for one hour. As shown in Figure 8, our method neither suffers catastrophic collapse nor gradual degradation over the full hour, confirming its infinite-length gameplay capability.

C Results on *Honor of Kings: World*

We use *Genshin Impact* as the primary testbed in the main paper because it enjoys a far broader player base, which makes its environments, characters, and mechanics widely recognizable and therefore makes our results easy for readers to judge. Its art direction, however, is deliberately stylized and cartoon-like: surfaces are flatly shaded, textures are comparatively clean, and scene clutter is limited. It is consequently not representative of the heavily detailed, physically-based rendering that characterizes contemporary AAA titles, and strong performance on it alone would leave open whether our framework can cope with genuinely photorealistic content.

To close this gap, we further apply the entire pipeline—data collection, training, and streaming inference—to *Honor of Kings: World*, whose markedly more photorealistic rendering, denser geometric and textural detail, and visually more complex scenes make it a substantially harder target for streaming generation. We collect data with the same OBS-based pipeline described in Appendix A and train with an unchanged model architecture and training recipe.

Under this setting we obtain strong results on the harder domain: the model sustains stable streaming generation, generalizes to out-of-distribution initial frames and unseen artistic styles, respects physical priors such as gravity and collision in scenarios absent from training, and remains free of collapse over an hour of continuous play. It also compares favorably against state-of-the-art baselines both qualitatively and on VBench metrics. That our design carries over to a substantially more demanding title without modification indicates that the proposed components are not tied to the visual characteristics of any particular game. The remainder of this section reports the full set of visualizations, comparisons, and ablations.

C.1 Qualitative Results

In-domain visualization. Figure 9 presents in-distribution results. Our method produces high-quality generations across diverse time-of-day and weather conditions, and faithfully synthesizes fine-grained details such as individual blades of grass, scattered flowers, and wheat stalks in farmland. More notably, as shown in the second and sixth rows, the model exhibits an emergent capacity for *proactive* agent-environment interaction—monsters initiating unprompted attacks, NPCs actively engaging the player avatar—a behavior rarely reported in prior video diffusion models trained on passive gameplay corpora. Enabled by our per-frame noise scheduling, which resolves temporally acute motion transitions, the model captures the millisecond-level causal chain from avatar proximity to monster alertness to attack animation, rather than merely memorizing passive environmental motion such as swaying foliage. Character actions are also well rendered, covering both walking on land and swimming on water surfaces.

Out-of-domain visualization. Figure 10 shows out-of-distribution (OOD) results, where an OOD-generated image resembling the visual style of the game serves as the starting frame. As the character moves, OOD content gradually transitions into in-distribution content. To probe farther OOD generalization, we further condition on initial frames whose artistic style resembles *Avatar: Frontiers of Pandora*, *Black Myth: Wukong*, and *Elden Ring: Shadow of the Erdtree*. As shown in Figure 11, our model exhibits strong zero-shot compatibility across these heterogeneous domains,



Figure 9: **In-distribution generation on *Honor of Kings: World*.** Results across diverse time-of-day and weather conditions, showing fine-grained scene detail and proactive agent-environment interaction.



Figure 10: **Out-of-distribution generation on *Honor of Kings: World*.** Starting from an OOD initial frame, the generated content gradually transitions back into the in-distribution domain as the character moves.



Figure 11: **Cross-style OOD generation on *Honor of Kings: World*.** Given initial frames in out-of-domain artistic styles from various AAA titles (left column), our model produces temporally consistent and visually faithful continuations (subsequent columns) while preserving scene structure.



Figure 12: **Physics prior generalization on *Honor of Kings: World*.** Three OOD cases: *free fall from mountain peaks*, *wind-driven grass swaying*, and *avatar obstacle traversal*.

preserving source structure while extrapolating plausible dynamics and player-environment interaction (notably, the player swims across the river rather than running over it).

Game physics generalization. We validate the generalization of physical priors via three OOD cases: free fall from mountain peaks, wind-driven grass swaying, and avatar obstacle traversal. As shown in Figure 12, generated continuations follow physical laws in all cases despite the absence of semantically matched training samples: free-fall trajectories preserve constant gravitational acceleration and valid collision responses, vegetation swaying aligns with airflow direction, and obstacle traversal maintains kinematic validity without geometry interpenetration. This disentanglement of scene appearance from physical rules suggests that **high-density, temporally and physically consistent data is foundational to building generalizable world models.**



Figure 13: **Qualitative comparison with state-of-the-art approaches on *Honor of Kings: World*.** Counterpart of Figure 7 on the more photorealistic title. Each block is one starting frame; rows are methods and columns are timestamps over a one-minute rollout.



Figure 14: **One hour of continuous play on *Honor of Kings: World*.** Our method sustains stable generation without collapse or degradation over the full hour.

Table 2: **Quantitative comparison with state-of-the-art approaches on *Honor of Kings: World*.**

Method	Subject Consist.	Background Consist.	Motion Smooth.	Imaging Quality	Average
Lingbot Team et al. (2026)	0.6753	0.8778	0.9813	0.4014	0.7339
YUME Mao et al. (2026)	0.7059	0.8641	0.9842	0.6492	0.8008
Matrix Game 3.0 Skywork AI Matrix-Game Team (2026)	0.6890	0.8870	**0.9869**	0.6700	0.8082
HY-World1.5 HunyuanWorld (2025)	0.7310	0.8934	0.9866	0.5534	0.7911
Ours	**0.7328**	**0.9018**	0.9746	**0.6927**	**0.8255**

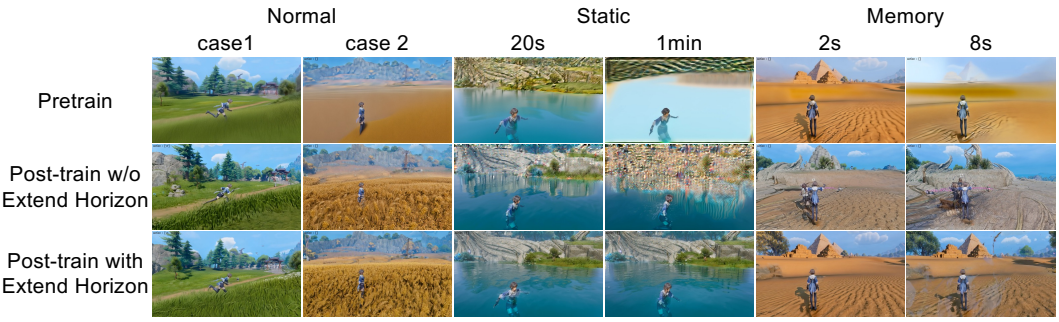

Figure 15: **Ablation study on *Honor of Kings: World*.** Pre-trained, post-trained, and full models (extended training length and memory) across three scenarios: *Normal* navigation in high-density (Case 1) and low-density (Case 2) scenes; *Static* standstill for 20 s and 1 min; and *Memory* retention after turning the camera away for 2 s/8 s and back.

Comparison with state-of-the-art approaches. We compare against HY-World1.5 HunyuanWorld (2025), Lingbot Team et al. (2026), Matrix Game 3.0 Skywork AI Matrix-Game Team (2026), and YUME Mao et al. (2026), generating 1800 frames of random actions (1 minute at 30 fps) from each of a set of randomly selected starting frames. As shown in Figure 13, the baselines fail in two characteristic modes. *Collapse:* HY-World1.5 is stable in the short term but its later frames break down, and Matrix Game 3.0 becomes progressively distorted until the scene turns abstract. *Drift:* Lingbot suffers gradual character degradation with increasingly monotonous backgrounds, while YUME drifts toward photorealistic rather than game-style imagery with noticeable global color shift. Our method instead sustains stable generation across all scenarios without quality degradation over extended horizons.

Infinite-length gameplay. As shown in Figure 14, our method neither suffers catastrophic collapse nor gradual degradation over the full hour, confirming its infinite-length gameplay capability.

C.2 Quantitative Results

Table 2 compares our method against state-of-the-art approaches on *Honor of Kings: World*. Our method attains the highest average score on *Honor of Kings: World* (0.8255), as well as the best imaging quality on each, confirming that the ranking reported in the main paper is not an artifact of averaging across the two domains.

C.3 Ablation Studies

Figure 15 compares our pre-trained model, post-trained model, and the full model with extended training length and memory across three scenarios.

Normal scenes. The pre-trained model generates reasonably in Case 1 (high-density scenes) but degrades in Case 2 (low-density scenes), whereas the post-trained models are consistently good in both, showing that post-training substantially improves generation quality.

Static scenes. Prolonged standstill is especially challenging, as autoregressive error accumulation induces degradation or drift. After 20 s the pre-trained model already degrades in distant regions; after 1 min only the full pipeline—post-training combined with extended training length and memory—remains stable without collapse or drift.

Memory. We perform a large camera rotation, hold for 2 s/8 s, then rotate back. The pre-trained model retains only short-term memory, and post-training with shorter sequences overfits to in-distribution scenes and fails to retain past observations. With extended-length post-training and our hybrid memory, the model reliably recalls the previously observed scene after turning away for 8 seconds.

D Limitations and Future Work

Our model is trained on domain-specific data from a single title, so generalization to unseen game environments, art styles, or mechanics remains limited; extending it to a universal game generation model without per-game fine-tuning is an important open challenge. We also believe the core techniques—efficient high-resolution VAE, controllable streaming diffusion, and long-horizon memory—can serve broader applications beyond playable generation, such as *game-specific visual enhancement* that refines low-quality rendered frames into high-fidelity outputs in real time.