# HELIXKV: ALLOCATING BITS WHERE THEY MATTER FOR AUTOREGRESSIVE VIDEO DIFFUSION

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

## ABSTRACT

Autoregressive video diffusion models enable streaming and long-horizon video generation, but their key–value (KV) caches grow linearly with sequence length, creating an increasingly severe memory bottleneck. Existing KV-cache quantization methods typically rely on uniform or coarse-grained precision allocation. This wastes memory on quantization-tolerant cache regions and leaves fewer bits for regions where precision matters most. To alleviate this problem, we introduce HELIXKV, a training-free KV-cache compression framework that allocates bits where they matter. Specifically, HELIXKV characterizes the impact of KV quantization through model-output distortion and allocates precision along two complementary dimensions. Across attention heads, it assigns head-specific bit-widths to keys and values according to their quantization sensitivity. Along the temporal axis, it further differentiates the precision of recent and historical KV states, allowing each head to retain only the precision required by different parts of its history. Combining these distortion profiles with byte-accurate storage costs, HELIXKV solves a global budgeted allocation problem to jointly allocate KV precision across the head and temporal axes. By compressing quantization-tolerant cache regions and reallocating the saved memory to regions where precision matters more, HELIXKV improves generation quality at a fixed KV-cache budget while preserving the complete KV history. Experiments on Self-Forcing-1.3B and HY-WorldPlay-5B show that HELIXKV achieves $6.97\times$ and $6.61\times$ KV-cache compression, respectively, while retaining 99.9% of the VBench Total score on both models, demonstrating a favorable memory–quality trade-off.

## 1 INTRODUCTION

Recent video diffusion models (Polyak et al., 2024; Yang et al., 2025; Kong et al., 2024; Wan et al., 2025) generate short clips with high visual fidelity and coherent motion, but most operate on fixed spatiotemporal windows with bidirectional attention. Autoregressive (AR) video diffusion (Yin et al., 2025; Huang et al., 2026; Team et al., 2025; Sun et al., 2025; Team et al., 2026) instead generates frame chunks causally, enabling streaming, long-horizon continuation, and interactive world simulation. To avoid recomputing the full history, AR models cache and reuse historical keys and values. The resulting key–value (KV) cache enables efficient causal generation but grows linearly with the retained history and scales with layers, heads, and tokens (Xi et al., 2026; Samuel et al., 2026; Yesiltepe et al., 2026), becoming a major memory bottleneck during long rollouts. Directly truncating the cache saves memory but discards context needed to preserve object identity, scene layout, and motion consistency. Therefore, efficient KV-cache management is critical to both deployment efficiency and long-horizon generation quality.

Existing methods largely follow two directions, as illustrated in Figure 1 (a): *1)* Quantization approaches exploit video redundancy (Xi et al., 2026) or mitigate low-bit attention bias (Tuncer et al., 2026), but typically adopt uniform or coarse-grained precision across layers, attention heads, and temporal positions. Such static allocation overlooks substantial variation in quantization sensitivity, wasting bits on cache regions that can tolerate more aggressive compression. Recency-aware schemes (Ranganath et al., 2026) preserve a shared high-precision window for recent states, but still apply a uniform temporal policy across heads. *2)* Cache-eviction methods (Guo et al., 2026a; Ji et al., 2026; Cai et al., 2026; Chen et al., 2026; Shen et al., 2026) instead exploit head-wise or temporal redundancy through head-dependent retention or by pruning heads, frames, or tokens. However,



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.

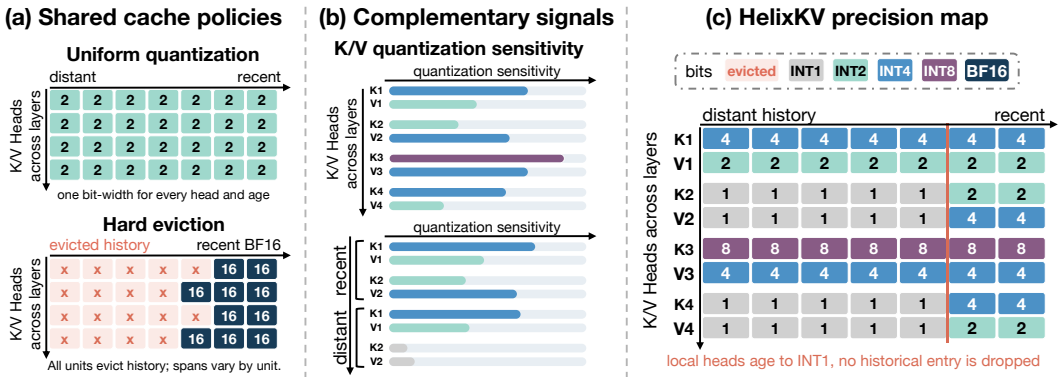



Figure 1: **Motivation and overview of HELIXKV.** (a) Uniform quantization uses one bit-width for all layer–head units and cache ages, whereas hard-eviction methods retain head-dependent recent BF16 windows but discard all earlier states. (b) Quantization sensitivity differs across the K and V states of layer–head units and also between recent and distant history, exposing complementary head- and age-wise precision demands. (c) HELIXKV jointly allocates K/V precision along these two axes: some units retain their assigned precision throughout the history, while others age distant states to an INT1 floor. Unlike eviction, every historical entry remains addressable.

such eviction permanently discards distant states. When a long-horizon video generator revisits a previously observed location, the loss of historical context can lead to inconsistent regeneration, with changes in scene content or spatial layout. Existing methods therefore either allocate precision too coarsely or reduce memory by discarding historical states. To exploit the redundancy overlooked by coarse-grained quantization without resorting to hard eviction, we analyze inference-time precision sensitivity across layer–head units and cache age.

Figure 1 (b) summarizes our central observation: precision requirements vary jointly across layer–head units and cache age. First, *layer–head units exhibit substantially different sensitivities to joint K/V quantization*, and even within the same unit, K and V may require different bit widths. Second, *quantization sensitivity also varies with cache age*. For some units, distant states remain sensitive and should retain their assigned precision, whereas for others, distant states tolerate INT1 quantization even when recent states require substantially higher precision. Together, these observations motivate joint precision allocation across layer–head units and cache age, reducing KV-cache memory while preserving the complete history.

Based on these observations, we introduce HELIXKV, a training-free framework that allocates KV-cache precision along two complementary dimensions: attention head and cache age, as illustrated in Figure 1 (c). First, *Head-wise Precision Allocation (HPA)* profiles the model-output distortion induced by joint K/V quantization to characterize the precision requirement of each head. Second, *Temporal Precision Aging (TPA)* exploits temporal locality by reducing only the distant states of tolerant heads to INT1 while preserving their complete history. To enable efficient aging, TPA represents recent states using an INT1 base with removable refinement bits, allowing distant states to transition to INT1 without requantizing the retained base. Finally, a *Budget-Aware Joint Allocator (BJA)* combines model-output distortion with physical storage cost to jointly determine the K/V precision assignment for each head under a target memory budget. The entire framework is plug-and-play: all model-specific profiles and precision assignments are obtained once through offline calibration, requiring neither model retraining nor online optimization.

Experiments on Self-Forcing-1.3B (Huang et al., 2026) and HY-WorldPlay-5B (Sun et al., 2025) show that HELIXKV achieves $6.97\times$ and $6.61\times$ KV-cache compression, respectively, while retaining 99.9% of the VBench (Huang et al., 2024) Total score on both models. At matched cache costs, HELIXKV consistently improves reconstruction fidelity and perceptual quality over uniform and representative KV-cache quantization baselines, demonstrating a favorable memory–quality trade-off across different model scales and generation settings. Qualitative results further show that the compressed cache preserves visual details, temporal consistency, and long-range content more faithfully under aggressive compression. Moreover, the byte-budget formulation supports flexible operating points across different memory constraints rather than being limited to fixed INT2/INT4 settings, making HELIXKV practical for deployment on devices with varying memory capacities.

## 2 Related Work

**Autoregressive Video Diffusion.** Conventional video diffusion models (Polyak et al., 2024; Yang et al., 2025; Kong et al., 2024; Wan et al., 2025) denoise fixed-length clips with bidirectional temporal attention, whereas autoregressive (AR) models (Yin et al., 2025; Huang et al., 2026; Team et al., 2025; Sun et al., 2025; Team et al., 2026) causally generate frame chunks, enabling streaming and long-horizon video generation. Prior work improves causal generation through few-step distillation and training on self-generated histories to mitigate rollout mismatch (Yin et al., 2025; Zhu et al., 2026; Huang et al., 2026; Guo et al., 2026b), while recent systems (Team et al., 2025; Sun et al., 2025; Team et al., 2026) further extend AR generation to minute-long synthesis and interactive world simulation. To avoid recomputing historical features, AR models cache the keys and values of previously generated chunks. However, the KV cache grows linearly with sequence length as new tokens are accumulated across transformer layers, making cache memory a major bottleneck for long-horizon inference and ultimately constraining the available temporal context (Xi et al., 2026; Samuel et al., 2026; Yesiltepe et al., 2026).

**KV-Cache Compression for Autoregressive Video Diffusion.** Existing methods reduce KV-cache memory by selectively retaining history at the head, frame, or token level (Guo et al., 2026a; Ji et al., 2026; Cai et al., 2026; Chen et al., 2026; Shen et al., 2026), exploiting sparse temporal correspondence (Samuel et al., 2026), or replacing cached states with learned low-rank representations (Yesiltepe et al., 2026). These approaches either discard historical information or introduce architectural modifications. Quantization provides an alternative by reducing the storage cost of each cached state while retaining the full history. Related LLM studies investigate asymmetric K/V quantization, rotation-based quantization, high-precision recent windows, and mixed-precision allocation (Liu et al., 2024; Hooper et al., 2024; Ashkboos et al., 2024; Duanmu et al., 2024; Yang et al., 2024; Zuo et al., 2026). For AR video diffusion, prior work has explored video-aware KV quantization, quantization-induced attention bias, and cache management (Xi et al., 2026; Tuncer et al., 2026; Ranganath et al., 2026). However, existing approaches for AR video diffusion largely apply shared precision across layer–head units or shared temporal precision schedules across heads, overlooking heterogeneity along both dimensions. In contrast, our HelixKV jointly allocates precision across attention heads and cache age, assigning higher precision to quantization-sensitive heads while more aggressively compressing tolerant ones. This targeted allocation preserves the complete KV history and yields higher video fidelity under the same memory budget.

## 3 HelixKV: Allocating Bits Where They Matter

We present HelixKV, a training-free framework for compressing the historical KV cache of AR video diffusion models. HelixKV allocates precision across layer–head units, K/V roles, and cache age to minimize model-output distortion under a target byte budget. Figure 2 illustrates the resulting precision assignments within causal attention. Section 3.1 reviews KV caching and quantization. Section 3.2 presents Head-wise Precision Allocation (HPA), which profiles head-wise sensitivity and constructs K/V precision candidates. Section 3.3 describes Temporal Precision Aging (TPA), which further compresses distant states for temporally local heads. Finally, Section 3.4 introduces the Budget-Aware Joint Allocator (BJA) and the plug-and-play inference workflow.

### 3.1 Preliminaries

**Causal attention with KV caching.** At denoising step $t$ of chunk $c$, layer $l$ produces $\mathbf{Q}_{l,c,t}$, $\mathbf{K}_{l,c,t}$, and $\mathbf{V}_{l,c,t}$. Let $\mathcal{K}_l^{<c}$ and $\mathcal{V}_l^{<c}$ collect the final clean K/V states cached for all preceding chunks, and define $\bar{\mathbf{K}}_{l,c,t} = [\mathcal{K}_l^{<c}; \mathbf{K}_{l,c,t}]$ and $\bar{\mathbf{V}}_{l,c,t} = [\mathcal{V}_l^{<c}; \mathbf{V}_{l,c,t}]$. The attention output is

$$\mathbf{O}_{l,c,t} = \mathrm{softmax}\left(\frac{\mathbf{Q}_{l,c,t}\bar{\mathbf{K}}_{l,c,t}^{\top}}{\sqrt{d}}\right)\bar{\mathbf{V}}_{l,c,t}. \tag{1}$$

Here, $[\cdot;\cdot]$ concatenates tokens. During denoising, $\mathbf{K}_{l,c,t}$ and $\mathbf{V}_{l,c,t}$ are transient and are not written to the cache. After chunk $c$ is fully denoised, only the K/V states computed from its final clean representation are appended once and reused by subsequent chunks. For $N_{<c}$ cached tokens, storing both K and V in BF16 requires $2N_{<c}d\,s_{\mathrm{BF16}}$ bytes, where $s_{\mathrm{BF16}}$ is the number of bytes per BF16

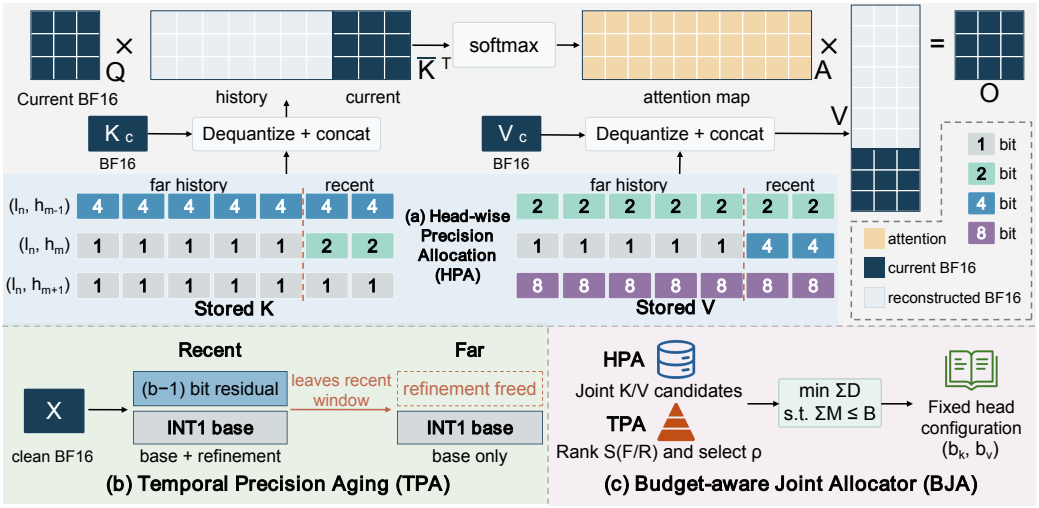



Figure 2: **Overview of HELIXKV within causal attention.** Historical K/V states are reconstructed and concatenated with the current BF16 K/V for attention. (a) HPA assigns K and V precisions across layer and head units. (b) TPA stores recent states as an INT1 base plus refinement bits. Once a state leaves the recent window, its refinement is released while the base is retained, preserving the complete history without requantization. (c) BJA selects one configuration per head by minimizing profiled distortion under a global byte budget using the HPA and TPA candidates.

scalar. This cost grows linearly with the generated context and becomes a major bottleneck for long-horizon generation, motivating KV-cache quantization.

**Quantized KV caching.** Let $\mathbf{X} = \{x_i\}$ denote a group of cached K or V values. A standard $b$-bit affine quantizer maps each value to one of $2^b$ integer levels and reconstructs it as

$$s = \max\left(\frac{x_{\max} - x_{\min}}{2^b - 1}, \epsilon\right), \quad q_i = \mathrm{clip}\left(\left\lfloor \frac{x_i - x_{\min}}{s} \right\rceil, 0, 2^b - 1\right), \quad \widehat{x}_i = x_{\min} + s q_i. \tag{2}$$

Here, $x_{\min}$ and $x_{\max}$ are the group extrema, $s$ is the quantization scale, $q_i$ is the stored $b$-bit code, $\widehat{x}_i$ is the reconstructed value, and $\epsilon$ prevents a zero scale. The procedure is applied independently to each group. Since both packed codes and quantization metadata occupy memory, the final allocation uses the materialized storage cost rather than the nominal bit width alone.

### 3.2 HEAD-WISE PRECISION ALLOCATION

The quantization process above determines how cached tensors are compressed, but not how bits should be allocated. Existing methods typically assign uniform precision at the model level, despite differences in quantization sensitivity across layers and attention heads. We therefore characterize head-wise sensitivity and construct per-head K/V precision candidates for adaptive bit allocation.

**Bits matter differently across heads.** We first examine head-wise precision requirements by quantizing the cached K/V of one layer–head pair at a time to INT2 while keeping all others in BF16. As shown in Figure 3 (a), the ratio between the maximum and minimum attention-output errors across layers has a median of $54.0\times$ and reaches up to $127.5\times$. Substantial variation also exists within each layer: Figure 3 (b) shows a median max/min ratio of $14.7\times$ and a maximum of $74.9\times$. This heterogeneity arises because heads differ in both K/V distributions and functional roles, so the same bit width induces markedly different errors. Moreover, the error is highly concentrated: the top 10% and 40% of heads account for 35% and 79% of the total error, whereas the bottom 40% contribute only 9% (Figure 3 (c)). These results show that uniform precision overprotects many tolerant heads, while a small subset of sensitive heads dominates the degradation, motivating precision allocation at the layer–head level.

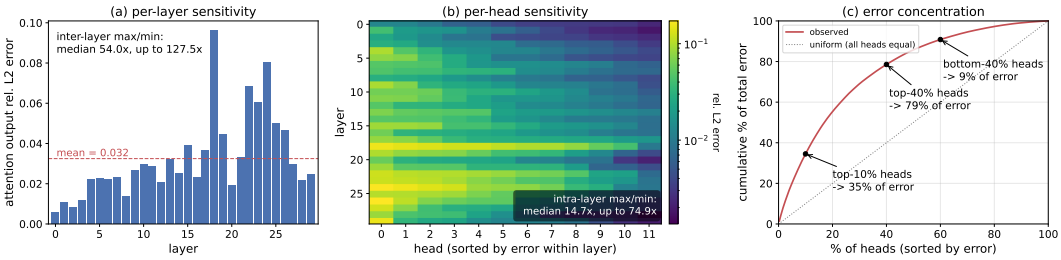



Figure 3: **KV-cache quantization sensitivity varies substantially across attention heads.** We quantize one layer–head pair at a time to INT2 on Self-Forcing while keeping all others in BF16, and measure the resulting attention-output relative $\ell_2$ error. Sensitivity varies strongly across layers and heads, with most distortion concentrated in a small subset of heads, revealing substantial redundancy in uniform precision assignment.

**Head-wise precision allocation.** Guided by these observations, we propose Head-wise Precision Allocation (HPA). Let $\mathcal{B}$ denote the candidate precision set and $\mathcal{C}_{\text{head}}$ the calibration points, where each $(p, c, t)$ specifies a prompt, autoregressive chunk, and denoising step. For each head $(l, h)$ and precision pair $(b_K, b_V) \in \mathcal{B}^2$, we jointly quantize its full historical K/V cache while keeping all other heads in BF16. With $\mathbf{Y}^{\text{BF16}}_{p,c,t}$ denoting the BF16 reference prediction, we define

$$D^{\text{Full}}_{l,h}(b_K, b_V) = \frac{1}{|\mathcal{C}_{\text{head}}|} \sum_{(p,c,t)\in\mathcal{C}_{\text{head}}} \frac{\left\| \mathbf{Y}^{\text{Full},b_K,b_V}_{l,h,p,c,t} - \mathbf{Y}^{\text{BF16}}_{p,c,t} \right\|_F}{\left\| \mathbf{Y}^{\text{BF16}}_{p,c,t} \right\|_F + \epsilon}. \tag{3}$$

Jointly perturbing K and V captures their coupled effect on the model output, while evaluating every candidate from the same BF16 cache avoids cross-candidate contamination. We pair each distortion with its materialized storage cost $M^{\text{Full}}_{l,h}(b_K, b_V)$, including packed codes and quantization metadata, to form a model-specific precision profile for each head. These profiles are then used for budget-aware allocation (Section 3.4), assigning more bits to sensitive heads and fewer to tolerant ones for higher generation quality at the same cache cost.

### 3.3 Temporal Precision Aging

HPA determines how precision is distributed across the model, but assigns a single K/V precision pair to the entire history of each head. This overlooks temporal heterogeneity, as recent and distant states may contribute differently to future predictions, especially as the autoregressive context grows. We therefore characterize quantization sensitivity across cache age and construct age-aware precision candidates that allocate precision differently to recent and distant states while preserving the full history.

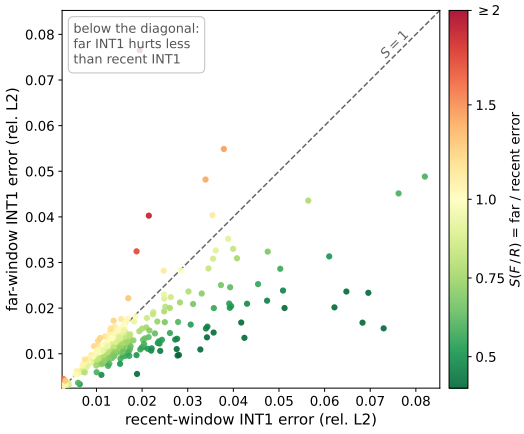



Figure 4: **Distant history is more quantization-tolerant for most heads.** Each point represents a layer–head pair in Self-Forcing, with the recent-region error on the horizontal axis and the far-region error on the vertical axis.

**Precision requirements vary with cache age.** We probe temporal sensitivity on BF16 trajectories by quantizing one layer–head pair at a time. Using Self-Forcing as an example, at chunk $n$, we define the far and recent regions as $\mathcal{F}_n = [0, n-4)$ and $\mathcal{R}_n = [n-4, n)$, respectively. The four-chunk recent window corresponds to half of the eight-chunk quantization block (Xi et al., 2026), while the window size is configurable, allowing distant history to be compressed more aggressively for temporally local heads. For each head, we independently quantize the K/V in either region to INT1 while keeping the rest in BF16, and restore the cache after each probe.

Averaging the model-output relative $\ell_2$ error over prompts gives $D^{\text{far}}$ and $D^{\text{recent}}$. We define $S(F/R) = D^{\text{far}}/D^{\text{recent}}$, where $S(F/R) < 1$ indicates greater tolerance to quantizing distant history. As shown in Figure 4, temporal sensitivity varies substantially across heads, with 76% satisfying $S(F/R) < 1$. For these heads, distant states can use lower precision, freeing budget for recent states or other sensitive heads, while the remaining heads still depend strongly on accurate long-range history. This heterogeneity rules out uniform precision aging and motivates head-specific temporal allocation.

**Temporal precision aging.** Guided by this observation, we propose *Temporal Precision Aging* (TPA), which ranks layer–head pairs by $S(F/R)$ and selects the lowest-scoring $\rho$ fraction as $\mathcal{H}_{\text{TPA}}$. For each selected head, the far-region K/V are fixed to INT1, while candidate precisions $(b_K, b_V) \in \mathcal{B}^2$ are profiled for the recent region. For all other heads, the same candidates are profiled over the full history. Each candidate is evaluated on the same BF16 calibration trajectories using model-output relative $\ell_2$ error and materialized storage cost. As in HPA, these profiles are used for budget-aware allocation under a fixed memory budget (Section 3.4).

A naive transition from $b$ bits to INT1 requires dequantizing and requantizing each chunk as it leaves the recent window, introducing online overhead and compounding quantization error. TPA instead encodes each K/V tensor $\mathbf{X}$ at cache-write time as a shared INT1 base plus a removable refinement. Let $\mathcal{Q}_b$ and $\mathcal{D}_b$ denote $b$-bit quantization and dequantization. We define

$$\mathbf{X}_{\text{ref}} = \mathcal{D}_{b-1}(\mathcal{Q}_{b-1}(\mathbf{X} - \mathbf{X}_{\text{base}})), \qquad \mathbf{X}_{\text{base}} = \mathcal{D}_1(\mathcal{Q}_1(\mathbf{X})), \qquad b > 1. \tag{4}$$

A state from source chunk $\tau$ is reconstructed as

$$\widehat{\mathbf{X}}_\tau^{(b)} = \mathbf{X}_{\text{base}} + \begin{cases} \mathbf{X}_{\text{ref}}, & \tau \in \mathcal{R}_n, \\ \mathbf{0}, & \tau \in \mathcal{F}_n. \end{cases} \tag{5}$$

Thus, recent and far regions share the same INT1 base, and aging simply discards the refinement once a state enters the far region. When $b = 1$, no refinement is stored. This avoids cache eviction and online requantization. By extending precision allocation along cache age, TPA exploits temporal heterogeneity to improve long-horizon consistency under the same memory budget, while preserving the complete KV history unlike eviction-based methods.

### 3.4 Budget-Aware End-to-End Inference

**Budget-Aware Joint Allocator (BJA).** Given the offline profiles constructed by HPA and TPA, BJA represents each precision choice for head $(l, h)$ as $\gamma = (b_K, b_V) \in \mathcal{B}^2$. For heads in $\mathcal{H}_{\text{TPA}}$, $\gamma$ specifies the recent-region precision while the far region remains INT1; otherwise, it applies to the full history. Let $D_{l,h}(\gamma)$ denote the measured model-output distortion and $M_{l,h}(\gamma)$ the materialized storage cost at the target cache length. Given a cache budget $B$, BJA selects one precision choice per head by solving

$$\min_{\gamma_{l,h}} \sum_{l=1}^{L} \sum_{h=1}^{H_l} D_{l,h}(\gamma_{l,h}) \quad \text{s.t.} \quad \sum_{l=1}^{L} \sum_{h=1}^{H_l} M_{l,h}(\gamma_{l,h}) \le B, \qquad \gamma_{l,h} \in \mathcal{B}^2. \tag{6}$$

This is a multiple-choice knapsack problem: each head contributes exactly one joint K/V precision candidate. BJA removes dominated candidates and solves the remaining discrete problem with dynamic programming. By varying $B$, BJA supports finely adjustable global compression ratios (Figure 6) beyond fixed uniform INT2/INT4 operating points, allowing the same model to adapt to devices with different memory capacities.

**Plug-and-play generation.** BJA fixes precision assignments before generation. Full-history heads retain their allocated $(b_K, b_V)$ throughout the rollout, while TPA heads store recent chunks with refinement bits and discard the refinement once they leave the recent window, retaining only the INT1 base. At each denoising step, reconstructed historical K/V states are concatenated with the current BF16 states and processed by the attention operation in Equation 1. All profiles and assignments are computed once offline, requiring no retraining, architectural modification, or online optimization.

Table 1: **Quantitative comparison on different autoregressive diffusion models with KV-cache quantization baselines.** The lower-right label indicates the preset quantization precision or compression ratio. The best quantized results are marked in **bold**.

| Method | Compression Ratio (BF16) ↑ | Fidelity Scores | | | VBench / VBench-I2V Scores | | |
|---|---|---|---|---|---|---|---|
| | | PSNR ↑ | SSIM ↑ | LPIPS ↓ | Total ↑ | Quality ↑ | Semantic / I2V ↑ |
| *Self-Forcing-1.3B* (Huang et al., 2026) | 1.00× | - | - | - | 81.30 | 84.00 | 70.50 |
| RTN $_{\text{INT2}}$ | 6.40× | 18.85 | 0.6249 | 0.3388 | 79.61 | 82.05 | 69.86 |
| KIVI $_{\text{INT2}}$ (Liu et al., 2024) | 6.40× | 10.22 | 0.2356 | 0.6845 | 75.63 | 78.78 | 63.00 |
| QuaRot-KV $_{\text{INT2}}$ (Ashkboos et al., 2024) | 6.40× | 18.69 | 0.6243 | 0.3200 | 80.23 | 82.79 | 70.01 |
| QVG $_{\text{INT2}}$ (Xi et al., 2026) | **6.97**× | 23.34 | 0.7644 | 0.1896 | 81.13 | 83.86 | 70.22 |
| QVG $_{\text{INT4}}$ (Xi et al., 2026) | 3.72× | 31.64 | 0.9171 | 0.0943 | 81.24 | 83.90 | **70.62** |
| HELIXKV$_{6.97\times}$ (Ours) | **6.97**× | 26.44 | 0.8474 | 0.1379 | 81.23 | 83.94 | 70.37 |
| HELIXKV$_{3.72\times}$ (Ours) | 3.72× | **32.88** | **0.9315** | **0.0811** | **81.28** | **83.99** | 70.42 |
| *HY-WorldPlay-5B* (Sun et al., 2025) | 1.00× | - | - | - | 88.42 | 81.75 | 95.09 |
| RTN $_{\text{INT2}}$ | 6.40× | 16.64 | 0.4213 | 0.4468 | 84.72 | 76.87 | 92.58 |
| KIVI $_{\text{INT2}}$ (Liu et al., 2024) | 6.40× | 9.83 | 0.1648 | 0.6980 | 76.19 | 74.04 | 78.34 |
| QuaRot-KV $_{\text{INT2}}$ (Ashkboos et al., 2024) | 6.40× | 15.11 | 0.3717 | 0.5032 | 84.85 | 77.08 | 92.62 |
| QVG $_{\text{INT2}}$ (Xi et al., 2026) | **6.61**× | 18.24 | 0.5053 | 0.3448 | 87.19 | 80.73 | 93.66 |
| QVG $_{\text{INT4}}$ (Xi et al., 2026) | 3.62× | 23.19 | 0.7872 | 0.1797 | 88.35 | 81.76 | 94.94 |
| HELIXKV$_{6.61\times}$ (Ours) | **6.61**× | 21.12 | 0.6968 | 0.2231 | 88.29 | 81.59 | 94.98 |
| HELIXKV$_{3.62\times}$ (Ours) | 3.62× | **23.91** | **0.8014** | **0.1703** | **88.40** | **81.77** | **95.04** |

# 4 EXPERIMENTS

## 4.1 IMPLEMENTATION DETAILS

**Model configurations.** We evaluate HELIXKV on two representative open-source autoregressive video diffusion models: Self-Forcing-1.3B (Huang et al., 2026), a text-to-video model built on Wan2.1-T2V-1.3B (Wan et al., 2025), and HY-WorldPlay-5B (Sun et al., 2025), a Wan-based world model. Both use their released checkpoints and native sampling configurations. Our implementation is built upon the released QVG framework (Xi et al., 2026). We keep model weights, activations, and current-chunk K/V in BF16, compressing only completed historical K/V states. We compare against the uncompressed BF16 cache and four representative quantization baselines: Round-to-Nearest (RTN), KIVI (Liu et al., 2024), QuaRot-KV (Ashkboos et al., 2024), and QVG (Xi et al., 2026). Unless otherwise stated, HPA uses $\mathcal{B} = \{1, 2, 3, 4, 5\}$ under the INT2-equivalent budget and $\mathcal{B} = \{1, 2, 3, 4, 5, 6, 7, 8\}$ under the INT4-equivalent budget. Model-specific profiles are computed once from 50 held-out prompts and fixed throughout evaluation without training or online adaptation. All experiments are conducted on NVIDIA H100 GPUs with 80GB memory.

**Evaluation.** For Self-Forcing, we generate videos using the standard VBench prompts (Huang et al., 2024). For HY-WorldPlay, we use the action-conditioned cases from Ying et al. (2026), preserving the provided initial images, scene descriptions, and camera/action trajectories. We evaluate reconstruction fidelity against paired BF16 outputs using PSNR, SSIM, and LPIPS. Since the initial portion of each rollout is generated before KV quantization takes effect and is therefore identical across methods, we exclude these frames from metric computation and evaluate only the quantization-affected portion. Perceptual quality is measured with VBench (Huang et al., 2024) for Self-Forcing and VBench-I2V (Huang et al., 2025) for HY-WorldPlay. We also report the realized KV-cache compression ratio. Within each model, all methods use identical inputs, actions, random seeds, sampling schedules, and rollout lengths.

## 4.2 COMPARISON WITH STATE-OF-THE-ART METHODS

**Quantitative comparison.** Table 1 compares HELIXKV with KV-cache quantization methods on text-conditioned video generation and action-conditioned world modeling. We match our operating points to the materialized cache costs of QVG INT2 and INT4. Across Self-Forcing and HY-WorldPlay, HELIXKV consistently achieves higher reconstruction fidelity than QVG, with larger gains under aggressive compression. It also outperforms RTN, KIVI, and QuaRot-KV across PSNR, SSIM, and LPIPS at these operating points while using less cache memory. Perceptual metrics follow a similar trend: VBench scores on Self-Forcing are largely preserved or improved, while all




Figure 5: **Qualitative comparison with previous competing methods.** Zoom in for more details.

Table 2: **Ablation studies on Self-Forcing.** Left: effects of the sensitivity target, allocation granularity, and candidate precision space in HPA. Right: effects of the selected-head fraction and temporal treatment in TPA. $\rho$ denotes the lowest-scoring fraction of layer–head pairs ranked by $S(F/R)$ in ascending order. All variants use a matched $6.97\times$ KV-cache compression ratio.

| Method | Bits | PSNR ↑ | SSIM ↑ | LPIPS ↓ |
|---|---|---|---|---|
| QVG (INT2) | $\{2\}$ | 20.19 | 0.6786 | 0.2644 |
| Layer-wise (attn.-out.) | $\{1, 2, 4\}$ | 20.37 | 0.6891 | 0.2623 |
| Layer-wise (model-out.) | $\{1, 2, 4\}$ | 20.49 | 0.6964 | 0.2521 |
| Head-wise (model-out.) | $\{1, 2, 4\}$ | 21.50 | 0.7333 | 0.2277 |
| Head-wise (model-out.) | $\{1, \ldots, 5\}$ | **21.80** | **0.7497** | **0.2153** |
| Head-wise (model-out.) | $\{1, \ldots, 8\}$ | **21.80** | **0.7497** | **0.2153** |

| Method | $\rho$ | PSNR ↑ | SSIM ↑ | LPIPS ↓ |
|---|---|---|---|---|
| HPA (full history) | – | 21.80 | 0.7497 | 0.2153 |
| Hard eviction | 10% | 18.02 | 0.6489 | 0.3233 |
| TPA (requant.) | 10% | 21.93 | 0.7413 | 0.2137 |
| TPA (base–ref.) | 10% | 22.19 | 0.7582 | 0.2114 |
| TPA (base–ref.) | 20% | **22.76** | **0.7667** | **0.2026** |
| TPA (base–ref.) | 30% | 22.65 | 0.7620 | 0.2041 |

reported VBench-I2V scores on HY-WorldPlay improve over QVG. Overall, HELIXKV provides a better memory–quality trade-off across both tasks.

**Qualitative comparison.** Figure 5 shows temporally separated frames from paired BF16 and quantized rollouts on Self-Forcing and HY-WorldPlay under an INT2-equivalent cache budget. In the skyline sequence, RTN, KIVI, and QuaRot-KV blur or distort building structures, while QVG better preserves the scene but still exhibits structural overlap and appearance changes in later frames. HELIXKV more faithfully preserves landmark silhouettes and scene details. The canyon sequence is more challenging: LLM-oriented methods degrade more severely, while QVG still shows changes in geometry, river trajectory, and visible artifacts. In contrast, HELIXKV remains closer to the BF16 reference in scene layout and fine-grained structure throughout the rollout.

### 4.3 ABLATION STUDY

We conduct all ablations on Self-Forcing-1.3B (Huang et al., 2026) using the same 200 randomly sampled VBench prompts (Huang et al., 2024) for efficient evaluation across all variants. Unless otherwise specified, quantized variants are evaluated at a matched $6.97\times$ KV-cache compression ratio relative to BF16, with all other generation settings held constant.

**Sensitivity target, allocation granularity, and precision space.** Table 2 (left) studies three HPA factors. Model-output profiling propagates quantization errors to the final denoising prediction and better captures their generation impact than attention-output profiling, improving PSNR/SSIM/LPIPS from 20.37/0.6891/0.2623 to 20.49/0.6964/0.2521. Head-wise allocation further raises PSNR to 21.50 by allowing heads with different sensitivities to use different precisions. Expanding $\{1, 2, 4\}$ to $\{1, \ldots, 5\}$ improves PSNR to 21.80 through finer precision choices, while $\{1, \ldots, 8\}$ yields the same assignment. Thus, 5 bits already cover the useful precision range under the INT2-equivalent budget.

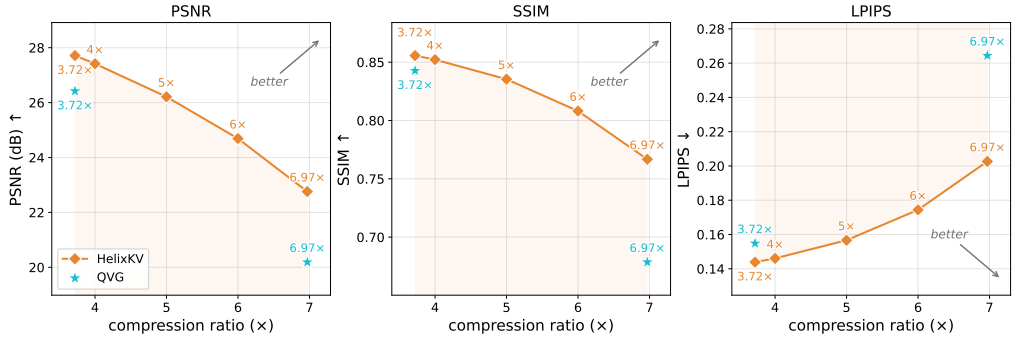



Figure 6: **Quality under varying cache budgets enabled by BJA on Self-Forcing.** BJA exposes intermediate operating points rather than being restricted to a small set of uniform bit widths.

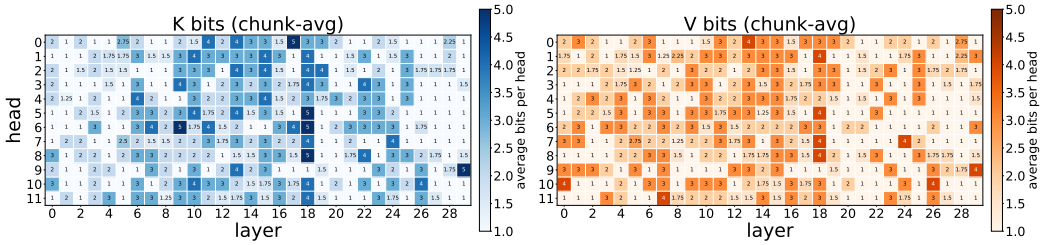



Figure 7: **Structure of the BJA allocation on Self-Forcing.** Each cell shows the bit width of one head at one layer, averaged across cached chunks under a target average precision of 2 bits. The left and right panels show K and V allocations, respectively. Fractional values arise from different precisions assigned to recent and distant chunks, with darker cells indicating higher average precision.

**Temporal head selection and aging mechanisms.** Table 2 (right) studies TPA's head fraction and aging mechanism. We rank layer–head pairs by $S(F/R)$ and apply aging to the lowest-scoring $\rho$ fraction. Evicting distant states from even the lowest 10% causes a clear fidelity drop, indicating that weakly attended history should still be preserved. At $\rho = 10\%$, base-and-refinement aging consistently outperforms direct requantization by retaining the original INT1 base. Performance peaks at $\rho = 20\%$ and slightly declines at 30%, as more precision-sensitive heads are included.

**Budget adaptation and allocation structure.** Figure 6 shows how BJA translates a global byte budget into flexible operating points. Across compression ratios from $3.72\times$ to $6.97\times$, all three fidelity metrics vary smoothly with the budget, enabling finer control than fixed INT2/INT4 settings. Figure 7 visualizes BJA's nonuniform precision assignments across the model. The resulting pattern shows that BJA exploits heterogeneity across layers, heads, K/V roles, and cache age, reclaiming memory from tolerant units and reallocating it to more sensitive ones.

## 5 Conclusion

We presented HelixKV, a training-free framework for compressing the growing KV cache of autoregressive video diffusion models while preserving the full history. We show that quantization sensitivity varies substantially across attention heads, K/V roles, and cache age. Accordingly, HPA captures head-wise sensitivity, TPA exploits temporal heterogeneity through precision aging, and BJA jointly allocates precision under exact byte budgets. Experiments on Self-Forcing-1.3B and HY-WorldPlay-5B demonstrate improved fidelity and perceptual quality over uniform precision and representative KV-cache quantization baselines at matched cache costs. Overall, our results show that KV-cache precision is a structured resource that can be allocated where it matters most, enabling more memory-efficient long-horizon video generation and interactive world modeling.

**Limitations and Future Work.** Our current method is training-free and relies on offline precision profiling. Future work will integrate precision allocation into model training to further improve quantization robustness and generation quality.

### AI USE STATEMENT

In this work, we used generative AI tools to improve the clarity and readability of the manuscript and to assist with editing research code. The authors reviewed all AI-assisted changes and take full responsibility for the final manuscript, implementation, and results.

## REFERENCES

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. Quarot: Outlier-free 4-bit inference in rotated llms. *Advances in Neural Information Processing Systems*, 37:100213–100240, 2024.

Peiliang Cai, Evelyn Zhang, Jiacheng Liu, Hao Lin, Ruiqi Zhang, Weile Mo, Yue Ma, Shikang Zheng, Jiehang Huang, Dongrui Liu, et al. Focused forcing: Content-aware per-frame kv selection for efficient autoregressive video diffusion. *arXiv preprint arXiv:2605.18346*, 2026.

Jiayu Chen, Junbei Tang, Wenbiao Zhao, Maoliang Li, Jiayi Luo, Zihao Zheng, Jiawei Yang, Guojie Luo, and Xiang Chen. Pyramid forcing: Head-aware pyramid kv cache policy for high-quality long video generation. *arXiv preprint arXiv:2605.13111*, 2026.

Haojie Duanmu, Zhihang Yuan, Xiuhong Li, Jiangfei Duan, Xingcheng Zhang, and Dahua Lin. Skvq: Sliding-window key and value cache quantization for large language models. *arXiv preprint arXiv:2405.06219*, 2024.

Hang Guo, Zhaoyang Jia, Jiahao Li, Bin Li, Yuanhao Cai, Jiangshan Wang, Yawei Li, and Yan Lu. Efficient autoregressive video diffusion with dummy head. *arXiv preprint arXiv:2601.20499*, 2026a.

Yuwei Guo, Ceyuan Yang, Hao He, Yang Zhao, Meng Wei, Zhenheng Yang, Weilin Huang, and Dahua Lin. End-to-end training for autoregressive video diffusion via self-resampling. In *European Conference on Computer Vision*, pp. 324–344. Springer, 2026b.

Coleman Hooper, Sehoon Kim, Hiva Mohammadzadeh, Michael W Mahoney, Yakun S Shao, Kurt Keutzer, and Amir Gholami. Kvquant: Towards 10 million context length llm inference with kv cache quantization. *Advances in Neural Information Processing Systems*, 37:1270–1303, 2024.

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, et al. Vbench: Comprehensive benchmark suite for video generative models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818. IEEE, 2024.

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

Yicheng Ji, Zhizhou Zhong, Jun Zhang, Qin Yang, XiTai Jin, Ying Qin, Wenhan Luo, Shuiyang Mao, Wei Liu, and Huan Li. Forcing-kv: Hybrid kv cache compression for efficient autoregressive video diffusion models. *arXiv preprint arXiv:2605.09681*, 2026.

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.

Zirui Liu, Jiayi Yuan, Hongye Jin, Shaochen Zhong, Zhaozhuo Xu, Vladimir Braverman, Beidi Chen, and Xia Hu. Kivi: A tuning-free asymmetric 2bit quantization for kv cache. *arXiv preprint arXiv:2402.02750*, 2024.

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

Suraj Ranganath, Vaishak Menon, and Anish Patnaik. Kv cache quantization for self-forcing video generation: A 33-method empirical study. *arXiv preprint arXiv:2603.27469*, 2026.

Dvir Samuel, Issar Tzachor, Matan Levy, Michael Green, Gal Chechik, and Rami Ben-Ari. Fast autoregressive video diffusion and world models with temporal cache compression and sparse attention. *arXiv preprint arXiv:2602.01801*, 2026.

Jinliang Shen, Lianghao Su, Zheming Li, Kang He, ZiLiang Lai, Yanbing Jiang, and Chengru Song. Headcast: Casting attention heads for efficient autoregressive video generation. *arXiv preprint arXiv:2607.20125*, 2026.

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.

Meituan LongCat Team, Xunliang Cai, Qilong Huang, Zhuoliang Kang, Hongyu Li, Shijun Liang, Liya Ma, Siyu Ren, Xiaoming Wei, Rixu Xie, and Tong Zhang. Longcat-video technical report, 2025. URL `https://arxiv.org/abs/2510.22200`.

Robbyant Team, Zelin Gao, Qiuyu Wang, Yanhong Zeng, Jiapeng Zhu, Ka Leong Cheng, Yixuan Li, Hanlin Wang, Yinghao Xu, Shuailei Ma, et al. Advancing open-source world models. *arXiv preprint arXiv:2601.20540*, 2026.

Tuna Tuncer, Felix Becker, and Thomas Pfeil. Quantized keys steal attention: Bias correction for kv-cache compression in video diffusion. *arXiv preprint arXiv:2605.26266*, 2026.

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.

Haocheng Xi, Shuo Yang, Yilong Zhao, Muyang Li, Han Cai, Xingyang Li, Yujun Lin, Zhuoyang Zhang, Jintao Zhang, Xiuyu Li, et al. Quant videogen: Auto-regressive long video generation via 2-bit kv-cache quantization. *arXiv preprint arXiv:2602.02958*, 2026.

June Yong Yang, Byeongwook Kim, Jeongin Bae, Beomseok Kwon, Gunho Park, Eunho Yang, Se Jung Kwon, and Dongsoo Lee. No token left behind: Reliable kv cache compression via importance-aware mixed precision quantization. *arXiv preprint arXiv:2402.18096*, 2024.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Hidir Yesiltepe, Jiazhen Hu, Tuna Han Salih Meral, Adil Kaan Akan, Kaan Oktay, Hoda Eldardiry, and Pinar Yanardag. Videomla: Low-rank latent kv cache for minute-scale autoregressive video diffusion. *arXiv preprint arXiv:2605.30351*, 2026.

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 *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22963–22974. IEEE, 2025.

Kaining Ying, Hengrui Hu, Siyu Ren, Jiamu Li, Fengjiao Chen, Ziwen Wang, Xuezhi Cao, Xunliang Cai, and Henghui Ding. Wbench: A comprehensive multi-turn benchmark for interactive video world model evaluation. *arXiv preprint arXiv:2605.25874*, 2026.

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.

Fei Zuo, Zikang Zhou, Hao Cong, Xiaoyan Xi, and Ho Fai Leung. Ratequant: Optimal mixed-precision kv cache quantization via rate-distortion theory. *arXiv preprint arXiv:2605.06675*, 2026.