Q-ARVD: Quantizing Autoregressive Video Diffusion Models

Siao Tang[1] **Xinyin Ma**[1] **Gongfan Fang**[1] **Xingyi Yang**[2] **Xinchao Wang**[1*]

[1]National University of Singapore [2]The Hong Kong Polytechnic University

{siao, maxinyin, gongfan}@u.nus.edu xingyi.yang@polyu.edu.hk

xinchao@nus.edu.sg

Abstract

Autoregressive video diffusion models (ARVDs) have emerged as a promising architecture for streaming video generation, paving the way for real-time interactive video generation and world modeling. Despite their potential, the substantial inference cost of ARVDs remains a major obstacle to practical deployment, making model quantization a natural direction for improving efficiency. However, quantization for ARVDs remains largely unexplored. Our empirical analysis shows that directly applying existing quantization schemes developed for standard diffusion transformers to ARVDs leads to suboptimal performance, revealing quantization behaviors that differ from those observed in bidirectional diffusion models. In this paper, we identify two critical challenges in quantizing ARVDs: **(C1)** Highly unbalanced frame-wise quantization sensitivity. Error accumulation during autoregressive generation can induce severely skewed quantization sensitivity across frames, following an exponential-like decay pattern. **(C2)** Prominent and heterogeneous outlier patterns in weights. Weight distributions exhibit pronounced outlier channels, whose patterns vary substantially across layer types and block depths. To address these issues, we propose **Q-ARVD**, a novel framework for accurate ARVD quantization. (S1) To tackle the highly unbalanced frame-wise sensitivity, Q-ARVD incorporates a final-quality aware frame-weighting mechanism into the quantization objective. (S2) To prevent heterogeneous outliers from degrading performance, Q-ARVD introduces an outlier-aware adaptive dual-scale quantization, which automatically detects the presence and quantity of outlier channels for an arbitrary layer, and isolates them to protect normal channels. Extensive experiments on state-of-the-art open-source ARVDs (i.e., self-forcing and causal-forcing) demonstrate the superiority of Q-ARVD. Practical deployment of INT8 model shows 1.30x speedup and 1.97x model size reduction.

1 Introduction

Video diffusion models (Wan et al., 2025; HaCohen et al., 2024; Kong et al., 2024; Yang et al., 2025a; Wu et al., 2025; Xiao and Wang, 2026a,b) have demonstrated strong capabilities in high-fidelity and temporally coherent video content generation. While traditional bidirectional video diffusion models excel at offline generation, they fundamentally struggle with real-time interactive applications due to their full-sequence joint generation paradigm. Recently, Autoregressive Video Diffusion Models (ARVDs) (Huang et al., 2025a; Zhu et al., 2026; Yin et al., 2025; Teng et al., 2025; Jin et al., 2024; Chen et al., 2025; Deng et al., 2025; Ma et al., 2026) have emerged as an appropriate architecture for streaming video generation. By transforming video synthesis into a chunk-by-chunk or frame-by-frame causal generation process, ARVDs pave the way for applications such as real-time interactive video content generation (Shin et al., 2025; Ki et al., 2026; Feng et al., 2025a) and world modeling (Mao et al., 2025; Sun et al., 2025; Huang et al., 2025a; Zhang et al., 2026).

*Corresponding author

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Similar to other foundation models, enhancing the inference efficiency of ARVDs by model quantization (Nagel et al., 2021; Krishnamoorthi, 2018) is of great practical importance, particularly for real-time scenarios and deployment on resource-constrained devices. However, directly applying quantization to ARVDs remains a non-trivial endeavor due to the paradigm shift from bidirectional to autoregressive. Off-the-shelf quantization methods optimized for bidirectional diffusion transformers (Wu et al., 2024; Li et al., 2025a) or large language models (LLMs) (Xiao et al., 2023) often yield suboptimal performance. In this work, we bridge this gap by identifying and addressing two bottlenecks that uniquely characterize the quantization of ARVDs.

First, we observe a *highly unbalanced quantization sensitivity across frames* caused by error accumulation. In ARVDs, the generation of the current frame is conditioned on the past generated frames. Consequently, quantization errors introduced in early frames rapidly compound over the autoregressive rollout. This suggests that frame-wise quantization sensitivity is heavily skewed toward early frames. Our empirical study reveals that this sensitivity follows an exponential-like decay along the temporal axis, indicating that the quality of the generated video is disproportionately governed by the precision of the early frames. As a result, treating all frames equally during quantization calibration is sub-optimal. **Second**, we observe that weight distributions in ARVDs exhibit *prominent channel-wise outliers*. A small fraction of input channels (e.g., 2.1%) show substantially larger magnitudes than the majority, elevating the difficulty of quantization. Furthermore, these outlier patterns are highly heterogeneous, varying markedly across different layer types (e.g., self-attention, cross-attention, and FFN) and block depths. Some layers exhibit severe outliers, while certain layers are well-behaved, so a static solution for outliers is inherently not appropriate.

To tackle the two challenges, we propose **Q-ARVD**, a novel quantization framework specifically tailored for autoregressive video diffusion models. To cope with the first challenge, i.e., the unbalanced frame-wise sensitivity, Q-ARVD introduces a *final-quality guided frame-weighting* mechanism into the quantization objective. We directly quantify this sensitivity by evaluating how quantizing a certain frame affects the overall generated video, thereby modeling the actual effect of autoregressive error propagation. We then assign importance weights to different frames during quantization calibration, emphasizing precision preservation for critical early frames. To address the second challenge, i.e., the heterogeneous outlier patterns, Q-ARVD proposes an *outlier-aware adaptive dual-scale* quantization. This strategy automatically identifies the presence and optimal number of outlier channels for arbitrary layers. To prevent the identified outlier channels from interfering with normal channels, we employ separate quantizers for them, resulting in a lower scaling factor for normal channels and thereby reducing quantization errors. Our main contributions can be summarized as follows:

- We identify two critical challenges for quantizing autoregressive video diffusion models, i.e., unbalanced frame-wise quantization sensitivity, and prominent heterogeneous outlier patterns of model weights.
- To resolve the two challenges, we propose **Q-ARVD**, which features a final-quality guided frame-weighting mechanism to handle sensitivity discrepancy, and an adaptive dual-scale strategy to automatically detect and address outliers. To the best of our knowledge, Q-ARVD is the first quantization framework tailored for autoregressive video diffusion models.
- Extensive experiments demonstrate that Q-ARVD significantly outperforms existing diffusion quantization baselines, achieving near-lossless visual quality. In practical deployment, the INT8 model delivers a $1.97\times$ reduction in model size and a $1.30\times$ latency speedup.

2 Related Works

2.1 Autoregressive Video Diffusion Models

Recent video generation models are shifting from full-sequence bidirectional generation (Wan et al., 2025; Kong et al., 2024; Yang et al., 2025a) to autoregressive generation (Teng et al., 2025; Huang et al., 2025a; Zhu et al., 2026). Similar to causal decoding in large language models, autoregressive video diffusion models generate frames or chunks sequentially, conditioning each new frame on previously generated ones, formulated as $P(\boldsymbol{x}_0^{1:N}) = \prod_{i=1}^{N} p_\theta(\boldsymbol{x}_0^i \mid \boldsymbol{x}_0^{<i})$, where $p_\theta(\boldsymbol{x}_0^i \mid \boldsymbol{x}_0^{<i})$ is modeled by diffusion denoising conditioned on past clean frames. Early ARVDs rely on multi-step diffusion denoising and thus suffer from high inference latency. Recent methods improve efficiency and quality through few-step distillation, exposure-bias mitigation, and teacher-student architecture

alignment (Yin et al., 2025; Huang et al., 2025a; Zhu et al., 2026), while another line of work extends fixed-length autoregressive models to long-horizon generation (Yang et al., 2025b; Yesiltepe et al., 2025; Liu et al., 2025a; Yi et al., 2025). These advances make ARVDs well-suited for streaming video generation, enabling real-time interactive generation (Shin et al., 2025) and world modeling (Sun et al., 2025). Our work further improves their inference efficiency, particularly for deployment on resource-constrained devices.

2.2 Model Quantization Preliminaries

Model quantization (Nagel et al., 2021; Krishnamoorthi, 2018) is one of the most significant techniques of efficient model inference Lu et al. (2026); Chen et al. (2026a); Wang et al. (2026); Li and Wang (2026); Chen et al. (2026b); Li and Wang (2026). Quantization methods compress neural networks by representing model weights and input activations using low-precision formats, e.g., INT4 or INT8. The quantization process can be formulated as:

$$x_q = \text{clip}\left(\text{round}\left(\frac{x}{s}\right) + z, q_{\min}, q_{\max}\right), \tag{1}$$

where x_q is the low-precision representation, s is the scaling factor, and z is the zero-point. For symmetric quantization, $s = \frac{\max(|x|)}{2^{b-1}-1}$. $q_{\min}$ and $q_{\max}$ denote the lower and upper bounds of the low-precision format. To better maintain the model performance, a common practice (Nagel et al., 2020; Li et al., 2021) is to optimize quantization parameters through reconstruction on a calibration dataset $\mathcal{D}_{cal}$:

$$\mathcal{L} = \mathbb{E}_{\mathbf{X}\sim\mathcal{D}_{cal}} \left\|\mathbf{X}\mathbf{W} - Q(\mathbf{X})Q(\mathbf{W})\right\|_{\text{F}}^2, \tag{2}$$

where $\mathbf{X}$ and $\mathbf{W}$ are activations and weights. $Q(.)$ denotes the quantize-then-dequantize operation. Learnable parameters include scaling factors, rounding schemes, etc.

2.3 Model Quantization for Diffusion Models

Quantization methods have been widely applied to improve the inference efficiency of diffusion models. Early works (Shang et al., 2023; Li et al., 2023; He et al., 2023; So et al., 2023; Huang et al., 2024a; Tang et al., 2024) focus on quantizing the UNet backbone in diffusion models, and incorporate specific designs to accommodate the temporal denoising characteristics. With the architectural shift toward diffusion transformers (DiTs) (Peebles and Xie, 2023), subsequent works (Wu et al., 2024; Li et al., 2025a; Zhao et al., 2025; Li et al., 2025b; Feng et al., 2025b; Huang et al., 2025b) propose dedicated quantization schemes designed for DiT-based diffusion models. Likewise, the recent paradigm shift from bidirectional to autoregressive video diffusion introduces new challenges for quantization, as mentioned before. Motivated by this, we develop a quantization framework tailored for autoregressive video diffusion models.

3 Method: Q-ARVD

In this section, we elaborate on the proposed Q-ARVD framework. There are two key innovations of our framework. First, to address the issue of unbalanced frame-wise sensitivity, we propose the *final-quality guided frame-weighting* mechanism (§3.1). Second, to deal with the heterogeneous outlier patterns in model weights, we introduce an *outlier-aware adaptive dual-scale* quantization strategy (§3.2). The overall framework is illustrated in Figure 1.

3.1 Final-quality Guided Frame-weighting

Autoregressive video diffusion models combine high-quality diffusion sampling with the LLM-like autoregressive decoding paradigm. The generation of a new frame $\boldsymbol{x}_0^i$ is conditioned on previous clean frames $\boldsymbol{x}_0^{<i}$. However, unlike discrete tokens in LLMs, these frames are continuous and high in information density, making the errors in previous frames significantly undermine subsequent frames. Intuitively, earlier frames exert a greater impact on the overall video quality. In other words, the quality of generated videos is more sensitive to the quantization errors in earlier frames.

We formally denote the *frame-wise sensitivity* of the i-th frame as α_i, where a larger α_i indicates higher sensitivity. To accurately quantify α_i, we employ the final video quality degradation as a

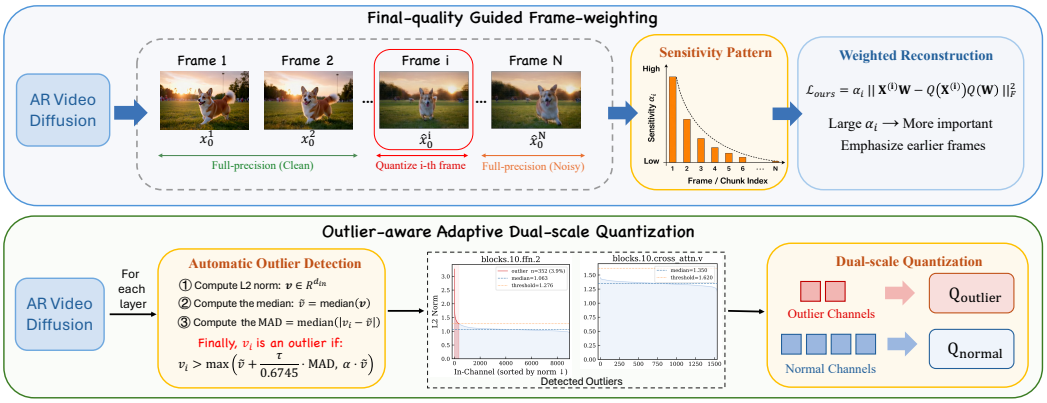

Figure 1: The illustration of our Q-ARVD framework.

direct indicator, which we find is simple but effective. Specifically, let $V = \{\boldsymbol{x}_0^i\}_{i=1}^N$ denote a video with N frames, where $\boldsymbol{x}_0^i$ is the i-th clean frame. The original autoregressive generation is $P(\boldsymbol{x}_0^{1:N}) = \prod_{i=1}^N p_\theta(\boldsymbol{x}_0^i \mid \boldsymbol{x}_0^{<i})$. To calculate α_i, we only enable quantization for the generation of the i-th frame. Then, the modified autoregressive process is:

$$\hat{P}_i(\hat{\boldsymbol{x}}_0^{1:N}) = \underbrace{\prod_{k=1}^{i-1} p_\theta(\boldsymbol{x}_0^k \mid \boldsymbol{x}_0^{<k})}_{\text{Full-precision (Clean)}} \cdot \underbrace{\hat{p}_\theta(\hat{\boldsymbol{x}}_0^i \mid \boldsymbol{x}_0^{<i})}_{\text{Quantize i-th frame}} \cdot \underbrace{\prod_{k=i+1}^{N} p_\theta(\hat{\boldsymbol{x}}_0^k \mid \boldsymbol{x}_0^{1:i-1}, \hat{\boldsymbol{x}}_0^{i:k-1})}_{\text{Full-precision (Noisy)}}, \tag{3}$$

where $\hat{p}_\theta$ represents the model in the quantized state, and $\hat{x}$ means this frame is influenced by quantization errors. Here, the quantized model is implemented without reconstruction in Equation (2). Note that for $k > i$, the model reverts to full-precision, but the generated frames are still impacted since they are conditioned on the quantized i-th and subsequent frames. Finally, the i-th frame sensitivity α_i is calculated as the quality degradation caused by quantization, i.e., the distance between the original video $P(\boldsymbol{x}_0^{1:N})$ and the quantized one $\hat{P}_i(\hat{\boldsymbol{x}}_0^{1:N})$, which can be formulated as:

$$\alpha_i = d(P(\boldsymbol{x}_0^{1:N}), \hat{P}_i(\hat{\boldsymbol{x}}_0^{1:N})). \tag{4}$$

In practice, we compute the distance using the mean-squared error (MSE) in the latent space. We use chunk-wise model following self-forcing (Huang et al., 2025a), where each chunk contains several frames. Through experiments on 100 videos with different prompts, we obtain the sensitivity pattern shown in Figure 2. The sensitivity varies significantly across chunks, exhibiting an exponential-like decay. For example, the sensitivity score of chunk 1 of self-forcing (W8A8) is 0.70, while the last chunk is less than 0.01. The finding indicates that treating all frames equally for quantization calibration is not optimal. Therefore, we use the sensitivity as the loss-weighting coefficients for the quantization reconstruction process. **The new reconstruction objective is:**

$$\mathcal{L}_{ours} = \mathbb{E}_{\mathbf{X}\sim\mathcal{D}_{cal},\, i\sim\mathcal{U}(1,N)} \left[\alpha_i \left\| \mathbf{X}^{(i)}\mathbf{W} - Q(\mathbf{X}^{(i)})Q(\mathbf{W}) \right\|_{\mathrm{F}}^2 \right], \tag{5}$$

where $\mathbf{X}^{(i)}$ means that the activation is obtained from the generation process of the i-th frame.

3.2 Outlier-aware Adaptive Dual-scale Quantization

We delve into the weight distributions of autoregressive video diffusion models. Concretely, we collect the statistics of input-channel-wise magnitudes for every layer, as demonstrated in Figure 3. We compute the per-channel L2 norms and sort them in descending order, from which we can draw the following observations.

(i) *There exist outlier channels which only account for a small fraction but possess obviously larger magnitudes than normal channels.*

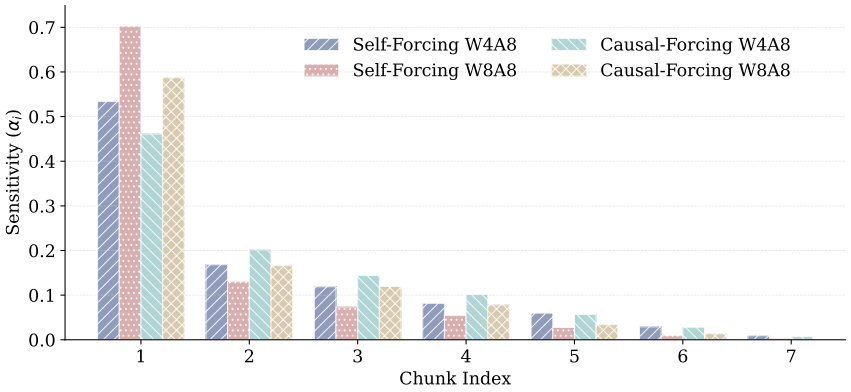

Figure 2: Quantization sensitivity patterns in autoregressive video diffusion models, with scores normalized to sum to 1.

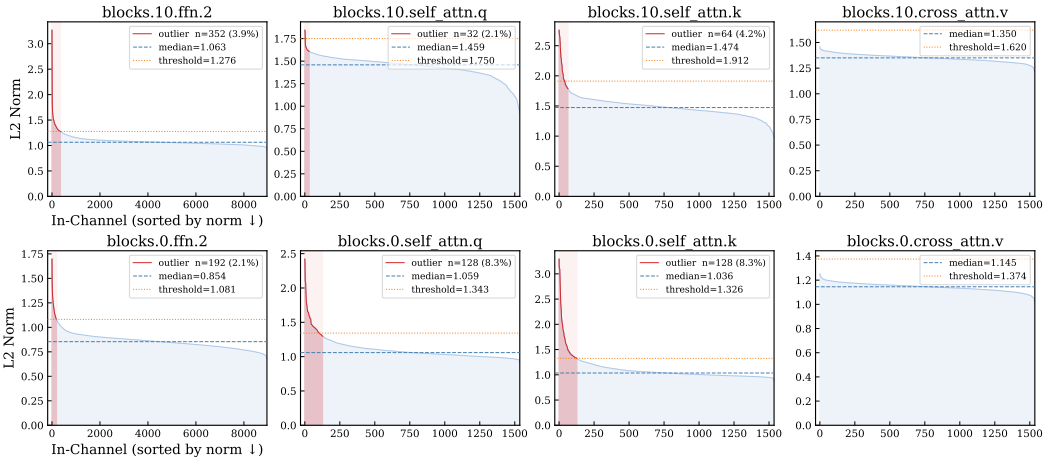

Figure 3: The outlier patterns in autoregressive video diffusion models. The x-axis denotes input channel index sorted in descending L2 norm, and the y-axis denotes the corresponding L2 norm values. Outlier channels identified by our method (§3.2, Equation (12)) are highlighted in red. In practice, we further align the number of detected outlier channels to a multiple of 32 for hardware-friendly deployment. Additional samples are provided in the Appendix §A.

(ii) *The outlier patterns are highly **heterogeneous**, varying significantly across different layer types and block depths. For example, the second FFN layers (ffn.2) are prominent, while the cross-attention value projections (cross_attn.v) are smooth.*

Addressing outliers with dual-scale quantization. Observation (i) reveals that there is ample room to improve quantization quality by addressing these outlier channels. Let us start by revisiting why the outliers are not welcome in quantization. The total quantization error ϵ consists of two components, i.e., the clipping error and the rounding error. The outliers mainly undermine quantization through increasing the rounding error. For example, in symmetric quantization, we have the scaling factor $s = \frac{\max(|x|)}{2^{b-1}-1}$. Let $\hat{x}$ denote the de-quantized value of x. From Equation (1), we can derive:

$$\hat{x} = (x_q - z) \cdot s = [(\text{round}\left(\frac{x}{s}\right) + z) - z] \cdot s = \text{round}\left(\frac{x}{s}\right) \cdot s \,, \tag{6}$$

$$\mathbb{E}[\epsilon] = \mathbb{E}[|\hat{x} - x|] = \mathbb{E}[|\,\text{round}\left(\frac{x}{s}\right) - \frac{x}{s}| \cdot s] = \frac{1}{4}s \,. \tag{7}$$

Here, we assume the rounding error follows the uniform distribution. Outliers inflate $\max|x|$ and consequently lead to a larger scaling factor s. As shown in Equation (7), this will lead to a higher quantization error. To address this problem, we propose a dual-scale quantization strategy to isolate

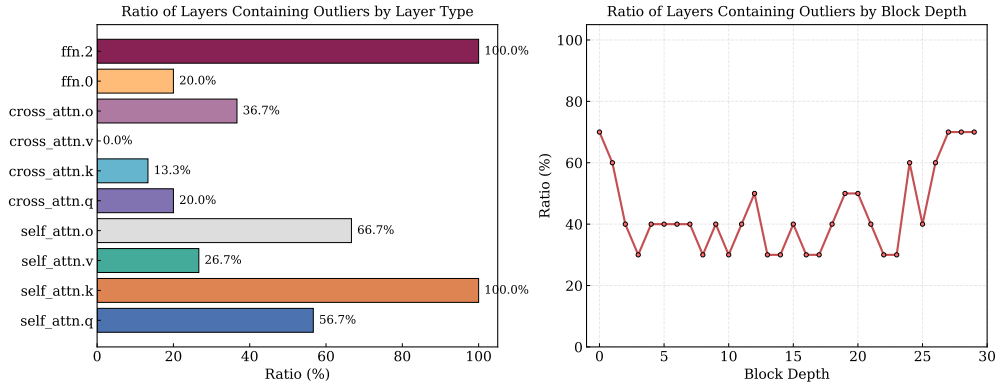

Figure 4: The ratio of layers containing outliers in terms of layer type and block depth. A layer is identified as outlier-containing upon detection of at least one outlier channel.

outlier channels from normal channels, thereby preventing them from inflating the quantization errors, which can be formulated as:

$$Q_{\text{dual}}(\mathbf{W}) = \left[Q_{\text{outlier}}(\mathbf{W}_{\text{outliers}}) \mid Q_{\text{normal}}(\mathbf{W}_{\text{normal}}) \right], \tag{8}$$

where $[\cdot|\cdot]$ denotes concatenation along the input-channel dimension, and Q_{outlier} and Q_{normal} are two independent quantizers, for outlier and normal channels respectively. The separate quantizer results in a lower scaling factor for normal channels, and theoretically reduce quantization errors according to Equation (7). We also discuss and compare related outlier-handling approaches in Appendix §D.

Adaptively detecting heterogeneous outlier patterns. However, Observation (ii) indicates that the outlier patterns are heterogeneous across layers. Some layers (e.g., ffn.2) manifest significant outliers, while certain layers (e.g., cross_attn.v) exhibit smooth distributions. This disparity raises a critical question: How to determine whether there exists an outlier pattern and how many top channels should be regarded as outliers? Manually tuning is labor-intensive and lacks generalizability. To achieve automatic and adaptive outlier detection, we employ the Modified Z-score (Iglewicz and Hoaglin, 1993). Given the L2 norm vector $\boldsymbol{v} \in \mathbb{R}^{d_{\text{in}}}$, we first compute the Median Absolute Deviation (MAD):

$$\text{MAD} = \text{median}\left(|v_i - \tilde{v}|\right), \text{where } \tilde{v} = \text{median}(\boldsymbol{v}). \tag{9}$$

The Modified Z-score for each channel is formulated as:

$$\text{M}_i = 0.6745 \cdot \frac{v_i - \tilde{v}}{\text{MAD}}, \tag{10}$$

The Modified Z-score measures how far a channel deviates from the median in a normalized manner. Following the standard Modified Z-score criterion, a channel is marked as an outlier when M_i exceeds a threshold τ, i.e., $0.6745 \cdot \frac{v_i - \tilde{v}}{\text{MAD}} > \tau$, which can be rewritten as:

$$v_i > \tilde{v} + \frac{\tau}{0.6745} \cdot \text{MAD}\,. \tag{11}$$

However, we observe that for certain smooth layers, the MAD can be extremely small, resulting in a low right-hand side of Equation (11). This will mark a lot of normal values as outliers, which we refer to as "false outliers". To avoid this issue, we introduce a minimum magnitude constraint. **Finally, a channel is classified as outlier when it satisfies both the Modified Z-score and the minimum magnitude conditions:**

$$v_i > \max(\tilde{v} + \frac{\tau}{0.6745} \cdot \text{MAD},\ \ \alpha \cdot \tilde{v})\,, \tag{12}$$

where $\tau = 3.5$ is the standard Modified Z-score threshold, and $\alpha = 1.2$ (default) denotes the minimum ratio relative to the median norm. A layer is considered to contain outlier channels if at least one outlier channel is detected, in which case dual-scale quantization will be applied. Figure 4 shows the proportion of layers containing outliers in terms of layer type and block depth.

4 Experiments

4.1 Experimental Setups

Models and Baselines. We use two state-of-the-art autoregressive video diffusion models, i.e., self-forcing (Huang et al., 2025a) and causal-forcing (Zhu et al., 2026), and follow their official configurations. Our baselines include five representative quantization paradigms. Specifically, MinMax (Nagel et al., 2021) serves as a vanilla quantization approach, while AdaRound (Nagel et al., 2020) represents a classical reconstruction-based method. SmoothQuant (Xiao et al., 2023) is a widely adopted method for handling activation outliers in transformers via channel-wise scaling. PTQ4DiT (Wu et al., 2024) is a framework tailored for diffusion transformers. SVDQuant (Li et al., 2025a) pioneers in mitigating weight outliers by introducing a low-rank full-precision branch.

Quantization Implementation. In all baselines and our method, we use per-channel quantization for weights and per-tensor static quantization for activations. In the initialization of scaling factors, we search for the optimal percentile of clipping from [0.999, 0.9999, 0.99999]. We choose the extended MovieGenVideoBench prompts (Polyak et al., 2024; Huang et al., 2025a) as calibration data. Following previous works (Wu et al., 2024; Li et al., 2023), we train adaptive rounding and scaling factors by reconstruction.

Benchmark and Metrics. Following common practice in evaluating video generative models, we evaluate quantized models on the VBench benchmark (Huang et al., 2024b). We adopt two types of metrics, i.e., reference-based metrics and reference-free metrics. **Reference-based** metrics measure the distance between videos generated by quantized models and those from the full-precision (FP) model (Zhao et al., 2025; Tang et al., 2024). Specifically, we adopt two popular distance metrics, i.e., FVD (Unterthiner et al., 2018) and LPIPS (Zhang et al., 2018), denoted as FVD-FP (Zhao et al., 2025) and LPIPS-FP in our quantization task, respectively. For reference-free evaluation, we report five VBench quality scores. To ensure reliable evaluation, especially for FVD-FP, we generate videos using all 946 extended VBench prompts. Empirically, we observe that VBench scores exhibit limited discriminative power for evaluating quantization performance, whereas reference-based metrics are far more sensitive and better aligned with actual quality. Therefore, we primarily rely on **FVD-FP and LPIPS-FP** for quantitative comparison, while using VBench scores as auxiliary evidence.

4.2 Main Results

Main comparison and visual results. Table 1 and Table 2 show the quantitative results on causal-forcing and self-forcing, respectively. We test three different bitwidths, i.e., W8A8, W4A8, and W4A6, with increasing quantization difficulty. The results show that Q-ARVD consistently achieves the best FVD-FP and LPIPS-FP scores, outperforming all baselines by a clear margin. The improvement is more pronounced under low-bit settings (i.e., W4A8 and W4A6), where the outlier issue becomes more severe. Moreover, Figure 5 shows the visual results. MinMax suffers from significant accumulated errors, leading to severe degradation in frame quality over time. SVDQuant introduces noticeable semantic changes compared to the original Bfloat16 video, such as the style and viewpoint of the beach, and the appearance and posture of the dog. In contrast, our method maintains high video quality consistently across the full temporal span. More visual examples can be found in the Appendix (§B Figure 11- Figure 16). We also validate the practical deployment of the W8A8 model using Triton (Tillet and Cox, 2019), and observe a 1.30x latency speedup on the NVIDIA A6000 GPU and a 1.97x model size reduction, with a batch size of 2 and default configurations.

Discussion of VBench scores in quantization. Our experiments reveal that standard VBench metrics have limited discriminability and can even exhibit counterintuitive behavior when evaluating quantized video models. For example, in Table 2, the motion smoothness score varies only marginally (98.70–99.06), and the average VBench scores of all W8A8 models fall within a narrow range of 85.28–85.33, while FVD-FP and LPIPS-LP changes substantially across settings. More surprisingly, our lower-precision W4A6 Q-ARVD outperforms W8A8, W4A8, and even BF16 in metrics such as subject consistency and motion smoothness. A similar phenomenon is observed in Table 1, for instance, the average score of W4A6 Q-ARVD surpasses W4A8. To systematically assess metric reliability, we introduce a unified *Discriminability Score* (DS), which measures both *sensitivity* and *faithfulness*. Sensitivity is quantified by the coefficient of variation (CV) (Abdi, 2010), $\mathrm{CV}_m = \sigma_m/\mu_m$, where higher values indicate stronger responsiveness to quality differences. Faithfulness is measured by the proposed bitwidth-order agreement (BOA), which evaluates whether a metric

Table 1: Quantitative results of causal-forcing (Zhu et al., 2026). We use the reference-based metrics, i.e., FVD-FP and LPIPS-FP, as the primary criteria for quantitative comparison, while treating VBench scores as complementary evidence.

Method	Bitwidth	Reference-free Metrics						**Reference-based Metrics**	
		Subj. Cons.	Back. Cons.	Motion Smooth.	Aesth. Qual.	Imag. Qual.	Avg.	**FVD-FP ↓**	**LPIPS-FP ↓**
Bfloat16	16/16	96.91	96.59	98.47	62.88	71.80	85.33	0.00	0.00
MinMax	W4A8	95.36	95.37	98.82	59.08	68.41	83.41	279.61	0.505
Adaround		96.38	96.02	98.56	61.71	71.40	84.81	143.49	0.463
SmoothQuant		96.56	95.84	98.71	60.77	71.25	84.63	220.94	0.507
PTQ4DiT		96.71	96.07	98.70	61.65	71.69	84.96	141.03	0.470
SVDQuant		96.24	95.94	98.81	59.98	70.35	84.26	135.62	0.492
Q-ARVD (Ours)		96.74	96.33	98.56	61.92	71.23	84.96	**106.04**	**0.452**
MinMax	W8A8	96.71	96.45	98.46	62.33	72.16	85.22	67.65	0.358
Adaround		96.95	96.60	98.52	62.68	71.93	85.34	62.57	0.341
SmoothQuant		96.81	96.50	98.64	62.40	72.10	85.29	69.67	0.360
PTQ4DiT		96.97	96.66	98.52	62.60	72.08	85.37	63.21	0.341
SVDQuant		96.80	96.55	98.43	62.66	72.07	85.30	63.52	0.361
Q-ARVD (Ours)		96.98	96.59	98.52	62.66	72.04	85.36	**61.67**	**0.335**
MinMax	W4A6	94.82	94.95	98.73	58.27	65.77	82.51	375.82	0.544
Adaround		96.40	95.96	98.58	61.62	69.96	84.50	233.43	0.507
SmoothQuant		96.57	95.61	98.75	59.22	69.72	83.97	326.97	0.539
PTQ4DiT		96.21	95.28	98.55	60.53	70.75	84.26	244.95	0.527
SVDQuant		96.13	95.76	98.65	59.20	68.76	83.70	210.28	0.532
Q-ARVD (Ours)		97.32	96.66	98.82	62.55	70.86	85.24	**140.38**	**0.486**

Table 2: Quantitative results of self-forcing (Huang et al., 2025a).

Method	Bitwidth	Subj. Cons.	Back. Cons.	Motion Smooth.	Aesth. Qual.	Imag. Qual.	Avg.	**FVD-FP ↓**	**LPIPS-FP ↓**
Bfloat16	16/16	97.25	96.68	98.99	62.49	71.35	85.35	0.00	0.00
MinMax	W4A8	95.82	95.00	98.92	58.39	68.53	83.33	260.14	0.514
Adaround		96.89	96.05	98.96	60.79	70.12	84.56	156.70	0.474
SmoothQuant		96.72	95.82	98.89	59.24	71.05	84.34	220.30	0.513
PTQ4DiT		97.17	96.20	98.96	61.32	71.11	84.95	124.20	0.477
SVDQuant		96.41	95.56	98.78	58.62	70.65	84.00	150.03	0.502
Q-ARVD (Ours)		97.05	96.27	98.95	61.00	70.71	84.80	**116.26**	**0.466**
MinMax	W8A8	97.31	96.64	98.99	61.83	71.80	85.31	77.65	0.351
Adaround		97.29	96.62	99.00	62.03	71.44	85.28	68.24	0.334
SmoothQuant		97.31	96.65	99.00	61.96	71.75	85.33	81.08	0.354
PTQ4DiT		97.33	96.69	99.01	62.07	71.43	85.31	68.47	0.333
SVDQuant		97.29	96.67	99.01	61.97	71.72	85.33	74.87	0.349
Q-ARVD (Ours)		97.26	96.62	98.99	62.13	71.49	85.30	**64.51**	**0.327**
MinMax	W4A6	96.11	94.95	99.04	57.84	67.30	83.05	321.32	0.534
Adaround		96.68	95.59	99.06	59.79	67.85	83.79	224.95	0.515
SmoothQuant		96.96	95.77	98.97	58.61	70.06	84.07	284.80	0.535
PTQ4DiT		97.16	95.95	98.98	60.66	70.17	84.58	174.17	0.515
SVDQuant		96.67	95.71	98.70	58.29	70.09	83.89	215.32	0.542
Q-ARVD (Ours)		97.39	96.30	99.03	60.79	70.33	84.77	**146.01**	**0.498**

follows the expected quantization ordering, i.e., BF16 $\succ$ W8A8 $\succ$ W4A8 $\succ$ W4A6. Formally, $\mathrm{BOA}_m = \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}\big(m^{\mathrm{BF16}} \succ m_i^{\mathrm{W8A8}} \succ m_i^{\mathrm{W4A8}} \succ m_i^{\mathrm{W4A6}}\big)$, where N is the number of quantization methods ($N = 6$ in our setting). A higher BOA indicates better consistency with the bitwidth ordering. We define the final score as $\mathrm{DS}_m = \mathrm{CV}_m \cdot \mathrm{BOA}_m$. As shown in Figure 7, FVD-FP and LPIPS-FP achieve substantially higher CV, BOA, and DS scores, indicating better sensitivity and consistency with true model quality. In contrast, VBench metrics show low dispersion and poor alignment

Table 3: Ablation study of frame weighting and dual scale quantization using self-forcing.

Dual Scale	Frame Weighting	W4A8		W8A8	
		FVD↓	LPIPS↓	FVD↓	LPIPS↓
✗	✗	156.70	0.474	68.24	0.334
✗	✓	147.16	**0.465**	65.39	**0.325**
✓	✗	121.83	0.469	67.48	0.332
✓	✓	**116.26**	0.466	**64.51**	0.327

Table 4: Comparison of different frame-weighting strategies (self-forcing).

Strategy	FVD ↓	LPIPS ↓
Uniform (no weighting)	121.83	0.469
Heuristic Exp 2^{-i}	119.61	0.471
Reverse	123.72	0.476
Ours	**116.26**	**0.466**



Figure 5: The visual comparison of the self-forcing model with W4A8. Additional samples are presented in the Appendix (§B Figure 11- Figure 16).

with the expected bitwidth ordering, suggesting that commonly used metrics may be unreliable for quantization evaluation without careful validation.

4.3 Ablation Study

Effectiveness of frame weighting and dual scale quantization. We validate the effectiveness of the final-quality guided frame-weighting (Frame Weighting) and outlier-aware adaptive dual-scale quantization (Dual Scale), as presented in Table 3. Both frame weighting and dual-scale quantization individually bring performance gains. Specifically, frame weighting delivers more pronounced improvements under the high-precision W8A8 setting. In contrast, dual-scale quantization exhibits superior efficacy in the low bit-width W4A8 setting. This is because low-precision weight representation is more vulnerable to performance degradation caused by outliers, making the use of outlier-aware dual-scale quantization particularly critical. Finally, the joint integration of the two modules achieves the best overall performance.

The sensitivity to threshold τ of Modified Z-score. We further conduct sensitivity experiments to validate the robustness of the outlier-aware adaptive dual-scale module, focusing on the threshold τ of the Modified Z-score. As illustrated in Figure 6, we vary τ from 2.5 to 4.5. The results show the FVD-FP score fluctuates slightly within a narrow range of 114.39 to 117.41, and the LPIPS-FP score remains stable between 0.460 and 0.470 across all threshold settings. The slight performance variations demonstrate that our dual-scale module is robust to the selection of τ.

Compared to heuristic frame-weighting strategies. We compare our final-quality guided weighting with a heuristic exponential decay of 2^{-i} and the reversed version of our weighting. As shown in Table 4, the heuristic decay is inferior to our final-quality guided strategy, and the reversed version even underperforms the uniform baseline, as it incorrectly emphasizes later frames.

5 Conclusion

We propose Q-ARVD, the first quantization framework tailored for autoregressive video diffusion models. Q-ARVD introduces a final-quality guided frame-weighting mechanism to handle the unbalanced frame-wise quantization sensitivity, and an outlier-aware adaptive dual-scale strategy to address the heterogeneous outlier patterns. Extensive quantitative and qualitative experiments show the superiority and rationality of our designs. We hope our method can shed light on frame-wise sensitivity-aware quantization and a new path for detecting and handling outliers.

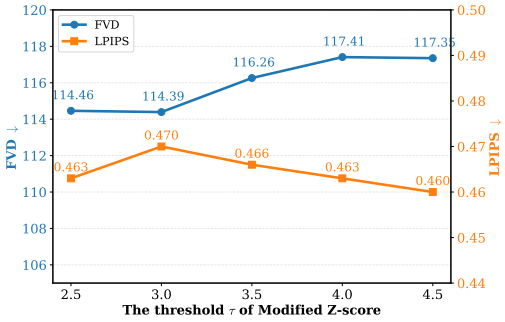

Figure 6: The sensitivity to threshold τ of Modified Z-score using self-forcing.

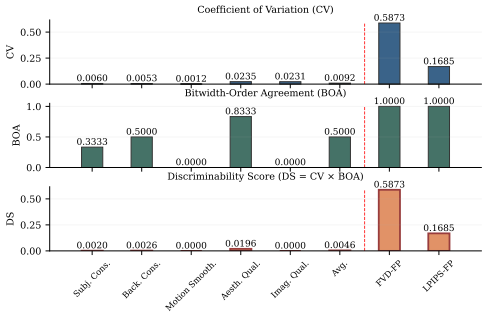

Figure 7: The coefficient of variation, bitwidth-order agreement and discriminability score of each metric. Use data of Table 1.

Acknowledgement

This project is supported by the Ministry of Education, Singapore, under the Academic Research Fund Tier 1 (FY2026, WBS: A-8004345-00-00), the National Research Foundation, Singapore, and Cyber Security Agency of Singapore under its National Cybersecurity R&D Programme and CyberSG R&D Cyber Research Programme Office (Award: CRPO-GC1-NTU-002).

References

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.

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

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.

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 *The Thirteenth International Conference on Learning Representations*, 2025a.

Bing Wu, Chang Zou, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Jack Peng, Jianbing Wu, Jiangfeng Xiong, Jie Jiang, et al. Hunyuanvideo 1.5 technical report. *arXiv preprint arXiv:2511.18870*, 2025.

Zeyu Xiao and Xinchao Wang. Event-guided online video super-resolution. *IEEE Transactions on Image Processing*, 2026a.

Zeyu Xiao and Xinchao Wang. Exploiting Blurry Representations for Event-guided Video Super-Resolution. In *AAAI Conference on Artificial Intelligence*, 2026b.

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

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.

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*, pages 22963–22974, 2025.

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.

Yang Jin, Zhicheng Sun, Ningyuan Li, Kun Xu, Hao Jiang, Nan Zhuang, Quzhe Huang, Yang Song, Yadong Mu, and Zhouchen Lin. Pyramidal flow matching for efficient video generative modeling. *arXiv preprint arXiv:2410.05954*, 2024.

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

Haoge Deng, Ting Pan, Haiwen Diao, Zhengxiong Luo, Yufeng Cui, Huchuan Lu, Shiguang Shan, Yonggang Qi, and Xinlong Wang. Autoregressive video generation without vector quantization. In *The Thirteenth International Conference on Learning Representations*, 2025.

Xinyin Ma, Julius Berner, Chao Liu, Arash Vahdat, Weili Nie, and Xinchao Wang. Flex-forcing: Towards a unified autoregressive and bidirectional video diffusion model. *International Conference on Machine Learning*, 2026.

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

Taekyung Ki, Sangwon Jang, Jaehyeong Jo, Jaehong Yoon, and Sung Ju Hwang. Avatar forcing: Real-time interactive head avatar generation for natural conversation. *arXiv preprint arXiv:2601.00664*, 2026.

Tianrui Feng, Zhi Li, Shuo Yang, Haocheng Xi, Muyang Li, Xiuyu Li, Lvmin Zhang, Keting Yang, Kelly Peng, Song Han, et al. Streamdiffusionv2: A streaming system for dynamic and interactive video generation. *arXiv preprint arXiv:2511.07399*, 2025a.

Xiaofeng Mao, Zhen Li, Chuanhao Li, Xiaojie Xu, Kaining Ying, Tong He, Jiangmiao Pang, Yu Qiao, and Kaipeng Zhang. Yume-1.5: A text-controlled interactive world generation model. *arXiv preprint arXiv:2512.22096*, 2025.

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.

Shihua Zhang, Qiuhong Shen, Shizun Wang, Tianbo Pan, and Xinchao Wang. Make geometry matter for spatial reasoning. In *European Conference on Computer Vision*, 2026.

Markus Nagel, Marios Fournarakis, Rana Ali Amjad, Yelysei Bondarenko, Mart Van Baalen, and Tijmen Blankevoort. A white paper on neural network quantization. *arXiv preprint arXiv:2106.08295*, 2021.

Raghuraman Krishnamoorthi. Quantizing deep convolutional networks for efficient inference: A whitepaper. *arXiv preprint arXiv:1806.08342*, 2018.

Junyi Wu, Haoxuan Wang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Ptq4dit: Post-training quantization for diffusion transformers. *Advances in neural information processing systems*, 37: 62732–62755, 2024.

Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. Svdquant: Absorbing outliers by low-rank component for 4-bit diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025a.

Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International conference on machine learning*, pages 38087–38099. PMLR, 2023.

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

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

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

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

Haiquan Lu, Gongfan Fang, Xinyin Ma, and Xinchao Wang. Vgedit: Unlocking video generation priors for reasoning-informed image editing. *European Conference on Computer Vision*, 2026.

Zigeng Chen, Gongfan Fang, Xinyin Ma, Ruonan Yu, and Xinchao Wang. DMax: Aggressive Parallel Decoding for dLLMs. In *Neural Information Processing Systems*, 2026a.

Zeqing Wang, Bowen Zheng, Xingyi Yang, Zhenxiong Tan, Yuecong Xu, and Xinchao Wang. Minute-long videos with dual parallelisms. *AAAI Conference on Artificial Intelligence*, 2026.

Qi Li and Xinchao Wang. Sponge tool attack: Stealthy denial-of-efficiency against tool-augmented agentic reasoning. In *International Conference on Machine Learning*, 2026.

Zigeng Chen, Gongfan Fang, Xinyin Ma, Ruonan Yu, and Xinchao Wang. dParallel: Learnable Parallel Decoding for dLLMs. In *International Conference on Learning Representations*, 2026b.

Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International conference on machine learning*, pages 7197–7206. PMLR, 2020.

Yuhang Li, Ruihao Gong, Xu Tan, Yang Yang, Peng Hu, Qi Zhang, Fengwei Yu, Wei Wang, and Shi Gu. Brecq: Pushing the limit of post-training quantization by block reconstruction. In *International Conference on Learning Representations*, 2021.

Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1972–1981, 2023.

Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-diffusion: Quantizing diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17535–17545, 2023.

Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Ptqd: Accurate post-training quantization for diffusion models. *arXiv preprint arXiv:2305.10657*, 2023.

Junhyuk So, Jungwon Lee, Daehyun Ahn, Hyungjun Kim, and Eunhyeok Park. Temporal dynamic quantization for diffusion models. *arXiv preprint arXiv:2306.02316*, 2023.

Yushi Huang, Ruihao Gong, Jing Liu, Tianlong Chen, and Xianglong Liu. Tfmq-dm: Temporal feature maintenance quantization for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7362–7371, 2024a.

Siao Tang, Xin Wang, Hong Chen, Chaoyu Guan, Zewen Wu, Yansong Tang, and Wenwu Zhu. Post-training quantization with progressive calibration and activation relaxing for text-to-image diffusion models. In *European Conference on Computer Vision*, pages 404–420. Springer, 2024.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023.

Tianchen Zhao, Tongcheng Fang, Haofeng Huang, Rui Wan, Widyadewi Soedarmadji, Enshu Liu, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, et al. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation. In *The Thirteenth International Conference on Learning Representations*, 2025.

Zhiteng Li, Hanxuan Li, Junyi Wu, Kai Liu, Haotong Qin, Linghe Kong, Guihai Chen, Yulun Zhang, and Xiaokang Yang. Dvd-quant: Data-free video diffusion transformers quantization. *arXiv preprint arXiv:2505.18663*, 2025b.

Weilun Feng, Chuanguang Yang, Haotong Qin, Xiangqi Li, Yu Wang, Zhulin An, Libo Huang, Boyu Diao, Zixiang Zhao, Yongjun Xu, et al. Q-vdit: Towards accurate quantization and distillation of video-generation diffusion transformers. *arXiv preprint arXiv:2505.22167*, 2025b.

Yushi Huang, Ruihao Gong, Jing Liu, Yifu Ding, Chengtao Lv, Haotong Qin, and Jun Zhang. Qvgen: Pushing the limit of quantized video generative models. *arXiv preprint arXiv:2505.11497*, 2025b.

Boris Iglewicz and David C Hoaglin. *Volume 16: how to detect and handle outliers*. Quality Press, 1993.

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.

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*, 2024b.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

Philippe Tillet and David Cox. Triton: an intermediate language and compiler for tiled neural network computations. In *Proceedings of the 3rd ACM SIGPLAN International Workshop on Machine Learning and Programming Languages*, pages 10–19, 2019.

Hervé Abdi. Coefficient of variation. *Encyclopedia of research design*, 1(5):169–171, 2010.

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.

Xuewen Liu, Zhikai Li, Jing Zhang, Mengjuan Chen, and Qingyi Gu. Ptq4arvg: Post-training quantization for autoregressive visual generation models. *arXiv preprint arXiv:2601.21238*, 2026.

Zhiteng Li, Hanxuan Li, Junyi Wu, Kai Liu, Haotong Qin, Linghe Kong, Guihai Chen, Yulun Zhang, and Xiaokang Yang. Dvd-quant: data-free video diffusion transformers quantization. In *International Conference on Learning Representations*, volume 2026, pages 35463–35475, 2026.

Kai Liu, Shaoqiu Zhang, Linghe Kong, and Yulun Zhang. Clq: Cross-layer guided orthogonal-based quantization for diffusion transformers. *arXiv preprint arXiv:2509.24416*, 2025b.

A Additional Visualization of Outlier Patterns

Figure 8, Figure 9, and Figure 10 show the outlier patterns of all 10 layers in block 0, 10, and 29.

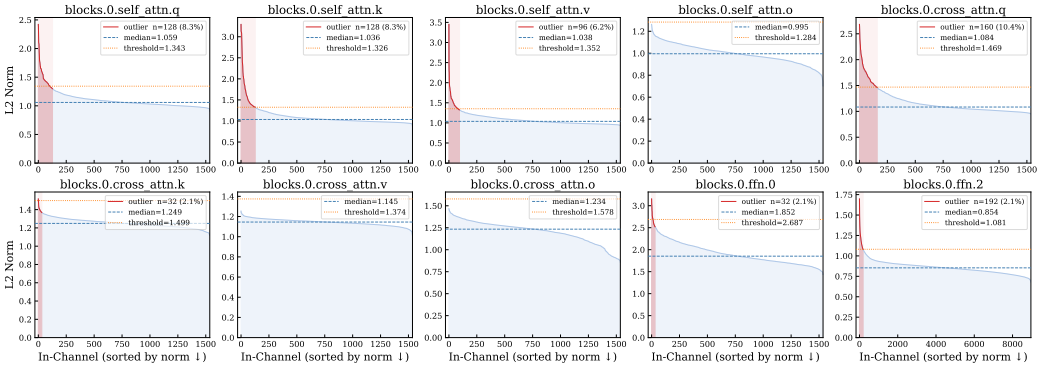

Figure 8: Outlier patterns of all layers in block 0.

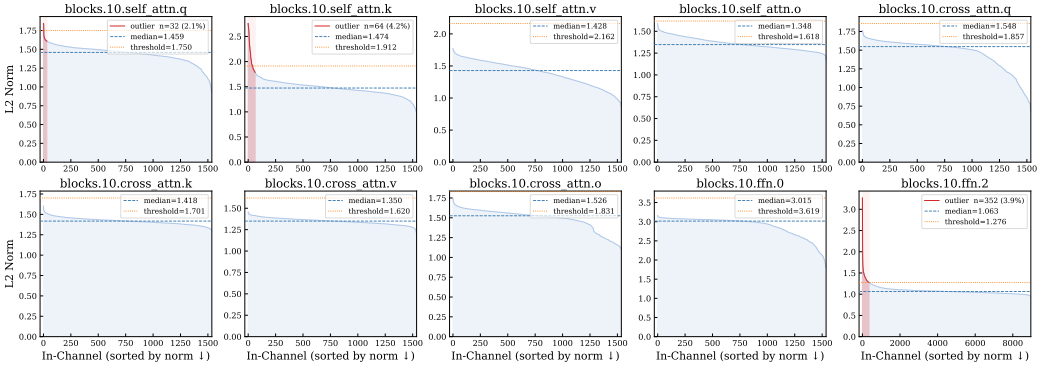

Figure 9: Outlier patterns of all layers in block 10.

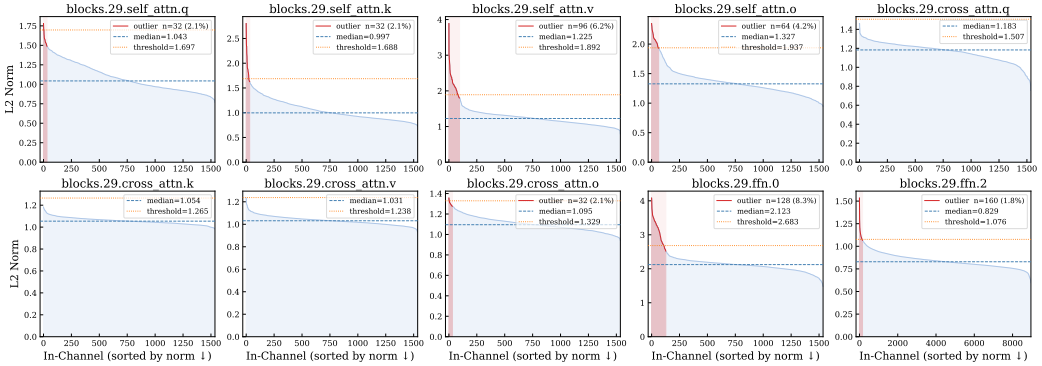

Figure 10: Outlier patterns of all layers in block 29.

B Additional Samples for Visual Comparison

Figure 11, Figure 12, and Figure 13 show visual results on self-forcing using bitwidth W4A8, W4A6, and W8A8. Figure 14, Figure 15, and Figure 16 show visual results on causal-forcing using bitwidth W4A8, W4A6, and W8A8. These qualitative results show that our method preserves video quality well in all bitwidth settings, while baselines induce noticeable quality degradation in low-bit W4A8 and W4A6 settings.

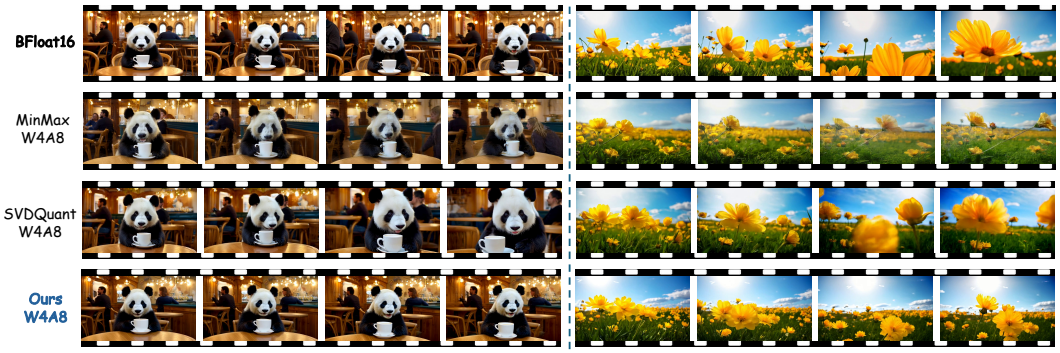

Figure 11: Visual comparison of the self-forcing model using W4A8.



Figure 12: Visual comparison of the self-forcing model using W4A6.



Figure 13: Visual comparison of the self-forcing model using W8A8.

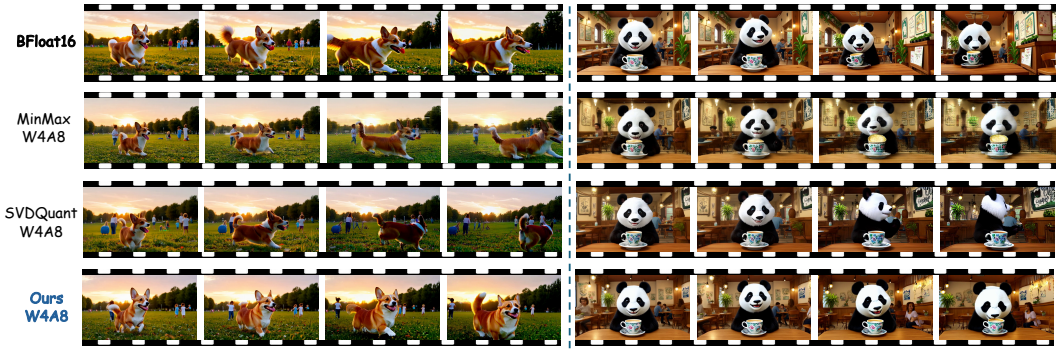

Figure 14: Visual comparison of the causal-forcing model using W4A8.

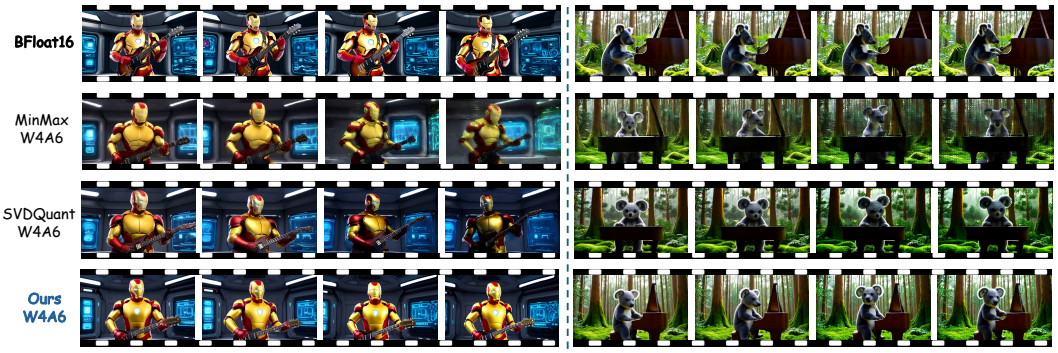

Figure 15: Visual comparison of the causal-forcing model using W4A6.



Figure 16: Visual comparison of the causal-forcing model using W8A8.

C Additional Implementation Details and Extra Cost Analysis

Quantization Implementation. We find that the time embedding and projection modules, and the final prediction head are important to maintaining model performance. Considering their tiny size, we preserve them in BFloat16; In the reconstruction, we set the batch size to 8, the learning rates of rounding and scale to 2e-3 and 4e-5 respectively, and the total training iterations to 2000. We use the Adam optimizer with cosine annealing schedule.

Quantization Kernels and Extra Cost Analysis. We use Triton (Tillet and Cox, 2019) to implement quantization kernels. We divide the quantization into two kernels. (i) Activation quantization kernel, which quantizes input float activations to INT8. (ii) INT8 GEMM and de-quantization kernel, which completes matrix multiplication of INT8 weight and INT8 activation, and also performs de-quantization based on scales. For our dual-scale quantization, we extract the outlier channels and normal channels offline, so we only need to extract the corresponding activations online (do permutation based on the split of outlier and normal weight channels), and we observe this cost is negligible. Besides, the dual-scale is only applied to part of the layers which have outliers, and remains unchanged for smooth layers. Overall, we observe a 1.30x (18.02s to 13.85s) latency speedup on NVIDIA A6000 GPU and a 1.97x (2.64GB to 1.34 GB) model size reduction, with a batch size of 2 and default inference configurations of self-forcing (4 denoising steps, 7 chunks, 21 frames, etc).

D Discussion about Related Outlier-handling Methods

How to address outlier issue is a long-standing question in the field of model quantization. As shown in Equation (7), outliers usually increase the quantization errors. There are several mainstream paradigms to tackle outliers. Scaling-based methods, exemplified by SmoothQuant (Xiao et al., 2023), scale the weights and activations simultaneously in a channel-wise way, to suppress outliers in activations. However, it is not a free lunch, since it just transfer the quantization difficulty from activations to weights. Rotation-based methods, represented by QuaRot (Ashkboos et al., 2024), apply orthogonal transformations to weights and activations, aiming to obtain smoother

distributions. Nevertheless, there is no theoretical guarantee that these rotated distributions are consistently favorable, and rotation can sometimes amplify certain channels in practice. Moreover, offline rotation is inapplicable in DiTs due to the usage of adaptive normalization layers (Peebles and Xie, 2023) and non-linear activation functions, while online rotation incurs notable extra overhead (Li et al., 2025a,b; Liu et al., 2026). Finally, low-rank branch methods, such as SVDQuant (Li et al., 2025a), mitigate weight outliers by absorbing them into a high-precision low-rank branch, which inevitably incurs additional computational overhead. Our outlier adaptive dual-scale method isolates the outliers from normal values, thereby reducing the scaling factor for normal channels. Guaranteed by Equation (7), the reduced scaling factor will theoretically ensure lower quantization errors. Besides, the dual-scale strategy only incurs negligible extra costs (the extraction/permutation of activations according to the split of outlier and normal weight channels), and is only applied to part of the layers which have outliers.

E Ablation of Minimum Constraint Alpha

In Equation (12), we introduce α to impose a minimum outlier threshold, which helps prevent false outlier detection. Specifically, α should not be set too large, as this would raise the outlier threshold. Conversely, it should not be too small, as this may make the constraint ineffective and lead to false detection. We set $\alpha = 1.20$ in all experiments and find it works well. Nevertheless, we further conduct an ablation study on α, as reported in Table 5. The results show that our method is robust over a reasonable range of α.

Table 5: Ablation of minimum constraint α using self-forcing.

alpha	1.10	1.15	1.20	1.25	1.30
FVD	115.28	117.26	116.26	114.35	119.96
LPIPS	0.460	0.462	0.466	0.463	0.462

F Additional Baselines

We compare to recent baselines DVD-Quant (Li et al., 2026) and CLQ (Liu et al., 2025b). The results are shown in Table 6, which show our method consistently outperforms these baselines.

G Calibration Costs

We report the calibration costs of our quantization process. Specifically, dual-scale outlier detection is very fast, taking only a few seconds. Frame-wise sensitivity estimation takes about 0.35 hours, while the subsequent reconstruction takes 1.75 hours.

H Real Low-bit Deployment Results

Table 7 shows the results of the real deployment of three bitwidth setting. Note that, since current CUDA kernels does not provide a native INT6 activation operation, W4A6 is executed through the W4A8-compatible kernel by representing the 6-bit activations in an INT8 container. So, we use W4A6 as an aggressive low-bit quality stress test, which is a common practice in quantization literature (e.g., Q-VDiT (Feng et al., 2025b), DVD-Quant (Li et al., 2026)), which could be realized by future native sub-8-bit kernels.

I Causal-Forcing Ablation

We further conduct ablation studies on the causal-forcing model. As shown in Table 8, both dual-scale quantization and frame-wise weighting consistently improve performance under W4A8 and W4A6 settings, and their combination achieves the best results.

Table 6: Comparison with additional baselines.

Setting	DVD-Quant		CLQ		Ours	
	FVD-FP ↓	LPIPS-FP ↓	FVD-FP ↓	LPIPS-FP ↓	FVD-FP ↓	LPIPS-FP ↓
Self-forcing W4A8	148.60	0.493	174.38	0.497	**116.26**	**0.466**
Causal-forcing W4A8	143.55	0.490	245.16	0.522	**106.04**	**0.452**
Self-forcing W8A8	70.64	0.359	84.82	0.389	**64.51**	**0.327**
Causal-forcing W8A8	65.21	0.360	75.82	0.404	**61.67**	**0.335**

Table 7: Inference latency and model size under different bit-width settings.

Bit-width	Latency	Model Size
Bfloat16	18.02s	2.64GB
W8A8	13.85s (1.30×)	1.34GB (1.97×)
W4A8	12.97s (1.39×)	0.67GB (3.94×)
W4A6	12.97s (1.39×)	0.67GB (3.94×)

Table 8: Ablation study on the causal-forcing model.

Dual Scale	Frame Weighting	W4A8 FVD-FP / LPIPS-FP	W4A6 FVD-FP / LPIPS-FP
×	×	143.49 / 0.463	233.43 / 0.507
×	✓	133.72 / 0.453	206.15 / 0.497
✓	×	117.44 / 0.458	180.40 / 0.499
✓	✓	**106.04 / 0.452**	**140.38 / 0.486**

J Limitations

While our method identifies an exponential-like frame-wise sensitivity, it currently leverages this property only during quantization reconstruction. This sensitivity could potentially be extended to other techniques, such as mixed-precision quantization. Besides, while we deploy our model using Triton, manually implementing CUDA kernels with dedicated optimizations might offer additional efficiency gains.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly claim our method and contributions, i.e., a novel quantization framework for autoregressive video diffusion models, which can effectively improve inference efficiency.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in the last section of the Appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For Equation (7), we provide the assumption that the rounding error follows the uniform distribution.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: In Section 4.1, we detailedly introduce the models and baselines, quantization implementation, and benchmark and metrics.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Although all our experiments are based on existing open-source codes and data, we do not submit our codes now. We will release our codes after the review.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: In Section 4.1, we detailedly introduce experimental settings, i.e., the models and baselines, quantization implementation, and benchmark and metrics.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: Due to the high computational costs and the standard practice of single-run reporting in this field, we do not provide error bars.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We partially report the used computing resource, i.e., we use A6000 to benchmark the actual deployment efficiency.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We strictly follow the NeurIPS Code of Ethics in our research.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: We do not find societal impact caused by our work. We only aim to reduce the deployment costs of autoregressive video diffusion models.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our work is based on open-source data and models, so it does pose such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use the open-source data and models. We cite them and follow the license.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer:[N/A]

    Justification: The paper does not release new assets now, and we will release our codes after the review.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer:[N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.