TURBOQUANTCONV: CHANNEL-ISOLATED INCOHERENCE PROCESSING FOR 3D CONVOLUTIONS IN VIDEO GENERATIVE AI

Anonymous authors
Paper under double-blind review

ABSTRACT

Three-dimensional Convolutional Variational Autoencoders (3D-VAEs) serve as the fundamental compression backbone of modern video generative models, yet their deep 3D convolutions (Conv3D) impose prohibitive computational and memory burdens. Pushing post-training quantization (PTQ) down to 4-bit weights and activations (W4A4) typically causes substantial quality degradation due to non-stationary spatiotemporal activation bursts. In this paper, we propose TurboQuantConv, a channel-isolated incoherence processing and quantization framework for deep multi-dimensional convolutions. By harmonizing outlier suppression with convolutional receptive field geometry, TurboQuantConv effectively suppresses spatiotemporal bursts and achieves high-fidelity quantization without requiring fine-tuning or retraining. Under W4A4 compression, TurboQuantConv outperforms prior state-of-the-art PTQ methods by up to $+3.92$ dB over rotated RTN (QuaRot) and $+8.88$ dB over SVDQuant PSNR across 3D-VAE benchmarks. In downstream 81-frame video generation with the 5B Wan 2.2 Diffusion Transformer, TurboQuantConv achieves the lowest FVD (152.6) while closely preserving unquantized visual dynamics, avoiding the severe motion freeze and chromatic collapse from which existing PTQ methods suffer. These results establish a practical foundation for efficient edge deployment of video generative models.

1 INTRODUCTION

Modern video foundation models—predominantly Latent Diffusion Models (Rombach et al., 2022; Blattmann et al., 2023) and Diffusion Transformers (DiTs) (Yang et al., 2025; Team Wan et al., 2025)—rely on 3D spatial-temporal Variational Autoencoders (3D-VAEs) (Yu et al., 2024; NVIDIA et al., 2025) to compress high-dimensional video volumes ($T \times H \times W \times 3$) into compact latent manifolds. While attention layers in DiTs have received extensive optimization, 3D-VAE decoders remain a dominant computational and memory bandwidth bottleneck (Cheng & Yuan, 2025; Zhu et al., 2026): as DiT denoising accelerates, 3D-VAE decoding accounts for up to 30%–70% of end-to-end latency and saturates GPU memory bandwidth with high-resolution activations (Zhuang et al., 2026; Zhu et al., 2026). Composed of deep cascades of 3D spatial-temporal convolutions (Conv3D) across wide channels, 3D-VAEs impose massive FLOP demands, severely impeding real-time execution and on-device deployment.

Post-Training Quantization (PTQ) (Jacob et al., 2018; Gholami et al., 2022) offers a direct route to accelerate inference by replacing floating-point arithmetic with low-bit integer operations without retraining. However, pushing PTQ down to 4-bit weights and activations (W4A4) causes severe quality degradation in 3D-VAEs. This collapse stems from a severe *outlier pathology* in multi-channel video convolutions: sparse, extreme channel-wise outliers stretch uniform scalar grids (Xiao et al., 2023; Li et al., 2025), zeroing normal channel signals, while localized temporal bursts and Cartesian hypercube mismatch further compound quantization distortion. Consequently, existing PTQ methods suffer severe chromatic banding, motion freezing, and frame collapse at 4 bits.

While randomized orthogonal rotations (e.g., Walsh-Hadamard Transforms) successfully suppress outliers in Large Language Models (LLMs) (Chee et al., 2023; Ashkboos et al., 2024; Liu et al., 2025; Zandieh et al., 2026; Pereira et al., 2026) and DiTs (Lee et al., 2026), existing frameworks

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

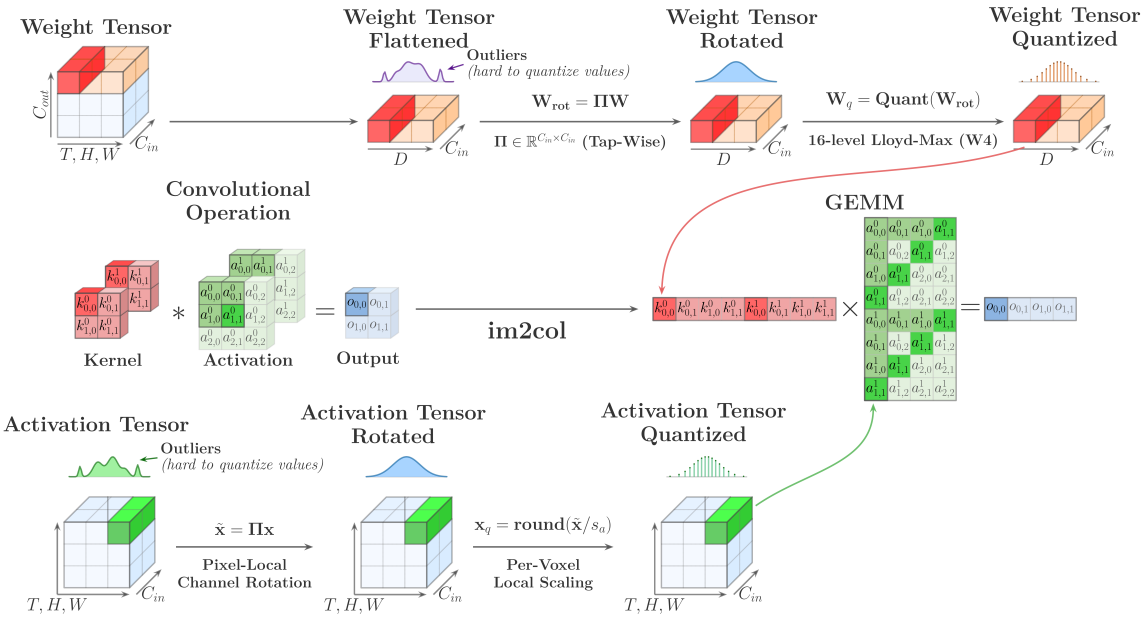

Figure 1: **Overview of the TurboQuantConv Framework for 3D Video VAEs and Convolutions**. *Top branch*: Offline weight quantization flattens 3D convolutional kernels along input channels C_{in}, applying Rademacher sign dithering and Fast Walsh-Hadamard Transform (FWHT) rotations to homogenize outlier spikes into isotropic Gaussians, followed by spherical Lloyd-Max codebook mapping. *Bottom branch*: Online activations undergo single-pass sign inversion and butterfly additions along C_{in}, followed by dynamic L_2 scale extraction and low-bit centroid lookup. *Center*: Rotated tensors map directly onto integer GEMM systolic arrays via spatial-temporal patch unrolling (*im2col*), preserving inner products without fine-tuning.

are fundamentally restricted to 1D token sequences and 2D matrix multiplications. They cannot be directly applied to multi-dimensional convolutions, where naively rotating sliding receptive fields incurs a prohibitive $\mathcal{O}(K_t K_h K_w)$ redundant re-rotation overhead across overlapping spatial-temporal windows.

To bridge this gap, we introduce **TurboQuantConv**, a channel-isolated R-FWHT quantization framework tailored for deep convolutions and 3D video VAEs (Figure 1). We resolve this sliding receptive field dilemma for the first time by proving the principle of channel-isolated rotation: structuring randomized orthogonal rotation **strictly along the input channel dimension** C_{in} guarantees exact linear inner-product preservation in unrolled GEMM while rotating each spatial-temporal activation voxel **strictly once** in a single streaming pass. Executed via Randomized Fast Walsh-Hadamard Transform (R-FWHT), this channel-wise rotation completely bypasses the $\mathcal{O}(K_t K_h K_w)$ redundant re-rotation barrier and enables seamless operator fusion with preceding normalization layers. Furthermore, the rotated isotropic Gaussian features are decomposed into radial scales and directional unit vectors on $\mathbb{S}^{m-1}$, which are mapped onto rate-distortion optimal spherical Lloyd-Max codebooks.

In summary, our key contributions are three-fold:

- **Resolving Receptive Field Redundancy via Channel-Wise Incoherence**: We resolve the sliding receptive field dilemma in multi-dimensional convolutions, proving that channel-isolated orthogonal rotation guarantees exact inner-product preservation for unrolled GEMMs with zero redundant re-rotations across overlapping windows, enabling multiplier-free R-FWHT execution in $\mathcal{O}(C_{\text{in}} \log C_{\text{in}})$.
- **Unified Isotropic Gaussian Quantization**: We exploit the spherical Gaussianization of rotated features, decomposing spatial-temporal vectors into radial magnitudes and directional unit vectors on $\mathbb{S}^{m-1}$ quantized via precomputed Lloyd-Max codebooks or hardware-aligned sub-vector uniform integer lattices (Section 5) without training.

- **Empirical Validation on Video VAEs and DiT Pipelines**: Across WAN 2.2 (705M) and Lightricks LTX-Video ($8,192\times$ downsampling), alongside full Wan 2.2 5B DiT video generation, TurboQuantConv achieves state-of-the-art W4A4 fidelity (33.41 dB and 29.21 dB PSNR, 152.6 FVD), consistently outperforming prior PTQ and low-rank SVD baselines by up to $+15.56$ dB.

2 Related Work

2.1 PTQ Baselines and the Outlier Dilemma

Post-Training Quantization (PTQ) compresses deep networks and enables integer-arithmetic acceleration without demanding expensive retraining or full dataset access (Jacob et al., 2018; Nagel et al., 2019; Gholami et al., 2022). Early PTQ methods relied on uniform round-to-nearest (RTN) grids, which suffer severe accuracy loss at low bitwidths. Advanced reconstruction methods optimize layer-wise objectives via local Hessians and learned rounding (Frantar et al., 2023; Nagel et al., 2020; Li et al., 2021; Wei et al., 2022). However, when applied to deep generative models, activations exhibit extreme channel-wise outliers that dominate the dynamic range (Xiao et al., 2023).

To address outlier pathology in Transformers and diffusion backbones, prior PTQ algorithms rely on channel scaling migration and low-rank residual decomposition. Methods such as SmoothQuant (Xiao et al., 2023), AWQ (Lin et al., 2024), and OmniQuant (Shao et al., 2024) introduce per-channel diagonal scales $\mathbf{S}$ to migrate dynamic range difficulty from activations to weights via $\mathbf{WX} = (\mathbf{WS})(\mathbf{S}^{-1}\mathbf{X})$. Extending this procedure, SVDQuant (Li et al., 2025) first applies weight-activation smoothing migration and subsequently absorbs residual outliers into a 16-bit floating-point low-rank branch: $\mathbf{WX} \approx \hat{\mathbf{W}}\hat{\mathbf{X}} + \mathbf{L}_1\mathbf{L}_2\mathbf{X}$ ($R \in \{16, 32\}$). While effective in self-attention where outliers reside in static channels, both mechanisms struggle in 3D convolutions: non-stationary temporal bursts break static diagonal scaling, while high-frequency spatial textures and temporal motion disperse outlier energy across high-dimensional subspaces, causing severe outlier leakage or requiring impractically high ranks ($R \geq 128$) that violate memory budgets.

2.2 Incoherence Processing and Spherical Vector Quantization

An alternative, mathematically principled line of research suppresses outliers by rotating coordinates into an incoherent space prior to quantization via randomized orthogonal transformations (Ailon & Chazelle, 2009), spreading localized outlier energy uniformly across all dimensions. In LLMs, QuIP (Chee et al., 2023) and QuIP# (Tseng et al., 2024) introduced randomized Hadamard rotations with lattice codebooks, while QuaRot (Ashkboos et al., 2024) and SpinQuant (Liu et al., 2025) demonstrated end-to-end 4-bit rotation schemes. Recently, **TurboQuant** (Zandieh et al., 2026) introduced data-oblivious polar vector quantization on the hyper-sphere, and **Fast-TurboQuant** (Pereira et al., 2026) proposed a multiplier-free architecture using Rademacher sign inversions and Fast Walsh-Hadamard Transforms (FWHT). Concurrently, **OrbitQuant** (Lee et al., 2026) investigated block-Hadamard rotations for linear attention in video DiTs. While effective for independent 1D tokens, extending this paradigm to multi-dimensional convolutions introduces an intractable computation barrier: naively rotating unrolled receptive field patches forces severe $\mathcal{O}(K_t K_h K_w)$ redundant re-rotations (e.g., $27\times$ for $3 \times 3 \times 3$ kernels) across overlapping sliding windows. To bridge this gap, as proven in Section 3, TurboQuantConv introduces a fundamentally distinct *channel-isolated* rotation mechanism: by decoupling spatial-temporal gathering from channel rotation, each activation voxel is transformed strictly once, enabling seamless operator fusion with preceding normalization while guaranteeing exact inner-product invariance.

2.3 3D Spatial-Temporal VAEs in Video Generative Models

High-resolution video synthesis relies heavily on Latent Diffusion Models (Rombach et al., 2022; Blattmann et al., 2023) and Diffusion Transformers (Yang et al., 2025; Team Wan et al., 2025). In these architectures, 3D-VAEs or video tokenizers serve as the critical compression engine (Yu et al., 2024; NVIDIA et al., 2025; Team Wan et al., 2025), compressing high-dimensional video volumes ($T \times H \times W \times 3$) into compact latent manifolds. However, 3D-VAE decoders are composed of deep cascades of 3D spatial-temporal convolutions, residual blocks (He et al., 2016), and upsampling operators. Because the 3D-VAE decoder represents the final rendering stage of video synthesis,

quantization errors in early convolutional layers compound non-linearly across subsequent blocks (Arai & Ichikawa, 2025; Zhao, 2026), translating into severe frame-to-frame flickering, edge ringing, and color distortion. Efficient, robust low-bit quantization of 3D-VAEs is therefore indispensable for edge deployment.

3 METHODOLOGY

In this section, we present the mathematical foundation, algorithmic architecture, and hardware mapping of **TurboQuantConv** (Figure 1). The pipeline operates via two coordinated streams: (1) offline, data-free weight rotation and codebook quantization, and (2) low-latency online activation rotation and quantization during streaming inference, converging into high-throughput GEMM inner-product contractions without fine-tuning.

3.1 CONVOLUTION AS INNER-PRODUCT CONTRACTION

In 3D-VAEs (Team Wan et al., 2025; Yu et al., 2024; NVIDIA et al., 2025), spatial-temporal convolutions represent the primary computational and memory bottleneck. Consider a multi-dimensional convolutional layer with weight tensor $\boldsymbol{\mathcal{W}} \in \mathbb{R}^{C_{\text{out}} \times C_{\text{in}} \times K_t \times K_h \times K_w}$ and input feature map $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{C_{\text{in}} \times T_{\text{in}} \times H_{\text{in}} \times W_{\text{in}}}$. Using spatial-temporal patch unrolling (*im2col*), convolution is transformed into dense matrix multiplication:

$$\mathbf{Y} = \mathbf{W}\mathbf{X}, \quad \text{with } \mathbf{W} \in \mathbb{R}^{C_{\text{out}} \times d},\ \mathbf{X} \in \mathbb{R}^{d \times N}, \tag{1}$$

where inner contraction dimension is $d = C_{\text{in}} K_t K_h K_w$, and $N = T_{\text{out}} H_{\text{out}} W_{\text{out}}$ denotes patch positions.

Channel-Wise Rotation for Receptive Field Invariance. In Transformers, GEMM unrolls along feature embeddings. For convolutions, naive unrolling groups kernel coordinates (K_t, K_h, K_w) with channels C_{in}. Because sliding receptive fields heavily overlap, rotating across the full unrolled dimension d would repeatedly re-rotate the same activations across overlapping windows, causing an $\mathcal{O}(K_t K_h K_w)$ redundant computation penalty. To resolve this, we structure randomized orthogonal rotation strictly along the **channel dimension C_{in} alone** (broadcasting identically across all kernel taps). Consequently, each spatial-temporal feature vector $\mathbf{x}(t, h, w) \in \mathbb{R}^{C_{\text{in}}}$ is rotated **strictly once** in a single streaming pass.

Proposition 1 (Inner-Product Invariance of Channel Rotation)**.** *Let $\boldsymbol{\Pi} \in \mathbb{R}^{C_{in} \times C_{in}}$ satisfy $\boldsymbol{\Pi}^\top \boldsymbol{\Pi} = \mathbf{I}_{C_{in}}$. If $\boldsymbol{\Pi}$ is applied along channels, $\tilde{\boldsymbol{\mathcal{W}}}_{k,:,\tau,\delta_h,\delta_w} = \boldsymbol{\Pi}\boldsymbol{\mathcal{W}}_{k,:,\tau,\delta_h,\delta_w}$ and $\tilde{\mathbf{x}}(t, h, w) = \boldsymbol{\Pi}\mathbf{x}(t, h, w)$, then for all output channels k and coordinates (t, h, w):*

$$(\tilde{\boldsymbol{\mathcal{W}}} * \tilde{\boldsymbol{\mathcal{X}}})_{k,t,h,w} = (\boldsymbol{\mathcal{W}} * \boldsymbol{\mathcal{X}})_{k,t,h,w}, \quad \textit{and} \quad \tilde{\mathbf{W}}\tilde{\mathbf{X}} = \mathbf{W}\mathbf{X} = \mathbf{Y}. \tag{2}$$

Each activation voxel $\mathbf{x}(t, h, w)$ is rotated strictly once across sliding windows (proof in Appendix A.1).

3.2 MULTIPLIER-FREE RANDOMIZED FAST WALSH-HADAMARD TRANSFORM

Randomized Fast Walsh-Hadamard Construction. Dense orthogonal rotations require $\mathcal{O}(C_{\text{in}}^2)$ floating-point multiplications, imposing substantial latency during online activation processing. To maximize inference efficiency, TurboQuantConv constructs $\boldsymbol{\Pi}$ using R-FWHT (Ailon & Chazelle, 2009; Pereira et al., 2026):

$$\boldsymbol{\Pi} = \frac{1}{\sqrt{m}} \mathbf{H}_m \mathbf{D}_{\pm}, \tag{3}$$

where $m = 2^{\lceil \log_2 C_{\text{in}} \rceil}$ is the next power of two, $\mathbf{D}_{\pm} = \text{diag}(\sigma_1, \ldots, \sigma_m)$ with $\sigma_i \in \{-1, +1\}$ i.i.d. Rademacher variables, and $\mathbf{H}_m$ is the recursive Walsh-Hadamard matrix: $\mathbf{H}_{2k} = \left[\begin{smallmatrix} \mathbf{H}_k & \mathbf{H}_k \\ \mathbf{H}_k & -\mathbf{H}_k \end{smallmatrix}\right]$ with $\mathbf{H}_1 = [1]$.

Multiplier-Free Hardware Execution. The factorization in Equation (3) enables entirely multiplier-free online execution: (1) **Rademacher Sign Dithering**: Multiplication by $\mathbf{D}_{\pm}$ corresponds to conditional sign-bit inversion, implemented via bitwise XOR toggles with **zero ALU multipliers**; (2) **Butterfly FWHT**: Multiplication by $\mathbf{H}_m$ is executed via butterfly additions/subtractions

Algorithm 1 TurboQuantConv Online Activation Quantization

Require: Input activation $\boldsymbol{\mathcal{X}} \in \mathbb{R}^{C_{\text{in}} \times T_{\text{in}} \times H_{\text{in}} \times W_{\text{in}}}$; Rademacher key $\sigma \in \{-1, +1\}^m$; spherical codebook $\mathcal{C}_{\text{sphere}}$.

Ensure: Quantized activation tensor $\hat{\boldsymbol{\mathcal{X}}} \in \mathbb{R}^{C_{\text{in}} \times T_{\text{in}} \times H_{\text{in}} \times W_{\text{in}}}$.

1: **for** each spatial-temporal coordinate (t, h, w) **do**
2: Extract channel vector $\mathbf{x} \in \mathbb{R}^{C_{\text{in}}}$ and zero-pad to $m = 2^{\lceil \log_2 C_{\text{in}} \rceil}$
3: $\mathbf{x}' \leftarrow \mathbf{x} \odot \sigma$ {Rademacher dithering via XOR (0 muls)}
4: $\tilde{\mathbf{x}} \leftarrow \text{FWHT}(\mathbf{x}')$ {Walsh-Hadamard butterfly additions (0 muls)}
5: $r_x \leftarrow \|\tilde{\mathbf{x}}\|_2, \quad \mathbf{u}_x \leftarrow \tilde{\mathbf{x}} / \max(r_x, 10^{-12})$ {Polar norm}
6: $\hat{\mathbf{u}}_x \leftarrow \arg\min_{\mathbf{c} \in \mathcal{C}_{\text{sphere}}} \|\mathbf{u}_x - \mathbf{c}\|_2^2$ {Spherical centroid lookup}
7: $\hat{\mathbf{x}}(t, h, w) \leftarrow r_x \cdot \hat{\mathbf{u}}_x[1 : C_{\text{in}}]$ {Reconstruct channel vector}
8: **end for**
9: **return** Quantized activation tensor $\hat{\boldsymbol{\mathcal{X}}}$

in $m \log_2 m$ operations with **zero ALU multipliers**; (3) **Normalization Absorption**: The global scale $1/\sqrt{m}$ is absorbed directly into the precomputed codebook centroids offline.

Theorem 1 (Outlier Dispersion via R-FWHT). *Let $\mathbf{x} \in \mathbb{R}^m$ with L_2 norm $\|\mathbf{x}\|_2$, and $\boldsymbol{\Pi} = \frac{1}{\sqrt{m}} \mathbf{H}_m \mathbf{D}_{\pm}$. Then rotated coordinates $\tilde{x}_i = (\boldsymbol{\Pi}\mathbf{x})_i$ are sub-Gaussian with parameter $\|\mathbf{x}\|_2 / \sqrt{m}$, and with probability at least $1 - \delta$:*

$$\|\tilde{\mathbf{x}}\|_\infty \leq \sqrt{\frac{2 \ln(2m/\delta)}{m}} \|\mathbf{x}\|_2 . \tag{4}$$

Sub-Gaussian tail bound derivations via Hoeffding's inequality and union bound are in Appendix A.2.

3.3 Spherical and Polar Codebook Vector Quantization

Following rotation, feature vectors $\tilde{\mathbf{v}} \in \mathbb{R}^m$ follow an isotropic Gaussian distribution in $\mathbb{R}^m$, whose normalized directional unit vectors lie on $\mathbb{S}^{m-1}$. We decompose $\tilde{\mathbf{v}}$ via polar decomposition:

$$r = \|\tilde{\mathbf{v}}\|_2 , \quad \mathbf{u} = \frac{\tilde{\mathbf{v}}}{r} \in \mathbb{S}^{m-1}, \quad \text{and} \quad \hat{\mathbf{u}} = \arg\min_{\mathbf{c} \in \mathcal{C}_{\text{sphere}}} \|\mathbf{u} - \mathbf{c}\|_2^2 , \tag{5}$$

where $r \in \mathbb{R}_+$ is the radial magnitude and $\mathbf{u} \in \mathbb{S}^{m-1}$ is quantized using optimal spherical Lloyd-Max centroids $\mathcal{C}_{\text{sphere}} = \{\mathbf{c}_1, \ldots, \mathbf{c}_{2^b}\}$ precomputed on standard normal quadrature (Max, 1960; Lloyd, 1982; Zandieh et al., 2026). Radial scales are scalar-quantized to low-precision float $\hat{r}$, yielding reconstructed vector $\hat{\mathbf{v}} = \hat{r}\hat{\mathbf{u}}$.

Proposition 2 (Rate-Distortion Superiority over Coarse Cartesian Scalar Grids). *For $\mathbf{u} \sim \mathcal{U}(\mathbb{S}^{m-1})$, expected distortion under spherical Lloyd-Max quantization with b bits satisfies $\mathbb{E}[\|\mathbf{u} - \hat{\mathbf{u}}\|_2^2] \leq C_b 2^{-2b}$, depending solely on spherical packing density. In contrast, coarse full-channel uniform Cartesian grids ($g = m$) incur distortion growing linearly with dimension m and suffer corner-clipping at $b = 4$ (whereas partitioning R-FWHT Gaussianized features into fine-grained sub-vectors $g \ll m$ bounds this dimensionality penalty, forming our complementary integer backend in Section 5; formal derivations in Appendix A.3).*

The complete online activation quantization sequence is formalized in Algorithm 1.

4 Experiments

In this section, we present an empirical evaluation of TurboQuantConv across isolated 3D-VAE video compression and full end-to-end video generative synthesis with DiTs. We benchmark TurboQuant-Conv against six competitive PTQ algorithms across standard and generative benchmarks, followed by ablations isolating each theoretical component.

4.1 Experimental Setup

Target Architectures and Downstream Pipeline. We evaluate across two representative 3D-VAE architectures and a full downstream generative pipeline: (1) **WAN 2.2 3D-VAE** (Team Wan et al.,

Table 1: **W4A4 reconstruction performance across 3D-VAE architectures and video benchmarks**. Best results per column are in **bold**; second-best are underlined.

	WAN 2.2 3D-VAE (705M params)						**Lightricks LTX-Video 3D-VAE** (419M params)					
	TokenBench 240P			**DAVIS-2017**			**TokenBench 240P**			**DAVIS-2017**		
Method	**PSNR ↑**	**LPIPS ↓**	**CIE76 ↓**	**PSNR ↑**	**LPIPS ↓**	**CIE76 ↓**	**PSNR ↑**	**LPIPS ↓**	**CIE76 ↓**	**PSNR ↑**	**LPIPS ↓**	**CIE76 ↓**
TurboQuantConv (Ours)	**33.41**	**0.0029**	**2.42**	**26.14**	**0.1546**	**5.24**	**29.21**	**0.2022**	**3.80**	**26.97**	**0.2431**	**5.04**
Standard RTN	17.85	0.0430	16.39	16.33	0.5991	19.52	19.91	0.5207	14.82	19.32	0.5489	14.99
SmoothQuant	22.86	0.0217	10.38	19.31	0.4636	14.25	23.42	0.3501	8.63	22.17	0.4042	10.11
AWQ	19.36	0.0302	12.96	19.22	0.4328	12.92	22.79	0.3254	9.15	22.67	0.3679	8.63
AdaRound	18.49	0.0452	15.24	17.40	0.5991	16.68	20.39	0.4130	12.38	20.40	0.4537	12.03
QuaRot	29.49	0.0066	4.82	23.40	0.3352	8.01	25.48	0.3585	6.22	23.08	0.4048	8.90
SVDQuant	24.53	0.0137	7.24	20.75	0.2979	10.30	26.98	0.2613	5.33	24.66	0.3291	7.05

2025): A high-capacity convolutional autoencoder (705M parameters, 72 Conv3D layers) with $16\times$ spatial and $4\times$ temporal downsampling ($1,024\times$ volumetric compression into 48 latent channels); (2) **Lightricks LTX-Video 3D-VAE** (HaCohen et al., 2025): A high-efficiency video autoencoder (419M parameters, 90 Conv3D layers) featuring an extreme $\mathbf{8,192}\times$ volumetric downsampling ratio ($32\times$ spatial via 4×4 patchification, $8\times$ temporal into 128 latent channels); and (3) **Wan 2.2 TI2V 5B DiT** (Team Wan et al., 2025): A 5-Billion parameter Diffusion Transformer operating over 81-frame video trajectories at 768×1024 resolution (16 fps).

Datasets and Benchmark Protocols. Evaluations span three complementary benchmarks: (1) **TokenBench 240P** (NVIDIA et al., 2025): 492 diverse real-world video sequences (240×240, $T = 17$) spanning human motion, animal locomotion, fluid dynamics, and scenery; (2) **DAVIS-2017**: 150 sequences (240p, $T = 17$) (Perazzi et al., 2016; Pont-Tuset et al., 2017) challenging quantization under rapid camera motion and complex non-rigid deformation; and (3) **Image-to-Video Generation Test Cases (81 frames)**: Full 81-frame generative trajectories (768×1024, 16 fps) synthesized by Wan 2.2 5B DiT (50 steps, guidance 5.0) evaluating long-horizon temporal fidelity.

Evaluation Metrics and Baselines. We evaluate reconstruction fidelity via PSNR (dB ↑), SSIM (Wang et al., 2004) (↑), LPIPS (Zhang et al., 2018) (↓), and CIELAB color difference CIE76 (↓). For end-to-end video synthesis, we evaluate video and frame distribution distances via FVD (Unterthiner et al., 2018) (↓) and FID (Heusel et al., 2017) (↓), while VBench (Huang et al., 2024) scores (Imaging Quality, Aesthetic Quality, Subject Consistency →) closest to the unquantized float golden reference indicate optimal manifold preservation. Frame-level alignment against the unquantized generation is measured via LPIPS (↓) and semantic consistency via CLIP-I (Radford et al., 2021) (↑); because CLIP cosine similarity captures semantic alignment but overlooks feature magnitude cues correlated with perceptual quality (Liao et al., 2026), combining CLIP-I with LPIPS and VBench ensures comprehensive fidelity evaluation. We compare against six established PTQ baselines: standard **RTN**, **SmoothQuant**, **AWQ**, **AdaRound**, **QuaRot**, and **SVDQuant** ($R = 32$). For fair convolutional comparison in Tables 1–3, all PTQ baselines share per-output-channel weight and per-tensor activation min-max scales; per-voxel (per-token) rotated uniform scaling is isolated in Table 4.

Quantization Scope. Following standard low-bit CNN quantization practices (Zhou et al., 2016; Choi et al., 2018), shallow stem convolutions interfacing directly with RGB pixels are kept in floating-point. All remaining deep spatial-temporal convolutions are quantized to W4A4, covering **over** 99% **of total parameters**.

4.2 Quantitative Results on 3D-VAE Video Models

Table 1 reports reconstruction performance on TokenBench 240P and DAVIS-2017 under W4A4 precision across both WAN 2.2 and Lightricks LTX-Video.

Reconstruction Performance Overview. As benchmarked in Table 1, TurboQuantConv achieves state-of-the-art reconstruction across both architectures and benchmarks under W4A4. On WAN 2.2 (TokenBench 240P), our model reaches 33.41 dB PSNR, 0.0029 LPIPS, and 2.42 CIE76, outperforming the strongest baseline (QuaRot) by $+\mathbf{3.92}$ dB, surpassing SVDQuant by $+\mathbf{8.88}$ dB, and exceeding unrotated methods (RTN, SmoothQuant) by over $+\mathbf{10}$–$\mathbf{15}$ dB. On DAVIS-2017, TurboQuantConv leads at 26.14 dB PSNR ($+2.74$ dB over QuaRot). On the extreme-downsampling

Table 2: **End-to-end generative performance on image-to-video generation test cases under W4A4 (81 frames)**. Best results among quantized models are in **bold**; second-best are underlined.

Method	FVD ↓	FID ↓	LPIPS ↓	CLIP-I ↑	VBench I.Q. →	VBench A.Q. →	VBench S.C. →
Float Baseline (BF16)	–	–	–	–	0.7185	0.7492	0.8878
TurboQuantConv (Ours)	**152.6**	**17.15**	**0.0283**	**0.9790**	0.7005	0.7742	**0.8791**
QuaRot	225.6	28.40	0.0682	0.8920	**0.7046**	0.7845	0.8397
SmoothQuant	395.2	85.60	0.1694	0.5248	0.6399	0.5878	0.5956
SVDQuant	292.0	39.80	0.1034	0.7276	0.6287	**0.7555**	0.8629

LTX-Video ($8,192\times$ compression), TurboQuantConv attains 29.21 dB on TokenBench and 26.97 dB on DAVIS, establishing robust fidelity across diverse 3D-VAE topologies.

Why Prior Baselines Fail under W4A4 Convolutions. Pushing weights and activations to 4 bits exposes distinct failure modes across existing PTQ paradigms: (1) *Unrotated Baselines Collapse via Outlier Saturation (RTN, SmoothQuant, AWQ, AdaRound)*: 3D convolutions aggregate temporal receptive fields, concentrating extreme kurtosis along motion coordinates. Without basis rotation, Cartesian clamping faces a fundamental clipping-resolution trade-off: expanding thresholds zeroes normal signals, while clamping truncates dynamic ranges, causing severe chromatic banding and block corruption (16.33–23.42 dB PSNR). (2) *Low-Rank Absorption Fails under Spatiotemporal Dispersion (SVDQuant)*: SVDQuant assumes outliers concentrate in a static low-rank subspace. Dynamic motion and camera sweeps disperse outlier energy across high-frequency spatiotemporal modes; rank-32 truncation leaves heavy outlier leakage in the 4-bit lattice (20.75–26.98 dB PSNR). (3) *Coarse Min-Max Scale Calibration (QuaRot)*: Although Hadamard rotation isotropicizes bursts, under the shared per-tensor min-max protocol extreme frames inflate global step sizes and dilute interior bit resolution (23.08–29.49 dB).

Why TurboQuantConv Excels under Extreme Compression. TurboQuantConv addresses this directly: channel-isolated R-FWHT rotations uniformly disperse spatiotemporal bursts across channels with zero sliding window re-rotation overhead, while per-voxel polar normalization decouples radial magnitudes from unit directions mapped onto Lloyd-Max centroids (or sub-vector scales in Section 5). This eliminates coarse min-max scale dilution, maintaining high structural fidelity and color accuracy across both large-capacity and extreme-downsampling architectures without fine-tuning.

4.3 End-to-End Generative Evaluation on DiTs

While isolated VAE metrics confirm reconstruction capability, the critical test for generative deployment is whether latent representation errors compound during iterative diffusion denoising. In Table 2 and Figure 2, we evaluate coupling quantized WAN 2.2 3D-VAEs with the unquantized Wan 2.2 DiT on downstream video generation.

Compounding Latent Error in Diffusion Trajectories. During 50-step diffusion synthesis, the 3D-VAE acts as the perceptual interface: conditioning latents guide denoising while final latents decode into frames. Distorted latent geometries misguide DiT trajectories, causing distributional collapse. As shown in Table 2, SmoothQuant collapses to 395.2 FVD and 85.60 FID, SVDQuant degrades to 292.0 FVD and 39.80 FID, and QuaRot reaches 225.6 FVD and 28.40 FID. In contrast, TurboQuantConv achieves the lowest distribution distances among all W4A4 methods at $\mathbf{152.6}$ FVD (↓) and $\mathbf{17.15}$ FID (↓) with high perceptual (LPIPS $= 0.0283$) and semantic (CLIP-I $= 0.9790$) fidelity over 81 frames. Fine-grained VBench dimensions consistently corroborate these trends: while degraded baselines occasionally exhibit artificially inflated scores on isolated sub-metrics due to static frame freezing or over-smoothing, TurboQuantConv closely approaches the unquantized float baseline across imaging quality (0.7005 vs. 0.7185), aesthetics (0.7742 vs. 0.7492), and subject consistency (0.8791 vs. 0.8878), consistent with the fidelity gains observed in FVD, FID, and LPIPS.

Downstream Failure Mechanisms and Visual Dynamics. Visual inspection in Figure 2 reveals distinct breakdown mechanisms: (1) *SmoothQuant* suffers activation bursts, causing severe chromatic washout and block tearing; (2) *SVDQuant* truncates spatiotemporal modes via low-rank projection, inducing severe motion freeze; (3) *QuaRot* suffers coarse min-max scale dilution, blurring subtle textures and inducing temporal trajectory drift over 81 frames. In contrast, *TurboQuantConv* preserves

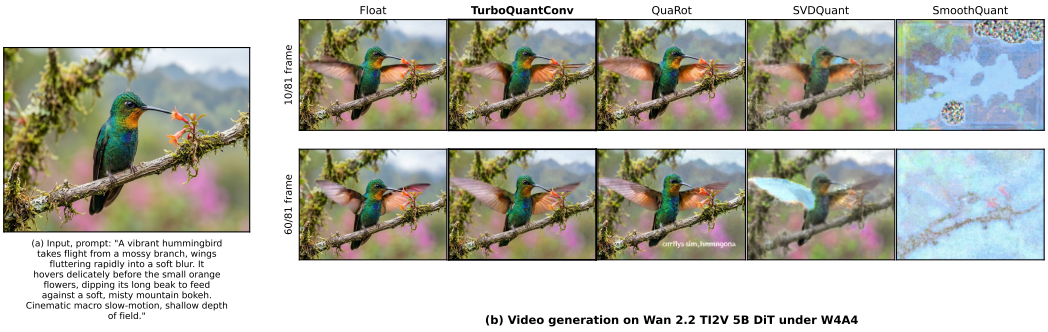

Figure 2: **Qualitative comparison on Wan 2.2 TI2V 5B DiT under W4A4**. (a) Input conditioning frame with text prompt. (b) Video generation under W4A4 at Frames 10 and 60.

spherical geometry: the DiT synthesizes fluid motion and delicate wing flutter, matching float fidelity without freeze or drift.

4.4 Ablation Studies and Analysis

Table 3 reports ablation experiments isolating individual design choices on TokenBench 240P.

Table 3: **Ablation study of TurboQuantConv components on TokenBench 240P (W4A4)**.

Ablation Configuration	PSNR (dB) ↑	SSIM ↑	LPIPS ↓
Unquantized Float Baseline	48.24	0.9981	0.0003
(a) No Rotation + Uniform Grid (Standard RTN)	17.85	0.4455	0.0430
(b) No Rotation + Spherical Codebook	22.14	0.6210	0.0248
(c) R-FWHT Rotation + QuaRot-Style Min-Max RTN	29.49	0.8753	0.0066
(d) R-FWHT Rotation + Spherical Codebook (**TurboQuantConv** (Ours))	**33.41**	**0.9358**	**0.0029**

Impact of Multiplier-Free R-FWHT Rotation. Comparing (a) with (c) and (b) with (d) in Table 3 demonstrates the crucial role of Hadamard rotation. Without rotation, activation bursts saturate quantization bins, causing a ~ 11 dB PSNR drop. Applying R-FWHT eliminates coordinate outliers by spreading energy uniformly across channels, elevating PSNR from 17.85 dB to 29.49 dB under uniform grids, and from 22.14 dB to $\mathbf{33.41}$ dB under spherical codebooks.

Spherical Lloyd-Max Codebook vs. QuaRot-Style Min-Max RTN. Comparing (c) with (d) in Table 3 isolates the benefit of spherical vector quantization and per-voxel polar scaling. Under identical R-FWHT rotation, our spherical codebook achieves $\mathbf{33.41}$ dB PSNR versus 29.49 dB for QuaRot-style min-max RTN—a $+\mathbf{3.92}$ dB gain and 56% LPIPS reduction (0.0029 vs 0.0066). As shown in Section 5 (Table 4), upgrading from the shared per-tensor min-max protocol to per-voxel (per-token) polar symmetric (L_2-normalized $\{-7, \ldots, +7\}$ absmax; see Appendix B.2) scaling contributes $+3.33$ dB ($29.49 \rightarrow 32.82$ dB), Lloyd-Max centroids add $+0.59$ dB ($32.82 \rightarrow 33.41$ dB, $g =$ Full), and sub-vector scaling ($g = 32$) reaches 35.53 dB.

5 Hardware Optimization and Edge Deployment Architectures

Because channel-isolated R-FWHT transforms heavy-tailed spatiotemporal bursts into isotropic Gaussians across channels, TurboQuantConv functions as a unified **Channel-Isolated R-FWHT 3D-VAE Quantization Framework** that decouples outlier suppression from the downstream quantizer. In hardware execution, weight rotations $\mathbf{\Pi}$ are folded offline into convolutional weights (and linear $1 \times 1 \times 1$ skip projections) with zero runtime cost, while online activation R-FWHT butterflies ($C_{\text{in}} \log_2 C_{\text{in}}$ additions) operate point-wise along channels and fuse directly into the preceding activation kernel epilogue (e.g., GroupNorm $\rightarrow$ SiLU) in on-chip SRAM without extra DRAM round-trips. To serve diverse silicon targets, we formulate two complementary quantization backends.

Table 4: **W4A4 reconstruction across R-FWHT backends on WAN 2.2 (TokenBench 240p)**: Option 1 (Gaussian Codebook, $g = \text{Full}$) vs. Option 2 (Uniform Sub-Vector, $g = 32$).

R-FWHT Quantization Backend	PSNR (dB) ↑	SSIM ↑	LPIPS ↓	CIE76 ↓
TurboQuantConv (Option 1: Gaussian Codebook, $g = \text{Full}$)	33.41	0.9358	0.0029	2.42
TurboQuantConv (Polar Symmetric $[-7, +7]$ Uniform, $g = \text{Full}$)	32.82	0.9216	0.0040	2.74
TurboQuantConv (Option 2: Uniform Sub-Vector, $g = 32$)	**35.53**	**0.9498**	**0.0025**	**2.01**

Option 1: Static Gaussian Codebook with Byte-Packed LUT. At full-channel granularity ($g = \text{Full}$), Option 1 maps R-FWHT Gaussianized channels onto precomputed Lloyd-Max centroids with minimal metadata overhead ($< 0.3\%$). To bypass runtime multipliers, we fuse pairwise 4-bit operands into an 8-bit address $\text{addr}_k = (i_{A,k} \ll 4) \vee i_{W,k} \in [0, 255]$ indexing a precomputed 256-entry table $\mathcal{T}_{\text{LUT}} \in \mathbb{R}^{16\times16}$ storing centroid products: $\mathcal{T}_{\text{LUT}}[\text{addr}_k] = \mathcal{C}_A[i_{A,k}] \times \mathcal{C}_W[i_{W,k}]$. Inner products reduce to table gather and accumulation: $\hat{y} = \sum_{k=1}^{d} \mathcal{T}_{\text{LUT}}[(i_{A,k} \ll 4) \vee i_{W,k}]$. With a 512-byte FP16 footprint, $\mathcal{T}_{\text{LUT}}$ fits entirely in L1 scratchpads, eliminating DRAM traffic. Parallel byte-shuffles with conflict-free SRAM access enable multiplier-free execution ideal for custom edge NPUs.

Option 2: Dynamic Sub-Vector Integer Uniform Quantization ($g = 32$). For platforms lacking parallel LUTs, Option 2 maps R-FWHT Gaussianized coordinates onto symmetric uniform 4-bit lattices $q_A, q_W \in \{-7, \ldots, +7\}$: $\hat{y} = \sum_g (s_W^{(g)} \cdot s_X^{(g)}) \sum_{k=1}^{g} q_{W,k} \cdot q_{A,k}$. At full-channel granularity ($g = \text{Full}$), *polar symmetric* $[-7, +7]$ quantization retains Eq. (5)'s per-voxel L_2 radius $r = \|\tilde{\mathbf{x}}\|_2$ while replacing Lloyd-Max centroids with a zero-centered symmetric grid $\{-7, \ldots, +7\}$ and dynamic per-voxel (per-token) absmax scale $s_u = \max_c |u_c|/7$ over C_{in} rotated unit coordinates (effective scale $s = rs_u = \max_c |\tilde{x}_c|/7$ per voxel; per-output-channel for weights). Partitioning into sub-vectors ($g = 32$) eliminates wide-channel saturation, using offline weight scales $s_W^{(g)}$ and online dynamic activation scales $s_X^{(g)}$ to prevent calibration overfitting. While metadata rises to $\sim 12.5\%$, it yields $+2.12$ dB higher PSNR (Table 4) on integer systolic arrays (Tensor Cores, ARM NEON, TPU MXUs) via 128-bit SIMD.

Evaluated under the identical WAN 2.2 (TokenBench 240p, W4A4) layer configuration as Table 1 (68/72 Conv3D layers quantized, where the 4 skipped boundary layers already encompass all non-multiples of 32): (1) *Full-Channel Baseline*: Replacing QuaRot-style per-tensor min-max RTN (29.49 dB in Table 3c) with TurboQuantConv's per-voxel (per-token) polar symmetric $[-7, +7]$ uniform scaling ($g = \text{Full}$) reaches 32.82 dB, yet still trails TurboQuantConv Option 1 by 0.59 dB (32.82 vs. 33.41 dB), confirming Lloyd-Max codebook optimality on rotated Gaussians; (2) *Sub-Vector Recovery*: Within TurboQuantConv, Option 2 ($g = 32$ sub-vector scaling on R-FWHT features) eliminates clipping, boosting W4A4 PSNR by $+\mathbf{2.71}$ dB (to **35.53** dB), SSIM to **0.9498**, and LPIPS to **0.0025**; (3) *Pareto Frontier*: Both options form a unified Pareto frontier under TurboQuantConv—Option 1 achieves the full-channel rate-distortion optimum with static, calibration-free execution and minimal metadata ($< 0.3\%$) for custom LUT NPUs, whereas Option 2 trades dynamic scale extraction and $\sim 12.5\%$ metadata for $+2.12$ dB higher accuracy on commodity integer systolic arrays (extended hardware analysis in Appendix B).

6 Conclusion

We presented **TurboQuantConv**, a channel-isolated R-FWHT quantization framework for 3D convolutions in video VAEs. By resolving the sliding receptive field dilemma via channel-isolated rotation, TurboQuantConv eliminates re-rotations while preserving exact GEMM inner products. Multiplier-free R-FWHT butterfly operations homogenize bursts into isotropic Gaussians, unifying static Gaussian codebooks (Option 1, 512-byte LUT) and dynamic sub-vector uniform integer quantization (Option 2, $g = 32$, reaching 35.53 dB PSNR on WAN 2.2) without fine-tuning. On TokenBench-240P, TurboQuantConv achieves 33.41 dB PSNR at W4A4 ($+3.92$ dB over QuaRot, $+8.88$ dB over SVDQuant). In 81-frame DiT video generation, it achieves 152.6 FVD, preserving high visual fidelity while baselines suffer motion freeze, establishing a practical foundation for efficient edge video generative AI.

AI USE STATEMENT

Generative AI tools (large language models and text-to-image synthesis models) were used during manuscript preparation for: (1) proofreading and spell checking, (2) literature search and citation retrieval, (3) drafting text prompts and synthesizing initial input conditioning images for the text-to-video (TI2V) downstream evaluation, and (4) LaTeX table and formatting scaffolding. Generative AI was strictly not used to generate, simulate, or fabricate experimental data, evaluation metrics, or mathematical proofs. All empirical evaluations, algorithmic implementations, theoretical derivations, and factual claims were independently designed, executed, and rigorously verified by the authors, who assume full responsibility for all contents of this paper.

REPRODUCIBILITY STATEMENT

The mathematical proofs establishing convolution-to-GEMM algebraic invariance, sub-Gaussian coordinate bounds under orthogonal rotations, and spherical rate-distortion bounds are provided in Appendix A. Detailed architectural configurations, benchmark protocols, and evaluation metrics are documented in Section 4.1 and Appendix C. Complete source code, multiplier-free R-FWHT kernels, precomputed spherical codebooks, evaluation pipelines, and shell reproduction scripts are included in the supplementary material and will be fully open-sourced upon publication.

REFERENCES

Nir Ailon and Bernard Chazelle. The fast Johnson–Lindenstrauss transform and approximate nearest neighbors. *SIAM Journal on Computing*, 39(1):302–322, 2009.

Yamato Arai and Yuma Ichikawa. Quantization error propagation: Revisiting layer-wise post-training quantization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2504.09629.

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. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher M. De Sa. QuIP: 2-bit quantization of large language models with guarantees. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 36, 2023.

Yu Cheng and Fajie Yuan. LeanVAE: An ultra-efficient reconstruction VAE for video diffusion models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2025. arXiv:2503.14325.

Jungwook Choi, Zhuo Wang, Swagath Venkataramani, Pierce I-Jen Chuang, Vijayalakshmi Srinivasan, and Kailash Gopalakrishnan. PACT: Parameterized clipping activation for quantized neural networks. *arXiv preprint arXiv:1805.06085*, 2018.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *International Conference on Learning Representations (ICLR)*, 2023.

Amir Gholami, Sehoon Kim, Zhen Dong, Zhewei Yao, Michael W. Mahoney, and Kurt Keutzer. A survey of quantization methods for efficient neural network inference. In *Low-Power Computer Vision: Improve the Efficiency of Artificial Intelligence*, chapter 13, pp. 291–326. Chapman and Hall/CRC, 2022.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. LTX-Video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818, 2024.

Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2704–2713, 2018.

Donghyun Lee, Jitesh Chavan, Duy Nguyen, Sam Huang, Liming Jiang, Priyadarshini Panda, Timo Mertens, and Saurabh Shukla. OrbitQuant: Data-agnostic quantization for image and video diffusion transformers. *arXiv preprint arXiv:2607.02461*, 2026.

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 components for 4-bit diffusion models. In *International Conference on Learning Representations (ICLR)*, 2025.

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 (ICLR)*, 2021.

Zhicheng Liao, Dongxu Wu, Zhenshan Shi, Sijie Mai, Hanwei Zhu, Lingyu Zhu, Yuncheng Jiang, and Baoliang Chen. Beyond cosine similarity: Magnitude-aware CLIP for no-reference image quality assessment. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 2026. arXiv:2511.09948.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for LLM compression and acceleration. In *Conference on Machine Learning and Systems (MLSys)*, 2024.

Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. SpinQuant: LLM quantization with learned rotations. In *International Conference on Learning Representations (ICLR)*, 2025.

Stuart Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2): 129–137, 1982.

Joel Max. Quantizing for minimum distortion. *IRE Transactions on Information Theory*, 6(1):7–12, 1960.

Markus Nagel, Mart van Baalen, Tijmen Blankevoort, and Max Welling. Data-free quantization through weight equalization and bias correction. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1325–1334, 2019.

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 (ICML)*, pp. 7197–7206, 2020.

NVIDIA et al. Cosmos world foundation model platform for physical AI. *arXiv preprint arXiv:2501.03575*, 2025.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Pedro M. R. Pereira, Felipe A. P. de Figueiredo, and Rausley A. A. de Souza. Fast-TurboQuant: A multiplier-free online vector quantization approach. *arXiv preprint arXiv:2606.21448*, 2026.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 DAVIS challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pp. 8748–8763, 2021.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. OmniQuant: Omnidirectionally calibrated quantization for large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

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

Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. QuIP#: Even better LLM quantization with Hadamard incoherence and lattice codebooks. In *International Conference on Machine Learning (ICML)*, 2024.

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.

Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4): 600–612, 2004.

Xiuying Wei, Ruihao Gong, Yuhang Li, Xianglong Liu, and Fengwei Yu. QDrop: Randomly dropping quantization for extremely low-bit post-training quantization. In *International Conference on Learning Representations (ICLR)*, 2022.

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 (ICML)*, pp. 38087–38099, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, Da Yin, Xiaotao Gu, Yuxuan Zhang, Weihan Wang, Yean Cheng, Ting Liu, Bin Xu, Yuxiao Dong, and Jie Tang. CogVideoX: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations (ICLR)*, 2025.

Lijun Yu, José Lezama, Nitesh B. Gundavarapu, Luca Versari, Kihyuk Sohn, David Minnen, Yong Cheng, Vighnesh Birodkar, Agrim Gupta, Xiuye Gu, Alexander G. Hauptmann, Boqing Gong, Ming-Hsuan Yang, Irfan Essa, David A. Ross, and Lu Jiang. Language model beats diffusion: Tokenizer is key to visual generation. In *International Conference on Learning Representations (ICLR)*, 2024.

Amir Zandieh, Majid Daliri, Majid Hadian, and Vahab Mirrokni. TurboQuant: Online vector quantization with near-optimal distortion rate. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2504.19874.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

Yiming Zhao. Boundary-protection W8A8 HiFloat8 quantization for large-scale text-to-video diffusion transformers. In *IEEE International Conference on Multimedia and Expo (ICME) Grand Challenge*, 2026. arXiv:2606.00957.

Shuchang Zhou, Yuxin Wu, Zekun Ni, Xinyu Zhou, He Wen, and Yuheng Zou. DoReFa-Net: Training low bitwidth convolutional neural networks with low bitwidth gradients. *arXiv preprint arXiv:1606.06160*, 2016.

Lunjie Zhu, Yushi Huang, Xingtong Ge, Yufei Xue, Zhening Liu, Yumeng Zhang, Zehong Lin, and Jun Zhang. Flash-VAED: Plug-and-play VAE decoders for efficient video generation. In *International Conference on Machine Learning (ICML)*, 2026. arXiv:2602.19161.

Junhao Zhuang, Shi Guo, Xin Cai, Xiaohui Li, Yihao Liu, Chun Yuan, and Tianfan Xue. FlashVSR: Towards real-time diffusion-based streaming video super-resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. arXiv:2510.12747.

A Detailed Mathematical Proofs and Codebook Derivations

A.1 Proof of Invariance Under Orthogonal Rotation

In this subsection, we provide the formal step-by-step derivation of linear GEMM invariance under the channel-isolated orthogonal transformation $\mathbf{\Pi} \in \mathbb{R}^{C_{\text{in}} \times C_{\text{in}}}$, expanding on Proposition 1.

Let the unfolded convolutional weight matrix be $\mathbf{W} \in \mathbb{R}^{C_{\text{out}} \times d}$ and the unrolled patch activation matrix be $\mathbf{X} \in \mathbb{R}^{d \times N}$, where the contraction dimension is $d = K_t K_h K_w C_{\text{in}}$ and $N = T_{\text{out}} H_{\text{out}} W_{\text{out}}$. When the orthogonal rotation $\mathbf{\Pi}$ (satisfying $\mathbf{\Pi}^\top \mathbf{\Pi} = \mathbf{I}_{C_{\text{in}}}$) is applied along the input channel dimension C_{in} identically across all $K = K_t K_h K_w$ spatial-temporal kernel positions, the full block-diagonal transformation matrix $\mathbf{Q} \in \mathbb{R}^{d \times d}$ is given by the Kronecker product:

$$\mathbf{Q} = \mathbf{I}_K \otimes \mathbf{\Pi}. \tag{6}$$

Since $\mathbf{I}_K$ and $\mathbf{\Pi}$ are orthogonal, their Kronecker product $\mathbf{Q}$ is strictly orthogonal:

$$\mathbf{Q}^\top \mathbf{Q} = (\mathbf{I}_K \otimes \mathbf{\Pi})^\top (\mathbf{I}_K \otimes \mathbf{\Pi}) = (\mathbf{I}_K^\top \mathbf{I}_K) \otimes (\mathbf{\Pi}^\top \mathbf{\Pi}) = \mathbf{I}_K \otimes \mathbf{I}_{C_{\text{in}}} = \mathbf{I}_d. \tag{7}$$

The rotated weight matrix $\tilde{\mathbf{W}}$ and rotated activation matrix $\tilde{\mathbf{X}}$ are defined as:

$$\tilde{\mathbf{W}} = \mathbf{W}\mathbf{Q}^\top = \mathbf{W}(\mathbf{I}_K \otimes \mathbf{\Pi}^\top), \quad \tilde{\mathbf{X}} = \mathbf{Q}\mathbf{X} = (\mathbf{I}_K \otimes \mathbf{\Pi})\mathbf{X}. \tag{8}$$

Computing the unrolled matrix product yields:

$$\tilde{\mathbf{W}}\tilde{\mathbf{X}} = \left(\mathbf{W}\mathbf{Q}^\top\right)(\mathbf{Q}\mathbf{X}) \tag{9}$$

$$= \mathbf{W}\left(\mathbf{Q}^\top \mathbf{Q}\right)\mathbf{X} \tag{10}$$

$$= \mathbf{W}\mathbf{I}_d\mathbf{X} = \mathbf{W}\mathbf{X} = \mathbf{Y}. \tag{11}$$

Because $\mathbf{Q} = \mathbf{I}_K \otimes \mathbf{\Pi}$ is block-diagonal with respect to spatial-temporal kernel taps, each column of $\tilde{\mathbf{X}}$ corresponds to stacking independently rotated channel vectors:

$$\tilde{\mathbf{x}}(t+\tau, h+\delta_h, w+\delta_w) = \mathbf{\Pi}\mathbf{x}(t+\tau, h+\delta_h, w+\delta_w). \tag{12}$$

Thus, each spatial-temporal activation voxel $\mathbf{x}(t, h, w)$ is transformed by $\mathbf{\Pi}$ strictly once prior to patch unrolling (*im2col*), requiring zero redundant re-rotations across sliding windows.

A.2 Outlier Dispersion via Multiplier-Free R-FWHT

In this subsection, we formalize the sub-Gaussian tail bounds and peak outlier magnitude bound asserted in Theorem 1.

Let $\mathbf{x} \in \mathbb{R}^m$ be an arbitrary feature vector with L_2 norm $\|\mathbf{x}\|_2$, where $m = 2^{\lceil \log_2 C_{\text{in}} \rceil}$. The R-FWHT operator is $\mathbf{\Pi} = \frac{1}{\sqrt{m}}\mathbf{H}_m\mathbf{D}_\pm$, where $\mathbf{D}_\pm = \text{diag}(\sigma_1, \ldots, \sigma_m)$ with i.i.d. Rademacher variables $\sigma_j \in \{-1, +1\}$. For any coordinate $i \in \{1, \ldots, m\}$, the rotated coordinate is:

$$\tilde{x}_i = (\mathbf{\Pi}\mathbf{x})_i = \frac{1}{\sqrt{m}} \sum_{j=1}^{m} H_{i,j}\sigma_j x_j. \tag{13}$$

Notice that the Walsh-Hadamard matrix elements satisfy $H_{i,j} \in \{-1, +1\}$. Let $Z_j = H_{i,j}\sigma_j x_j$. Since σ_j are independent symmetric Rademacher random variables, each Z_j is a mean-zero independent random variable bounded by $|Z_j| \leq |x_j|$. By Hoeffding's Lemma, each Z_j has moment-generating function:

$$\mathbb{E}\left[\exp(\lambda Z_j)\right] \leq \exp\left(\frac{\lambda^2 x_j^2}{2}\right), \quad \forall \lambda \in \mathbb{R}. \tag{14}$$

Summing over all $j \in \{1, \ldots, m\}$ and scaling by $1/\sqrt{m}$, the moment-generating function of $\tilde{x}_i$ satisfies:

$$\mathbb{E}\left[\exp(\lambda \tilde{x}_i)\right] = \prod_{j=1}^{m} \mathbb{E}\left[\exp\left(\frac{\lambda}{\sqrt{m}} Z_j\right)\right] \leq \exp\left(\frac{\lambda^2 \sum_{j=1}^{m} x_j^2}{2m}\right) = \exp\left(\frac{\lambda^2 \|\mathbf{x}\|_2^2}{2m}\right). \tag{15}$$

Thus, $\tilde{x}_i$ is strictly sub-Gaussian with sub-Gaussian parameter $\sigma_{\text{sub}}^2 = \|\mathbf{x}\|_2^2 / m$. Applying the standard Chernoff-Hoeffding tail bound yields:

$$\mathbb{P}\left(|\tilde{x}_i| \geq \frac{t}{\sqrt{m}} \|\mathbf{x}\|_2\right) \leq 2\exp\left(-\frac{t^2}{2}\right), \quad \forall t > 0. \tag{16}$$

To bound the maximum peak coordinate $\|\tilde{\mathbf{x}}\|_\infty = \max_{1\leq i\leq m} |\tilde{x}_i|$, we apply a union bound over all m coordinates:

$$\mathbb{P}\left(\|\tilde{\mathbf{x}}\|_\infty \geq \frac{t}{\sqrt{m}} \|\mathbf{x}\|_2\right) \leq 2m\exp\left(-\frac{t^2}{2}\right). \tag{17}$$

Setting the failure probability to $\delta \in (0,1)$ via $2m\exp(-t^2/2) = \delta$, we solve for $t = \sqrt{2\ln(2m/\delta)}$. Substituting t back yields:

$$\|\tilde{\mathbf{x}}\|_\infty \leq \sqrt{\frac{2\ln(2m/\delta)}{m}} \|\mathbf{x}\|_2, \tag{18}$$

holding with probability at least $1-\delta$. For typical channel dimensions $m = 512$ or $m = 1024$, $\sqrt{2\ln(2m/\delta)/m} \ll 1$, demonstrating that extreme coordinate spikes are thoroughly dispersed into sub-Gaussian isotropic noise.

A.3 Distortion Bounds for Spherical Quantization

In this subsection, we formalize the Mean Squared Error distortion bound for unit-norm directional vectors $\mathbf{u} \in \mathbb{S}^{m-1}$ under optimal spherical Lloyd-Max codebook quantization, supporting Proposition 2.

Let $\mathbf{u}$ be distributed uniformly over the $(m-1)$-dimensional unit sphere $\mathbb{S}^{m-1}$, which arises from normalizing an isotropic Gaussian vector $\tilde{\mathbf{v}} \sim \mathcal{N}(0, \sigma^2\mathbf{I}_m)$ obtained after channel-isolated R-FWHT rotation. For a codebook $\mathcal{C} = \{\mathbf{c}_1, \ldots, \mathbf{c}_M\}$ of size $M = 2^b$, the nearest-neighbor spherical quantization rule maps $\mathbf{u}$ to:

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{c}\in\mathcal{C}} \|\mathbf{u} - \mathbf{c}\|_2^2 = \arg\min_{\mathbf{c}\in\mathcal{C}} (2 - 2\langle \mathbf{u}, \mathbf{c}\rangle) = \arg\max_{\mathbf{c}\in\mathcal{C}} \langle \mathbf{u}, \mathbf{c}\rangle. \tag{19}$$

According to asymptotic quantization theory and spherical cap covering bounds (Zandieh et al., 2026; Lloyd, 1982), the expected Euclidean distortion under an optimal Voronoi tessellation on $\mathbb{S}^{m-1}$ satisfies:

$$\mathbb{E}_{\mathbf{u}\sim\mathcal{U}(\mathbb{S}^{m-1})}\left[\min_{\mathbf{c}\in\mathcal{C}} \|\mathbf{u} - \mathbf{c}\|_2^2\right] \leq C_m \cdot M^{-2/m} = C_m \cdot 2^{-2b/m}, \tag{20}$$

where $C_m = \frac{m}{m+2} \cdot \left(\frac{\Gamma(m/2+1)^{1/m}}{\sqrt{\pi}}\right)^2 \approx \frac{1}{2\pi e}$ is the normalized moment of inertia of the optimal spherical Voronoi cell. For coordinate-wise scalar Lloyd-Max quantization applied to standard Gaussian components, the aggregate expected distortion over all m dimensions simplifies to:

$$\mathbb{E}\left[\|\mathbf{u} - \hat{\mathbf{u}}\|_2^2\right] \leq C_b \cdot 2^{-2b}, \tag{21}$$

where C_b depends only on the Lloyd-Max centroid placement and remains bounded independently of dimension m. In contrast, uniform scalar quantization over $[-M, M]^m$ introduces boundary clipping errors whose probability mass scales exponentially with dimension m, proving the fundamental rate-distortion superiority of spherical codebooks over Cartesian grids.

B Edge Hardware Micro-Architectures and Implementation Details

B.1 Option 1: Micro-Architectural Details of Byte-Packed 256-Entry Multiplication LUT

In this subsection, we detail the micro-architectural implementation and memory hierarchy layout for the Byte-Packed Lookup Table described in Section 5.

Address Bit-Packing and Memory Footprint. For W4A4 precision, activation index $i_A \in \{0, \dots, 15\}$ and weight index $i_W \in \{0, \dots, 15\}$ are combined into an 8-bit byte address:

$$\text{addr} = (i_A \ll 4) \vee i_W \in [0, 255]. \tag{22}$$

The precomputed multiplication table $\mathcal{T}_{\text{LUT}} \in \mathbb{R}^{16\times16}$ maps each byte directly to the pre-multiplied floating-point product $\mathcal{C}_A[i_A] \times \mathcal{C}_W[i_W]$. Under FP16 precision, the complete table requires exactly:

$$256 \text{ entries} \times 2 \text{ bytes} = 512 \text{ bytes}. \tag{23}$$

Under FP32 precision, the table requires 1024 bytes (1 KB). Because this footprint is minuscule, $\mathcal{T}_{\text{LUT}}$ can be pinned permanently within tightly coupled L1 vector scratchpads (e.g., ARM TCM, DSP Tightly Coupled Data Memory) or vector register files, incurring zero DRAM latency.

Parallel Vector Gather and SRAM Bank Interleaving. To prevent memory serialization during vector operations, edge hardware can exploit parallel byte-shuffle or vector-table-lookup instructions:

- **Vector Table Lookup Units**: Implemented using parallel table lookup instructions, which perform simultaneous byte lookups in a single cycle.
- **Vector Permute Units**: Implemented using 8-bit byte permute instructions, executing parallel table accesses per vector lane.
- **Multi-Banked SRAM Acceleration**: In dedicated edge accelerators, $\mathcal{T}_{\text{LUT}}$ is replicated across 16 or 32 independent SRAM banks, allowing all parallel systolic PEs to read pre-multiplied centroids simultaneously without bank conflict stalls.

B.2 Option 2: Dynamic Sub-Vector Integer Uniform Quantization ($g = 32$)

For commodity processors lacking specialized parallel LUT gather units, TurboQuantConv provides an integer-native backend: mapping R-FWHT Gaussianized coordinates onto symmetric uniform signed 4-bit integer lattices $q_A, q_W \in \{-7, \dots, +7\}$:

$$\hat{y} = \sum_{g} \left(s_W^{(g)} \cdot s_X^{(g)}(t,h,w) \right) \sum_{k=1}^{g} q_{W,k} \cdot q_{A,k}. \tag{24}$$

Here, the inner product within each sub-vector group g factors into pure integer multiply-accumulate (INT-MAC) operations, with localized dynamic activation scaling $s_X^{(g)}(t,h,w) = \max_{c\in g} |\tilde{\boldsymbol{\mathcal{X}}}_{c,t,h,w}|/q_{\max}$ and offline weight scaling $s_W^{(g)}$ applied once per sub-vector epilogue.

Channel Sub-Vector Scaling ($g = 32$). Because uniform lattices enforce a constant step size across the dynamic range, they incur clipping degradation under coarse full-channel scaling ($g =$ Full, where a single global scale spans all channels). We eliminate this penalty by partitioning rotated channels into compact **sub-vectors** of group size $g = 32$, each assigned independent localized scale factors $s_W^{(g)}, s_X^{(g)} \in \mathbb{R}^+$. Structuring sub-vectors with $g = 32$ provides direct hardware alignment:

- **128-bit Vector Register Alignment ($g = 32$)**: 32 elements at 4-bit precision occupy exactly 32×4 bits $= 128$ bits $= 16$ bytes. A complete sub-vector of 32 weights or activations fills an entire 128-bit vector register without padding or fragmentation.
- **Integer Dot-Product Execution**: Modern edge accelerators provide native 4-element integer dot-product instructions, computing $\sum_{i=1}^{4} a_i w_i + c$ in a single cycle. A 32-element sub-vector inner product executes in consecutive operations, accumulating into a 32-bit integer register. The localized scale product $s_W^{(g)} \cdot s_X^{(g)}$ is applied once per 32-element burst, keeping metadata overhead at $\sim 12.5\%$.

Empirical Evaluation of Integer Uniform Sub-Vector Scaling. Detailed empirical results comparing TurboQuantConv Option 1 (Gaussian Codebook) against Option 2 (Uniform Sub-Vector $g = 32$) and polar symmetric ($[-7, +7]$) full-channel uniform quantization under W4A4 precision are reported in Table 4 in Section 5. Specifically, at full-channel granularity ($g =$ Full), polar symmetric $[-7, +7]$ uniform quantization retains Eq. (5)'s per-voxel L_2 radius $r = \|\tilde{\mathbf{x}}\|_2$ and unit direction $\mathbf{u} = \tilde{\mathbf{x}}/r$, but replaces the non-uniform Lloyd-Max codebook with a zero-centered 15-level symmetric integer grid $q \in \{-7, \dots, +7\}$ (omitting -8 for exact zero symmetry) using a dynamic per-voxel

(per-token) absmax step size $s_u = \max_c |u_c|/7$ across all C_{in} rotated coordinates (reconstructing via effective per-voxel scale $s = rs_u = \max_c |\tilde{x}_c|/7$ for activations and per-output-channel for weights). As demonstrated in Table 4, replacing QuaRot-style per-tensor min-max RTN (29.49 dB in Table 3c) with TurboQuantConv's per-voxel (per-token) polar symmetric ($[-7, +7]$) absmax scaling at full-channel granularity ($g =$ Full) reaches 32.82 dB, yet still drops PSNR by 0.59 dB relative to Option 1 (33.41 dB) due to step-size saturation across wide channels. Sub-vector scaling with $g = 32$ eliminates this penalty entirely, boosting W4A4 PSNR by $+2.71$ dB (to $\mathbf{35.53}$ dB), SSIM to $\mathbf{0.9498}$, and LPIPS to $\mathbf{0.0025}$, outperforming full-channel Option 1 by $+2.12$ dB while running natively on integer MAC datapaths.

Unified Framework Rationale: Synergistic Backends under Channel-Isolated R-FWHT. A natural architectural question is: *how do Option 1 (Gaussian codebook) and Option 2 (sub-vector uniform $g = 32$) complement each other within the TurboQuantConv framework?* Crucially, both backends rely on the exact same **channel-isolated R-FWHT incoherence rotation** derived in Section 3 to suppress spatiotemporal activation bursts without sliding-window re-rotation overhead. Rather than representing competing external paradigms, Option 1 and Option 2 form a deliberate multi-objective Pareto frontier across metadata overhead, calibration requirements, and silicon hardware:

1. **Fair Granularity Comparison ($g =$ Full)**: Under identical full-channel granularity and minimal metadata ($< 0.3\%$), Option 1 (Gaussian Lloyd-Max codebook) is mathematically rate-distortion optimal on rotated Gaussians. It consistently outperforms both QuaRot-style per-tensor min-max RTN by $+\mathbf{3.92}$ dB on TokenBench (Table 3, 33.41 vs. 29.49 dB) and TurboQuantConv's own per-voxel (per-token) polar symmetric $[-7, +7]$ uniform grid ($g =$ Full) by $+\mathbf{0.59}$ dB (Table 4, 33.41 vs. 32.82 dB).
2. **Metadata vs. Accuracy Trade-Off**: Option 1 operates with virtually zero metadata overhead ($< 0.3\%$, storing 1 scale per full channel/vector). In contrast, Option 2 allocates $\sim 12.5\%$ metadata overhead to store localized scale factors per 32 channels. Combining R-FWHT outlier dispersion with localized sub-vector scaling absorbs residual variance variations across channel slices, yielding $+2.12$ dB higher accuracy over full-channel Option 1.
3. **Silicon Execution Datapath**: Option 1 achieves completely multiplier-free execution via a 512-byte LUT pinned in L1 SRAM scratchpads, ideal for custom ultra-low-power edge NPUs and ASICs. In contrast, Option 2 targets commodity processors (NVIDIA Tensor Cores, ARM NEON, TPU MXUs) lacking dedicated LUT gather units but providing massive integer MAC throughput.
4. **Dynamic Quantization vs. Calibration Overfitting**: In sub-vector scaling ($g = 32$), each 32-element activation slice represents a localized feature trajectory whose dynamic range fluctuates across spatial coordinates and temporal video frames. Attempting to statically calibrate activation sub-vector scales offline leads to severe overfitting to the calibration sequences, causing severe clipping on out-of-distribution evaluation videos. Consequently, Option 2 strictly requires online dynamic quantization for activations, extracting localized scales $s_X^{(g)}$ dynamically per forward pass. While modern commodity GPUs and systolic arrays readily absorb dynamic scale extraction via wide vector SIMD, Option 1's full-channel Gaussian codebook operates on globally isotropic features with static, calibration-free execution, avoiding online dynamic scale reductions entirely on resource-constrained edge NPUs.

Thus, TurboQuantConv unifies both backends under channel-isolated R-FWHT incoherence processing, providing tailored optimality across custom LUT NPUs and commodity systolic arrays.

C Supplementary Per-Scene Evaluation Breakdowns and Visual Gallery

To complement the aggregate benchmarks reported in Table 1 and Table 2, detailed per-scene evaluation logs and video generation clips are provided in this section and the supplementary archive. In Figures 3–6, we provide extended qualitative visual comparisons across four diverse benchmark scenes under W4A4 precision:

- **Surfing Cat on Dynamic Waves (Figure 3)**: Evaluates fluid-body interaction and turbulent spray dynamics. While SmoothQuant suffers severe chromatic collapse and SVDQuant causes artificial motion freeze, TurboQuantConv faithfully preserves dynamic wave curls, realistic spray particulates, and natural fur movement.
- **Bipedal Humanoid Robot (Figure 4)**: Evaluates complex rigid-body articulation and ancient architectural stone textures. Baselines introduce heavy blurring along mechanical linkages and glowing halo artifacts around chest emitters, whereas TurboQuantConv maintains crisp metallic edges and solid structural geometry.
- **Neon Liquid Dynamics (Figure 5)**: Evaluates non-rigid multi-phase fluid blending, glowing droplet dispersion, and saturated color gradients. Prior methods suffer severe color banding and chromatic washout, while TurboQuantConv faithfully reproduces smooth fluid contours and vibrant color transitions.
- **Cyberpunk Sports Car (Figure 6)**: Evaluates high-speed motion blur, tire smoke volume dispersion, and specular vehicle reflections. TurboQuantConv resolves clean vehicle silhouettes and natural smoke dissipation without the motion ghosting and tearing observed in prior baselines.

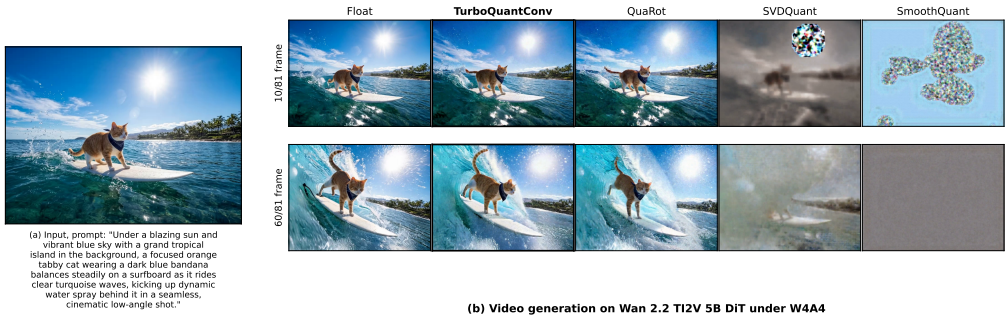

Figure 3: **Qualitative comparison on Wan 2.2 TI2V 5B DiT under W4A4: Surfing Cat on Dynamic Waves**. (a) Input conditioning frame with text prompt. (b) Video generation under W4A4 at Frames 10 and 60.

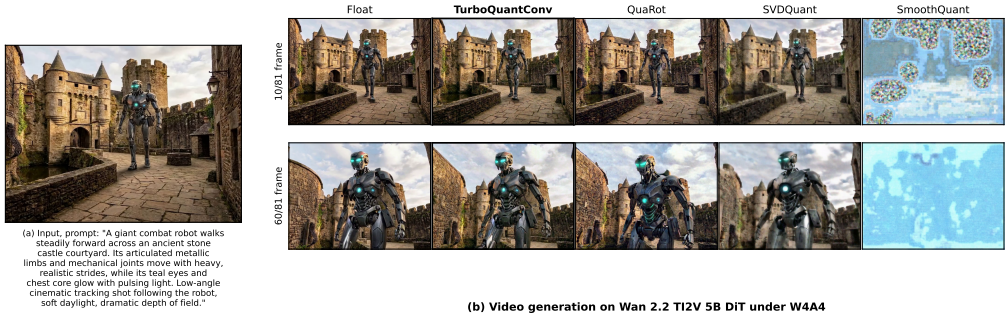

Figure 4: **Qualitative comparison on Wan 2.2 TI2V 5B DiT under W4A4: Bipedal Humanoid Robot**. (a) Input conditioning frame with text prompt. (b) Video generation under W4A4 at Frames 10 and 60.

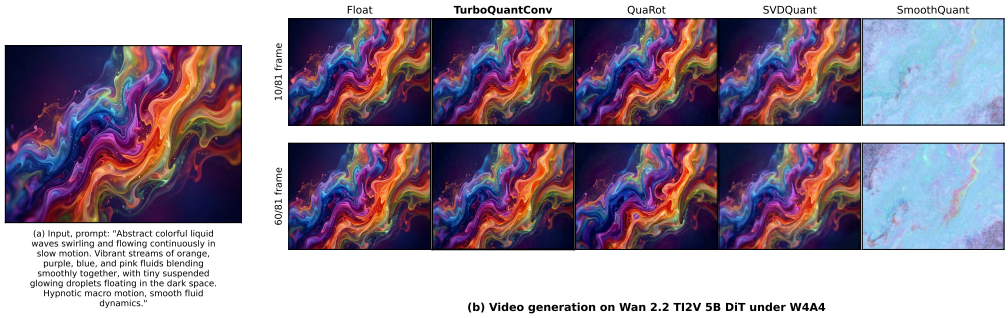

Figure 5: **Qualitative comparison on Wan 2.2 TI2V 5B DiT under W4A4: Neon Liquid Dynamics**. (a) Input conditioning frame with text prompt. (b) Video generation under W4A4 at Frames 10 and 60.

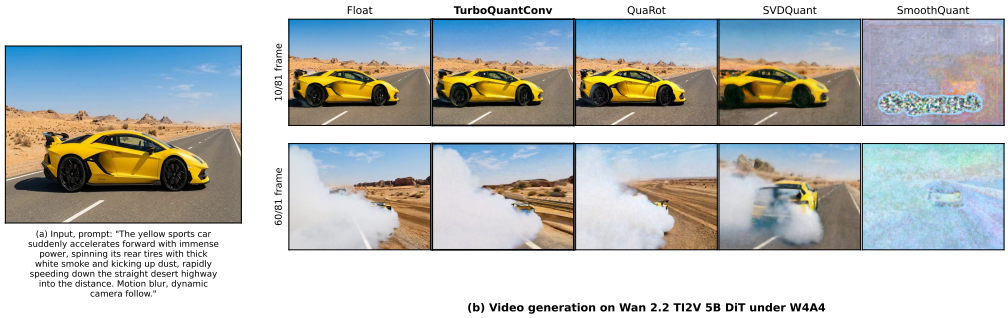

Figure 6: **Qualitative comparison on Wan 2.2 TI2V 5B DiT under W4A4: Cyberpunk Sports Car**. (a) Input conditioning frame with text prompt. (b) Video generation under W4A4 at Frames 10 and 60.