SemanticDialect: Semantic-Aware Mixed-Format Quantization for Video Diffusion Transformers

Wonsuk Jang
Department of Electrical Engineering
Stanford University
wsjang@stanford.edu

Thierry Tambe
Department of Electrical Engineering
Stanford University
ttambe@stanford.edu

Abstract

Diffusion Transformers (DiTs) achieve state-of-the-art video generation, but their substantial memory and computational footprints hinder edge deployment. Quantization can reduce these costs, yet existing methods often degrade video quality due to high activation variation and the difficulty of preserving semantic and temporal coherence. We propose **SemanticDialect**, advancing block-wise mixed-format quantization. In this framework, each block selects an optimal format (*dialect*) from a candidate set (*formatbook*), which is augmented with lookup tables that store quantization errors and quantized indices, enabling efficient per-block format selection and quantization with minimal online overhead. We further introduce attention-guided *activation decomposition*, which reduces quantization error via residual quantization, and *semantic-aware dialect assignment (SeDA)*, which reduces cross-token quantization inconsistency by enforcing format uniformity among semantically correlated tokens. Experiments on multiple video diffusion models, including Open-Sora 1.0 and 2.0, show that SemanticDialect outperforms prior quantization methods and block-wise formats such as MXFP4 and NVFP4. We also validate hardware deployability via RTL design and GPU kernel implementation. Code is available at https://github.com/Tambe-Lab/SemanticDialect.



Figure 1: SemanticDialect is a post-training quantization method for video diffusion transformers. It quantizes Open-Sora 2.0 to 4-bit weights and activations while maintaining near-FP16 visual quality.

1 Introduction

Diffusion Transformers (DiTs) [Peebles and Xie, 2023] have gained prominence in video generation [Ma et al., 2024, Yang et al., 2024, Peng et al., 2025], as their scalability and capacity to capture long-range spatiotemporal context, and are increasingly deployed as a backbone of many applications such as Physical AI [Ye et al., 2026]. However, their large parameter counts and necessity of iterative

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

denoising (often tens of steps) substantially increase compute and memory demands, exacerbated by the long sequence lengths of multi-frame video—making edge deployment challenging.

Quantization mitigates these costs by reducing memory footprints and data movement, while enabling efficient low-bit arithmetic for enhanced throughput and energy efficiency [Kim et al., 2023, Xiao et al., 2023, Rouhani et al., 2023a, Cao et al., 2024]. However, prior DiT quantization methods [Wu et al., 2024, Chen et al., 2025a] often struggle to preserve text-to-video generation quality [Zhao et al., 2024]. This degradation stems from two key challenges: (i) high activation variation across tokens and timesteps, where a few large-magnitude outliers dominate the scaling factor, reducing effective quantization precision for the majority of elements [Liu et al., 2023], and (ii) strong spatiotemporal correlations across frames, which standard MSE-based quantization objectives often fail to capture.

In response to the first challenge, fine-grained block-wise quantization has become widely adopted [Dai et al., 2021, Rouhani et al., 2023a, Hu et al., 2026, Lo et al., 2024], assigning separate scaling factors to small blocks to localize outlier impact. This trend is reflected in industry efforts such as the OCP Microscaling (MX) specification, which standardizes block-wise formats using power-of-two scales [Open Compute Project, 2023]. Recent accelerators from NVIDIA and AMD have also embraced MX formats [NVIDIA, 2025, AMD, 2025]. To further improve accuracy, higher-precision block-scale variants (e.g., NVFP4) have been introduced, albeit at the expense of increased metadata and runtime overhead for scale normalization and rescaling. Despite these advances, a single low-precision format still struggles to match the diverse per-block value distributions.

Motivated by this, prior work proposes a mixed-format quantization that selects an optimal per-block format from a predefined set (the *formatbook*) to better match block value distributions [Jang and Tambe, 2025]. However, extending this strategy to VDiTs is non-trivial for three reasons: (i) VDiT activations exhibit higher variability, calling for a larger, more adaptive formatbook; (ii) on-the-fly format selection and quantization impose substantial overhead, as the complexity of decision logic grows with the number of representable values across all candidate formats; and (iii) independent block quantization ignores the semantic and temporal correlations essential for video consistency.

To tackle these challenges, we propose **SemanticDialect**, a VDiT post-training quantization (PTQ) framework built on the **FB4** format, which enables calibration-free block-wise mixed-format 4-bit quantization. To handle high activation variability, FB4 employs a 32-entry formatbook with lookup table (LUT)-based online format selection and quantization. To mitigate quality loss in quantization-sensitive layers, we introduce attention-guided *activation decomposition*, which reduces quantization error by re-quantizing the residual and adding it back, without conventional mixed-precision overhead. Finally, to avoid block-wise over-specialization—where inconsistent quantization of identical activation values across blocks undermines spatiotemporal consistency—we propose *semantic-aware dialect assignment (SeDA)*, which encourages semantically aligned tokens to share a subset of the formatbook. Our contributions are as follows:

- We introduce FB4, a 4-bit number format enabling calibration-free, online mixed-format quantization via per-block LUT-based format selection from a 32-entry formatbook.
- We propose SemanticDialect, a VDiT quantization framework that integrates FB4 with activation decomposition for sensitive layers and SeDA for spatiotemporal consistency.
- We demonstrate that SemanticDialect outperforms existing VDiT quantization methods and block-wise format baselines on Open-Sora 1.0 and 2.0, two distinct VDiT architectures.

2 Related Work

2.1 Video Diffusion Transformers (VDiT)

DiTs [Peebles and Xie, 2023] replace traditional convolutional U-Net backbones with Transformer blocks, serving as a scalable foundation for modern video generation. Similar to prior diffusion models, DiTs iteratively denoise latents over multiple timesteps, with Transformers better capturing global context over large token sequences. VDiTs commonly exhibit two architectural patterns: (i) injecting external conditions (e.g., text and timestep) via cross-attention or adaptive LayerNorm (adaLN) modulation, and (ii) spatiotemporal modeling via either factorized attention that decouples spatial and temporal interactions [Ma et al., 2024, Zheng et al., 2024] or full 3D attention [Peng et al., 2025, Yang et al., 2024, Kong et al., 2024] that jointly attends across space–time tokens paired

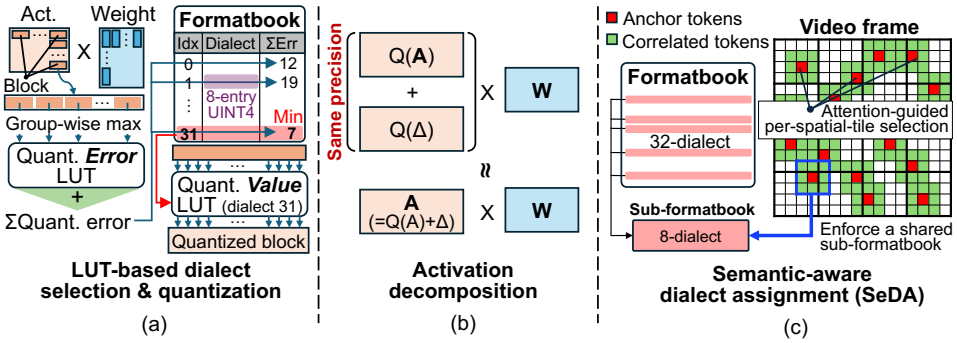


Figure 2: SemanticDialect overview. (a) LUT-based format selection and quantization. (b) Activation decomposition for quantization-sensitive layers. (c) SeDA for consistent block-wise quantization.

with 3D rotary positional embeddings (3D RoPE). To enhance text-to-video alignment, many VDiT models use classifier-free guidance (CFG), combining the outputs of a conditional branch (with condition tokens) and an unconditional branch (with null tokens).

2.2 VDiT Quantization

A growing body of work explores post-training quantization for DiTs. Q-DiT and PTQ4DiT [Chen et al., 2025a, Wu et al., 2024] aim to stabilize activation quantization across channels and timesteps, while SVDQuant [Li et al., 2024] reduces error by incorporating a lightweight SVD-based low-rank compensation branch. However, these designs do not fully address the pronounced spatiotemporal variation in VDiTs, nor do they optimize for video-specific quality metrics beyond MSE. Recent VDiT-focused approaches improve quality through video-aware heuristics, including metric-aware mixed precision (ViDiT-Q [Zhao et al., 2024]), inter-frame distribution distillation to better preserve temporal behavior (Q-VDiT [Feng et al., 2025b]), and specialized calibration-data selection (S^2Q-VDiT [Feng et al., 2025a]). In contrast, SemanticDialect targets the root cause—diverse block-level distributions across layers and timesteps—through online, per-block format selection to match local statistics without intensive calibration. We also introduce uniform-precision activation decomposition for quantization-sensitive layers without mixed-precision activations, and semantic-aware dialect assignment (SeDA) to preserve spatiotemporal consistency of quantized values.

2.3 Fine-Grained Block-Wise Mixed-Format Quantization

Fine-grained block-wise quantization has become widely adopted [Dai et al., 2021, Rouhani et al., 2023a, Hu et al., 2026, Lo et al., 2024, Rouhani et al., 2023b] by assigning separate scaling factors to small blocks, effectively localizing outlier impact. However, mapping normalized values to a fixed low-precision format (e.g., FP4) can still incur large error due to limited representational capacity. Mixed-format methods attempt to improve value representation by selecting among multiple candidates: RaZeR [Chen et al., 2025b] repurposes redundant ± 0 encodings in NVFP4 into predefined special values, and BlockDialect [Jang and Tambe, 2025] selects a per-block format from a predefined set (a *formatbook*) while retaining power-of-two scaling. However, their expressiveness is often limited by predefined special values or small formatbooks, and scaling to a larger set of candidates is challenging because they require costly repeated quantization or comparison logic over individual representable values. We address this by scaling to a larger, more expressive formatbook with low online overhead via LUT-based format selection and quantization. We use the term *dialect* to refer to each same-precision candidate with slightly different representable values in the formatbook.

3 SemanticDialect: Semantic-Aware Mixed-Format Quantization for Video Diffusion Transformers

Figure 2 summarizes our approach. We address four challenges to enable quality-preserving, fine-grained block-wise mixed-format quantization for VDiTs: (1) designing an expressive formatbook;

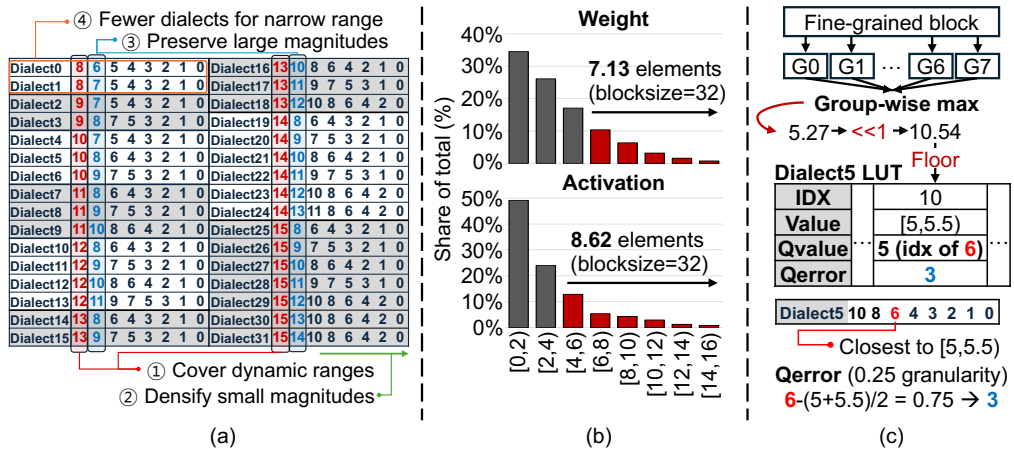


Figure 3: Proposed formatbook and lookup table. (a) Four formatbook construction ground rules. (b) Block-wise normalized value distributions. (c) Lookup table details and data-to-index mapping.

(2) enabling efficient online per-block dialect selection and quantization; (3) mitigating errors in quantization-sensitive layers; and (4) ensuring semantic awareness across spatial and temporal axes.

3.1 Constructing a Highly Expressive Formatbook

Block Normalization and 4-bit Representation. For each block, we compute a shared power-of-two scale by applying $\lfloor \log_2(\cdot) \rfloor$ to the block's maximum magnitude. We then normalize each block so that its values lie in $[0, 2)$, with the maximum in $[1, 2)$. Elements are represented by a 1-bit sign and a 3-bit index (i.e., 4-bit quantization) that maps to one of eight representable magnitudes in the dialect. These magnitudes are selected from 4-bit unsigned integers (0-15), enabling hardware-efficient low-precision multiply–accumulate (MAC). For seamless implementation, we offset the shared exponent by 3 (equivalently, scale the normalized values by 2^3), mapping $[0, 2)$ to $[0, 16)$ (with the maximum in $[8, 16)$) so magnitudes can be directly encoded as 4-bit unsigned integers.

Formatbook Design. Since the 16-dialect formatbook in prior work [Jang and Tambe, 2025] cannot fully capture the higher variation in VDiT weights and activations (see Table 1), we build a larger *32-dialect* formatbook (Fig. 3a) based on four design rules. (1) *Cover all dynamic ranges:* since power-of-two scaling discretizes range, insufficient coverage can waste dynamic range or leave values underrepresented [Jang and Tambe, 2025, Lo et al., 2024]. (2) *Densify small magnitudes:* Open-Sora 2.0 profiling (Fig. 3b) shows most values cluster near zero, so we allocate more points to small magnitudes. (3) *Preserve large magnitudes:* large-magnitude values often dominate MAC outputs yet suffer high quantization error under small-magnitude–biased dialects due to sparse high-range coverage [Cook et al., 2025]; thus, each dynamic range includes dialects that better represent large magnitudes. (4) *Use fewer dialects for narrower ranges:* this reduces dialect ID (DID) metadata and simplifies per-block dialect selection. The total per-block metadata overhead is 10 bits (5-bit DID and 5-bit shared exponent). Although this rule-based formatbook may not be globally optimal, our goal is to demonstrate scalable large-formatbook mixed-format quantization rather than exhaustively optimizing the formatbook. Appendix B confirms that all dialects are meaningfully leveraged.

3.2 LUT-Based Per-Block Dialect Selection and Quantization

Group-wise Maximum Extraction. A straightforward dialect selection method computes the MSE for each of the 32 dialects across all block elements (e.g., 32-element) and picks the minimum. However, this full comparison is costly across numerous blocks. Profiling (Fig. 3b) on Open-Sora 2.0 shows that only a small fraction of elements have large magnitudes, yet they disproportionately contribute to MAC outputs. To balance accuracy and compute overhead, we restrict MSE evaluation to the top-k largest-magnitude values (e.g., $k = 8$). Since exact top-k requires sorting, we instead approximate by partitioning each block into k groups (*num_groups* $= k$) and taking the maximum from each group (Fig. 3c), enabling parallel group-wise computation.

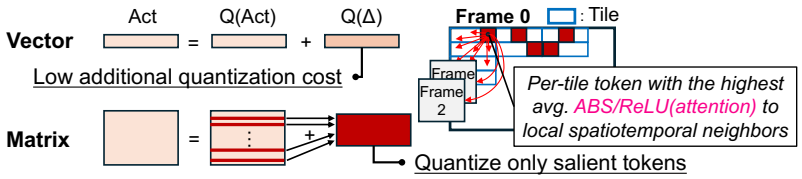


Figure 4: Activation decomposition for vector and matrix activations.

LUT-Based Quantization Error Approximation. Even with group-wise maximum extraction, evaluating quantization MSE remains compute-intensive. This is because our dialects span diverse distributions, so quantization cannot be reduced to a simple shift-and-round operation as in standard 4-bit formats (e.g., INT4, FP4). Instead, for each dialect we must (i) find the nearest representable value and (ii) accumulate its deviation from the input. To make this efficient for a large formatbook, we introduce two lookup tables per dialect, `Qvalue` and `Qerror` (Fig. 3c): `Qvalue` returns the 3-bit index of the nearest representable value, while `Qerror` provides an approximate quantization error.

Since non-integer inputs (e.g., FP16) cannot directly index a LUT, we apply a shift-and-truncate to map them to integer bin indices. Because representable values are integers (with decision boundaries at half-integers), `Qvalue` uses 0.5-wide bins so that each bin maps to the *exact* quantized index. For `Qerror`, we *approximate* absolute quantization error using the midpoint error of each 0.5-wide bin, rather than squared error, which can amplify midpoint estimation bias. Dialect selection then reduces to summing `Qerror` over group-wise maxima and choosing the dialect with minimum approximate error. Once selected, quantizing all elements proceeds efficiently via direct `Qvalue` queries.

Two-stage Dialect Selection. A remaining challenge is the cost of comparing all 32 dialects to select the best per-block dialect. Since matching the block's dynamic range is critical to avoid underestimating or wasting range, we adopt a two-stage approach similar to the approach in [Jang and Tambe, 2025]. In the first stage, we select a sub-formatbook based on the block maximum. In the second stage, we compute the approximate error and compare only the dialects within that selected subset. We refer to our 4-bit quantization format, which utilizes the formatbook with LUT-based dialect selection and quantization, as **FB4** (FormatBook 4-bit format) for the remainder of the paper.

3.3 Compensating Quantization-Sensitive Layers

Layer-wise Profiling Results. SemanticDialect primarily targets linear layers (`nn.Linear`), which perform activation–weight MACs and dominate inference latency [Zhao et al., 2024]. For each layer, we profile (i) *sensitivity* by quantizing only that layer in an otherwise FP16 model, and (ii) *recovery potential* by keeping only that layer in FP16 in an otherwise FB4-quantized model. Appendix A shows that *modulation layers* are the most quantization-sensitive in Open-Sora 2.0 (Flux-based [Labs et al., 2025]), likely because their inputs—single vectors encoding text and timestep—are already highly compressed. In Open-Sora 1.0 (STDiT-based [Zheng et al., 2024]), these layers are relatively small and typically kept in FP16 [Zhao et al., 2024, Feng et al., 2025b]. Beyond modulation, we find the *final MLP linear layer*—often sensitive since it directly precedes the layer output [Yan et al., 2026, Ashkboos et al., 2024a]—and the *temporal attention QKV projection* to also be highly sensitive.

Activation Decomposition. In diffusion transformers, activations are often harder to quantize than weights due to timestep-dependent, channel-wise outliers [Ding et al., 2025, Chen et al., 2025a]. In contrast, weights are timestep-invariant and can be quantized offline, allowing exhaustive dialect selection without runtime approximations. We therefore focus on improving *activation* quantization in sensitive layers. Mixed precision (i.e., using higher precision for sensitive layers) is a straightforward option, but it is undesirable: it complicates GPU kernels (multiple data types and conversions, fewer fusion opportunities), increases memory-management complexity, and requires extra datapaths or format-specific kernels—especially for non-standard formats such as FB4 or on dedicated hardware.

Instead, we propose *activation decomposition*. As shown in Figure 4, we quantize the activation, forming residual Δ where $Act = Q(Act) + \Delta$, then re-quantize Δ to obtain $Q(Act) + Q(\Delta)$ as a closer approximation to Act using the same low-precision format. Since decomposition applies only to activations, the parameter count remains unchanged and a linear layer is approximated as $Act \cdot W + b \approx \big(Q(Act) + Q(\Delta)\big)W + b$. The modulation layer in Open-Sora 2.0 is particularly suitable because its activation is a single *vector*, incurring negligible re-quantization overhead.

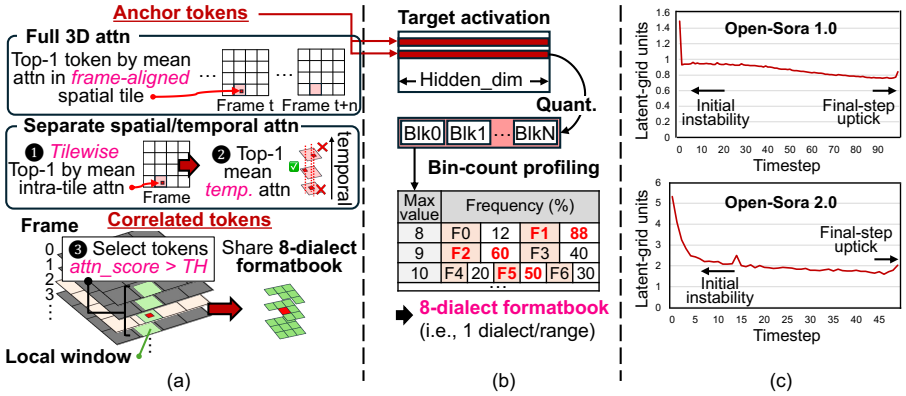


Figure 5: Semantic-Aware Dialect Assignment (SeDA). (a) Anchor/correlated token selection. (b) Sub-formatbook construction. (c) Average anchor-token displacement across denoising timesteps.

For *matrix* activations, decomposing all tokens is costly, substantially increasing the effective bitwidth (two 4-bit values per token, ≈ 8 bits plus metadata). We therefore apply decomposition only to one salient token per tile, where a tile is a local group of tokens. Concretely, we score candidates by their mean pre-softmax attention score ($Q \cdot K^{\mathsf{T}}$) with respect to a local spatiotemporal neighborhood and select the highest-scoring token in each tile. To prevent large negative scores from canceling meaningful correlations, we use ReLU for temporal attention to focus on positive attention connections across frames, while using ABS for spatial / 3D attention since both positive and negative responses can be informative (similarity and contrast). Furthermore, in some layers, allocating the salient token budget exclusively to the *conditional* CFG branch proves more effective than splitting it evenly across conditional and unconditional branches. Detailed evaluation is provided in Appendix H.

3.4 Incorporating Semantic Awareness Across Spatial and Temporal Axes

Block-Wise Over-Specialization. Although block-wise mixed-format quantization better matches local distributions, a large formatbook can introduce dialect inconsistency: the same value may be assigned different dialects across blocks due to minor shifts in block statistics. For example, the same spatial token across frames, or spatially adjacent tokens within a semantic region may be quantized inconsistently, potentially leading to visual artifacts. In short, fine-grained dialect matching can over-specialize to local blocks—optimizing the "trees" while harming the "forest."

Semantic-Aware Dialect Assignment (SeDA). To mitigate this over-specialization, we encourage semantically related tokens to use consistent dialect assignments. However, forcing a single shared dialect can degrade video quality because their blocks may have different dynamic ranges. Instead, we enforce consistency at the *sub-formatbook* level: related tokens share the same 8-dialect sub-formatbook, which provides one dialect per dynamic range (mapped to block maxima 8–15).

Constructing an 8-Dialect Sub-Formatbook. We mainly apply SeDA to post-attention linear layers (e.g., MLP), as they directly precede residual addition and heavily influence layer outputs. After quantizing the target layer, we perform bin-count profiling on anchor tokens to measure how often each dialect is selected in each dynamic range. We then construct an 8-dialect sub-formatbook by selecting the most frequent dialect for each dynamic-range bucket (Figure 5b), and apply it to constrain the semantically related tokens at the same timestep. We also evaluate simpler variants, including sharing a pre-calibrated formatbook across tokens, as discussed in Appendix I.4.

Attention-Guided Semantic Relatedness Extraction. Prior works use attention scores as a criterion for token similarity [Lu et al., 2025] and show that VDiT attention map exhibits strong spatiotemporal locality and encodes prompt-conditioned structure [Cai et al., 2025, Wen et al., 2025]. Following this line, we treat attention scores ($Q \cdot K^{T}$) as a *proxy* for semantic relatedness. Concretely, we identify *anchor tokens* and their *correlated tokens*—tokens that receive strong attention from the anchor—and constrain them to share the same sub-formatbook (Figure 5a).

Since selecting anchors by global top-k attention score is computationally expensive and suffers from poor spatial coverage (clustering in large homogeneous regions such as background), we propose a

Table 1: Text-to-video generation quality on VBench benchmark suite for Open-Sora 1.0 and 2.0.

Model	Method	Block Size	Eff. bit (A/W)	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
Open-Sora 1.0 factorized attn	FP16	-	16 / 16	58.00	62.03	96.41	54.17	91.51	96.43	36.17	26.14
	MXFP4	16	4.31 / 4.31	33.22	30.78	98.74	82.87	85.24	94.28	4.70	14.39
		32	4.16 / 4.16	34.69	37.80	98.59	84.72	85.99	94.53	5.69	15.00
	NVFP4	16	4.5 / 4.5	51.41	52.46	97.29	56.94	88.77	95.50	16.28	23.17
		32	4.25 / 4.25	46.82	47.26	97.90	50.00	88.80	95.61	13.42	20.88
	Block Dialect	16	4.56 / 4.56	49.03	47.65	98.02	58.80	88.25	95.12	15.72	21.97
		32	4.28 / 4.28	44.27	46.14	98.33	68.06	87.71	94.92	9.08	19.72
	ViDiT-Q Q-VDiT	A: Token W: Ch	4.81 / 4.73	30.71	62.73	82.43	45.37	91.61	95.08	0.39	8.33
				28.72	53.98	84.91	86.57	97.01	97.37	0.00	2.08
	FB4	16	4.63 / 4.63	51.83	54.84	97.63	42.59	90.99	96.21	25.73	23.80
		32	4.31 / 4.31	45.95	48.69	98.31	47.22	88.90	95.60	10.63	20.98
	Semantic Dialect	16	5.10 / 4.63	54.00	59.52	97.09	44.44	91.57	95.88	32.15	24.64
		32	4.76 / 4.31	51.09	57.04	97.40	51.39	89.48	95.22	20.93	22.94
Open-Sora 2.0 3D attn	FP16	-	16 / 16	57.35	63.26	98.86	62.50	94.79	97.98	48.76	27.98
	MXFP4	16	4.31 / 4.31	50.31	39.40	98.98	53.24	86.87	94.98	36.60	26.00
		32	4.16 / 4.16	51.11	41.84	98.99	54.17	87.26	95.15	39.97	26.25
	NVFP4	16	4.5 / 4.5	56.24	62.44	98.60	62.96	92.21	97.00	48.74	28.07
		32	4.25 / 4.25	55.74	60.33	98.55	65.28	91.22	96.43	46.49	27.97
	Block Dialect	16	4.56 / 4.56	56.83	62.22	98.72	59.72	93.09	97.40	48.04	28.12
		32	4.28 / 4.28	56.41	61.00	98.70	61.57	92.53	96.89	47.55	28.08
	ViDiT-Q	A: Token W: Ch	4.82 / 4.40	54.15	58.01	98.74	52.31	89.43	94.81	41.33	27.30
	FB4	16	4.63 / 4.63	56.82	63.12	98.66	61.57	93.24	97.29	48.35	28.13
		32	4.31 / 4.31	56.48	61.66	98.66	64.81	92.62	96.86	48.21	28.07
	Semantic Dialect	16	4.63 / 4.63	56.50	63.75	98.59	67.59	93.02	96.90	49.54	28.14
		32	4.31 / 4.31	56.55	63.24	98.51	66.20	92.34	96.31	48.57	28.27

localized selection strategy. For *factorized attention*, ❶ we partition each frame into spatial tiles and identify one anchor candidate per frame per tile (N candidates per tile, N: # frames) with the highest mean attention within each tile; ❷ we select a global main anchor from these N candidates using the highest temporal mean attention score and prune frames whose candidates are weakly correlated with the main anchor; and ❸ within a local square window centered at the anchor candidates of the unpruned frames, we mark tokens whose attention scores from the anchor exceed a threshold as correlated tokens. For *3D attention*, we directly select the main anchor as the token with the highest mean attention score over $k \times k \times N$ tokens ($k \times k$ spatial tile across N frames), then mark tokens within a local cuboid window whose attention score from the anchor exceeds a threshold as correlated tokens. In both cases, we use raw attention scores to exclude weakly or negatively related tokens.

Mitigating Profiling Overhead. Although SeDA is applied to only a small number of layers, bin-count profiling and anchor/correlated token identification can still be expensive. Figure 5c reveals a key insight: anchor token sets exhibit a U-shaped stability pattern across denoising timesteps. In early timesteps, the attention map is unstable, leading to high variance; we therefore skip SeDA in this region. In late timesteps, where video details are refined, updating tokens at every timestep is beneficial. For the intermediate stable region, we perform periodic updates (e.g., every 10 timesteps), which cause little quality degradation. These strategies collectively minimize SeDA's runtime overhead throughout the denoising process. Ablation results are in Appendix I.

4 Experiments

4.1 Experimental Setup

Models and Evaluation Settings. We evaluate SemanticDialect on Open-Sora 1.0 (724M) [Zheng et al., 2024] and Open-Sora 2.0 (11B) [Peng et al., 2025], covering two representative architectures: factorized spatial/temporal attention and full 3D attention, respectively. For Open-Sora 1.0, we use 100 denoising steps with CFG scale 4.0; Open-Sora 2.0 uses 50 steps with CFG scale 7.5 (text) and 3.0 (image). We employ VBench [Huang et al., 2024] and report eight key dimensions, following prior works [Ren et al., 2024, Feng et al., 2025b]. We abbreviate *Background Consistency* as *BG Consist.* and *Semantic & Style Consistency* as *SS. Consist.* Ablation studies use a reduced prompt set for computational efficiency. In addition, following EvalCrafter [Liu et al., 2024b], we report CLIPSIM for text–video alignment and CLIP-Temp for temporal consistency, DOVER [Wu et al.,

Table 2: Comparison with additional quantization baselines on Open-Sora 1.0. Effective bitwidths are reported as activation/weight (A/W). Higher is better for all quality metrics.

Method	Eff. BW (A/W)	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
SemanticDialect	5.1/4.6	54.00	59.52	97.09	44.44	91.57	95.88	32.15	24.64
PTQ4DiT	4.0/4.0	30.99	48.32	85.39	77.78	90.87	0.00	0.00	4.65
Q-DiT	4.1/4.1	41.70	45.19	96.24	89.81	86.33	94.75	7.10	17.28
SVDQuant	4.9/16[†]	53.41	61.05	95.03	58.33	88.10	95.04	33.38	24.71

[†]SVDQuant retains 16-bit activation storage; its MAC-weighted effective activation precision is approximately 4.7 bits.

2023] for aesthetic and technical quality (VQA-A/T), and flow score for motion information. Details are in Appendices C and D.

Baseline. We compare SemanticDialect against representative block-wise formats: MXFP4 [Open Compute Project, 2023], which uses power-of-two block scaling with a custom 5-bit exponent (covering the FP16 range), and NVFP4 [NVIDIA, 2025], which uses FP8 block scaling with additional FP32 per-tensor scales. We also compare against two VDiT quantization schemes: ViDiT-Q [Zhao et al., 2024], which combines fine-grained grouping, channel balancing, and mixed precision; and Q-VDiT [Feng et al., 2025b], which applies low-rank approximation of quantization error and inter-frame distribution distillation. Q-VDiT is compared only on Open-Sora 1.0 as its implementation requires learning parameters for isolated temporal attention. We additionally compare against BlockDialect [Jang and Tambe, 2025], a mixed-format quantization that also performs per-block dialect selection but uses a 16-dialect formatbook and a different dialect-selection strategy. SemanticDialect uses *num_groups*=8 by default. For all block-wise methods, we use block sizes of 16 and 32. For ViDiT-Q and Q-VDiT, we apply their mixed-precision settings for comparable effective bitwidths.

4.2 Main Results

Table 1 summarizes the main results. On Open-Sora 1.0, prior VDiT quantization methods and MXFP4 often fail to generate recognizable videos in the sub-5-bit activation regime, as evidenced by abnormally elevated dynamic degree scores caused by noise and degraded scene consistency. While BlockDialect and NVFP4 produce better results, SemanticDialect achieves superior aesthetic and imaging quality with enhanced scene consistency. On Open-Sora 2.0, SemanticDialect still broadly outperforms or matches all baselines, approaching FP16 quality. Although a few VBench metrics exceed FP16, these isolated score reversals should not be interpreted as superior overall video quality, as quantization can perturb the stochastic generation trajectory and different VBench dimensions should be considered collectively. Note that SemanticDialect operates at a higher effective bitwidth on Open-Sora 1.0 than 2.0 due to matrix activation decomposition. Notably, FB4 alone (without decomposition and SeDA) already shows strong performance against NVFP4 across most metrics; additional metric comparisons are provided in Appendix J. A more rigorous bitwidth-matched comparison is provided in Appendix E.3.

4.3 Additional Baselines

We additionally evaluate SVDQuant Li et al. [2024], PTQ4DiT Wu et al. [2024], and Q-DiT Chen et al. [2025a] on Open-Sora 1.0. As shown in Table 2, PTQ4DiT and Q-DiT degrade substantially under W4A4 quantization. This suggests that these methods do not directly transfer from their published configurations, which primarily use 8-bit activations, to the more aggressive 4-bit activation regime. SVDQuant remains competitive; however, this comparison is not strictly resource matched and favors SVDQuant in both weight and activation storage: its rank-32 high-precision branch increases weight storage by approximately 15%, while its low-rank path requires retaining FP16/BF16 inter-layer activations.

Importantly, these methods are complementary rather than mutually exclusive with SemanticDialect. SemanticDialect is primarily a format-level contribution, whereas the compared baselines introduce algorithm-level quantization techniques. For example, Appendix L further shows that rotation-based outlier suppression produces block distributions that are better matched to FB4.

Table 3: Dialect selection strategy comparison (LUT-based vs. MSE-based). LUT-based selection matches or outperforms MSE-based strategy. We denote *num_group* by g.

Model	Method	Eff. bit (A/W)	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
Open-Sora 1.0	Exact MSE	4.31/4.31	45.84	50.44	98.40	45.83	90.71	95.78	9.52	19.94
	FB4 (g=32)		45.97	50.61	98.37	48.61	90.46	95.63	10.12	19.77
	FB4 (g=8)		45.06	50.15	98.41	44.44	90.68	95.41	9.60	19.73
Open-Sora 2.0	Exact MSE	4.31/4.31	57.52	64.01	98.67	59.72	92.32	97.01	40.63	27.46
	FB4 (g=32)		57.62	63.31	98.67	54.17	92.35	97.03	42.63	27.54
	FB4 (g=8)		57.44	63.80	98.69	58.33	92.07	96.81	40.63	27.54

Table 4: Stepwise ablation on Open-Sora 1.0 (OpenSora t2v_sample prompt set).

Method	Block size	FVD-FP16 (↓)	CLIP-Temp	CLIPSIM	VQA-A	VQA-T	Δ Flow(↓)
FP16	–	–	0.9987	0.1798	51.66	48.39	–
FB4	16	1.24	0.9983	0.1781	50.72	47.87	0.26
+Decomp.		0.80	0.9989	0.1751	51.01	48.14	0.31
+SeDA		0.76	0.9988	0.1757	51.03	48.10	0.26
FB4	32	1.59	0.9986	0.1778	50.29	47.78	0.84
+Decomp.		1.36	0.9986	0.1758	50.88	48.06	0.55
+SeDA		1.26	0.9983	0.1763	50.86	48.05	0.44

4.4 Comparison with MSE-based Dialect Selection

Table 3 compares LUT-based and MSE-based dialect selection strategies at block size 32, with two observations. First, FB4 (*num_groups*=32) with LUT-based dialect selection performs comparably to exact MSE-based selection, suggesting that MSE-driven optimization, often dominated by a few extreme outliers, does not necessarily translate to superior end-to-end video quality. Second, even coarse group-wise maximum extraction (*num_groups*=8) remains competitive with exact MSE selection, indicating that a rough approximation of large-magnitude values is sufficient to guide effective dialect selection. SemanticDialect uses a default *num_groups* of 8 to balance performance and efficiency.

4.5 Stepwise Evaluation with Additional Metrics

To analyze the contribution of each proposed technique, we conduct a stepwise ablation on Open-Sora 1.0 using the OpenSora `t2v_sample` prompt set, averaged over random seeds. As shown in Table 4, FVD-FP16, which measures feature-distribution similarity to the FP16 baseline and correlates with human perception, consistently improves as more techniques are incorporated. Similarly, ΔFlow score, which captures the gap in motion intensity relative to the FP16 baseline, is minimized when all techniques are applied.

Across most metrics, the most substantial gain occurs upon applying activation decomposition, highlighting the importance of handling quantization-sensitive layers. SeDA complements this error reduction by promoting consistent dialect assignments across semantically related tokens. Table 5 further evaluates its incremental contribution to FP16-referenced appearance and motion fidelity. FrameLPIPS measures the average frame-wise perceptual deviation from the FP16 reference, whereas FloLPIPS emphasizes perceptual errors in regions with altered motion. SeDA reduces both FloLPIPS and FrameLPIPS on VBench and UCF-101, with all four paired 95% confidence intervals (CIs) excluding zero.

4.6 Hardware Deployment Analysis

Implementation. We implement SemanticDialect in RTL and CUDA to demonstrate its deployment practicality. For RTL, we extend FlexASR [Tambe et al., 2022] on TSMC 7nm at 250MHz (Appendix E.1). For GPU kernel, FB4's normalized integer values (0–15) are incompatible with native MXFP4 kernels but representable in FP8 e4m3, making them MXFP8-compatible. As H100 is our only available GPU and does not natively support MXFP8 GEMM, we estimate Blackwell-class latency by halving FP16 GEMM latency, reflecting MXFP8's 2$\times$ throughput (Appendix E.2). We also implement an MXFP4 kernel (same power-of-two scaling with FP4) assuming the same MXFP8 GEMM to isolate FB4 quantization overhead, not to compare latency against native 4-bit hardware.

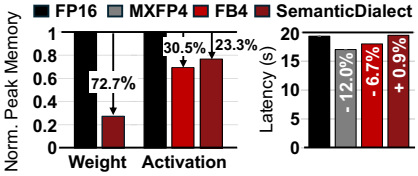
Table 5: FP16-referenced perceptual fidelity (w/ SeDA−w/o SeDA); lower is better.

Metric	VBench	UCF-101
ΔFloLPIPS	−0.36	−0.52
ΔFrameLPIPS	−0.22	−0.47

*** Synthesized @ 7nm, 250MHz**

Format	MAC Unit		Quantization Unit	
	Area (μm^2)	Power (mW)	Area (μm^2)	Power (mW)
NVFP4	15487.5	1.88	16426.4	6.77
MXFP4	8302.4	1.05	1504.9	1.47
FB4	9554.6	1.13	4042.5	3.29

(a)



(b) (c)

Figure 6: Hardware deployment analysis. (a) RTL: MAC and quantization unit comparison. (b) CUDA: Peak memory. (c) CUDA: DiT block inference latency (projected Blackwell-class latency).

RTL Results. FB4 achieves notably higher hardware efficiency than NVFP4 (Figure 6a). NVFP4's inefficiency stems from FP8 block-wise and FP32 tensor-wise scaling, which require extra multipliers and complicate FSM logic, whereas FB4 and MXFP4 reduce scaling to simple exponent arithmetic. While MXFP4 shows marginally better efficiency, it substantially degrades generation quality. Formatbook area of FB4 MAC accounts for only 12.6% of multipliers (excluding accumulators).

CUDA Kernel Results. *Peak Memory:* SemanticDialect reduces weight and activation peak memory by 72.7% and 23.3%, respectively (Figure 6b). Activation-memory savings are limited by the coexistence of quantized and non-quantized tensors (e.g., residual connections and GeLU inputs). SeDA requires additional activation storage for attention-score processing; disabling it increases the activation-memory reduction to 30.5%.

Inference Latency: Figure 6c shows a 12.0% latency reduction over FP16 for MXFP4 executed via MXFP8 GEMM. For context, MXFP8 achieves an average latency reduction of approximately 17.5% over BF16 in published B200 benchmarks [PyTorch, 2026]. FB4's online quantization adds 5.9% latency overhead relative to MXFP4, while the full SemanticDialect pipeline incurs 0.9% overhead relative to FP16. Further reductions may be possible through kernel fusion and native MXFP8 execution (Appendix E.2).

5 Conclusion

In this work, we propose SemanticDialect, a PTQ method for VDiTs that improves generation quality under 4-bit quantization via LUT-based dialect selection, attention-guided activation decomposition, and SeDA for spatiotemporal consistency. Experiments demonstrate that SemanticDialect consistently outperforms prior VDiT quantization methods and block-wise format baselines, showing that accurate edge deployment is achievable with scalable mixed-format quantization. **Limitation**: While this work demonstrates the effectiveness of semantic-aware mixed-format quantization, SeDA's configurations—including anchor and correlated token selection, sub-formatbook design, and timestep scheduling—may not be globally optimal, and its incremental benefits may depend on the model architecture. Future work could tailor these configurations to architectural differences and video-specific characteristics such as motion intensity.

Acknowledgments

This work was supported in part by the Stanford Center for Portable Accelerated Learning (PORTAL), Apple Inc., Fixstars Corporation, and the National Science Foundation (NSF) under Award No. 2541050. This work used Marlowe [Kapfer et al., 2025], Stanford University's GPU-based Computational Instrument, supported by Stanford HAI and Stanford Research Computing. The authors declare no competing interests.

References

AMD. AMD CDNA™ 4 Architecture Whitepaper. AMD CDNA 4 Whitepaper, June 2025. Accessed 2026-02-06.

Saleh Ashkboos, Ilia Markov, Elias Frantar, Tingxuan Zhong, Xincheng Wang, Jie Ren, Torsten Hoefler, and Dan Alistarh. Quik: Towards End-to-End 4-Bit Inference on Generative Large Language Models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3355–3371, 2024a.

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, 2024b.

Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align Your Latents: High-Resolution Video Synthesis with Latent Diffusion Models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023.

Minghong Cai, Xiaodong Cun, Xiaoyu Li, Wenze Liu, Zhaoyang Zhang, Yong Zhang, Ying Shan, and Xiangyu Yue. DiTCtrl: Exploring Attention Control in Multi-Modal Diffusion Transformer for Tuning-Free Multi-Prompt Longer Video Generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 7763–7772, 2025.

Yasong Cao, Mei Wen, Zhongdi Luo, Xin Ju, Haolan Huang, Junzhong Shen, and Haiyan Chen. ABS: Accumulation Bit-Width Scaling Method for Designing Low-Precision Tensor Core. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2024.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

Lei Chen, Yuan Meng, Chen Tang, Xinzhu Ma, Jingyan Jiang, Xin Wang, Zhi Wang, and Wenwu Zhu. Q-DiT: Accurate Post-Training Quantization for Diffusion Transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 28306–28315, 2025a.

Yuzong Chen, Xilai Dai, Jake Hyun, Chi-Chih Chang, Wonsuk Jang, Yuheng Wu, Thierry Tambe, Jae-sun Seo, and Mohamed S Abdelfattah. RaZeR: Pushing the Limits of NVFP4 Quantization with Redundant Zero Remapping. *arXiv preprint arXiv:2501.04052*, 2025b.

Jack Cook, Junxian Guo, Guangxuan Xiao, Yujun Lin, and Song Han. Four Over Six: More Accurate NVFP4 Quantization with Adaptive Block Scaling. *arXiv preprint arXiv:2512.02010*, 2025.

Steve Dai, Rangha Venkatesan, Mark Ren, Brian Zimmer, William Dally, and Brucek Khailany. VS-Quant: Per-Vector Scaled Quantization for Accurate Low-Precision Neural Network Inference. *Proceedings of Machine Learning and Systems*, 3:873–884, 2021.

Ning Ding, Jing Han, Yuchuan Tian, Chao Xu, Kai Han, and Yehui Tang. Post-Training Quantization for Diffusion Transformer via Hierarchical Timestep Grouping. *arXiv preprint arXiv:2503.06930*, 2025.

Weilun Feng, Haotong Qin, Chuanguang Yang, Xiangqi Li, Han Yang, Yuqi Li, Zhulin An, Libo Huang, Michele Magno, et al. S^2 Q-VDiT: Accurate Quantized Video Diffusion Transformer with Salient Data and Sparse Token Distillation. *arXiv preprint arXiv:2508.04016*, 2025a.

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.

Weiming Hu, Zihan Zhang, Haoyan Zhang, Chen Zhang, Cong Guo, Yu Feng, Tianchi Hu, Guanglin Li, Guipeng Hu, Junsong Wang, et al. M2XFP: A Metadata-Augmented Microscaling Data Format for Efficient Low-bit Quantization. *arXiv preprint arXiv:2601.19213*, 2026.

Xinyu Huang, Youcai Zhang, Jinyu Ma, Weiwei Tian, Rui Feng, Yuejie Zhang, Yaqian Li, Yandong Guo, and Lei Zhang. Tag2Text: Guiding Vision-Language Model via Image Tagging. *arXiv preprint arXiv:2303.05657*, 2023.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. VBench: Comprehensive Benchmark Suite for Video Generative Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024.

Wonsuk Jang and Thierry Tambe. BlockDialect: Block-wise Fine-grained Mixed Format Quantization for Energy-Efficient LLM Inference. In *Forty-second International Conference on Machine Learning*, 2025.

Craig Kapfer, Kurt Stine, Balasubramanian Narasimhan, Christopher Mentzel, and Emmanuel Candès. Marlowe: Stanford's gpu-based computational instrument, 2025. URL `https://doi.org/10.5281/zenodo.14751899`.

Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. MUSIQ: Multi-Scale Image Quality Transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5148–5157, 2021.

Sehoon Kim, Coleman Hooper, Amir Gholami, Zhen Dong, Xiuyu Li, Sheng Shen, Michael W Mahoney, and Kurt Keutzer. SqueezeLLM: Dense-and-Sparse Quantization. *arXiv preprint arXiv:2306.07629*, 2023.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. HunyuanVideo: A Systematic Framework for Large Video Generative Models. *arXiv preprint arXiv:2412.03603*, 2024.

Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, et al. FLUX. 1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space. *arXiv preprint arXiv:2506.15742*, 2025.

LAION-AI. Aesthetic predictor. LAION-AI Aesthetic Predictor, 2022. Accessed: 2026-02-17.

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. *arXiv preprint arXiv:2411.05007*, 2024.

Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. AMT: All-Pairs Multi-Field Transforms for Efficient Frame Interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9801–9810, 2023.

Bingyan Liu, Chengyu Wang, Tingfeng Cao, Kui Jia, and Jun Huang. Towards Understanding Cross and Self-Attention in Stable Diffusion for Text-Guided Image Editing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7817–7826, 2024a.

Shih-yang Liu, Zechun Liu, Xijie Huang, Pingcheng Dong, and Kwang-Ting Cheng. LLM-FP4: 4-bit Floating-Point Quantized Transformers. *arXiv preprint arXiv:2310.16836*, 2023.

Yaofang Liu, Xiaodong Cun, Xuebo Liu, Xintao Wang, Yong Zhang, Haoxin Chen, Yang Liu, Tieyong Zeng, Raymond Chan, and Ying Shan. EvalCrafter: Benchmarking and Evaluating Large Video Generation Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22139–22149, 2024b.

Yun-Chen Lo, Gu-Yeon Wei, and David Brooks. Nanoscaling Floating-Point (NxFP): NanoMantissa, Adaptive Microexponents, and Code Recycling for Direct-Cast Compression of Large Language Models. *arXiv preprint arXiv:2412.19821*, 2024.

Wenbo Lu, Shaoyi Zheng, Yuxuan Xia, and Shengjie Wang. ToMA: Token Merge with Attention for Diffusion Models. *arXiv preprint arXiv:2509.10918*, 2025.

Xin Ma, Yaohui Wang, Xinyuan Chen, Gengyun Jia, Ziwei Liu, Yuan-Fang Li, Cunjian Chen, and Yu Qiao. Latte: Latent Diffusion Transformer for Video Generation. *arXiv preprint arXiv:2401.03048*, 2024.

NVIDIA. Introducing NVFP4 for Efficient and Accurate Low-Precision Inference. NVIDIA Developer Blog, June 2025. Accessed 2026-02-06.

Open Compute Project. OCP Microscaling Formats (MX) Specification Version 1.0. OCP MX Specification v1.0, September 2023. Version 1.0, accessed 2026-02-06.

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

Xiangyu Peng, Zangwei Zheng, Chenhui Shen, Tom Young, Xinying Guo, Binluo Wang, Hang Xu, Hongxin Liu, Mingyan Jiang, Wenjun Li, et al. Open-Sora 2.0: Training a Commercial-Level Video Generation Model in $200K. *arXiv preprint arXiv:2503.09642*, 2025.

PyTorch. Faster Diffusion on Blackwell: MXFP8 and NVFP4 with Diffusers and TorchAO. PyTorch Blog, 2026. Accessed 2026-04-29.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning Transferable Visual Models from Natural Language Supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

Weiming Ren, Huan Yang, Ge Zhang, Cong Wei, Xinrun Du, Wenhao Huang, and Wenhu Chen. ConsistI2V: Enhancing Visual Consistency for Image-to-Video Generation. *arXiv preprint arXiv:2402.04324*, 2024.

Bita Darvish Rouhani, Ritchie Zhao, Venmugil Elango, Rasoul Shafipour, Mathew Hall, Maral Mesmakhosroshahi, Ankit More, Levi Melnick, Maximilian Golub, Girish Varatkar, et al. With Shared Microexponents, A Little Shifting Goes a Long Way. In *Proceedings of the 50th Annual International Symposium on Computer Architecture*, pages 1–13, 2023a.

Bita Darvish Rouhani, Ritchie Zhao, Ankit More, Mathew Hall, Alireza Khodamoradi, Summer Deng, Dhruv Choudhary, Marius Cornea, Eric Dellinger, Kristof Denolf, et al. Microscaling Data Formats for Deep Learning. *arXiv preprint arXiv:2310.10537*, 2023b.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A Dataset of 101 Human Actions Classes from Videos in the Wild. *arXiv preprint arXiv:1212.0402*, 2012.

Thierry Tambe, En-Yu Yang, Glenn G Ko, Yuji Chai, Coleman Hooper, Marco Donato, Paul N Whatmough, Alexander M Rush, David Brooks, and Gu-Yeon Wei. A 16-nm SoC for Noise-Robust Speech and NLP Edge AI Inference with Bayesian Sound Source separation and Attention-based DNNs. *IEEE Journal of Solid-State Circuits*, 58(2):569–581, 2022.

Zachary Teed and Jia Deng. RAFT: Recurrent All-Pairs Field Transforms for Optical Flow. In *European conference on computer vision*, pages 402–419. Springer, 2020.

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. InternVid: A Large-Scale Video-Text Dataset for Multimodal Understanding and Generation. *arXiv preprint arXiv:2307.06942*, 2023.

Yuxin Wen, Jim Wu, Ajay Jain, Tom Goldstein, and Ashwinee Panda. Analysis of Attention in Video Diffusion Transformers. *arXiv preprint arXiv:2504.10317*, 2025.

Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring Video Quality Assessment on User-Generated Contents from Aesthetic and Technical Perspectives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20144–20154, 2023.

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.

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.

Xianglong Yan, ChengZhu Bao, Zhiteng Li, Tianao Zhang, Shaoqiu Zhang, Ruobing Xie, Samm Sun, and Yulun Zhang. D ^{2} Quant: Accurate Low-bit Post-Training Weight Quantization for LLMs. *arXiv preprint arXiv:2602.02546*, 2026.

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. *arXiv preprint arXiv:2408.06072*, 2024.

Seonghyeon Ye, Yunhao Ge, Kaiyuan Zheng, Shenyuan Gao, Sihyun Yu, George Kurian, Suneel Indupuru, You Liang Tan, Chuning Zhu, Jiannan Xiang, et al. World Action Models are Zero-Shot Policies. *arXiv preprint arXiv:2602.15922*, 2026.

Tianchen Zhao, Tongcheng Fang, Haofeng Huang, Enshu Liu, Rui Wan, Widyadewi Soedarmadji, Shiyao Li, Zinan Lin, Guohao Dai, Shengen Yan, et al. ViDiT-Q: Efficient and Accurate Quantization of Diffusion Transformers for Image and Video Generation. *arXiv preprint arXiv:2406.02540*, 2024.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-Sora: Democratizing Efficient Video Production for All. *arXiv preprint arXiv:2412.20404*, 2024.

A Layer-wise Quantization Sensitivity Test

Table 6: Open-Sora 1.0: per-layer sensitivity. Top: FP16 baseline with only one layer quantized. Bottom: FB4 baseline with only one layer set to FP16. **Bold**: notable degradation or improvement.

	Aesthetic quality	Imaging quality	Motion smooth.	Dynamic degree	Subject consist.	BG. consist.	Scene consist.	SS. consist.
FP16	58.53	65.33	97.29	47.22	93.04	95.98	25.00	25.28
attn_temp.qkv	**56.64**	**59.63**	97.90	44.44	93.34	96.26	25.00	25.05
attn_temp.proj	58.47	65.55	97.27	47.22	93.12	95.88	27.90	25.21
attn.qkv	57.32	64.44	97.21	58.33	92.29	95.37	29.09	25.27
attn.proj	58.37	65.15	97.18	45.83	93.07	95.95	25.30	25.62
x_attn.q_linear	58.92	64.99	97.04	47.22	92.44	95.93	28.94	25.16
x_attn.kv_linear	58.99	66.11	97.23	51.39	93.11	96.10	28.05	25.25
x_attn.proj	59.47	65.88	97.27	48.61	93.07	96.19	27.08	25.46
mlp.fc1	57.55	65.48	97.13	50.00	**91.80**	95.87	26.71	24.67
mlp.fc2	**54.79**	**60.12**	98.59	**29.17**	93.69	96.60	**18.38**	**24.19**
FB4	45.06	50.15	98.41	44.44	90.68	95.41	9.60	19.73
attn_temp.qkv	**49.23**	**58.51**	98.02	47.22	91.06	95.51	**14.14**	**21.62**
attn_temp.proj	46.12	50.85	98.45	37.50	90.98	95.60	9.97	20.21
attn.qkv	47.30	51.62	98.43	36.11	90.96	95.91	13.62	20.96
attn.proj	46.38	49.90	98.48	40.28	90.66	95.78	11.76	20.69
x_attn.q_linear	45.65	51.65	98.50	43.06	91.06	95.67	11.38	20.15
x_attn.kv_linear	45.44	50.76	98.36	40.28	90.51	95.44	11.16	20.56
x_attn.proj	45.01	50.00	98.50	47.22	90.72	95.30	11.68	19.72
mlp.fc1	46.44	51.25	98.45	38.89	90.99	96.09	13.47	20.85
mlp.fc2	**51.05**	**53.37**	97.61	44.44	90.23	95.28	**20.68**	**22.32**

Table 7: Open-Sora 2.0: per-layer sensitivity. Top: FP16 baseline with only one layer quantized. Bottom: FB4 baseline with only one layer set to FP16. **Bold**: notable degradation or improvement.

	Aesthetic quality	Imaging quality	Motion smooth.	Dynamic degree	Subject consist.	BG. consist.	Scene consist.	SS. consist.
FP16	58.05	64.58	98.83	59.72	94.43	97.80	40.85	27.37
img_attn.qkv	57.74	64.51	98.80	56.94	93.83	97.74	41.07	27.40
img_attn.proj	57.91	64.64	98.83	55.56	94.21	97.83	41.74	27.46
img_mlp.0	**57.35**	64.81	98.63	59.72	93.51	97.27	41.44	27.61
img_mlp.2	**57.41**	64.85	98.79	55.56	94.27	97.76	41.89	27.49
img_mod.lin	57.90	65.31	98.71	54.17	93.65	97.57	40.63	27.42
txt_attn.qkv	58.07	64.62	98.82	61.11	94.39	97.87	40.03	27.36
txt_attn.proj	58.07	64.76	98.83	56.94	94.23	97.62	40.48	27.38
txt_mlp.0	57.85	64.67	98.84	58.33	94.24	97.76	41.15	27.43
txt_mlp.2	58.01	64.51	98.79	55.56	94.03	97.81	43.30	27.34
txt_mod.lin	58.08	64.46	98.80	56.94	94.10	97.62	41.67	**27.19**
linear1	58.09	64.46	98.78	**51.39**	93.95	97.93	**39.06**	27.33
linear2	58.19	64.86	98.74	55.56	94.07	97.73	40.85	27.51
modulation.lin	58.49	**61.84**	98.86	61.11	94.06	97.92	39.58	27.33
FB4	57.44	63.80	98.69	58.33	92.07	96.81	40.63	27.54
img_attn.qkv	57.74	63.35	98.64	54.17	92.19	96.97	42.04	27.63
img_attn.proj	57.60	63.46	98.66	56.94	92.38	96.77	40.55	27.62
img_mlp.0	57.63	63.28	98.73	59.72	92.87	97.28	42.78	27.51
img_mlp.2	57.50	63.58	98.69	55.56	92.29	97.05	41.89	27.41
img_mod.lin	57.66	62.49	98.68	52.78	92.12	97.16	41.59	27.64
txt_attn.qkv	57.60	63.99	98.63	58.33	92.22	96.65	42.56	27.70
txt_attn.proj	57.56	63.79	98.67	55.56	92.27	96.98	41.89	27.59
txt_mlp.0	57.42	63.75	98.66	58.33	92.19	96.84	43.53	27.63
txt_mlp.2	57.40	63.92	98.68	63.89	92.02	96.80	43.15	27.70
txt_mod.lin	57.33	63.49	98.56	55.56	92.16	96.68	37.35	27.57
linear1	57.69	64.13	98.65	62.50	92.30	96.86	39.51	27.49
linear2	57.28	63.49	98.78	55.56	92.35	96.95	41.59	27.48
modulation.lin	57.64	**65.42**	98.53	54.17	91.84	96.09	**44.05**	**27.73**

Tables 6 and 7 summarize the layer-wise quantization sensitivity of Open-Sora 1.0 and 2.0. In Open-Sora 1.0, `attn_temp.qkv` and `mlp.fc2` are clearly the most sensitive layers. In the larger Open-

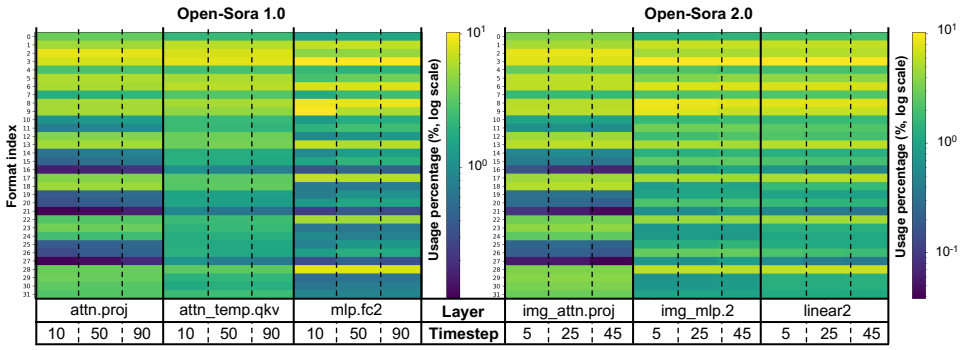


Figure 7: Block-wise format usage.

Sora 2.0, the model is generally more robust to quantization and exhibits smaller layer-wise variation, but `modulation.lin` (the modulation layer in `SingleProcessBlock`) remains the most sensitive. Additonally, we prioritize `txt_mod.lin` (the text modulation layer in `DoubleProcessBlock`) for activation decomposition because its vector-shaped activation makes decomposition computationally inexpensive, and it empirically shows strong results when decomposed.

B Block-wise Format Usage

Figure 7 reports the average format usage across layers and denoising timesteps, averaged over 10 prompts. We observe clear layer-specific preferences, but no format collapse (i.e., every format is selected at least occasionally). The least-used formats (Formats 16, 21, and 27) appear at around 0.1% in the projection layer immediately after spatial attention, yet are meaningfully used in other layers. This indicates that the full format set is leveraged when beneficial. Overall, lower-index formats, which provide a narrower dynamic range (smaller block maxima), are selected more often, consistent with the predominance of small-magnitude normalized values. Importantly, usage patterns depend primarily on layer type rather than denoising timestep. For example, in Open-Sora 1.0, temporal attention QKV projection (`attn_temp.qkv`) shows relatively uniform usage, whereas the final MLP layer (`mlp.fc2`) consistently concentrates on a few formats (e.g., Formats 28, 22, and 17) even within the same dynamic range group. These results suggest that block-wise mixed formats systematically adapt to each layer without explicit calibration.

C Evaluation Metrics Description

C.1 VBench Benchmark.

Following VBench [Huang et al., 2024] and prior VDiT quantization work [Zhao et al., 2024, Feng et al., 2025b], we evaluate on 8 VBench dimensions spanning three aspects of video generation: **(i) Frame-wise Quality**, **(ii) Temporal Quality**, and **(iii) Semantics**. We use the official VBench prompt sets below and generate three videos per prompt using different random seeds to improve stability. For the main results, we use the full prompt set, while for other evaluations we subsample one-third of the prompts within each category to reduce inference cost.

- `overall_consistency.txt` (93 prompts) for semantic & style consistency, aesthetic quality and imaging quality
- `subject_consistency.txt` (72 prompts) for subject consistency, dynamic degree and motion smoothness
- `scene.txt` (86 prompts) for background consistency and scene consistency

C.2 VBench dimensions

Frame-wise Quality measures per-frame visual quality without considering temporal effects:

- *Imaging Quality* captures distortions such as blur, noise, and over-/under-exposure using the MUSIQ [Ke et al., 2021] image quality predictor.
- *Aesthetic Quality* reflects human-perceived aesthetic appeal using the LAION [LAION-AI, 2022] aesthetic predictor.

Temporal Quality measures cross-frame consistency and motion:

- *Motion Smoothness* measures whether motion is smooth by comparing interpolated frames from even-numbered frames using AMT [Li et al., 2023] with the original odd-numbered frames.
- *Dynamic Degree* quantifies motion magnitude via optical-flow statistics using RAFT [Teed and Deng, 2020].
- *Subject Consistency* assesses whether subject appearance remains consistent over time using DINO [Caron et al., 2021] features.
- *Background Consistency* evaluates temporal consistency of background scenes using CLIP [Radford et al., 2021] image features.

Semantics measures text adherence:

- *Scene Consistency* evaluates whether captions generated from the video [Huang et al., 2023] match the intended scene described by the prompt.
- *Semantic&Style Consistency* measures overall text–video alignment (including style consistency) computed by ViCLIP [Wang et al., 2023]. VBench refers to this metric as *Overall Consistency*; we rename it to avoid the misleading implication that it is a representative aggregate over other metrics.

C.3 Additional Metrics

In addition to reporting the VBench dimension scores, we include the following widely used metrics. We use the 10-prompt and 48-prompt sets provided by Open-Sora (t2v_sample and t2v_sora, respectively), 93-prompt VBench `overall_consistency.txt`, and the 101-prompt UCF-101 [Soomro et al., 2012] prompt set for these evaluations. As acknowledged in prior works [Zhao et al., 2024, Feng et al., 2025b], finite-sample FVD can yield unstable results; we therefore report FVD-FP16 across multiple prompt sets to improve reliability. All experiments are conducted with three random seeds.

- *CLIPSIM* and *CLIP-Temp* are computed using the EvalCrafter implementation [Liu et al., 2024b]: CLIPSIM averages frame-wise image–text similarity, while CLIP-Temp averages similarity between consecutive frames to reflect semantic temporal consistency.
- *DOVER VQA* [Wu et al., 2023] reports *VQA-T* (technical quality; distortions such as noise/blur/exposure) and *VQA-A* (aesthetic quality; composition, color harmony, naturalness).
- *Flow Score* uses RAFT [Teed and Deng, 2020] to estimate dense optical flow between consecutive frames and averages the flow magnitude as a proxy for motion strength; following prior practice [Feng et al., 2025b], we report Δ*Flow* (difference to the FP16 baseline) for robustness against pathological failures. We additionally report *Flow MAE*, which averages the absolute per-video difference between the quantized and paired FP16 videos' mean RAFT optical-flow magnitudes, to measure motion fidelity without cancellation between over- and under-estimation.
- *FVD-FP16* computes feature distribution distance between generated videos from FP16 model and quantized model using I3D features [Blattmann et al., 2023]; lower is better. It quantifies semantic and quality regression due to quantization. FVD-FP16 has a high correlation with human perception [Zhao et al., 2024].

D Evaluation Setting Details

Following prior work [Zhao et al., 2024, Feng et al., 2025b], for Open-Sora 1.0 we keep `embedder`, `t_block`, and `final_layer` in FP16. These modules lie at the network boundaries and account for

a negligible fraction of the total computation. For Open-Sora 2.0, we similarly target only the linear layers in DiT blocks, `single_blocks` and `double_blocks`. We use FP16 weights and activations for the full-precision baseline. In our experiments, Open-Sora 1.0 generates 16 frames (8 fps) at 512×512 resolution, while Open-Sora 2.0 generates 33 frames (24 fps) at 256 px resolution with a 16:9 aspect ratio. Weight dialect selection and quantization follow an MSE-based criterion.

We apply activation decomposition to quantization-sensitive layers: `attn_temp.qkv` and `mlp.fc2` in Open-Sora 1.0, and `modulation.lin` and `txt_mod.lin` in Open-Sora 2.0. For layers that precede attention score calculation (e.g., `attn_temp.qkv`), salient tokens from the previous timestep are reused. We apply SeDA to MLP layers: `mlp.fc1` and `mlp.fc2` in Open-Sora 1.0, and `v_proj`, `linear1` (fc1 equivalent), and `linear2` (fc2 equivalent) in `SingleStreamBlock` for Open-Sora 2.0. For `v_proj` and `linear1`, which perform the first MLP operation yet precede attention in the same timestep, we reuse the 8-dialect sub-formatbook from the previous timestep. To mitigate SeDA overhead, we skip the first 20% (40%) of timesteps in Open-Sora 1.0 (2.0) and update anchor/correlated tokens every 10 (5) timesteps for block sizes 32 (16), with per-timestep updates in the last 10% of timesteps.

In Open-Sora 1.0, `mlp.fc2` uses both activation decomposition and SeDA; we constrain both the original-value and residual quantization to use the sub-formatbook assigned by SeDA. For SeDA, we use attention thresholds of 8 (spatial) and 3 (temporal) for Open-Sora 1.0, with a 4×4 spatial tile and a 5×5 local correlation window. For Open-Sora 2.0, we use the same tile and window sizes and set the threshold to 5. For overlapping window regions, we assign them to the subsequent anchor token.

All experiments are run on 80GB NVIDIA H100 GPUs (CUDA 12.6). Each video generation runs on a single GPU and takes tens to hundreds of seconds depending on the number format and model. Four GPUs are used in parallel to accelerate evaluation across prompts.

E Hardware Deployment Details

E.1 RTL Implementation

We modify two modules of the FlexASR [Tambe et al., 2022] accelerator: *PECore*, which handles main MAC computation and buffer management of processing elements (PEs), and *GBControl*, which aggregates PE outputs and broadcasts data to PEs via control FSM logic and global shared buffer management. Specifically, we replace the input/output interface and MAC computation to support the target format in PECore, and add a quantization unit in GBControl.

Each MAC has 8 lanes, where each lane handles block-wise (block size 16) multiply-accumulate operations per cycle. For FB4, each lane uses its own formatbook (Figure 3a), shared among its multipliers. During synthesis, each index (0–7) is optimized separately (e.g., index 0 is mapped to constant 0, eliminating LUT retrieval). As a result, the total formatbook area accounts for only 12.6% of the multiplier area (excluding accumulators).

For the quantization unit, NVFP4 incurs higher latency and hardware complexity due to tensor-level scaling, which cannot be pipelined at block granularity, and floating-point block-wise scaling, which requires more cycles than the power-of-two scaling used by MXFP4 and FB4 (accomplished via exponent addition). Consequently, NVFP4 requires 6 cycles per block for quantization, while FB4 requires only one at 250MHz.

E.2 CUDA Kernel Implementation

We implement our CUDA kernels on an NVIDIA H100 GPU (CUDA 12.6), using Open-Sora 2.0 as the base implementation. Since FB4 uses power-of-two scaling and its normalized integer values (0–15) are exactly representable in FP8 E4M3, it is directly compatible with MXFP8 GEMM. However, H100 does not natively support MXFP8 GEMM or the fine-grained block-wise scaling required by MX formats. We therefore use cuBLAS FP16 GEMM while decoding 4-bit weights and activations to FP16 on the fly and applying the corresponding block-wise scales. Quantization is fused with the preceding operation (e.g., GeLU or LayerNorm) where possible, except for FlashAttention kernels where such fusion is less straightforward.

Peak memory usage is measured using the PyTorch API. Since our H100 implementation materializes decoded operands in FP16 rather than FP8, we halve the decoded weight and activation storage

Table 8: Sensitivity of the Blackwell latency projection to the assumed GEMM and decoding speedups. All estimates start from directly measured H100 profiles and preserve the measured non-GEMM/non-decoding time and overlap structure.

GEMM / Decode Speedup	FB4 vs. FP16	SemanticDialect vs. FP16	SemanticDialect vs. FB4
2× / 2×	−7.2%	+0.9%	+8.2%
2× / 1.5×	−4.3%	+3.5%	+8.0%
1.5× / 1.5×	+1.0%	+8.7%	+7.7%

when estimating MXFP8-equivalent peak memory. The pre-decoded 4-bit weight storage remains unchanged.

Measured H100 execution. We distinguish directly measured H100 results from projected Blackwell performance. Across five denoising runs on H100, FB4 and SemanticDialect require 22.5 s and 24.0 s, respectively, corresponding to a 6.8% incremental latency overhead for SemanticDialect over FB4. In a representative NVIDIA Nsight trace, the additional critical-path GEMM and decoding contributions introduced by SemanticDialect are only 0.105 s and 0.001 s, respectively. Most of the remaining overhead arises from the additional SeDA execution path and limited kernel-fusion opportunities in our current H100 implementation.

Blackwell projection. MXFP8-capable Blackwell hardware was unavailable during our evaluation, so the reported Blackwell results are trace-based projections derived from the measured H100 profiles rather than direct hardware measurements. In the measured SemanticDialect trace, GEMM and decoding account for approximately 26% and 12% of wall-clock time, respectively. We leave all measured non-GEMM/non-decoding time unchanged and apply hardware speedup factors only to the isolated GEMM and decoding contributions, including their measured overlap.

Although FB4 stores 4-bit codes, its commercial-GPU execution path targets MXFP8-compatible GEMMs. We therefore use Blackwell's nominal 2× MXFP8-to-FP16 Tensor Core throughput ratio as the GEMM speedup factor. On H100, decoded FP16 operands must be materialized before the cuBLAS GEMM because native MXFP8 execution and operand-side GEMM fusion are unavailable. Native MXFP8 would reduce these intermediate operands from FP16 to FP8, while a custom fused GEMM prologue could potentially eliminate the explicit materialization altogether. We therefore use 2× as the nominal decoding speedup.

To account for sub-peak GEMM efficiency on the evaluated model shapes and for decoding costs that may not scale purely with memory traffic, we additionally evaluate more conservative projection assumptions. Table 8 summarizes the resulting sensitivity analysis.

The sensitivity analysis shows that the absolute performance relative to FP16 depends on the achievable MXFP8 GEMM and decoding efficiency, while the incremental overhead of SemanticDialect over FB4 remains approximately 8% across the evaluated assumptions. This is because the sensitivity primarily affects the shared low-bit execution path rather than the additional SemanticDialect-specific computation.

A substantial portion of the remaining H100 overhead comes from additional kernel launches introduced by FB4 quantization and SeDA, as well as the limited prologue/epilogue fusion available with cuBLAS GEMM on H100. We therefore expect further optimizations, such as CUDA Graphs and CUTLASS-based native MXFP8 GEMM kernels with fused quantization/dequantization on Blackwell, to reduce this overhead.

E.3 Bitwidth-Matched and FP16-Referenced Evaluation

We further evaluate SemanticDialect under closely matched effective bitwidths using paired, prompt-clustered bootstrap 95% confidence intervals (CIs). This analysis helps distinguish improvements that are consistently observed across paired prompts from differences that may be attributable to sampling variability.

Open-Sora 1.0. Table 9 compares the format-only methods and their activation-decomposition counterparts under closely matched effective bitwidths. For stricter resource matching, we reduce the

Table 9: Bitwidth-matched evaluation on Open-Sora 1.0. Lower is better for ΔFlow, ΔFloLPIPS, and ΔFrameLPIPS.

Method (blk size)	Eff. BW (A/W)	AQ	IQ	MS	DD	SJ	BG	SC	SS	ΔFlow↓	ΔFloL↓	ΔFrmL↓
NVFP4 (12)	4.67/4.67	52.82	54.40	97.61	53.70	89.87	95.06	20.76	23.40	12.71	49.47	51.05
BlockDialect (16)	4.56/4.56	49.03	47.65	98.02	58.80	88.25	95.12	15.72	21.97	20.38	53.03	54.50
FB4 (16)	4.63/4.63	51.83	54.84	97.63	43.06	90.99	96.21	25.73	23.80	9.38	46.68	48.13
NVFP4+decomp (12)	5.15/4.67	54.97	58.10	97.23	55.56	90.65	95.01	28.88	24.09	11.24	48.70	49.69
BlockDialect+decomp (16)	5.03/4.56	52.03	53.51	97.41	66.20	88.55	94.84	22.99	23.11	16.13	50.18	50.87
SemanticDialect (16)	5.10/4.63	54.00	59.52	97.09	44.91	91.57	95.88	32.15	24.64	8.33	43.68	44.43

Table 10: Bitwidth-matched evaluation on Open-Sora 2.0. Lower is better for ΔFlow, ΔFloLPIPS, and ΔFrameLPIPS.

Method (blk size)	Eff. BW (A/W)	AQ	IQ	MS	DD	SJ	BG	SC	SS	ΔFlow↓	ΔFloL↓	ΔFrmL↓
NVFP4 (24)	4.33/4.33	56.25	61.76	98.54	63.89	91.53	96.51	46.97	28.21	0.71	27.05	26.13
BlockDialect (32)	4.28/4.28	56.41	61.00	98.70	61.57	92.53	96.89	47.55	28.08	0.16	27.66	26.95
FB4 (32)	4.31/4.31	56.48	61.66	98.66	64.81	92.62	96.86	48.21	28.07	0.06	25.70	25.04
NVFP4+decomp (24)	4.33/4.33	56.33	62.94	98.51	66.20	92.45	96.70	45.76	28.22	0.74	26.75	25.89
BlockDialect+decomp (32)	4.28/4.28	56.45	63.46	98.48	63.89	92.11	96.36	48.79	28.23	0.15	26.25	25.55
SemanticDialect (32)	4.31/4.31	56.55	63.24	98.52	66.20	92.34	96.31	48.57	28.27	0.09	26.15	25.44

NVFP4 block size to 12, yielding effective bitwidths close to those of FB4 and SemanticDialect in the corresponding configurations.

Without activation decomposition, FB4 improves Subject Consistency (SJ), BG Consistency (BG), and Scene Consistency (SC) over iso-bit NVFP4, with the corresponding paired 95% CIs excluding zero. With activation decomposition, SemanticDialect improves Imaging Quality (IQ), SJ, BG, and SS Consistency (SS) over the matched NVFP4+decomp configuration under the same criterion. Relative to BlockDialect+decomp, the paired differences favor SemanticDialect in Aesthetic Quality (AQ), IQ, SJ, BG, SC, SS, ΔFlow, ΔFloLPIPS, and ΔFrameLPIPS, with the corresponding CIs excluding zero.

Dynamic Degree (DD) is the only VBench dimension in which matched NVFP4 consistently outperforms FB4 or SemanticDialect under the paired-CI criterion. However, this higher DD is consistent with quantization-induced motion overshoot rather than closer preservation of FP16 motion, as reflected by its larger ΔFlow.

Open-Sora 2.0. We repeat the matched-bitwidth evaluation on Open-Sora 2.0. For a conservative format-only comparison, NVFP4 uses block size 24, corresponding to an effective bitwidth of 4.33 bits, closely matching FB4 with block size 32 at 4.31 bits.

Without activation decomposition, FB4 improves SJ, BG, SC, ΔFlow, ΔFloLPIPS, and ΔFrameLPIPS over iso-bit NVFP4, with the corresponding paired 95% CIs excluding zero. Relative to BlockDialect, FB4 improves IQ, ΔFloLPIPS, and ΔFrameLPIPS under the same criterion.

When activation decomposition is applied uniformly across all formats, the remaining format-level differences become smaller, as decomposition itself reduces activation-quantization error. SemanticDialect achieves a lower ΔFlow than iso-bit NVFP4, with the paired 95% CI excluding zero, while its aggregate results remain broadly comparable to BlockDialect+decomp. This smaller separation should also be interpreted in the context of the relatively small quantized-to-FP16 gap on the 11B Open-Sora 2.0 model, which leaves limited headroom for further improvement. Across these comparisons, no baseline shows a paired-CI-supported advantage over FB4 or SemanticDialect, except for BG Consistency in the NVFP4+decomp comparison.

Additional FP16-referenced fidelity analysis. Aggregate VBench and mean LPIPS scores may obscure deviations from the paired FP16 outputs. We therefore additionally evaluate standard full-reference appearance and optical-flow fidelity on Open-Sora 2.0. Each quantized output is paired with the FP16 video generated using the same prompt and seed. PSNR measures pixel-level similarity, SSIM measures structural similarity, and VMAF combines multiple reference-based visual-quality features into a video-quality score. We also estimate optical flow for both videos using the same RAFT model; the established EPE and angular-error quantify magnitude and directional discrepancies between the resulting flow fields, respectively.

Table 11: FP16-referenced appearance and optical-flow fidelity on Open-Sora 2.0.

Metric	SemanticDialect	NVFP4+decomp	Δ
PSNR↑ (dB)	18.749	18.327	+0.422
SSIM↑	0.6077	0.5969	+0.0109
VMAF↑	13.778	11.826	+1.952
RAFT EPE↓	1.8604	2.2269	-0.3665
RAFT angular error↓ (rad)	0.4444	0.4901	-0.0457

Table 12: Formatbook-size ablation on Open-Sora 1.0 with block size 16. Higher is better for all VBench metrics, while lower is better for Flow MAE.

Size	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.	Flow MAE↓
16	51.47	53.87	97.49	**50.00**	89.99	95.70	23.76	**23.96**	11.2
32	**51.83**	**54.84**	**97.63**	43.06	**90.99**	**96.21**	**25.73**	23.80	**9.4**
64	51.31	52.34	**97.63**	44.91	90.55	95.93	24.01	23.45	10.9

As shown in Table 11, SemanticDialect achieves higher PSNR, SSIM, and VMAF and lower RAFT EPE and angular error than NVFP4+decomp. For all five metrics, the paired 95% CI of the difference excludes zero. The previously reported mean FrameLPIPS and FloLPIPS point estimates also favor SemanticDialect, although their paired 95% CIs include zero.

We further examine the high-error prompt tail relative to BlockDialect+decomp. For each metric, we first average over frames and three seeds to obtain one score per prompt, and then evaluate each method's highest-error 10% of prompts using 10,000 paired bootstrap resamples, re-selecting the tail in each resample. SemanticDialect reduces worst-10% FrameLPIPS by 1.459 relative to BlockDialect+decomp, with a paired 95% CI of $[-3.009, -0.105]$. FloLPIPS follows the same direction (-1.559; 95% CI $[-3.397, +0.058]$). The FrameLPIPS tail sets overlap by 81%, indicating that the comparison largely involves the same high-error prompts. Across tail fractions from 5% to 25%, all FrameLPIPS and FloLPIPS point estimates favor SemanticDialect, with larger gaps in the more extreme tails. Thus, while the mean metrics do not clearly separate SemanticDialect from BlockDialect+decomp, its output-level advantage is more evident among the highest-error prompts.

F Ablation Studies of Formatbook-Based Quantization

F.1 Formatbook-size ablation.

We compare 16-, 32-, and 64-entry formatbooks on Open-Sora 1.0 with a block size of 16. All variants are constructed from activations collected using eight profiling prompts. The 16-entry formatbook retains the two most frequently selected dialects in each dynamic range, while the 64-entry formatbook doubles the per-range capacity by adding 32 dialects selected to minimize profiling MSE.

Increasing the formatbook size from 16 to 32 improves six of the eight quality metrics. Although the 16-entry variant achieves a higher Dynamic Degree score, RAFT analysis indicates that this reflects quantization-induced motion overshoot rather than improved fidelity to FP16: its per-video Flow MAE is 18.9% higher than that of the 32-entry formatbook (mean difference $+1.77$, 95% CI $[+0.60, +2.98]$). Expanding the formatbook to 64 entries provides no consistent VBench improvement and further increases the flow error. These results identify 32 entries as the best evaluated operating point, balancing VBench quality and FP16 motion fidelity.

F.2 Impact isolation of weight and activation quantization.

On Open-Sora 1.0 with a block size of 16, activation-only quantization causes larger degradation than weight-only quantization in frame-wise quality and scene consistency relative to FP16 (See Table 13). Specifically, Aesthetic Quality decreases by 4.1 vs. 3.0 points, Imaging Quality by 7.3 vs. 5.5 points, and Scene Consistency by 11.4 vs. 3.5 points. These results support our focus on improving activation quantization.

Table 13: Impact isolation of weight and activation quantization on Open-Sora 1.0 with block size 16. Effective bitwidths are reported as activation/weight (A/W). Higher is better for all quality metrics.

Method	Eff. BW (A/W)	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
FP16	16/16	58.00	62.03	96.41	54.17	91.51	96.43	36.17	26.14
W4A16	16/4.63	54.99	56.50	97.15	58.33	90.47	95.45	32.68	24.50
W16A4	4.63/16	53.94	54.66	97.34	53.24	90.33	96.07	24.83	25.01
FB4	4.63/4.63	51.83	54.84	97.63	43.06	90.99	96.21	25.73	23.80

Table 14: Cross-family evaluation on CogVideoX-2B. Activation decomposition is not applied.

Method	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
FP16	56.40	58.05	97.28	66.67	94.42	95.49	37.79	24.87
FB4	51.50	54.23	97.03	40.28	91.95	94.24	31.25	24.34
NVFP4	50.42	51.82	97.03	43.98	92.25	94.62	30.43	23.56
SemanticDialect	51.42	54.20	97.03	40.74	91.90	94.43	30.69	24.35

F.3 Cross-Family Applicability on CogVideoX-2B

As shown in Table 14, We further evaluate the cross-family applicability of our approach on CogVideoX-2B without architecture-specific redesign. Using the same FB4 formatbook developed for Open-Sora, FB4 achieves higher Aesthetic Quality, Imaging Quality, and SS Consistency than NVFP4, with paired pointwise 95% confidence intervals excluding zero. NVFP4 achieves higher Dynamic Degree and BG Consistency; however, the Dynamic Degree difference is not statistically significant, with its 95% confidence interval including zero, and NVFP4 also exhibits a higher Flow MAE.

We also transfer the SeDA structural configuration from Open-Sora and perform only lightweight threshold calibration without activation decomposition. Relative to FB4, SemanticDialect produces statistically indistinguishable changes on CogVideoX-2B: both FloLPIPS and FrameLPIPS have paired 95% confidence intervals that include zero, while the VBench changes are similarly small and mixed. These results support the cross-family applicability of FB4, while indicating that SeDA does not provide a consistent additional gain when directly transferred across different model architectures. Architectural differences, such as additional normalization layers and different VAE temporal compression ratios, may therefore require architecture-specific adaptation of SeDA.

G Attention Scoring Design Choices

G.1 Incoming vs. outgoing attention

As shown in Table 15a, we find that ranking tokens by the mean outgoing attention (i.e., selecting query tokens that strongly attend to other tokens on average) performs better than ranking by mean incoming attention (i.e., how much a token is attended by others). A plausible explanation is that highly attended tokens are often generic or globally shared tokens (e.g., background/context tokens), whereas tokens with strong outgoing attention are more likely to capture distinctive, semantically informative content. We also observe *attention sinks* — tokens at frame corners and edges that attract attention from most other tokens — further making incoming attention an unfavorable criterion for anchor selection.

G.2 Self-attention vs. cross attention

Cross-attention in Open-Sora 1.0 injects text conditions into visual tokens, and in Open-Sora 2.0's multimodal attention (with concatenated text and image tokens), cross-modal interactions occur similarly. Prior work on diffusion models suggests that cross-attention maps often provide useful correspondences between text tokens and visual regions, while self-attention tends to capture spatial structural patterns less directly tied to semantic categories [Wen et al., 2025, Liu et al., 2024a, Cai et al., 2025].

Table 15: Ablation of attention-score choices on Open-Sora 1.0.

	Method	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
(a)	**Act. decomposition scoring**								
	Incoming (avg. over queries)	50.10	57.01	98.22	47.22	91.93	95.68	11.61	21.70
	Outgoing (avg. over keys)	51.25	60.00	97.62	48.61	90.74	95.04	17.04	21.99
(b)	**SeDA anchor/corr. scoring**								
	Post-softmax	50.97	60.26	97.57	47.22	90.44	94.92	16.74	22.08
	Pre-softmax	51.22	60.40	97.61	47.22	91.17	94.98	20.61	22.32

Table 16: Ablation of salient token selection and attention scoring configurations for activation decomposition (w/o SeDA) on Open-Sora 1.0. "cond" denotes allocating the full salient token budget to the conditional branch only (rather than splitting it across conditional and unconditional branches).

Method			Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
Target selection		**Timestep range**								
Full tensor		Steps 0–24	47.90	54.30	97.98	41.67	90.28	95.42	18.01	20.66
Full tensor		Steps 75–99	46.19	49.68	98.57	43.06	90.74	95.41	11.46	20.12
Random (25%)		All steps	51.14	58.23	98.10	47.22	91.15	95.74	16.96	21.46
Top-25% magnitude		All steps	50.27	54.57	98.10	36.11	91.65	95.63	13.10	21.74
Top-1 mean attn / 1×4 tile		All steps	51.25	60.00	97.62	48.61	90.74	95.04	17.04	21.99
Spatial scoring	**Attn agg. tile**	**Temporal scoring**								
Raw	4×4	Raw	48.32	55.38	97.98	48.61	90.86	95.00	16.37	21.49
ReLU	4×4	ReLU	48.34	54.58	97.95	40.28	90.44	94.88	13.39	21.49
ABS	4×4	ABS	48.26	55.20	97.96	44.44	90.27	94.90	16.00	21.59
ABS	4×4	ReLU	47.71	55.08	97.98	45.83	90.75	94.91	16.59	21.52
Raw	4×4	Raw (cond)	50.86	59.80	97.64	51.39	90.67	95.21	17.41	22.09
ABS	4×4	Raw (cond)	50.73	59.42	97.61	47.22	90.41	94.85	18.15	22.17
Raw	4×4	ReLU (cond)	50.79	59.89	97.61	50.00	90.50	94.83	15.40	22.09
ABS	4×4	ABS (cond)	50.81	60.13	97.63	48.61	90.83	94.90	14.81	22.10
ABS	4×4	ReLU (cond)	51.25	60.00	97.62	48.61	90.74	95.04	17.04	21.99
ABS (cond)	4×4	ReLU (cond)	46.50	47.74	97.16	40.28	92.99	96.51	6.92	17.04
ABS	32×32	ReLU (cond)	51.30	59.41	97.68	48.61	90.87	95.19	16.82	22.14

However, our goal is not exact text-to-region grounding; rather, we need a reliable measure of token-level semantic relatedness. For this purpose, cross-attention is not ideal because text prompts contain many semantically weak tokens (e.g., a, the, of). Without an additional online stage to identify semantically informative text tokens, cross-attention may overemphasize weakly informative function or stylistic words.

Therefore, we use self-attention scores to estimate semantic relatedness for SeDA and for salient-token selection in activation decomposition. We leave leveraging cross-attention for more precise semantic extraction to future work.

G.3 Pre-softmax vs. post-softmax scores

We apply attention-score thresholding to filter out weakly correlated tokens. Empirically, pre-softmax thresholding performs better (Table 15b) and is simpler to use for two reasons. First, post-softmax scores are inherently concentrated on a few large values, which can suppress moderately but meaningfully correlated tokens and make them harder to retain with thresholding. Second, this concentration makes threshold selection less robust, since the effective score range varies with the input distribution.

H Activation Decomposition Salient Token Selection Criteria

Table 16 compares salient token selection strategies for activation decomposition.

H.1 Comparison with Alternative Selection Strategies

Full-tensor decomposition, whether applied to early or late timestep ranges, underperforms attention-guided selection across most metrics. Random (25%) and top-25%-magnitude selection provide more balanced results but still fall short, particularly in Imaging Quality and Scene Consistency, confirming the effectiveness of attention-based token scoring.

H.2 Token Scoring

We compare raw attention scores with ABS/ReLU-transformed scores. Without conditional-branch awareness, Raw-Raw, ABS-ReLU, and ABS-ABS achieve similar quality. Once conditional-branch awareness is applied, Raw-Raw (cond) and ABS-ReLU (cond) remain comparable. We adopt ABS-ReLU as it scores higher on frame-wise quality metrics (Aesthetic and Imaging Quality), which are particularly important given that SeDA forces a non-MSE-optimal format and may affect per-frame fidelity. Also, ABS/ReLU transformations prevent large negative scores from canceling meaningful positive correlations, providing a more stable scoring signal.

H.3 Conditional Branch Awareness

With a fixed salient-token budget, we compare splitting the budget across conditional/unconditional branches versus allocating the full budget to the conditional branch. For `attn_temp.qkv`, which relies on temporal attention scoring, conditional-only selection performs substantially better; however, applying the same strategy to `mlp.fc2`, which uses spatial attention scoring, severely degrades quality (e.g., scene consistency drops to 6.92). We attribute this to classifier-free guidance, where the update depends on the conditional–unconditional activation delta: improving only the conditional branch can strengthen the conditioning signal, but a noisier unconditional branch can corrupt the delta. This effect is stronger for `mlp.fc2`, which directly produces output activations, whereas `attn_temp.qkv` is followed by attention/normalization/linear layers that can partially absorb unconditional-branch noise.

H.4 Attention Score Aggregation Tile Size

Finally, selecting salient tokens within each 4×4 tile encourages uniform spatial coverage across the frame. By contrast, global frame-level selection often concentrates tokens in a few regions—typically large background areas that contain many tokens—leading to poor coverage elsewhere and requiring much higher computational cost while achieving comparable quality.

I SeDA Ablation Studies

I.1 Temporal Axis Integration

Because Open-Sora 1.0 uses factorized attention, we first identify anchor candidates for each frame and then select a single anchor token with the highest temporal mean attention score across frames. In Table 17a, we compare a *per-frame* SeDA strategy (applying SeDA independently to each frame-level anchor and its correlated tokens) with our *temporal-axis integration* strategy (selecting a single anchor across frames for the same spatial tile and excluding frames with low attention scores from that anchor). While per-frame SeDA shows advantages in frame-wise quality (Aesthetic, Imaging Quality), temporal-axis integration outperforms it on other metrics, with the most notable improvement in scene consistency, likely because cross-frame anchors better capture scene-level semantic content shared across frames.

I.2 Anchor Tile and Local Correlation Window Size

Table 17b compares different anchor tile and local correlation window sizes in Open-Sora 1.0. A smaller local correlation window tends to limit SeDA's effectiveness by reducing the number of correlated tokens. A larger window can improve some metrics but forces the same sub-formatbook on too many tokens and increases overlap between neighboring windows, making accurate anchor–correlated token pairing more difficult. Similarly, overly small or large anchor tile sizes also degrade performance, suggesting that both parameters benefit from careful balancing.

Table 17: SeDA configuration ablations on Open-Sora 1.0. In the timestep update schedule, the update period denotes the token-update interval in timesteps (1 = every timestep); "x (20–89)" means period x on timesteps 20–89 and period 1 otherwise.

	Method		Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
(a)	**Temporal-axis integration**									
	Per-frame anchors (2D SeDA)		51.49	60.93	97.55	45.83	91.04	94.74	14.73	22.17
	Cross-frame main anchor (3D SeDA)		51.22	60.40	97.61	47.22	91.17	94.98	20.61	22.32
(b)	**Anchor tile**	**Correlation window**								
	4×4	7×7	50.88	60.11	97.59	48.61	91.15	95.00	18.90	22.17
	4×4	3×3	51.07	60.18	97.58	43.06	90.67	94.81	19.12	21.92
	4×4	5×5	51.22	60.40	97.61	47.22	91.17	94.98	20.61	22.32
	8×8	7×7	51.19	60.65	97.56	52.78	90.99	94.86	18.75	22.28
	2×2	3×3	50.93	60.32	97.63	51.39	91.05	94.82	15.63	22.17
(c)	**Timesteps**	**Update period**								
	0–99	1	51.48	60.54	97.58	50.00	90.92	95.04	16.74	22.32
	20–99	1	51.45	60.99	97.58	45.83	91.14	95.04	16.15	21.99
	20–99	10	50.75	60.46	97.60	41.67	90.74	94.96	15.48	22.16
	20–99	10 (20–89)	51.22	60.40	97.61	47.22	91.17	94.98	20.61	22.32
	20–99	5 (20–89)	51.41	60.43	97.59	50.00	91.09	94.94	18.30	22.18
	20–99	20 (20–89)	51.10	60.58	97.60	47.22	91.02	94.98	15.40	22.11

Table 18: Comparison of alternative SeDA grouping mechanisms on Open-Sora 1.0.

Method	AQ	IQ	MS	DD	SJ	BG	SC	SS
SeDA	51.09	57.04	97.40	51.39	89.48	95.22	20.93	22.94
Temporal voting	50.79	56.60	97.32	55.09	88.96	95.01	21.17	22.97
Global-static	51.06	56.46	97.34	55.09	89.02	95.38	21.44	22.89

I.3 Timesteps for Applying SeDA Under Overhead Constraints

Table 17c compares when SeDA is applied and how often anchor/correlated tokens are updated over denoising timesteps. Skipping SeDA during the initial $\sim$20% of denoising causes minimal degradation, likely because early-stage attention maps are unstable and activations remain noise-dominated, making anchor/correlated token selection unreliable. Updating tokens every 10 timesteps in the intermediate region has little effect on performance, whereas infrequent updates in the final 10% of timesteps degrade quality, consistent with the non-negligible final-step anchor-token variation observed in the profiling results (Figure 5c). Results with 5- and 20-timestep update intervals further confirm that a balanced update frequency is important to preserve both trackability and consistency.

I.4 Alternative SeDA Grouping Mechanisms

We further evaluate two simpler SeDA grouping variants on Open-Sora 1.0 with a block size of 32. *Temporal voting* selects a dialect for each spatial position by majority voting across temporal tokens. This removes attention-score computation, but requires online histogram profiling over all temporal tokens. *Global-static* retains anchor-based grouping while applying a calibrated global sub-formatbook to all anchor-correlated groups, thereby eliminating online bin-count profiling for sub-formatbook construction.

Table 18 shows that both temporal voting and global-static maintain broadly comparable VBench quality to the default SeDA, with mixed changes across metrics and modest gains in Dynamic Degree.

Despite similar VBench performance, both alternatives improve FP16-referenced perceptual fidelity (Table 19). Temporal voting reduces both FloLPIPS and FrameLPIPS relative to the default SeDA, while global-static achieves the lowest values for both. This suggests that light sub-formatbook calibration can further improve fidelity with little impact on overall generation quality, whereas our default SeDA prioritizes minimal calibration and online profiling overhead. However, we do not observe the same trend consistently across other model architectures, suggesting that the relative benefit of these grouping strategies may depend on the underlying architecture.

Table 19: FP16-referenced perceptual fidelity of alternative SeDA grouping mechanisms. Lower is better.

Method	FloLPIPS ↓	FrameLPIPS ↓
w/o SeDA	51.53	52.87
w/ SeDA	51.16	52.65
Temporal voting	50.98	52.46
Global-static	**50.57**	**52.04**

Table 20: Comparison between NVFP4 and FB4 (w/o activation decomposition, SeDA) on additional metrics on Open-Sora 1.0. *num_group* is 8 for FB4.

Prompt	Method	Block size	FVD-FP16 (↓)	CLIP-Temp	CLIPSIM	VQA-A	VQA-T	△ Flow(↓)
t2v_sample	NVFP4	16	**1.12**	0.9978	0.1752	50.58	**47.97**	0.48
	FB4	16	1.24	**0.9983**	**0.1781**	**50.72**	47.87	**0.26**
	NVFP4	32	2.11	0.9976	0.1775	49.91	47.43	**0.21**
	FB4	32	**1.59**	**0.9986**	**0.1778**	**50.29**	**47.78**	0.84
UCF-101	NVFP4	16	67.08	0.9971	0.2021	49.69	47.31	3.49
	FB4	16	**61.80**	**0.9976**	**0.2072**	**49.89**	**47.54**	**1.33**
	NVFP4	32	**81.72**	**0.9973**	0.2052	**48.80**	46.93	**2.83**
	FB4	32	85.86	0.9972	**0.2061**	48.77	**46.93**	3.91

J Comparison to FB4 and NVFP4

We compare FB4 against NVFP4 on additional metrics using the t2v_sample and UCF-101 prompt sets in Table 20. Both are evaluated with the same block size and without activation decomposition or SeDA to ensure a fair comparison at the same effective bit-width. Although a block size of 32 on the UCF-101 dataset is less favorable for FB4, it outperforms NVFP4 in most cases.

K Additional Cumulative Impact Analysis

Table 21 presents the cumulative impact of the three proposed techniques on the UCF-101, VBench (`overall_consistency.txt`, 93 prompts), and t2v_sora (48 prompts) prompt sets. FVD-FP16 and flow score generally improve as additional components are incorporated, indicating closer alignment with the FP16 baseline, with activation decomposition contributing the most substantial gains.

L Combination with Another PTQ Method

Table 22 demonstrates additional gains when SemanticDialect is combined with Hadamard rotation [Ashkboos et al., 2024b]. Notably, aesthetic quality, imaging quality, and scene consistency improve significantly. Hadamard rotation redistributes activation magnitudes across channels, mitigating the dominance of outliers within quantization blocks. We hypothesize that while finding an optimal dialect for heavily skewed distributions with outliers is challenging, the less skewed post-rotation distributions are easier to capture with our formatbook, amplifying the benefit of mixed-format quantization. We exclude rotation from SemanticDialect's default configuration to isolate the contribution of our proposed techniques, given that rotation is an independently established PTQ method.

M Broader Impacts

SemanticDialect improves the accessibility of local video generation on edge devices using 4-bit quantization. However, as with all video generation methods, it could potentially lower the barrier to misuse for generating deepfakes or disinformation. Nonetheless, SemanticDialect is a post-training quantization method that does not introduce new generative capabilities beyond those of the base models.

Table 21: Stepwise quality improvement across multiple prompt sets (block size 32).

Dataset	Method	FVD-FP16 (↓)	CLIP-Temp	CLIPSIM	VQA-A	VQA-T	Δ Flow(↓)
UCF-101	FP16	-	0.9973	0.2044	51.11	47.96	-
	FB4	85.86	**0.9972**	**0.2061**	48.77	46.93	3.91
	+Decomp.	**67.04**	0.9972	0.2048	49.74	47.45	2.50
	+SeDA	67.46	0.9971	0.2050	**49.75**	**47.46**	**1.36**
VBench	FP16	-	0.9984	0.1953	51.54	48.15	-
	FB4	62.96	0.9980	**0.1916**	49.48	47.31	2.68
	+Decomp.	46.61	**0.9982**	0.1897	50.43	47.74	1.25
	+SeDA	**44.55**	0.9981	0.1895	**50.47**	**47.77**	**1.00**
t2v_sora	FP16	-	0.9983	0.1955	51.45	48.16	-
	FB4	27.31	0.9980	0.1869	49.37	47.22	4.99
	+Decomp.	16.60	**0.9983**	0.1879	50.31	47.71	3.80
	+SeDA	**16.13**	0.9983	**0.1884**	**50.32**	**47.75**	**2.93**

Table 22: Synergy of rotation-based PTQ methods on Open-Sora 1.0.

Method	Block Size	Aesthetic Quality	Imaging Quality	Motion Smooth.	Dynamic Degree	Subject Consist.	BG. Consist.	Scene Consist.	SS. Consist.
SemanticDialect	16	54.49	63.72	97.66	34.72	93.28	95.86	22.92	23.47
SemanticDialect +Rotation		55.52	64.95	97.40	36.11	93.18	95.51	26.64	23.93
SemanticDialect	32	51.22	60.40	97.61	47.22	91.17	94.98	20.61	22.32
SemanticDialect +Rotation		55.07	62.70	97.31	40.28	92.82	95.65	24.40	23.74

N More Qualitative Results

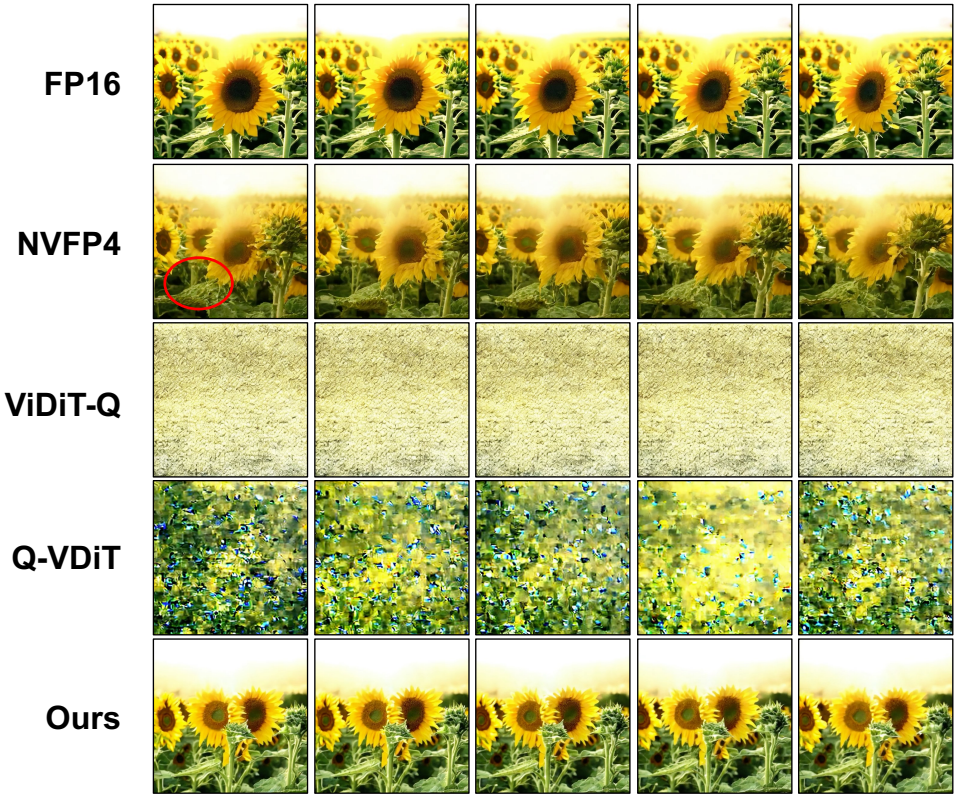


Figure 8: Qualitative results on Open-Sora 1.0 for the prompt: "The vibrant beauty of a sunflower field . . . ". Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.76/4.31. NVFP4 shows noisy leaves whose shapes are difficult to recognize throughout the video and exhibits poor structural consistency. In contrast, SemanticDialect preserves the sunflower and leaf structures more clearly and maintains better visual consistency over time.

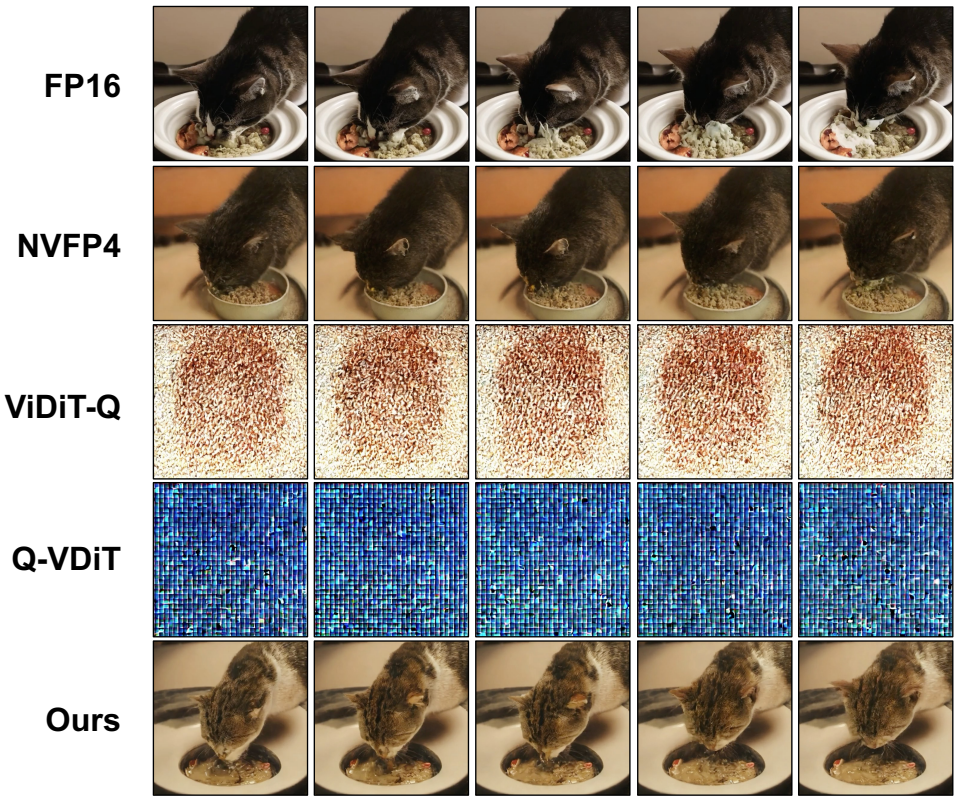


Figure 9: Qualitative results on Open-Sora 1.0 for the prompt: "A cat eating food out of a bowl". Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.76/4.31. NVFP4 frequently produces frames where the cat and food become entangled, with the food in the bowl exhibiting significant noise and flickering artifacts. FP16 also occasionally produces such entangled frames but remains visually sharp overall. In contrast, SemanticDialect maintains more stable cat and food shapes compared to NVFP4 and achieves better imaging quality.



Figure 10: Qualitative results on Open-Sora 1.0 for the prompt: "A vibrant scene of a snowy mountain landscape . . . ". Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.76/4.31. NVFP4 shows a blurred background due to lower imaging quality. In contrast, SemanticDialect preserves clearer background details and more detailed hot-air balloons, resulting in a more visually faithful scene.



Figure 11: Qualitative results on Open-Sora 1.0 for the prompts “A snowy forest landscape with a dirt road . . . ” and “The video captures the majestic beauty of a waterfall . . . ”. Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.76/4.31. SemanticDialect produces higher-quality videos that more closely match FP16 than NVFP4.

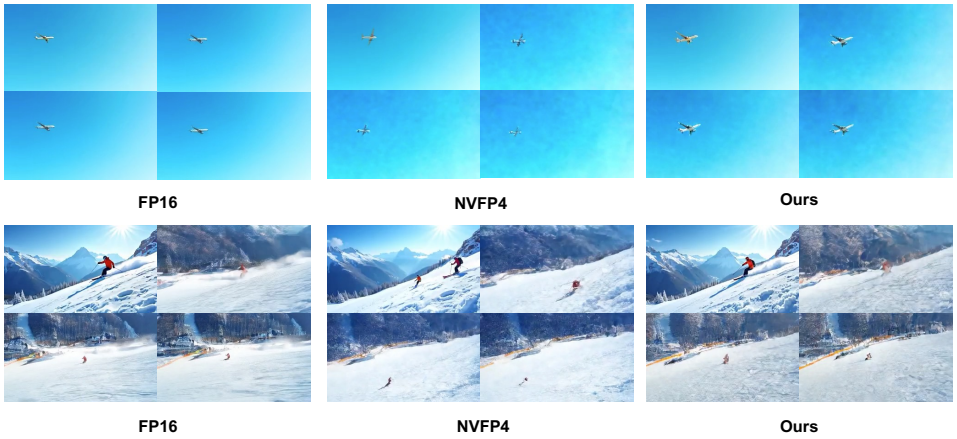


Figure 12: Qualitative results on Open-Sora 2.0 for the prompt: “an airplane soaring through a clear blue sky” and "ski slope". Effective bit-width (A/W): NVFP4 = 4.25/4.25, Ours = 4.31/4.31. SemanticDialect shows better subject quality over NVFP4.



Figure 13: Qualitative results on Open-Sora 2.0 for the prompt: “A woman in a red coat and black scarf walks through a crowded night market. . . ". Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.63/4.63.



Figure 14: Qualitative results on Open-Sora 2.0 for the prompt: "A man in a navy suit gives a presentation in a modern glass-walled office meeting room...". Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.63/4.63.

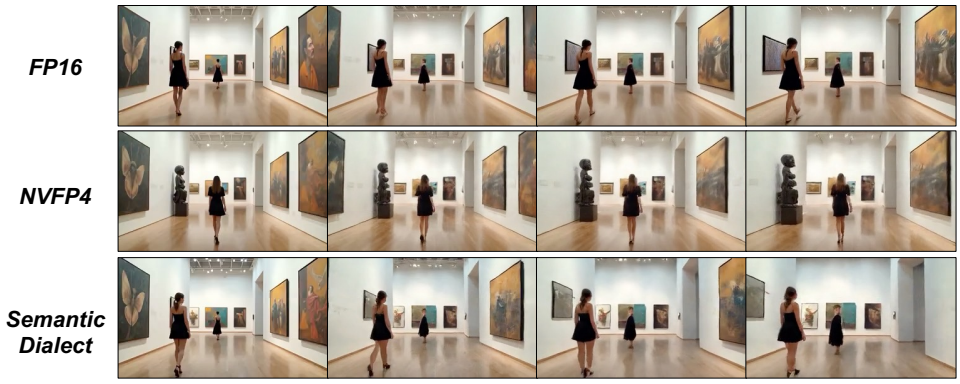


Figure 15: Qualitative results on Open-Sora 2.0 for the prompt: "A woman in a black dress slowly walks through a contemporary art museum gallery with large paintings...". Effective bit-width (A/W): NVFP4 = 4.5/4.5, Ours = 4.63/4.63.

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 three technical contributions (FB4, activation decomposition, and SeDA) and the main claims—outperforming prior quantization methods and block-wise formats, achieving near-FP16 video quality on Open-Sora 2.0, and validating practical deployability through RTL and CUDA kernel implementation—are clearly stated in both the abstract and introduction.

   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: Limitations are discussed in the Conclusion, noting that SeDA's large configuration space may not be globally optimized and offers opportunities for further improvement.

   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: [N/A]

Justification: This paper contains no theoretical results.

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: Detailed instructions for reproducibility are provided in Appendix D. We also provide code in the supplemental material.

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: [Yes]

   Justification: We provide code in the supplemental material with a detailed README describing the environment setup and execution instructions.

   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: Model and dataset details (Section 4, Appendix C), hyperparameters and their selection (Appendices D, H, I) are provided in the paper and released code.

   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: For video generation results, three random seeds are used per prompt to increase sample diversity within a single VBench evaluation, rather than as independent runs; thus per-seed variance is not separately reported. For hardware deployment, peak memory usage shows no variation across runs, and latency standard deviation is less than 0.1% of the mean; both are therefore reported without 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 mention the type and amount of compute (inference time for each run, number of prompts) in Appendices D and C.

   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 have reviewed the NeurIPS Code of Ethics and confirm that this paper conforms with it in all respects.

   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: [Yes]

Justification: Broader impacts are discussed in Appendix M.

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: This paper releases only the quantization code, not new models or datasets.

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: All models, evaluation benchmarks, datasets, and code used in this paper are properly cited. All assets are publicly available under open-source licenses.

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: [Yes]

    Justification: We release the quantization code as a new asset, provided in the supplemental material with a README describing the environment setup and execution instructions.

    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: This 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: This 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.