DyMPQ: Dynamic Mixed-Precision Quantization for Efficient Diffusion Transformers

Anonymous authors
Paper under double-blind review

Abstract

Diffusion transformers (DiTs) have demonstrated strong performance in image and video generation, but their large model sizes and iterative denoising incur substantial computational and memory costs. However, timestep-dependent activation outliers and channel-wise magnitude disparities hinder accurate 4-bit quantization. To address these challenges, we propose Dynamic Channel-wise Mixed-Precision Quantization (DyMPQ) for DiTs. It combines a shared W4A4 main branch with stage-specific high-precision branches to absorb timestep-varying activation outliers. We then apply trajectory-based channel reordering to reduce within-group magnitude disparities in the W4A4 main branch. Building on this mixed-precision representation, we further introduce cross-stage joint calibration to reduce weight quantization error and kernel fusion to enable efficient inference. Extensive experiments on PixArt-Σ, Z-Image, and SANA-Video demonstrate that our method effectively preserves image and video generation quality under low-bit quantization. On an NVIDIA RTX 4090, our method achieves up to $2.24\times$ single-step speedup and reduces peak GPU memory by up to 61.8% on Z-Image compared with BF16.

1 Introduction

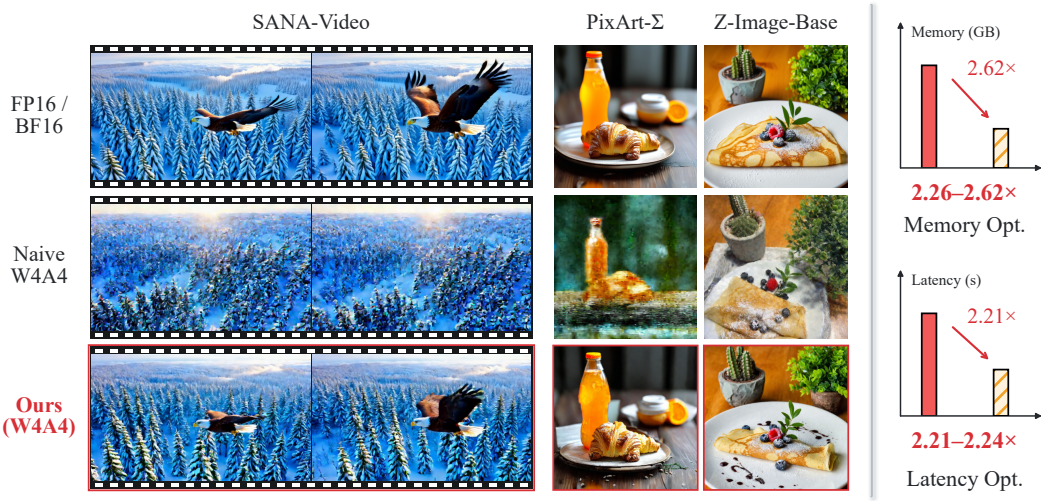


Figure 1: Our method preserves visual quality at W4A4 on SANA-Video, PixArt-Σ, and Z-Image. On Z-Image, it achieves 2.21–$2.24\times$ speedup and reduces peak allocated GPU memory by a factor of 2.26–2.62 relative to BF16.

Transformer-based diffusion models (DiTs) have demonstrated capabilities in generating high-quality images and videos (Bao et al., 2023; Chen et al., 2024b; Kong et al., 2024). However, larger models and higher generation resolutions increase the cost of each forward pass. Iterative denoising

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

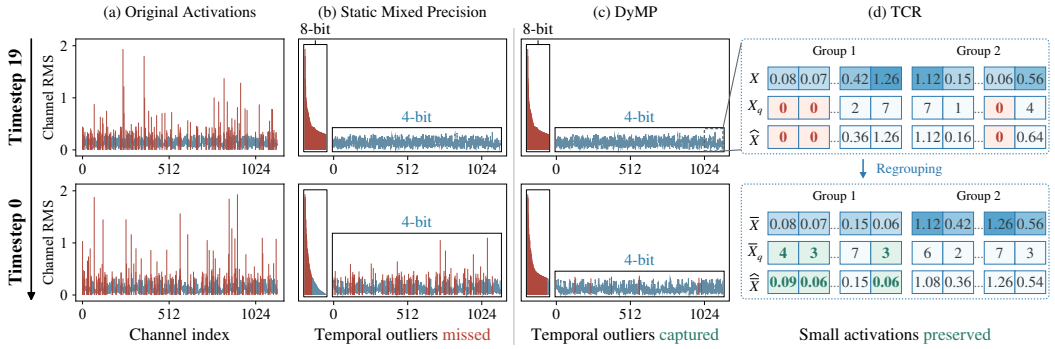


Figure 2: Illustration of dynamic channel-wise mixed precision (DyMP) and trajectory-based channel reordering (TCR). (a) Activation outliers vary across denoising timesteps. (b) Static channel selection misses temporal outliers, whereas (c) DyMP assigns them to the 8-bit branch. (d) TCR groups channels with similar magnitudes to better preserve small activations under 4-bit quantization. X, X_q, and $\widehat{X}$ denote floating-point, integer, and dequantized values, respectively.

further amplifies this cost through repeated model evaluations. For example, Z-Image Base (Z-Image Team, 2025) requires several PFLOPs of computation to generate a single 1024×1024 image. Even with efficient linear attention, SANA-Video (Chen et al., 2025b) takes approximately 71 seconds end to end to generate a five-second 720p video in BF16 on a single RTX 5090. Reducing inference costs is therefore essential for the deployment of diffusion models.

Model quantization reduces the storage and inference costs of DiTs by representing weights and activations at low precision. However, activation outliers and timestep-dependent activation distributions pose challenges for low-bit DiT quantization. One approach mitigates these effects through channel balancing and temporal adaptation. ViDiT-Q (Zhao et al., 2025) combines channel balancing with dynamic activation quantization, while TQ-DiT (Hwang et al., 2025) calibrates activation quantization parameters for different timestep groups. Another approach retains higher precision for sensitive components. SVDQuant (Li et al., 2025) follows this strategy by using a high-precision low-rank branch to absorb outliers transferred from activations to weights. A related strategy has been explored in large language models, where QUIK (Ashkboos et al., 2023) and Atom (Zhao et al., 2024) retain higher precision for selected outlier channels and their corresponding weights while using 4-bit computation for the main workload.

However, the outlier channels that need to retain higher precision in DiTs vary across denoising timesteps. As illustrated in Figure 2, a fixed configuration cannot adapt its channel selection to timestep-varying activation outliers under a limited high-precision channel budget. Moreover, retaining high precision for a small subset of outlier channels does not eliminate magnitude disparities among the remaining low-bit channels. With only a limited number of quantization levels, 4-bit quantization is particularly sensitive to these disparities: larger values expand the shared quantization range, making smaller values difficult to represent accurately. Accurate low-bit DiT quantization therefore requires considering both the temporal variation of outlier channels and the magnitude profiles of the remaining channels along the denoising trajectory to coordinate high-precision channel selection and low-bit grouping.

To address these challenges, we propose a dynamic channel-wise mixed-precision quantization method for DiTs. It combines a shared W4A4 main branch with stage-specific high-precision branches to absorb timestep-varying activation outliers. Within this representation, we introduce trajectory-based channel reordering to reduce within-group magnitude disparities in the W4A4 main branch. We construct a channel permutation shared across stages using channel-wise RMS magnitudes aggregated over the denoising trajectory. We further apply cross-stage joint calibration to reduce weight quantization error. Kernel fusion reduces data movement and kernel launch overhead in activation preparation and mixed-precision computation, enabling efficient inference.

Our main contributions are as follows:

- We introduce a dynamic channel-wise mixed-precision representation that combines a shared W4A4 main branch with stage-specific high-precision branches to accommodate timestep-varying activation outliers under a limited high-precision channel budget.
- We introduce trajectory-based channel reordering to reduce within-group magnitude disparities and improve quantization accuracy in the W4A4 main branch while preserving a shared channel layout across stages.
- We implement efficient mixed-precision inference with fused kernels and conduct extensive experiments on PixArt-Σ, Z-Image, and SANA-Video, demonstrating that our method effectively preserves image and video generation quality. On an NVIDIA RTX 4090, Transformer profiling on Z-Image shows up to $2.24\times$ speedup and up to 61.8% lower peak GPU memory compared with BF16.

2 Related Work

Diffusion transformers. Diffusion transformers (DiTs) replace U-Net backbones with transformers to capture long-range spatial dependencies and improve generation quality through model scaling (Peebles & Xie, 2023; Bao et al., 2023; Gao et al., 2023; Ma et al., 2024; Esser et al., 2024). However, iterative denoising requires repeated evaluations of these large models, resulting in substantial inference costs. Existing acceleration methods therefore follow two complementary directions: reducing the number of denoising steps and lowering the cost of each step. Fast samplers (Song et al., 2021; Liu et al., 2022; Lu et al., 2022; 2025; Zhao et al., 2023) and distillation methods (Salimans & Ho, 2022; Luo et al., 2023; Sauer et al., 2024; Yin et al., 2024b;a) reduce the number of model evaluations. Other approaches improve per-step efficiency through quantization (Li et al., 2023; He et al., 2023; 2024), sparse computation (Xi et al., 2025; Zhang et al., 2025c; Yang et al., 2025; Zhang et al., 2025b), and efficient architectures (Xie et al., 2024; Zhuo et al., 2024; Tian et al., 2024; Chen et al., 2025a; Becker et al., 2025). Our work reduces per-step computation and memory costs through low-bit quantization and also applies to few-step diffusion models.

Diffusion model quantization. Post-training quantization (PTQ) is an effective approach to reducing the storage and computational costs of DiTs. PTQ4DiT and ViDiT-Q mitigate activation outliers through channel balancing (Wu et al., 2024; Zhao et al., 2025), while SVDQuant absorbs outliers transferred to weights with a high-precision low-rank branch (Li et al., 2025). For group-wise quantization, Q-DiT adopts fine-grained grouping (Chen et al., 2025c), while PermuQuant improves channel grouping through reordering based on joint activation and weight second-moment statistics (Cheng et al., 2026). ViDiT-Q and Q-DiT additionally employ dynamic activation quantization, while TQ-DiT explicitly accounts for temporal variation by calibrating activation quantization parameters for different timestep groups (Hwang et al., 2025). Our method jointly addresses temporal variation and channel grouping. Stage-specific high-precision branches absorb timestep-varying activation outliers, while trajectory-based channel reordering reduces magnitude disparities within low-bit quantization groups.

Channel-wise mixed-precision quantization. Channel-wise mixed-precision quantization retains higher precision for a small subset of outlier channels to improve low-bit quantization accuracy (Lee et al., 2024; Chen et al., 2026). This strategy has been adopted in large language models (LLMs). LLM.int8() isolates outlier feature dimensions for 16-bit computation while processing the remaining dimensions in 8 bits (Dettmers et al., 2022). QUIK and Atom retain higher precision for selected outlier channels while quantizing most weights and activations to 4 bits (Ashkboos et al., 2023; Zhao et al., 2024). Since activation distributions in DiTs vary across denoising timesteps (Zhao et al., 2025), a fixed high-precision channel configuration may not suit all stages. Our method therefore uses stage-specific high-precision branches to absorb activation outliers that vary across denoising timesteps.

3 Quantization Analysis

Consider a linear layer in a DiT at denoising step t, with output $\boldsymbol{Y}_t = \boldsymbol{X}_t\boldsymbol{W}^\top$, where $\boldsymbol{X}_t \in \mathbb{R}^{N_t\times D}$ denotes the input activations and $\boldsymbol{W} \in \mathbb{R}^{O\times D}$ denotes the weights. We use group-wise

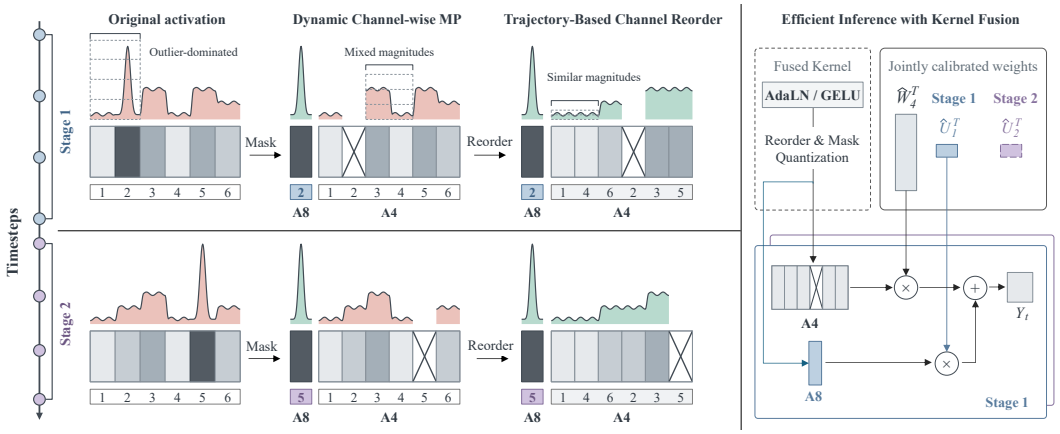


Figure 3: Overview of dynamic channel-wise mixed-precision quantization framework.

symmetric quantization: each row of the activation and weight matrices is partitioned into groups along the input-channel dimension, and each group is quantized independently. For a vector $\boldsymbol{x}$ within a nonzero quantization group, its b-bit quantization is defined as

$$Q_b(\boldsymbol{x}) = s_{\boldsymbol{x}} \operatorname{round}\left(\frac{\boldsymbol{x}}{s_{\boldsymbol{x}}}\right), \qquad s_{\boldsymbol{x}} = \frac{\max(|\boldsymbol{x}|)}{2^{b-1}-1},$$

where Q_b returns dequantized values. Let $\widehat{\boldsymbol{X}}_t$ denote the dequantized activations aligned with the original channel order, and let $\widehat{\boldsymbol{Y}}_t$ denote the actual quantized output. The output error satisfies

$$\boldsymbol{Y}_t - \widehat{\boldsymbol{Y}}_t = (\boldsymbol{X}_t - \widehat{\boldsymbol{X}}_t)\boldsymbol{W}^\top + \big(\widehat{\boldsymbol{X}}_t\boldsymbol{W}^\top - \widehat{\boldsymbol{Y}}_t\big). \tag{1}$$

The first term captures the output error introduced by activation quantization, while the second captures the weight representation error for the given quantized activations. We first analyze activation quantization, whose accuracy is affected by the following two interrelated factors.

Timestep-varying activation outliers. As shown in Figure 2(a), the channels containing large activations and their relative magnitudes vary across denoising timesteps. With a fixed high-precision channel configuration, some large-magnitude channels remain in the 4-bit region during certain stages, increasing the variation in activation magnitudes within a group and making low-bit quantization more difficult (Figure 2(b)).

Grouping sensitivity of low-bit activations. Even when outlier channels are assigned to a high-precision branch, magnitude disparities remain among the low-bit channels. With limited quantization levels in 4-bit quantization, large-magnitude entries within a group can increase the shared step size, causing smaller values to be coarsely represented or even rounded to zero. Figure 2(d) shows that the same floating-point values produce different quantization results under different groupings. This error depends on both the channel magnitudes and which channels share a scale.

The above analysis indicates that activation quantization needs to account for both the temporal variation of outlier channels and the grouping of the remaining low-bit channels. According to Eq. (1), improving the activation representation does not necessarily eliminate the output error introduced by weight quantization. Once the mixed-precision activation representation is fixed, further weight calibration is therefore needed to reduce output reconstruction error.

4 METHOD

We propose a dynamic channel-wise mixed-precision quantization method for DiTs, as illustrated in Figure 3. We first introduce a dynamic channel-wise mixed-precision representation, assigning activation channels to low-bit and high-precision branches and adapting the high-precision channel

assignment across denoising stages to absorb timestep-varying activation outliers. We then apply trajectory-based channel reordering to improve the grouping of low-bit channels. Based on this representation, we introduce cross-stage joint calibration to calibrate the weights of both branches, allowing the high-precision branch to also compensate for quantization errors in the low-bit weights. Finally, we enable efficient inference through kernel fusion.

4.1 Dynamic Channel-wise Mixed-Precision Representation (DyMP)

We introduce a dynamic channel-wise mixed-precision representation (DyMP), which partitions the T denoising steps into G contiguous stages $\{\mathcal{T}_g\}_{g=1}^{G}$ in execution order and combines a shared W4A4 main branch with stage-specific W8A8 branches to absorb timestep-varying activation outliers while controlling branch storage and calibration costs. For a given layer, let K denote the number of high-precision channels and $\boldsymbol{X}_t \in \mathbb{R}^{N_t \times D}$ the calibration activations at timestep t. We compute the channel-wise root mean square (RMS) magnitudes $r_{g,c}$ within each stage and select the K channels with the largest magnitudes, whose index set is denoted by $\mathcal{I}_g$:

$$r_{g,c} = \sqrt{\frac{\sum_{t\in\mathcal{T}_g} \|\boldsymbol{X}_t[:,c]\|_2^2}{\sum_{t\in\mathcal{T}_g} N_t}}, \qquad \mathcal{I}_g = \mathrm{TopK}_c(r_{g,c}). \tag{2}$$

For $t \in \mathcal{T}_g$, we zero the input positions indexed by $\mathcal{I}_g$ before computing the low-bit quantization scales and separately quantize these channels to A8. The low-bit input retains the full channel layout, with high-precision channels represented by zeros in this branch. Let $\boldsymbol{m}_g$ be the binary mask corresponding to $\mathcal{I}_g$, and let Q_b denote group-wise b-bit quantization followed by dequantization. We denote the shared W4 weights by $\widehat{\boldsymbol{W}}_4 \in \mathbb{R}^{O\times D}$ and the W8 weights for stage g by $\widehat{\boldsymbol{U}}_g \in \mathbb{R}^{O\times K}$. The two branch inputs and their combined output are

$$\boldsymbol{B}_t = Q_4(\boldsymbol{X}_t \odot (\mathbf{1} - \boldsymbol{m}_g)), \quad \boldsymbol{Z}_t = Q_8(\boldsymbol{X}_t[:,\mathcal{I}_g]), \quad \widehat{\boldsymbol{Y}}_t = \boldsymbol{B}_t\widehat{\boldsymbol{W}}_4^\top + \boldsymbol{Z}_t\widehat{\boldsymbol{U}}_g^\top. \tag{3}$$

Stage boundaries are determined during offline calibration. For each candidate contiguous interval, we select high-precision channels using the RMS criterion above and evaluate the resulting mixed-precision activation reconstruction error. We then use dynamic programming to find the partition that minimizes the total normalized reconstruction error. The cost definition and search procedure are detailed in Appendix B.3.

4.2 Trajectory-Based Channel Reordering

To improve channel grouping in the W4A4 main branch, we introduce trajectory-based channel reordering (TCR), which constructs a channel permutation shared across stages to reduce within-group magnitude disparities. We reuse the RMS definition in Eq. (2), aggregating the calibration activations over the entire denoising trajectory. Denoting the resulting channel magnitude by r_c^{traj}, we sort the channels in ascending order:

$$\boldsymbol{\pi} = \mathrm{argsort}_c\big(r_c^{\mathrm{traj}}\big). \tag{4}$$

Using a permutation matrix $\boldsymbol{P}$ that remains fixed across stages, we reorder the columns of the original weights $\boldsymbol{W}$ offline and map the high-precision channel indices to $\bar{\mathcal{I}}_g$, eliminating weight reordering during inference. Input activations use the same permutation, with activation reordering fused into preceding operators as described in Sec. 4.4. We then construct the mixed-precision representation in Sec. 4.1 using the reordered layout. The permutation preserves the floating-point linear mapping:

$$\bar{\boldsymbol{X}}_t = \boldsymbol{X}_t\boldsymbol{P}, \qquad \bar{\boldsymbol{W}} = \boldsymbol{W}\boldsymbol{P}, \qquad \bar{\boldsymbol{X}}_t\bar{\boldsymbol{W}}^\top = \boldsymbol{X}_t\boldsymbol{W}^\top. \tag{5}$$

4.3 Cross-Stage Joint Calibration

In the mixed-precision representation above, different stages have different quantized inputs and selected high-precision channels, while all stages share the same W4 weights. Calibrating the shared weights therefore requires considering the inputs and W8 branches of all stages. We introduce cross-stage joint calibration (CJC) to calibrate both branches under a unified reconstruction objective,

allowing the high-precision branches to perform selected-channel computation while also compensating for quantization errors in the low-bit weights. Under group-wise W4 and W8 quantization constraints, the joint calibration objective is

$$\min_{\widehat{\boldsymbol{W}}_4,\{\widehat{\boldsymbol{U}}_g\}_{g=1}^{G}} \sum_{g=1}^{G}\sum_{t\in\mathcal{T}_g} \left\| \boldsymbol{Y}_t^{\mathrm{ref}} - \boldsymbol{B}_t\widehat{\boldsymbol{W}}_4^{\top} - \boldsymbol{Z}_t\widehat{\boldsymbol{U}}_g^{\top} \right\|_F^2, \quad \boldsymbol{Y}_t^{\mathrm{ref}} = \boldsymbol{B}_t\bar{\boldsymbol{W}}^{\top} + \boldsymbol{Z}_t\bar{\boldsymbol{W}}[:,\bar{\mathcal{I}}_g]^{\top}. \tag{6}$$

This objective measures the layer output error introduced by weight quantization for the given mixed-precision activations. To jointly optimize both branches, we stack $\boldsymbol{B}_t$ and $\boldsymbol{Z}_t$ row-wise within each stage into $\boldsymbol{B}^{(g)}$ and $\boldsymbol{Z}^{(g)}$, respectively, and construct the augmented linear representation:

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{B}^{(1)} & \boldsymbol{Z}^{(1)} & \mathbf{0} & \cdots & \mathbf{0} \\ \boldsymbol{B}^{(2)} & \mathbf{0} & \boldsymbol{Z}^{(2)} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{B}^{(G)} & \mathbf{0} & \mathbf{0} & \cdots & \boldsymbol{Z}^{(G)} \end{bmatrix}, \qquad \widehat{\boldsymbol{\Theta}}^{\top} = \begin{bmatrix} \widehat{\boldsymbol{W}}_4^{\top} \\ \widehat{\boldsymbol{U}}_1^{\top} \\ \widehat{\boldsymbol{U}}_2^{\top} \\ \vdots \\ \widehat{\boldsymbol{U}}_G^{\top} \end{bmatrix}. \tag{7}$$

Stacking the outputs in the same order gives $\widehat{\boldsymbol{Y}} = \boldsymbol{\Phi}\widehat{\boldsymbol{\Theta}}^{\top}$. The first block column collects the low-bit inputs across stages, jointly constraining the shared weights $\widehat{\boldsymbol{W}}_4$, while the remaining block columns correspond to the stage-specific W8 branches. We apply GPTQ's second-order error compensation to this augmented representation (Frantar et al., 2023). We initialize the shared weights with $\bar{\boldsymbol{W}}$ and each stage-specific branch with $\bar{\boldsymbol{W}}[:,\bar{\mathcal{I}}_g]$. Using the second-order statistics of the augmented inputs, we quantize the W4 branch column by column. At each step, we update both the remaining unquantized shared columns and the stage-specific W8 weights to compensate for the output error introduced by quantization. Once the W4 branch has been quantized, we quantize the compensated stage-specific weights to W8 using round-to-nearest quantization.

4.4 Efficient Inference with Kernel Fusion

We pack the shared W4 weights offline and select the channel indices and W8 weights corresponding to the current denoising stage during inference. To reduce activation preparation overhead, we fuse channel reordering, masking, quantization scale computation, and INT4/INT8 packing into preceding operators such as AdaLayerNorm and GELU. These fused operators directly produce low-bit inputs for both branches, reducing separate kernel launches and intermediate tensor reads and writes. Following Atom's mixed-precision GEMM design, we execute the W4A4 and W8A8 branches in a single kernel. Both branches perform integer multiply-accumulate operations within each quantization group. At each group boundary, we rescale the INT32 partial sums using the corresponding activation and weight scales and accumulate the results into shared FP32 accumulators to produce the final output.

5 Experiments

5.1 Experimental Setup

Models and Evaluation Setup. We evaluate three text-to-image models, PixArt-Σ, Z-Image-Turbo, and Z-Image Base, and one text-to-video model, SANA-Video (Chen et al., 2024a; Z-Image Team, 2025; Chen et al., 2025b). For PixArt-Σ, we use the first 1k COCO captions (Chen et al., 2015) and report CLIP (Radford et al., 2021), ImageReward (Xu et al., 2023), Aesthetic, and LPIPS (Zhang et al., 2018). For Z-Image-Turbo and Base, we report GenEval (Ghosh et al., 2023) and PRISM-EN (Fang et al., 2025) scores for compositional alignment, instruction following, and visual quality. For SANA-Video, we use VBench (Huang et al., 2024), following prior evaluations of quantized diffusion transformers (Chen et al., 2025c; Zhao et al., 2025). We report imaging quality, aesthetic quality, overall consistency, motion smoothness, dynamic degree, background consistency, subject consistency, and scene consistency. Model and generation settings are provided in Appendix A.

Method	Precision (W/A)	LPIPS ↓	CLIP ↑	IR ↑	Aes. ↑
FP16	16/16	—	0.264	0.917	5.99
Q-DiT	4/4	0.83	0.192	-1.92	4.23
ViDiT-Q	4/8	0.46	0.262	0.798	5.93
SVDQuant	4/4	0.38	**0.264**	0.879	5.91
ConvRot	4/4	0.51	**0.264**	0.864	5.90
Ours	4/4	**0.35**	**0.264**	**0.915**	**5.94**

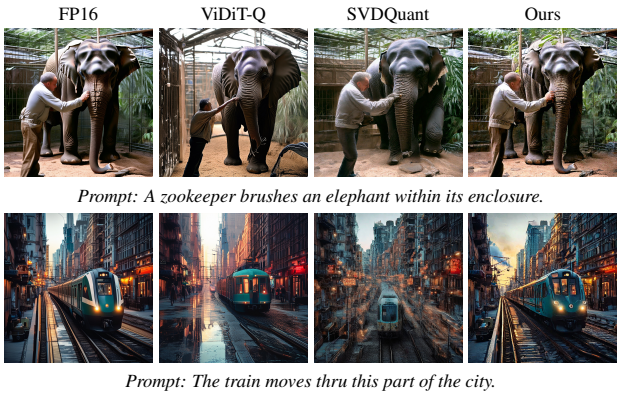


Figure 4: Quantization results on PixArt-Σ. Left: quality metrics; right: qualitative comparisons. Best and second-best results among quantized methods are shown in **bold** and underline, respectively. Enlarged comparisons with additional methods and prompts are provided in Appendix C.2.

Table 1: Generation quality on Z-Image-Turbo and Z-Image Base. Best and second-best quantized results are shown in **bold** and underline, respectively. For GenEval, Single, Two, Count, Color, Pos., and Attr. denote single object, two objects, counting, color, position, and color attribution, respectively. Full PRISM-EN results are in Appendix C.1.

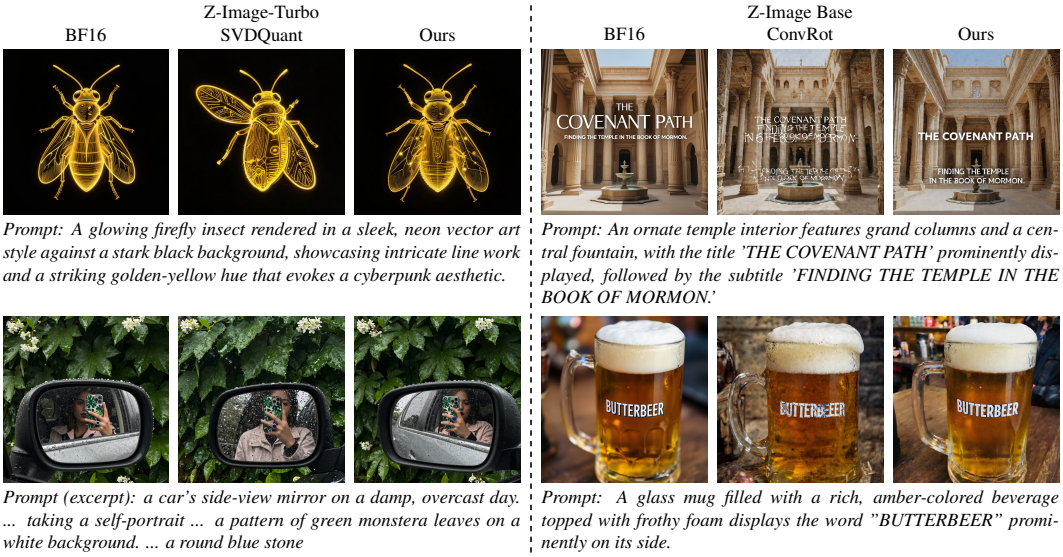
Model	Method	Precision (W/A)	GenEval							PRISM-EN		
			Single ↑	Two ↑	Count ↑	Color ↑	Pos. ↑	Attr. ↑	Overall ↑	Style	Long Text	Overall ↑
Z-Image Turbo (8 steps)	BF16	16/16	100.00	90.15	74.06	84.04	46.00	58.00	75.37	5.87	5.31	5.74
	PTQ4DiT	4/4	0.000	0.000	0.300	0.000	0.000	0.000	0.050	—	—	—
	Q-DiT	4/4	98.75	88.13	72.50	79.79	48.25	56.00	73.90	5.79	5.51	5.81
	SVDQuant	4/4	**99.69**	90.91	69.06	83.51	50.50	**62.00**	75.94	5.64	5.43	5.70
	ConvRot	4/4	99.38	88.64	**74.06**	81.91	49.25	59.75	75.49	—	—	—
	Ours	4/4	**99.69**	**91.92**	71.25	**84.31**	**51.00**	59.50	**76.28**	**5.92**	**5.56**	**5.84**
Z-Image Base (50 steps)	BF16	16/16	98.75	90.90	73.75	87.77	40.50	61.50	75.53	6.01	5.13	5.55
	PTQ4DiT	4/4	0.000	0.000	0.000	0.000	0.000	0.000	0.000	—	—	—
	Q-DiT	4/4	82.81	53.79	43.12	60.64	23.75	35.00	49.85	4.32	3.86	3.92
	ConvRot	4/4	96.25	88.89	**74.25**	73.40	38.00	50.00	70.13	—	—	—
	Ours	4/4	**98.75**	**91.16**	72.81	**86.97**	**41.75**	**61.00**	**75.40**	**6.13**	**5.56**	**5.72**

Baselines. We compare against five representative post-training quantization (PTQ) methods. PTQ4DiT (Wu et al., 2024) uses channel-wise salience balancing to smooth salient channels in activations and weights, together with temporal salience calibration. Q-DiT (Chen et al., 2025c) employs fine-grained grouping and dynamic activation quantization. ViDiT-Q (Zhao et al., 2025) combines channel scaling and rotation with dynamic activation quantization. SVDQuant (Li et al., 2025) transfers activation outliers to the weights and absorbs the resulting weight outliers with a high-precision low-rank branch. ConvRot (Huang et al., 2025) applies group-wise Hadamard rotation for W4A4 quantization.

Implementation Details. We apply our dynamic mixed-precision quantization to the weights and input activations of transformer linear layers. Implementation details are provided in Appendix B.

5.2 Generation Quality

Image generation on PixArt-Σ. As shown in Figure 4, our method closely preserves FP16 generation quality, achieving an ImageReward score of 0.915 compared with 0.917 for FP16. Relative to SVDQuant, it reduces LPIPS from 0.38 to 0.35 and improves ImageReward from 0.879 to 0.915, indicating better preservation of the floating-point outputs and perceptual quality.



Prompt: A glowing firefly insect rendered in a sleek, neon vector art style against a stark black background, showcasing intricate line work and a striking golden-yellow hue that evokes a cyberpunk aesthetic.

Prompt: An ornate temple interior features grand columns and a central fountain, with the title 'THE COVENANT PATH' prominently displayed, followed by the subtitle 'FINDING THE TEMPLE IN THE BOOK OF MORMON.'

Prompt (excerpt): a car's side-view mirror on a damp, overcast day. ... taking a self-portrait ... a pattern of green monstera leaves on a white background. ... a round blue stone

Prompt: A glass mug filled with a rich, amber-colored beverage topped with frothy foam displays the word "BUTTERBEER" prominently on its side.

Figure 5: Qualitative comparisons on Z-Image-Turbo (left) and Z-Image Base (right).

Table 2: Generation quality on SANA-Video across eight VBench dimensions. Best and second-best results among quantized methods are shown in **bold** and underline, respectively.

Method	Precision (W/A)	Imaging Quality	Aesthetic Quality	Overall Consistency	Motion Smooth.	Dynamic Degree	BG. Consist.	Subject Consist.	Scene Consist.
BF16	16/16	68.33	67.97	27.18	98.27	68.89	95.82	96.24	57.21
PTQ4DiT	4/4	56.52	46.31	19.43	94.31	9.72	94.54	91.62	18.75
Q-DiT	4/4	63.96	61.20	25.81	96.80	45.56	**95.41**	91.93	52.33
SVDQuant	4/4	67.71	67.12	27.00	97.50	61.11	95.24	95.51	56.76
ConvRot	4/4	66.31	65.84	27.31	96.74	**62.50**	94.94	93.99	58.64
Ours	4/4	**67.92**	**67.42**	**27.32**	**98.07**	61.11	94.63	**95.65**	**59.88**

Image generation on Z-Image. Table 1 summarizes the results on Z-Image-Turbo and Base. On Z-Image-Turbo, our method achieves the highest GenEval Overall score of 76.28% among the reported quantized methods, demonstrating its applicability to few-step distilled models. The advantage is more pronounced on Z-Image Base: under 50-step sampling, our method achieves an Overall score of 75.40%, only 0.13 percentage points below BF16, while outperforming ConvRot and Q-DiT by 5.27 and 25.55 percentage points, respectively.

Video generation on SANA-Video. As shown in Table 2, our method achieves the best results among the reported quantized methods in six of the eight VBench dimensions, covering visual quality, motion smoothness, and consistency. These results demonstrate that our quantization scheme also effectively preserves video generation quality.

5.3 Inference Efficiency and Resource Usage

We evaluate the single-step inference efficiency and resource usage of Z-Image-Turbo ($G = 2$) and Base ($G = 6$) on an NVIDIA RTX 4090. As shown in Figure 6, compared with BF16, our method reduces linear-layer weight storage, lowers peak GPU memory usage by 55.8–61.8%, and achieves 2.21–$2.24\times$ single-step speedups. Although Base stores more stage-specific W8 weights, each step executes only the current stage's W8 branch, yielding speedups similar to Turbo. Adding SageAttention (Zhang et al., 2025a) increases the speedup to $2.53\times$ on both models. As shown in Figure 7, our method adds only approximately 3% single-step latency compared with naive W4A4. This is because kernel fusion reduces memory traffic and kernel launches for channel reordering and activation quantization.

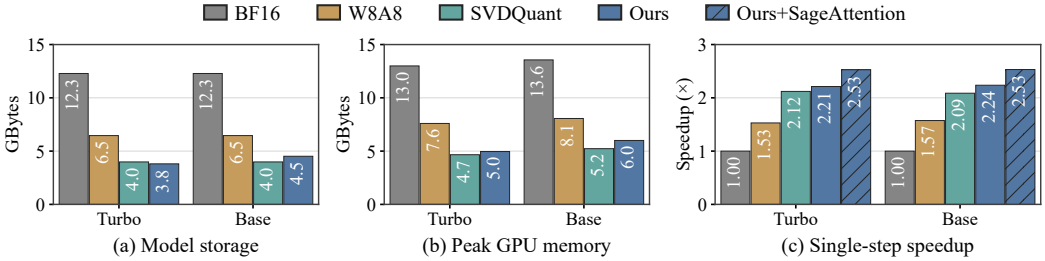


Figure 6: Resource usage and single-step speedup on Z-Image-Turbo and Z-Image Base.

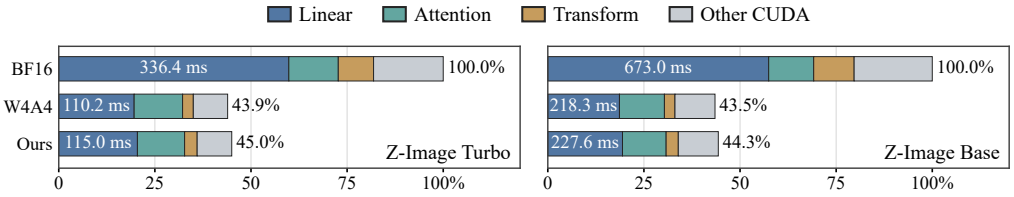


Figure 7: Single-step latency breakdown on Z-Image-Turbo and Z-Image Base.

5.4 Component Ablations

Table 3 shows that naive W4A4 and TCR alone severely degrade generation quality on PixArt-Σ. With TCR alone, some channels may still share quantization scales with extreme outliers, limiting their effective quantization resolution. DyMP alone achieves an ImageReward score of -0.029, while combining it with TCR raises the score to 0.881, demonstrating their complementary roles. Under the same per-step high-precision channel budget, replacing static ($G = 1$) with dynamic ($G = 2$) channel assignment improves ImageReward from 0.845 to 0.881, highlighting the importance of adapting channel selection across denoising stages. Finally, CJC reduces LPIPS from 0.39 to 0.35 and improves ImageReward to 0.915, approaching the FP16 score of 0.917.

Table 3: Component ablation on PixArt-Σ. "+" denotes additions to Naive W4A4.

Configuration	LPIPS ↓	ImageReward ↑	Aesthetic ↑
FP16	—	0.917	5.99
Naive W4A4	0.85	-2.010	4.21
+ TCR	0.84	-2.264	3.49
+ DyMP	0.63	-0.029	5.51
+ TCR + Static MP	0.48	0.845	5.86
+ TCR + DyMP	0.39	0.881	5.90
+ TCR + DyMP + CJC (Full method)	0.35	0.915	5.94

6 Conclusion

We presented DyMPQ, a dynamic mixed-precision quantization method for diffusion transformers. By combining a shared W4A4 branch with stage-specific high-precision branches, our method dynamically handles timestep-varying activation outliers. Trajectory-based channel reordering improves low-bit channel grouping, while cross-stage joint calibration enables high-precision branches to compensate for quantization errors in the shared weights. Efficient kernel design enables these stage-specific configurations to execute with minimal overhead. Experiments on image and video generation demonstrate that DyMPQ preserves generation quality while reducing storage, peak GPU memory, and inference latency.

AI USE STATEMENT

In this work, we used OpenAI Codex and other generative AI tools to assist with code implementation and debugging, manuscript drafting and language refinement, feedback on the presentation of research methodology and experimental results, and the preparation of figures, tables, and mathematical expressions. The authors reviewed all AI-assisted content, checked mathematical statements and descriptions of experimental results against the underlying analyses and experimental outputs, and inspected and tested AI-assisted code. All final decisions regarding the methodology, interpretation of results, and conclusions were made by the authors. The authors take full responsibility for the final content of this work, including all text, claims, code, and figures.

REFERENCES

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. arXiv preprint arXiv:2310.09259, 2023. URL https://arxiv.org/abs/2310.09259.

Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All Are Worth Words: A ViT Backbone for Diffusion Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. URL https://arxiv.org/abs/2209.12152.

Philipp Becker, Abhinav Mehrotra, Ruchika Chavhan, Malcolm Chadwick, Luca Morreale, Mehdi Noroozi, Alberto Gil C. P. Ramos, and Sourav Bhattacharya. EDiT: Efficient Diffusion Transformers with Linear Compressed Attention. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19608–19616, 2025. URL https://openaccess.thecvf.com/content/ICCV2025/html/Becker_EDiT_Efficient_Diffusion_Transformers_with_Linear_Compressed_Attention_ICCV_2025_paper.html.

Chen Chen, Rui Qian, Wenze Hu, Tsu-Jui Fu, Jialing Tong, Xinze Wang, Lezhi Li, Bowen Zhang, Alex Schwing, Wei Liu, and Yinfei Yang. DiT-Air: Revisiting the Efficiency of Diffusion Model Architecture Design in Text to Image Generation. arXiv preprint arXiv:2503.10618, 2025a. URL https://arxiv.org/abs/2503.10618.

Junsong Chen, Chongjian Ge, Enze Xie, Yue Wu, Lewei Yao, Xiaozhe Ren, Zhongdao Wang, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-Σ: Weak-to-strong training of diffusion transformer for 4k text-to-image generation. arXiv preprint arXiv:2403.04692, 2024a. URL https://arxiv.org/abs/2403.04692.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. PixArt-α: Fast Training of Diffusion Transformer for Photorealistic Text-to-Image Synthesis. In *International Conference on Learning Representations*, 2024b. URL https://arxiv.org/abs/2310.00426.

Junsong Chen, Yuyang Zhao, Jincheng Yu, Ruihang Chu, Junyu Chen, Shuai Yang, Xianbang Wang, Yicheng Pan, Daquan Zhou, Huan Ling, Haozhe Liu, Hongwei Yi, Hao Zhang, Muyang Li, Yukang Chen, Han Cai, Sanja Fidler, Ping Luo, Song Han, and Enze Xie. SANA-Video: Efficient video generation with block linear diffusion transformer. arXiv preprint arXiv:2509.24695, 2025b. URL https://arxiv.org/abs/2509.24695.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025c. URL https://arxiv.org/abs/2406.17343.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollar, and C. Lawrence Zitnick. Microsoft COCO Captions: Data Collection and Evaluation Server. arXiv preprint arXiv:1504.00325, 2015. URL https://arxiv.org/abs/1504.00325.

Zihan Chen, Bike Xie, Jundong Li, and Cong Shen. Adapting to Any Bit-Width: Channel-Wise Mixed-Precision Quantization for LLMs. *Transactions on Machine Learning Research*, 2026. URL https://openreview.net/forum?id=1t6sEhdLxf.

Yongsen Cheng, Kai Liu, Kaiwen Tao, Junxian Li, Zhixin Wang, Zhikai Chen, Renjing Pei, and Yulun Zhang. PermuQuant: Lowering Per-Group Quantization Error by Reordering Channels for Diffusion Models. arXiv preprint arXiv:2605.09503, 2026. URL https://arxiv.org/abs/2605.09503.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. LLM.int8(): 8-bit Matrix Multiplication for Transformers at Scale. In *Advances in Neural Information Processing Systems*, 2022. URL https://arxiv.org/abs/2208.07339.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 12606–12633, 2024. URL https://proceedings.mlr.press/v235/esser24a.html.

Rongyao Fang, Aldrich Yu, Chengqi Duan, Linjiang Huang, Shuai Bai, Yuxuan Cai, Kun Wang, Si Liu, Xihui Liu, and Hongsheng Li. FLUX-Reason-6M & PRISM-Bench: A million-scale text-to-image reasoning dataset and comprehensive benchmark. arXiv preprint arXiv:2509.09680, 2025. URL https://arxiv.org/abs/2509.09680.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate Post-Training Quantization for Generative Pre-trained Transformers. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2210.17323.

Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Masked Diffusion Transformer is a Strong Image Synthesizer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 23164–23173, 2023. URL https://openaccess.thecvf.com/content/ICCV2023/html/Gao_Masked_Diffusion_Transformer_is_a_Strong_Image_Synthesizer_ICCV_2023_paper.html.

Dhruba Ghosh, Hanna Hajishirzi, and Ludwig Schmidt. GenEval: An object-focused framework for evaluating text-to-image alignment. arXiv preprint arXiv:2310.11513, 2023. URL https://arxiv.org/abs/2310.11513.

Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. PTQD: Accurate Post-Training Quantization for Diffusion Models. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2305.10657.

Yefei He, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. EfficientDM: Efficient Quantization-Aware Fine-Tuning of Low-Bit Diffusion Models. In *International Conference on Learning Representations*, 2024. URL https://arxiv.org/abs/2310.03270.

Feice Huang, Zuliang Han, Xing Zhou, Yihuang Chen, Lifei Zhu, and Haoqian Wang. ConvRot: Rotation-Based Plug-and-Play 4-bit Quantization for Diffusion Transformers. arXiv preprint arXiv:2512.03673, 2025. URL https://arxiv.org/abs/2512.03673.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, Yaohui Wang, Xinyuan Chen, Limin Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. VBench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. URL https://openaccess.thecvf.com/content/CVPR2024/html/Huang_VBench_Comprehensive_Benchmark_Suite_for_Video_Generative_Models_CVPR_2024_paper.html.

Younghye Hwang, Hyojin Lee, and Joonhyuk Kang. TQ-DiT: Efficient Time-Aware Quantization for Diffusion Transformers. arXiv preprint arXiv:2502.04056, 2025. URL https://arxiv.org/abs/2502.04056.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, Kathrina Wu, Qin Lin, Junkun Yuan, Yanxin Long, Aladdin Wang, Andong Wang, Changlin Li, Duojun Huang, Fang Yang, Hao Tan, Hongmei Wang, Jacob Song, Jiawang Bai, Jianbing Wu, Jinbao Xue, Joey Wang, Kai Wang, Mengyang Liu, Pengyu Li, Shuai Li, Weiyan Wang, Wenqing Yu, Xinchi Deng, Yang Li, Yi Chen, Yutao Cui, Yuanbo Peng, Zhentao Yu, Zhiyu He, Zhiyong Xu, Zixiang Zhou, Zunnan Xu, Yangyu Tao, Qinglin Lu, Songtao Liu, Dax Zhou, Hongfa Wang, Yong Yang, Di Wang, Yuhong Liu, Jie Jiang, and Caesar Zhong. HunyuanVideo: A Systematic Framework for Large Video Generative Models. arXiv preprint arXiv:2412.03603, 2024. URL https://arxiv.org/abs/2412.03603.

Changhun Lee, Jungyu Jin, Taesu Kim, Hyungjun Kim, and Eunhyeok Park. OWQ: Outlier-Aware Weight Quantization for Efficient Fine-Tuning and Inference of Large Language Models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13355–13364, 2024. doi: 10.1609/aaai.v38i12.29237. URL https://ojs.aaai.org/index.php/AAAI/article/view/29237.

Muyang Li, Yujun Lin, Zhekai Zhang, Tianle Cai, Xiuyu Li, Junxian Guo, Enze Xie, Chenlin Meng, Jun-Yan Zhu, and Song Han. Svdquant: Absorbing outliers by low-rank components for 4-bit diffusion models. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2411.05007.

Xiuyu Li, Yijiang Liu, Long Lian, Huanrui Yang, Zhen Dong, Daniel Kang, Shanghang Zhang, and Kurt Keutzer. Q-Diffusion: Quantizing Diffusion Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023. URL https://arxiv.org/abs/2302.04304.

Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo Numerical Methods for Diffusion Models on Manifolds. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2202.09778.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps. arXiv preprint arXiv:2206.00927, 2022. URL https://arxiv.org/abs/2206.00927.

Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-Solver++: Fast Solver for Guided Sampling of Diffusion Probabilistic Models. *Machine Intelligence Research*, 22:730–751, 2025. doi: 10.1007/s11633-025-1562-4. URL https://link.springer.com/article/10.1007/s11633-025-1562-4.

Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent Consistency Models: Synthesizing High-Resolution Images with Few-Step Inference. arXiv preprint arXiv:2310.04378, 2023. URL https://arxiv.org/abs/2310.04378.

Nanye Ma, Mark Goldstein, Michael S. Albergo, Nicholas M. Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring Flow and Diffusion-based Generative Models with Scalable Interpolant Transformers. In *European Conference on Computer Vision*, 2024. URL https://arxiv.org/abs/2401.08740.

William Peebles and Saining Xie. Scalable Diffusion Models with Transformers. arXiv preprint arXiv:2212.09748, 2023. URL https://arxiv.org/abs/2212.09748.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Tim Salimans and Jonathan Ho. Progressive Distillation for Fast Sampling of Diffusion Models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2202.00512.

Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial Diffusion Distillation. In *European Conference on Computer Vision*, 2024. URL https://www.ecva.net/papers/eccv_2024/papers_ECCV/html/11557_ECCV_2024_paper.php.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising Diffusion Implicit Models. In *International Conference on Learning Representations*, 2021. URL https://arxiv.org/abs/2010.02502.

Yuchuan Tian, Zhijun Tu, Hanting Chen, Jie Hu, Chao Xu, and Yunhe Wang. U-DiTs: Downsample Tokens in U-Shaped Diffusion Transformers. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2405.02730.

Junyi Wu, Haoxuan Wang, Yuzhang Shang, Mubarak Shah, and Yan Yan. PTQ4DiT: Post-training Quantization for Diffusion Transformers. arXiv preprint arXiv:2405.16005, 2024. URL https://arxiv.org/abs/2405.16005.

Haocheng Xi, Shuo Yang, Yilong Zhao, Chenfeng Xu, Muyang Li, Xiuyu Li, Yujun Lin, Han Cai, Jintao Zhang, Dacheng Li, Jianfei Chen, Ion Stoica, Kurt Keutzer, and Song Han. Sparse VideoGen: Accelerating Video Diffusion Transformers with Spatial-Temporal Sparsity. arXiv preprint arXiv:2502.01776, 2025. URL https://arxiv.org/abs/2502.01776.

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. SANA: Efficient High-Resolution Image Synthesis with Linear Diffusion Transformers. arXiv preprint arXiv:2410.10629, 2024. URL https://arxiv.org/abs/2410.10629.

Jiazheng Xu, Xiao Liu, Yuchen Wu, Yuxuan Tong, Qinkai Li, Ming Ding, Jie Tang, and Yuxiao Dong. ImageReward: Learning and evaluating human preferences for text-to-image generation. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2304.05977.

Shuo Yang, Haocheng Xi, Yilong Zhao, Muyang Li, Jintao Zhang, Han Cai, Yujun Lin, Xiuyu Li, Chenfeng Xu, Jianfei Chen, Song Han, Kurt Keutzer, and Ion Stoica. Sparse VideoGen2: Accelerate Video Generation with Sparse Attention via Semantic-Aware Permutation. In *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2505.18875.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T. Freeman. Improved Distribution Matching Distillation for Fast Image Synthesis. arXiv preprint arXiv:2405.14867, 2024a. URL https://arxiv.org/abs/2405.14867.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T. Freeman, and Taesung Park. One-step Diffusion with Distribution Matching Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024b. URL https://arxiv.org/abs/2311.18828.

Z-Image Team. Z-Image: An Efficient Image Generation Foundation Model with Single-Stream Diffusion Transformer. arXiv preprint arXiv:2511.22699, 2025. URL https://arxiv.org/abs/2511.22699.

Jintao Zhang, Jia Wei, Pengle Zhang, Jun Zhu, and Jianfei Chen. Sageattention: Accurate 8-bit attention for plug-and-play inference acceleration. In *International Conference on Learning Representations*, 2025a. URL https://arxiv.org/abs/2410.02367.

Jintao Zhang, Chendong Xiang, Haofeng Huang, Jia Wei, Haocheng Xi, Jun Zhu, and Jianfei Chen. SpargeAttention: Accurate and Training-free Sparse Attention Accelerating Any Model Inference. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pp. 76397–76413, 2025b. URL https://proceedings.mlr.press/v267/zhang25ch.html.

Peiyuan Zhang, Yongqi Chen, Runlong Su, Hangliang Ding, Ion Stoica, Zhengzhong Liu, and Hao Zhang. Fast Video Generation with Sliding Tile Attention. In *International Conference on Machine Learning*, 2025c. URL https://arxiv.org/abs/2502.04507.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. URL https://openaccess.thecvf.com/content_cvpr_2018/html/Zhang_The_Unreasonable_Effectiveness_CVPR_2018_paper.html.

Tianchen Zhao, Tongcheng Fang, et al. Vidit-q: Efficient and accurate quantization of diffusion transformers for image and video generation. In *International Conference on Learning Representations*, 2025. URL https://arxiv.org/abs/2406.02540.

Wenliang Zhao, Lujia Bai, Yongming Rao, Jie Zhou, and Jiwen Lu. UniPC: A Unified Predictor-Corrector Framework for Fast Sampling of Diffusion Models. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2302.04867.

Yilong Zhao, Chien-Yu Lin, Kan Zhu, Zihao Ye, Lequn Chen, Size Zheng, Luis Ceze, Arvind Krishnamurthy, Tianqi Chen, and Baris Kasikci. Atom: Low-Bit Quantization for Efficient and Accurate LLM Serving. In *Proceedings of Machine Learning and Systems*, volume 6, 2024. URL https://proceedings.mlsys.org/paper_files/paper/2024/hash/5edb57c05c81d04beb716ef1d542fe9e-Abstract-Conference.html.

Le Zhuo, Ruoyi Du, Han Xiao, Yangguang Li, Dongyang Liu, Rongjie Huang, Wenze Liu, Lirui Zhao, Fu-Yun Wang, Zhanyu Ma, Xu Luo, Zehan Wang, Kaipeng Zhang, Xiangyang Zhu, Si Liu, Xiangyu Yue, Dingning Liu, Wanli Ouyang, Ziwei Liu, Yu Qiao, Hongsheng Li, and Peng Gao. Lumina-Next: Making Lumina-T2X Stronger and Faster with Next-DiT. In *Advances in Neural Information Processing Systems*, 2024. URL https://arxiv.org/abs/2406.18583.

A Model Details and Generation Settings

We evaluate three text-to-image models and one text-to-video model.

PixArt-Σ. PixArt-Σ (Chen et al., 2024a) is a 600M-parameter diffusion transformer with 28 blocks, each consisting of self-attention, cross-attention, and feed-forward layers. We use the XL/2 1024-MS checkpoint to generate 1024×1024 images with 20 DPM-Solver++ sampling steps and a classifier-free guidance (CFG) scale of 4.0.

Z-Image-Turbo and Z-Image Base. Z-Image-Turbo and Z-Image Base (Z-Image Team, 2025) share a 6B-parameter single-stream diffusion transformer with 30 main blocks that jointly process text and image tokens. We evaluate the distilled Turbo checkpoint and the non-distilled Z-Image checkpoint, which we refer to as Base throughout. Both models generate 1024×1024 images using the flow-matching Euler scheduler provided with their respective checkpoints. Turbo uses 8 denoising steps with CFG disabled, whereas Base uses 50 denoising steps with a CFG scale of 4.0.

SANA-Video. SANA-Video (Chen et al., 2025b) is a 2B-parameter video diffusion transformer with 20 blocks and linear self-attention. We use its 480p checkpoint to generate 81-frame videos at 832×480 resolution (width $\times$ height) and 16 frames per second. Sampling uses 50 DPM-Solver++ steps with a CFG scale of 6.0. For each model, all compared methods use the same checkpoint, resolution, sampler, number of sampling steps, CFG scale, evaluation prompts, and random seeds.

B Data Preparation and Calibration

B.1 Quantization Settings and Layer Scope

We use group-wise symmetric quantization with a group size of 64 for both weights and activations. Activation scales are computed dynamically. Each quantized input site uses $K = 128$ high-precision channels. We use $G = 2, 2, 6, 4$ stages for PixArt-Σ, Z-Image-Turbo, Z-Image Base, and SANA-Video, respectively. For ViDiT-Q, we follow its mixed-precision configuration, using W8A8 for the linear layers in the first and last transformer blocks and for all feed-forward layers, and W4A8 for the remaining linear layers. For SVDQuant, we use a low-rank branch with rank $r = 128$ for the evaluated Z-Image and SANA-Video models.

For PixArt-Σ, we quantize the query, key, value, and output projections in both self-attention and cross-attention, together with the two feed-forward projections. For Z-Image-Turbo and Base, we quantize the attention projections and three feed-forward projections in the main transformer blocks and the noise and context refiners. For SANA-Video, we quantize the self-attention projections, the query and output projections of cross-attention, and the two 1×1 pointwise convolutions in each feed-forward block. The pointwise convolutions are equivalently implemented as linear layers before quantization.

B.2 Video Evaluation Prompts and Sampling

Following the dimension selection of ViDiT-Q (Zhao et al., 2025), we use the subject-consistency, scene, and overall-consistency prompt suites from VBench (Huang et al., 2024), totaling 251 prompts. For the main comparison, we generate five videos per prompt with seeds 42–46, yielding 1,255 videos per method. All methods use the same prompts, random seeds, and generation settings. Generation uses fixed expanded prompts, while text-related metrics use the original VBench prompts for evaluation. Calibration prompts are separate from the evaluation prompts.

B.3 Stage Partitioning

We partition the denoising trajectory into contiguous stages by minimizing the normalized reconstruction error of mixed-precision activations, allowing timesteps within each stage to share one high-precision configuration. Reconstruction errors across all quantized input sites jointly determine the stage boundaries, while high-precision channels are selected separately at each site.

Criterion. We index the denoising steps in descending order as $t = T-1, \ldots, 0$, and let $\ell \in \mathcal{S}$ index quantized input sites. Let $\bar{\boldsymbol{X}}_{\ell,t}$ denote the calibration activations reordered using the fixed channel permutation in Sec. 4.2. For a candidate interval $[a, b)$, we aggregate the calibration activations within the interval and select the K_ℓ channels with the largest RMS magnitudes following Sec. 4.1, where K_ℓ is the high-precision channel budget at site ℓ. We denote their indices in the reordered layout by $\bar{\mathcal{I}}_\ell(a, b)$ and apply the A4/A8 quantization scheme in Sec. 4.1 to obtain the reconstructed mixed-precision activations $\widetilde{\boldsymbol{X}}_{\ell,t}^{(a,b)}$. The interval cost is

$$C(a,b) = \sum_{\ell \in \mathcal{S}} \frac{\sum_{t=a}^{b-1} \left\| \bar{\boldsymbol{X}}_{\ell,t} - \widetilde{\boldsymbol{X}}_{\ell,t}^{(a,b)} \right\|_F^2}{\sum_{t=0}^{T-1} \left\| \bar{\boldsymbol{X}}_{\ell,t} \right\|_F^2}. \tag{8}$$

This cost accounts for reconstruction errors in both A4 and A8 channels; a lower value indicates more accurate activation reconstruction under the interval's shared configuration. The denominator uses the sum of squared activation magnitudes over the full trajectory at each site, reducing the influence of scale differences across sites while remaining constant across candidate intervals. In practice, we estimate these statistics through stratified row sampling, weighting each estimate by the ratio of the original row count to the sampled row count.

Partitioning objective. For the prescribed high-precision channel budgets, we partition the trajectory into G contiguous stages with descending boundaries $T = b_0 > b_1 > \cdots > b_G = 0$. Each stage must contain at least L timesteps. We seek

$$\boldsymbol{b}^\star = \underset{\substack{T=b_0>b_1>\cdots>b_G=0 \\ b_{g-1}-b_g \geq L,\ g=1,\ldots,G}}{\arg\min} \sum_{g=1}^{G} C(b_g, b_{g-1}). \tag{9}$$

Stage g contains $\mathcal{T}_g = \{t \mid b_g \leq t < b_{g-1}\}$ and is executed from $b_{g-1} - 1$ down to b_g. Here, G controls the number of stage-specific configurations, while L prevents excessively short stages; a feasible partition requires $GL \leq T$.

Dynamic programming. We precompute the costs of all feasible intervals. Let $F(g, a)$ denote the minimum cumulative cost obtained by partitioning the first $T - a$ executed steps, from $T-1$ down to a, into g stages. The recurrence is

$$F(g,a) = \min_{a+L \leq b \leq T-(g-1)L} \left[F(g-1,b) + C(a,b) \right], \tag{10}$$

where the minimization considers only feasible predecessor states. We initialize $F(0, T) = 0$, record the minimizing split at each state, and backtrack from $F(G, 0)$ to obtain the boundaries. Each input site then uses $\bar{\mathcal{I}}_\ell(b_g, b_{g-1})$ as the selected high-precision channels for stage g. After interval-cost precomputation, the dynamic program takes $O(GT^2)$ time. This process is performed offline; inference selects the high-precision branch corresponding to the current denoising stage.

C Additional experimental results

C.1 Complete image quality metrics

Table 4 provides all seven PRISM-EN categories and the overall results for both Z-Image models, including alignment (Ali.), aesthetic (Aes.), and average (Avg., equivalent to Combined) scores.

Table 4: Complete PRISM-EN results for Z-Image-Turbo and Z-Image Base. Ali., Aes., and Avg. denote alignment, aesthetic, and their average (Combined) scores, respectively. Overall aggregates all seven categories; all scores are higher-is-better. Scores use a 10-point scale. Best and second-best results among the reported quantized methods are shown in **bold** and underline, respectively.

Method	Precision (W/A)	Imagination			Entity			Text Rendering			Style		
		Ali.	Aes.	Avg.	Ali.	Aes.	Avg.	Ali.	Aes.	Avg.	Ali.	Aes.	Avg.
Z-Image-Turbo (8 steps)													
BF16	16/16	4.84	3.08	3.96	6.23	4.69	5.46	5.55	5.92	5.74	5.61	6.13	5.87
Q-DiT	4/4	4.81	3.10	3.96	**6.32**	**5.49**	**5.91**	4.86	**5.46**	5.16	5.39	6.18	5.79
SVDQuant	4/4	4.75	**3.13**	3.94	6.31	5.19	5.75	4.76	5.15	4.96	5.36	5.91	5.64
Ours	4/4	**4.99**	3.06	**4.03**	6.25	5.48	5.87	**5.03**	5.43	**5.23**	**5.46**	**6.38**	**5.92**
Z-Image Base (50 steps)													
BF16	16/16	5.40	3.04	4.22	6.66	4.49	5.58	5.00	4.60	4.80	5.90	6.12	6.01
Q-DiT	4/4	4.73	2.35	3.54	5.27	2.82	4.05	3.52	2.30	2.91	4.50	4.14	4.32
Ours	4/4	**5.76**	**3.52**	**4.64**	**7.20**	**4.70**	**5.95**	**5.15**	**3.98**	**4.57**	**5.98**	**6.27**	**6.13**

Method	Precision (W/A)	Affection			Composition			Long Text			Overall		
		Ali.	Aes.	Avg.	Ali.	Aes.	Avg.	Ali.	Aes.	Avg.	Ali.	Aes.	Avg.
Z-Image-Turbo (8 steps)													
BF16	16/16	7.64	5.89	6.77	8.47	5.72	7.10	7.29	3.32	5.31	6.52	4.96	5.74
Q-DiT	4/4	7.34	**6.76**	7.05	8.49	**6.18**	7.34	**7.25**	3.77	5.51	6.35	**5.28**	5.81
SVDQuant	4/4	**7.72**	6.55	**7.14**	8.45	5.61	7.03	7.20	3.66	5.43	6.37	5.03	5.70
Ours	4/4	7.36	6.21	6.79	**8.91**	6.14	**7.53**	7.18	**3.94**	**5.56**	**6.45**	5.23	**5.84**
Z-Image Base (50 steps)													
BF16	16/16	7.77	4.95	6.36	8.74	4.83	6.79	7.40	2.86	5.13	6.70	4.41	5.55
Q-DiT	4/4	5.38	2.35	3.87	7.20	2.63	4.92	5.49	2.23	3.86	5.16	2.69	3.92
Ours	4/4	**7.63**	**4.96**	**6.30**	**8.61**	**5.21**	**6.91**	**7.64**	**3.48**	**5.56**	**6.85**	**4.59**	**5.72**

C.2 Additional Qualitative Comparisons

Figures 8, 9, and 10 provide additional qualitative comparisons for PixArt-Σ, Z-Image-Turbo, and Z-Image Base, respectively.

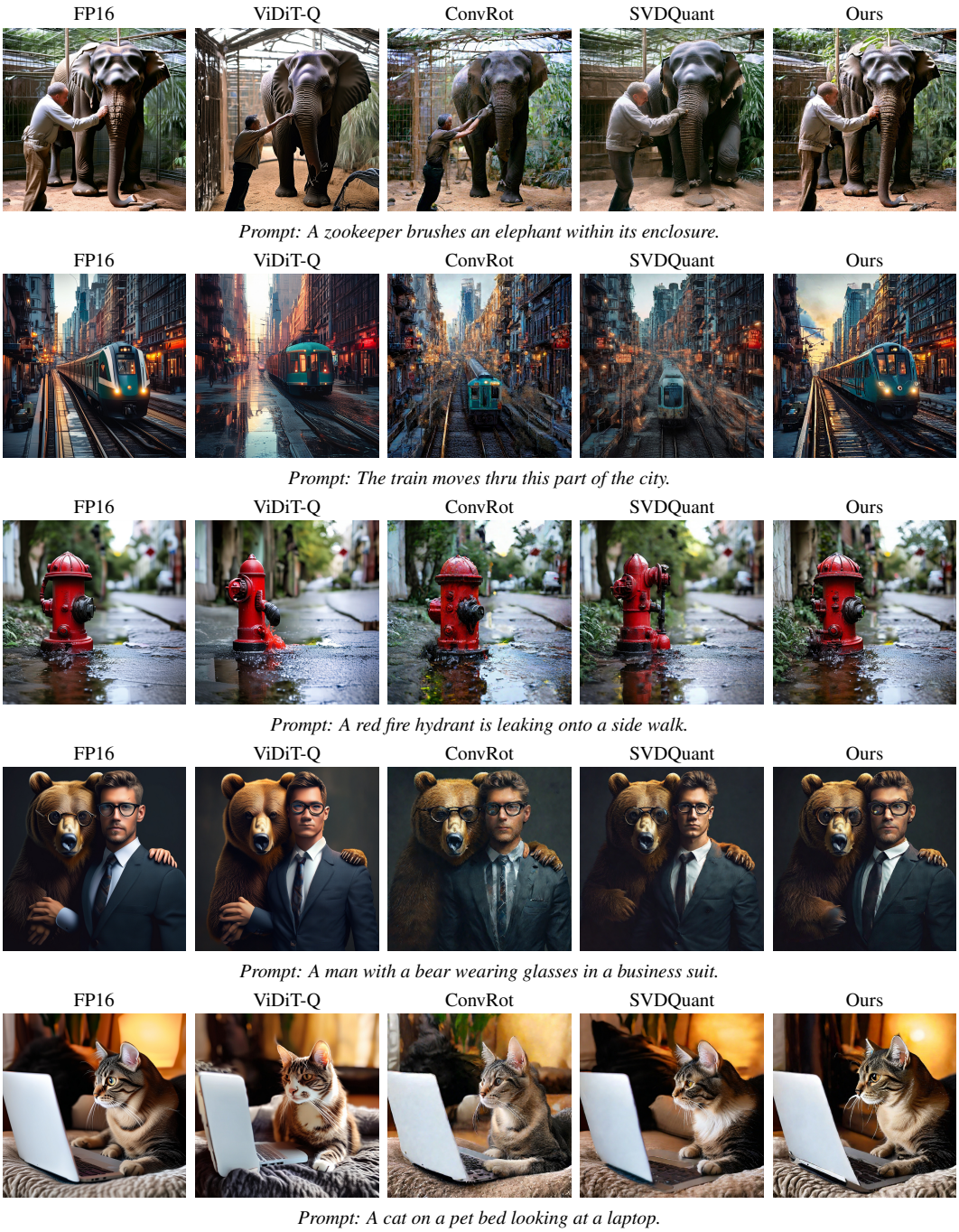
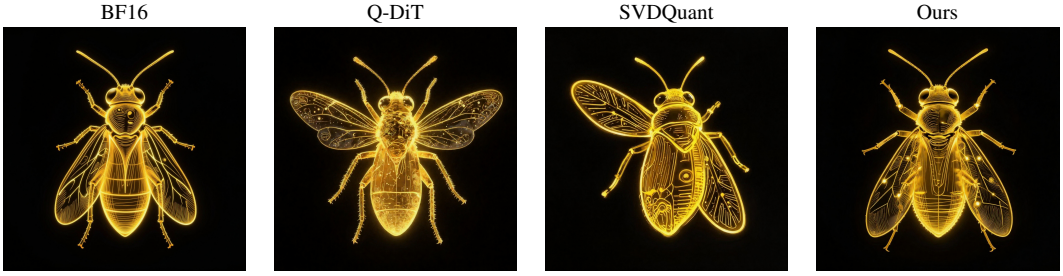


Figure 8: Additional qualitative comparisons on PixArt-Σ using five COCO prompts. Each row shows FP16, ViDiT-Q, ConvRot, SVDQuant, and Ours from left to right.

BF16 Q-DiT SVDQuant Ours



Prompt: A glowing firefly insect rendered in a sleek, neon vector art style against a stark black background, showcasing intricate line work and a striking golden-yellow hue that evokes a cyberpunk aesthetic.

BF16 Q-DiT SVDQuant Ours



Prompt: The reflection of a woman's is captured within the dark frame of a car's side-view mirror on a damp, overcast day. She is in the process of taking a self-portrait, holding a smartphone with a distinctive case that features a pattern of green monstera leaves on a white background. As she grips the phone, a large, striking ring with a round blue stone is prominent on her finger. Although her face is partially hidden by the phone, her dark, bold lipstick is a noticeable feature. She seems to be wearing a light-colored jacket, possibly pink or beige, with black long sleeves visible underneath. The surface of the mirror itself, as well as the car window seen in the reflection, is speckled with numerous small raindrops, indicating recent or current rainfall. The main background of the photograph, outside the car, is a dense, lush wall of green foliage. The broad, waxy leaves of this hedge are a deep shade of green and are thoroughly wet, glistening with countless water droplets that catch the ambient light. Peeking out from the upper parts of the greenery are small clusters of delicate white blossoms. The overall aesthetic of the image is moody and atmospheric, characterized by a cool, muted color palette that emphasizes the textures of the wet leaves and the creative framing of the selfie.

BF16 Q-DiT SVDQuant Ours



Prompt: A cozy bedroom features a neatly arranged bed with colorful pillows and blankets, accompanied by a vase of greenery on a side table, while the wall above displays the humorous quote "WHY FALL IN LOVE, WHEN YOU CAN... FALL ASLEEP?" in bold black letters.

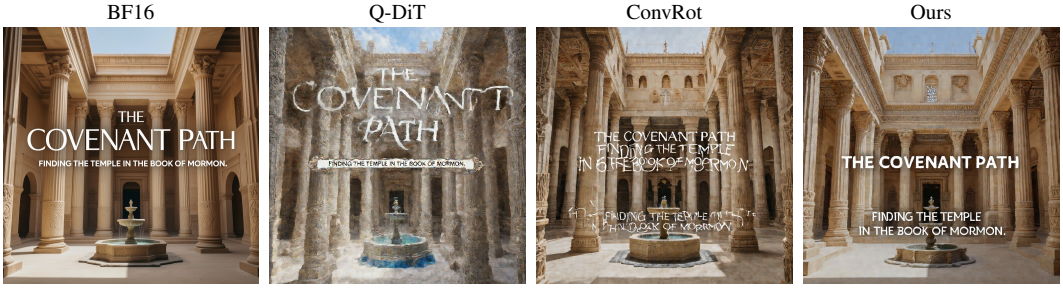
BF16 Q-DiT SVDQuant Ours



Prompt: Exploring the vibrant atmosphere of Asakusa in Tokyo, featuring the iconic Senso-ji Temple, a must-visit destination in this historic neighborhood.

Figure 9: Additional qualitative comparisons on Z-Image-Turbo using four PRISM-EN prompts. Each row shows BF16, Q-DiT, SVDQuant, and Ours from left to right.

BF16 Q-DiT ConvRot Ours



Prompt: An ornate temple interior features grand columns and a central fountain, with the title 'THE COVENANT PATH' prominently displayed, followed by the subtitle 'FINDING THE TEMPLE IN THE BOOK OF MORMON.'

BF16 Q-DiT ConvRot Ours



Prompt: A glass mug filled with a rich, amber-colored beverage topped with frothy foam displays the word "BUTTERBEER" prominently on its side.

BF16 Q-DiT ConvRot Ours



Prompt: A surreal feline figure stands serenely, its back transformed into a lush, miniature ecosystem where birds soar gracefully among sprouting trees, while its belly glows with swirling cosmic patterns, defying the boundaries between nature and the universe.

Figure 10: Additional qualitative comparisons on Z-Image Base using three PRISM-EN prompts. Each row shows BF16, Q-DiT, ConvRot, and Ours from left to right.