VETA-DiT: Variance-Equalized and Temporally Adaptive Quantization for Efficient 4-bit Diffusion Transformers

Qinkai Xu* **Yijin Liu*** **Yang Chen** **Lin Yang** **Li Li†** **Yuxiang Fu†**
Nanjing University
{qinkaixu, yijinliu, yangchen_nju}@smail.nju.edu.cn,
{linyang, lili, yuxiangfu}@nju.edu.cn
*Equal Contribution. †Corresponding Author.

Abstract

Diffusion Transformers (DiTs) have recently demonstrated remarkable performance in visual generation tasks, surpassing traditional U-Net-based diffusion models by significantly improving image and video generation quality and scalability. However, the large model size and iterative denoising process introduce substantial computational and memory overhead, limiting their deployment in real-world applications. Post-training quantization (PTQ) is a promising solution that compresses models and accelerates inference by converting weights and activations to low-bit representations. Despite its potential, PTQ faces significant challenges when applied to DiTs, often resulting in severe degradation of generative quality. To address these issues, we propose VETA-DiT (**V**ariance-**E**qualized and **T**emporal **A**daptation for **Di**ffusion **T**ransformers), a dedicated quantization framework for DiTs. Our method first analyzes the sources of quantization error from the perspective of inter-channel variance and introduces a Karhunen–Loève Transform enhanced alignment to equalize variance across channels, facilitating effective quantization under low bit-widths. Furthermore, to handle the temporal variation of activation distributions inherent in the iterative denoising steps of DiTs, we design an incoherence-aware adaptive method that identifies and properly calibrates timesteps with high quantization difficulty. We validate VETA-DiT on extensive image and video generation tasks, preserving acceptable visual quality under the more aggressive W4A4 configuration. Specifically, VETA-DiT reduces FID by 33.65 on the DiT-XL/2 model and by 45.76 on the PixArt-Σ model compared to the baseline under W4A4, demonstrating its strong quantization capability and generative performance. Code is available at: https://github.com/xululi0223/VETA-DiT.

1 Introduction

Recently, Diffusion Transformers (DiTs) [33] have emerged as the dominant backbone architecture for diffusion models, replacing the U-Net structures [36]. They have been widely adopted in various generation tasks due to their superior performance [8, 30, 9]. A notable example is OpenAI's SoRA [32], which has attracted significant attention for its remarkable generation quality. Recent studies [11, 51] further demonstrate the impressive capability and scalability of DiTs across modalities.

Despite their success, DiTs face two major limitations: the inherently lengthy iterative denoising process and the growing model size, both of which lead to substantial computational and memory demands. These issues hinder the deployment of DiTs in resource-constrained environments and limit their applicability in real-time scenarios. For instance, generating a 1024×1024 resolution image with DiTs can take up to 10 seconds even on a NVIDIA A100 GPU.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

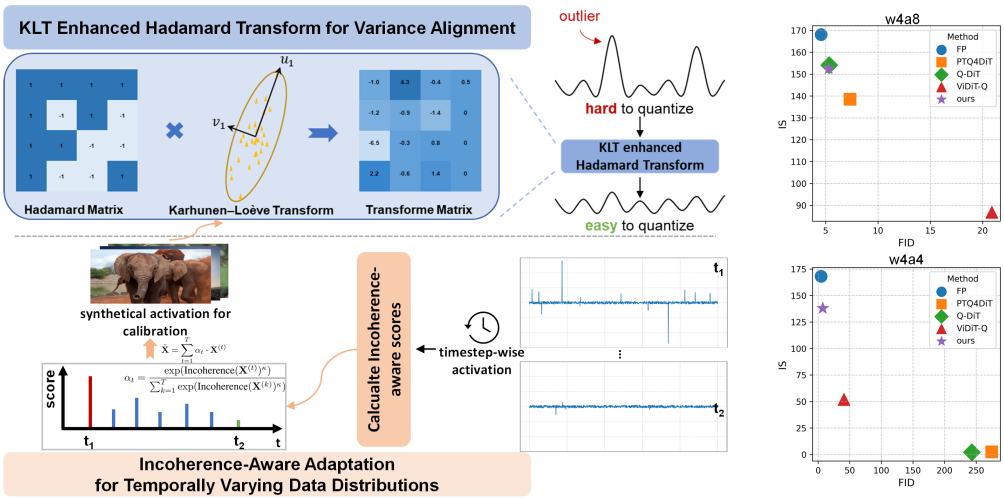


Figure 1: **(Left)** Overview of VETA-DiT. The Hadamard matrix is enhanced by a K-L transform to adapt to data distributions and reduce quantization difficulty. Calibration data fed into the KLT is synthesized via an incoherence-aware adaptive strategy to account for temporal variance across different timesteps. **(Right)** Quantization performance under W4A8 and W4A4, where points closer to the top-left corner indicate better performance. Please refer to Section 4.2 for detailed results.

Model quantization [31, 19] has been widely explored as an effective technique to accelerate inference by reducing memory footprint and computational overhead. It achieves this by compressing model weights and activations from floating-point to low-bit integer representations. However, the complex data distributions of weights and activations in DiTs pose significant challenges for existing quantization techniques, especially when targeting ultra-low bit-width without severely degrading performance. We identify two key challenges that hinder low-bit quantization of DiTs: (1) the presence of extreme outliers in specific channels results in significant inter-channel variance. Traditional approaches such as applying a smoothing factor [46, 45] or using simple Hadamard transforms [2, 49] fail to effectively align this variance, making low-bit quantization inaccurate; (2) the inherent iterative denoising nature of DiTs causes large variations in activation distributions across different timesteps, making it difficult to strike a balance between quantization effectiveness and computational efficiency.

To address these challenges, we propose a novel quantization method tailored for DiTs, termed VETA-DiT. We first analyze the limitations of directly applying the Hadamard transform from the perspective of inter-channel variance, and introduce a KLT enhanced Hadamard transform to achieve better variance alignment, enabling acceptable performance even under low-bit scenarios. To handle the temporal variation in activation distributions, we design an incoherence-aware adaptive strategy to identify the challenging timesteps for quantization. We then construct a synthetical calibration set that ensures both quantization accuracy and efficiency.

Our contributions are summarized as follows:

1. We analyze the limitations of directly applying the Hadamard transform in DiTs and propose a K-L transform-enhanced Hadamard method to effectively align inter-channel variance.
2. We investigate the quantization difficulty caused by temporal activation variation, and propose an incoherence-aware importance scoring strategy to construct a synthetical calibration set that captures representative distributions across different timesteps.
3. We conduct extensive experiments on both image and video generation tasks with multiple DiT models, and push the quantization precision to W4A4, demonstrating effectiveness of VETA-DiT in achieving high quantization performance without sacrificing visual fidelity.

2 Background and Related Works

2.1 Diffusion Transformer

Diffusion Models (DMs) [16] have attracted significant attention due to their powerful generative capabilities in visual domains such as image [33, 4] and video [29] synthesis. DMs simulate a

forward process that progressively adds Gaussian noise to the original data via a Markov chain [35], perturbing the data into a distribution close to standard Gaussian. A learned denoising network is then employed in the reverse process to iteratively reconstruct high-quality samples. The denoising process is typically parameterized by a deep neural networkϵ_θ, which predicts the noise at each timestep. Given a sample drawn from standard Gaussian noise $x_T \sim \mathcal{N}(0, I)$, the model iteratively denoises it through:

$$p_\theta(x_{t-1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \hat{\beta}_t I), \tag{1}$$

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_{\theta,t}), \quad \hat{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}. \tag{2}$$

The architecture of the noise prediction network plays a crucial role in determining the performance of diffusion models. DiT is a representative method that integrates Transformer architectures into diffusion models. Its core building block consists of multiple Transformer layers, each composed of a Multi-Head Self-Attention (MHSA) mechanism and a Feed-Forward Network (FFN) [42, 7, 33]. Specifically, MHSA and FFN are formulated as follows, respectively:

$$MHSA(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = Softmax(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V}, \tag{3}$$

$$FFN(\mathbf{X}) = LayerNorm(\mathbf{X} + MLP(\mathbf{X})). \tag{4}$$

To incorporate conditional information (e.g., class labels), each Transformer block employs MLPs to project a condition vector $\mathbf{c} \in \mathbb{R}^{d_{\text{in}}}$ into scale and shift parameters, which are then injected into the hidden state $\mathbf{Z} \in \mathbb{R}^{n \times d_{\text{in}}}$ via an adaptive LayerNorm (adaLN) [33]:

$$(\gamma, \beta) = MLPs(\mathbf{c}), \quad adaLN(\mathbf{Z}) = LN(\mathbf{Z}) \odot (1+\gamma) + \beta. \tag{5}$$

Although DiT achieves superior generation quality and expressive power compared to traditional architectures, its inference phase incurs high computational and memory costs due to the deeply stacked Transformer layers and iterative denoising process.

2.2 Model Quantization

Model quantization [19], is a widely adopted technique for reducing model size and accelerating inference. Quantization methods are typically categorized into Quantization-Aware Training (QAT) [50, 43, 12] and Post-Training Quantization (PTQ) [10, 46, 49, 45, 5, 20]. QAT integrates the quantization process into the training pipeline, often preserving model performance but requiring extensive computational resources and retraining time. In contrast, PTQ statically analyzes pretrained models using a small calibration dataset to estimate quantization parameters, offering advantages in deployment speed and computational efficiency.

A typical example is symmetric linear quantization, which can be expressed as:

$$x_{int} = clamp(\lfloor \frac{x}{s} \rceil - z, p_{min}, p_{max}), \tag{6}$$

where x and x_{int} denote the original and quantized data, respectively; s is the scaling factor, z is the zero point, $\lfloor \cdot \rceil$ denotes the rounding operation, and the clamp function constrains the quantized values within $[p_{min}, p_{max}]$.

Although PTQ techniques have demonstrated promising results on architectures such as CNNs [21, 22, 44] and ViTs [23, 44, 27], directly applying them to DiTs remains challenging. On one hand, DiTs often exhibit significant inter-channel distribution imbalance, making it difficult for a unified quantization range to accommodate all channels effectively. On the other hand, the stepwise sampling mechanism in diffusion processes introduces strong temporal dynamics in the activation distributions, necessitating quantization strategies that are adaptive to the timestep. To address these challenges, several studies have explored efficient PTQ methods for diffusion models. For instance, Q-Diffusion [21] and PTQ4DM [39] analyze activation variance across time steps to improve temporal robustness, while Q-DiT and PTQ4DiT incorporate joint temporal-channel characteristics into the design of adaptive quantization mechanisms. ViDiT-Q [49] proposes a unified approach that accounts for both temporal dependency and inter-channel imbalance by leveraging Hadamard rotations and asymmetric formats to enhance quantization effectiveness. However, they suffer from challenges under lower bit-width quantization settings. Recently, SVDQuant [20] employs a high-precision low-rank branch to take in the weight outliers with singular value decomposition, which is orthogonal to our method.

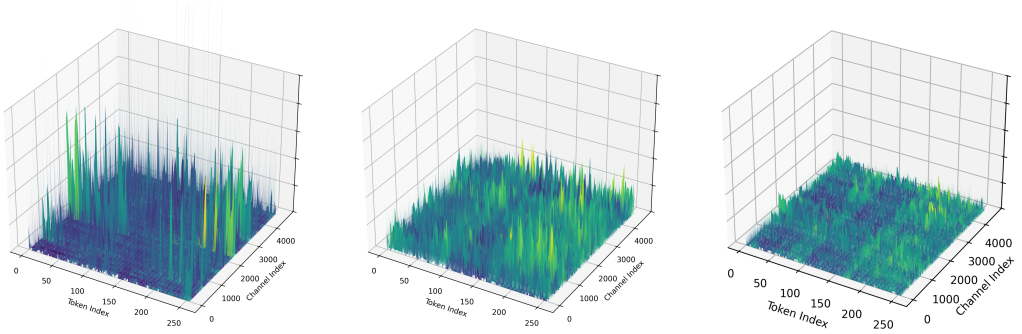


Figure 2: **(Left)** Activation distribution in the DiT model, showing extreme outliers in certain channels. Distributions after the Hadamard **(Middle)** and K-L-enhanced Hadamard transforms **(Right)**, respectively. While the Hadamard transform helps reduce variance imbalance, it has limitations; the K-L-enhanced method further improves inter-channel variance alignment.

3 Method

3.1 KLT Enhanced Hadamard Transform for Variance Alignment

We begin by analyzing the intermediate activations of various layers in the DiT-XL/2 model, focusing on selected linear layers to investigate their activation distribution characteristics. As shown in Figure 2 (Left), the activations exhibit a distribution pattern similar to those observed in LLMs and DMs [13, 39]: certain channels contain outlier values with extremely large magnitudes. In some layers, the magnitude of the outliers can be up to 100 times larger than the rest of the activations. These outliers significantly increase the variance within individual channels, resulting in substantial variance discrepancies across different channels [1, 2, 24]. Since per-token quantization methods assume uniform variance across channels, such discrepancies introduce considerable quantization errors, which are particularly problematic in hardware deployment.

Previous works in LLM quantization have proposed using the Hadamard transform to mitigate the influence of outliers and reduce inter-channel variance. Formally, given an input tensor $\mathbf{X} \in \mathbb{R}^{n \times m}$, where n is the number of samples and m is the number of channels, the Hadamard transform is defined as: $\mathbf{Z} = \mathbf{XH}$, where $\mathbf{H} \in \mathbb{R}^{m \times m}$ is the Hadamard matrix. The Hadamard matrix $\mathbf{H}$ is an orthogonal matrix whose entries are drawn from $\left\{+\frac{1}{\sqrt{m}}, -\frac{1}{\sqrt{m}}\right\}$. Let $\mathbf{z}_j$ denote the j-th column of $\mathbf{Z}$, i.e., the j-th transformed channel. Its variance is:

$$Var(\mathbf{z}_j) = \frac{1}{n}\sum_{i=1}^{n} z_{ij}^2 = \frac{1}{n}\sum_{i=1}^{n}\left(\sum_{k=1}^{m} x_{ik}h_{kj}\right)^2. \tag{7}$$

Utilizing the linearity of expectation and the properties of variance and covariance, we derive:

$$Var(\mathbf{z}_j) = \sum_{k=1}^{m} Var(\mathbf{x}_k)h_{kj}^2 + \sum_{k\neq l} Cov(\mathbf{x}_k, \mathbf{x}_l)h_{kj}h_{lj}. \tag{8}$$

This expression shows that the variance of the transformed channel consists of two components: a weighted sum of the original channel variances and a cross-covariance term. Since the cross-covariance term and the corresponding Hadamard weight vector $h_{:,j}$ vary across channels j, the resulting variances $Var(\mathbf{z}_j)$ cannot be assumed to be numerically similar in most cases. As a result, the transformed channels have unequal variances, limiting the effectiveness of the Hadamard transform for variance equalization in quantization. As shown in Figure 2 (Middle), the activations after the Hadamard transform alone still exhibit nonnegligible inter-channel variance, which adversely affects the quantization performance.

To achieve better variance alignment, we seek a transformation that ensures uniform average energy across all transformed channels. Let $v_k = Var(\mathbf{x}_k)$ be the variance of the k-th channel, and define the energy vector $\mathbf{v} = [v_1, v_2, \ldots, v_m]^T$. Our goal is to find a linear transformation $\mathbf{T} \in \mathbb{R}^{m \times m}$ such that the transformed tensor $\mathbf{Y} = \mathbf{XT}$ satisfies:

$$Var(\mathbf{y}_j) = \frac{1}{m}\sum_{k=1}^{m} v_k, \quad \forall j. \tag{9}$$

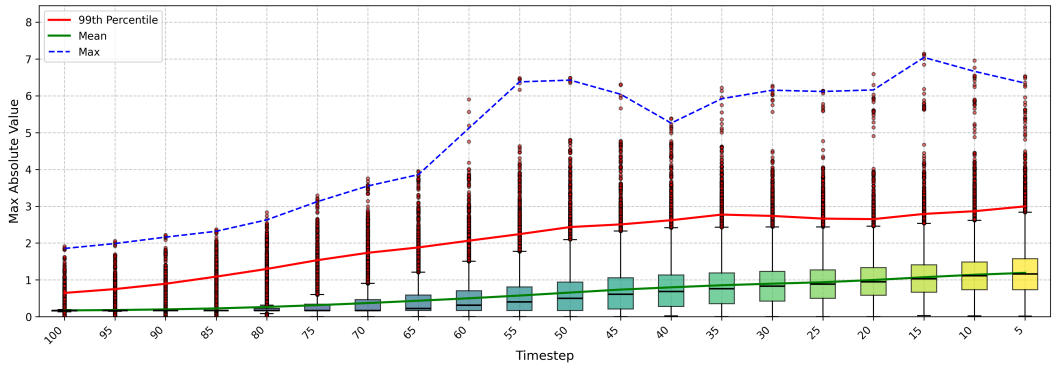


Figure 3: Boxplot of the maximum absolute activation values across channels at different timesteps in DiT, illustrating significant temporal variance that affects the effectiveness and efficiency of quantization methods.

That is, all output channels have the same average variance, maximizing the quantization range and minimizing quantization error.

This can be achieved using Karhunen–Loève Transform (KLT), which constructs an orthogonal basis from the eigenvectors of the input covariance matrix [6]. The KLT distributes total variance across orthogonal directions, minimizing redundancy and maximizing statistical independence. It has also been applied to improve quantization performance in state space models [48]. By combining KLT with the structured and computationally efficient Hadamard transform, we choose a composite transformation called the KLT enhanced Hadamard transform (KLT-H): $\mathbf{T}_{\text{KLT-H}} = \mathbf{KH}$, where $\mathbf{K}$ is the KLT matrix composed of eigenvectors of the input covariance matrix. Since both $\mathbf{K}$ and $\mathbf{H}$ are orthogonal, the composite transformation $\mathbf{T}_{\text{KLT-H}}$ remains orthogonal. Its inverse is simply $\mathbf{T}_{\text{KLT-H}}^{\top}$, which enables computational consistency by ensuring that the quantized model produces mathematically equivalent outputs to the original model, as illustrated by $(\mathbf{X}\mathbf{T}_{\text{KLT-H}})(\mathbf{T}_{\text{KLT-H}}^{\top}\mathbf{W}) = \mathbf{X}\mathbf{W}$. After transformation, the variance of each channel becomes:

$$Var(\mathbf{y}_j) = \sum_{k=1}^{m} \lambda_k h_{kj}^2, \tag{10}$$

where λ_k are the eigenvalues of the input covariance matrix. For normalized Hadamard matrices, $h_{kj}^2 = \frac{1}{m}$, leading to:

$$Var(\mathbf{y}_j) = \frac{1}{m} \sum_{k=1}^{m} \lambda_k, \tag{11}$$

which indicates that the variance becomes equalized across different channels. After the KLT enhanced Hadamard transform, the inter-channel variance is further reduced, as illustrated in Figure 2 (Right). Thus, the transformed tensor achieves variance alignment across channels, significantly reducing quantization error and enhancing the robustness of diffusion model inference under low-bit quantization.

3.2 Incoherence-Aware Adaptation for Temporally Varying Data Distributions

DiT models generate images through an iterative denoising process. As shown in Figure 3, we observe that the activation distributions at different timesteps during the denoising process vary significantly. Previous quantization methods [26, 49] typically apply the same transformation matrix $\mathbf{H}$ to all timesteps within a layer. While this approach ensures computational efficiency, it often leads to severe performance degradation under low-bit quantization. Unfortunately, the K-L transformation matrix is highly sensitive to the distribution characteristics of the calibration dataset, and temporal variance in activation distributions undermine its effectiveness.

A straightforward solution is to compute a dedicated K-L transformation matrix for each timestep. However, this would incur substantial storage and computation overhead, counteracting the efficiency benefits of quantization.

To address this challenge, we propose an incoherence-aware, time-adaptive strategy that improves the performance of quantized models while preserving computational efficiency. The concept of

incoherence, adapted from [3, 41], is used to measure the degree of anomaly in activation data. Specifically, a matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ is said to be μ-incoherent if for any i and j:

$$|\mathbf{X}_{ij}| = |e_i^T \mathbf{X} e_j| \leq \mu \cdot \frac{||\mathbf{X}||_F}{\sqrt{mn}}, \tag{12}$$

where $||\cdot||_F$ denotes the Frobenius norm. A higher incoherence value indicates that certain channel activations deviate significantly from the global mean, suggesting skewed distributions and outliers that complicate the quantization process.

During quantization, we assign each timestep t an importance score s_t, which guides the KLT to pay more attention to activation data with highly skewed distributions. The score is defined as:

$$s_t = \text{Incoherence}(\mathbf{X}^{(t)}) = \frac{\max |\mathbf{X}^{(t)}|}{||\mathbf{X}^{(t)}||_F / \sqrt{mn}}, \tag{13}$$

where $\mathbf{X}^{(t)}$ denote the activation data sampled at timestep t. However, since incoherence values can differ drastically across timesteps, directly using s_t may cause certain timesteps to dominate the calibration statistics. To mitigate this, we normalize the scores such that the sum of weights across all timesteps equals 1, ensuring a balanced contribution. The normalized form is given by:

$$\alpha_t = \frac{s_t^\kappa}{\sum_{k=1}^{T} s_k^\kappa}, \tag{14}$$

where $\kappa > 0$ is a control parameter. When $\kappa = 1$, the weighting is linear; higher values of κ emphasize timesteps with extreme incoherence more strongly.

In practice, we apply an entropy-based regularization and a softmax-based weighting formulation, which together enhance stability and differentiability:

$$\alpha_t = \frac{\exp(\text{Incoherence}(\mathbf{X}^{(t)})^\kappa)}{\sum_{k=1}^{T} \exp(\text{Incoherence}(\mathbf{X}^{(k)})^\kappa)}. \tag{15}$$

We construct a synthetical calibration dataset $\tilde{\mathbf{X}}$ by combining activations across all timesteps with their respective weights:

$$\tilde{\mathbf{X}} = \sum_{t=1}^{T} \alpha_t \cdot \mathbf{X}^{(t)}. \tag{16}$$

In this way, the synthetical calibration dataset reflects activation distribution characteristics across multiple timesteps. This enables more accurate variance alignment, reduces quantization error, and improves the robustness of the quantized model. We apply the proposed method to the linear layer analyzed in Figure 3, where the incoherence-based importance scores across timesteps are computed. As shown in Figure 4 (Left), these scores exhibit a strong positive correlation with the inter-channel variance at each timestep, indicating that the method effectively prioritizes timesteps with higher quantization difficulty. We then perform KLT enhanced Hadamard transforms using calibration datasets constructed from either randomly sampled timestep activations or the incoherence-aware aggregation. The resulting channel-wise variance distributions, shown in Figure 4 (Right), demonstrate that the incoherence-aware approach effectively adapts to temporal variance and significantly reduces inter-channel variance.

4 Experiments

4.1 Experimental Settings

Image Generation. We first evaluate our proposed VETA-DiT framework on the image generation task. Following the original evaluation settings of DiT [33], we use the pretrained DiT-XL/2 model [33] to generate images with resolutions of 256×256 on the ImageNet dataset [37]. The DDIM-solver [40] is employed during generation, with sampling steps set to 50 and 100. To further demonstrate the generality of VETA-DiT on diverse generation tasks, we also integrate it into the PixArt-Σ model [4] for prompt-based image generation on the COCO dataset [25]. A DPM-solver [28] with 20 steps is used, the classifier-free guidance (CFG) scale is set to 4.5. We use FID [15], spatial FID(sFID),

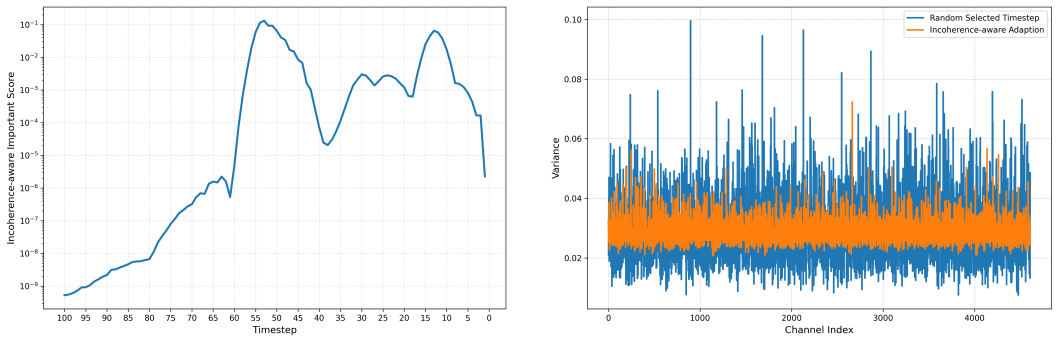


Figure 4: **(Left)** Incoherence-aware importance scores across timesteps, showing a positive correlation with inter-channel variance at each timestep. **(Right)** Channel-wise variance distributions obtained by applying the K-L enhanced Hadamard transform using calibration datasets constructed from either randomly sampled timestep activations or incoherence-weighted aggregated activations. The incoherence-aware adaptation effectively accommodates temporal variation.

Inception Score [38], and Precision to evaluate generation fidelity. Additionally, for the COCO task, we include ClipScore [14] to assess text-image alignment, and ImageReward [47] to estimate human preference under complex prompt conditions.

Video Generation. We further evaluate the effectiveness of VETA-DiT in the video generation task by integrating it into the STDiT3 model from the Open-Sora [17]. Videos are generated using a 50-step DDIM-solver with a CFG scale of 4.0. We conduct comprehensive evaluations on the VBench benchmark to obtain detailed quantitative results. Following [49, 34], we select 8 major evaluation dimensions from VBench [18].

4.2 Quantization Performance

We conduct a comprehensive evaluation of our proposed VETA-DiT and compare it against several state-of-the-art post-training quantization (PTQ) approaches specifically designed for DiTs under various experimental settings. In particular, we consider three representative baselines: PTQ4DiT [45], Q-DiT [5], and ViDiT-Q [49]. To ensure fair and consistent comparisons, we re-implement and adapt these methods based on their official open-source repositories to support different network architectures and task scenarios. For ViDiT-Q, we intentionally exclude the use of its original mixed-precision quantization strategy in our experiments, as it tends to quantize the majority of linear layers to INT8, which could obscure the performance differences among these methods. Further experimental details can be found in Appendix Section A.

Image Generation Results. Table 1 presents the quantitative results of DiT-XL/2 on the ImageNet 256×256 resolution task. Specifically, PTQ4DiT and Q-DiT achieve acceptable performance under the W4A8 setting, benefiting from their inter-channel importance balancing and dynamic group-wise quantization strategies. Our proposed VETA-DiT method further narrows the performance gap with the full-precision (FP) model under the same setting, achieving a best FID of 5.22 and an Inception Score of 152.52, indicating higher-quality image generation. When the quantization setting is further reduced to W4A4, PTQ4DiT and Q-DiT fail to produce meaningful images due to the limitations of their methodologies. Remarkably, VETA-DiT remains effective in generating visually acceptable images even under this low-precision setting, reducing FID and sFID by 33.65 and 16.68, respectively, compared to the second-best method. This demonstrates the effectiveness of our proposed variance-balancing strategy and temporal difference-aware adaptation. ViDiT-Q, without the use of mixed-precision quantization, which is incompatible with many existing hardware platforms, suffers from severe performance degradation. We further evaluate VETA-DiT on the COCO dataset using the PixArt-Σ model for text-to-image generation. As shown in Figure 5, the results are consistent with those observed on ImageNet: ViDiT-Q fails to generate acceptable outputs under W4A8 due to the lack of mixed-precision support. Although Q-DiT performs reasonably well at W4A8, it struggles with handling inter-channel imbalances, leading to poor-quality generated images. In contrast, VETA-DiT consistently delivers superior performance across both bit-width settings. Additional randomly generated images are provided in Appendix Figure 10, 11, 12, 13 for visual comparison.

Table 1: Performance comparison of DiT-XL/2 on the ImageNet 256×256.

Model	Bit-width	Method	IS(↑)	FID(↓)	sFID(↓)	Precision(↑)
DiT-XL/2 steps=100 cfg=1.0	W16A16	FP	90.32	12.25	19.28	0.65
	W4A8	PTQ4DiT	66.85	17.34	20.81	0.58
		Q-DiT	**77.07**	16.16	19.24	0.61
		ViDiT-Q	37.46	49.65	31.87	0.41
		Ours	75.90	**15.22**	**18.66**	**0.62**
	W4A4	PTQ4DiT	2.08	304.62	129.68	0.07
		Q-DiT	1.93	256.13	429.47	0.01
		ViDiT-Q	25.11	72.23	44.57	0.32
		Ours	**64.91**	**22.85**	**20.50**	**0.57**
DiT-XL/2 steps=100 cfg=1.5	W16A16	FP	168.08	4.56	17.72	0.79
	W4A8	PTQ4DiT	138.54	7.33	22.40	0.74
		Q-DiT	**154.13**	5.34	17.62	0.76
		ViDiT-Q	86.81	20.87	25.40	0.56
		Ours	152.53	**5.29**	**17.50**	**0.77**
	W4A4	PTQ4DiT	2.42	274.59	117.25	0.08
		Q-DiT	2.14	243.05	410.74	0.02
		ViDiT-Q	51.96	40.87	35.58	0.45
		Ours	**137.98**	**7.22**	**18.90**	**0.74**
DiT-XL/2 steps=50 cfg=1.0	W16A16	FP	88.07	13.38	19.07	0.65
	W4A8	PTQ4DiT	61.08	23.41	22.29	0.54
		Q-DiT	**76.12**	17.42	18.92	0.61
		ViDiT-Q	30.79	58.48	39.57	0.36
		Ours	75.05	**17.30**	**18.36**	**0.63**
	W4A4	PTQ4DiT	2.05	298.57	126.64	0.08
		Q-DiT	1.81	262.58	421.08	0.01
		ViDiT-Q	18.13	84.40	57.46	0.28
		Ours	**64.15**	**24.24**	**21.98**	**0.56**
DiT-XL/2 steps=50 cfg=1.5	W16A16	FP	164.61	4.86	17.65	0.80
	W4A8	PTQ4DiT	129.06	8.89	22.73	0.71
		Q-DiT	**148.83**	5.77	**17.57**	0.76
		ViDiT-Q	81.65	22.65	26.45	0.56
		Ours	147.12	**5.53**	17.65	**0.77**
	W4A4	PTQ4DiT	2.33	281.43	119.45	0.08
		Q-DiT	2.02	249.98	404.09	0.01
		ViDiT-Q	48.34	44.15	37.90	0.43
		Ours	**135.66**	**7.90**	**19.11**	**0.73**

'(WxYb)' indicates that the weights and activations are quantized to x-bit and y-bit, respectively.

Bit-width	Method	IS(↑)	FID(↓)	sFID(↓)	CLIP(↑)	IR(↑)
W16A16	FP	38.27	59.74	294.91	0.27	0.89
W4A8	Q-DiT	**38.34**	63.36	299.88	0.26	**0.93**
	ViDiT-Q	18.81	125.32	325.12	0.22	-0.55
	Ours	37.62	**60.46**	**297.20**	0.26	0.90
W4A4	Q-DiT	18.27	107.98	312.25	0.24	-0.43
	ViDiT-Q	9.70	227.80	332.97	0.20	-1.52
	Ours	**35.56**	**65.22**	**305.36**	**0.26**	**0.88**

Figure 5: **(Left)** Text-to-image generation performance of the PixArt-Σ model on the COCO dataset. **(Right)** Generated images comparison of W4A4 quantization.

Video Generation Results.. As shown in Table 2, similar to the image generation task, existing quantization methods achieve satisfactory performance under the W4A8 setting. However, due to their limited ability to balance inter-channel variance, they struggle under the more challenging W4A4 setting. In contrast, our method consistently outperforms these baselines across various metrics, demonstrating its effectiveness in preserving both the visual quality and temporal consistency of generated videos. Appendix Figure 14, 15 presents a visual comparison of a randomly generated video for visual evaluation. We also conduct additional experiments to further evaluate the video generation performance of the quantized models; see Appendix D.3 for details.

Table 2: Performance comparison of Open-Sora on the VBench evaluation benchmark.

Bit-width	Method	Subject Consist.	BG. Consist.	Motion Smooth.	Dynamic Degree	Aesthetic Quality	Imaging Quality	Scene Consist.	Overall Consist.
W16A16	FP	0.947	0.965	0.983	0.680	0.575	0.519	0.454	0.274
W4A8	Q-DiT	**0.951**	**0.962**	0.985	0.600	0.569	0.497	0.451	0.273
	ViDiT-Q	0.920	0.963	0.982	0.520	0.536	0.495	0.316	0.269
	Ours	0.950	0.961	**0.985**	**0.600**	**0.571**	**0.509**	**0.506**	**0.273**
W4A4	Q-DiT	0.934	0.955	0.983	0.440	0.508	0.452	0.311	0.252
	ViDiT-Q	0.897	0.962	0.978	0.600	0.499	0.468	0.333	0.254
	Ours	**0.939**	**0.959**	**0.984**	**0.480**	**0.546**	**0.499**	**0.431**	**0.269**

4.3 Ablation Studies

To assess the effectiveness of each component in our proposed framework, we conduct ablation experiments under the challenging W4A4 setting. These experiments are carried out using the DiT-XL/2 model on the ImageNet 256×256 dataset, with 50 sampling steps using the DDIM-solver. Detailed quantitative results are provided in Table 3. We begin our evaluation with a baseline that employs a simple group-wise linear quantization strategy. Under the W4A4 configuration, the baseline performs poorly across all metrics. Next, we incorporate a K-L enhanced Hadamard transform for variance alignment, which significantly improves the quality of generated images, achieving a FID of 48.32, and an IS of 43.68. Building upon this, we further introduce the incoherence-aware temporal adaptation method, which enhances the ability of the Hadamard transform to adapt across different timesteps. This results in a further reduction of the FID to 7.90. These findings demonstrate the individual effectiveness of each proposed component and highlight how their integration contributes to pushing our VETA-DiT method toward state-of-the-art performance under W4A4 settings, enabling the generation of visually plausible and semantically coherent images.

Table 3: Ablation study on ImageNet 256×256 with W4A4.

Method	IS	FID	sFID	Precision
FP	164.61	4.86	17.64	0.80
Baseline	1.84	260.47	409.15	0.01
+ K-L enhanced Hadamard Transform	43.68	48.32	40.19	0.36
+ Incoherence-aware adaption	135.66	7.90	19.10	0.72

5 Conclusion

This paper presents **VETA-DiT**, a post-training quantization framework designed for Diffusion Transformers (DiTs) to reduce inference cost while preserving generation quality. We address two major challenges: large inter-channel variance due to outliers and significant activation distribution shifts across denoising timesteps. To this end, we introduce a KLT-enhanced Hadamard transform for variance alignment and an incoherence-aware adaptive calibration method to handle temporal dynamics. Experiments on image and video generation tasks show that VETA-DiT achieves performance close to the full-precision model under W4A8, while still delivering acceptable visual quality under the W4A4 setting. Our approach advances the deployment of DiTs in resource-constrained scenarios.

6 Acknowledgement

This work was supported in part by the National Key Research and Development Program of China under Grant No. 2023YFB2806802 and Grant No. 2021YFB3600104, in part by the Joint Funds of the National Nature Science Foundation of China under Grant No. U21B2032 and in part by Suzhou's "Jiebang Guashuai" Project for Key Core Technologies under Grant No. SYG2024134. Lin Yang would like to thank the support from NSFC (No. 62306138), JiangsuNSF (No. BK20230784), and the Inovation Program of State Key Laboratory for Novel Software Technology at Nanjing University (No. ZZKT2024B15, ZZKT2025B25). The authors are grateful for the help from the Interdisciplinary Research Center for Future Intelligent Chips (Chip-X) and Yachen Foundation.

References

[1] Saleh Ashkboos, Maximilian L Croci, Marcelo Gennari do Nascimento, Torsten Hoefler, and James Hensman. Slicegpt: Compress large language models by deleting rows and columns. In *The Twelfth International Conference on Learning Representations*, 2024.

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

[3] Jerry Chee, Yaohui Cai, Volodymyr Kuleshov, and Christopher M De Sa. Quip: 2-bit quantization of large language models with guarantees. *Advances in Neural Information Processing Systems*, 36:4396–4429, 2023.

[4] 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. In *European Conference on Computer Vision*, pages 74–91, 2024.

[5] 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. *arXiv preprint arXiv:2406.17343*, 2024.

[6] R Dony et al. Karhunen-loeve transform. *The transform and data compression handbook*, 1(1-34):29, 2001.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, G Heigold, S Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

[8] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

[9] Shibo Feng, Chunyan Miao, Zhong Zhang, and Peilin Zhao. Latent diffusion transformer for probabilistic time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11979–11987, 2024.

[10] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan-Adrian Alistarh. Optq: Accurate post-training quantization for generative pre-trained transformers. In *11th International Conference on Learning Representations*, 2023.

[11] 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*, pages 23164–23173, 2023.

[12] Yefei He, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Efficientdm: Efficient quantization-aware fine-tuning of low-bit diffusion models. In *12th International Conference on Learning Representations*, 2024.

[13] 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 2023*, 2023.

[14] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528, 2021.

[15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, pages 6840–6851, 2020.

[17] HPC-AI. Open-Sora. `https://github.com/hpcaitech/Open-Sora`, 2024.

[18] 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.

[19] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713, 2018.

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

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

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

[23] Zhikai Li, Junrui Xiao, Lianwei Yang, and Qingyi Gu. Repq-vit: Scale reparameterization for post-training quantization of vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17227–17236, 2023.

[24] Haokun Lin, Haobo Xu, Yichen Wu, Jingzhi Cui, Yingtao Zhang, Linzhan Mou, Linqi Song, Zhenan Sun, and Ying Wei. Duquant: Distributing outliers via dual transformation makes stronger quantized llms. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[25] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer vision–ECCV 2014: 13th European conference, zurich, Switzerland, September 6-12, 2014, proceedings, part v 13*, pages 740–755, 2014.

[26] Wenxuan Liu and Sai Qian Zhang. Hq-dit: Efficient diffusion transformer with fp4 hybrid quantization. *arXiv preprint arXiv:2405.19751*, 2024.

[27] Yijiang Liu, Huanrui Yang, Zhen Dong, Kurt Keutzer, Li Du, and Shanghang Zhang. Noisyquant: Noisy bias-enhanced post-training activation quantization for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20321–20330, 2023.

[28] 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. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, pages 5775–5787, 2022.

[29] 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. *Transactions on Machine Learning Research*, 2025.

[30] Shentong Mo, Enze Xie, Ruihang Chu, HONG Lanqing, Matthias Nießner, and Zhenguo Li. Dit-3d: Exploring plain diffusion transformers for 3d shape generation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

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

[32] OpenAI. Video generation models as world simulators. `https://openai.com/index/video-generation-models-as-world-simulators/`, 2024.

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

[34] Weiming Ren, Huan Yang, Ge Zhang, Cong Wei, Xinrun Du, Wenhao Huang, and Wenhu Chen. Consisti2v: Enhancing visual consistency for image-to-video generation. *Transactions on Machine Learning Research*, 2024.

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241, 2015.

[37] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[38] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[39] Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1972–1981, 2023.

[40] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021.

[41] Albert Tseng, Jerry Chee, Qingyao Sun, Volodymyr Kuleshov, and Christopher De Sa. Quip-#: Even better llm quantization with hadamard incoherence and lattice codebooks. In *Forty-first International Conference on Machine Learning*, 2024.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[43] Haoxuan Wang, Yuzhang Shang, Zhihang Yuan, Junyi Wu, Junchi Yan, and Yan Yan. Quest: Low-bit diffusion model quantization via efficient selective finetuning. *arXiv preprint arXiv:2402.03666*, 2024.

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

[45] Junyi Wu, Haoxuan Wang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Ptq4dit: Post-training quantization for diffusion transformers. In *Advances in neural information processing systems*, 2024.

[46] 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, 2023.

[47] 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 *Proceedings of the 37th International Conference on Neural Information Processing Systems*, pages 15903–15935, 2023.

[48] Zukang Xu, Yuxuan Yue, Xing Hu, Zhihang Yuan, Zixu Jiang, Zhixuan Chen, Jiangyong Yu, Chen Xu, Sifan Zhou, and Dawei Yang. Mambaquant: Quantizing the mamba family with variance aligned rotation methods. *arXiv preprint arXiv:2501.13484*, 2025.

[49] 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.

[50] Xingyu Zheng, Xianglong Liu, Haotong Qin, Xudong Ma, Mingyuan Zhang, Haojie Hao, Jiakai Wang, Zixiang Zhao, Jinyang Guo, and Michele Magno. Binarydm: Accurate weight binarization for efficient diffusion models. *arXiv preprint arXiv:2404.05662*, 2024.

[51] Zheng Zhu, Xiaofeng Wang, Wangbo Zhao, Chen Min, Nianchen Deng, Min Dou, Yuqi Wang, Botian Shi, Kai Wang, Chi Zhang, et al. Is sora a world simulator? a comprehensive survey on general world models and beyond. *arXiv preprint arXiv:2405.03520*, 2024.

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: We state the contributions in the abstract and introduction sections.

   Guidelines:

   - The answer NA 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 NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

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

   Answer: [Yes]

   Justification: We discuss the main limitations in Appendix E.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

Justification: Please refer to Appendix C for related assumptions and proofs.

Guidelines:

- The answer NA 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: Implementation details are provided in Appendix A. We have released our code at anonymous repository: https://anonymous.4open.science/r/VETA-DiT.

   Guidelines:

   - The answer NA 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 have released our code at anonymous repository: https://anonymous.4open.science/r/VETA-DiT.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.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://nips.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, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to Section 4.1 and Appendix A for experimental settings.

   Guidelines:

   - The answer NA 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: Our study used a fixed seed for all quantization operatipons, following standards in post-training quantization, and thus did not report statistical significance.

   Guidelines:

   - The answer NA 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: As mentioned in Appendix A, all experiments are done on Nvidia A800 GPUs.

   Guidelines:

   - The answer NA 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 confirm the research conducted in thje paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA 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: We discuss the broader impacts of our work in Appendix E.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA 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., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This work poses no such risks.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: We cite all datasets and models utilized in our experiments.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: This work does not release new assets.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA 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: [NA]

    Justification: This work does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA 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 rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core methods of our research do not involve the use of LLMs.

Guidelines:

- The answer NA 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 (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Additional Experimental Details

In our implementation, we randomly selected 16 classes from the ImageNet dataset to perform image generation and calibrate the DiT-XL/2 model. For the PixArt-Σ model, we used the sample prompts provided in the official PixArt-Σ codebase for calibration. In video generation tasks, we calibrated the model using text-to-video prompts from the Open-Sora implementation.

For weight quantization, we adopted static quantization with a group size of 128. Additionally, we applied second-order information from the Hessian matrix, as used in GPTQ, to adjust the weights. This process is performed offline and introduces no additional overhead during inference. For activation quantization, we followed the same dynamic quantization strategy as Q-DiT, using a group size of 128 to balance quantization accuracy and inference cost. All experiments were conducted on Nvidia A800 GPUs.

For all baseline methods, we used the same calibration set, random noise, and classifier-free guidance, and quantized all Transformer layers in the DiT architecture.

In the ViDiT-Q implementation, we collected the maximum activation values of the layers to be quantized in order to determine the smooth factor. We set the hyperparameter α = 0.5 to control the smooth factor. The original ViDiT-Q applies mixed-precision quantization to both weights and activations, which we found leads to use higher precision in the linear layers of the FFN module. To ensure a fair comparison, we removed mixed-precision support in our implementation of ViDiT-Q and used the same precision settings as all other methods.

B Detailed Algorithm for VETA-DiT

Algorithm 1: KLT-H Based Quantization with Incoherence-Aware Temporal Sampling

Input: Activation tensor $\mathbf{X} \in \mathbb{R}^{t \times n \times m}$, weight matrix $\mathbf{W} \in \mathbb{R}^{m \times d}$
Output: Quantized activation tensor $\mathbf{X}_q$, quantized weight matrix $\mathbf{W}_q$

Step 1: Incoherence-aware sampling of activations
Initialize an importance vector $\boldsymbol{\alpha}$ of length t;
for *each time step* t_i **do**
    Extract the activation matrix $\mathbf{X}^{(t_i)}$ at time t_i;
    Compute the incoherence score s_{t_i} of $\mathbf{X}^{(t_i)}$;
    Compute important score α_{t_i} based on s_{t_i};
    Store α_{t_i} in $\boldsymbol{\alpha}$;

Step 2: Generate sampling distribution
Apply softmax to $\boldsymbol{\alpha}$ to obtain probability vector $\boldsymbol{p}$;
Sample multiple time steps based on $\boldsymbol{p}$;
Concatenate the sampled activation matrices to form $\mathbf{X}_{\text{sampled}}$;

Step 3: Construct KLT-H transformation matrix
Compute the covariance matrix of $\mathbf{X}_{\text{sampled}}$ and derive its eigenvectors to form the KLT matrix $\mathbf{K}$;
Construct the Hadamard matrix $\mathbf{H}$;
Form the composite orthogonal transform: $\mathbf{T}_{\text{KLT-H}} = \mathbf{K} \cdot \mathbf{H}$;

Step 4: Transform activation and weight using shared transform
Transform activation: $\mathbf{X}' = \mathbf{X} \cdot \mathbf{T}_{\text{KLT-H}}$;
Transform weight: $\mathbf{W}' = \mathbf{T}_{\text{KLT-H}}^{\top} \cdot \mathbf{W}$;
This ensures that the computation $(\mathbf{X} \cdot \mathbf{W})$ is equivalent to $(\mathbf{X}' \cdot \mathbf{W}')$, preserving functional correctness;

Step 5: Quantize the transformed activation and weight
Quantize $\mathbf{X}'$ using an dynamic activation quantization method, producing $\mathbf{X}_q$;
Quantize $\mathbf{W}'$ using a static weight quantization method, producing $\mathbf{W}_q$;

return *Quantized activation* $\mathbf{X}_q$ *and quantized weight* $\mathbf{W}_q$

The VETA-DiT pipeline is detailed in Algorithm 1. The algorithm consists of five key steps. **Step 1**, we compute the importance scores for each diffusion time step using incoherence-aware sampling. Time steps with higher quantization difficulty are assigned larger weights according to Equation (13, 14), allowing the transformation matrix to adapt to the temporally varying activation distributions. **Step 2**, after obtaining the importance scores, we enhance the numerical stability of the sampling process by applying Equation (15, 16) to generate a composite calibration set, which includes activations sampled in proportion to their importance. **Step 3**, we calculate the eigenvectors and eigenvalues of the covariance matrix of the incoherence-aware calibration set to construct the Karhunen–Loève Transform (KLT) matrix, which is then combined with a randomly generated Hadamard matrix, to form the KLT-enhanced Hadamard transformation matrix. **Step 4**, we apply the transformation matrix $\mathbf{T}_{\text{KLT-H}}$ to the activation tensor and its inverse (the transpose of the matrix due to orthogonality) to the weight tensor, aligning the inter-channel variance of both activations and weights while ensuring the correctness of the computation. **Step 5**, we apply dynamic quantization to the transformed activations and static quantization to the transformed weights, resulting in low-bit quantized activations and weights for efficient inference.

C Proofs

This section provides a detailed derivation demonstrating that the Hadamard transform enhanced by the K-L transform can achieve variance equalization across channels. Recall that the variance of the j-th transformed channel can be expressed as Equation 7. Expanding the squared summation, we obtain:

$$Var(\mathbf{z}_j) = \frac{1}{n} \sum_{i=1}^{n} \sum_{k=1}^{m} \sum_{l=1}^{m} x_{ik} x_{il} h_{kj} h_{lj}. \tag{17}$$

Interchanging the summation order and grouping terms:

$$Var(\mathbf{z}_j) = \sum_{k=1}^{m} \sum_{l=1}^{m} h_{kj} h_{lj} \left(\frac{1}{n} \sum_{i=1}^{n} x_{ik} x_{il} \right). \tag{18}$$

We define the sample covariance matrix $\mathbf{C} \in \mathbb{R}^{m \times m}$ as:

$$C_{kl} = \frac{1}{n} \sum_{i=1}^{n} x_{ik} x_{il}. \tag{19}$$

Then the variance of the j-th transformed channel becomes:

$$Var(\mathbf{z}_j) = \sum_{k=1}^{m} \sum_{l=1}^{m} C_{kl} h_{kj} h_{lj} = \mathbf{h}_j^\top \mathbf{C} \mathbf{h}_j, \tag{20}$$

where $\mathbf{h}_j \in \mathbb{R}^m$ is the j-th column of the orthogonal matrix $\mathbf{H}$.

To further interpret this expression, recall that:

- The diagonal elements of $\mathbf{C}$ are $C_{kk} = Var(\mathbf{x}_k)$.
- The off-diagonal elements are $C_{kl} = Cov(\mathbf{x}_k, \mathbf{x}_l)$ for $k \neq l$.

Thus, we can split the summation in Equation 20 into diagonal and off-diagonal parts to derive Equation 8, which shows even after applying an orthogonal transformation like the Hadamard transform, the resulting channel variances may still differ substantially due to the interaction between original variances and covariances. Hence, Hadamard transform alone does not guarantee uniform variance across channels.

Karhunen–Loève Transform (KLT), which uses the eigenvectors of the covariance matrix $\mathbf{C}$ to form an orthogonal basis. Let $\mathbf{C} = \mathbf{U} \mathbf{\Lambda} \mathbf{U}^\top$, where $\mathbf{U}$ contains the orthonormal eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix of eigenvalues $\lambda_1, \lambda_2, ..., \lambda_m$. Define $\mathbf{K} = \mathbf{U}^\top$ as the KLT matrix. The KLT transform is then $\mathbf{XK}$, with covariance:

$$\mathbf{C}_{\text{KLT}} = \frac{1}{n} (\mathbf{XK})^\top (\mathbf{XK}) = \mathbf{K}^\top \mathbf{C} \mathbf{K} = \mathbf{\Lambda}. \tag{21}$$

Thus, the transformed features are decorrelated and their variances are exactly the eigenvalues $\lambda_1, \ldots, \lambda_m$.

To combine the statistical decorrelation of KLT with the computational efficiency of Hadamard, we obtain the transformed tensor $\mathbf{Y} = \mathbf{XKH} = \mathbf{XT}_{\text{KLT-H}}$.

This combines the fast computation of the Hadamard transform with the statistical decorrelation property of the KLT. The covariance matrix of the transformed output $\mathbf{Y}$ is:

$$\mathbf{C_Y} = \frac{1}{n}\mathbf{Y}^\top\mathbf{Y} = \mathbf{H}^\top\mathbf{K}^\top\mathbf{CKH} = \mathbf{H}^\top\mathbf{\Lambda H}. \tag{22}$$

Then, the variance of the j-th channel in $\mathbf{Y}$ is:

$$Var(\mathbf{y}_j) = \mathbf{h}_j^\top\mathbf{\Lambda h}_j = \sum_{k=1}^{m}\lambda_k h_{kj}^2. \tag{23}$$

For a normalized Hadamard matrix, $h_{kj}^2 = \frac{1}{m}$ for all k, j, so we can obtain the Equation 11.

D Additional Empirical Results

D.1 Validation of KLT-H Transformation Effectiveness

We further analyze the effectiveness of the proposed KLT-H transformation in terms of incoherence reduction for both activations and weights. Specifically, we compute the normalized incoherence values across all layers of the DiT-XL/2 Transformer backbone under three settings: without transformation (original), with Hadamard transformation, and with our proposed KLT-H transformation. As shown in Figure 6 and Figure 7, the KLT-H consistently achieves lower incoherence than the other two settings, indicating its superior ability to decorrelate channels and better align with the intrinsic data structure. These improvements hold consistently for both weights (Figure 6) and activations (Figure 7), further validating the generality of our approach across different components of the model.

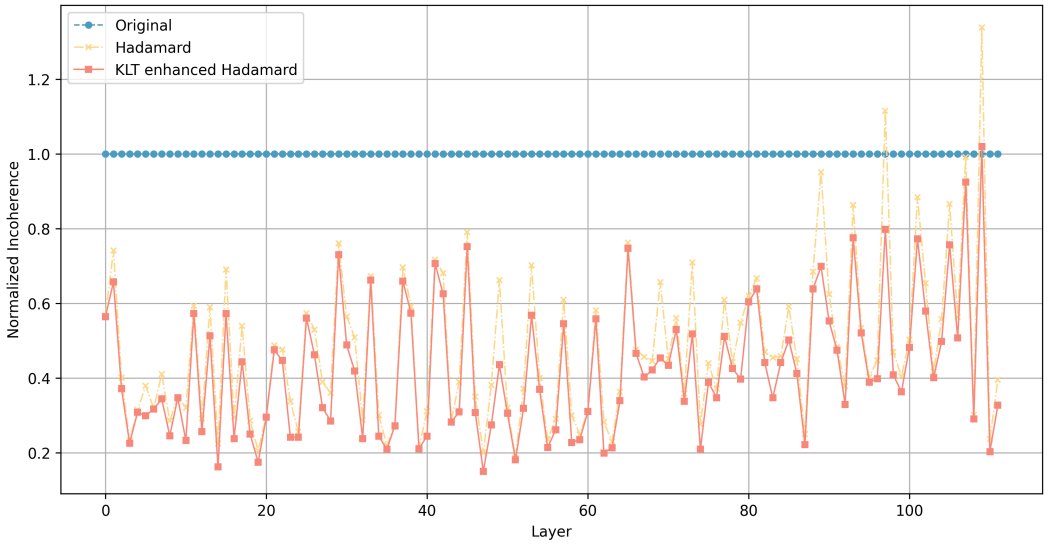


Figure 6: Normalized incoherence values of weights across all layers in the DiT-XL/2 Transformer backbone under different transformations. The KLT-H transformation consistently achieves lower incoherence, indicating its effectiveness in improving weight channel alignment.

In addition to the layer-wise results, we also compute the average incoherence and standard deviation across all layers. As summarized in Table 4, the KLT-H transformation yields the lowest average incoherence and the smallest standard deviation, further demonstrating its effectiveness and robustness

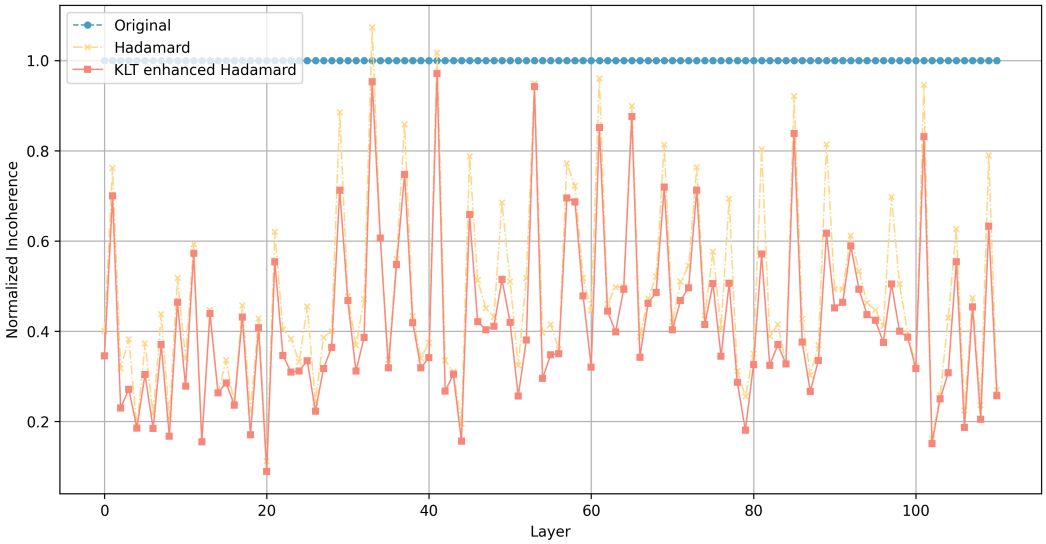


Figure 7: Normalized incoherence values of activations across all layers in the DiT-XL/2 Transformer backbone under different transformations. The KLT-H transformation significantly reduces incoherence compared to both the original and Hadamard-transformed activations, demonstrating improved inter-channel decorrelation.

in suppressing inter-channel redundancy. These results provide strong empirical evidence supporting the use of KLT-H as a principled and practical enhancement for quantization under temporally varying data distributions.

Table 4: Average and standard deviation of incoherence values across all layers of the DiT-XL/2 Transformer under different transformations.

Method	**Weight-Incoherence**	**Activation-Incoherence**
Original	19.91 ± 13.26	49.29 ± 44.05
Hadamard	8.56 ± 5.56	19.22 ± 11.80
KLT enhanced Hadamard	7.42 ± 4.40	17.13 ± 10.74

D.2 Evaluation of Incoherence-Aware Adaptation Strategy

To further validate the effectiveness of our proposed incoherence-aware adaptation strategy, we conduct additional experiments comparing three methods: (1) the standard RTN baseline without any transformation, (2) the KLT-H transform constructed using randomly selected samples, and (3) our incoherence-aware KLT-H transform. For each method, we compute the reconstruction error (mean squared error, MSE) between the output of each linear layer and the corresponding full-precision output.

As shown in Figure 8, under the W4A8 setting, the incoherence-aware KLT-H transform consistently achieves lower reconstruction errors across the majority of layers compared to the other two baselines. Similarly, Figure 9 presents the results under the W4A4 setting, where the advantage of our method becomes even more pronounced due to the more aggressive quantization constraints. These results highlight the robustness of the incoherence-aware adaptation in handling temporally varying data distributions under low-bit quantization.

D.3 Evaluation of Temporal and Textual Consistency in Video Generation

To comprehensively evaluate the generative capability of the quantized models, we additionally adopt three metrics from different perspectives. Specifically, CLIPSIM and CLIP-Temp are used to measure

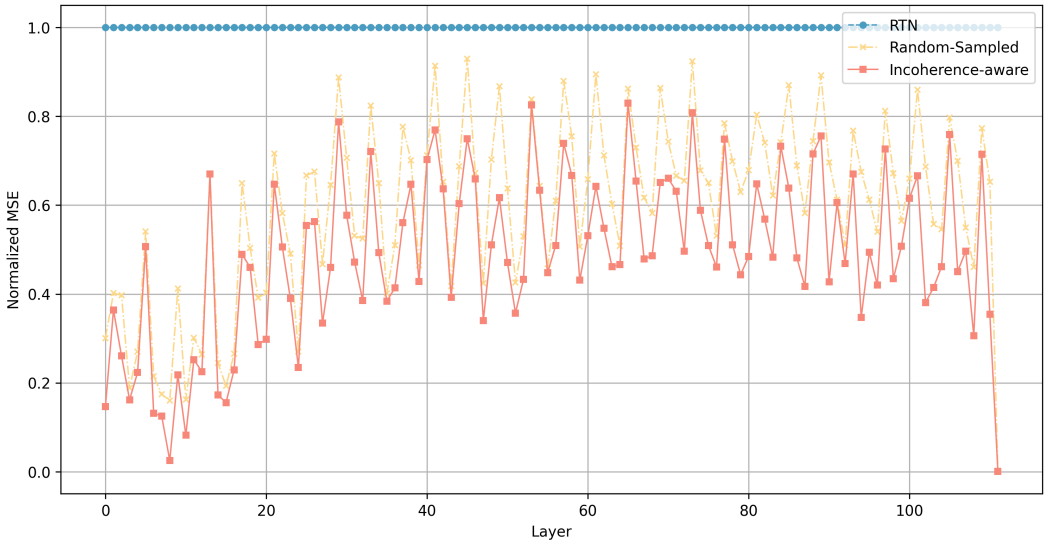


Figure 8: Normalized layer-wise MSE reconstruction error comparison under W4A8 quantization.

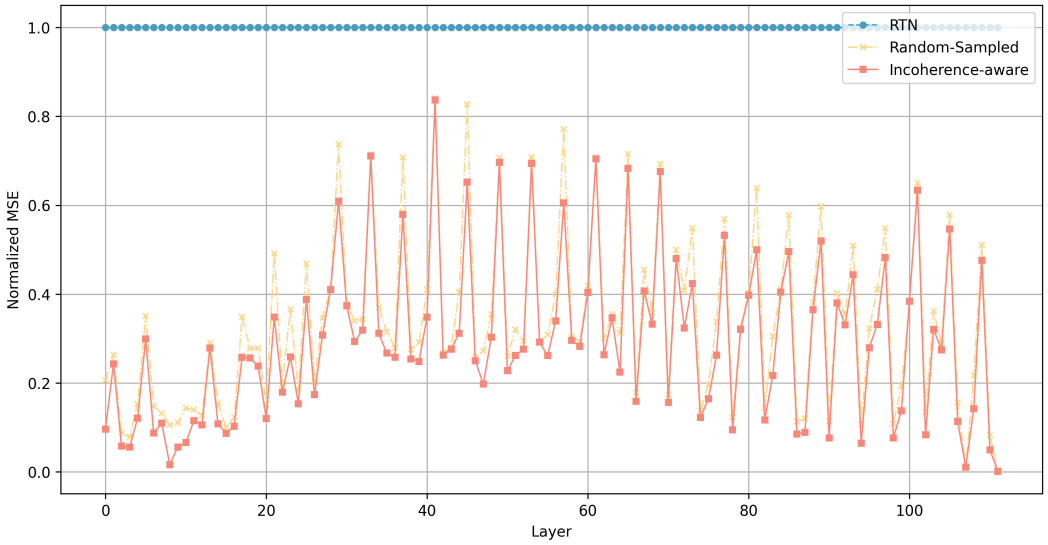


Figure 9: Normalized layer-wise MSE reconstruction error comparison under W4A4 quantization.

text-video alignment and temporal semantic consistency, respectively, while Temporal Flickering is employed to assess temporal smoothness. As shown in Table 5, our method demonstrates consistently strong performance under both W4A8 and W4A4 quantization settings, outperforming other PTQ methods in terms of both textual alignment and temporal consistency.

E Limitations and Broader Impacts

In this work, we present an effective post-training quantization framework that promotes the broader deployment and applicability of Diffusion Transformers (DiTs). By leveraging variance equalization and low-bit data representation, our method significantly reduces the computational and memory overhead, thereby improving the usability of DiTs in resource-constrained settings. VETA-DiT relies on a carefully designed incoherence-aware sampling strategy that assigns higher importance to timesteps with higher quantization difficulty. However, its applicability to other domains or modalities—such as audio and 3D data—requires further investigation. In future work, we plan

Table 5: Comparison of video generation quality under W4A8 and W4A4 settings using CLIPSIM, CLIP-Temp, and Temporal Flickering.

Bit-Width	Method	CLIPSIM	CLIP-Temp	Temporal Flick
W16A16	FP	0.2217	0.9980	0.9782
W4A8	Q-DiT	0.2198	0.9973	0.9798
	ViDiT-Q	0.2200	0.9975	**0.9820**
	Ours	**0.2204**	**0.9982**	0.9804
W4A4	Q-DiT	0.2161	0.9952	0.9702
	ViDiT-Q	0.2156	0.9953	0.9762
	Ours	**0.2176**	**0.9974**	**0.9770**

to explore hardware acceleration of VETA-DiT to enable real-time inference. In terms of broader impacts, this work advances efficient generative modeling by improving quantization techniques for diffusion models. Nonetheless, as with many optimization techniques that facilitate model deployment in low-resource scenarios, there is a risk of misuse in surveillance, deepfakes, or other sensitive content generation without proper regulation. We encourage the community to adopt responsible practices that align with ethical AI development principles.

F Additional Visualization Results

In this section, we present additional qualitative results to further demonstrate the effectiveness of our proposed VETA-DiT. Figure 10, 11 shows randomly generated ImageNet images using the DiT-XL/2 model under W4A8 and W4A4 quantization settings. Figure 12, 13 also provides examples of images generated by the PixArt-Sigma model. In addition, a frame sample from a randomly generated video is shown in Figure 14, 15. These results qualitatively highlight the strong generation capability of VETA-DiT under both moderate and aggressive quantization.

G Inference overhead

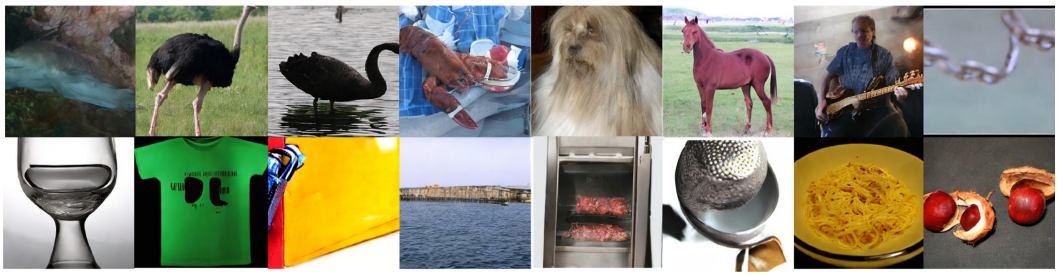
The KLT-enhanced Hadamard matrix is defined as $\mathbf{T}_{\text{KLT-H}} = \mathbf{KH}$, where $\mathbf{K}$ is a data-driven KLT matrix, precomputed offline using the calibration set, and $\mathbf{H}$ is a random Hadamard matrix.

During inference, this transformation can be fused into model weights, incurring no additional cost for most layers. For layers like out-proj and down-proj, online transformation is necessary to preserve computational invariance. We evaluated the associated overhead on DiT-XL/2 via implementing a custom CUDA kernel on NVIDIA A100 GPU, targeting matrix multiplication operations within these layers.

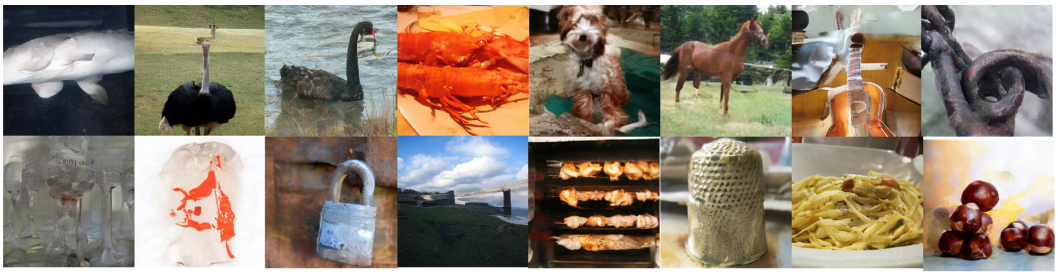
Table 6: Comparison of latency, speedup, and memory reduction.

Bit-Width	Method	Latency (ms)	Speedup	Memory (MB) / Reduction
W16A16	FP	6.167 ± 0.045	-	38.53 / -
W4A8	ViDiT-Q	3.879 ± 0.012	1.59×	18.68 / 2.06×
W4A4	Ours	2.960 ± 0.006	**2.08**×	9.68 / **3.98**×
W4A4 + Online Transform	Ours	3.163 ± 0.007	1.95×	10.31 / 3.74×

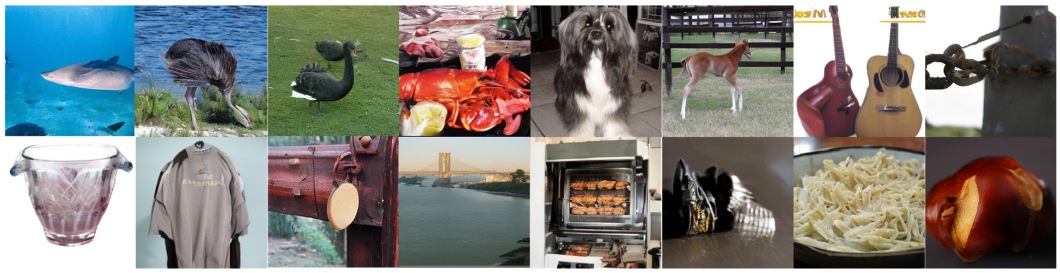
As shown in Table 6, due to the availability of efficient Hadamard transform implementations, the online transform introduces only 6.8% overhead, while still achieving 1.95× acceleration and 3.74× memory savings over FP16.



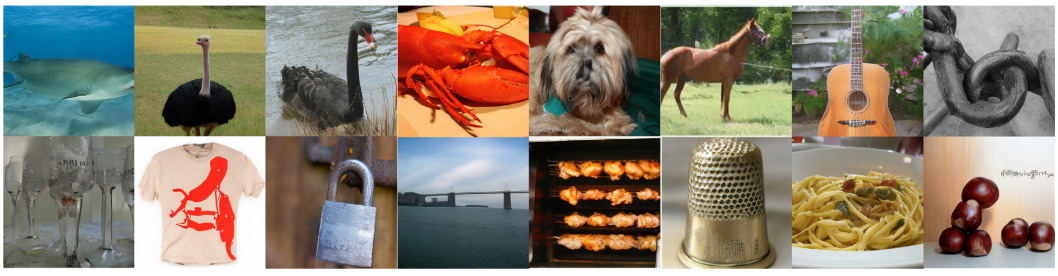
PTQ4DIT(W4A8)



VIDITQ(W4A8)



Q-DIT(W4A8)



Ours(W4A8)

Full-Precision

Figure 10: Random images generated under W4A8 quantization using different PTQ methods with the DiT-XL/2 model on ImageNet 256×256.

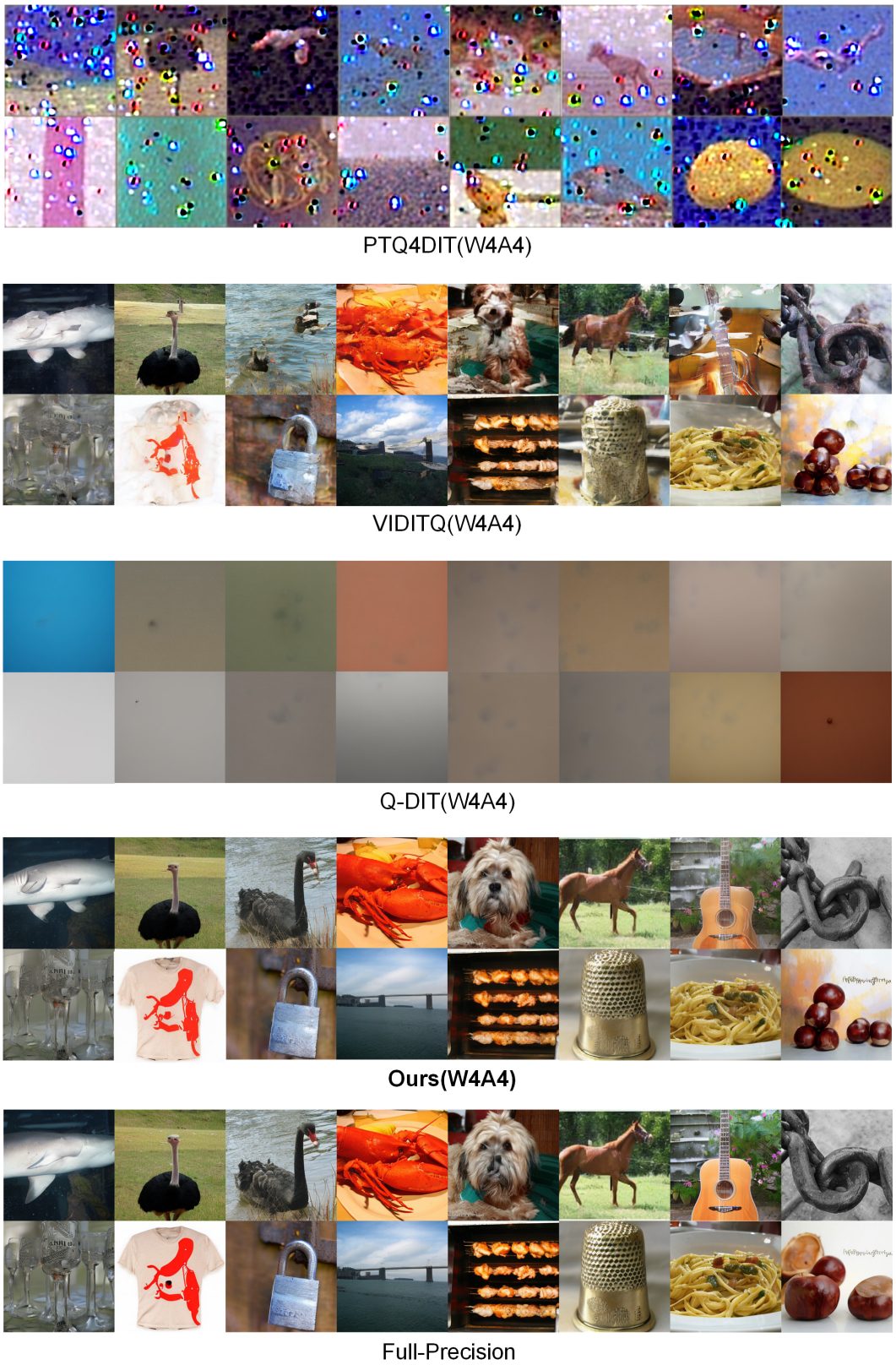


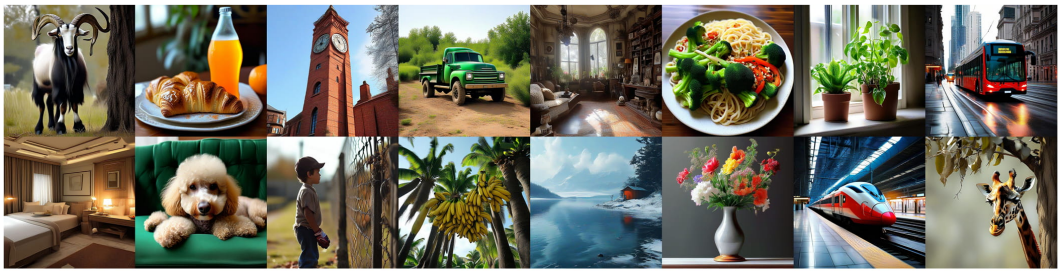
Figure 11: Random images generated under W4A4 quantization using different PTQ methods with the DiT-XL/2 model on ImageNet 256×256.

VIDITQ(mixed-presicion*)

VIDITQ(W4A8)

Q-DIT(W4A8)

Ours(W4A8)



Full-Precision

Figure 12: Random images generated under W4A8 quantization using different PTQ methods with the Pixart-Σ model on COCO. * indicates that ViDiT-Q uses mixed-precision quantization, where most linear layers are quantized with higher bit-widths.

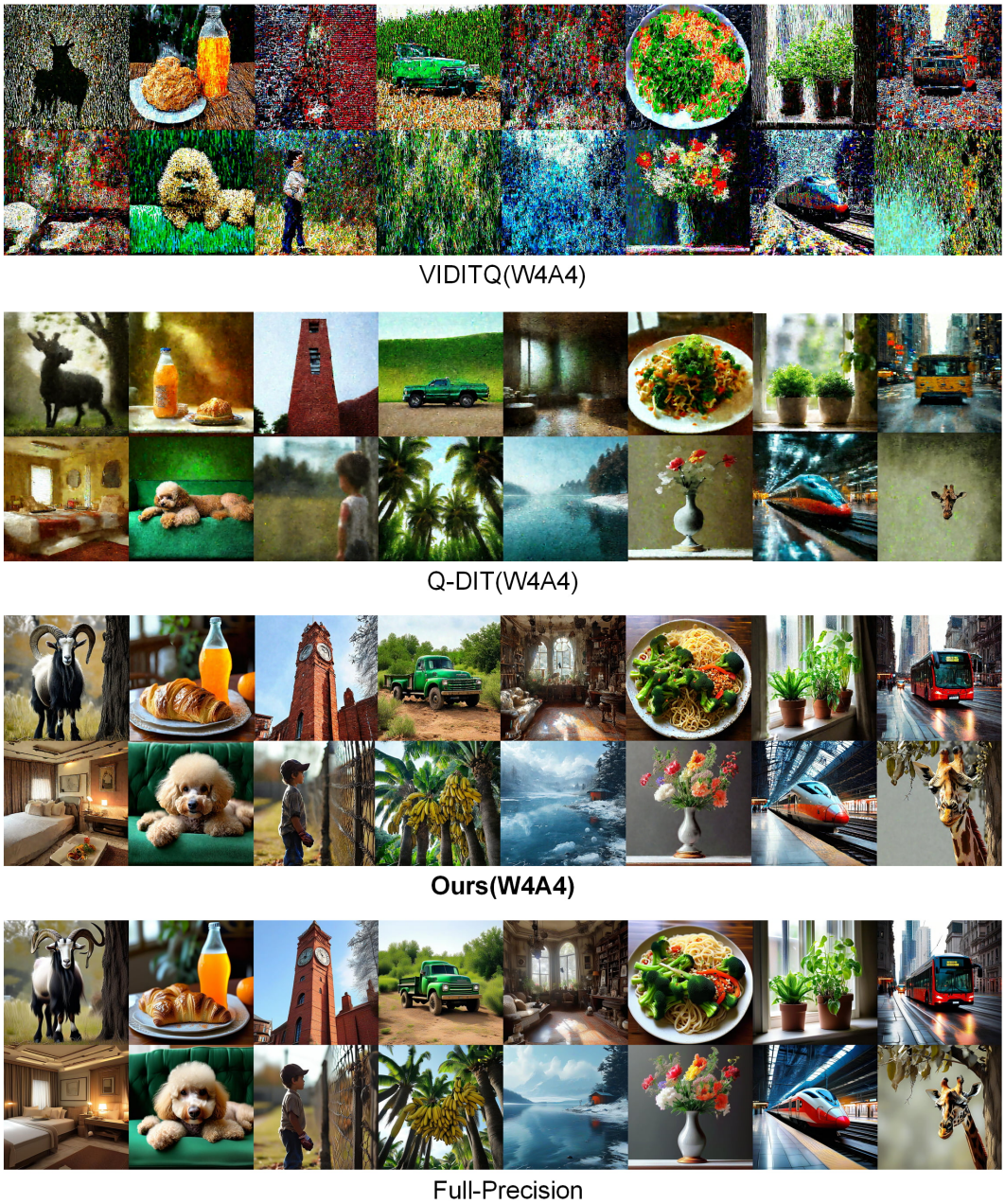


Figure 13: Random images generated under W4A4 quantization using different PTQ methods with the Pixart-Σ model on COCO.

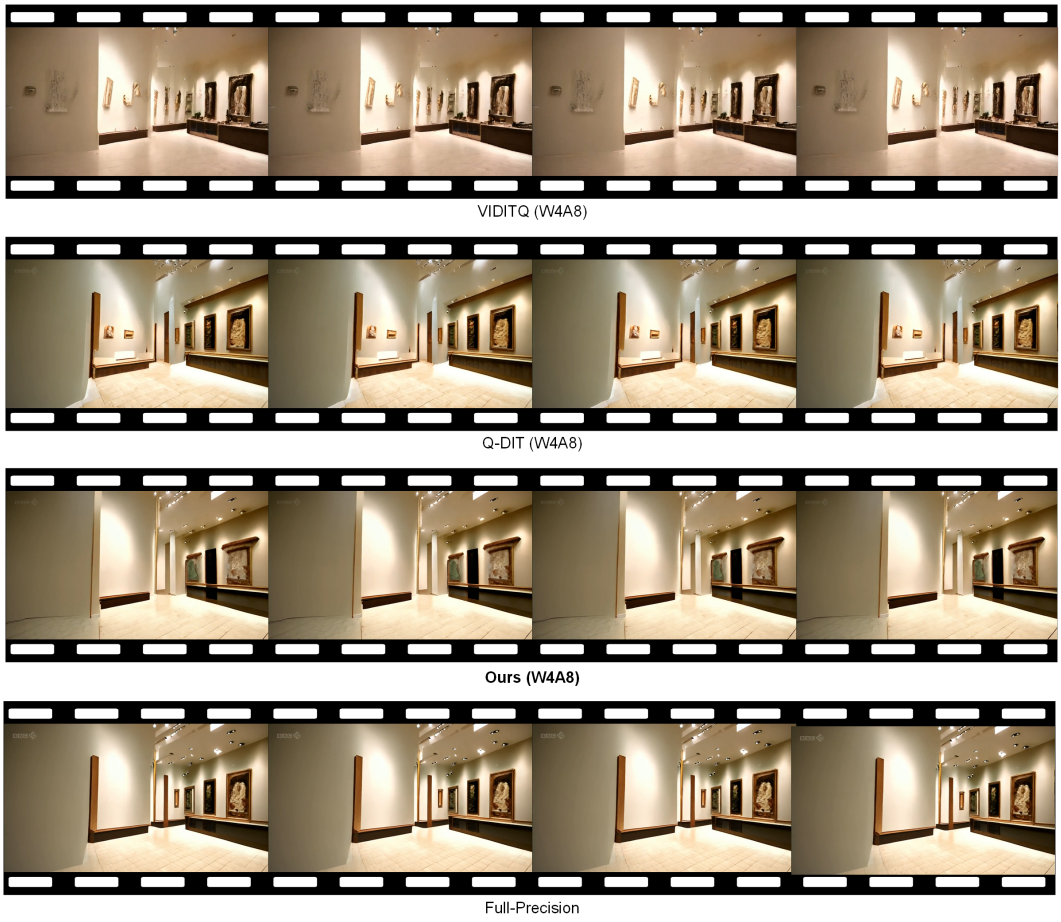


Figure 14: Random video generated under W4A8 quantization using different PTQ methods with the Open-Sora 1.2 model on VBench.

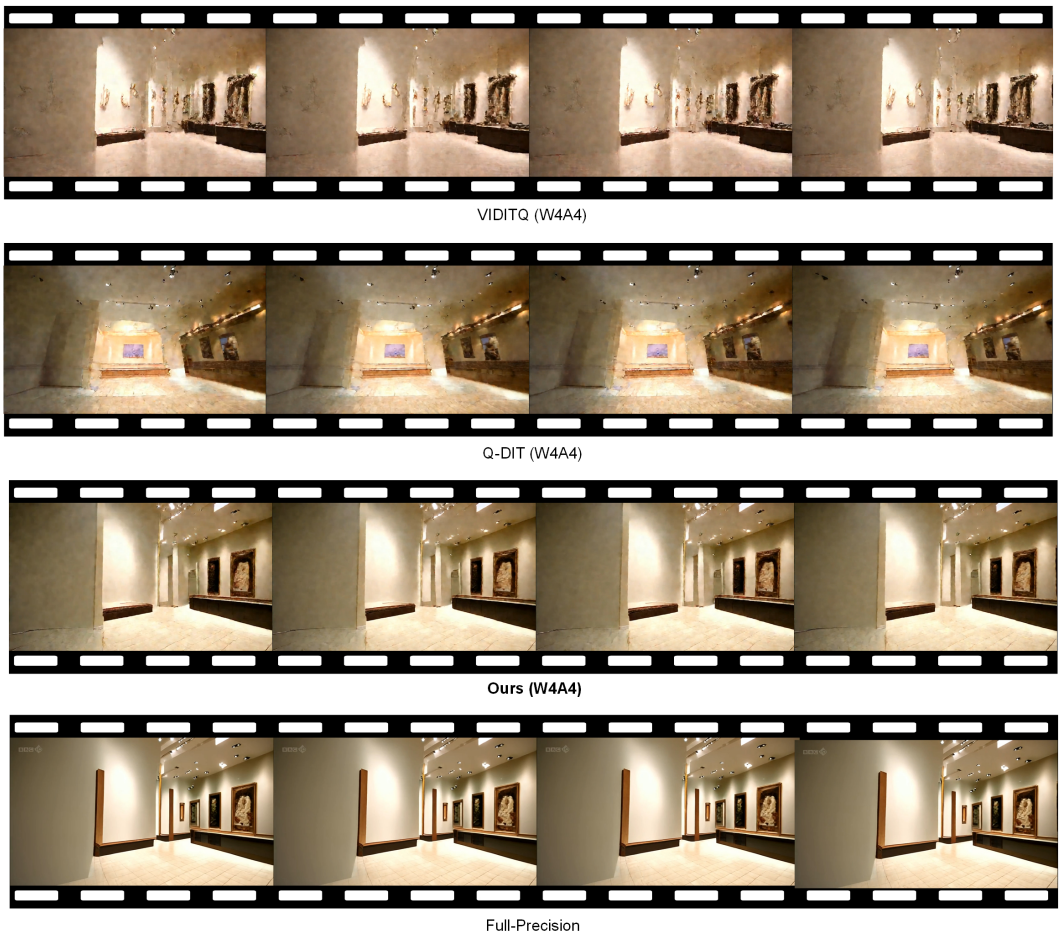


Figure 15: Random video generated under W4A4 quantization using different PTQ methods with the Open-Sora 1.2 model on VBench.