Q-DiT: Accurate Post-Training Quantization for Diffusion Transformers

Lei Chen[1] Yuan Meng[1] Chen Tang[1,2] Xinzhu Ma[2] Jingyan Jiang[3] Xin Wang[1]
Zhi Wang[1] Wenwu Zhu[1]
[1]Tsinghua University [2]MMLab, CUHK [3]Shenzhen Technology University

Abstract

Recent advancements in diffusion models, particularly the architectural transformation from UNet-based models to Diffusion Transformers (DiTs), significantly improve the quality and scalability of image and video generation. However, despite their impressive capabilities, the substantial computational costs of these large-scale models pose significant challenges for real-world deployment. Post-Training Quantization (PTQ) emerges as a promising solution, enabling model compression and accelerated inference for pretrained models, without the costly retraining. However, research on DiT quantization remains sparse, and existing PTQ frameworks, primarily designed for traditional diffusion models, tend to suffer from biased quantization, leading to notable performance degradation. In this work, we identify that DiTs typically exhibit significant spatial variance in both weights and activations, along with temporal variance in activations. To address these issues, we propose Q-DiT, a novel approach that seamlessly integrates two key techniques: automatic quantization granularity allocation to handle the significant variance of weights and activations across input channels, and sample-wise dynamic activation quantization to adaptively capture activation changes across both timesteps and samples. Extensive experiments conducted on ImageNet and VBench demonstrate the effectiveness of the proposed Q-DiT. Specifically, when quantizing DiT-XL/2 to W6A8 on ImageNet (256×256), Q-DiT achieves a remarkable reduction in FID by 1.09 compared to the baseline. Under the more challenging W4A8 setting, it maintains high fidelity in image and video generation, establishing a new benchmark for efficient, high-quality quantization in DiTs. Code is available at https://github.com/Juanerx/Q-DiT.

1. Introduction

Diffusion models [6, 16, 26, 32] have emerged as a powerful base model for various tasks, ranging from computer vision, natural language processing, multi-modal modeling, *etc.* The architectural design of diffusion models has evolved significantly. Traditionally, these models employed UNet [28] architecture due to their efficiency in managing hierarchical feature representations. However, recent advances have shifted the focus towards diffusion transformers (DiTs) [27], and notable examples, including Stable Diffusion 3 [8] and Sora [4], have demonstrated its superior performance and scalability for complex generative tasks.

Despite their success, a significant limitation of DiTs lies in their inherently high latency in the inference. The iterative denoising process, although effective, requires numerous sampling steps, making real-time or large-scale applications computationally intensive. Model quantization offers a particularly promising avenue to reduce inference latency, and Post Training Quantization (PTQ) is particularly appealing for large models as it eliminates the need for retraining. However, the application of quantization techniques to transformer-based diffusion models remains limited. Existing quantization methods for diffusion models [12, 13, 30] primarily focus on UNet architecture and heavily rely on reconstruction-based methods, challenging to scale to large models [21, 25].

In this work, we aim to propose a customized quantization method for DiTs. To achieve this, we first explore the distinct characteristics of DiT models and identify two key issues in DiT quantization: *significant variance of weights and activations across input channels* and *varying activations across different timesteps*. Therefore, we propose Q-DiT, consisting of a fine-grained group quantization strategy and a dynamic activation quantization strategy. These two designs address the aforementioned challenges individually and collaboratively contribute to the proposed quantization framework (see Fig. 1).

In particular, for the first challenge, a promising solution is group quantization [23, 40] which can manage high-magnitude values at the group level. However, we observe the non-monotonicity in group sizes, *e.g.,* reducing group size (increasing group number) does not always lead to better performance. Consequently, we employ an evolutionary search algorithm to configure group sizes for quantization across different model layers. This method utilizes the Fréchet Inception Distance (FID) and Fréchet Video Dis-

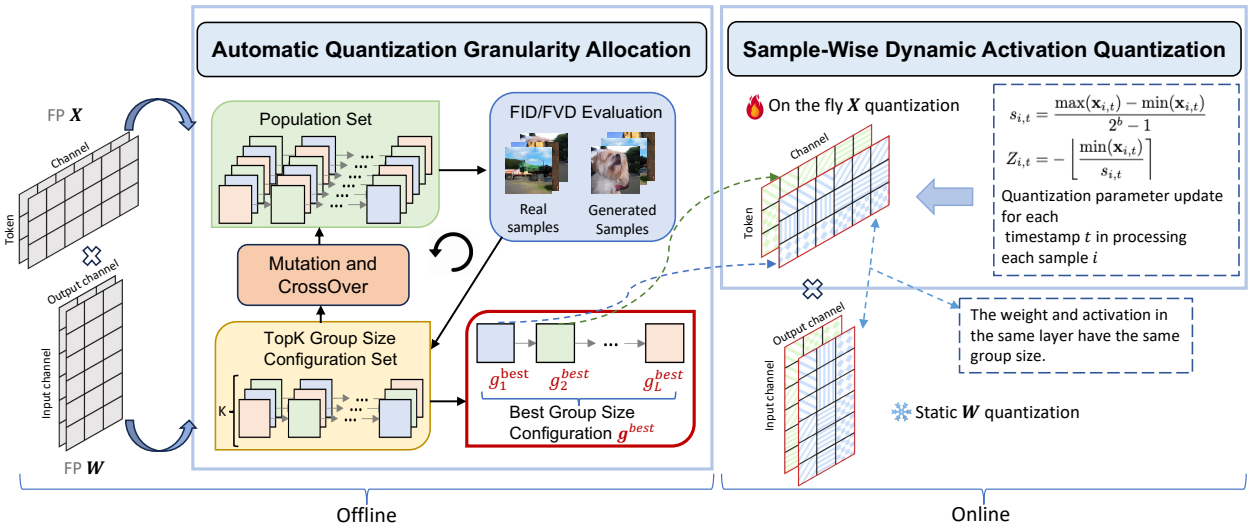


Figure 1. Overview of the proposed Q-DiT. The weights and activations within each layer are quantized with the same group size. Group size configurations allocated for each layer are based on the evolutionary search results, which are guided by the FID/FVD score between the real samples and samples generated by the quantized model. The activations are dynamically quantized during runtime.

tance (FVD) as metrics to directly correlate the quantization effects with the visual quality of generated samples, enabling a more targeted and effective quantization strategy. The evolutionary approach not only identifies the optimal group sizes but also ensures that the quantization process adheres to predefined computational constraints, effectively balancing performance with efficiency.

Furthermore, varying activations across the timesteps, the second challenge, indicates that quantization parameters calibrated at specific timesteps may not generalize well in all timesteps. To address this variability, Q-DiT adopts a sample-wise dynamic activation quantization mechanism, which adapts with sample granularity to the changing distribution of activations throughout the diffusion process. This approach significantly reduces quantization error by adjusting quantization parameters on-the-fly, ensuring high-quality image/video generation with minimal overhead.

In summary, our main contributions are as follows:

- We introduce Q-DiT, an accurate PTQ method designed for DiTs. This method employs fine-grained group quantization to effectively manage input channel variance in both weights and activations, and it adopts sample-wise dynamic activation quantization to adapt to activation variations across different timesteps and samples.
- We identify that the default group size configuration is sub-optimal and propose an evolutionary search strategy to optimize group size allocation, which enhances the efficiency and efficacy of the quantization process.
- Extensive experiments on ImageNet and VBench demonstrate that Q-DiT achieves lossless compression under a W6A8 configuration and minimal degradation under W4A8 for image and video generation, highlighting its superior performance.

2. Related Work

Model quantization. Model quantization is a widely used technique to reduce the model size and accelerate its inference speed by converting the model's weights and activations from high-precision floating-point numbers to lower-precision numbers. There are two primary approaches to quantization: Quantization-Aware Training (QAT) [2, 5, 9] and Post-Training Quantization (PTQ) [21, 25]. QAT integrates the quantization process directly into the fine-tuning phase, leveraging STE [1] to simultaneously optimize quantizer parameters and model parameters during fine-tuning. This approach restores the model's performance degradation caused by quantization. However, QAT is resource-intensive because it necessitates fine-tuning the model on the original training dataset. In contrast, PTQ is far more efficient and practical, as it does not require model retraining. PTQ operates by utilizing a small calibration dataset to adjust the quantization parameters for weights and activations, facilitating significant model compression with minimal effort. Although PTQ is highly efficient, it can result in significant performance degradation when applied to low-bit quantization. Reconstruction-based method [21, 25] tries to minimize performance degradation by reducing the reconstruction error of each layer or each block. Although the reconstruction-based method performs well in CNN, they are not easy to scale up to a large model.

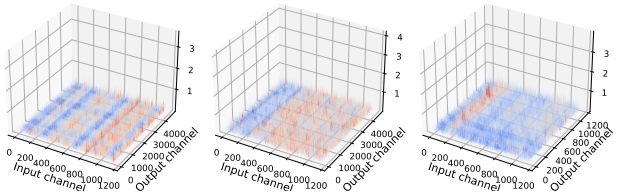
Quantization of transformers. Quantization of transformers has been extensively researched in the contexts of both

Vision Transformers (ViTs) and Large Language Models (LLMs). Specifically, PTQ4ViT [39] proposed the twin uniform quantizer to handle the special distributions of post-softmax and post-GELU activations. RepQ-ViT [22] used scale reparameterization to reduce the quantization error of activations. For LLM, weight-only quantization quantizes the weight to reduce the heavy memory movements to achieve better inference efficiency. GPTQ [10] reduced the bit-width to 4 bits per weight based on approximate second-order information with weight-only quantization. AWQ [23] proposed activation-aware weight quantization to reduce the quantization error of salient weight. On the other hand, weight-activation quantization further enhances inference efficiency by quantizing both weight and activation but has to face activation outliers. LLM.`int8()` [7] reduces the effect of outliers by keeping them in `FP16` with mixed-precision computations. Outlier Suppression [35] reduces the quantization error of activations by using the non-scaling LayerNorm. However, these quantization techniques may not be directly applied to DiTs, due to their diffusion model characteristics.

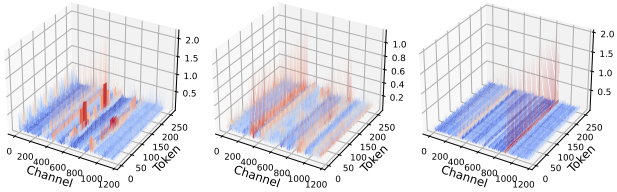
Quantization of diffusion models. Diffusion models tend to have a slow inference speed due to the large number of sampling steps required. Consequently, some recent studies have focused on accelerating these models through quantization techniques. PTQ4DM [30] and Q-diffusion [20] discover activation variance across different denoising steps and adopt reconstruction-based methods for quantization. PTQD [13] finds the correlation between the quantization noise and model output and proposes variance schedule calibration to rectify the uncorrelated part. TDQ [31] utilizes an MLP layer to estimate the activation quantization parameters for each step. TMPQ-DM [33] further reduces the sequence length of timestep along with the quantization to reduce the overall costs. PTQ4DiT [36] introduces a PTQ method tailored for Diffusion Transformers, addressing challenges like extreme channel magnitudes and temporal activation variability using Channel-wise Salience Balancing (CSB) and Spearman's ρ-guided Salience Calibration (SSC), while achieving W4A8 quantization. Comparison table of PTQ methods for diffusion models can be found in the supplementary materials. These methods are unable to handle simultaneously the characteristics of the transformer architecture and the dynamics of activation during denoising process, leading to significant performance drops.

3. Observations of DiT Quantization

We find directly applying recent UNet-based quantization methods to quantize DiTs will lead to significant performance degradation. To understand the underlying reasons, we explore the distinct characteristics of DiT models, particularly how they differ from UNet-based architectures in terms of weight and activation distributions.



(a) **Weight** distribution



(b) **Activation** distribution

Figure 2. Distributions of weights and activations in different layers of DiT-XL/2. The red peaks indicate higher values, while the blue areas represent lower values.

Observation 1: *DiTs exhibit significant variance of weights and activations across input channels.* As shown in Fig. 2a, the variance of weights and activations across input channels is much more significant than the output channels. This variance substantially affects quantization since common methods typically apply channel-wise quantization along the output channel for diffusion models [13, 30]. Besides, we also find outliers persist in specific channels of the activation, as shown in Fig. 2b. This suggests that, if we continue to use tensor-wise quantization, these outliers will significantly impact the quantization parameters, resulting in substantial quantization errors for non-outliers.

Observation 2: *Significant distribution shift of activations across timesteps.* We observe that the distribution of activations in DiT models undergoes significant changes at different timesteps during the denoising process, as demonstrated in Fig. 3 and Fig. 4. Further, we discovered that this temporal shift also exhibits significant variability across different samples. The relevant experimental results are presented in the supplementary materials.

4. Preliminary

We use uniform quantization to quantize both weights and activations in this work, as it is more hardware-friendly [11, 19]. Particularly, uniform quantization divides the range of floating-point values into equally spaced intervals, and each interval is mapped to a discrete value. The uniform quantization function Q that quantize input floating-point tensor $\mathbf{x}$ into b bit integer tensor $\hat{\mathbf{x}}$ can be expressed as:

$$\hat{\mathbf{x}} = Q(\mathbf{x}; b) = s \cdot (\text{clip}(\lfloor \frac{\mathbf{x}}{s} \rceil + Z, 0, 2^b - 1) - Z), \quad (1)$$

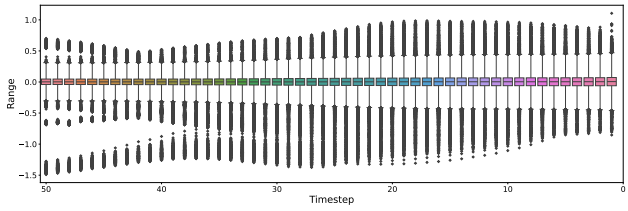


Figure 3. Box plot showing the distribution of activation values across various timesteps (from 50 to 0) for the DiT-XL/2 model when generating one image from ImageNet at 256 × 256 resolution..

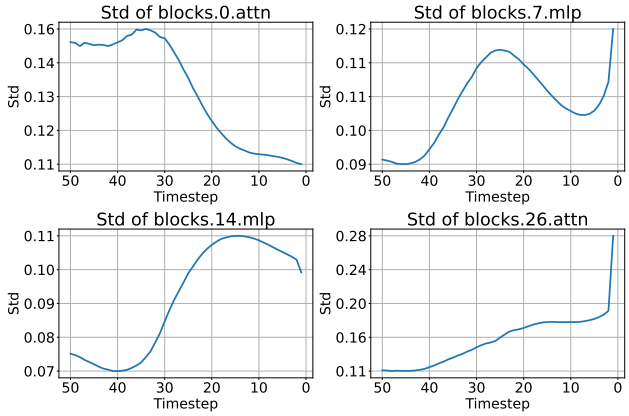


Figure 4. Standard deviation of activations in MLP and attention layers across different blocks over 50 timesteps for DiT-XL/2 when generating one image from ImageNet at 256 × 256 resolution.

where $s = \frac{\max(\mathbf{x})-\min(\mathbf{x})}{2^b-1}$ and $Z = -\left\lfloor \frac{\min(\mathbf{x})}{s} \right\rceil$. Here, $\lfloor \cdot \rceil$ is the round operation, s is the scaling factor and Z denotes the zero point.

5. Method: Q-DiT

As shown in Fig. 1, the proposed Q-DiT involves two novel designs, including automatic quantization granularity allocation and dynamic activation quantization. Here we introduce them in the following parts.

5.1. Automatic Quantization Granularity Allocation

Base solution. A straightforward solution to deal with the input channel-wise variance, as highlighted in **observation 1**, is to apply input channel-wise quantization, using different quantization parameters for each channel. However, this approach compromises computational efficiency during hardware deployment, as it prevents the full utilization of low-precision computation due to the need for repeated intermediate rescaling [3, 38].

Fine-grained group quantization. As discussed in recent LLM quantization work [23, 40], a compromise approach between input channel-wise quantization and tensor quantization is the group quantization. As shown in Fig. 1, the weight and activation matrices are divided into groups, and then we perform quantization for each group separately. Specifically, consider a matrix multiplication $\mathbf{Y} = \mathbf{X}\mathbf{W}$ in a linear layer, where $\mathbf{X} \in \mathbb{R}^{n \times d_{\text{in}}}$ and $\mathbf{W} \in \mathbb{R}^{d_{\text{in}} \times d_{\text{out}}}$. The quantized value for each output element can be obtained by:

$$\hat{\mathbf{Y}}_{i,j} = \sum_{k=0}^{d_{\text{in}}} \hat{\mathbf{X}}_{i,k} \hat{\mathbf{W}}_{k,j} \tag{2}$$

$$= \sum_{u=0}^{d_{\text{in}}/g_{ll}-1} \sum_{v=0}^{g_{ll}} Q_u^{\mathbf{X}}(\mathbf{X}_{i,ug_{ll}+v}) Q_u^{\mathbf{W}}(\mathbf{W}_{ug_{ll}+v,j}), \tag{3}$$

where g_{ll} denotes the group size.

Non-monotonicity in quantization group selection. Ideally, we can improve model performance by reducing the group size (*i.e.,* increase the group number) because finer-grained quantization reduces quantization error. However, as shown in Tab. 1, we have observed that smaller group sizes do not always yield better results, where such existence of non-monotonicity in the quantization group further demonstrates that DiT quantization is rather different compared with LLM and ViT quantizations. For instance, when the group size reduces from 128 to 96, the FID increases by about 11.8%, from 17.87 to 19.97, indicating a degradation in the quality of generated images. This suggests that there is an optimal group size configuration that can achieve better quantization effects with the same average group size or achieve the same quantization results with a larger average group size. Additionally, the sensitivity of each layer in the model varies. By assigning different group sizes to different layers, we can achieve high efficiency and quality in both model performance and image generation.

Automatic group allocation. The primary challenge in allocating group sizes lies in identifying the correlation between the group size of each layer and the final generation performance of the diffusion model. Previous works on mixed precision quantization focus on identifying sensitivity indicators [21, 34] for each layer, such as the MSE between the quantized layer and the full precision layer, and then transforming this into an integer linear programming (ILP) problem for optimization. However, we find that a smaller MSE does not necessarily correspond to a reduced

Table 1. Quantization results with varying group sizes on ImageNet 256×256 and 512×512.

256×256			512×512		
Group	FID ↓	sFID ↓	Group	FID ↓	sFID ↓
128	17.87	20.45	96	20.76	21.97
96	19.97	21.42	64	20.90	22.58

performance degradation for DiT quantization, indicating that the previous methods may be ineffective.

To this end, we directly use the FID as our metric for image generation, defined as follows:

$$L(\mathbf{g}) = \text{FID}(R, G_{\mathbf{g}}). \tag{4}$$

Similarly, for video generation models, we use the FVD as our metric:

$$L(\mathbf{g}) = \text{FVD}(R, G_{\mathbf{g}}), \tag{5}$$

where $\mathbf{g} = \{g_1, g_2, \ldots, g_N\}$ is the layer-wise group size configuration and N is the number of quantized layers. R and $G_{\mathbf{g}}$ denote the real samples and the samples generated by the quantized model, respectively. We then employ an evolutionary algorithm to optimize the following objective function:

$$\mathbf{g}^* = \arg\min_{\mathbf{g}} L(\mathbf{g}), \quad \text{s.t. } B(\mathbf{g}) \leq N_{bitops}, \tag{6}$$

where $B(\cdot)$ is the measurement of bit-operations (BitOps), and N_{bitops} is the predefined threshold.

This approach allows us to better capture the nuanced impacts of group size on quantization performance, leading to improved outcomes in both efficiency and image quality. The algorithm is located in Alg. 1.

5.2. Sample-wise Dynamic Activation Quantization

In UNet-based diffusion quantization, previous studies either allocate a set of quantization parameters for all activations at each timestep [12] or design a multi-layer perceptron (MLP) [31] to predict quantization parameters based on the timestep. However, due to **observation 2**, these methods are not compatible with our fine-grained quantization because assigning quantization parameters to each group at every timestep results in considerable memory overheads. Specifically, for a sampler with 50 timesteps, the memory overhead could reach up to 39% of the full-precision model's size.

Inspired by the recent work in LLM optimization [37], we design an on-the-fly dynamic quantization approach for activations. Specifically, during inference, the quantization parameters for each group of the activations are calculated based on their `min-max` values. For a given sample i at timestep t, the quantization parameters for the activation $\mathbf{x}_{i,t}$ can be expressed as:

$$s_{i,t} = \frac{\max(\mathbf{x}_{i,t}) - \min(\mathbf{x}_{i,t})}{2^b - 1}, \tag{7}$$

$$Z_{i,t} = -\left\lfloor \frac{\min(\mathbf{x}_{i,t})}{s_{i,t}} \right\rceil. \tag{8}$$

Furthermore, we integrate the dynamic quantization with `min-max` computation into the prior operator, which can benefit from the operator fusion and the overhead becomes negligible compared to the costly matrix multiplications in transformer blocks.

Algorithm 1: Automatic quantization granularity allocation of Q-DiT

Input: Group size search space $\mathcal{S}_g$; number of layers L; population size N_p; iterations N_{iter}; mutation probability p; Constraint N_{bitops}

Initialize population $\mathcal{P} = \{\mathbf{g}^j\}_{j=1}^{N_p}$, where each element in configuration $\mathbf{g}^j \in \mathbb{R}^L$ is randomly selected from $\mathcal{S}_g$;

Initialize TopK candidate set $\mathcal{S}_{\text{TopK}} = \emptyset$;

for $t = 1, 2, \ldots, N_{iter}$ **do**

  for $i = 1, 2, \ldots, N_p$ **do**

    Calculate FID (or FVD) for each configuration $\mathbf{g}^j$ based on Eq. 4 (or Eq. 5);

  Update $\mathcal{S}_{\text{TopK}}$ with K configurations, according to ranked FID (or FVD) scores;

  Clear population $\mathcal{P} = \emptyset$;

  repeat

    $\mathbf{g}_{cross} = \text{CrossOver}(\mathcal{S}_{\text{TopK}})$ with probability $1 - p$;

    Append $\mathbf{g}_{cross}$ to $\mathcal{P}$ if $B(\mathbf{g}_{cross}) < N_{bitops}$;

  until $|\mathcal{P}| = N_p/2$;

  repeat

    $\mathbf{g}_{mutate} = \text{Mutate}(\mathcal{S}_{\text{TopK}})$ with probability p;

    Append $\mathbf{g}_{mutate}$ to $\mathcal{P}$ if $B(\mathbf{g}_{mutate}) < N_{bitops}$;

  until $|\mathcal{P}| = N_p$;

Get the best group size configuration $\mathbf{g}^{\text{best}}$, and use it to quantize the model;

return quantized model

6. Experiments

6.1. Experimental Setup

Image generation. We first evaluate our Q-DiT on the image generation task, closely following the evaluation setting used in DiT [27]. We use the pre-trained DiT-XL/2 models with image resolutions of 256×256 and 512×512, converting them to FP16 as our full precison baseline model. For fast and accurate sampling, we adopt the DDIM sampler [32] with 50 and 100 sampling steps. Performance is also evaluated both with and without classifier-free guidance [15]. Note that "cfg" denotes the classifier-free guidance scale. We sample 10K images for both ImageNet 256×256 and ImageNet 512×512 in each setting, and employ four metrics in our experiments, including Fréchet Inception Distance (FID) [14], spatial FID (sFID) [29], Inception Score (IS), and Precision.

Video generation. We also evaluate Q-DiT on video generation, with the STDiT3 model from the Open-Sora project [41]. Specifically, we sample five 2-second videos for each

Table 2. Results on image generation. We show the quantization results of DiT-XL/2 on ImageNet 256×256 and 512×512. 'W/A' indicates the bit-width of weight and activation, respectively.

Model	Bit-width (W/A)	Method	Size (MB)	FID ↓	sFID ↓	IS ↑	Precision ↑
DiT-XL/2 256×256 (steps = 100)	16/16	FP	1349	12.40	19.11	116.68	0.6605
	6/8	PTQ4DM	508	17.86	25.33	92.24	0.6077
		RepQ-ViT	508	27.74	20.91	63.41	0.5600
		TFMQ-DM	508	22.33	27.44	72.74	0.5869
		PTQ4DiT	508	15.21	21.34	105.03	0.6440
		G4W+P4A	520	16.72	24.61	100.09	0.6123
		Ours	518	**12.21**	**18.48**	**117.75**	**0.6631**
	4/8	PTQ4DM	339	213.66	85.11	3.26	0.0839
		RepQ-ViT	339	224.14	81.24	3.25	0.0373
		TFMQ-DM	339	143.47	61.09	5.61	0.0497
		PTQ4DiT	339	28.90	34.56	65.73	0.4931
		G4W+P4A	351	25.48	25.57	73.46	0.5392
		Ours	347	**15.76**	**19.84**	**98.78**	**0.6395**
DiT-XL/2 256×256 (steps = 100 cfg = 1.5)	16/16	FP	1349	5.31	17.61	245.85	0.8077
	6/8	PTQ4DM	508	8.41	25.56	196.73	0.7622
		RepQ-ViT	508	10.77	18.53	163.11	0.7264
		TFMQ-DM	508	8.87	23.52	194.08	0.7737
		PTQ4DiT	508	5.34	18.48	209.90	**0.8047**
		G4W+P4A	520	6.41	19.52	225.48	0.7705
		Ours	518	**5.32**	**17.40**	**243.95**	0.8044
	4/8	PTQ4DM	339	215.68	86.63	3.24	0.0741
		RepQ-ViT	339	226.60	77.93	3.61	0.0337
		TFMQ-DM	339	141.90	56.01	6.24	0.0439
		PTQ4DiT	339	7.75	22.01	190.38	0.7292
		G4W+P4A	351	7.66	20.76	193.76	0.7261
		Ours	347	**6.40**	**18.60**	**211.72**	**0.7609**
DiT-XL/2 512×512 (steps = 50)	16/16	FP	1349	16.01	20.50	97.79	0.7481
	6/8	PTQ4DM	508	21.22	20.11	80.07	0.7131
		RepQ-ViT	508	19.67	22.35	75.78	0.7082
		TFMQ-DM	508	20.99	22.01	71.08	0.6918
		PTQ4DiT	508	19.42	21.94	77.35	0.7024
		G4W+P4A	520	19.55	22.43	85.56	0.7158
		Ours	517	**16.21**	**20.41**	**96.78**	**0.7478**
	4/8	PTQ4DM	339	131.66	75.79	11.54	0.1847
		RepQ-ViT	339	105.32	65.63	18.01	0.2504
		TFMQ-DM	339	80.70	59.34	29.61	0.2805
		PTQ4DiT	339	35.82	28.92	48.62	0.5864
		G4W+P4A	351	26.58	24.14	70.24	0.6655
		Ours	348	**21.59**	**22.26**	**81.80**	**0.7076**
DiT-XL/2 512×512 (steps = 50 cfg = 1.5)	16/16	FP	1349	6.27	18.45	204.47	0.8343
	6/8	PTQ4DM	508	9.84	26.57	164.91	0.8215
		RepQ-ViT	508	8.30	19.19	158.80	0.8153
		TFMQ-DM	508	8.34	17.94	162.16	0.8262
		PTQ4DiT	508	7.69	18.86	178.34	0.8121
		G4W+P4A	520	7.28	19.62	185.92	0.8143
		Ours	517	**6.24**	**18.36**	**202.48**	**0.8341**
	4/8	PTQ4DM	339	88.45	50.80	26.79	0.3206
		RepQ-ViT	339	79.69	49.76	29.46	0.3413
		TFMQ-DM	339	54.61	44.27	58.77	0.4215
		PTQ4DiT	339	11.69	22.86	117.34	0.7121
		G4W+P4A	351	9.98	20.76	156.07	0.7840
		Ours	347	**7.82**	**19.60**	**174.18**	**0.8127**

prompt in the VBench prompt suite [18] using a 30-step rectified flow scheduler with a cfg scale of 7.0. The results are assessed across 16 dimensions provided by VBench.

Baselines. We compare Q-DiT with five strong baselines:

1) PTQ4DM [30]: A method specifically designed for UNet-based diffusion models, focusing on calibration of

Table 3. Results on video generation. We show the quantization results of STDiT3 on VBench. Higher metrics indicate better performance.

Method	Bit-width (W/A)	Subject Consistency	Overall Consistency	Temporal Style	Appearance Style	Scene	Spatial Relationship	Color	Human Action
FP	16/16	0.9522	0.2667	0.2507	0.2352	0.4094	0.3441	0.7864	0.8680
G4W+P4A	4/8	0.9444	0.2628	0.2489	0.2344	**0.3924**	0.3265	0.7657	0.8600
Ours	4/8	**0.9498**	**0.2663**	**0.2511**	**0.2346**	0.3871	**0.3810**	**0.7947**	**0.8620**

Method	Bit-width (W/A)	Multiple Objects	Object Class	Imaging Quality	Aesthetic Quality	Dynamic Degree	Motion Smoothness	Temporal Flickering	Background Consistency
FP	16/16	0.4143	0.8383	0.5829	0.5173	0.6139	0.9855	0.9917	0.9678
G4W+P4A	4/8	0.3540	0.8225	0.5730	0.5018	0.5639	0.9849	0.9895	0.9651
Ours	4/8	**0.3904**	**0.8475**	**0.5812**	**0.5160**	**0.6167**	**0.9859**	**0.9915**	**0.9687**

activations.

2) RepQ-ViT [22]: A technique developed for the quantization of ViTs, aiming to reduce quantization errors in transformer activations.
3) TFMQ-DM [17]: A PTQ framework specifically developed for diffusion models to preserve temporal features during quantization.
4) PTQ4DiT [36]: A tailored PTQ approach for DiTs that addresses the quantization challenges through CSB and SSC.
5) G4W+P4A: A robust baseline we build in this work for both video and image generation tasks, utilizing GPTQ [10] for weight quantization and PTQ4DM for activation quantization.

Others. Q-DiT applies asymmetric quantization for both weights and activations, and uses GPTQ [10] for weight quantization. A default group size of 128 is adopted, with optimal group size allocation for each layer determined through evolutionary search. The search space for group size $\mathcal{S}_g$ is $\{32, 64, 128, 192, 288\}$. Note that the group size for weights and activations within the same layer should be the same.

6.2. Main Results

Image generation results. The quantitative results for image generation are shown in Tab. 2. Specifically, in experiments conducted on ImageNet at a resolution of 256×256, the PTQ4DM, RepQ-ViT, TFMQ-DM, PTQ4DiT and G4W+P4A methods exhibit significant performance degradation at a bit-width of W6A8. In contrast, Q-DiT shows marked improvements over PTQ4DM under the W6A8 setting, effectively minimizing the impact of quantization. Notably, Q-DiT is closely aligned with the full-precision configuration, achieving an FID of 12.21 and an IS of 117.75. These results highlight the effectiveness of our method in achieving near-lossless compression in the W6A8 quantization setting. When the bit-width is reduced to W4A8, the performance disparities among the methods become more pronounced. In particular, the other five baselines have severe performance degradation, while our method substantially outperforms them, dramatically reducing quantization loss with an FID of 15.76 and an IS of 98.78. This demonstrates a significant preservation of quality and diversity at lower bit-widths, highlighting the robustness of our approach under stringent quantization constraints. Across varying steps (100 and 50) and classifier-free guidance scales, our method consistently shows superior performance, closely emulating the full-precision model metrics. The evaluation on the ImageNet 512×512 dataset demonstrates consistent trends with the 256×256 dataset, indicating that Q-DiT can also perform well in high-resolution image generation. Visual demonstrations in Fig. 5 further illustrate that our method maintains superior image generation quality compared to the baseline.

Video generation results. Tab. 3 shows the results for video generation. Under a stringent W4A8 quantization setting, our method consistently outperforms G4W+P4A in 15 out of 16 metrics, exhibiting minimal degradation compared to the full-precision model. This indicates that our method performs well in terms of preserving video quality and maintaining video-condition consistency.

6.3. Ablation Studies

To evaluate the effectiveness of each proposed component, we conduct ablation studies on ImageNet 256×256 with the DiT-XL/2 model. The sampling steps and classifier-free guidance scale are set to 100 and 1.5, as detailed in Tab. 4.

Incremental analysis of Q-DiT. We begin our assessment with a round-to-nearest (RTN) baseline, which simply rounds weights and activations to the nearest available quantization level. Under the W4A8 configuration, RTN demonstrates significantly low performance across all metrics. Enhancing RTN by adjusting the quantization granularity to a group size of 128 markedly improves the results. The introduction of dynamic activation quantization led to a significant boost in generation quality, evidenced by an FID of 6.64, an sFID of 19.29, and an IS of 211.27. By further incorporating group size allocation, our approach achieves an impressive FID of 6.40, approaching the performance of the full-precision model.

Figure 5. Qualitative results. Samples generated by G4W+P4A (one of our baselines) and Q-DiT with W4A8 on ImageNet 256×256 and ImageNet 512×512. For each example (a-e), the image generated by G4W+P4A shows notable artifacts and distortions. In contrast, our method produces cleaner and more realistic images, with better preservation of textures.

Table 4. Incremental analysis of individual components in our proposed method under the W4A8 setting.

	FID ↓	sFID ↓	IS ↑
FP (W16A16)	5.31	17.61	245.85
W4A8 RTN	225.50	88.54	2.96
+ Group size 128	13.77	27.41	146.93
+ Sample-wise Dynamic activation quantization	6.64	19.29	211.27
+ Automatic quantization granularity allocation	**6.40**	**18.60**	**211.72**

Comparisons of dynamic activation quantization methods. We also conducted experiments on activation quantization, as shown in Tab. 5. Both TFMQ-DM and our method are quantized to W16A8 to isolate and compare the impact of activation quantization on overall performance. Our method achieves an FID of 5.34, demonstrating a significant improvement over TFMQ-DM, which has an FID of 7.74. This highlights the effectiveness of our sample-aware dynamic activation quantization in maintaining model accuracy while reducing performance degradation compared to TFMQ-DM.

Comparisons of search methods. Furthermore, we also evaluate the effectiveness of the proposed search method (Alg. 1) used in group quantization and show the results in Tab. 6. We can find the proposed method significantly performs better than ILP method [24], Hessian-based search method [21], and the baseline, which demonstrates the effectiveness the our method.

7. Conclusion

Our study presents Q-DiT, a novel post-training quantization framework designed for DiTs. To address the significant spatial variance of weights and activations in input channels, we introduced an automatic quantization granularity allocation method. Furthermore, to manage variations in activation ranges across different timesteps, we implemented dynamic activation quantization that adaptively adjusts quantization parameters during runtime. Extensive experiments have underscored the effectiveness of our approach, showcasing its superiority over existing baselines. Notably, even when quantizing the model to W4A8 on the ImageNet 256×256 dataset, the FID increased by only 1.09.

Limitations and future work. One of the primary limitations of the current Q-DiT approach is its reliance on evolutionary algorithms to determine the optimal group size configuration for quantization. This process is computationally expensive and time-consuming, increasing the overall cost and duration of optimization. We plan to optimize this part in the future work.

Table 5. Comparisons of dynamic activation quantization methods with W16A8 setting. TFMQ-DM is a method for timestep-aware activation quantization, whereas our approach is both timestep-wise and sample-wise.

Method	FID ↓	sFID ↓	IS ↑	Precision↑
FP (W16A16)	5.31	17.61	245.85	0.8077
TFMQ-DM	7.74	19.23	204.56	0.7765
Ours	**5.34**	**17.44**	**245.24**	**0.8048**

Table 6. Comparisons of the proposed search method and potential counterparts.

Search method	FID ↓	sFID ↓	IS ↑	Precision↑
Group size = 128	6.64	19.29	211.27	0.7548
ILP	6.71	19.20	205.54	0.7538
Hessian-based	7.38	19.41	197.48	0.7385
Ours	**6.40**	**18.60**	**211.72**	**0.7609**

References

[1] Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013. 2

[2] Yash Bhalgat, Jinwon Lee, Markus Nagel, Tijmen Blankevoort, and Nojun Kwak. Lsq+: Improving low-bit quantization through learnable offsets and better initialization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pages 696–697, 2020. 2

[3] Yelysei Bondarenko, Markus Nagel, and Tijmen Blankevoort. Understanding and overcoming the challenges of efficient transformer quantization. *arXiv preprint arXiv:2109.12948*, 2021. 4

[4] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1

[5] Jungwook Choi, Zhuo Wang, Swagath Venkataramani, Pierce I-Jen Chuang, Vijayalakshmi Srinivasan, and Kailash Gopalakrishnan. Pact: Parameterized clipping activation for quantized neural networks. *arXiv preprint arXiv:1805.06085*, 2018. 2

[6] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10850–10869, 2023. 1

[7] Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3. int8 (): 8-bit matrix multiplication for transformers at scale. *Advances in Neural Information Processing Systems*, 35:30318–30332, 2022. 3

[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. *arXiv preprint arXiv:2403.03206*, 2024. 1

[9] Steven K Esser, Jeffrey L McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S Modha. Learned step size quantization. *arXiv preprint arXiv:1902.08153*, 2019. 2

[10] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022. 3, 7

[11] Song Han, Huizi Mao, and William J Dally. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015. 3

[12] Yefei He, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Efficientdm: Efficient quantization-aware fine-tuning of low-bit diffusion models. *arXiv preprint arXiv:2310.03270*, 2023. 1, 5

[13] Yefei He, Luping Liu, Jing Liu, Weijia Wu, Hong Zhou, and Bohan Zhuang. Ptqd: Accurate post-training quantization for diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024. 1, 3

[14] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017. 5

[15] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 5

[16] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[17] Yushi Huang, Ruihao Gong, Jing Liu, Tianlong Chen, and Xianglong Liu. Tfmq-dm: Temporal feature maintenance quantization for diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7362–7371, 2024. 7

[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. 6

[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. 3

[20] 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. 3

[21] 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. *arXiv preprint arXiv:2102.05426*, 2021. 1, 2, 4, 8

[22] 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. 3, 7

[23] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for llm compression and acceleration. *arXiv preprint arXiv:2306.00978*, 2023. 1, 3, 4

[24] Jaehyeon Moon, Dohyung Kim, Junyong Cheon, and Bumsub Ham. Instance-aware group quantization for vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16132–16141, 2024. 8

[25] Markus Nagel, Rana Ali Amjad, Mart Van Baalen, Christos Louizos, and Tijmen Blankevoort. Up or down? adaptive rounding for post-training quantization. In *International*

Conference on Machine Learning, pages 7197–7206. PMLR, 2020. 1, 2

[26] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, pages 8162–8171. PMLR, 2021. 1

[27] 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. 1, 5

[28] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015. 1

[29] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NeurIPS*, 2016. 5

[30] Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *CVPR*, 2023. 1, 3, 6

[31] Junhyuk So, Jungwon Lee, Daehyun Ahn, Hyungjun Kim, and Eunhyeok Park. Temporal dynamic quantization for diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024. 3, 5

[32] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021. 1, 5

[33] Haojun Sun, Chen Tang, Zhi Wang, Yuan Meng, Xinzhu Ma, Wenwu Zhu, et al. Tmpq-dm: Joint timestep reduction and quantization precision selection for efficient diffusion models. *arXiv preprint arXiv:2404.09532*, 2024. 3

[34] Chen Tang, Kai Ouyang, Zhi Wang, Yifei Zhu, Wen Ji, Yaowei Wang, and Wenwu Zhu. Mixed-precision neural network quantization via learned layer-wise importance. In *European Conference on Computer Vision*, pages 259–275. Springer, 2022. 4

[35] Xiuying Wei, Yunchen Zhang, Xiangguo Zhang, Ruihao Gong, Shanghang Zhang, Qi Zhang, Fengwei Yu, and Xianglong Liu. Outlier suppression: Pushing the limit of low-bit transformer language models. *Advances in Neural Information Processing Systems*, 35:17402–17414, 2022. 3

[36] Junyi Wu, Haoxuan Wang, Yuzhang Shang, Mubarak Shah, and Yan Yan. Ptq4dit: Post-training quantization for diffusion transformers. In *NeurIPS*, 2024. 3, 7

[37] Xiaoxia Wu, Zhewei Yao, and Yuxiong He. Zeroquant-fp: A leap forward in llms post-training w4a8 quantization using floating-point formats. *arXiv preprint arXiv:2307.09782*, 2023. 5

[38] 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 *Proceedings of the 40th International Conference on Machine Learning*, 2023. 4

[39] Zhihang Yuan, Chenhao Xue, Yiqi Chen, Qiang Wu, and Guangyu Sun. Ptq4vit: Post-training quantization for vision transformers with twin uniform quantization. In *European conference on computer vision*, pages 191–207. Springer, 2022. 3

[40] 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. *Proceedings of Machine Learning and Systems*, 6:196–209, 2024. 1, 4

[41] Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all, 2024. 5