Q-Diffusion: Quantizing Diffusion Models (Supplementary Materials)

Xiuyu Li[1] Yijiang Liu[2] Long Lian[1] Huanrui Yang[1] Zhen Dong[1]
Daniel Kang[3] Shanghang Zhang[4] Kurt Keutzer[1]
[1]UC Berkeley [2]Nanjing University [3]University of Illinois Urbana-Champaign [4]Peking University

This document provides additional visualizations and experimental results to support the main paper.

A. Extended Experimental Settings

A.1. Implementation Details

We describe the implementation and compute details of the experiments in this section. We adapt the official implementation for DDIM [11] [1] and Latent Diffusion [9] [2]. For Stable Diffusion, we use the CompVis codebase [3] and its v1.4 checkpoint. We use the torch-fidelity library [4] to evaluate FID and IS scores as done in [9]. We use 100 denoising time steps for DDIM CIFAR-10. We select 200 and 500 denoising time steps for LSUN-Bedrooms and LSUN-Churches respectively, which are the configurations that achieve the best results provided by [9]. For text-guided image generation with Stable Diffusion, we choose the default PNDM sampler with 50 time steps.

For quantization experiments, we quantize all weights and activations involved in matrix multiplications, but leave activation functions (e.g. `SoftMax`, `SiLU`) and normalization layers (e.g. `GroupNorm`) running with full precision. Additionally, for Linear Quantization [13] and SQuant [2] experiments, we dynamically update the activation quantizers throughout the image generation process to establish the strongest possible baselines, which explains why sometimes their results are better than weight-only quantization cases. For text-guided image generation with Stable Diffusion, we find that attention matrices in cross attentions are difficult to quantize after the `SoftMax` and may have considerable influences on the generation quality, so we utilize INT16 mixed-precision for attention scores under W8A8 & W4A8 cases, while q, k, v matrices are still quantized down to 8-bit. No special modifications or mixed precision are done for other experiments.

A.2. Text-guided Image Generation Calibration Dataset Generation Details

For text-guided image generation with Stable Diffusion, we need to also include text conditioning in the calibration dataset. We randomly sample text prompts from the MS-COCO dataset, and for each prompt we add a pair of data with both a conditional feature $\mathbf{c}_t$ and an unconditional feature $\mathbf{uc}_t$ derived from the prompt. This updated calibration dataset creation process is described by Algorithm 1. Note that we ignore showing the corresponding time embedding $\mathbf{t}_t$ for each time step t is also added with the sample in Algorithm 1 of the main paper.

A.3. Hyperparameters

Here we provide the hyperparameters used for our Q-Diffusion calibration in Table 4.

For all unconditional generation experiments, we keep the total calibration dataset size as 5120 and the amount of calibration data per sampling step as 256. Q-Diffusion is able to obtain high-quality images with insignificant fidelity loss by uniformly sampling from 20 time steps without any hyperparameters tuning. For text-guided image generation with Stable Diffusion, the introduction of text conditioning makes activation quantization harder, thus we sample a larger calibration dataset using all time steps.

[1] https://github.com/ermongroup/ddim
[2] https://github.com/CompVis/latent-diffusion
[3] https://github.com/CompVis/stable-diffusion
[4] https://github.com/toshas/torch-fidelity

Algorithm 1 Q-Diffusion Calibration for Text-guided Image Generation

```
Require: Pretrained full precision diffusion model and the quantized diffusion model $[W_\theta, \hat{W}_\theta]$
Require: Empty calibration dataset $\mathcal{D}$
Require: Number of denoising sampling steps $T$
Require: Calibration sampling interval $c$, amount of calibration data per sampling step $n$
  for $t = 1, \dots, T$ time step do
    if t % c = 0 then
      Sample $2n$ intermediate inputs $(\mathbf{x}_t^{(1)}, \mathbf{c}_t^{(1)}, \mathbf{t}_t^{(1)}), (\mathbf{x}_t^{(1)}, \mathbf{uc}_t^{(1)}, \mathbf{t}_t^{(1)}), \dots, (\mathbf{x}_t^{(n)}, \mathbf{c}_t^{(n)}, \mathbf{t}_t^{(n)}), (\mathbf{x}_t^{(n)}, \mathbf{uc}_t^{(n)}, \mathbf{t}_t^{(n)})$ randomly at $t$ from $W_\theta$ and add them to $\mathcal{D}$
    end if
  end for
  for all $i = 1, \dots, N$ blocks do
    Update the weight quantizers of the $i$-th block in $\hat{W}_\theta$ with $\mathcal{D}$ and $W_\theta$
  end for
  if do activation quantization then
    for all $i = 1, \dots, N$ blocks do
      Update the activation quantizers step sizes of the $i$-th block with $\hat{W}_\theta$, $W_\theta$, $\mathcal{D}$.
    end for
  end if
```

Experiment	T	c	n	N
DDIM CIFAR-10	100	5	256	5120
LDM-4 LSUN-Bedroom	200	10	256	5120
LDM-8 LSUN-Church	500	25	256	5120
Stable Diffusion (weights only)	50	2	256 (128)	6400
Stable Diffusion (weights & activations)	50	1	256 (128)	12800

Table 4: Hyperparameters for all experiments, including the number of denoising time steps T, intervals for sampling calibration data c, amount of calibration data per sampling step n, and the size of calibration dataset N. Note that for Stable Diffusion with classifier-free guidance, every text prompt (128 in total for each sampling step) will add a pair of two samples to the calibration dataset.

B. Layer-wise Activations Distribution in DDIM and LDM

We analyze the ranges of activation values across all time steps in DDIM on CIFAR-10, LDM on LSUN-Bedroom and LSUN-Church, and Stable Diffusion on the text-to-image task. Figure 13 shows that all Conv layers with residual connections in DDIM exhibit noticeably wider activation ranges. Specifically, the first Conv layer can reach up to 1200 and others with residual connections have ranges larger than 100, whereas the majority of the layers without residual connections have ranges less than 50. Similar results could be observed from Stable Diffusion with the text-to-image generation task with COCO captions as well as LSUN-Bedroom in latent diffusion. On the other hand, all layers in LDM on LSUN-Church share relatively uniform activation distributions, with ranges < 15.

Furthermore, Figure 14 illustrates that the distribution of activation values of multiple layers in DDIM on CIFAR-10 varies significantly across different time steps.

C. Quantitative Evaluation on Text-guided Image Generation

To quantitatively evaluate the extent of the impacts on generation performance induced by quantization, we follow the practice in [9], Stable Diffusion v1-5 model card [5], and Diffusers library [6] to generate 10k images using prompts from the MS-COCO [5] 2017-val dataset. Subsequently, we compute the FID [4] and CLIP score [3] against the 2017-val dataset. The ViT-B/16 is used as the backbone when computing the CLIP score. Results are illustrated in Figure 15.

[5] https://huggingface.co/runwayml/stable-diffusion-v1-5

[6] https://huggingface.co/docs/diffusers/main/en/conceptual/evaluation

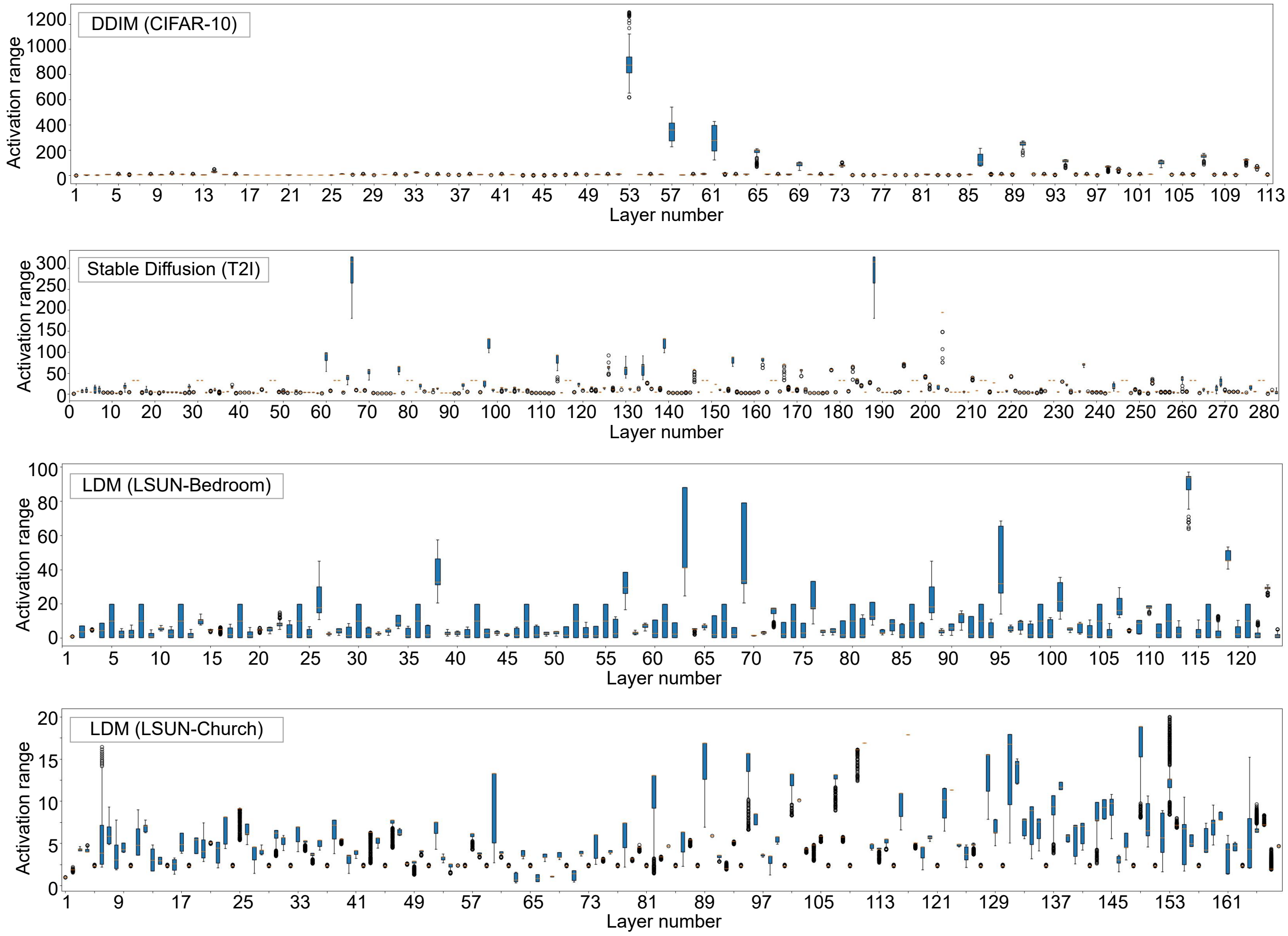


Figure 13: Activation ranges of FP32 outputs across layers averaging among all time steps. The figures, from top to bottom, are respectively DDIM, Stable Diffusion, LDM-Bedroom, and LDM-Church.

Our Q-Diffusion has minimal quality degradation in generated images measured by these two metrics under all settings, while the direct Linear Quantization incurs significant quality drops, especially when the activations are also quantized. Note that FID and CLIP scores on COCO may not be good metrics that align well with human preferences; we do observe that slight artifacts appear more often on images generated with models that have both weights and activations quantized by Q-Diffusion, while these are not reflected by the FID results.

D. Study of Combining with Fast Samplers

Another line of work to speed-up diffusion models is to find shorter and more effective sampling trajectories in order to reduce the number of steps in the denoising process. These approaches tackle an orthogonal factor that Q-Diffusion is addressing, indicating that there's great potential to design a method to take advantage of both directions. Here we investigate if Q-Diffusion can be combined with DPM-Solver [6, 7], a fast high-order solver for diffusion ODEs that can greatly bring down the number of steps required for generation. For unconditional generation, we use a 3rd-order DPM-Solver++ as suggested by the authors, and sample for 50 time steps, which is the number of steps required to get a converged sample. For text-guided image generation with Stable Diffusion, we use 2nd-order DPM-Solver++ with 20 time steps. We directly apply this DPM-Solver++ sampler to our INT4 quantized model. Results are shown in Table 6 and Figure 16.

Q-Diffusion only has a minor performance drop when only weights are quantized. The generation quality degrades under W4A8 precision, but all Q-Diffusion results still outperform Linear Quant and SQuant with 100, 200, and 500 steps for CIFAR-10, LSUN-Bedrooms, and LSUN-Churches respectively. The reason is likely due to the distribution of activations

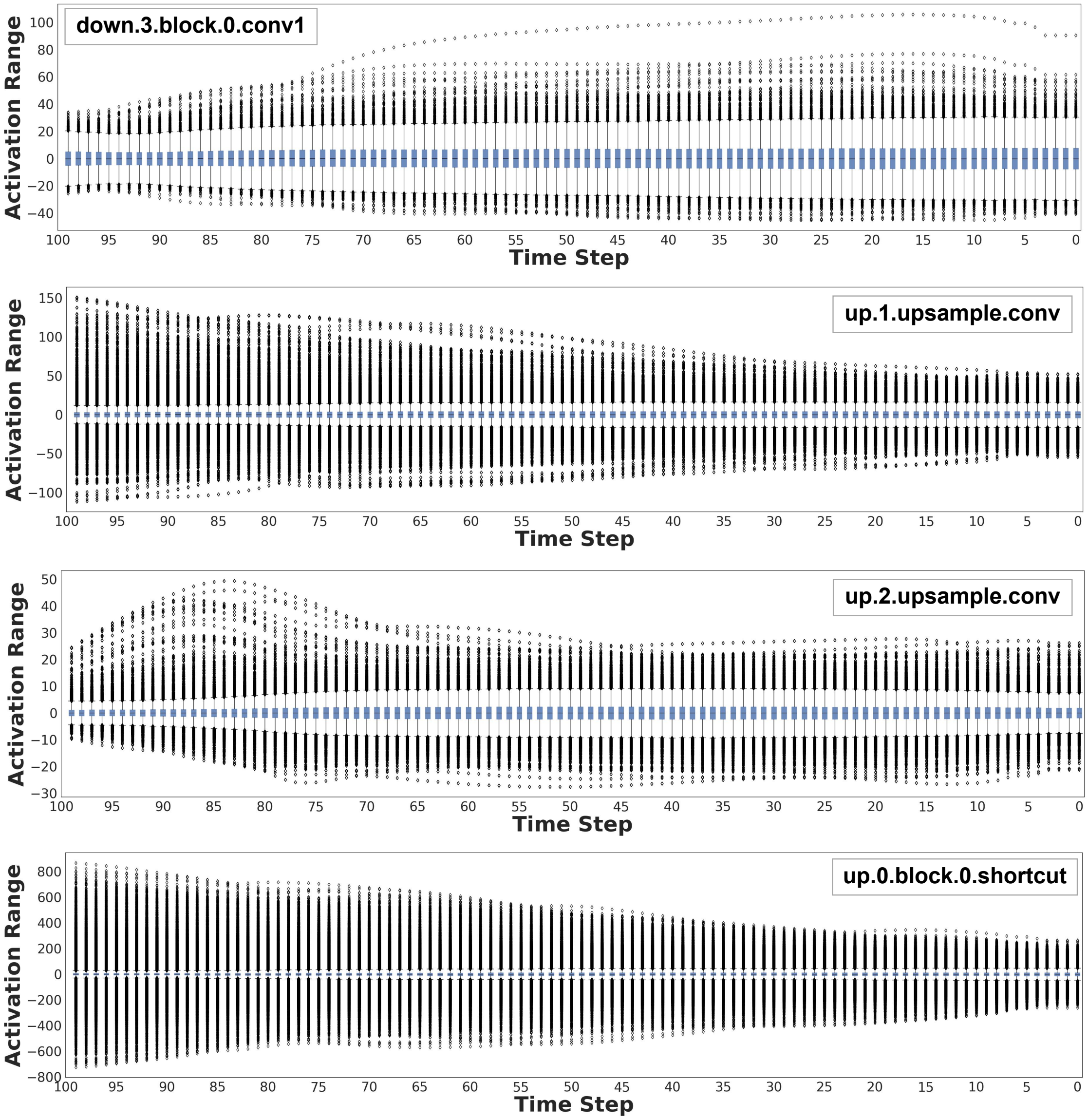


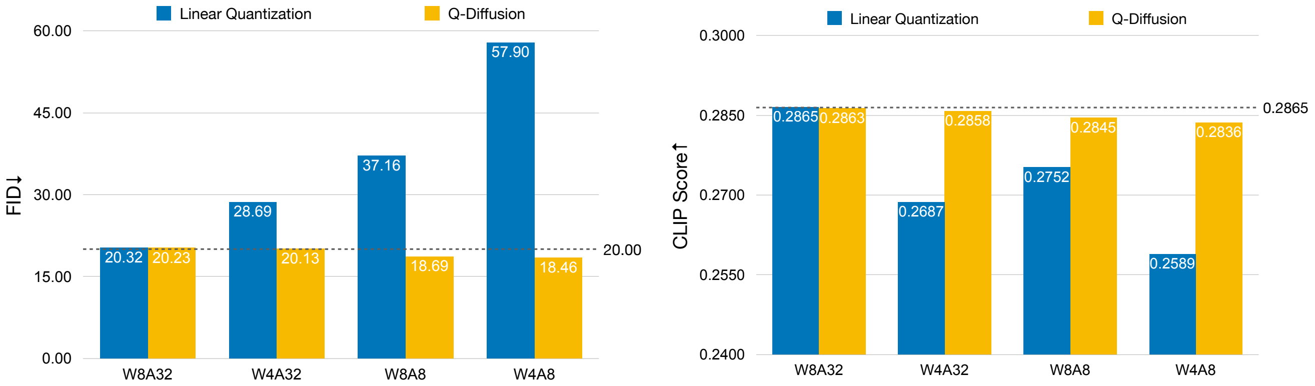
Figure 14: The distribution of activation values of multiple layers in DDIM on CIFAR-10 varies significantly across different time steps.

becoming inconsistent with how Q-Diffusion is calibrated when the sampling trajectories change. We leave the design for a systematic pipeline that can effectively combine these two directions in diffusion model acceleration as future work.

E. Comparing with PTQ4DM [10]

We evaluated Q-Diffusion on the settings employed in [10], which computed Inception Score (IS), Frechet Inception Distance (FID), and sFID [8] over only 10k generated samples. Although [10] did not specify details in the paper, their official implementation computed activation-to-activation matrix multiplications in the attention ($q * k$ and $attn * v$) in FP16/32[7], while we conducted them in full-integer. These matmuls account for a substantial portion of FLOPs (e.g. 9.8% of the model in SD) and can induce considerable memory overheads [1], which subsequently increase the inference costs. Contrarily, our work reduces the memory & compute in this part by 2x/4x theoretically.

[7] https://github.com/42Shawn/PTQ4DM (05/31/2023)



(a) Q-Diffusion has negligible increases in FID

(b) Q-Diffusion maintains similar CLIP score

Figure 15: Stable Diffusion (cfg scale = 7.5) 512 × 512 text-guided image synthesis FID and CLIP score results quantized using Q-Diffusion and Linear Quantization under different precisions. The dotted line values are obtained under full precision.

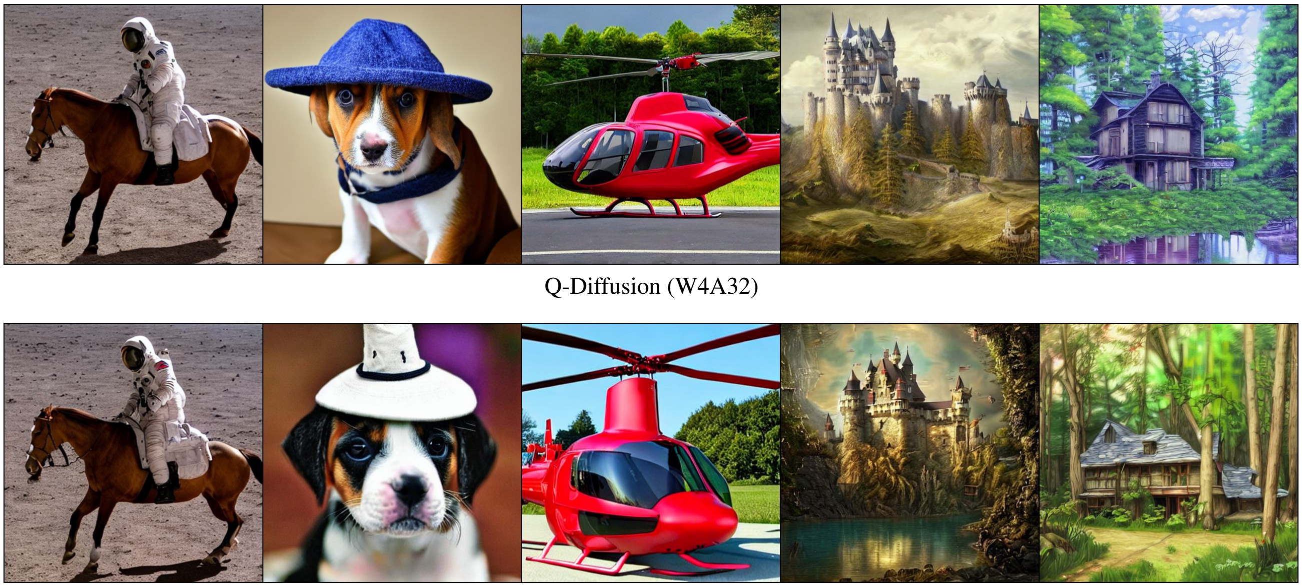
The evaluation results are demonstrated in Table 5, where numbers inside the parentheses of PTQ4DM are its results reproduced with integer attention matmuls. Q-Diffusion consistently outperforms PTQ4DM [10], which achieves better results with attention matmuls in INT8 than PTQ4DM with them computed in FP16/32. Note that directly applying [10] to quantize attention matmuls in 8-bit would even further degrade generation quality, as shown by the numbers inside parentheses.

Table 5: Q-Diffusion and PTQ4DM [10] results. The numbers inside the PTQ4DM parentheses refer to [10] results with INT8 attention act-to-act matmuls.

Task	Method	IS↑	FID↓	sFID↓
CIFAR-10 DDIM 100 steps	FP	9.18	10.05	19.71
	PTQ4DM (W8A8)	9.31 (9.02)	14.18 (19.59)	22.59 (20.89)
	Q-Diffusion (W8A8)	**9.47**	**7.82**	**17.96**
	Q-Diffusion (W4A8)	9.19	8.85	19.64
CIFAR-10 DDIM 250 steps	FP	9.19	8.83	18.31
	PTQ4DM (W8A8)	**9.70** (9.30)	11.66 (16.54)	19.71 (20.08)
	Q-Diffusion (W8A8)	9.60	**8.00**	**18.13**
	Q-Diffusion (W4A8)	9.18	8.54	18.58

Table 6: Q-Diffusion results when directly applying 3rd-order DPM-Solver++ with 50 denoising time steps.

Task	Bits (W/A)	FID↓
DDIM CIFAR-10	32/32	3.57
DDIM CIFAR-10	4/32	5.38
DDIM CIFAR-10	4/8	10.27
LDM-4 LSUN-Bedrooms	32/32	4.27
LDM-4 LSUN-Bedrooms	4/32	4.88
LDM-4 LSUN-Bedrooms	4/8	10.77
LDM-8 LSUN-Churches	32/32	5.40
LDM-8 LSUN-Churches	4/32	5.74
LDM-8 LSUN-Churches	4/8	8.19



Q-Diffusion (W4A32)

Q-Diffusion DPM-Solver++ (W4A32)

Figure 16: Text-guided image generation results on 512 × 512 resolution from our INT4 weights-quantized Stable Diffusion with default PNDM 50 time steps and DPM-Solver++ 20 time steps.

F. Limitations of This Work

This work focuses on providing a PTQ solution for the noise estimation network of the diffusion models on the unconditional image generation task. Meanwhile, we notice the recent advancement of text-guided image generation [9] and other multi-modality conditional generation tasks. As we have demonstrated the possibility of directly applying Q-Diffusion to the noise estimation network of Stable Diffusion, we believe it is important to provide a systematic analysis of the quantization's impact on the text encoder and the cross-attention mechanism for the classifier-free guidance conditioning, to enable a fully quantized conditional generation framework. For unconditional generation, this work discovers the need to sample calibration data across all time steps, and apply specialized split quantizers for the concatenation layers in the noise estimation model. The combination of these techniques demonstrates good performance for quantized diffusion models. Meanwhile, there exist other interesting design choices, like non-uniform sampling across different time steps, and additional quantizer design for attention softmax output, *etc*., that can be explored. We leave further investigation of these points as future work.

F.1. Non-uniform sampling methods that Did Not Work

As a preliminary exploration of non-uniform calibration data sampling across time steps, we explore the following 3 sampling methods. Yet none of those achieves better performance than Uniform sampling as proposed in this paper under the same amount of calibration data (5120), as shown in Table 7.

Std Since we observe the diverse activation range across time steps in Fig. 5, we would like to sample more data from the time step with a larger variance in its distribution, so as to better represent the overall output distribution across all time steps. To this end, we propose to sample calibration data from each time step in proportion to the pixel-wise standard deviation (Std) of each time step. Specifically, we randomly sample 256 x_t among all time steps and compute the Std of all pixel values in x_t at each time step, which we denote as s_t. Then for calibration data, we sample $\frac{s_t}{\sum_t s_t}N$ examples out of the total N calibration data from time step t.

Norm Std Similar to Std, we also consider modeling the variance of each time step's distribution with the standard deviation of $||x_t||_2$, instead of the Std of all pixel values. We expect the Norm Std can better capture the diversity across different samples instead of capturing the pixel-wise diversity within each sample compared to pixel-wise Std.

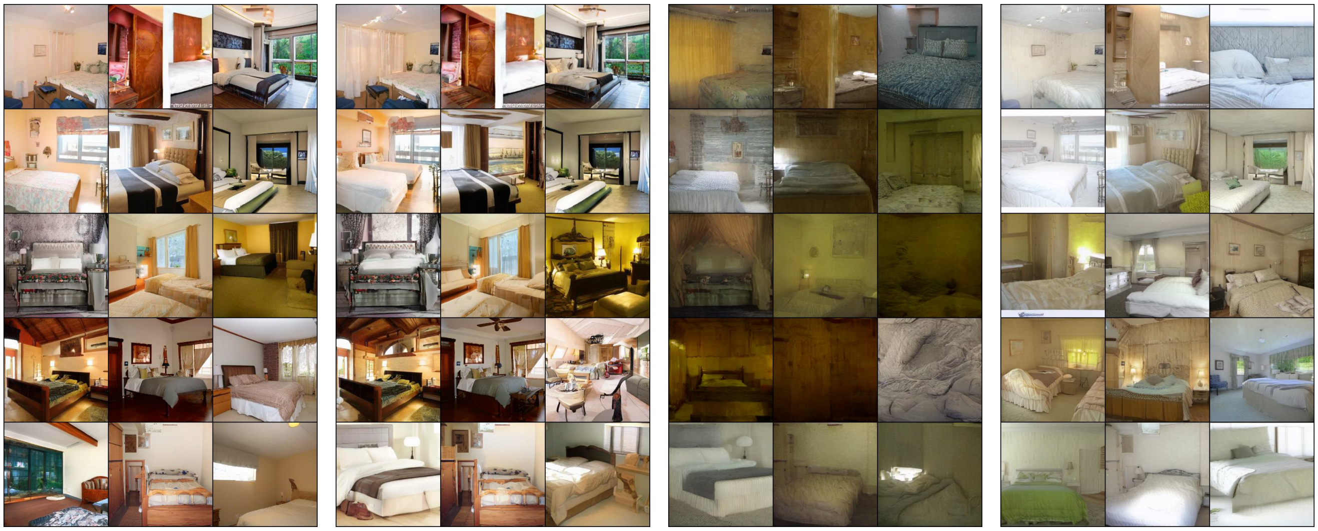
Unsupervised Selective Labeling (USL) We also try to use Unsupervised Selecting Labeling [12] to select both representative and diverse samples as the calibration samples. The intuition is that samples that are both representative and diverse could provide a wide range of activations that we will encounter at inference time, focusing on which could bring us good performance on generation most of the time. We select 5120 samples in total for CIFAR-10 by combining the samples for all time steps. We adopt the training-free version of Unsupervised Selective Labeling for sample selection, with the pooled latent space feature from the noise estimation UNet as the selection feature.

Table 7: Quantization results for unconditional image generation with DDIM on CIFAR-10 (32 × 32). We compare different calibration data sampling schemes under W4A32 quantization.

Method	Std	Norm Std	USL	Uniform (ours)
FID↓	5.66	5.58	5.54	**5.09**

G. Additional Random Samples

In this section, we provide more random samples from our weight-only quantized and fully quantized diffusion models obtained using Q-Diffusion and Linear Quantization under 4-bit quantization. Results are shown in the figures below.



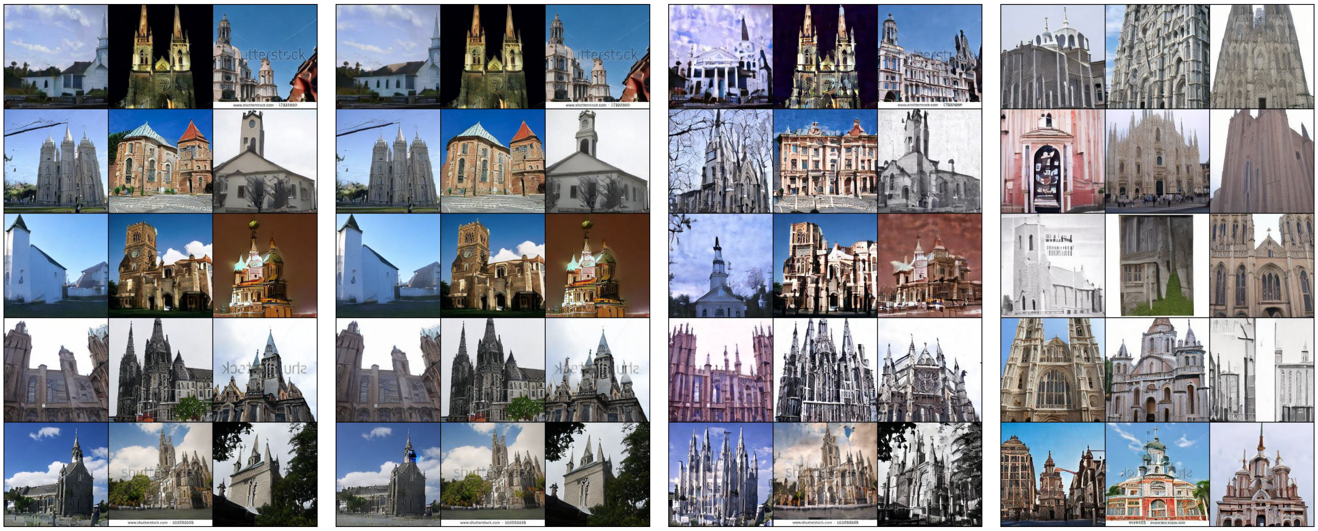
Q-Diffusion (W4A32)

Q-Diffusion (W4A8)

Linear Quant (W4A32)

Linear Quant (W4A8)

Figure 17: Random samples from our INT4 quantized 256 ×256 LSUN-Bedroom models with a fixed random seed.



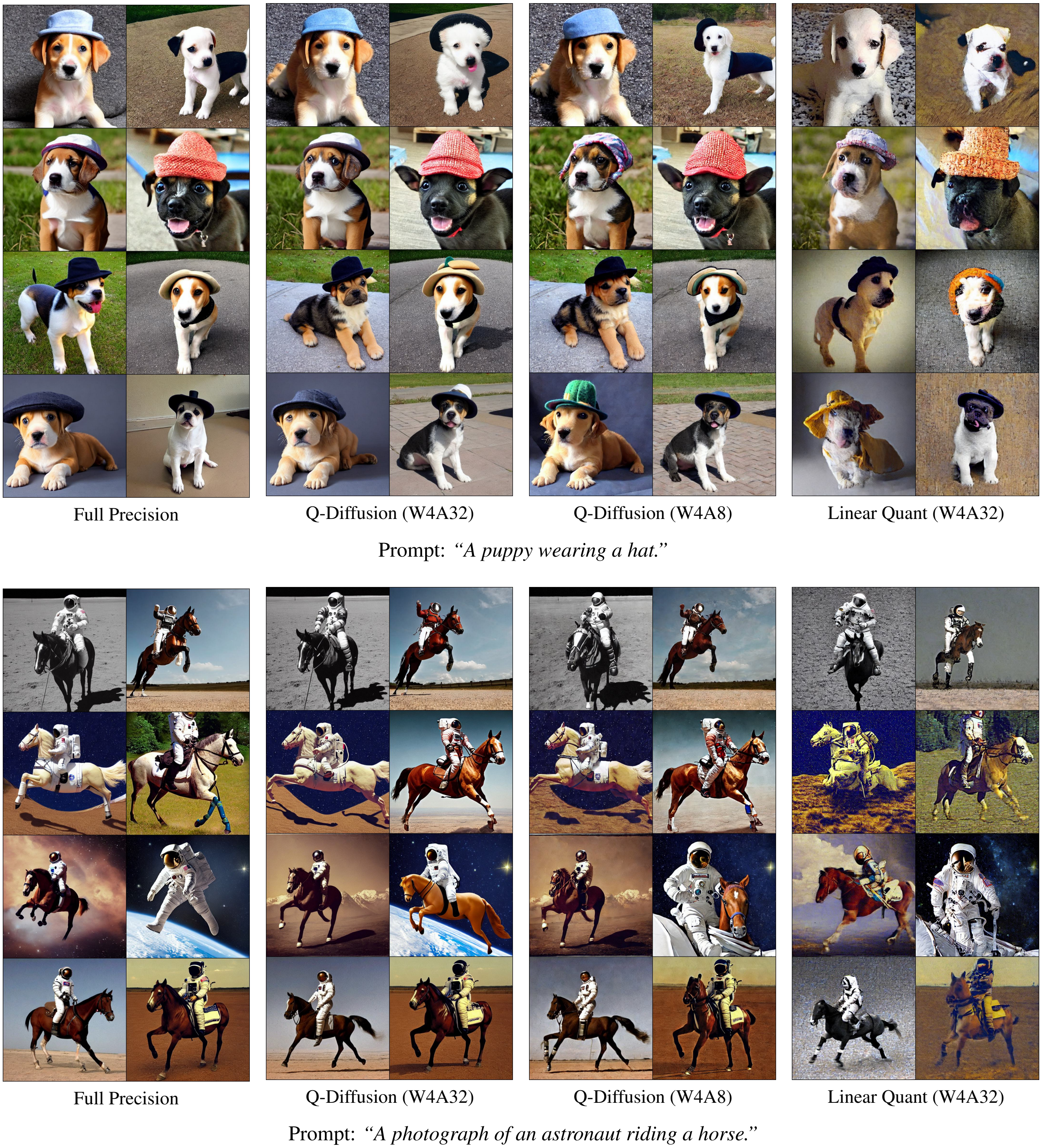
Q-Diffusion (W4A32)

Q-Diffusion (W4A8)

Linear Quant (W4A32)

Linear Quant (W4A8)

Figure 18: Random samples from our INT4 quantized 256 ×256 LSUN-Church models with a fixed random seed.



Full Precision Q-Diffusion (W4A32) Q-Diffusion (W4A8) Linear Quant (W4A32)

Prompt: *"A puppy wearing a hat."*

Full Precision Q-Diffusion (W4A32) Q-Diffusion (W4A8) Linear Quant (W4A32)

Prompt: *"A photograph of an astronaut riding a horse."*

Figure 19: Text-guided image generation on 512 × 512 LAION-5B from our INT4 quantized Stable Diffusion model with a fixed random seed.

References

[1] Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022. 4

[2] Cong Guo, Yuxian Qiu, Jingwen Leng, Xiaotian Gao, Chen Zhang, Yunxin Liu, Fan Yang, Yuhao Zhu, and Minyi Guo. Squant: On-the-fly data-free quantization via diagonal hessian approximation. *ArXiv*, abs/2202.07471, 2022. 1

[3] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *EMNLP*, 2021. 2

[4] 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. 2

[5] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014. 2

[6] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *ArXiv*, abs/2206.00927, 2022. 3

[7] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models. *ArXiv*, abs/2211.01095, 2022. 3

[8] Charlie Nash, Jacob Menick, Sander Dieleman, and Peter W. Battaglia. Generating images with sparse representations. *International Conference On Machine Learning*, 2021. 4

[9] Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10674–10685, 2021. 1, 2, 6

[10] Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. *CVPR*, 2023. 4, 5

[11] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1

[12] Xudong Wang, Long Lian, and Stella X Yu. Unsupervised selective labeling for more effective semi-supervised learning. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXX*, pages 427–445. Springer, 2022. 7

[13] Zhewei Yao, Zhen Dong, Zhangcheng Zheng, Amir Gholami, Jiali Yu, Eric Tan, Leyuan Wang, Qijing Huang, Yida Wang, Michael Mahoney, et al. Hawq-v3: Dyadic neural network quantization. In *International Conference on Machine Learning*, pages 11875–11886. PMLR, 2021. 1