Pioneering 4-Bit FP Quantization for Diffusion Models: Mixup-Sign Quantization and Timestep-Aware Fine-Tuning

Supplementary Material

A. Supplementary Material Overview

In this supplementary material, we provide additional explanations and experimental results referenced in the main paper. The content is organized as follows:

- Methodology of Mixup-Sign Quantization in Appendix B.
- More Implementation Details in Appendix C.
- FP vs. INT in Post-Training Quantization in Appendix D.
- Comprehensive Analysis of TALoRA Performance in Appendix E.
- Supplementary Performance Evaluation in Appendix F.
- Extensive Comparison with EfficientDM and QUEST in Appendix G.
- Additional Visualization Results in Appendix H.

B. Methodology of Mixup-Sign Quantization

We implement the proposed MSFP strategy using a search-based method [1, 10], wherein the quantization parameters are determined by minimizing the MSE between the distributions before and after quantization.

To clarify, the quantization parameters for the signed FP quantization include the $format$, bias b, and sign bit s set to 1, whereas the quantization parameters for the unsigned FP quantization include the $format$, bias b, sign bit s set to 0, and zero point zp. All quantization parameters are assigned a search space during initialization.

As mentioned in the main text, the bias b serves as a threshold in FP quantization:

$$maxval = 2^{2^x - 1 - b} \cdot \left(1 - \frac{1}{2^y}\right) \tag{1}$$

The maximum value, denoted as $maxval$, is determined by the $format$ (e.g.,ExMy) abd the bias b, and represents the maximum discrete value achievable in FP quantization. Notably, $maxval$ and b are directly correlated, and for convenience, we will refer to $maxval$ in subsequent discussions as the equivalent to the bias b.

In the MSFP strategy, initialization is divided into two parts: weight initialization and activation initialization. During initialization, we determine the optimal quantization parameter settings, and the process is outlined in Algorithm 1: In the first stage, the search for signed FP quantization parameters is applicable to all cases. In the second stage, the search for unsigned FP quantization parameters is specifically applied to the activation initialization of the Anomalous-Activation-Distribution Layers (AALs) mentioned in the main text.

Algorithm 1 Initialization of Quantization Parameters

Input: $format\_options$, $maxval\_options$, $(zp\_options)$, $(unsigned\_format\_options)$
Output: $format$, $maxval$, (zp)

#10000 is huge enough
$min\_mse = 10000$
$s = 1$
for f in $format\_options$ **do**
  for $prev\_m$ in $maxval\_options$ **do**
    $prev\_mse = calculate\_mse(f, prev\_m, s)$
    if $prev\_mse < min\_mse$ **then**
      $min\_mse = prev\_mse$
      $format = f$
      $maxval = prev\_m$
    end if
  end for
end for

#only for unsigned FP quantization
$s = 0$
for f in $unsigned\_format\_options$ **do**
  for $prev\_m$ in $maxval\_options$ **do**
    for $prev\_zp$ in $zp\_options$ **do**
      $prev\_mse = calculate\_mse(f, prev\_m, prev\_zp, s)$
      if $prev\_mse < min\_mse$ **then**
        $min\_mse = prev\_mse$
        $format = f$
        $maxval = prev\_m$
        $zp = prev\_zp$
      end if
    end for
  end for
end for

Additionally, due to the significant variability in the search space for $maxval$, which depends on the differing distributions of the data, therefore, prior to initiating the search for quantizer parameters, the first step involves performing several random forward passes to capture the maximum value observed for each quantizer. This value is then used as the initial $maxval\_0$.

Weight Initialization. For weight initialization, since the distribution of weights typically approximates a normal distribution (as shown in Figure 1), we deploy signed FP quantization. In the search for the $format$ of signed

Search Space	Bits (W/A)	FID ↓
[0,maxval_0]	6/32	10.14
[0,2maxval_0]	6/32	10.26
[0.6maxval_0,2maxval_0]	6/32	9.36
[0.7maxval_0,2maxval_0]	6/32	6.46
[0.8maxval_0,2maxval_0]	6/32	5.58
[0.9maxval_0,2maxval_0]	6/32	**5.13**
[maxval_0,2maxval_0]	6/32	5.83

Table 1. The impact of different $maxval$ search spaces in weight initialization on the DDIM model performance on CelebA dataset.

FP quantization, we define a search space of size 4 for 4-bit, 6-bit, and 8-bit representations, encompassing the most expressive data formats for each bit-width while striking a balance between computational overhead and performance [6, 13, 15].

For the search of $maxval$ in weights, we extend the previous search range of $range(0, maxval\_0, 0.001)$ to explore a more refined and reasonable search space. On the one hand, considering that large-value weights are relatively few but have a significant impact, we set the lower bound of the search to a value slightly smaller than $maxval\_0$ to avoid excessive loss of essential large-value weights. On the other hand, setting the upper bound to $maxval\_0$ may not guarantee the minimization of MSE. As inferred from the representation of FP quantization, any quantizer with its $maxval$ larger than $2 \times maxval\_0$ cannot result in a smaller MSE, so we set the upper bound of the search to $2 \times maxval\_0$. As shown in Table 1, our exploration across different search spaces demonstrates the effectiveness of the redefined search space of $maxval$.

Activation Initialization. For activation initialization, based on the analysis in the main text, we employ signed FP quantization for NALs with distribution approximately following a normal distribution, and adopt a mixup-sign FP quantization strategy for AALs with asymmetric distributions. Unlike weight initialization, where weights remain static, activation initialization needs to account for potential activation distributions. To ensure that the activations used for initialization are representative, we introduce a calibration dataset [7, 14], as is common in INT quantization.

Given the increased complexity and randomness of activation distributions, we include all possible formats for different bit-widths within the search space for $format$. Notably, for n-bit unsigned FP quantization with the ExMy $format$, the condition $x + y + s = n$ applies, where $s = 0$, distinguishing its format from that of signed FP quantization, which includes s set to 1 under the same bit-width. Accordingly, the search range for $maxval$ is adjusted to $linspace(0, maxval\_0, 100)$, preventing excessive computational overhead. Lastly, for the zero point zp introduced in unsigned FP quantization, since the minimum value of the distribution is constrained by $SiLU$ to approximately -0.278, assigning zp a search space of $linsapce(-0.3, 0, 6)$ is sufficient.

C. More Implementation Details

FP PTQ Configuration. Following the procedure outlined in Appendix B, we deploy our MSFP strategy for both weights and activations. The initialization of $maxval\_0$ is achieved by generating 2000 images through random forward passes. Subsequently, a calibration dataset is constructed based on the output of the full-precision model, following the approach of Q-Diffusion [7]. Specifically, 256 samples are used for the DDIM model, while 128 samples are used for the LDM model.

For weight initialization, the search spaces for $maxval$ and $format$ are presented in Table 2. For activation initialization, the search spaces for $maxval$, $fomrat$ and zp are thoroughly discussed and provided in Appendix B.

Bit	Search Space ($maxval$)	Search Space ($format$)
4	[0.8maxval_0,2maxval_0]	[E3M0,E2M1,E1M2,E0M3]
6	[0.9maxval_0,2maxval_0]	[E4M1,E3M2,E2M3,E1M4]
8	[0.9maxval_0,2maxval_0]	[E5M2,E4M3,E3M4,E2M5]

Table 2. Search spaces for different quantization parameters under different bit-widths in weight initialization.

Fine-tuning Configuration. For the noise estimation U-Net, all quantized layers, except for the input and output layers, are quantized and equipped with QLoRA-based TALoRAs [3]. Each TALoRA is initialized with a rank of 32. The selection of different TALoRAs at each timestep is managed by a router, which is implemented as a linear layer. The input channels of the router match the channel count of the timestep embedding in the diffusion model.

Adam optimizers are assigned to both the TALoRAs and the router, with a learning rate of 1e-4 for both components. Fine-tuning is performed for 160 epochs with a batch size of 16 on DDIM models and 320 epochs with a batch size of 8 on LDM models. Notably, the batch size for the ImageNet dataset is reduced to 4.

D. FP vs. INT in Post-Training Quantization

Table 3 presents a performance comparison between the 6-bit model initialized with MSFP and several 6-bit models based on traditional INT quantization [4, 7, 14, 17]. As shown, even without fine-tuning, our approach significantly outperforms existing SOTA methods in handling 6-bit quantization for diffusion models. This highlights that FP quantization is a more effective choice for handling low-bit activation quantization in diffusion models, a task that is both

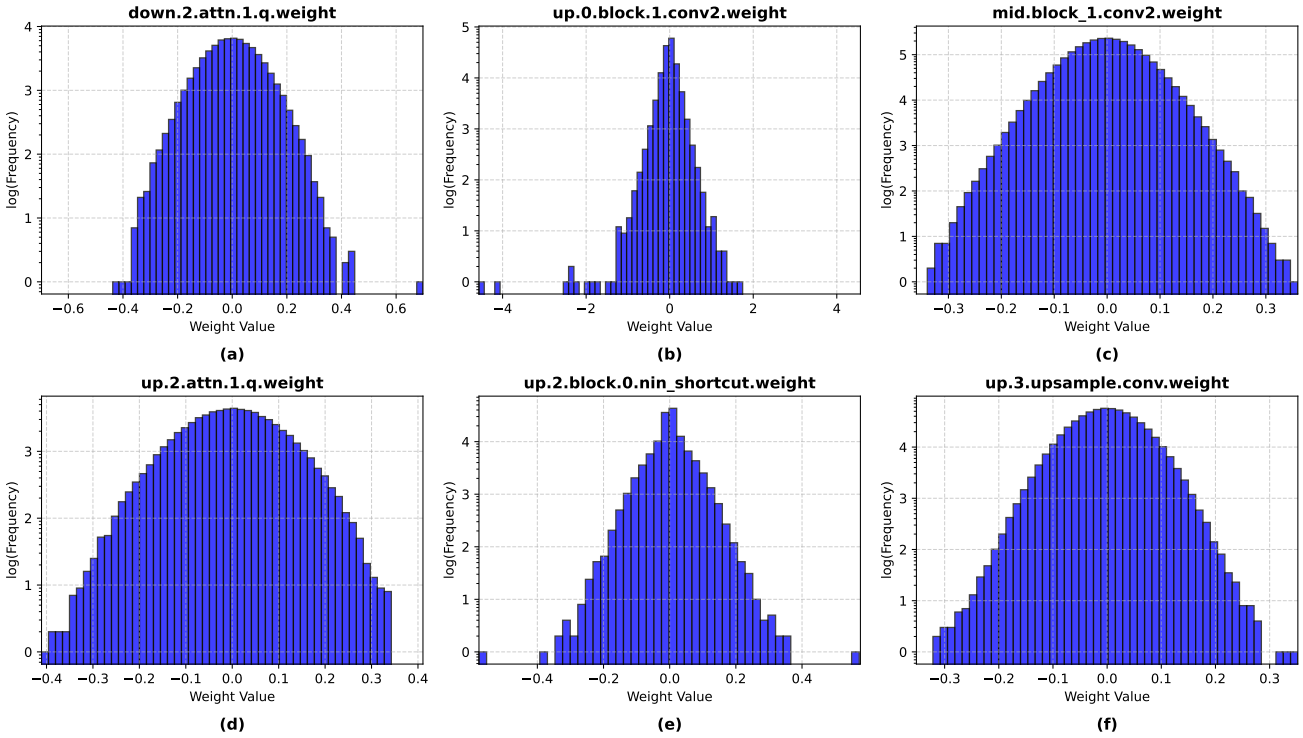


Figure 1. The weight distribution of certain layers in the DDIM model on CelebA dataset.

Task	Method	Prec. (W/A)	FID ↓	IS ↑
CelebA 64x64 DDIM steps = 100	FP	32/32	6.49	2.61
	LSQ	6/6	78.37	1.94
	PTQ4DM	6/6	24.96	2.13
	Q-Diffusion	6/6	23.37	2.16
	ADP-DM	6/6	16.86	2.30
	Ours(MSFP)	6/6	**9.51**	**2.78**

Table 3. Quantization performance of unconditional generation. In this case, 'Ours' refers to the method that deploys only the MSFP strategy without any fine-tuning. 'Prec. (W/A)' denotes the quantization bit-width.

challenging and crucial, compared to INT-based methods.

E. Comprehensive Analysis of TALoRA Performance

E.1. TALoRA Outperforms Rank-Scaled LoRA

In our approach, we introduce multiple TALoRAs for the majority of quantized layers, which leads to an increase in the model size. Some may question whether the observed performance improvement is simply due to the larger memory footprint of the LoRAs. However, in practice, only one TALoRA is active at each timestep, which differs fundamentally from using a larger-rank LoRA, as the latter would result in higher training and inference costs. Furthermore, Table 4 presents the results of fine-tuning with two TALoRAs (rank=32) and a single QLoRA (rank=64). Our method achieves even better performance, demonstrating that our timestep-aware fine-tuning strategy effectively recovers the performance lost during quantization in diffusion models, with lower overhead and enhanced performance.

Method	Rank	Bits(W/A)	FID↓
FP	/	32/32	6.49
single-LoRA	64	4/4	7.75
TALoRA(h=2)	32	4/4	**7.69**

Table 4. Comparison between TALoRA and rank-scaled LoRA in fine-tuning 4-bit DDIM models on CelebA dataset. 'Rank' refers to the LoRA rank.

E.2. Impact of TALoRA Quantity

As illustrated in Figure 2, when deploying four TALoRAs, the distributions of LoRA allocation across different timesteps exhibits a strong regularity: in most cases, regardless of the dataset, the majority of timesteps utilize only two LoRAs. This suggests that fine-tuning low-bit diffu-

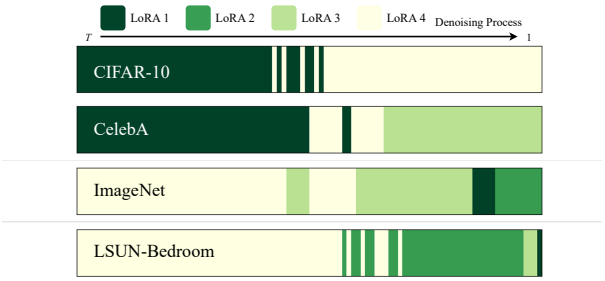


Figure 2. Distribution of LoRA allocations over timesteps obtained after router training on different datasets, when $h = 4$.

sion models predominantly follows a two-stage task pattern, which aligns with the motivation behind introducing TALoRAs—viewing the denoising process as a progression from restoring coarse structures to refining intricate details [16].

Experimental results in the main text further demonstrates that deploying four TALoRAs does not yield better results compared to deploying two TALoRAs. In fact, in most cases, the latter achieves superior results on 4-bit diffusion models. This aligns with our earlier analysis: two TALoRAs are sufficient to handle the fine-tuning task effectively, while the introduction of additional TALoRAs could reduce the training opportunities for the most impactful LoRAs, ultimately compromising fine-tuning performance.

F. Supplementary Performance Evaluation

To further validate the effectiveness of our approach, we conduct supplementary experiments. For the DDIM model, where prior methods have struggled, our approach is evaluated on the CelebA dataset [11]—a more complex dataset with higher image resolutions corresponding to a more intricate DDIM model. As shown in Table 5, our method achieves cutting-edge performance under both 4-bit and 6-bit settings. Notably, our 4-bit diffusion model exhibits performance on FID and IS metrics comparable to full precision, and our method even outperforms the full-precision model under the 6-bit setting.

For the LDM model, we further evaluate it on the ImageNet dataset [2] using two advanced sampling methods, PLMS [9] and DPM-Solver [12], which are more sophisticated and computationally demanding during fine-tuning. Table 6 demonstrates that our method maintains robust performance under both 4-bit and 6-bit quantization settings, achieving SOTA results on the more reliable sFID and IS metrics in ImageNet.

Furthermore, we apply our method to the task of quantizing text-to-image diffusion models, specifically deploying it on Stable Diffusion with the MS-COCO dataset [8]. Our approach also delivers highly satisfactory results, with detailed visualizations provided in Appendix H.

Task	Method	Prec. (W/A)	FID ↓	IS ↑
CelebA 64x64 DDIM steps = 100	FP	32/32	6.49	2.61
	Q-Diffusion	6/6	23.37	2.16
	ADP-DM	6/6	16.86	2.30
	Ours(h=2)	6/6	5.38	**2.67**
	Ours(h=4)	6/6	**5.36**	2.66
	Q-Diffusion	4/4	N/A	N/A
	ADP-DM	4/4	N/A	N/A
	Ours(h=2)	4/4	**7.69**	2.59
	Ours(h=4)	4/4	7.84	**2.60**

Table 5. Quantization performance of unconditional generation. 'Prec. (W/A)' denotes the quantization bit-width. 'N/A' denotes failed image generation. h denotes the size of LoRA Hub.

Task	Method	Prec. (W/A)	sFID ↓	FID ↓	IS ↑
LDM-4 PLMS steps = 20	FP	32/32	7.08	11.71	379.19
	EDA-DM	6/6	6.59	11.27	363.00
	EfficientDM	6/6	9.36	**9.85**	325.13
	Ours(h=2)	6/6	5.63	10.35	363.79
	Ours(h=4)	6/6	**5.33**	10.25	**364.27**
	EDA-DM	4/4	32.63	17.56	**203.15**
	EfficientDM	4/4	9.89	14.78	103.34
	Ours(h=2)	4/4	**7.39**	**7.27**	196.32
	Ours(h=4)	4/4	7.83	7.83	193.11
LDM-4 DPM-Solver steps = 20	FP	32/32	6.85	11.44	373.12
	EDA-DM	6/6	7.95	11.14	357.16
	EfficientDM	6/6	9.30	**8.54**	336.11
	Ours(h=2)	6/6	**6.86**	9.61	363.71
	Ours(h=4)	6/6	6.88	9.59	**364.30**
	EDA-DM	4/4	39.40	30.86	138.01
	EfficientDM	4/4	13.82	14.36	109.52
	Ours(h=2)	4/4	**12.61**	**8.46**	**257.33**
	Ours(h=4)	4/4	14.56	9.64	238.07

Table 6. Quantization performance of conditional generation for fully-quantized LDM-4 models on ImageNet 256×256 with 20 steps, using PLMS and DPM-Solver as sampling methods. 'Prec. (W/A)' denotes the quantization bit-width. h denotes the size of LoRA Hub.

G. Extensive Comparison with EfficientDM and QUEST

As mentioned in the main text, prior fine-tuning-based methods, such as EfficientDM [5] and Quest [18], adopt specialized experimental setups. EfficientDM retains all $skip\_connection$ layers and the op layers within $Upsample$ blocks in full precision. These layers consti-

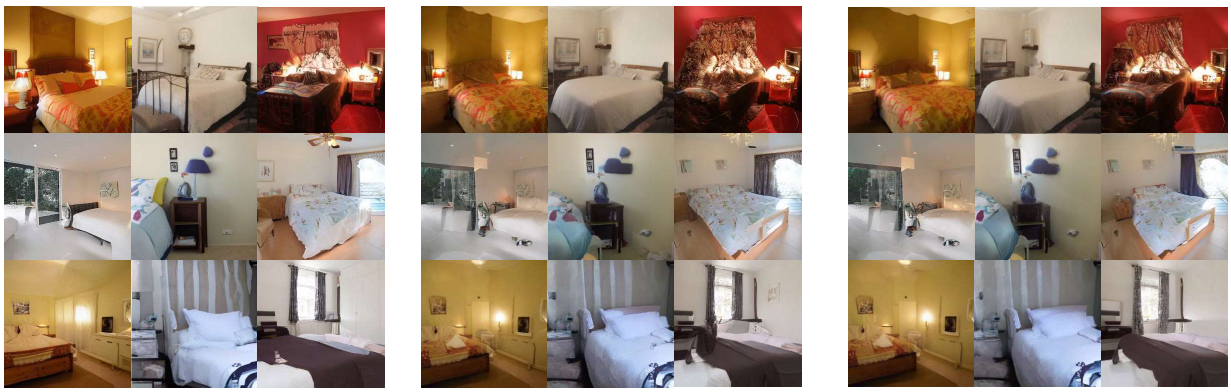
Task	Settings	Method	Prec. (W/A)	FID ↓
LSUN-Church 256 × 256 LDM-8 steps = 100 eta = 0.0	-	FP	32/32	4.06
	Partial Quantization	EfficientDM	4/4	13.68
		Ours(h=2)	4/4	**7.95**
	Full Quantization	EfficientDM	4/4	18.40
		Ours(h=2)	4/4	**8.81**
	Channel-wise for Activation	QuEST	4/4	11.76
		Ours(h=2)	4/4	-
	Layer-wise for Activation	QuEST	4/4	13.03
		Ours(h=2)	4/4	**8.81**

Table 7. Comparison with EfficientDM and QuEST under specific settings. 'Prec. (W/A)' denotes the quantization bit-width. h denotes the size of LoRA Hub.

tute a significant portion of the model, and their quantization significantly affects performance. Therefore, in our comparative experiments, we apply standard quantization to these layers. In contrast, Quest adopts a different strategy by modifying the quantization granularity for activations. Specifically, in low-bit quantization, channel-wise quantization of weights is a common approach. However, Quest extends this to activations, introducing substantial computational overhead compared to the mainstream layer-wise quantization. To ensure a fair comparison, we employ conventional layer-wise quantization for activations.

For a comprehensive evaluation, we align our method with the specific settings of EfficientDM and consider the implications of Quest's setup. As shown in Table 7, under EfficientDM's configuration, our 4-bit LDM model achieves significantly better results on the Church dataset, with an FID score that is 6.39 lower than EfficientDM's. However, we choose not to replicate Quest's specific settings for two key reasons. First, despite using the more efficient layer-wise quantization for both weights and activations, our method already surpasses Quest's performance. Specifically, under the 4-bit setting, our method achieves an FID of 8.81, compared to Quest's 11.76, which relies on computationally expensive channel-wise quantization for both. Second, our approach relies on FP quantization, and incorporating channel-wise quantization necessitates search-based initialization for every channel, which is computationally infeasible.

H. Additional Visualization Results



Full-precision(W32A32) Ours(h=2) Ours(h=4)

Figure 3. Visualization of random samples from 4-bit LDM-4 on LSUN-Bedroom across different LoRA Hub sizes h.

Full-precision(W32A32) Ours(W6A6) Ours(W4A4)

Figure 4. Visualization of random samples from quantized LDM-4 on ImageNet. The size of LoRA Hub is 2.

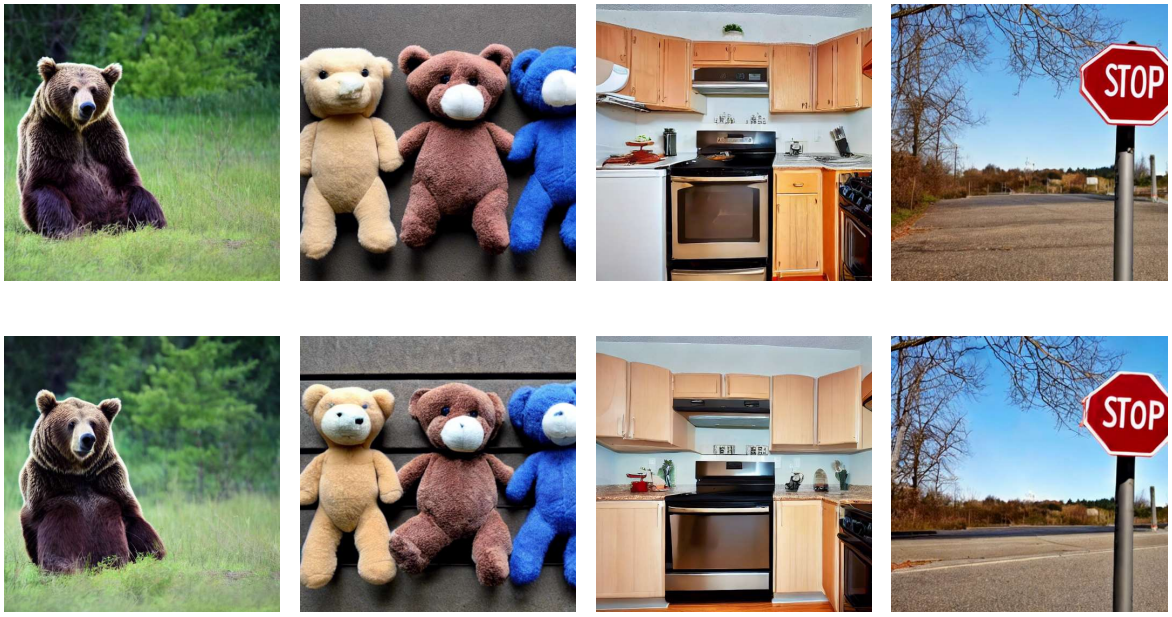


Figure 5. Comparison of text-to-image outputs from 6-bit quantized and full-precision Stable Diffusion models. h denotes the size of LoRA Hub.

References

[1] Cheng Chen, Christina Giannoula, and Andreas Moshovos. Low-bitwidth floating point quantization for efficient high-quality diffusion models. *arXiv preprint arXiv:2408.06995*, 2024. 1

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 4

[3] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: efficient finetuning of quantized llms (2023). *arXiv preprint arXiv:2305.14314*, 52:3982–3992, 2023. 2

[4] Steven K Esser, Jeffrey L McKinstry, Deepika Bablani, Rathinakumar Appuswamy, and Dharmendra S Modha. Learned step size quantization. In *International Conference on Learning Representations*, 2019. 2

[5] 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. 4

[6] Andrey Kuzmin, Mart Van Baalen, Yuwei Ren, Markus Nagel, Jorn Peters, and Tijmen Blankevoort. Fp8 quantization: The power of the exponent. *Advances in Neural Information Processing Systems*, 35:14651–14662, 2022. 2

[7] 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. 2

[8] 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. Springer, 2014. 4

[9] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. *arXiv preprint arXiv:2202.09778*, 2022. 4

[10] Shih-yang Liu, Zechun Liu, Xijie Huang, Pingcheng Dong, and Kwang-Ting Cheng. Llm-fp4: 4-bit floating-point quantized transformers. *arXiv preprint arXiv:2310.16836*, 2023. 1

[11] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 4

[12] 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. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 4

[13] Paulius Micikevicius, Dusan Stosic, Neil Burgess, Marius Cornea, Pradeep Dubey, Richard Grisenthwaite, Sangwon Ha, Alexander Heinecke, Patrick Judd, John Kamalu, et al. Fp8 formats for deep learning. *arXiv preprint arXiv:2209.05433*, 2022. 2

[14] 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. 2

[15] Mart van Baalen, Andrey Kuzmin, Suparna S Nair, Yuwei Ren, Eric Mahurin, Chirag Patel, Sundar Subramanian, Sanghyuk Lee, Markus Nagel, Joseph Soriaga, et al. Fp8 versus int8 for efficient deep learning inference. *arXiv preprint arXiv:2303.17951*, 2023. 2

[16] Binxu Wang and John J Vastola. Diffusion models generate images like painters: an analytical theory of outline first, details later. *arXiv preprint arXiv:2303.02490*, 2023. 4

[17] Changyuan Wang, Ziwei Wang, Xiuwei Xu, Yansong Tang, Jie Zhou, and Jiwen Lu. Towards accurate data-free quantization for diffusion models. *arXiv preprint arXiv:2305.18723*, 2(5), 2023. 2

[18] 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. 4