Appendix

A Implementation Details

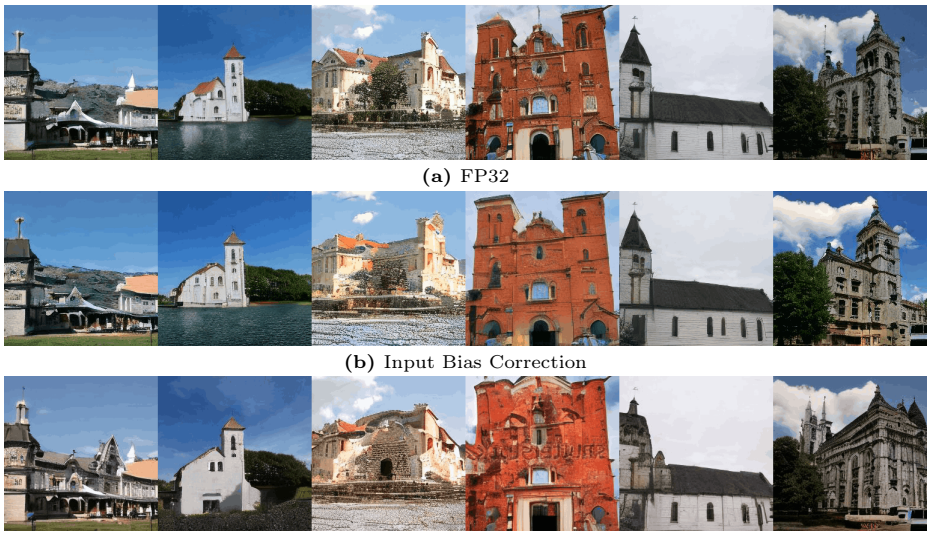
For CIFAR-10 image generation [22], we employ the 100-step DDIM approach [50]. For the LSUN-Bedroom and LSUN-Church datasets [54], we implement 200 steps with LDM-4 and 500 steps with LDM-8 [39], respectively. In conditional image generation, we use the official pre-trained Stable Diffusion version 1.4 [39], generating images with both 50-step PLMS and DDIM samplers. We adopt methods from [7,24,27,32] for model quantization and calibration, and use code from [24] to quantize model.

To calculate the reconstruction coefficients and input bias, we first run the full-precision model to generate a batch of samples, capturing the input and noise estimations at each timestep. This is followed by running the quantized model to determine these coefficients. Batch sizes are tailored to each task: 64 for CIFAR-10, 128 for LSUN experiments, and 256 for text-guided image generation with Stable Diffusion v1.4. In general, larger sample size may lead to better results. we leave this for future investigation.

We evaluate the FID score [16] using the official PyTorch implementation. For the IS score [45] evaluation on CIFAR-10, we utilize code from [6]. For high-resolution datasets like LSUN-Bedroom and LSUN-Church, we efficiently assess the results using pre-computed statistics over the entire dataset, as provided by [6]. For comparative experiments, we rerun the official scripts from [15,24,47].

B Comparison of Input Bias Correction and Noise Estimation Correction

In this section, we perform a comparative analysis between Input Bias Correction (IBC), as introduced in Section 4.2, and the noise estimation bias correction approach inspired by [33]. While the former method simultaneously corrects both the estimated noise, $\hat{\boldsymbol{\epsilon}}_t$, and the corrupted input, $\hat{\mathbf{x}}_t$, the latter focuses exclusively on correcting the corrupted noise estimation, $\hat{\boldsymbol{\epsilon}}_t$. The visualization results, presented in Figure 4, clearly demonstrate that the noise estimation correction strategy is less effective at preserving original content, often resulting in the loss of important objects and causing structural distortions in the generated images. Conversely, the IBC strategy, as implemented in TAC-Diffusion, produces images that are more closely aligned with those generated by the full-precision model. This efficacy can be attributed to IBC's ability to adjust the deviated model input back onto the correct path, consistent with the analysis of exposure bias discussed in Section 4.2.



(a) FP32

(b) Input Bias Correction

(c) Noise Estimation Bias Correction

Fig. 4: Comparison between different correction strategies in 256 × 256 unconditional generation on LSUN-Church with W3A8 500 steps LDM-8

C Overall Algorithm

The overall algorithm of TAC-Diffusion is described in Algorithm 1.

Algorithm 1 Timestep-Aware Correction

Pre-calculation:
Input: Full-precision diffusion model $\boldsymbol{\epsilon}_\theta$ and its quantized version $\hat{\boldsymbol{\epsilon}}_\theta$
Output: Reconstruction Coefficient $\mathbf{K}$ and Bias Corrector $\mathbf{B}$
for $t = T$ to 0 **do**
  Collect model output $\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)$ and $\hat{\boldsymbol{\epsilon}}_\theta(\hat{\mathbf{x}}_t, t)$
  Calculate the reconstruction coefficient $\mathbf{K}_t$
  Calculate the element-level bias $\mathbf{B}_t$
  Correct $\hat{\boldsymbol{\epsilon}}_\theta(\hat{\mathbf{x}}_t, t)$ and $\hat{\mathbf{x}}_t$ with $\mathbf{K}_t$ and $\mathbf{B}_t$
  Save $\mathbf{K}_t$ and $\mathbf{B}_t$ for inference
end for

Inference:
Input: Quantized noise estimator $\hat{\boldsymbol{\epsilon}}_\theta$, model input $\hat{\mathbf{x}}_t$, coefficient $\mathbf{K}_t$ and corrector $\mathbf{B}_t$
Output: Corrected Output $\tilde{\mathbf{x}}_0$
for $t = T$ to 0 **do**
  Correct input $\hat{\mathbf{x}}_t$ with $\mathbf{B}_t$
  Estimate noise with corrected input $\tilde{\mathbf{x}}_t$
  Reconstruct noise estimation with $\mathbf{K}_t$ and $\hat{\boldsymbol{\epsilon}}_\theta(\tilde{\mathbf{x}}_t, t)$
end for

D Extending TAC-Diffusion to DPM-Solver++

In this section, we extend TAC-Diffusion to an advanced high-order solver, *e.g.* DPM-Solver++ [31]. The procedure for this integration is summarized in Algorithm 2, where we exclude the pre-calculation process for simplicity.

To align with the notation used in [31], we define $\lambda_t = \log\left(\frac{\alpha_t}{\sigma_t}\right)$ within Algorithm 2. Here, σ_t represents the square root of the predefined forward variance schedule, and $\alpha_t = \sqrt{1-\sigma_t^2}$. During the sampling phase, we iterate i backward from M to 1, identifying intermediate timesteps s_i that fall between t_{i-1} and t_i, thus ensuring a sequence $t_0 > s_1 > t_1 > \cdots > t_{M-1} > s_M > t_M$. To evaluate the performance of integrating TAC-Diffusion with DPM-Solver++, we conduct experiment on CIFAR-10, comparing our results with those of Q-Diffusion [24]. The quantitative results are detailed in Tab. 4.

Algorithm 2 Timestep-Aware Correction with DPM-Solver++(2S) Sampler

Input: Quantized noise estimation model $\hat{\boldsymbol{\epsilon}}_\theta$, data prediction model $\mathbf{x}_\theta$, reconstruction coefficient $\mathbf{K}$ and initial data $\mathbf{x}_T$
Output: Corrected Output $\tilde{\mathbf{x}}_{t_M}$
$\hat{\mathbf{x}}_{t_0} \leftarrow \mathbf{x}_T$
for $i \leftarrow 1$ to M **do**
  $h_i \leftarrow \lambda_{t_i} - \lambda_{t_{i-1}}$
  $r_i \leftarrow \frac{\lambda_{s_i} - \lambda_{t_{i-1}}}{h_i}$
  $\tilde{\boldsymbol{\epsilon}}_\theta\left(\hat{\mathbf{x}}_{t_{i-1}}, t_{i-1}\right) \leftarrow \mathbf{K}_{t_{i-1}} \hat{\boldsymbol{\epsilon}}_\theta\left(\hat{\mathbf{x}}_{t_{i-1}}, t_{i-1}\right)$
  $\mathbf{x}_\theta\left(\hat{\mathbf{x}}_{t_{i-1}}, t_{i-1}\right) \leftarrow \frac{1}{\alpha_{t_{i-1}}}\left(\hat{\mathbf{x}}_{t_{i-1}} - \sigma_{t_{i-1}} \tilde{\boldsymbol{\epsilon}}_\theta\left(\hat{\mathbf{x}}_{t_{i-1}}, t_{i-1}\right)\right)$
  $\mathbf{u}_i \leftarrow \frac{\sigma_{s_i}}{\sigma_{t_{i-1}}} \hat{\mathbf{x}}_{t_{i-1}} - \alpha_{s_i}\left(e^{-r_i h_i} - 1\right) \mathbf{x}_\theta\left(\hat{\mathbf{x}}_{t_{i-1}}, t_{i-1}\right)$
  $\tilde{\boldsymbol{\epsilon}}_\theta\left(\mathbf{u}_i, s_i\right) \leftarrow \mathbf{K}_{s_i} \hat{\boldsymbol{\epsilon}}_\theta\left(\mathbf{u}_i, s_i\right)$
  $\mathbf{x}_\theta\left(\mathbf{u}_i, s_i\right) \leftarrow \frac{1}{\alpha_{s_i}}\left(\mathbf{u}_i - \sigma_{s_i} \tilde{\boldsymbol{\epsilon}}_\theta\left(\mathbf{u}_i, s_i\right)\right)$
  $\mathbf{D}_i \leftarrow \left(1 - \frac{1}{2r_i}\right) \mathbf{x}_\theta\left(\hat{\mathbf{x}}_{t_{i-1}}, t_{i-1}\right) + \frac{1}{2r_i} \mathbf{x}_\theta\left(\mathbf{u}_i, s_i\right)$
  $\tilde{\mathbf{x}}_{t_i} \leftarrow \frac{\sigma_{t_i}}{\sigma_{t_{i-1}}} \hat{\mathbf{x}}_{t_{i-1}} - \alpha_{t_i}\left(e^{-h_i} - 1\right) \mathbf{D}_i$
end for
return $\tilde{\mathbf{x}}_{t_M}$

Table 4: Unconditional generation results on CIFAR-10 (32×32), with a W3A8 diffusion model and a 50 steps DPM-Solver++

Method	**Bits(W/A)**	**FID↓**
Q-Diffusion [24]	4/32	5.38
Ours	4/32	**5.29**
Q-Diffusion [24]	4/8	10.27
Ours	4/8	**10.05**
Q-Diffusion [24]	3/8	38.82
Ours	3/8	**18.70**

E Model Efficiency

In this section, we test the efficiency of the quantized diffusion model relative to its full-precision counterpart. We employed the official PyTorch Quantization API for model quantization. Given that this API does not support quantization to precisions lower than 8-bit, we quantized both the weights and activations to 8-bit precision. Tab. 5 showcases the average inference time for a 100-step DDIM process on the CIFAR-10 dataset, conducted on an Intel Xeon Platinum 8358 CPU. Operating with a batch size of 32, the quantized diffusion model achieves a speed-up of morethan 3.9 times, while its size is diminished to about one-fourth of that of the full-precision model. Furthermore, we note that the additional computational overhead for our proposed method is minimal, resulting in a mere 0.65% increase in inference time compared to the Q-Diffusion [24] with a batch size of 32.

Table 5: Inference speed test on CIFAR-10 (32×32), with pixel-space DDIM.

Model	Method	Batch Size	Bits (W/A)	Size (Mb)	Time(s)	Acceleration(×)
DDIM (steps = 100 eta = 0.0)	Full-Precision	64	32/32	143.20	77.95	1
	Q-Diffusion [24]	64	8/8	36.21	26.79	2.91
	Ours	64	8/8	36.21	26.98	2.89
	Full-Precision	32	32/32	143.20	36.18	1
	Q-Diffusion [24]	32	8/8	36.21	9.17	3.95
	Ours	32	8/8	36.21	9.23	3.92
	Full-Precision	16	32/32	143.20	13.48	1
	Q-Diffusion [24]	16	8/8	36.21	5.86	2.30
	Ours	16	8/8	36.21	6.03	2.24
	Full-Precision	1	32/32	143.20	3.59	1
	Q-Diffusion [24]	1	8/8	36.21	2.69	1.33
	Ours	1	8/8	36.21	2.76	1.30

F Visualization of Dynamic Activation Distribution in Noise Estimation Network

We visualize the activation distribution in several layers of LDM-8 during the denoising process in Fig. 5. We can observe that the range of activation varies greatly across timesteps in these layers. Since low-precision diffusion models maintain a fixed quantization step size, a significant portion of activation values inevitably becomes clamped during numerous timesteps. This clamping phenomenon, occurring in many timesteps, leads to substantial information loss.

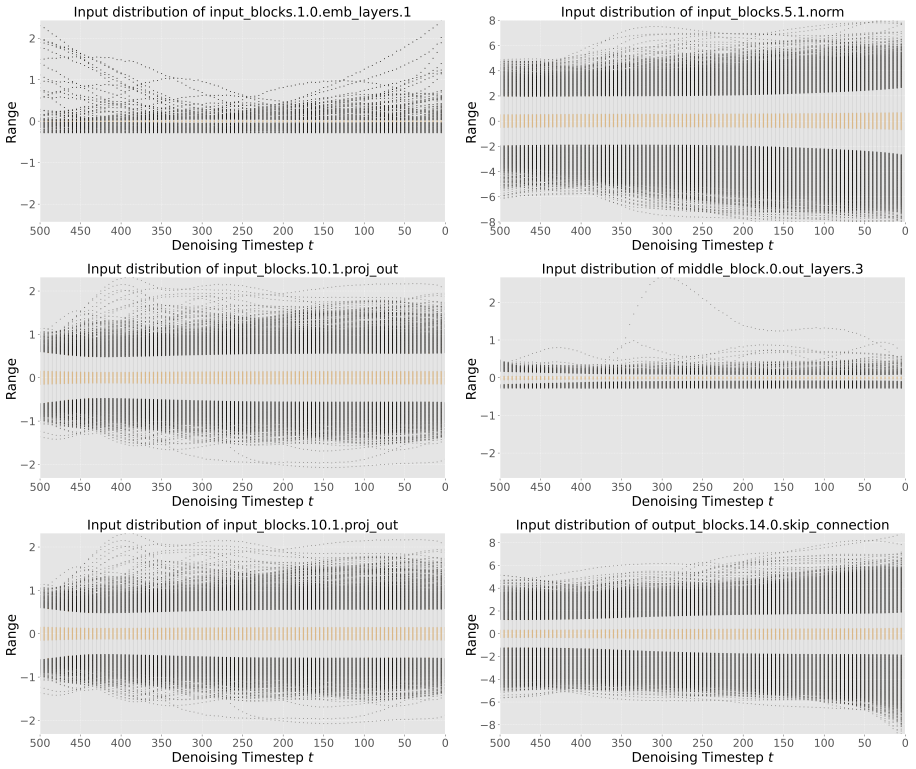


Fig. 5: The activation distribution of multiple layers in full-precision LDM-8 on LSUN-Church. The distribution varies during the denoising process. This dynamic nature of activation is the main source of clipping error in low-precision diffusion model.

G Ablation Study on rQSNR weight in the Reconstruction Loss Function

In this section, we conduct an ablation study on the weighting coefficient λ_1 of the rQNSR penalty in the reconstruction loss function. This study employs a W3A8 100-step DDIM on the CIFAR-10 dataset. The analysis, illustrated in Fig. 6, explores the balance between the mean square error and the relative quantization noise sensitivity by uniformly adjusting λ_1 in increments of 0.1. This ensures that both λ_1 and $1-\lambda_1$ remain within positive bounds. The fitted curve to the observed data points identifies an trade-off between these two components in the reconstruction loss function. While minimizing MSE is a prevalent strategy in numerous post-training quantization methods [1, 5], our findings suggest that integrating a balanced consideration of both absolute and relative error can enhance reconstruction outcomes in quantized diffusion models, thereby leading to improved noise estimation fidelity.

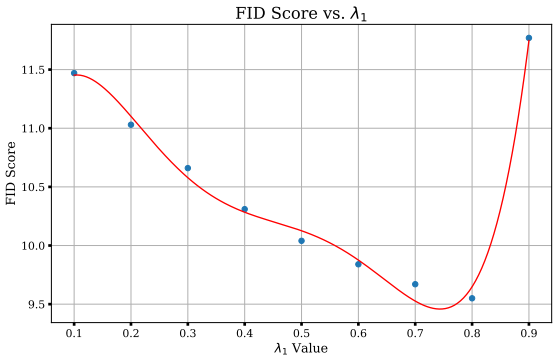


Fig. 6: Ablation study on rQSNR weight

H Potential Negative Impact

The ability to create semantically coherent and visually compelling images with ease raises concerns over the potential misuse of such technology. It can be exploited to generate fake or misleading content, including deepfakes, that can have serious ramifications in areas such as politics, security, and personal reputation. While our method enhances the fidelity of generated images on low-precision devices, it also necessitates the development and enforcement of ethical guidelines and technological solutions to detect and prevent the misuse of synthetic media.

I Limitations

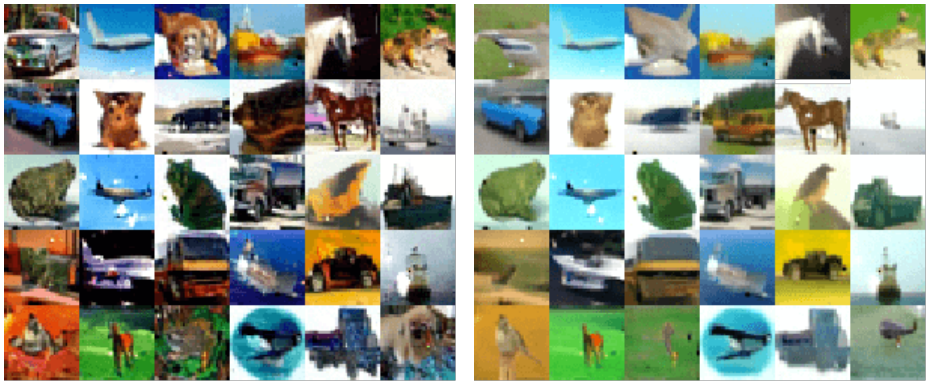
In this study, our primary focus is on addressing the accumulation of quantization errors introduced by the dynamic nature of diffusion models. Extensive experiments conducted on diverse datasets demonstrate that, with the input correction at each timestep, low-precision diffusion models can effectively mitigate the accumulation of quantization errors, resulting in image quality comparable to that of full-precision diffusion models. However, it is important to note that our proposed method is applied exclusively to the model's input and the noise estimation, suggesting that quantization errors may still impact the model's inference at each timestep. Therefor, a more fine-grained correction strategy, such as correction within the residual block, might further improve the performance of quantized models. Moreover, we acknowledge the potential alternative approaches for mitigating quantization errors in low-precision diffusion models, *e.g.* adaptive step size. We leave the exploration of these approaches as future work.

J Qualitative Result on CIFAR10

In this section, we present the quantitative result from experiments conducted on the CIFAR-10 [22]. The generated images using the PTQ4DM [47] and Q-

Diffusion [24], both implemented with W3A8 100 steps DDIM, are illustrated in Fig. 7. Additionally, we display results achieved with the W3A8 50 steps DPM-Solver++ in Fig. 8.

Fig. 7: Unconditional image generation using the W3A8 100 steps DDIM on the CIFAR-10 dataset. The presented sequences, from top to bottom, are Full Precision model, TAC-Diffusion, and Q-Diffusion [24].



(a) TAC-Diffusion **(b)** Q-Diffusion

Fig. 8: Unconditional image generation using the W3A8 50 steps DPM-Solver++ on the CIFAR-10 dataset.

K Results on ImageNet

To further explore the performance of our method on ImageNet, we report the LPIPS, PSNR, SSIM, and FID in Tab. 6. Visualization results with W3A8 precision are provided in Fig. 9.

Table 6: Conditional generation results on ImageNet (256 × 256), with 20 steps LDM-8

Methods	Bits (W/A)	LPIPS↓	PSNR↑	SSIM↑	FID↓
Full-Precision	32/32	–	–	–	10.91
TFMQ-DM [19]	8/8	0.026	33.59	0.958	**10.79**
Ours	8/8	**0.023**	**34.14**	**0.962**	10.82
TFMQ-DM [19]	3/8	0.227	20.61	0.776	8.62
Ours	3/8	**0.206**	**22.15**	**0.788**	**8.36**

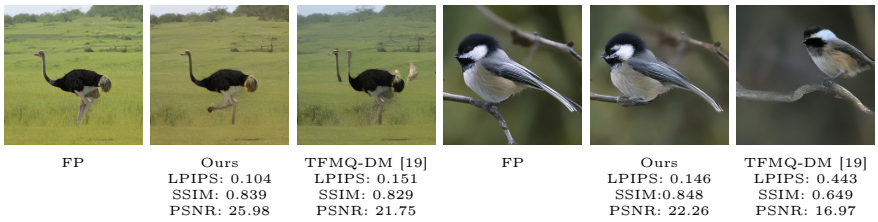


Fig. 9: Conditional generation on ImageNet (256 × 256) with W3A8 20 steps LDM-8

While LPIPS, SSIM, and PSNR evaluate the similarity between images generated by the quantized and full-precision models, improved performance on these metrics indicates our method's ability to enhance the performance of quantized models towards that of the FP models.

L Quantitative Results on LSUN-Bedroom

In this sectiom, we provide more quantitative results with diffusion models of extremely low precision. Unconditional generation results on LSUN-Bedroom with W3A8 and W2A8 LDM-4 are visualized in Figs. 10 and 11. A constant improvement in image quality can be observed. Notably, when the diffusion model is quantized to 2-bit, our method can still guarantee the quality of generated image.

M Conditional Generation with Stabel Diffusion

In text-guided image generation using Stable Diffusion [39], the diffusion model provides estimates for two types of noise at each timestep: $\boldsymbol{\epsilon}_{uc}$ for unconditional noise in the input image and $\boldsymbol{\epsilon}_c$ for conditional noise, which is closely tied to the given prompt. During the collection of calibration samples for implementing our proposed method, we observed that the conditional noise associated with the input text can exhibit significant diversity when compared to unconditional noise. Consequently, a considerably larger set of prompts may be required to comprehensively capture the entire distribution of conditional noise. A practical approach is to focus solely on correcting the unconditional noise. We present images synthesized using a 50-step PLMS sampler and a 50-step DDIM sampler, as illustrated in Figs. 12 and 13. Compared to Q-Diffusion, our method shows remarkable improvements, especially in creating more accurate human faces and more accurately depicting the number of objects as specified in the prompt.

(a) LDM-8 FP32

(b) TAC-Diffusion W3A8

(c) Q-Diffusion [24] W3A8

(d) PTQD W3A8

Fig. 10: 256 × 256 unconditional generation on LSUN-Bedroom with W3A8 200 steps LDM-4.

(a) LDM-8 FP32

(b) TAC-Diffusion W2A8

(c) Q-Diffusion [24] W2A8

Fig. 11: 256 × 256 unconditional generation on LSUN-Bedroom with W2A8 200 steps LDM-4.

(a) Stable Diffusion FP32 with 50 steps PLMS sampler

(b) TAC-Diffusion W4A8 with 50 steps PLMS sampler

(c) Q-Diffusion W4A8 with 50 steps PLMS sampler

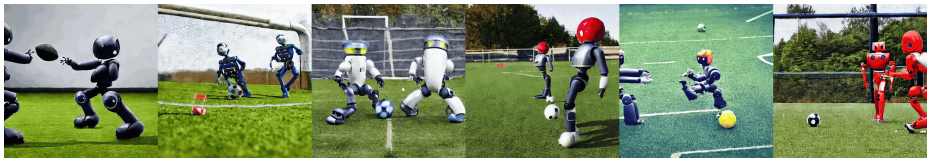
(d) Stable Diffusion FP32 with 50 steps DDIM sampler

(e) TAC-Diffusion W4A8 with 50 steps DDIM sampler

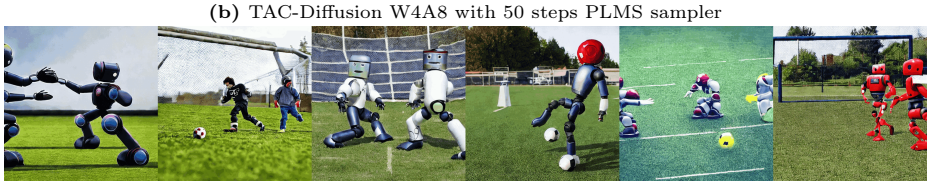
(f) Q-Diffusion W4A8 with 50 steps DDIM sampler

Fig. 12: Text-to-image generation at a resolution of 512 × 512 using Stable Diffusion, with prompt *A photograph of an astronaut playing piano.*

(a) Stable Diffusion FP32 with 50 steps PLMS sampler



(b) TAC-Diffusion W4A8 with 50 steps PLMS sampler



(c) Q-Diffusion W4A8 with 50 steps PLMS sampler

(d) Stable Diffusion FP32 with 50 steps DDIM sampler

(e) TAC-Diffusion W4A8 with 50 steps DDIM sampler

(f) Q-Diffusion W4A8 with 50 steps DDIM sampler

Fig. 13: Text-to-image generation at a resolution of 512 × 512 using Stable Diffusion, with prompt *A photo of two robots playing football.*