On Distillation of Guided Diffusion Models

Chenlin Meng[1] chenlin@cs.stanford.edu
Robin Rombach[2] robin@stability.ai
Ruiqi Gao[3] ruiqig@google.com
Diederik Kingma[3] durk@google.com
Stefano Ermon[1] ermon@cs.stanford.edu
Jonathan Ho[3] jonathanho@google.com
Tim Salimans[3] salimans@google.com

[1]Stanford University [2]Stability AI & LMU Munich [3] Google Research, Brain Team

Figure 1. Distilled Stable Diffusion samples generated by our method. Our two-stage distillation approach is able to generate realistic images using only 1 to 4 denoising steps on various tasks. Compared to standard classifier-free guided diffusion models, we reduce the total number of sampling steps by at least 20×.

Abstract

Classifier-free guided diffusion models have recently been shown to be highly effective at high-resolution image generation, and they have been widely used in large-scale diffusion frameworks including DALL·E 2, Stable Diffusion and Imagen. However, a downside of classifier-free guided diffusion models is that they are computationally expensive at inference time since they require evaluating two diffusion models, a class-conditional model and an unconditional model, tens to hundreds of times. To deal with this limitation, we pro-

*Work partially done during an internship at Google

*pose an approach to distilling classifier-free guided diffusion models into models that are fast to sample from: Given a pre-trained classifier-free guided model, we first learn a single model to match the output of the combined conditional and unconditional models, and then we progressively distill that model to a diffusion model that requires much fewer sampling steps. For standard diffusion models trained on the pixel-space, our approach is able to generate images visually comparable to that of the original model using as few as 4 sampling steps on ImageNet 64x64 and CIFAR-10, achieving FID/IS scores comparable to that of the original model while being up to 256 times faster to sample from. For diffusion models trained on the latent-space (*e.g.*, Stable Diffusion), our approach is able to generate high-fidelity images using as few as 1 to 4 denoising steps, accelerating inference by at least 10-fold compared to existing methods on ImageNet 256x256 and LAION datasets. We further demonstrate the effectiveness of our approach on text-guided image editing and inpainting, where our distilled model is able to generate high-quality results using as few as 2-4 denoising steps.*

1. Introduction

Denoising diffusion probabilistic models (DDPMs) [4, 37, 39, 40] have achieved state-of-the-art performance on image generation [22, 26–28, 31], audio synthesis [11], molecular generation [44], and likelihood estimation [10]. Classifier-free guidance [6] further improves the sample quality of diffusion models and has been widely used in large-scale diffusion model frameworks including GLIDE [23], Stable Diffusion [28], DALL·E 2 [26], and Imagen [31]. However, one key limitation of classifier-free guidance is its low sampling efficiency—it requires evaluating two diffusion models tens to hundreds of times to generate one sample. This limitation has hindered the application of classifier-free guidance models in real-world settings. Although distillation approaches have been proposed for diffusion models [33, 38], these approaches are not directly applicable to classifier-free guided diffusion models. To deal with this issue, we propose a two-stage distillation approach to improving the sampling efficiency of classifier-free guided models. In the first stage, we introduce a single student model to match the combined output of the two diffusion models of the teacher. In the second stage, we *progressively distill* the model learned from the first stage to a fewer-step model using the approach introduced in [33]. Using our approach, a *single* distilled model is able to handle a wide range of different guidance strengths, allowing for the trade-off between sample quality and diversity efficiently. To sample from our model, we consider existing deterministic samplers in the literature [33, 38] and further propose a stochastic sampling process.

Our distillation framework can not only be applied to standard diffusion models trained on the pixel-space [4, 36, 39], but also diffusion models trained on the latent-space of an autoencoder [28, 35] (*e.g.*, Stable Diffusion [28]). For diffusion models directly trained on the pixel-space, our experiments on ImageNet 64x64 and CIFAR-10 show that the proposed distilled model can generate samples visually comparable to that of the teacher using only 4 steps and is able to achieve comparable FID/IS scores as the teacher model using as few as 4 to 16 steps on a wide range of guidance strengths (see Fig. 2). For diffusion model trained on the latent-space of an encoder [28, 35], our approach is able to achieve comparable visual quality to the base model using as few as 1 to 4 sampling steps (at least 10× fewer steps than the base model) on ImageNet 256×256 and LAION 512×512, matching the performance of the teacher (as evaluated by FID) with only 2-4 sampling steps. To the best of our knowledge, our work is the first to demonstrate the effectiveness of distillation for both pixel-space and latent-space classifier-free diffusion models. Finally, we apply our method to text-guided image inpainting and text-guided image editing tasks [20], where we reduce the total number of sampling steps to as few as 2-4 steps, demonstrating the potential of the proposed framework in style-transfer and image-editing applications [20, 41].

Figure 2. Class-conditional samples from our two-stage (deterministic) approach on ImageNet 64x64 for diffusion models trained on the pixel-space. By varying the guidance weight w, our distilled model is able to trade-off between sample diversity and quality, while achieving good results using as few as *one* sampling step.

2. Background on diffusion models

Given samples $\mathbf{x}$ from a data distribution $p_{\text{data}}(\mathbf{x})$, noise scheduling functions α_t and σ_t, we train a diffusion model $\hat{\mathbf{x}}_{\boldsymbol{\theta}}$, with parameter $\boldsymbol{\theta}$, via minimizing the weighted mean squared error [4, 36, 39, 40]

$$\mathbb{E}_{t\sim U[0,1],\mathbf{x}\sim p_{\text{data}}(\mathbf{x}),\mathbf{z}_t\sim q(\mathbf{z}_t|\mathbf{x})}[\omega(\lambda_t)||\hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{z}_t)-\mathbf{x}||_2^2], \quad (1)$$

where $\lambda_t = \log[\alpha_t^2/\sigma_t^2]$ is a signal-to-noise ratio [10], $q(\mathbf{z}_t|\mathbf{x}) = \mathcal{N}(\mathbf{z}_t; \alpha_t\mathbf{x}, \sigma_t^2\mathbf{I})$ and $\omega(\lambda_t)$ is a pre-specified weighting function [10].

Once the diffusion model $\hat{\mathbf{x}}_{\boldsymbol{\theta}}$ is trained, one can use discrete-time DDIM sampler [38] to sample from the model. Specifically, the DDIM sampler starts with $\mathbf{z}_1 \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$

and updates as follows

$$\mathbf{z}_s = \alpha_s \hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{z}_t) + \sigma_s \frac{\mathbf{z}_t - \alpha_t \hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{z}_t)}{\sigma_t}, \; s = t - 1/N \quad (2)$$

with N the total number of sampling steps. The final sample will then be generated using $\hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{z}_0)$.

Classifier-free guidance Classifier-free guidance [6] is an effective approach shown to significantly improve the sample quality of class-conditioned diffusion models, and has been widely used in large-scale diffusion models including GLIDE [23], Stable Diffusion [28], DALL·E 2 [26] and Imagen [31]. Specifically, it introduces a guidance weight parameter $w \in \mathbb{R}^{\geq 0}$ to trade-off between sample quality and diversity. To generate a sample, classifier-free guidance evaluates both a conditional diffusion model $\hat{\mathbf{x}}_{c,\boldsymbol{\theta}}$, where c is the context (*e.g.*, class label, text prompt) to be conditioned on, and a jointly trained unconditional diffusion model $\hat{\mathbf{x}}_{\boldsymbol{\theta}}$ at each update step, using $\hat{\mathbf{x}}^w_{\boldsymbol{\theta}} = (1+w)\hat{\mathbf{x}}_{c,\boldsymbol{\theta}} - w\hat{\mathbf{x}}_{\boldsymbol{\theta}}$ as the model prediction in Eq. (2). As each sampling update requires evaluating two diffusion models, sampling with classifier-free guidance is often expensive [6].

Progressive distillation Our approach is inspired by *progressive distillation* [33], an effective method for improving the sampling speed of (unguided) diffusion models by repeated distillation. Until now, this method could not be directly applied to distilling classifier-free guided models or studied for samplers other than the deterministic DDIM sampler [33, 38]. In this paper we resolve these shortcomings.

Latent diffusion models (LDMs) [21, 24, 28, 35] increase the training and inference efficiency of diffusion models (directly learned on the pixel-space) by modeling images in the latent space of a pre-trained regularized autoencoder, where the latent representations are usually of lower dimensionality than the pixel-space. Latent diffusion models can be considered as an alternative to cascaded diffusion approaches [5], which rely on one or more super-resolution diffusion models to scale up a low-dimensional image to the desired target resolution.

In this work, we will apply our distillation framework to classifier-free guided diffusion models learned on both pixel-space [4, 36, 39] and latent-space [21, 24, 28, 35].

3. Distilling a guided diffusion model

In the following, we discuss our approach for distilling a classifier-free guided diffusion model [6] into a student model that requires fewer steps to sample from. Using a *single* distilled model conditioned on the guidance strength, our model can capture a wide range of classifier-free guidance levels, allowing for the trade-off between sample quality and diversity efficiently.

Given a trained guided model $[\hat{\mathbf{x}}_{c,\boldsymbol{\theta}}, \hat{\mathbf{x}}_{\boldsymbol{\theta}}]$ (teacher) either on the pixel-space or latent-space, our approach can be decomposed into two stages.

3.1. Stage-one distillation

In the first stage, we introduce a student model $\hat{\mathbf{x}}_{\boldsymbol{\eta}_1}(\mathbf{z}_t, w)$, with learnable parameter $\boldsymbol{\eta}_1$, to match the output of the teacher at any time-step $t \in [0,1]$. The student model can either be a continuous-time model [40] or a discrete-time model [4, 38] depending on whether the teacher model is discrete or continuous. For simplicity, in the following discussion, we assume both the student and teacher models are continuous as the algorithm for discrete models is almost identical.

A key functionality of classifier-free guidance [6] is its ability to easily trade-off between sample quality and diversity, which is controlled by a "guidance strength" parameter. This property has demonstrated utility in real-world applications [6, 23, 26, 28, 31], where the optimal "guidance strength" is often a user preference. Thus, we would also want our distilled model to maintain this property. Given a range of guidance strengths $[w_{\text{min}}, w_{\text{max}}]$ we are interested in, we optimize the student model using the following objective

$$\mathbb{E}_{w\sim p_w, t\sim U[0,1], \mathbf{x}\sim p_{\text{data}}(\mathbf{x})}\Big[\omega(\lambda_t)\|\hat{\mathbf{x}}_{\boldsymbol{\eta}_1}(\mathbf{z}_t, w) - \hat{\mathbf{x}}^w_{\boldsymbol{\theta}}(\mathbf{z}_t)\|_2^2\Big], \quad (3)$$

where $\hat{\mathbf{x}}^w_{\boldsymbol{\theta}}(\mathbf{z}_t) = (1+w)\hat{\mathbf{x}}_{c,\boldsymbol{\theta}}(\mathbf{z}_t) - w\hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{z}_t)$, $\mathbf{z}_t \sim q(\mathbf{z}_t|\mathbf{x})$ and $p_w(w) = U[w_{\text{min}}, w_{\text{max}}]$. Note that here, our distilled model $\hat{\mathbf{x}}_{\boldsymbol{\eta}_1}(\mathbf{z}_t, w)$ is also conditioned on the context c (*e.g.*, text prompt), but we drop the notation c in the paper for simplicity. We provide the detailed training algorithm in Algorithm 1 in the supplement.

To incorporate the guidance weight w, we introduce a w-conditioned model, where w is fed as an input to the student model. To better capture the feature, we apply Fourier embedding to w, which is then incorporated into the diffusion model backbone in a way similar to how the time-step was incorporated in [10, 33]. As initialization plays a key role in the performance [33], we initialize the student model with the same parameters as the conditional model of the teacher, except for the newly introduced parameters related to w-conditioning. The model architecture we use is a U-Net model similar to the ones used in [6] for pixel-space diffusion models and [1, 28] for latent-space diffusion models. We use the same number of channels and attention as used in [6] and the open-sourced Stable Diffusion repository* for our experiments. We provide more details in the supplement.

3.2. Stage-two distillation

In the second stage, we consider a discrete time-step scenario and progressively distill the learned model from the first-stage $\hat{\mathbf{x}}_{\boldsymbol{\eta}_1}(\mathbf{z}_t, w)$ into an fewer-step student model $\hat{\mathbf{x}}_{\boldsymbol{\eta}_2}(\mathbf{z}_t, w)$ with learnable parameter $\boldsymbol{\eta}_2$, by halving the number of sampling steps each time. Letting N denote the number of sampling steps, given $w \sim U[w_{\text{min}}, w_{\text{max}}]$ and

*https://github.com/CompVis/stable-diffusion

(a) 2 denoising steps (b) 4 denoising steps (c) 8 denoising steps

Figure 4. Text-guided generation on LAION (512x512) using our distilled Stable Diffusion model. Our model is able to generate high-quality image samples using 2, 4 or 8 denoising steps, significantly improving the inference efficiency of Stable Diffusion.

	$w = 0$		$w = 0.3$		$w = 1$		$w = 4$	
Method	FID (↓)	IS (↑)	FID (↓)	IS (↑)	FID (↓)	IS (↑)	FID (↓)	IS (↑)
Ours 1-step (D/S)	22.74 / 26.91	25.51 / 23.55	14.85 / 18.48	37.09 / 33.30	7.54 / 8.92	75.19 / 67.80	18.72 / **17.85**	157.46 / 148.97
Ours 4-step (D/S)	4.14 / 3.91	46.64 / 48.92	2.17 / 2.24	69.64 / 73.73	7.95 / 8.51	128.98 / 135.36	26.45 / 27.33	207.45 / 216.56
Ours 8-step (D/S)	2.79 / 2.44	50.72 / 55.03	**2.05** / 2.31	76.01 / 83.00	9.33 / 10.56	136.47 / 147.39	26.62 / 27.84	203.47 / **219.89**
Ours 16-step (D/S)	2.44 / **2.10**	52.53 / **57.81**	2.20 / 2.56	79.47 / **87.50**	9.99 / 11.63	139.11 / **153.17**	26.53 / 27.69	204.13 / 218.70
Single-w 1-step	19.61	24.00	11.70	36.95	**6.64**	74.41	19.857	170.69
Single-w 4-step	4.79	38.77	2.34	62.08	8.23	118.52	27.75	219.64
Single-w 8-step	3.39	42.13	2.32	68.76	9.69	125.20	27.67	218.08
Single-w 16-step	2.97	43.63	2.56	70.97	10.34	127.70	27.40	216.52
DDIM 16x2-step [38]	7.68	37.60	5.33	60.83	9.53	112.75	21.56	195.17
DDIM 32x2-step [38]	5.03	40.93	7.47	9.33	9.26	126.22	23.03	213.23
DDIM 64x2-step [38]	3.74	43.16	5.52	9.51	9.53	133.17	23.64	217.88
Teacher (DDIM 1024x2-step)	2.92	44.81	2.36	74.83	9.84	139.50	23.94	224.74

Table 1. ImageNet 64x64 distillation results for pixel-space diffusion models ($w = 0$ refers to non-guided models). For our method, D and S stand for deterministic and stochastic sampler respectively. We observe that training the model conditioned on a guidance interval $w \in [0, 4]$ performs comparably with training a model on a fixed w (see Single-w). Our approach significantly outperforms DDIM when using fewer steps, and is able to match the teacher performance using as few as 8 to 16 steps.

$t \in \{1, ..., N\}$, we train the student model to match the output of two-step DDIM sampling of the teacher (i.e., from t/N to $t - 0.5/N$ and from $t - 0.5/N$ to $t - 1/N$) in one step, following the approach of [33]. After distilling the $2N$ steps in the teacher model to N steps in the student model, we can use the N-step student model as the new teacher model, repeat the same procedure, and distill the teacher model into a $N/2$-step student model. At each step, we initialize the student model with the parameters of the teacher. We provide the training algorithm and extra details in the supplementary material.

3.3. N-step deterministic and stochastic sampling

Once the model $\hat{\mathbf{x}}_{\boldsymbol{\eta}_2}$ is trained, given a specified guidance strength $w \in [w_{\min}, w_{\max}]$, we can perform sampling via the DDIM update rule in Eq. (2). We note that given the distilled model $\hat{\mathbf{x}}_{\boldsymbol{\eta}_2}$, this sampling procedure is *deterministic* given the initialization $\mathbf{z}_1^w$. In fact, we can also perform N-step *stochastic* sampling: We apply one deterministic sampling step with two-times the original step-length (i.e., the same as a $N/2$-step deterministic sampler) and then perform one stochastic step backward (i.e., perturb with noise) using the original step-length, a process inspired by [9]. With $\mathbf{z}_1^w \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, we use the following update rule when $t > 1/N$

$$\mathbf{z}_k^w = \alpha_k \hat{\mathbf{x}}_{\boldsymbol{\eta}_2}(\mathbf{z}_t^w) + \sigma_k \frac{\mathbf{z}_t^w - \alpha_t \hat{\mathbf{x}}_{\boldsymbol{\eta}_2}^w(\mathbf{z}_t)}{\sigma_t}, \quad (4)$$

$$\text{where } \mathbf{z}_s^w = (\alpha_s/\alpha_k)\mathbf{z}_k^w + \sigma_{s|k}\boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}); \quad (5)$$

$$\mathbf{z}_h^w = \alpha_h \hat{\mathbf{x}}_{\boldsymbol{\eta}_2}(\mathbf{z}_s^w) + \sigma_h \frac{\mathbf{z}_s^w - \alpha_s \hat{\mathbf{x}}_{\boldsymbol{\eta}_2}^w(\mathbf{z}_s^w)}{\sigma_s}, \quad (6)$$

$$\text{where } \mathbf{z}_k^w = (\alpha_k/\alpha_h)\mathbf{z}_h^w + \sigma_{k|h}\boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \quad (7)$$

In the above equations, $h = t - 3/N$, $k = t - 2/N$, $s = t - 1/N$ and $\sigma_{a|b}^2 = (1 - e^{\lambda_a - \lambda_b})\sigma_a^2$. When $t = 1/N$, we use deterministic update Eq. (2) to obtain $\mathbf{z}_0^w$ from $\mathbf{z}_{1/N}^w$. We provide an illustration of the process in Fig. 5, where the number of denoising steps is 4. We note that compared to the *deterministic* sampler, performing *stochastic* sampling requires evaluating the model at slightly different time-steps,

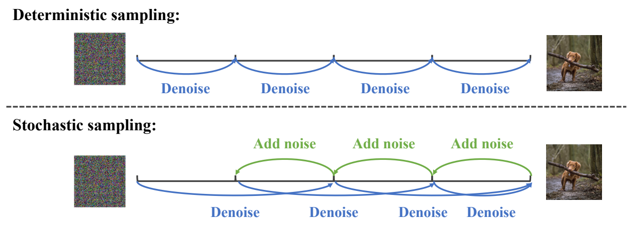


Figure 5. Sampling procedures of the distilled model where the number of denoising steps is 4.

and would require small modifications to training algorithm for the edge cases. We provide the algorithm and more details in the supplementary material.

4. Experiments

In this section, we evaluate the performance of our distillation approach on pixel-space diffusion models (*i.e.* DDPM [4]) and latent-space diffusion models (*i.e.* Stable Diffusion [28]). We further apply our approach to text-guided image editing and inpainting tasks. Experiments show that our approach is able to achieve competitive performance while using as few as 2-4 steps on all tasks.

4.1. Distillation for pixel-space guided models

In this experiment, we consider class-conditional diffusion models directly trained on the pixel-space [4, 6, 33].

Settings We focus on ImageNet 64x64 [30] and CIFAR-10 [12] as higher-resolution image generation in this scenario often relies on combining with other super-resolution techniques [5, 31]. We explore different ranges for the guidance weight and observe that all ranges work comparably and therefore use $[w_{min}, w_{max}] = [0, 4]$ for the experiments. The baselines we consider include DDPM ancestral sampling [4] and DDIM [38]. The teacher model we use is a 1024x2-step DDIM model, where the conditional and unconditional components both use 1024 DDIM denoising steps. To better understand how the guidance weight w should be incorporated, we also include models trained using a single fixed w as a baseline. We use the same pre-trained teacher model for all the methods for fair comparisons. Following [4, 6, 39], we use a U-Net [29, 39] architecture for the baselines, and the same U-Net backbone with the introduced w-embedding for our two-step student models (see Sec. 3). Following [33], we use a $\mathbf{v}$-prediction model for both datasets.

Results We report the performance as evaluated in FID [3] and Inception scores (IS) [32] for all approaches on ImageNet 64x64 in Fig. 6 and Tab. 1 and provide extended ImageNet 64x64 and CIFAR-10 results in the supplement. We observe that our distilled model is able to match a teacher guided DDIM model with 1024x2 sampling steps using only 4-16 steps, achieving a speedup for up to 256×. We emphasize that, using our approach, a *single* distilled model is able to match the teacher performance on a wide range of guidance strengths. This has not been achieved by any previous methods.

4.2. Distillation for latent-space guided models

After demonstrating the effectiveness of our method on pixel-space class-guided diffusion models in Sec. 4.1, we now expand its scope to latent-space diffusion models. In the following sections, we show the effectiveness of our approach on Latent Diffusion [28] on a variety of tasks, including class-conditional generation, text-to-image generation, image inpainting and text-guided style-transfer [20].

In the following experiments, we use the open-sourced latent-space diffusion models [28] as the teacher models. As $\mathbf{v}$-prediction teacher model tends to perform better than ϵ-prediction model, we *fine-tune* the open-sourced ϵ-prediction models into $\mathbf{v}$-prediction teacher models. We provide more details in the supplementary material.

4.2.1 Class-conditional generation

In this section, we apply our method to a class-conditional latent diffusion model pre-trained on ImageNet 256×256. We start from the DDIM teacher model with 512 sampling steps, and use the output as the target to train our distilled model. We use a batch size of 512 and uniformly sample the guidance strength $w \in [w_{min} = 0, w_{max} = 14]$ during training.

Results Empirically, we find that our distilled model is able to match the performance of the teacher model (originally trained on 1000 steps) in terms of FID scores while using only 2 or 4 sampling steps. We also achieve significantly better performance than DDIM when using 1-4 sampling steps (see Fig. 11). Qualitatively, we find that samples synthesized using a single denoising step still yield satisfying results, while the baseline fails to generate images with meaningful contents. We provide extra samples in the supplementary material.

Similar to the pixel-based results in Fig. 6, we also observe the trade-off between sampling quality and diversity as measured by FID and Inception Score for our distilled latent diffusion model. Following *Kynkäänniemi et al* [13], we further compute improved precision and recall metrics for this experiment in the appendix.

4.2.2 Text-guided image generation

In this section, we focus on the text-guided Stable Diffusion model pretrained on subsets [†] of LAION-5B [34] at a resolution 512×512. We then follow our two-stage approach introduced in Sec. 3 and distill the guided model in 3000 gradient updates into a w-conditioned model using

[†] https://github.com/CompVis/stable-diffusion/blob/main/Stable_Diffusion_v1_Model_Card.md

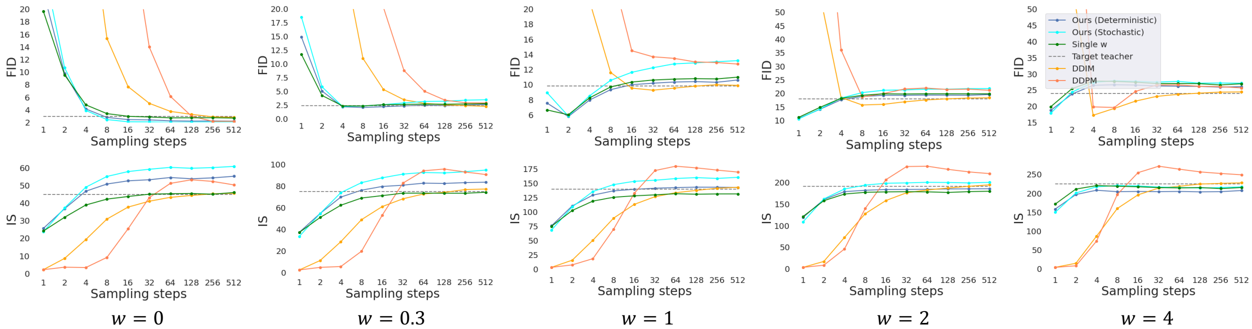


Figure 6. ImageNet 64x64 sample quality evaluated by FID and IS scores. Our distilled model significantly outperform the DDPM and DDIM baselines, and is able to match the performance of the teacher using as few as 8 to 16 steps. By varying w, a *single* distilled model is able to capture the trade-off between sample diversity and quality.

Figure 7. Text-guided Stable Diffusion results. We distill the public *Stable Diffusion* model using the proposed pipeline, arriving at a model that achieves high sample quality using only four denoising steps (*left*). When sampling from the original model using four DDIM steps, the generated samples have clear artifacts (*middle*). When using eight DDIM steps, the results get better (*right*), but are still blurry and less consistent than the distilled results using fewer steps. More samples are provided in Fig. 4.

$w \in [w_{min} = 2, w_{max} = 14]$, and a batch size of 512. Although we can condition on a broader range of w for the distilled (student) model, the utility remains unclear as we typically do not exceed the normal guidance range when sampling with the teacher model. The final model is obtained by applying progressive distillation for 2000 training steps per stage, except when for the low-step regime of 1,2, and 4 steps, where we train for 20000 gradient updates. A detailed analysis of the convergence properties of this model in the supplement.

Method	2-step	4-step	8-step
DPM [16]	98.9/0.20	34.3/0.29	31.7/0.32
DPM++ [18]	98.8/0.20	34.1/0.29	25.6/0.32
Ours	37.3/0.27	26.0/0.30	26.9/0.30

Table 2. FID/CLIP scores on LAION 512X512 ($w = 8.0$). We point out that DPM and DPM++ use both the conditional and unconditional components for sampling. Depending on the implementation, this either requires higher peak memory or two times more sampling steps.

Results We present samples in Fig. 4. We evaluate the resulting model both qualitatively and quantitatively. For the latter analysis, we follow [31] and evaluate CLIP [25] and FID scores to asses text-image alignment and quality, respectively. We use the open-sourced ViT-g/14 [7] CLIP model for evaluation. The quantitative results in Fig. 10 show that our method can significantly increase the performance in both metrics over DDIM sampling on the base model for 2 and 4 sampling steps. For 8 steps, these metrics do not show a significant difference. However, when considering the corresponding samples in Fig. 7 we can observe a stark difference in terms of visual image quality. In contrast to the 8-step DDIM samples from the original model, the distilled samples are sharper and more coherent. We hypothesize that FID and CLIP do not fully capture these differences in our evaluation setting on COCO2017 [14], where we used 5000 random captions from the validation set. We further compute the FID and CLIP scores for our distilled LAION 512x512 model and compare them with the DPM [16] and DPM++ [18] solver in Tab. 2. We observe that our method is able to achieve significantly better performance when the denoising step is 2 or 4. Furthermore, we stress that stage-one of our method already decreases the number of function evaluations by a factor of 2, as we distill the classifier-free guidance step into a single model. Depending on the exact implementation (batched vs. sequential network evaluation), this either decreases peak memory or sampling time compared to existing solvers [16, 18, 38].

4.2.3 Text-guided image-to-image translation

In this section, we perform experiments on text-guided image-to-image translation with SDEdit [20] using our distilled model from Sec. 4.2.2. Following SDEdit [20], we perform stochastic encoding in the latent space, but instead

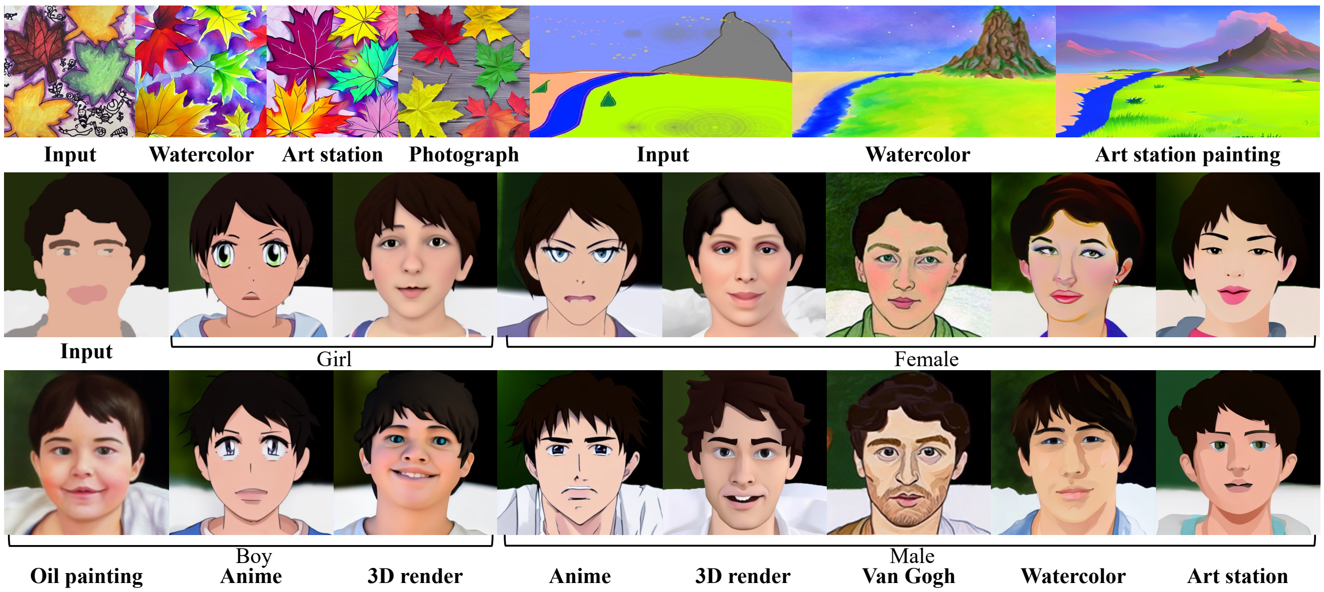


Figure 8. Text-guided image-to-image translation [20] with the distilled Stable Diffusion model (3 denoising steps). We observe that our model is able to generate high-quality and faithful outputs using only 3 denoising steps.

Figure 9. Image inpainting with our distilled Stable Diffusion model (4 denoising steps). Our model is able to generate high-quality image inpainting results using 4 denoising steps on unseen data.

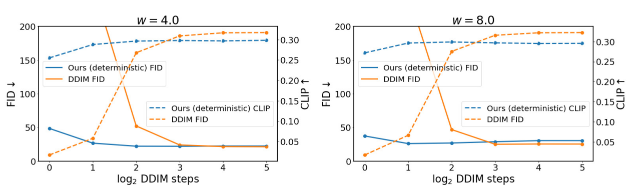


Figure 10. FID and CLIP ViT-g/14 score for text-to-image generation at 512×512 px using the distilled Stable Diffuion model. The results are evaluated on 5000 captions from the COCO2017 [14] validation set. Our distilled latent diffusion model is able to generate high-quality image samples using significantly less sampling steps than the original model while achieving similar or better FID and CLIP scores, especially in the low-step regime.

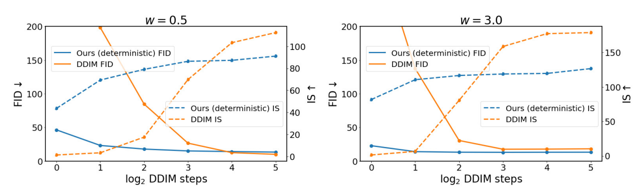


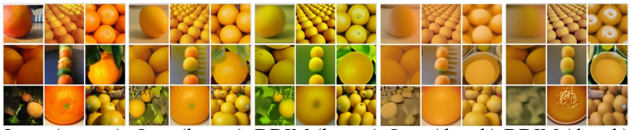
Figure 11. FID and Inception Score for class-conditional image generation on ImageNet (256×256) with distilled latent diffusion. The results are evaluated on 5000 samples. Our distilled latent diffusion model is able to generate high-quality image samples using significantly less sampling steps (up to a factor of 16) than the original model while achieving similar or better FID scores.

use the deterministic sampler of the distilled model to perform deterministic decoding. We consider input image and text of various kinds and provide qualitative results in Fig. 8. We observe that our distilled model generates high-quality style-transfer results using as few as 3 denoising steps. We provide more analysis on the trade-off between sample quality, controllability and efficiency in the supplement.

4.2.4 Image inpainting

In this section, we apply our approach to a pre-trained image inpainting latent diffusion model. We use the open-source *Stable Diffusion Inpainting*[‡] image-inpainting model. This model is a fine-tuned version of the pure text-to-image *Stable Diffusion* model from above, where additional input channels

[‡]https://huggingface.co/runwayml/stable-diffusion-inpainting



Input (orange) Ours (lemon) DDIM (lemon) Ours (dough) DDIM (dough)

Figure 12. Style transfer comparison on ImageNet 64x64 for pixel-space models. For our approach, we use a distilled encoder and decoder. For the baseline, we encode and decode using DDIM. We use $w = 0$ and 16 sampling steps for both the encoder and decoder. We observe that our method achieves more realistic outputs.

were added to process masks and masked images.

We use the same distillation algorithm as used in the previous section. For training, we start from the $\mathbf{v}$-prediction teacher model sampled with 512 DDIM steps, and use the output as the target to optimize our student model. We present qualitative results in Fig. 9, demonstrating the potential of our method for fast, real-world image editing applications. For additional training details and a quantitative evaluation, see the supplementary.

4.3. Progressive distillation for encoding

In this experiment, we explore distilling the encoding process for the teacher model and perform experiments on style-transfer in a setting similar to [41]. We focus on pixel-space diffusion models pre-trained on ImageNet 64× 64. Specifically, to perform style-transfer between two domains A and B, we encode the image from domain-A using a diffusion model trained on domain-A, and then decode with a diffusion model trained on domain-B. As the encoding process can be understood as reversing the DDIM sampling process, we perform distillation for both the encoder and decoder with classifier-free guidance, and compare with a DDIM encoder and decoder in Fig. 12. We also explore how modifying the guidance strength w can impact the performance and provide more details in the supplementary material.

5. Related Work

Our approach is related to existing works on improving the sampling speed of diffusion models [4, 37, 40]. For instance, denoising diffusion implicit models (DDIM [38]), probability flow sampler [40], fast SDE integrators [8] have been proposed to improve the sampling speed of diffusion models. Other works develop higher-order solvers [17], exponential integrators [15], and dynamic programming based approach [43] to accelerating sampling speed. However, none of these approaches have achieved comparable performance as our method on distilling classifier-free guided diffusion models.

Existing distillation-based methods for diffusion models are mainly designed for non-classifier-free guided diffusion models. For instance, [19] proposes to predict the data from noise in one single step by inverting a deterministic encoding of DDIM, [2] proposes to achieve faster sampling speed by distilling higher order solvers into an additional prediction head of the neural network backbone [2]. *Progressive distillation* [33] is perhaps the most relevant work. Specifically, it proposes to progressively distill a pre-trained diffusion model into a fewer-step student model with the same model architecture. However, none of these approaches are directly applicable or have been applied to classifier-free guided diffusion models. They are also unable to capture a range of different guidance strengths using one single distilled model. On the contrary, by incorporating the guidance strength into the model architecture and training the model using a two-stage procedure, our approach is able to match the performance of the teacher model on a wide range of guidance strength using one *single* model. Using our method, one single model can capture the trade-off between sample quality and diversity, enabling the real-world application of classifier-free guided diffusion models, where the guidance strength is often specified by users. Moreover, none of the above distillation approaches have been applied to or shown effectiveness for latent-space text-to-image models. Finally, most fast sampling approaches [33, 38, 40] only consider using deterministic sampling schemes to improve the sampling speed. In this work, we further develop an effective stochastic sampling approach to sample from the distilled models.

6. Conclusion

In this paper, we propose a distillation approach for guided diffusion models [6]. Our two-stage approach allows us to significantly speed up popular but relatively inefficient guided diffusion models. We show that our approach can reduce the inference cost of classifier-free guided pixel-space and latent-space diffusion models by at least an order of magnitude. Empirically, we show that our approach is able to produce visually appealing results with only 2 steps, achieving a comparable FID score to the teacher with as few as 4 to 8 steps. We further demonstrate practical applications of our distillation approach to text-guided image-to-image translation and inpainting tasks. We hope that by significantly reducing the inference cost of classifier-free guided diffusion models, our method will promote creative applications as well as the wider adoption of image generation systems. In the future work, we aim to further improve the performance in the two and one sampling step regimes.

Acknowledgements

We thank the anonymous reviewers for their insightful discussions and feedback. All experiments on Stable Diffusion are supported by Stability AI.

References

[1] Prafulla Dhariwal and Alex Nichol. Diffusion models beat GANs on image synthesis. *arXiv preprint arXiv:2105.05233*, 2021. 3

[2] Tim Dockhorn, Arash Vahdat, and Karsten Kreis. Genie: Higher-order denoising diffusion solvers. In *Advances in Neural Information Processing Systems*, 2022. 8

[3] 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 *Advances in Neural Information Processing Systems*, pages 6626–6637, 2017. 5

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, pages 6840–6851, 2020. 2, 3, 5, 8

[5] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *arXiv preprint arXiv:2106.15282*, 2021. 3, 5

[6] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 2, 3, 5, 8, 11

[7] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. If you use this software, please cite it as below. 6, 24

[8] Alexia Jolicoeur-Martineau, Ke Li, Rémi Piché-Taillefer, Tal Kachman, and Ioannis Mitliagkas. Gotta go fast when generating data with score-based models. *arXiv preprint arXiv:2105.14080*, 2021. 8

[9] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *arXiv preprint arXiv:2206.00364*, 2022. 4

[10] Diederik P Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *arXiv preprint arXiv:2107.00630*, 2021. 2, 3, 11, 22, 23, 25

[11] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. DiffWave: A Versatile Diffusion Model for Audio Synthesis. *International Conference on Learning Representations*, 2021. 2

[12] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[13] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 5, 24

[14] 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 *European conference on computer vision*, pages 740–755. Springer, 2014. 6, 7, 24

[15] Luping Liu, Yi Ren, Zhijie Lin, and Zhou Zhao. Pseudo numerical methods for diffusion models on manifolds. In *International Conference on Learning Representations*, 2022. 8

[16] 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, 2022. 6, 23, 24

[17] 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:2206.00927*, 2022. 8, 11

[18] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver++: Fast solver for guided sampling of diffusion probabilistic models, 2022. 6, 11, 13, 14, 23, 24, 28

[19] Eric Luhman and Troy Luhman. Knowledge distillation in iterative generative models for improved sampling speed. *arXiv preprint arXiv:2101.02388*, 2021. 8

[20] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 2, 5, 6, 7, 25, 26, 27

[21] Gautam Mittal, Jesse Engel, Curtis Hawthorne, and Ian Simon. Symbolic music generation with diffusion models. *arXiv preprint arXiv:2103.16091*, 2021. 3

[22] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. *International Conference on Machine Learning*, 2021. 2

[23] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2, 3

[24] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsa, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10619–10629, 2022. 3

[25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 6, 24

[26] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2, 3

[27] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021. 2

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2, 3, 5

[29] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation.

In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 5

[30] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. ImageNet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015. 5

[31] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 2, 3, 5, 6

[32] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Advances in Neural Information Processing Systems*, pages 2234–2242, 2016. 5

[33] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 2, 3, 4, 5, 8, 11, 19, 22, 23, 25

[34] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022. 5

[35] Abhishek Sinha, Jiaming Song, Chenlin Meng, and Stefano Ermon. D2c: Diffusion-decoding models for few-shot conditional generation. *Advances in Neural Information Processing Systems*, 34:12533–12548, 2021. 2, 3

[36] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265, 2015. 2, 3

[37] Jascha Sohl-Dickstein, Eric A Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. *arXiv preprint arXiv:1503.03585*, March 2015. 2, 8

[38] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *International Conference on Learning Representations*, 2021. 2, 3, 4, 5, 6, 8, 11, 13, 14, 16, 22, 24, 28

[39] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, pages 11895–11907, 2019. 2, 3, 5

[40] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *International Conference on Learning Representations*, 2021. 2, 3, 8

[41] Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. *arXiv preprint arXiv:2203.08382*, 2022. 2, 8, 12

[42] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2149–2159, 2022. 25

[43] Daniel Watson, William Chan, Jonathan Ho, and Mohammad Norouzi. Learning fast samplers for diffusion models by differentiating through sample quality. In *International Conference on Learning Representations*, 2022. 8

[44] Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. *arXiv preprint arXiv:2203.02923*, 2022. 2

[45] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 24