Refining Generative Process with Discriminator Guidance in Score-based Diffusion Models

Dongjun Kim [*1] **Yeongmin Kim** [*1] **Se Jung Kwon** [2] **Wanmo Kang** [1] **Il-Chul Moon** [1 3]

Abstract

The proposed method, **Discriminator Guidance**, aims to improve sample generation of pre-trained diffusion models. The approach introduces a discriminator that gives explicit supervision to a denoising sample path whether it is realistic or not. Unlike GANs, our approach does not require joint training of score and discriminator networks. Instead, we train the discriminator after score training, making discriminator training stable and fast to converge. In sample generation, we add an auxiliary term to the pre-trained score to deceive the discriminator. This term corrects the model score to the data score at the optimal discriminator, which implies that the discriminator helps better score estimation in a complementary way. Using our algorithm, we achive state-of-the-art results on ImageNet 256x256 with FID 1.83 and recall 0.64, similar to the validation data's FID (1.68) and recall (0.66). We release the code at `https://github.com/alsdudrla10/DG`.

[*]Equal contribution [1]KAIST, South Korea [2]NAVER Cloud [3]Summary.AI. Correspondence to: Dongjun Kim <dongjoun57@kaist.ac.kr>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

1. Introduction

The diffusion model has recently been highlighted for its success in image generation (Dhariwal & Nichol, 2021; Ho et al., 2022a; Karras et al., 2022; Song et al., 2020b), video generation (Singer et al., 2022; Ho et al., 2022b; Voleti et al., 2022), and text-to-image generation (Rombach et al., 2022; Ramesh et al., 2022; Saharia et al., 2022). The State-Of-The-Art (SOTA) models perform human-level generation, but there is still much more room to be investigated for a deep understanding on diffusion models.

The generative model community widely uses well-trained score models (Dhariwal & Nichol, 2021; Rombach et al., 2022) in downstream tasks (Meng et al., 2021; Kawar et al., 2022; Su et al., 2022; Kim et al.). This is partially because training a new score model from scratch can be computationally expensive. However, as the demand for reusing pre-trained models increases, there are only a few research efforts that focus on improving sample quality with a pre-trained score model.

To avoid issues such as overfitting (Nichol & Dhariwal, 2021) or memorization (Carlini et al., 2023) that may arise from further score training (Figure 24), our approach keeps the pre-trained score model fixed and introduces a new component that provides a supervision during sample generation. Specifically, we propose using a discriminator as an auxiliary degree of freedom to the pre-trained model. This

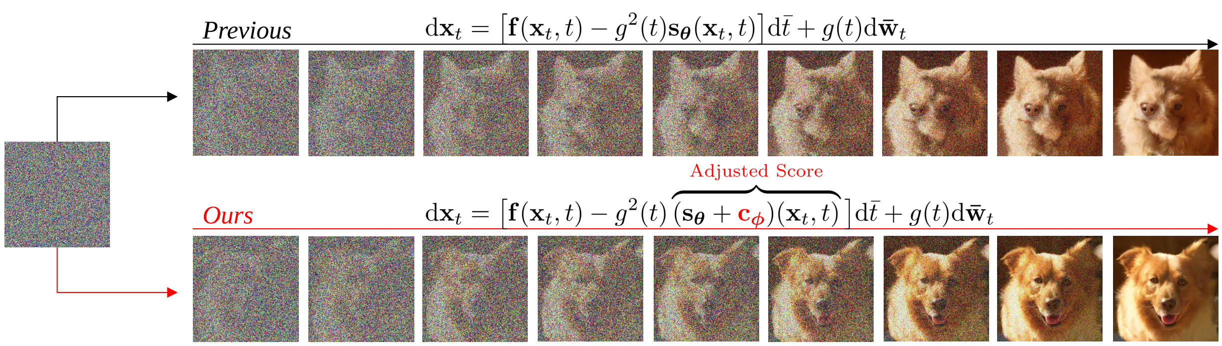
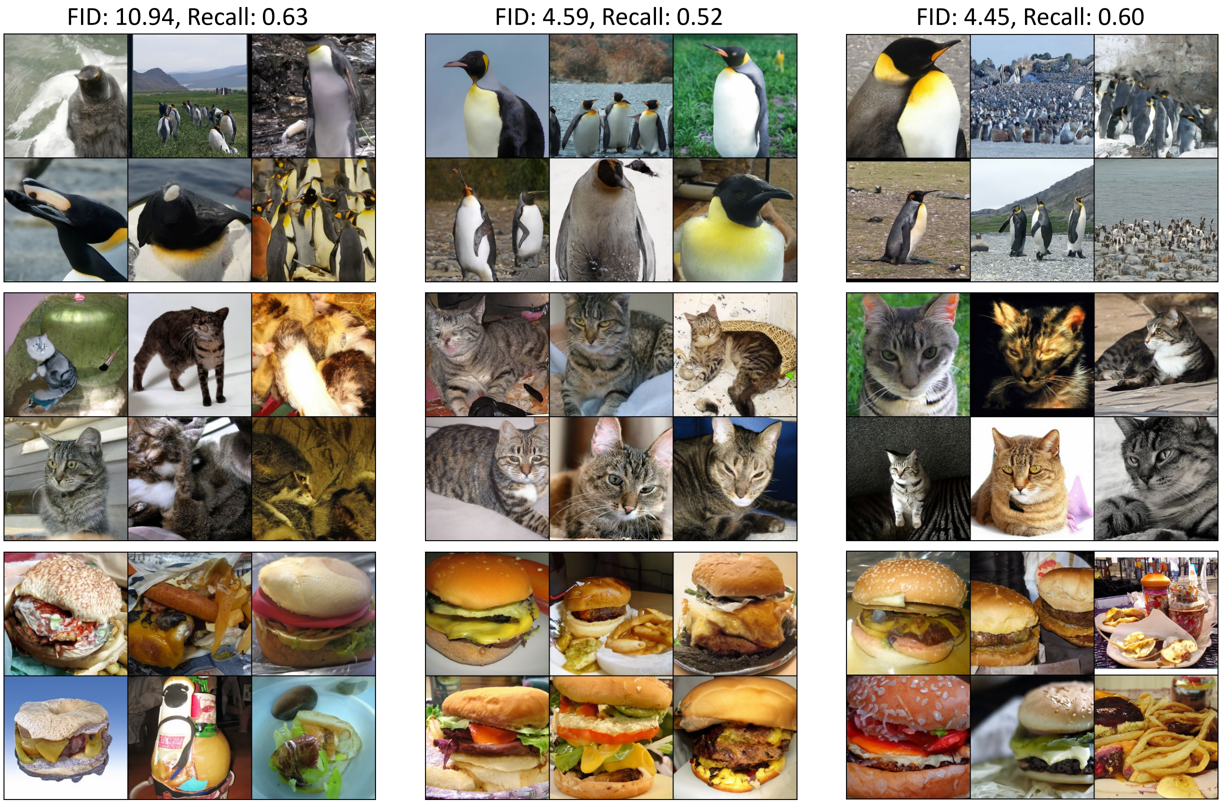


Figure 1: Comparison of the denoising processes. Discriminator Guidance adjusts the score function by estimating the gap $\mathbf{c}_{\phi}$ between the predicted model score and the true data score. As a result, the sample generated using Discriminator Guidance is indistinguishable from real data according to the discriminator.



(a) ADM (Dhariwal & Nichol, 2021) (b) ADM-G (Dhariwal & Nichol, 2021) (c) ADM-G++ (Ours)

Figure 2: Samples from ImageNet 256x256 on (a) ADM, (b) ADM with Classifier Guidance. Classifier Guidance generates high-fidelity but mode degenerated samples. (c) Classifier Guidance combined with Discriminator Guidance improves both sample quality and intra-class diversity. See Appendix D.4 for uncurated samples of SOTA models.

discriminator classifies real and generated data at all noise scales, providing direct feedback to the sample denoising process, indicating whether the sample path is realistic or not. We achieve this by adding a correction term to the model score, constructed by the discriminator, which steers the sample path towards more realistic regions (Figure 1). This term is designed to adjust the model score to match the data score at the optimal discriminator (Theorem 1), allowing our approach to find a realistic sample path by adjusting the model score. In experiments, we achieve new SOTA performances on image datasets such as CIFAR-10, CelebA/FFHQ 64x64, and ImageNet 256x256. As discriminator training is a minimization problem that is stable and fast to converge (Figure 3), such a significant gain can be achieved with a cheap budget (Table 6). We summarize the contributions as follows.

- ✓ We propose a new generative process, **Discriminator Guidance**, with an adjusted score of a given *pre-trained* score model.
- ✓ We show that the discriminator-guided samples are *closer* to the real-world data than the non-guided samples, theoretically and empirically.

2. Preliminary

Suppose $p_r(\mathbf{x}_0)$ be the data distribution and $p_{\boldsymbol{\theta}}(\mathbf{x}_0)$ be the model distribution. Likelihood-based latent variable models optimize their parameters by minimizing the upper bound of the KL divergence $D_{KL}(p_r(\mathbf{x}_0)\|p_{\boldsymbol{\theta}}(\mathbf{x}_0))$, given by

$$D_{KL}(p_r(\mathbf{x}_0)\|p_{\boldsymbol{\theta}}(\mathbf{x}_0)) \leq D_{KL}(q(\mathbf{x}_{0:T})\|p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T})),$$

where $\mathbf{x}_{1:T}$ are T latent variables; $q(\mathbf{x}_{0:T})$ is an inference distribution with marginal density $q(\mathbf{x}_0) := p_r(\mathbf{x}_0)$; and $p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T})$ is a generative distribution with marginal density $p_{\boldsymbol{\theta}}(\mathbf{x}_T) := \pi(\mathbf{x}_T)$, where π is an easy-to-sample prior distribution for generation purpose.

Denoising Diffusion Probabilistic Models (DDPM) (Ho et al., 2020) perturb the data variable $\mathbf{x}_0$ step-by-step to construct $\mathbf{x}_{1:T}$ by adding iterative Gaussian noises, leading q to be a non-parametrized fixed inference distribution with $q(\mathbf{x}_{0:T}) = p_r(\mathbf{x}_0)\prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1})$. Most (Okhotin et al., 2023) of diffusion models assume a Markov chain for the generative process so to satisfy $p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T}) = \pi(\mathbf{x}_T)\prod_{t=1}^{T} p_{\boldsymbol{\theta}}(\mathbf{x}_{t-1}|\mathbf{x}_t)$, and this modeling choice enables to optimize the surrogate objective $D_{KL}(q(\mathbf{x}_{0:T})\|p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T}))$ in a tractable way.

The continuous-time counterpart (Song et al., 2020b) of DDPM describes the diffusion process in the language of stochastic differential equations (SDE) by

$$\mathrm{d}\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}_t, \tag{1}$$

with t now being a continuum of the diffusion index in $[0, T]$, and $\mathbf{f}(\mathbf{x}_t, t)$ and $g(t)$ being the drift and the volatility coefficients, respectively. We describe our model under the continuous-time framework mainly for notational simplicity. Our model is applicable to both discrete- and continuous-time settings.

Under the continuous-time framework, the forward-time diffusion process of Eq. (1) has a unique reverse-time diffusion process (Anderson, 1982)

$$\mathrm{d}\mathbf{x}_t = \left[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\nabla \log p_r^t(\mathbf{x}_t)\right]\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t, \tag{2}$$

where $\mathrm{d}\bar{t}$ and $\bar{\mathbf{w}}_t$ are the infinitesimal reverse-time and the reverse-time Brownian motion, respectively. Subsequently, the continuous-time generative process becomes

$$\mathrm{d}\mathbf{x}_t = \left[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\right]\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

where the estimation target of the score network $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)$ is the actual data score $\nabla \log p_r^t(\mathbf{x}_t)$. Here, p_r^t is the diffused probability density of the data distribution following the forward-time diffusion process in Eq. (1).

The continuous-time model trains the score network with the denoising score matching loss (Song & Ermon, 2019)

$$\mathcal{L}_{\boldsymbol{\theta}} = \frac{1}{2}\int_0^T \xi(t)\mathbb{E}\left[\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla \log p_{0t}(\mathbf{x}_t|\mathbf{x}_0)\|_2^2\right]\mathrm{d}t,$$

where ξ is the temporal weight and p_{0t} is the transition probability from $\mathbf{x}_0$ to $\mathbf{x}_t$. This denoising score objective coincides to the joint KL divergence $D_{KL}(q(\mathbf{x}_{0:T})\|p_{\boldsymbol{\theta}}(\mathbf{x}_{0:T}))$ if $\xi(t) = g^2(t)$ (Chen et al., 2016; Song et al., 2021). Also, under different weighting functions, this objective could be equivalently interpreted as the noise matching loss $\int_0^T \mathbb{E}[\|\boldsymbol{\epsilon}_{\boldsymbol{\theta}} - \boldsymbol{\epsilon}\|_2^2]$ (Ho et al., 2020) or the data reconstruction loss $\int_0^T \mathbb{E}[\|\hat{\mathbf{x}}_{\boldsymbol{\theta}}(\mathbf{x}_t) - \mathbf{x}_0\|_2^2]$ (Kingma et al., 2021).

There are various approaches to enhance the precision of score training. For instance, Kim et al. (2022b); Kingma & Gao (2023); Hang et al. (2023) have proposed updating the score network using Maximum Perturbed Likelihood Estimation to improve large-time denoising accuracy. Conversely, Lai et al. (2022); Daras et al. (2023) have studied the invariant characteristics of the data diffusion process and recommended adding an extra regularization term to the denoising score loss to meet these invariant properties. Our work, on the other hand, aims to refine the fixed model score with noise contrastive estimation, which is distinct from prior attempts to improve score accuracy.

Algorithm 1 Discriminator Training

1: Construct $\mathcal{D} = \{\mathbf{x}_1, ..., \mathbf{x}_M\}$ from the real-world
2: Construct $\mathcal{G} = \{\hat{\mathbf{x}}_1, ..., \hat{\mathbf{x}}_N\}$ by sampling from $p_{\boldsymbol{\theta}}$
3: **while** converged **do**
4: Sample $\mathbf{x}_1, ..., \mathbf{x}_{B/2}$ from the real dataset $\mathcal{D}$
5: Sample $\mathbf{x}_{B/2+1}, ..., \mathbf{x}_B$ from the sample dataset $\mathcal{G}$
6: Sample $t_1, ..., t_B$ from $[0, T]$
7: Diffuse $\mathbf{x}_i^{t_i} \leftarrow e^{-\int_0^{t_i}\beta_s\,\mathrm{d}s}\mathbf{x}_i + \sqrt{1 - e^{-\int_0^{t_i}\beta_s\,\mathrm{d}s}}\boldsymbol{\epsilon}_i$ for $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, \mathbf{I}), \forall i = 1, ..., B$
8: Calculate $\hat{\mathcal{L}}_{\boldsymbol{\phi}} \leftarrow -\sum_{i=1}^{B/2}\lambda(t_i)\log d_{\boldsymbol{\phi}}(\mathbf{x}_i^{t_i}, t_i) - \sum_{i=B/2+1}^{B}\lambda(t_i)\log\left(1 - d_{\boldsymbol{\phi}}(\mathbf{x}_i^{t_i}, t_i)\right)$
9: Update $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \frac{\partial \hat{\mathcal{L}}_{\boldsymbol{\phi}}}{\partial \boldsymbol{\phi}}$
10: **end while**

3. Refining Generative Process with Discriminator Guidance

3.1. Correction of Pre-trained Model Score

After score training, we synthesize samples with the time-reversal generative process

$$\mathrm{d}\mathbf{x}_t = \left[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t)\right]\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t, \tag{3}$$

where $\mathbf{s}_{\boldsymbol{\theta}_\infty}$ represents the score network after the convergence. This generative process could differ from the reverse-time data process if the local optimum $\boldsymbol{\theta}_\infty$ deviates from the global optimum $\boldsymbol{\theta}_*$. We show in Theorem 1 that the generative process of Eq. (3) coincides with the data process of Eq. (2) if we adjust the model score. We call this gap by the *correction term*, which is *nonzero* as long as $\boldsymbol{\theta}_\infty \neq \boldsymbol{\theta}_*$.

Theorem 1. *Suppose $p_{\boldsymbol{\theta}_\infty}$ be the solution of the time-reversal generative process of Eq. (3). Let p_r^t and $p_{\boldsymbol{\theta}_\infty}^t$ be the marginal densities (at t) of the forward-time SDE $\mathrm{d}\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}_t$ starting from p_r and $p_{\boldsymbol{\theta}_\infty}$, respectively. If $\mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}, T) = \nabla \log \pi(\mathbf{x})$, where π is the prior distribution, and the log-likelihood $\log p_{\boldsymbol{\theta}_\infty}$ equals its evidence lower bound $\mathcal{L}_{\boldsymbol{\theta}_\infty}$, then the reverse-time SDE*

$$\mathrm{d}\mathbf{x}_t = \left[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\nabla \log p_r^t(\mathbf{x}_t)\right]\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

coincides with a diffusion process with adjusted score,

$$\mathrm{d}\mathbf{x}_t = \left[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + \mathbf{c}_{\boldsymbol{\theta}_\infty})(\mathbf{x}_t, t)\right]\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

for $\mathbf{c}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t) := \nabla \log \frac{p_r^t(\mathbf{x}_t)}{p_{\boldsymbol{\theta}_\infty}^t(\mathbf{x}_t)}$.

3.2. Discriminator Guidance

The correction term $\mathbf{c}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t) = \nabla \log \frac{p_r^t(\mathbf{x}_t)}{p_{\boldsymbol{\theta}_\infty}^t(\mathbf{x}_t)}$ is intractable in general because the likelihood ratio $\frac{p_r^t}{p_{\boldsymbol{\theta}_\infty}^t}$ is inaccessible. Therefore, we estimate this likelihood ratio by

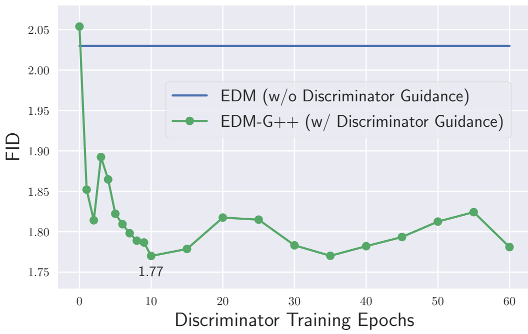


Figure 3: Discriminator guidance refines FID on CIFAR-10.

training a discriminator at all noise level t. For discriminator training, we first draw fake samples from the generative process of Eq. (3) as many as data instances. Then we classify the real and fake data using the noise-embedded Binary Cross Entropy (BCE)

$$\mathcal{L}_{\boldsymbol{\phi}} = \int \lambda(t)\big(\mathbb{E}_{p_r^t(\mathbf{x}_t)}[-\log d_{\boldsymbol{\phi}}(\mathbf{x}_t,t)] + \mathbb{E}_{p_{\boldsymbol{\theta}_\infty}^t(\mathbf{x}_t)}[-\log(1-d_{\boldsymbol{\phi}}(\mathbf{x}_t,t))]\big)\,\mathrm{d}t, \quad (4)$$

where λ is the temporal weight, see Algorithm 1 and Appendix A.4 for details.

As the correction term is represented by

$$\mathbf{c}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t,t) = \nabla \log \frac{d_{\boldsymbol{\phi}_*}(\mathbf{x}_t,t)}{1-d_{\boldsymbol{\phi}_*}(\mathbf{x}_t,t)},$$

in terms of the optimal discriminator $d_{\boldsymbol{\phi}_*}$ of $\mathcal{L}_{\boldsymbol{\phi}}$, we estimate the correction term $\mathbf{c}_{\boldsymbol{\theta}_\infty}$ with a neural discriminator $d_{\boldsymbol{\phi}}$ by

$$\mathbf{c}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t,t) \approx \mathbf{c}_{\boldsymbol{\phi}}(\mathbf{x}_t,t) := \nabla \log \frac{d_{\boldsymbol{\phi}}(\mathbf{x}_t,t)}{1-d_{\boldsymbol{\phi}}(\mathbf{x}_t,t)}.$$

With the above tractable correction estimate, we define the **Discriminator Guidance** (DG) by

$$\mathrm{d}\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + \mathbf{c}_{\boldsymbol{\phi}})(\mathbf{x}_t,t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t. \quad (5)$$

Figure 3 shows that the discriminator indeed improves sample quality with a quick convergence.

3.3. Theoretical Analysis

Although we introduced Discriminator Guidance in the context of differential equations, this section examines the approach from the perspective of statistical divergence between the data and the sample distributions. Specifically, we define $p_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$ as the discriminator-guided sample distribution of Eq. (5). The central question becomes

Is $p_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$ closer to the data distribution p_r than $p_{\boldsymbol{\theta}_\infty}$?

We answer the question in Theorem 2.

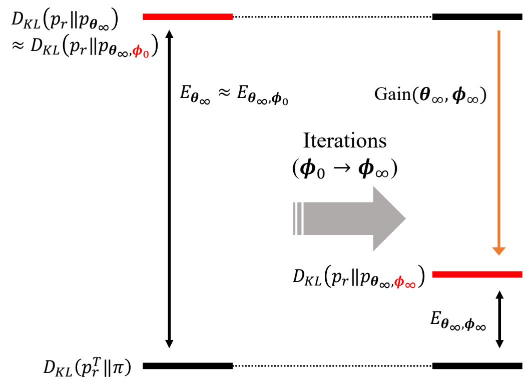


Figure 4: Schematic illustration of the analysis in Section 3.3. The gain increases as discriminator learns.

Table 1: Discriminator-adjusted score error $E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$ and corresponding Gain.

Discriminator	$E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$	Gain
Blind $d_{\boldsymbol{\phi}_b}$ ($\equiv 0.5$)	$E_{\boldsymbol{\theta}_\infty}$	0
Optimal $d_{\boldsymbol{\phi}_*}$	0	$E_{\boldsymbol{\theta}_\infty}$ (Maximum)
Untrained $d_{\boldsymbol{\phi}_0}$ (≈ 0.5)	$\approx E_{\boldsymbol{\theta}_\infty}$	≈ 0
Trained $d_{\boldsymbol{\phi}_\infty}$	$\ll E_{\boldsymbol{\theta}_\infty}$	$\nearrow E_{\boldsymbol{\theta}_\infty}$

Theorem 2. *If the assumptions of Theorem 1 hold, then*

$$D_{KL}(p_r\|p_{\boldsymbol{\theta}_\infty}) = D_{KL}(p_r^T\|\pi) + E_{\boldsymbol{\theta}_\infty},$$
$$D_{KL}(p_r\|p_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}) \le D_{KL}(p_r^T\|\pi) + E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}},$$

where $E_{\boldsymbol{\theta}_\infty}$ is the score error

$$E_{\boldsymbol{\theta}_\infty} = \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\big[\|\nabla \log p_r^t - \mathbf{s}_{\boldsymbol{\theta}_\infty}\|_2^2\big]\,\mathrm{d}t,$$

and $E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$ is the discriminator-adjusted score error

$$E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}} = \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\big[\|\nabla \log p_r^t - (\mathbf{s}_{\boldsymbol{\theta}_\infty} + \mathbf{c}_{\boldsymbol{\phi}})\|_2^2\big]\,\mathrm{d}t$$
$$= \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\big[\|\mathbf{c}_{\boldsymbol{\theta}_\infty} - \mathbf{c}_{\boldsymbol{\phi}}\|_2^2\big]\,\mathrm{d}t.$$

To measure the effect of discriminator training, we use Theorem 2 to compute the gain by subtracting two KLs,

$$D_{KL}(p_r\|p_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}) \le D_{KL}(p_r\|p_{\boldsymbol{\theta}_\infty}) - \mathrm{Gain}(\boldsymbol{\theta}_\infty,\boldsymbol{\phi}),$$

where $\mathrm{Gain}(\boldsymbol{\theta}_\infty,\boldsymbol{\phi}) = E_{\boldsymbol{\theta}_\infty} - E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$ represents the difference between the score error and the discriminator-adjusted score error. Note that while Theorem 2 does not guarantee that the gain is strictly positive, it is initialized near zero and gradually increases throughout discriminator training, as summarized in Table 1. Specifically, when the discriminator is completely blind ($d_{\boldsymbol{\phi}_b} \equiv 0.5$), there is no signal from the discriminator gradient, and the discriminator-adjusted

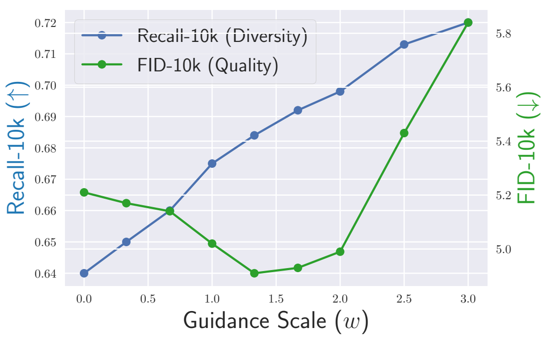


Figure 5: DG improves sample quality and diversity.

score error $E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}_b}$ equals the score error $E_{\boldsymbol{\theta}_\infty}$. Therefore, the gain is approximately zero when the discriminator is untrained ($d_{\boldsymbol{\phi}_0} \approx 0.5$) as shown in Figure 3. On the other hand, at the optimal discriminator $d_{\boldsymbol{\phi}_*}$, the neural correction $\mathbf{c}_{\boldsymbol{\phi}_*}$ matches the target correction $\mathbf{c}_{\boldsymbol{\theta}_\infty}$ and satisfies $E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}_*} = 0$, allowing Gain to be maximized as discriminator parameters are updated. See Figure 4 for a schematic visualization.

In other words, we can interpret that Discriminator Guidance introduces an additional axial degree of freedom $\boldsymbol{\phi}$ that reparametrizes the score error $E_{\boldsymbol{\theta}_\infty}$ into a discriminator-adjusted score error $E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$. As a result, the score error $E_{\boldsymbol{\theta}_\infty}$ is no longer optimized with the denoising score loss $\mathcal{L}_{\boldsymbol{\theta}}$, but the reparametrized error $E_{\boldsymbol{\theta}_\infty,\boldsymbol{\phi}}$ can be further optimized with an alternative loss $\mathcal{L}_{\boldsymbol{\phi}}$ of Eq. (4).

3.4. Optimality Analysis

Let's take a closer look at the score component. Our discriminator training is stable because we keep a pre-trained score model fixed during training, unlike unstable GAN training. Hence, after the discriminator has reached its optimal point, the resulting adjusted model score is given by

$$\begin{aligned}
&\mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t,t) + w\mathbf{c}_{\boldsymbol{\phi}_*}(\mathbf{x}_t,t) \\
&\quad = \nabla \log p_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t) + w\nabla \log \frac{p_r^t(\mathbf{x}_t)}{p_{\boldsymbol{\theta}_\infty}^t(\mathbf{x}_t)} \\
&\quad = \nabla \log \left[(p_r^t(\mathbf{x}_t))^w (p_{\boldsymbol{\theta}_\infty}^t(\mathbf{x}_t))^{1-w}\right],
\end{aligned}$$

Therefore, the sample distribution $(p_r^t)^w(p_{\boldsymbol{\theta}_\infty}^t)^{1-w}$ balances data distribution and non-guided distribution. The argument also holds for the conditional case, leading DG as a controller for the intra-class diversity. Figure 5 experiments on ImageNet, demonstrating that there is a sweet spot of DG weight regarding both quality (FID) and diversity (recall).

3.5. Connection with Classifier Guidance

Classifier Guidance (CG) (Dhariwal & Nichol, 2021) is a milestone technique to guide a sample with a pre-trained classifier $p_{\psi_\infty}(c|\mathbf{x}_t,t)$. The classifier-guided generative process is $\mathrm{d}\mathbf{x}_t = [\mathbf{f}(\mathbf{x}_t,t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t,t) + \nabla \log p_{\boldsymbol{\psi}_\infty}(y|\mathbf{x}_t,t))]\,\mathrm{d}\bar{t} + g_t\,\mathrm{d}\bar{\mathbf{w}}_t$. This is equivalent to sampling from the joint distribution of $(\mathbf{x}_t, y)$ because

$$\begin{aligned}
\nabla \log p_r^t(\mathbf{x}_t, y) &= \nabla \log p_r^t(\mathbf{x}_t) + \nabla \log p(y|\mathbf{x}_t,t) \\
&\approx \mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t,t) + \nabla \log p_{\boldsymbol{\psi}_\infty}(y|\mathbf{x}_t,t),
\end{aligned}$$

where $p(y|\mathbf{x}_t,t)$ is the oracle classifier at t. Classifier Guidance provides supervision information on a sample path, evaluating whether the sample is correctly classified by the class label y, or not. However, using Classifier Guidance may lead to mode collapse as it maximizes the classifier probability $p(y|\mathbf{x}_t,t)$. In contrast, Discriminator Guidance offers enhanced mode coverage, as elaborated in Section 3.4, by providing distinctive supervision information on whether a sample path is realistic or not.

As sampling from the joint distribution of $(\mathbf{x}_t, y)$ requires accurate score estimation, Discriminator Guidance and Classifier Guidance can be combined for a synergistic effect. We suggest the combination of guidance techniques by

$$\begin{aligned}
\mathrm{d}\mathbf{x}_t = \Big[&\mathbf{f}(\mathbf{x}_t,t) - g^2(t)\big((\mathbf{s}_{\boldsymbol{\theta}_\infty} + w_t^{DG}\mathbf{c}_{\boldsymbol{\phi}_\infty})(\mathbf{x}_t,t) \\
&+ w_t^{CG}\nabla \log p_{\boldsymbol{\psi}_\infty}(c|\mathbf{x}_t,t)\big)\Big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,
\end{aligned} \tag{6}$$

where w_t^{DG} and w_t^{CG} are the time-dependent weights, respectively. The two pieces of information could ideally guide the sample toward the common likely region of classifier and discriminator in a complementary way.

Table 2: Algorithm 2 includes DDPM, DDIM, and EDM samplers.

Sampler	γ_t	η	w_t^{CG}	w_t^{DG}
DDPM	0	1	0	0
DDIM	0	0	0	0
EDM	≥0	0	0	0
ADM-G	0	1	>0	0
EDM-G++	≥0	0	0	>0
ADM-G++	0	1	>0	>0

Algorithm 2 describes the full details of our sampling procedure for Eq. (6). The algorithm reduces the samplers of

Algorithm 2 Sampling with Guidance Techniques

Sample $\mathbf{x}_T \sim \mathcal{N}(0, \sigma_{max}^2\mathbf{I})$
for $i = N$ to 1 **do**
  Sample $\boldsymbol{\epsilon}_i \sim \mathcal{N}(0, S_1^2\mathbf{I})$ and $\boldsymbol{\epsilon}_i' \sim \mathcal{N}(0,\mathbf{I})$
  $\hat{t}_i \leftarrow \sigma^{-1}((1+\gamma_{t_i})\sigma(t_i))$ (Karras et al., 2022)
  $\mathbf{x}_{\hat{t}_i} \leftarrow \mathbf{x}_{t_i} + \sqrt{\sigma^2(\hat{t}_i) - \sigma^2(t_i)}\,\boldsymbol{\epsilon}_{t_i}$
  $\mathbf{s}_{\hat{t}_i} \leftarrow \mathbf{f}(\mathbf{x}_{\hat{t}_i},\hat{t}_i) - \frac{1+\eta^2}{2} g_{\hat{t}_i}^2 \mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_{\hat{t}_i},\hat{t}_i)$
  $\mathbf{c}_{\hat{t}_i} \leftarrow -\frac{1+\eta^2}{2} g_{\hat{t}_i}^2 \nabla \log \frac{d_{\boldsymbol{\phi}_\infty}(\nu_{\tau_{\hat{t}_i}}\mathbf{x}_{\hat{t}_i},\tau_{\hat{t}_i})}{1-d_{\boldsymbol{\phi}_\infty}(\nu_{\tau_{\hat{t}_i}}\mathbf{x}_{\hat{t}_i},\tau_{\hat{t}_i})}$ (Eq. (8))
  $\mathbf{g}_{\hat{t}_i} \leftarrow -\frac{1+\eta^2}{2} g_{\hat{t}_i}^2 \nabla \log p_{\boldsymbol{\psi}_\infty}(y|\mathbf{x}_{\hat{t}_i},\hat{t}_i)$
  $\mathbf{x}_{t_{i-1}} \leftarrow \mathbf{x}_{\hat{t}_i} + (t_{i-1} - \hat{t}_i)(\mathbf{s}_{\hat{t}_i} + w_{\hat{t}_i}^{DG}\mathbf{c}_{\hat{t}_i} + w_{\hat{t}_i}^{CG}\mathbf{g}_{\hat{t}_i})$
  $\mathbf{x}_{t_{i-1}} \leftarrow \mathbf{x}_{t_{i-1}} + \eta g_{\hat{t}_i}\sqrt{t_{i-1} - \hat{t}_i}\,\boldsymbol{\epsilon}_i'$
end for

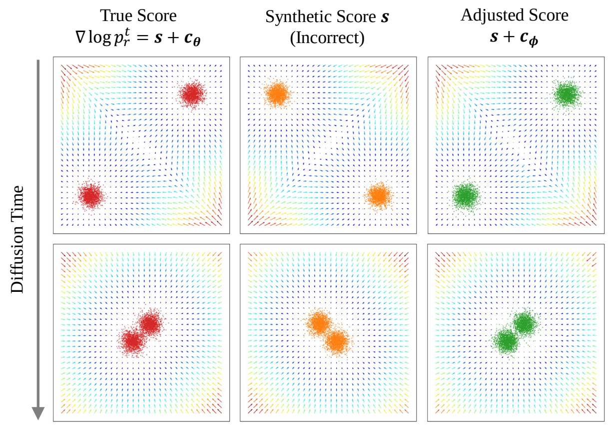


Figure 6: Comparison of the true data score, synthetic score, and adjusted score. We assume $p_r(\mathbf{x}) := \frac{1}{2}\mathcal{N}(\mathbf{x}; (10, 10)^T, \mathbf{I}) + \frac{1}{2}\mathcal{N}(\mathbf{x}; (-10, -10)^T, \mathbf{I})$ and $p_g(\mathbf{x}) := \frac{1}{2}\mathcal{N}(\mathbf{x}; (10, -10)^T, \mathbf{I}) + \frac{1}{2}\mathcal{N}(\mathbf{x}; (-10, 10)^T, \mathbf{I})$ with $\mathbf{s}(\mathbf{x}_t, t) := \nabla \log p_g^t(\mathbf{x}_t)$. We solve the probability-flow ODE (Song et al., 2020b) to visualize samples from each score function.

DDPM (Ho et al., 2020; Dhariwal & Nichol, 2021), DDIM (Song et al., 2020a), and EDM (Karras et al., 2022) with corresponding hyperparameters in Table 2. Our sampler is denoted as the postfix with **G++** upon a basic sampler by a prefix. See Appendix D.1 for detailed sampling procedure.

4. Related Works

A line of research merges diffusion models with GAN models. Zheng et al. (2022b); Lyu et al. (2022) synthesize the diffused data $\mathbf{x}_{\sigma_{mid}}$ with a GAN generator (by putting $\mathbf{x}_T$ as generator's input), and denoise $\mathbf{x}_{\sigma_{mid}}$ to $\mathbf{x}_0$ with a diffusion model. Xiao et al. (2022) exchange thousands of denoising steps with a small number of sequential conditional GAN generators. Wang et al. (2022) utilize the diffusion concept to train GAN. On the contrary, Jolicoeur-Martineau et al. (2021) train the diffusion model with an adversarial loss.

Noise Contrastive Estimation (NCE) (Gutmann & Hyvärinen, 2010) has been widely used in generative modeling to improve sample quality. NCE employs supervised learning to estimate the likelihood ratio between the target distribution and the noise distribution. To ensure accurate likelihood ratio estimation, it is crucial to assume that the noise distribution is close to the target distribution. For instance, in GAN literature (Azadi et al., 2019; Che et al., 2020), the target distribution is the data distribution, and the noise distribution is the sample distribution. In VAE context (Aneja et al., 2021; Bauer & Mnih, 2019), the target distribution is the aggregate posterior, and the noise distribution is the prior distribution. Similarly, we apply NCE to diffusion models with the time-incorporated target (data) distribution and noise (sample) distribution. Our approach is particularly successful because the sample distribution of diffusion models is highly similar to the data distribution.

Table 3: Performance on CIFAR-10.

Model	Diffusion Space	NFE↓	Unconditional NLL↓	Unconditional FID↓	Conditional FID↓
VDM (Kingma et al., 2021)	Data	1000	**2.49**	7.41	-
DDPM (Ho et al., 2020)	Data	1000	3.75	3.17	-
iDDPM (Nichol & Dhariwal, 2021)	Data	1000	3.37	2.90	-
Soft Truncation (Kim et al., 2022b)	Data	2000	2.91	2.47	-
INDM (Kim et al., 2022a)	Latent	2000	3.09	2.28	-
CLD-SGM (Dockhorn et al., 2022)	Data	312	3.31	2.25	-
NCSN++ (Song et al., 2020b)	Data	2000	3.45	2.20	-
LSGM (Vahdat et al., 2021)	Latent	138	3.43	2.10	-
NCSN++-G (Chao et al., 2022)	Data	2000	-	-	2.25
EDM†(random seed)	Data	39	2.60	2.03	1.82
EDM (reported, manual seed)	Data	**35**	2.60	1.97	1.79
LSGM-G++	Latent	138	3.42	1.94	-
EDM-G++	Data	**35**	2.55	**1.77**	**1.64**

† We recalculate FID of EDM (Karras et al., 2022) under a random seed for a fair comparison with previous research. We report our performances under the random seed by default.

Table 4: Performance on CelebA/FFHQ 64x64.

Model	NFE↓	CelebA	FFHQ
DDPM++ (Song et al., 2020b)	131	2.32	-
Soft Truncation (Kim et al., 2022b)	131	1.90	-
Soft Diffusion (Daras et al., 2022)	300	1.85	-
INDM (Kim et al., 2022a)	132	1.75	-
Diffusion StyleGAN2 (Wang et al., 2022)	**1**	1.69	-
EDM (Karras et al., 2022)	79	-	2.39
Soft Truncation-G++	131	**1.34**	-
EDM-G++	71	-	**1.98**

5. Experiments

5.1. A Toy 2-dimensional Case

Figure 6 shows the experimental result of a tractable 2-dimensional toy case. We train a 4-layered MLP discriminator with 256 neurons until convergence, and we hypothesize an incorrect score function $\mathbf{s} := \nabla \log p_g^t$ (for a synthetic generative distribution p_g) that misfits to the data score. If there is no guidance, the incorrect score $\mathbf{s}$ will generate samples from a wrong distribution as in Figure 6. On the contrary, $\mathbf{s} + \mathbf{c}_{\phi_\infty}$ successfully guides $\mathbf{s}$ to $\nabla \log p_r^t$, see Figure 15 for additional visualization.

5.2. Image Generation

We experiment on CIFAR-10, CelebA/FFHQ 64x64, and ImageNet 256x256. We use the pre-trained networks on CIFAR-10 and FFHQ from Karras et al. (2022); Vahdat et al. (2021), CelebA from Kim et al. (2022b) and ImageNet from Dhariwal & Nichol (2021); Peebles & Xie (2022).

Discriminator Network. We use the encoder of U-Net structure[1] as our discriminator network. For diffusion models on data space, we attach two noise-embedded U-Net en-

[1]We tested MLP, ResNet18, and a transformer, but the U-Net performed the best for Discriminator Guidance.

Table 5: Performance on ImageNet 256x256.

Model	Diffusion Space	FID↓	sFID↓	IS↑	Prec↑	Rec↑	F1↑
Validation Data		1.68	3.67	232.21	0.75	0.66	0.70
ADM (Dhariwal & Nichol, 2021)	Data	10.94	6.02	100.98	0.69	0.63	0.66
DiT-XL/2 (Peebles & Xie, 2022)	Latent	9.62	6.85	121.50	0.67	0.67	0.67
ADM-G (Dhariwal & Nichol, 2021)	Data	4.59	5.25	186.70	0.82	0.52	0.64
RIN (Jabri et al., 2022)	Data	4.51	-	161.00	-	-	-
LDM-4-G (Rombach et al., 2022)	Latent	3.60	-	247.67	**0.87**	0.48	0.62
RIN + schedule (Chen, 2023)	Data	3.52	-	186.20	-	-	-
simple diffusion (Hoogeboom et al., 2023)	Data	2.77	-	211.80	-	-	-
DiT-XL/2-G (Peebles & Xie, 2022)	Latent	2.27	4.60	278.24	0.83	0.57	0.68
ADM-G++	Data	3.18	**4.53**	255.74	0.84	0.53	0.66
DiT-XL/2-G++	Latent	**1.83**	5.16	**281.53**	0.78	**0.64**	**0.70**

Table 6: Component-wise computational budget.

Dataset (Model)	Score Training	Sample Generation	Discriminator Training
CIFAR-10 (EDM)	200M $\mathbf{s}_{\boldsymbol{\theta}_\infty}$ (480hr)	1.75M $\mathbf{s}_{\boldsymbol{\theta}_\infty}$ (1hr)	1M $d_{\boldsymbol{\phi}}$ (10min)
ImageNet 256x256 (ADM)	2.5T $\mathbf{s}_{\boldsymbol{\theta}_\infty}$	100M $\mathbf{s}_{\boldsymbol{\theta}_\infty}$	25.6M $d_{\boldsymbol{\phi}}$

coders: the pre-trained ADM classifier (Dhariwal & Nichol, 2021) and an auxiliary (shallow) U-Net encoder. We put $(\mathbf{x}_t, t)$ to the ADM classifier, and we extract the latent $\mathbf{z}_t$ of $\mathbf{x}_t$ from the last pooling layer of the pre-trained classifier. Then, we put $(\mathbf{z}_t, t)$ to the auxiliary U-Net encoder and predict real/fake by its output. We freeze the ADM classifier, and we only fine-tune shallow U-Net encoder as default. Not to mention that fine-tuning save the training cost, fine-tuning performs better or equivalent to training the entire architecture (Kato & Teshima, 2021). For LSGM-G++, we train the U-Net encoder from scratch. For DiT-XL-G++, we train the latent classifier with the same architecture of the ADM classifier except the input dimension, and fine-tuning the shallow U-Net encoder for discriminator. We train a class-conditional discriminator for the class-conditional generation. See Table 8 for detailed training configuration.

Quantitative Analysis. We achieve new SOTA FIDs on all datasets including CIFAR-10, CelebA, FFHQ, and ImageNet. On CIFAR-10, Table 3 shows that Discriminator Guidance is effective in both data diffusion (EDM) and latent diffusion (LSGM) models. Other than Discriminator Guidance, we use all the hyperparameters of EDM and LSGM, so the performance gain purely comes from the discriminator componenet. The gain of Discriminator Guidance is also notable on human-face datasets in Table 4.

On ImageNet 256x256, we present SOTA results in various metrics, including FID, sFID, IS, and recall in Table 5. For reference, we also measure these metrics on the ImageNet 50k validation data. Notably, IS and precision of the validation data are on par with the best models. Thus, optimizing other metrics, such as FID, sFID, and recall, becomes more important once IS and precision of a model reach the level of the validation data. Our experiments show that with Discriminator Guidance, we achieve the strongest performances with respect to FID and recall in DiT-XL/2-G++, indicating that the sample quality and diversity are significantly improved. See Table 8 for detailed hyperparameters and Appendix D.4 for uncurated samples.

(a) Original Data (b) EDM (c) EDM-G++

Figure 7: Comparison of (a) original sample, and the average of 100 regenerated samples of (b) EDM and (c) EDM-G++ on FFHQ. The regeneration FID is 1.25 (w/o DG) and 1.09 (w/ DG) if we perturb data by the standard Gaussian.

In terms of carbon footprint, Discriminator Guidance requires additional sampling from the pre-trained score model plus discriminator training. Table 6 summarizes the component-wise neural network evaluation budget. In both CIFAR-10 and ImageNet experiments, Discriminator Guidance requires a lightweight burden compared to score training. Furthermore, we measure the actual elapsed time in GPU (A100) hours, including backpropagation time.

Qualitative Analysis. Figure 7 illustrates a comparison between the original sample and the averages of its regenerated samples (Meng et al., 2021) on FFHQ. If the score estimation is correct, the average reconstructed image should be approximately equal to the original image when the perturbation noise is small enough. To explain this, suppose the original image is $\mathbf{y}$ and we perturb it with a fixed direction $\boldsymbol{\epsilon}$ by $\mathbf{y} + \sigma(t)\boldsymbol{\epsilon}$. Then, by putting this perturbed data into the Tweedie's formula (Robbins, 1992), the average reconstruction is $\mathbb{E}[\mathbf{x}_0|\mathbf{x}_t = \mathbf{y} + \sigma(t)\boldsymbol{\epsilon}] = \mathbf{y} + \sigma(t)\boldsymbol{\epsilon} + \sigma^2(t)\nabla \log p_r^t\big(\mathbf{y} + \sigma(t)\boldsymbol{\epsilon}\big)$. If $\sigma(t)$ is sufficiently small, we obtain $p_r^t(\mathbf{y} + \sigma(t)\boldsymbol{\epsilon}) \underset{\sim}{\propto} p(\boldsymbol{\epsilon})$, which leads to $\mathbb{E}_{\boldsymbol{\epsilon}}[\nabla \log p_r^t(\mathbf{y} + \sigma(t)\boldsymbol{\epsilon})] \approx 0$. Therefore, the average reconstructed image of $\mathbf{y}$ with a random direction $\boldsymbol{\epsilon}$ would approximately be the original data by $\mathbb{E}_{\boldsymbol{\epsilon}}\big[\mathbb{E}[\mathbf{x}_0|\mathbf{x}_t = \mathbf{y}+\sigma(t)\boldsymbol{\epsilon}]\big] \approx \mathbf{y}$. In conclusion, the closeness of the average reconstructed image to the original one indirectly diagnose whether the estimated score is accurate because the above argument holds for data score. Figure 7 suggests that the adjusted score provides a more accurate estimation compared to the original model score.

Figure 8 shows that the trained discriminator is able to accurately distinguish between the diffusion path of real data and the denoising path of the generated sample. In contrast, the discriminator-adjusted denoising path deceives the discriminator, resulting in the likelihood ratio curve of

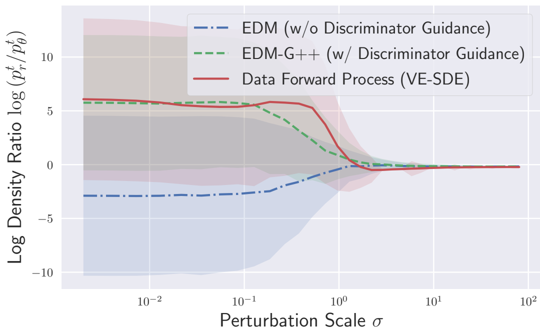


Figure 8: Comparison of sample trajectories with respect to the likelihood ratio on FFHQ.

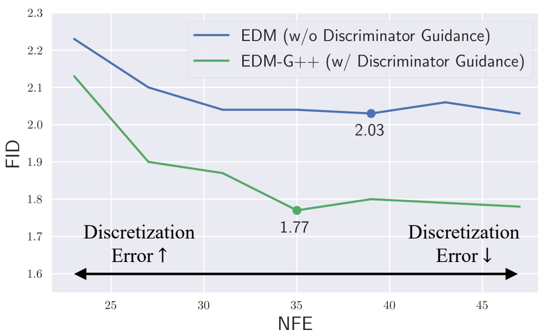


Figure 9: Ablation study for NFE on CIFAR-10.

the adjusted generative SDE (Eq. 5) closely approximating that of the data forward SDE.

Figure 9 illustrates the effect of Discriminator Guidance with respect to the sampling NFEs on CIFAR-10. As NFE decreases, the discretization error dominates the sampling error (De Bortoli, 2022), and the gain from Discriminator Guidance becomes suboptimal. We leave it as a future work to fit Discriminator Guidance on samplers with extremely small NFEs. See Appendix D.3 for more ablation studies.

Figure 10 shows the precision and recall curve by discriminator training. At the zero-th epoch, before discriminator training, we observe that precision/recall for vanilla DiT-XL-G are higher/lower than those of the validation data, respectively. This is because Classifier Guidance generates overconfident samples in terms of the classifier. However, Discriminator Guidance significantly mitigates this precision-recall trade-off of Classifier Guidance.

Figure 11 displays the normalized cumulative objective loss by noise scale. To ensure a fair comparison, we utilize the same weighting function $\xi(t) = \lambda(t)$ to evaluate the discriminator loss $\mathcal{L}_{\boldsymbol{\phi}}$ and the score loss $\mathcal{L}_{\boldsymbol{\theta}}$. The results reveal that the discriminator can capture the estimation error, particularly at a large diffusion time that determines the sample diversity. This finding highlights the potential of Discriminator Guidance as a supplementary approach to

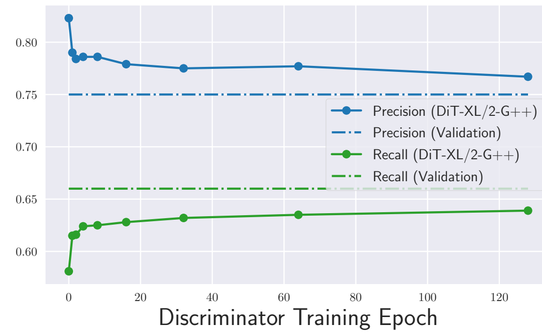


Figure 10: Precision/recall by discriminator training.

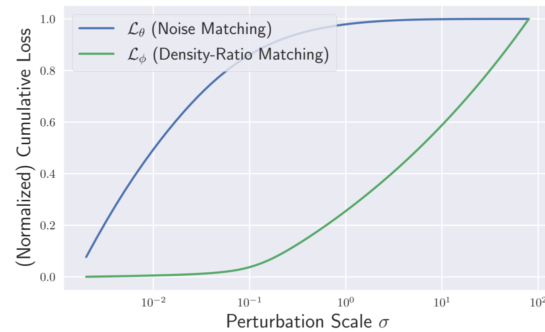


Figure 11: Loss contribution by noise scale.

address the problem of poor estimation at large time (Kim et al., 2022b) in the score matching framework.

5.3. Image-2-Image Translation

Discriminator Guidance could be applied to the Image-2-Image (I2I) translation task. I2I (Meng et al., 2021) denoise a perturbed source image with a score network trained on the target domain. For discriminator training, we first translate source training images, and then we aggregate these translated images with source images to a fake dataset $\mathcal{G}$. By specifying $\mathcal{D}$ as target images, we follow Algorithm 1 to train the discriminator. Discriminator Guidance avoids going neither the *source* domain nor the *translated* domain, and it leads to the *target* domain. Empirically, the curve in Figure 12-(b) shows that our approach remedies the trade-off between *realism* (to target) and *faithfulness* (with source). We leave Appendix C for details.

6. Discussion

This section discusses two possible avenues for the future development of Discriminator Guidance. The first direction involves rewriting our approach in terms of Bregman divergence, while the second direction explores the simultaneous training of score and discriminator networks. In the first direction, the likelihood ratio $r_{\boldsymbol{\theta}_\infty}^t = \frac{p_r^t}{p_{\boldsymbol{\theta}_\infty}^t}$ is the target of the discriminator, and BCE loss $\mathcal{L}_{\boldsymbol{\phi}}$ can be generalized into

(a) SDEdit+DG improves translation quality

(b) Realism vs. Faithfulness

Figure 12: I2I translation task. (a) The discriminator-guided translated samples from SDEdit are more realistic than non-guided samples from SDEdit given the same translation starting time $t_{mid} = 0.5$. (b) DG partially mitigates the trade-off between realism and faithfulness. Also, the weighted DG improves sample realism without hurting faithfulness.

Table 7: Ablation study for h-function on CIFAR-10.

Model	Density-Ratio Matching $D_h(r_{\theta_\infty} \| r_\phi)$	FID
EDM	-	2.03
EDM-G++	UKL (Nguyen et al., 2010) $h(r) = r \log r - r$	1.84
	LSIF (Kanamori et al., 2009) $h(r) = (r-1)^2/2$	1.84
	BCE ($\mathcal{L}_\phi$ of Eq. 4) $h(r) = r \log r - (r+1) \log(1+r)$	1.77

the family of h-Bregman divergence (Sugiyama et al., 2012)

$$D_h(r_{\theta_\infty} \| r_\phi) = \int \lambda(t) \mathbb{E}_{p^t_{\theta_\infty}(\mathbf{x}_t)} \big[h(r^t_{\theta_\infty}(\mathbf{x}_t)) - h(r^t_\phi(\mathbf{x}_t)) - \partial h(r^t_\phi(\mathbf{x}_t))(r^t_{\theta_\infty}(\mathbf{x}_t) - r^t_\phi(\mathbf{x}_t)) \big] \, \mathrm{d}t,$$

where $r_\phi = \frac{d_\phi}{1-d_\phi}$. It is worth noting that the BCE loss is the unique divergence that belongs to both the h-Bregman divergence and the f-divergence (Amari, 2009). Therefore, our approach suggests a new divergence family for score estimation, see Appendix B for more discussion. Table 7 presents an experiment on Bregmena divergence.

Another potential direction is the simultaneous training of discriminator and score networks, which could be more appealing than GAN as it is a min-min problem instead of a mini-max GAN problem. However, their loss functions are independent, so their joint interplay would be restricted and marginally enhances sample quality in Figure 13.

As an alternative, we could modify the score loss to an f-divergence (Song et al., 2021)

$$D_f(p_r(\mathbf{x}_0) \| p_\theta(\mathbf{x}_0)) = D_f(p^T_r(\mathbf{x}_T) \| \pi(\mathbf{x}_T)) + E^f_{\theta,\phi},$$

where $E^f_{\theta,\phi} = \frac{1}{2} \int_0^T g^2(t) \mathbb{E} \big[f'' \big(\frac{p^t_r(\mathbf{x}_t)}{p^t_\theta(\mathbf{x}_t)} \big) \frac{p^t_r(\mathbf{x}_t)}{p^t_\theta(\mathbf{x}_t)} \| \nabla \log p^t_r(\mathbf{x}_t) - \mathbf{s}_\theta(\mathbf{x}_t, t) \|_2^2 \big] \, \mathrm{d}t$, which estabilishes a connection between discriminator and score losses. The f-divergence allows

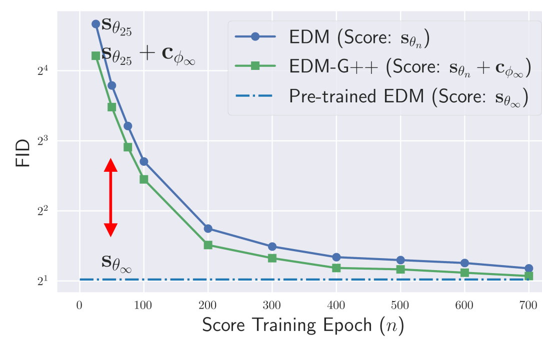


Figure 13: FID of simultaneous training on CIFAR-10.

better score estimation in perceptually plausible region by weighting the score matching more on the spatial domain with high likelihood ratio $\frac{p^t_r}{p^t_\theta}$, which could be an advantage over KL divergence training. We leave it as future work.

7. Conclusion

This paper refines the denoising process with an adjusted score estimation. With the proposed method, we could further optimize the divergence of a pre-trained score model. Empirical results demonstrate that our approach achieves new SOTA FIDs on all datasets. The deepfake images are one of the potential risks of the negative usage of this work.

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (NO. 2022-0-00077, AI Technology Development for Commonsense Extraction, Reasoning, and Inference from Heterogeneous Data). Also, this work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government(MSIT) (NRF-2019R1A5A1028324). We thank to Jiaming Song, Byunghu Na and Seungjae Shin for their sincere and warm advice to our work.

References

Amari, S.-I. $alpha$-divergence is unique, belonging to both f-divergence and bregman divergence classes. *IEEE Transactions on Information Theory*, 55(11):4925–4931, 2009.

Anderson, B. D. Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

Aneja, J., Schwing, A., Kautz, J., and Vahdat, A. A contrastive learning approach for training variational autoencoder priors. *Advances in neural information processing systems*, 34:480–493, 2021.

Ascher, U. M. and Petzold, L. R. *Computer methods for ordinary differential equations and differential-algebraic equations*, volume 61. Siam, 1998.

Azadi, S., Olsson, C., Darrell, T., Goodfellow, I., and Odena, A. Discriminator rejection sampling. In *International Conference on Learning Representations*, 2019.

Bauer, M. and Mnih, A. Resampled priors for variational autoencoders. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 66–75. PMLR, 2019.

Carlini, N., Hayes, J., Nasr, M., Jagielski, M., Sehwag, V., Tramèr, F., Balle, B., Ippolito, D., and Wallace, E. Extracting training data from diffusion models. *arXiv preprint arXiv:2301.13188*, 2023.

Chao, C.-H., Sun, W.-F., Cheng, B.-W., Lo, Y.-C., Chang, C.-C., Liu, Y.-L., Chang, Y.-L., Chen, C.-P., and Lee, C.-Y. Denoising likelihood score matching for conditional score-based data generation. In *International Conference on Learning Representations*, 2022.

Che, T., Zhang, R., Sohl-Dickstein, J., Larochelle, H., Paull, L., Cao, Y., and Bengio, Y. Your gan is secretly an energy-based model and you should use discriminator driven latent sampling. *Advances in Neural Information Processing Systems*, 33:12275–12287, 2020.

Chen, T. On the importance of noise scheduling for diffusion models. *arXiv preprint arXiv:2301.10972*, 2023.

Chen, Y., Georgiou, T. T., and Pavon, M. On the relation between optimal transport and schrödinger bridges: A stochastic control viewpoint. *Journal of Optimization Theory and Applications*, 169(2):671–691, 2016.

Daras, G., Delbracio, M., Talebi, H., Dimakis, A. G., and Milanfar, P. Soft diffusion: Score matching for general corruptions. *arXiv preprint arXiv:2209.05442*, 2022.

Daras, G., Dagan, Y., Dimakis, A. G., and Daskalakis, C. Consistent diffusion models: Mitigating sampling drift by learning to be consistent. *arXiv preprint arXiv:2302.09057*, 2023.

De Bortoli, V. Convergence of denoising diffusion models under the manifold hypothesis. *Transactions on Machine Learning Research*, 2022.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34, 2021.

Dockhorn, T., Vahdat, A., and Kreis, K. Score-based generative modeling with critically-damped langevin diffusion. In *International Conference on Learning Representations*, 2022.

Dormand, J. R. and Prince, P. J. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

Gutmann, M. and Hyvärinen, A. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, 2010.

Hang, T., Gu, S., Li, C., Bao, J., Chen, D., Hu, H., Geng, X., and Guo, B. Efficient diffusion training via min-snr weighting strategy. *arXiv preprint arXiv:2303.09556*, 2023.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer, 2009.

Ho, J., Jain, A., and Abbeel, P. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Ho, J., Saharia, C., Chan, W., Fleet, D. J., Norouzi, M., and Salimans, T. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23(47):1–33, 2022a.

Ho, J., Salimans, T., Gritsenko, A., Chan, W., Norouzi, M., and Fleet, D. J. Video diffusion models. 2022b.

Hoogeboom, E., Heek, J., and Salimans, T. simple diffusion: End-to-end diffusion for high resolution images. *arXiv preprint arXiv:2301.11093*, 2023.

Jabri, A., Fleet, D., and Chen, T. Scalable adaptive computation for iterative generation. *arXiv preprint arXiv:2212.11972*, 2022.

Jolicoeur-Martineau, A., Piché-Taillefer, R., Combes, R. T. d., and Mitliagkas, I. Adversarial score matching and improved sampling for image generation. 2021.

Kanamori, T., Hido, S., and Sugiyama, M. A least-squares approach to direct importance estimation. *The Journal of Machine Learning Research*, 10:1391–1445, 2009.

Karras, T., Aittala, M., Aila, T., and Laine, S. Elucidating the design space of diffusion-based generative models. 2022.

Kato, M. and Teshima, T. Non-negative bregman divergence minimization for deep direct density ratio estimation. In *International Conference on Machine Learning*, pp. 5320–5333. PMLR, 2021.

Kawar, B., Elad, M., Ermon, S., and Song, J. Denoising diffusion restoration models. 2022.

Kim, D., Na, B., Kwon, S. J., Lee, D., Kang, W., and Moon, I.-C. Maximum likelihood training of implicit nonlinear diffusion models. 2022a.

Kim, D., Shin, S., Song, K., Kang, W., and Moon, I.-C. Soft truncation: A universal training technique of score-based diffusion model for high precision score estimation. In *International Conference on Machine Learning*, pp. 11201–11228. PMLR, 2022b.

Kim, Y., Kim, D., Lee, H., and Moon, I.-c. Unsupervised controllable generation with score-based diffusion models: Disentangled latent code guidance. In *NeurIPS 2022 Workshop on Score-Based Methods*.

Kingma, D. P. and Gao, R. Understanding the diffusion objective as a weighted integral of elbos. *arXiv preprint arXiv:2303.00848*, 2023.

Kingma, D. P., Salimans, T., Poole, B., and Ho, J. Variational diffusion models. In *Advances in Neural Information Processing Systems*, 2021.

Lai, C.-H., Takida, Y., Murata, N., Uesaka, T., Mitsufuji, Y., and Ermon, S. Regularizing score-based models with score fokker-planck equations. 2022.

Lyu, Z., Xu, X., Yang, C., Lin, D., and Dai, B. Accelerating diffusion models via early stop of the diffusion process. *arXiv preprint arXiv:2205.12524*, 2022.

Meng, C., He, Y., Song, Y., Song, J., Wu, J., Zhu, J.-Y., and Ermon, S. Sdedit: Guided image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations*, 2021.

Nguyen, X., Wainwright, M. J., and Jordan, M. I. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

Nichol, A. Q. and Dhariwal, P. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pp. 8162–8171. PMLR, 2021.

Okhotin, A., Molchanov, D., Arkhipkin, V., Bartosh, G., Alanov, A., and Vetrov, D. Star-shaped denoising diffusion probabilistic models. *arXiv preprint arXiv:2302.05259*, 2023.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. *arXiv preprint arXiv:2212.09748*, 2022.

Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., and Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

Rhodes, B., Xu, K., and Gutmann, M. U. Telescoping density-ratio estimation. *Advances in neural information processing systems*, 33:4905–4916, 2020.

Robbins, H. E. *An empirical Bayes approach to statistics*. Springer, 1992.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3): 211–252, 2015.

Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E. L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35: 36479–36494, 2022.

Singer, U., Polyak, A., Hayes, T., Yin, X., An, J., Zhang, S., Hu, Q., Yang, H., Ashual, O., Gafni, O., et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022.

Song, J., Meng, C., and Ermon, S. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020a.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems*, 32, 2019.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020b.

Song, Y., Durkan, C., Murray, I., and Ermon, S. Maximum likelihood training of score-based diffusion models. *Advances in Neural Information Processing Systems*, 34, 2021.

Su, X., Song, J., Meng, C., and Ermon, S. Dual diffusion implicit bridges for image-to-image translation. In *The Eleventh International Conference on Learning Representations*, 2022.

Sugiyama, M., Suzuki, T., and Kanamori, T. Density-ratio matching under the bregman divergence: a unified framework of density-ratio estimation. *Annals of the Institute of Statistical Mathematics*, 64(5):1009–1044, 2012.

Vahdat, A., Kreis, K., and Kautz, J. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34, 2021.

Voleti, V., Jolicoeur-Martineau, A., and Pal, C. Mcvd: Masked conditional video diffusion for prediction, generation, and interpolation. 5(4.1):4, 2022.

Wang, Z., Zheng, H., He, P., Chen, W., and Zhou, M. Diffusion-gan: Training gans with diffusion. *arXiv preprint arXiv:2206.02262*, 2022.

Xiao, Z., Kreis, K., and Vahdat, A. Tackling the generative learning trilemma with denoising diffusion gans. In *International Conference on Learning Representations*, 2022.

Zheng, G., Li, S., Wang, H., Yao, T., Chen, Y., Ding, S., and Li, X. Entropy-driven sampling and training scheme for conditional diffusion generation. In *European Conference on Computer Vision*, pp. 754–769. Springer, 2022a.

Zheng, H., He, P., Chen, W., and Zhou, M. Truncated diffusion probabilistic models and diffusion-based adversarial auto-encoders. *arXiv preprint arXiv:2202.09671*, 2022b.

A. Proofs and More Analysis

A.1. Proof of Theorem 1

Throughout the section, we assume that the assumptions made in Appendix A of Song et al. (2021) hold.

Theorem 1. *Suppose $p_{\boldsymbol{\theta}}$ be the solution of the time-reversal generative process of Eq. (3). Let p_r^t and $p_{\boldsymbol{\theta}}^t$ be the marginal densities (at t) of the forward-time SDE $\mathrm{d}\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}_t$ starting from p_r and $p_{\boldsymbol{\theta}}$, respectively. If $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}, T) = \nabla \log \pi(\mathbf{x})$, where π is the prior distribution, and the log-likelihood $\log p_{\boldsymbol{\theta}}$ equals its evidence lower bound $\mathcal{L}_{\boldsymbol{\theta}}$, then*

$$\mathrm{d}\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\nabla \log p_r^t(\mathbf{x}_t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

coincides with a diffusion process with adjusted score,

$$\mathrm{d}\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}} + \mathbf{c}_{\boldsymbol{\theta}})(\mathbf{x}_t, t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

for $\mathbf{c}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) := \nabla \log \frac{p_r^t(\mathbf{x}_t)}{p_{\boldsymbol{\theta}}^t(\mathbf{x}_t)}$.

To prove the theorems, we define a family of rotation-free score functions by $\mathbf{S}_{sol}$ in Definition 1.

Definition 1 (Definition 1 of Kim et al. (2022a))**.** Let $\mathbf{S}_{div} = \{\mathbf{v} : \mathbb{R}^d \to \mathbb{R}^d | \mathbf{v} = \nabla \log p \text{ for some probability } p\}$ be the family of rotation-free score functions. Define $\mathbf{S}_{sol}$ be a family of time-conditioned score network $\mathbf{s}$ that satisfies the following: there exists q_0 and q_t such that $\mathbf{s}(\mathbf{x}, t) = \nabla \log q_t(\mathbf{x})$ almost everywhere, where q_t is the marginal density at time t of $\mathrm{d}\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}_t$, starting from $\mathbf{x}_0 \sim q_0$.

With Definition 1, Kim et al. (2022a) introduces the necessary and sufficient condition for $\mathbf{s}_{\boldsymbol{\theta}} \in \mathbf{S}_{sol}$.

Lemma 1 (Theorem 2 of Kim et al. (2022a))**.** *Suppose $\mathbf{s}_{\boldsymbol{\theta}}$ is twice continuously differentiable with respect to $\mathbf{x}$. Then,*

$$D_{KL}(p_r \| p_{\boldsymbol{\theta}}) = \frac{1}{2}\int_0^T g^2(t)\mathbb{E}\big[\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla \log p_r^t(\mathbf{x}_t)\|_2^2\big]\,\mathrm{d}t + D_{KL}(p_r^T \| \pi)$$

holds if and only if $\mathbf{s}_{\boldsymbol{\theta}} \in \mathbf{S}_{sol}$.

Lemma 2. *The log-likelihood equals the evidence lower bound if and only if*

$$D_{KL}(p_r \| p_{\boldsymbol{\theta}}) = \frac{1}{2}\int_0^T g^2(t)\mathbb{E}\big[\|\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) - \nabla \log p_r^t(\mathbf{x}_t)\|_2^2\big]\,\mathrm{d}t + D_{KL}(p_r^T \| \pi).$$

Combining Lemmas 1 and 2, we yield that $\mathbf{s}_{\boldsymbol{\theta}} \in \mathbf{S}_{sol}$ if and only if the log-likelihood equals the evidence lower bound. Now, we provide the proof.

Proof of Theorem 1. From Lemmas 1 and 2 combined, we obtain $\mathbf{s}_{\boldsymbol{\theta}} \in \mathbf{S}_{sol}$. Therefore, there exists q_0 such that q_t is the marginal density of $\mathrm{d}\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}_t$ and satisfies $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) = \nabla \log q_t(\mathbf{x}_t)$ almost everywhere. Then, the solution of the generative process

$$\begin{aligned}\mathrm{d}\mathbf{x}_t &= \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t \\ &= \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\nabla \log q_t(\mathbf{x}_t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,\end{aligned} \tag{7}$$

starting from q_T at T attains q_t as its marginal density at t. As $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_T, T) = \nabla \log \pi(\mathbf{x}_T)$, the marginal density q_T at T becomes the prior distribution π, and we get $q_0 = p_{\boldsymbol{\theta}}$.

Hence, $p_{\boldsymbol{\theta}}^t$ is the marginal density of $\mathrm{d}\mathbf{x}_t = \mathbf{f}(\mathbf{x}_t, t)\,\mathrm{d}t + g(t)\,\mathrm{d}\mathbf{w}_t$, starting from $\mathbf{x}_0 \sim p_{\boldsymbol{\theta}} = q_0$. From the uniqueness, we conclude that $p_{\boldsymbol{\theta}}^t = q_t$ for all t and thus $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t, t) = \nabla \log p_{\boldsymbol{\theta}}^t(\mathbf{x}_t)$, which leads the desired result. □

A.2. Proof of Theorem 2

Theorem 2. *If the assumptions of Theorem 1 hold, then*

$$D_{KL}(p_r \| p_{\boldsymbol{\theta}}) = D_{KL}(p_r^T \| \pi) + E_{\boldsymbol{\theta}},$$

$$D_{KL}(p_r\|p_{\boldsymbol{\theta},\boldsymbol{\phi}}) \le D_{KL}(p_r^T\|\pi) + E_{\boldsymbol{\theta},\boldsymbol{\phi}},$$

where

$$E_{\boldsymbol{\theta}} = \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\left[\|\nabla\log p_r^t - \nabla\log p_{\boldsymbol{\theta}}^t\|_2^2\right],$$
$$E_{\boldsymbol{\theta},\boldsymbol{\phi}} = \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\left[\|\mathbf{c}_{\boldsymbol{\theta}} - \mathbf{c}_{\boldsymbol{\phi}}\|_2^2\right] \mathrm{d}t.$$

Proof of Theorem 2. We have

$$\begin{aligned}
\mathrm{d}\mathbf{x}_t &= \big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}} + \mathbf{c}_{\boldsymbol{\phi}})\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t \\
&= \Big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)\Big(\mathbf{s}_{\boldsymbol{\theta}} + \nabla\log\frac{d_{\boldsymbol{\phi}}}{1-d_{\boldsymbol{\phi}}}\Big)\Big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t \\
&= \Big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)\Big(\nabla\log p_g^t + \nabla\log\frac{d_{\boldsymbol{\phi}}}{1-d_{\boldsymbol{\phi}}}\Big)\Big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t \\
&= \Big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)\Big(\nabla\log p_g^t + \nabla\log\frac{d_{\boldsymbol{\phi}_*}}{1-d_{\boldsymbol{\phi}_*}} - \nabla\log\frac{d_{\boldsymbol{\phi}_*}}{1-d_{\boldsymbol{\phi}_*}} + \nabla\log\frac{d_{\boldsymbol{\phi}}}{1-d_{\boldsymbol{\phi}}}\Big)\Big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t \\
&= \Big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)\Big(\nabla\log p_g^t + \nabla\log\frac{p_r^t}{p_g^t} - \nabla\log\frac{d_{\boldsymbol{\phi}_*}}{1-d_{\boldsymbol{\phi}_*}} + \nabla\log\frac{d_{\boldsymbol{\phi}}}{1-d_{\boldsymbol{\phi}}}\Big)\Big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t \\
&= \Big[\mathbf{f}(\mathbf{x}_t,t) - g^2(t)\Big(\nabla\log p_r^t - \nabla\log\frac{d_{\boldsymbol{\phi}_*}}{1-d_{\boldsymbol{\phi}_*}} + \nabla\log\frac{d_{\boldsymbol{\phi}}}{1-d_{\boldsymbol{\phi}}}\Big)\Big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t.
\end{aligned}$$

Applying the Girsanov theorem to this generative SDE with the reverse-time data SDE with the data-processing inequality, we get

$$D_{KL}(p_r\|p_{\boldsymbol{\theta},\boldsymbol{\phi}}) \le D_{KL}(p_r^T\|\pi) + \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\left[\left\|\nabla\log\frac{d_{\boldsymbol{\phi}_*}}{1-d_{\boldsymbol{\phi}_*}} - \nabla\log\frac{d_{\boldsymbol{\phi}}}{1-d_{\boldsymbol{\phi}}}\right\|_2^2\right]\mathrm{d}t.$$

Also, the equality of $D_{KL}(p_r\|p_{\boldsymbol{\theta}}) = D_{KL}(p_r^T\|\pi) + \frac{1}{2}\int_0^T g^2(t)\mathbb{E}_{p_r^t}\left[\|\nabla\log p_r^t - \nabla\log p_{\boldsymbol{\theta}}^t\|_2^2\right]\mathrm{d}t$ holds by Lemma 2. □

A.3. Validity of Assumption in Theorem 1

It now remains to show if the assumptions of Theorem 1 holds in practice. Figure 14 compares the NLL and NELBO curve of $\mathbf{s}_{\boldsymbol{\theta}_\infty}$. Figure 14 shows that the equality condition of NLL and NELBO holds approximately on a wide range of time horizon, in practice.

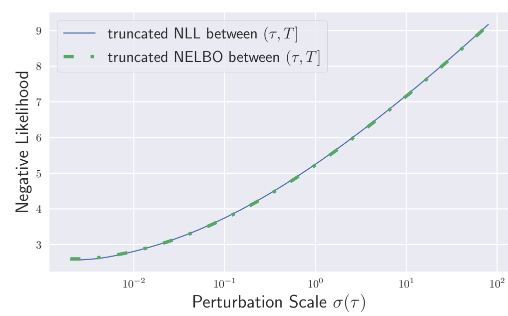


Figure 14: NELBO is uniformly close to NLL on CIFAR-10.

Figure 14 is obtained from the EDM checkpoint (Karras et al., 2022). For the NLL and NELBO computation, we follow Kim et al. (2022a;b): we estimate the *truncated* NLL and NELBO for the purpose of thorough investigation throughout timescales. The truncated NLL and NELBO assumes that the score network is given only on $(\tau, T]$ with τ a truncation bound. Then, the truncated NLL is the right-hand-side of

$$\mathbb{E}_{\mathbf{x}_0\sim p_r}[-\log p_g(\mathbf{x}_0)] \le \mathbb{E}_{\mathbf{x}_\tau\sim p_r^\tau}[-\log p_{\boldsymbol{\theta},\tau}(\mathbf{x}_\tau)] + R_\tau(\boldsymbol{\theta}),$$

where $R_\tau(\boldsymbol{\theta}) = \mathbb{E}_{\mathbf{x}_0\sim p_r}\left[\int p_{0\tau}(\mathbf{x}_\tau|\mathbf{x}_0)\log\frac{p_{0\tau}(\mathbf{x}_\tau|\mathbf{x}_0)}{p_{\boldsymbol{\theta}}(\mathbf{x}_0|\mathbf{x}_\tau)}\mathrm{d}\mathbf{x}_\tau\right]$ and $p_{\boldsymbol{\theta},\tau}$ is the marginal density at τ of the generative process. We evaluate $p_{\boldsymbol{\theta},\tau}(\mathbf{x}_\tau)$ by solving the instantaneous change-of-variable formula

$$\frac{\mathrm{d}}{\mathrm{d}t}\log p_{\boldsymbol{\theta},t}(\mathbf{x}_t) = -\mathrm{tr}\Big(\nabla\Big[\mathbf{f}(\mathbf{x}_t,t) - \frac{1}{2}g^2(t)\mathbf{x}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)\Big]\Big)$$

from $t=\tau$ to T, of which probability flow ODE is

$$\mathrm{d}\mathbf{x}_t = \Big[\mathbf{f}(\mathbf{x}_t,t) - \frac{1}{2}g^2(t)\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)\Big]\mathrm{d}t.$$

Analogously, we evaluate the truncated evidence lower bound as the right-hand-side of

$$\mathcal{L}([0,T]) \leq \mathcal{L}([\tau,T]) + R_\tau(\boldsymbol{\theta}),$$

where

$$\mathcal{L}([\tau,T]) = \frac{1}{2}\int_\tau^T \mathbb{E}\Big[g^2(t)\|\nabla \log p_{0t}(\mathbf{x}_t|\mathbf{x}_0) - \mathbf{s}_{\boldsymbol{\theta}}(\mathbf{x}_t,t)\|_2^2 - g^2(t)\|\nabla \log p_{0t}(\mathbf{x}_t|\mathbf{x}_0)\|_2^2 - 2\mathrm{div}(\mathbf{f})\Big]\,\mathrm{d}t - \mathbb{E}\big[\log \pi(\mathbf{x}_T)\big].$$

We utilize the importance sampling technique (for the integration with respect to t in $\mathcal{L}$) to minimize the estimation variance of the evidence lower bound. To derive the closed-form importance-weighted time, first, observe that the EDM forward diffusion is given by $\sigma^2(t) = (\sigma_{min}^{1/\rho} + t(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho}))$. Then, with the importance weight of $\frac{g^2(t)}{\sigma^2(t)}$, if we define $F(t) = \frac{1}{Z}\int_\tau^t \frac{g^2(s)}{\sigma^2(s)}\,\mathrm{d}s$ to be the cumulative distribution function of the importance sampler, the importance-weighted time becomes $t = F^{-1}(u)$ for uniformly sampled u on $[0,T]$.

Now, the antiderivative of the importance weight becomes

$$\begin{aligned}\mathcal{F}(t) &= \int \frac{g^2(t)}{\sigma^2(t)}\,\mathrm{d}t \\ &= \int 2\rho \frac{\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho}}{\sigma_{min}^{1/\rho} + t(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}\,\mathrm{d}t \\ &= 2\rho \log(\sigma_{min}^{1/\rho} + t(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})),\end{aligned}$$

and the normalizing constant becomes

$$\begin{aligned}Z &= \int_\tau^T \frac{g^2(t)}{\sigma^2(t)}\,\mathrm{d}t \\ &= \mathcal{F}(T) - \mathcal{F}(\tau) \\ &= 2\rho \log \frac{\sigma_{min}^{1/\rho} + T(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}.\end{aligned}$$

Therefore, we have

$$\begin{aligned}& t = F^{-1}(u) \\ \iff\ & u = F(t) = \frac{1}{Z}\int_\tau^t \frac{g^2(s)}{\sigma^2(s)}\,\mathrm{d}s = \frac{1}{Z}\big(\mathcal{F}(t) - \mathcal{F}(\tau)\big) \\ \iff\ & u \log \frac{\sigma_{min}^{1/\rho} + T(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})} = \log \frac{\sigma_{min}^{1/\rho} + t(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})} \\ \iff\ & \left(\frac{\sigma_{min}^{1/\rho} + T(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}\right)^u = \frac{\sigma_{min}^{1/\rho} + t(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})} \\ \iff\ & \big(\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})\big)\left(\frac{\sigma_{min}^{1/\rho} + T(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}\right)^u = \sigma_{min}^{1/\rho} + t(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho}) \\ \iff\ & t = \left(\big(\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})\big)\left(\frac{\sigma_{min}^{1/\rho} + T(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}{\sigma_{min}^{1/\rho} + \tau(\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho})}\right)^u - \sigma_{min}^{1/\rho}\right) \Big/ (\sigma_{max}^{1/\rho} - \sigma_{min}^{1/\rho}).\end{aligned}$$

A.4. Why $p_{\boldsymbol{\theta}}^t$ is defined as a forward marginal rather than a generative marginal

Unintuitively, we define $p_{\boldsymbol{\theta}}^t$ as a forward-time marginal density rather than a reverse-time generative marginal. We design $p_{\boldsymbol{\theta}}^t$ as a forward-time marginal by two reasons. First, it saves memory. If it was the reverse-time generative marginal, the generated dataset $\mathcal{G}$ should contain the whole sample trajectories to optimize the time-embedded discriminator, and this could be prohibitive when we run ~ 1000 steps to generate a sample. Instead, if we use the forward-time marginal, we could only save the final sample as $\mathcal{G}$, and we optimize the discriminator by diffusing it with *arbitrary* diffusion time with *arbitrary* diffusion noise. Here comes the second advantage: as we could update the discriminator with arbitrary diffusion noise, it attains the data augmentation effect in training the discriminator network, and it could prevent the overfitting issue at large time. On the other hand, if it was the generative marginal, the discriminator guidance needs more sample data to prevent overfitting.

B. More on Bregman Divergences

The most general framework for learning the likelihood ratio $\frac{p_r^t}{p_{\boldsymbol{\theta}}^t}$ is using the Bregman divergence (Sugiyama et al., 2012). In this section, we investigate the effect of representative Bregman divergences. To begin with, let us define the Bregman divergence in an abstract form. Suppose $r^*(\mathbf{x}) = \frac{p_{nu}(\mathbf{x})}{p_{de}(\mathbf{x})}$ be the target density-ratio to be estimated with $r_{\boldsymbol{\phi}}$, parametrized by $\boldsymbol{\phi}$. Then,

$$\begin{aligned} D_h(r^* \| r_{\boldsymbol{\phi}}) &= \int p_{de}(\mathbf{x}) B_h(r^*(\mathbf{x}) \| r_{\boldsymbol{\phi}}(\mathbf{x}))\,\mathrm{d}\mathbf{x} \\ &= \int p_{de}(\mathbf{x}) \Big(h\big(r^*(\mathbf{x})\big) - h\big(r_{\boldsymbol{\phi}}(\mathbf{x})\big) - \partial h\big(r_{\boldsymbol{\phi}}(\mathbf{x})\big)\big(r^*(\mathbf{x}) - r_{\boldsymbol{\phi}}(\mathbf{x})\big) \Big)\,\mathrm{d}\mathbf{x}, \end{aligned}$$

where B_h is the data-level Bregman divergence. For a twice continuously differentiable convex function h with a bounded derivative ∂h, the Bregman divergence quantifies the discrepancy between two likelihood ratios. Subtracting a constant term $\int p_{de}(\mathbf{x}) h(r^*(\mathbf{x}))\,\mathrm{d}x$, we obtain (up to a constant)

$$D_h(r^* \| r_{\boldsymbol{\phi}}) = \int p_{de}(\mathbf{x}) \Big[\partial h\big(r_{\boldsymbol{\phi}}(\mathbf{x})\big) r_{\boldsymbol{\phi}}(\mathbf{x}) - h\big(r_{\boldsymbol{\phi}}(\mathbf{x})\big) \Big]\,\mathrm{d}\mathbf{x} - \int p_{nu}(\mathbf{x}) \Big[\partial h\big(r_{\boldsymbol{\phi}}(\mathbf{x})\big) \Big]\,\mathrm{d}\mathbf{x}.$$

A few non-exhaustive examples of the Bregman divergence are Least-Squared Importance Fitting (LSIF) (Kanamori et al., 2009), Binary Cross Entropy (BCE) (Hastie et al., 2009), and Kullback-Leibler Importance Estimation Procedure (KLIEP) (Nguyen et al., 2010). LSIF is equivalent to

$$\begin{aligned} BD_{f_{LSIF}}(r^* \| r_{\boldsymbol{\phi}}) &= \frac{1}{2} \int p_{de}(\mathbf{x}) \big(r^*(\mathbf{x}) - r_{\boldsymbol{\phi}}(\mathbf{x})\big)^2\,\mathrm{d}\mathbf{x} \\ &= \frac{1}{2} \int p_{de}(\mathbf{x}) r_{\boldsymbol{\phi}}^2(\mathbf{x})\,\mathrm{d}\mathbf{x} - \int p_{nu}(\mathbf{x}) r_{\boldsymbol{\phi}}(\mathbf{x})\,\mathrm{d}\mathbf{x}, \end{aligned}$$

with $h_{LSIF}(r) = (r-1)^2/2$. BCE is also widely denoted as Binary Kullback-Leibler (BKL), and is defined with $h_{BKL}(r) = r \log r - (1+r)\log(1+r)$. KLIEP is also known as the Unbounded Kullback-Leibler (UKL) with $h_{UKL}(r) = r \log r - r$, and is a Lagrangian of the constrained optimization problem of

$$D_{KL}(p_{nu} \| p_{nu,\boldsymbol{\phi}}) \text{ subject to } \int p_{nu,\boldsymbol{\phi}}(\mathbf{x})\,\mathrm{d}\mathbf{x} = 1,$$

where $p_{nu,\boldsymbol{\phi}}(\mathbf{x}) := p_{de}(\mathbf{x}) r_{\boldsymbol{\phi}}(\mathbf{x})$. The UKL Bregman divergence is defined as the Lagrangian of the above problem by

$$BD_{f_{UKL}}(r^* \| r_{\boldsymbol{\phi}}) = \int p_{de}(\mathbf{x}) r_{\boldsymbol{\phi}}(\mathbf{x})\,\mathrm{d}\mathbf{x} - \int p_{nu}(\mathbf{x}) \log r_{\boldsymbol{\phi}}(\mathbf{x})\,\mathrm{d}\mathbf{x}.$$

Now, we consider the time-dependent Bregman divergence to train the discriminator network. The time-dependent Bregman divergence is defined by

$$\int \lambda(t) D_h\bigg(\frac{p_r^t(\cdot)}{p_{\boldsymbol{\theta}}^t(\cdot)} \bigg\| \frac{1 - d_{\boldsymbol{\phi}}(\cdot, t)}{d_{\boldsymbol{\phi}}(\cdot, t)} \bigg)\,\mathrm{d}t,$$

where $\lambda(t)$ is a time-weighting function. We train the discriminator network with this time-dependent Bregman divergence of LSIF, BCE, and UKL. In a toy 2-dimensional case, Figure 15 visualizes the adjusted score by discriminators trained by aforementioned Bregman divergences. Figure 15 displays for the case where the estimated score is confusing on the exact location of bi-modalities of the data distribution. Figure 15 illustrates that BCE estimates the correction term $\mathbf{c}_{\boldsymbol{\theta}}$ most robust.

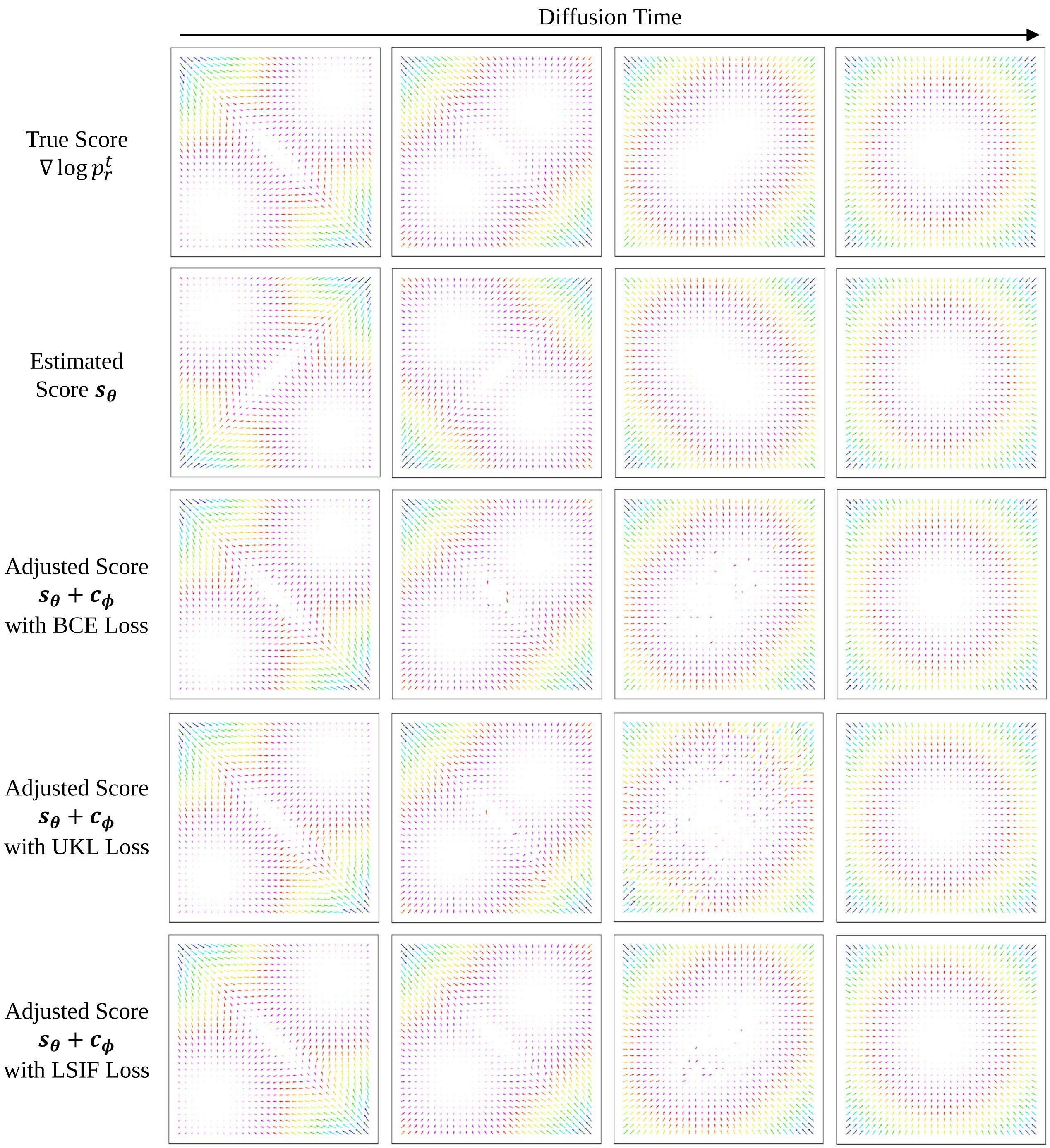
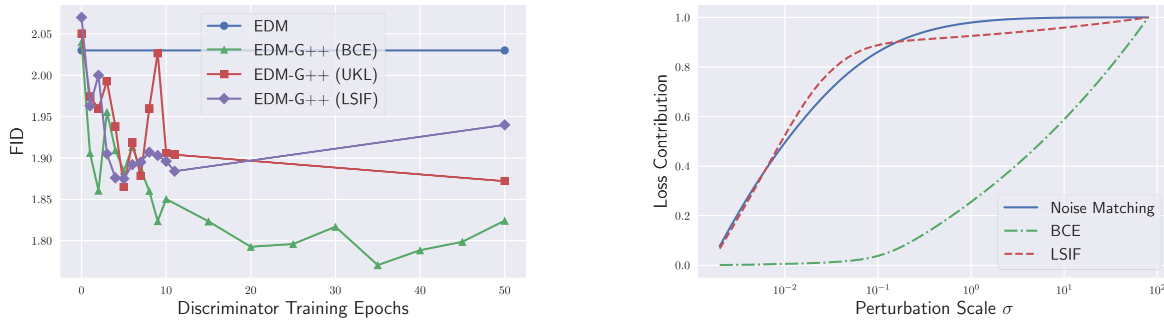


Figure 15: A 2-dimentionsal toy case with a bimodal Gaussian data distribution. In this experiment, we visualize the adjusted scores for BKL/UKL/LSIF losses under the assumption that the estimated score is misleadingly capturing the location of bimodalities.



(a) FID by Discriminator Training

(b) Loss Contribution

Figure 16: Study of EDM on CIFAR-10 with respect to Bregman divergences. (a) illustrates how sample quality is improved by discriminator training with various loss functions, and (b) shows the mechanism of such different FID.

Figure 16-(a) shows that the listed divergences have distinctive characteristics in sample quality. Similar to the 2-dimensional case, BCE performs the best. Figure 16-(b) illustrates why such a behavioral difference occurs. We visualize the (normalized) cumulative loss to the perturbation scale. For the noise matching, most loss contribution is on the range of small diffusion scale (Kim et al., 2022b). On the contrary, BCE is qualitatively different from the score loss: most of loss is concentrated on the range of large diffusion scale, and it would be presumably the reason for the arguments in Section 6. Figure 16-(b) shows that LSIF is similar to the noise matching loss. For the UKL loss, we do not plot the cumulative loss because the loss is neither strictly positive nor strictly negative for every diffusion scale.

Figure 17 conducts the ablation study of w_t^{DG} for various Bregman divergences on CIFAR-10. It illustrates that BCE loss performs the best but UKL loss is the most robust one.

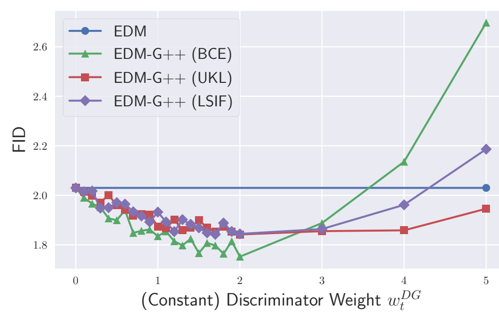


Figure 17: Study of w_t^{DG} for various Bregman divergences on CIFAR-10.

C. Details on Image-to-Image Translation

Figure 18 illustrates the step-by-step denoising process of I2I translation task. Suppose $\mathbf{x}_0 \sim \mathcal{S}$ is the source image, and $\mathbf{y}_0 \sim \mathcal{T}$ is the target image. Then, following SDEdit (Meng et al., 2021), we start denoising from $\mathbf{x}_\tau$ with

$$\mathrm{d}\mathbf{y}_t = [\mathbf{f}(\mathbf{y}_t, t) - g^2(t)\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{y}_t, t)]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

where $\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{y}_t, t)$ is the score network trained on the target domain. Suppose we define $\mathbf{z}_{\boldsymbol{\theta}}^t$ be the random variable of the solution process: $\mathbf{z}_{\boldsymbol{\theta}}^{t_{mid}} = \mathbf{x}_{t_{mid}}$ is the starting variable and $\mathbf{z}_{\boldsymbol{\theta}}^0$ is the translated variable. Suppose $q_{\boldsymbol{\theta}}^t$ be the probability distribution of $\mathbf{z}_{\boldsymbol{\theta}}^t$, $p_{\boldsymbol{\theta}} = q_{\boldsymbol{\theta}}^0$ be the probability distribution of $\mathbf{z}_{\boldsymbol{\theta}}^0$, and $p_{\boldsymbol{\theta}}^t$ be the probability distribution of the diffused variable by the forward SDE, starting from $\mathbf{z}_{\boldsymbol{\theta}}^0$. For the discriminator training, we sample $\mathbf{z}_t \sim \alpha p_{\mathcal{S}}^t(\mathbf{z}_t) + (1-\alpha)p_{\boldsymbol{\theta}}^t(\mathbf{z}_t)$ for the fake data and $\mathbf{y}_t \sim p_{\mathcal{T}}^t(\mathbf{y}_t)$ for the real data. Then, our discriminator guidance becomes

$$\mathrm{d}\mathbf{y}_t = \bigg[\mathbf{f}(\mathbf{y}_t, t) - g^2(t)\Big(\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{y}_t, t) + \nabla \log \frac{p_{\mathcal{T}}^t(\mathbf{y}_t)}{\alpha p_{\mathcal{S}}^t(\mathbf{y}_t) + (1-\alpha)p_{\boldsymbol{\theta}}^t(\mathbf{y}_t)}\Big)\bigg]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t.$$

When $t \approx t_{mid}$, we have $p_{\mathcal{S}}^t(\mathbf{z}_t) \gg p_{\boldsymbol{\theta}}^t(\mathbf{z}_t)$ for $\mathbf{z}_t \sim q_{\boldsymbol{\theta}}^t$ and the adjusted generative process is approximately

$$\mathrm{d}\mathbf{y}_t \approx \bigg[\mathbf{f}(\mathbf{y}_t, t) - g^2(t)\Big(\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{y}_t, t) + \nabla \log \frac{p_{\mathcal{T}}^t(\mathbf{y}_t)}{p_{\mathcal{S}}^t(\mathbf{y}_t)}\Big)\bigg]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

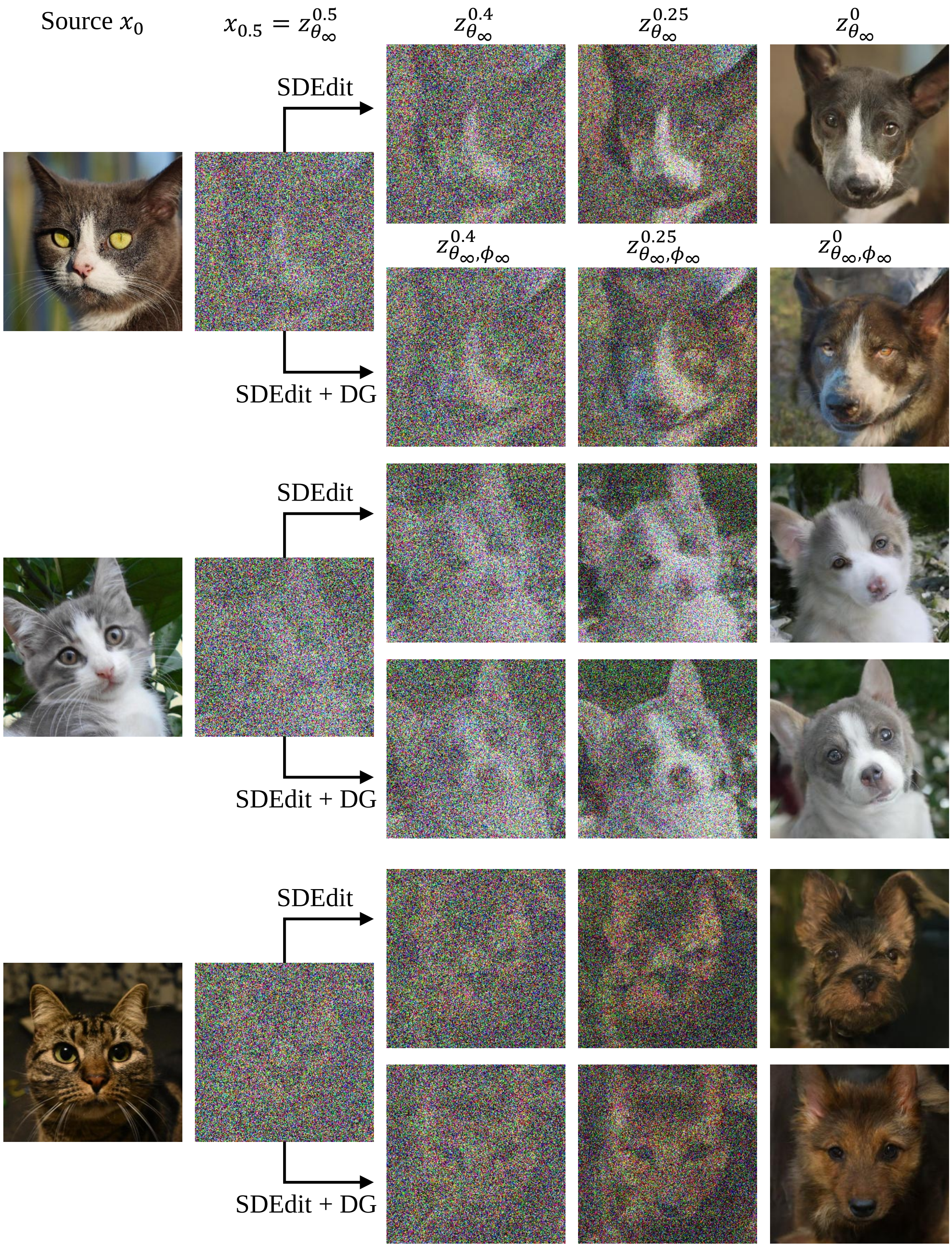


Figure 18: Step-by-step denoising illustration of image-to-image translation.

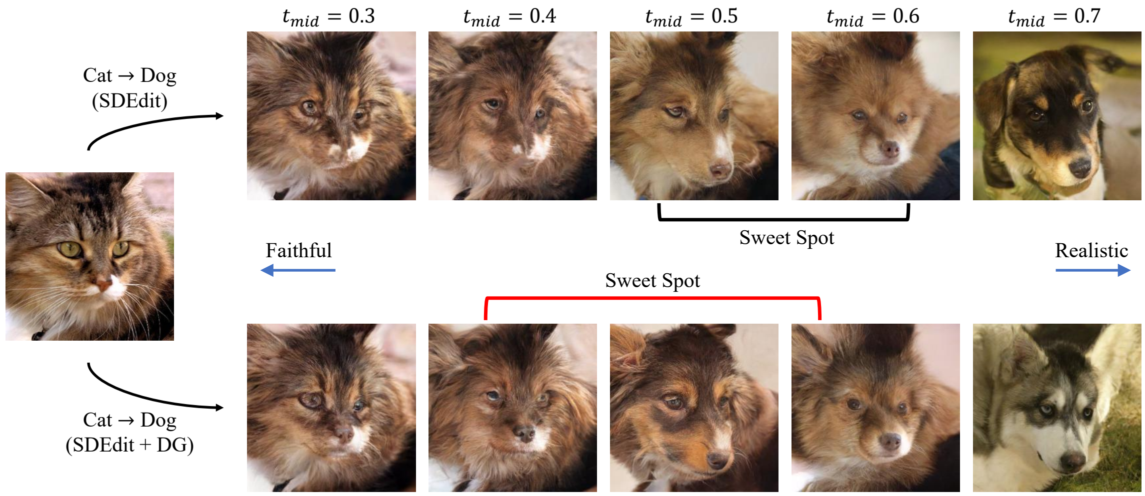


Figure 19: The discriminator guidance swifts the sweet spot to the range of small t_0.

so the discriminator guidance gives a direct signal to avoid from $p_{\mathcal{S}}^t$ when $t \approx t_{mid}$. When $t \approx 0$, we have $p_{\mathcal{S}}^t(\mathbf{z}_t) \ll p_{\boldsymbol{\theta}}^t(\mathbf{z}_t)$ for $\mathbf{z}_t \sim q_{\boldsymbol{\theta}}^t$ and the adjusted generative process is approximately

$$\mathrm{d}\mathbf{y}_t \approx \left[\mathbf{f}(\mathbf{y}_t, t) - g^2(t)\Big(\mathbf{s}_{\boldsymbol{\theta}}(\mathbf{y}_t, t) + \nabla \log \frac{p_{\mathcal{T}}^t(\mathbf{y}_t)}{p_{\boldsymbol{\theta}}^t(\mathbf{y}_t)}\Big)\right] \mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

and the discriminator is guiding the sample denoising toward the desired destination density $p_{\mathcal{T}}^t$, rather than the original destination density $p_{\boldsymbol{\theta}}^t$.

By applying the discriminator guidance, more realistic samples are generated with relatively small t_{mid}, compared to SDEdit, so we could lift the sweet spot of the image-to-image translation to a range of small t_{mid} as in Figure 19.

D. Experimental Details

D.1. Training and Sampling Details

Table 8 presents the experimental configuration for Tables 3, 4, and 5. Except for ImageNet 256x256, we solve the Probability-Flow ODE (PFODE) (Song et al., 2020b) for sampling. We use the adjusted PFODE

$$\frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}_t, t) - \frac{1}{2} g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + w_t^{DG} \mathbf{c}_{\boldsymbol{\phi}_\infty})(\mathbf{x}_t, t)$$

on $t \in [t_{mid}, T]$ and the unadjusted PFODE

$$\frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = \mathbf{f}(\mathbf{x}_t, t) - \frac{1}{2} g^2(t) \mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t)$$

on $t \in [0, t_{mid}]$.

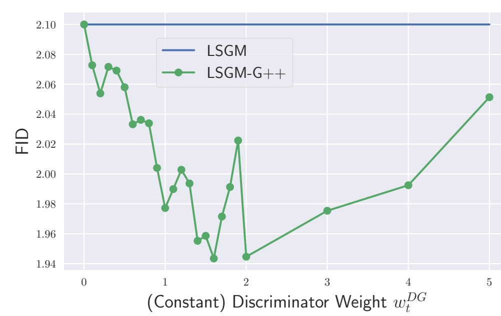


Figure 20: Ablation of w_t^{DG} in LSGM-G++ on CIFAR-10.

For LSGM-G++, we borrow the pre-trained CIFAR-10 checkpoint of LSGM with best FID (Vahdat et al., 2021) at `https://github.com/NVlabs/LSGM`. We do not use any of the pre-trained classifier for this latent-diffusion model, and we train a discriminator from scratch. We solve the unadjusted/adjusted PFODEs with explicit Runge-Kutta solver of order 5(4) (Dormand & Prince, 1980). We find $w_t^{DG} = 2$ works the best in practice and report the value in Table 3. Figure 20 presents the ablation study on DG weight.

For EDM-G++, we experiment with unconditional and conditional CIFAR-10, as well as FFHQ 64x64. We use the pre-trained score model of EDM (Karras et al., 2022) at `https://github.com/NVlabs/edm`. For the classifier, we borrow a pre-trained classifier from ADM (Dhariwal & Nichol, 2021) at `https://github.com/openai/guided-diffusion`

Table 8: Training and sampling configurations.

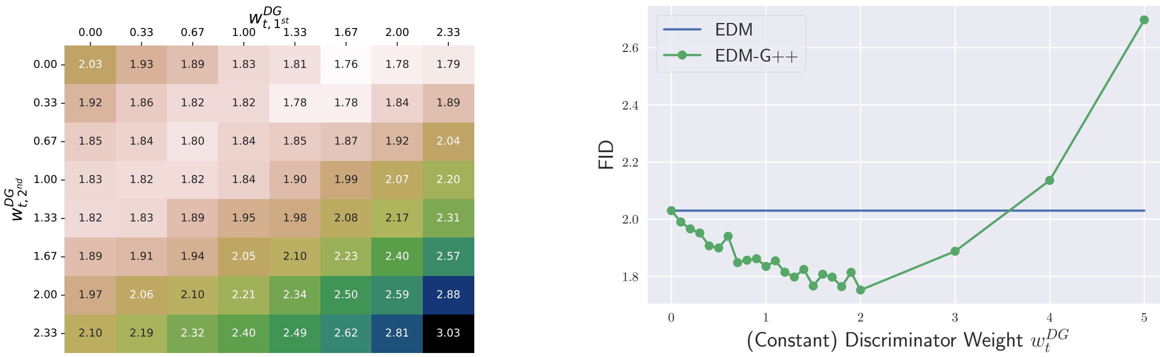
	CIFAR-10			CelebA	FFHQ	ImageNet256	
	LSGM-G++	EDM-G++		Soft Truncation-G++	EDM-G++	ADM-G++	DiT-G++
Pre-trained Score Network							
Model	LSGM	EDM	EDM	Soft Truncation	EDM	ADM	DiT
Class condition	✗	✗	✓	✗	✗	✓	✓
Discriminator Training							
SDE	LVP	LVP	LVP	CVP	CVP	LVP	LVP
Class condition	✗	✗	✓	✗	✗	✓	✓
Time sampling	Importance	Importance	Importance	Importance	Importance	Importance	Importance
λ	$\frac{g^2}{\sigma^2}$	$\frac{g^2}{\sigma^2}$	$\frac{g^2}{\sigma^2}$	$\frac{g^2}{\sigma^2}$	$\frac{g^2}{\sigma^2}$	$\frac{g^2}{\sigma^2}$	$\frac{g^2}{\sigma^2}$
Minimum diffusion time	0.01	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
EMA	✗	✗	✗	✗	✗	✗	✗
Batch size	128	128	128	128	128	512	512
$\#\mathcal{D}$	50,000	50,000	50,000	10,000	60,000	1,281,167	1,281,167
$\#\mathcal{G}$	50,000	25,000	50,000	10,000	60,000	400,000	1,281,167
# Epochs	280	60	250	150	250	10	7
GPUs	1x V100	1x V100	1x V100	1x V100	1x V100	1x A100	1x A100
Pre-trained Classifier							
Model	No classifier	ADM	ADM	ADM	ADM	ADM	ADM
Input shape (data dimension)	✗	(B,32,32,3)	(B,32,32,3)	(B,64,64,3)	(B,64,64,3)	(B,256,256,3)	(B,32,32,4)
Output shape (latent dimension)	✗	(B,8,8,512)	(B,8,8,512)	(B,8,8,512)	(B,8,8,512)	(B,8,8,512)	(B,8,8,384)
Shallow U-Net Architecture							
Input shape (latent dimension)	(B,16,16,180)	(B,8,8,512)	(B,8,8,512)	(B,8,8,512)	(B,8,8,512)	(B,8,8,512)	(B,8,8,384)
Output shape	(B,1)	(B,1)	(B,1)	(B,1)	(B,1)	(B,1)	(B,1)
Minimum value of discriminator (clip)	10^{-5}	0	0	0	0	10^{-5}	10^{-5}
Maximum value of discriminator (clip)	$1-10^{-5}$	1	1	1	1	$1-10^{-5}$	10^{-5}
Class condition	✗	✗	✓	✗	✗	✓	✓
# Resnet blocks	5	4	4	6	6	4	4
# Attention blocks	3	3	3	5	5	3	3
Attention resolutions	16, 8	8	8	8	8	8	8
Model channel	128	128	128	128	128	128	128
Channel multiplier	(1,2)	1	1	1	1	1	1
Sampling							
SDE	LVP	WVE	WVE	LVP	WVE	LVP	LVP
Class condition	✗	✗	✓	✗	✗	✓	✓
Minimum value of discriminator (clip)	10^{-3}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}	10^{-5}
Maximum value of discriminator (clip)	$1-10^{-3}$	$1-10^{-5}$	$1-10^{-5}$	$1-10^{-5}$	$1-10^{-5}$	$1-10^{-5}$	10^{-5}
Solver	PFODE	PFODE	PFODE	PFODE	PFODE	DDPM	DDPM
Solver accuracy of $\mathbf{s}_{\boldsymbol{\theta}}$	1st-order	2nd-order	2nd-order	1st-order	2nd-order	1st-order	1st-order
Solver type of $\mathbf{s}_{\boldsymbol{\theta}}$	RK45	Heun	Heun	RK45	Heun	Euler (DDPM)	Euler (DDPM)
Solver accuracy of $\mathbf{c}_{\boldsymbol{\phi}}$	1st-order	1st-order	1st-order	1st-order	1st-order	1st-order	1st-order
Solver type of $\mathbf{c}_{\boldsymbol{\phi}}$	RK45	Euler	Euler	RK45	Euler	Euler	Euler
t_{mid}	0.1	0.01	0.01	0.01	0.01	0.1	0.1
NFE	138	35	35	131	71	250	250
Classifier Guidance	✗	✗	✗	✗	✗	✓	✓
EDS	✗	✗	✗	✗	✗	✓	✗
w_t^{CG}	0	0	0	0	0	Adaptive	Adaptive
w_t^{DG}	2	2	Adaptive	Adaptive	Adaptive	Adaptive	1

on 64x64 FFHQ, and we train a 32x32 classifier (and freeze it at discriminator training phase) for CIFAR-10 experiment. We use the ImageNet dataset (Russakovsky et al., 2015) to train the 32x32 classifier. We follow the identical setting of Dhariwal & Nichol (2021) to train the 32x32 classifier, except the dataset resolution. We solve the unadjusted/adjusted PFODEs with a Heun solver (Ascher & Petzold, 1998) with pre-designated timesteps that is determined by NFE. As the Heun solver is a 2nd-order numerical solver, we divide the DG weight w_t^{DG} into $w_{t,1^{\text{st}}}^{DG}$ and $t_{t,2^{\text{nd}}}^{DG}$, where $w_{t,1^{\text{st}}}^{DG}$ is for the 1st-order and $t_{t,2^{\text{nd}}}^{DG}$ is for the 2nd-order. For denoising, we construct an intermediate state

$$\tilde{\mathbf{x}}_{t-\Delta t} = \mathbf{x}_t - \Delta t\Big[f(\mathbf{x}_t, t) - \frac{1}{2}g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + w_{t,1^{\text{st}}}^{DG}\mathbf{c}_{\boldsymbol{\phi}_\infty})(\mathbf{x}_t, t)\Big].$$

Then, we denoise $\mathbf{x}_t$ by

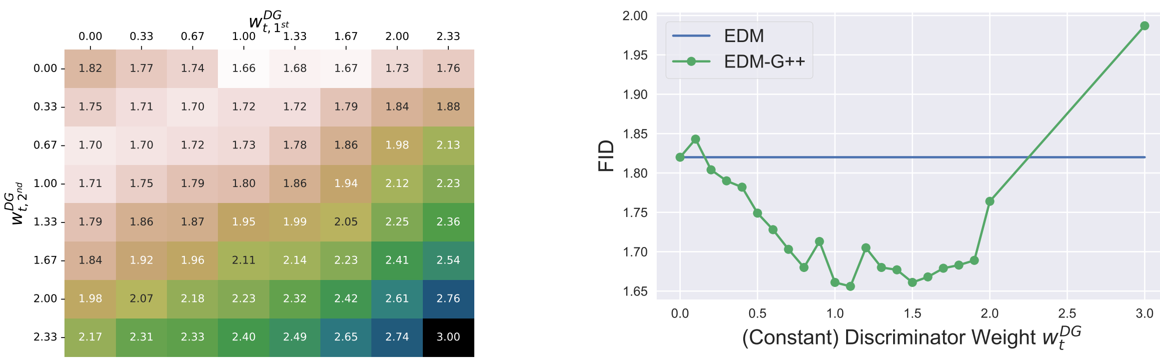
$$\begin{aligned}\mathbf{x}_{t-\Delta t} = \mathbf{x}_t - \Delta t\Big[&\frac{1}{2}\Big(\mathbf{f}(\mathbf{x}_t, t) - \frac{1}{2}g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + w_{t,1^{\text{st}}}^{DG}\mathbf{c}_{\boldsymbol{\phi}_\infty})(\mathbf{x}_t, t)\Big) \\ &+ \frac{1}{2}\Big(\mathbf{f}(\tilde{\mathbf{x}}_{t-\Delta t}, t-\Delta t) - \frac{1}{2}g^2(t-\Delta t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + w_{t,2^{\text{nd}}}^{DG}\mathbf{c}_{\boldsymbol{\phi}_\infty})(\tilde{\mathbf{x}}_{t-\Delta t}, t-\Delta t)\Big)\Big]\end{aligned}$$



(a) Ablation of $w^{DG}_{t,1^{\text{st}}}$ and $w^{DG}_{t,2^{\text{nd}}}$

(b) Detailed ablation of $w^{DG}_{t,1^{\text{st}}}$

Figure 21: Study of Heun solver for EDM on unconditional CIFAR-10.



(a) Ablation of $w^{DG}_{t,1^{\text{st}}}$ and $w^{DG}_{t,2^{\text{nd}}}$

(b) Detailed ablation of $w^{DG}_{t,1^{\text{st}}}$

Figure 22: Study of Heun solver for EDM on conditional CIFAR-10.

$$= \mathbf{x}_t - \Delta t \bigg[\frac{1}{2} \Big(\mathbf{f}(\mathbf{x}_t, t) + \mathbf{f}(\tilde{\mathbf{x}}_{t-\Delta t}, t - \Delta t) \Big) - \frac{1}{4} \Big(g^2(t) \mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t) + \mathbf{s}_{\boldsymbol{\theta}_\infty}(\tilde{\mathbf{x}}_{t-\Delta t}, t - \Delta t) \Big)$$
$$- \frac{1}{4} \Big(w^{DG}_{t,1^{\text{st}}} g^2(t) \mathbf{c}_{\boldsymbol{\phi}_\infty}(\mathbf{x}_t, t) + w^{DG}_{t,1^{\text{nd}}} g^2(t - \Delta t) \mathbf{c}_{\boldsymbol{\phi}_\infty}(\tilde{\mathbf{x}}_{t-\Delta t}, t - \Delta t) \Big) \bigg].$$

Figure 21-(a) illustrates the FID heatmap with respect to $w^{DG}_{t,1^{\text{st}}}$ and $w^{DG}_{t,2^{\text{nd}}}$. It shows that the effect of $w^{DG}_{t,1^{\text{st}}}$ is inverse proportional to that of $w^{DG}_{t,2^{\text{nd}}}$. Also, the line $w^{DG}_{t,1^{\text{st}}} + w^{DG}_{t,2^{\text{nd}}} = 2$ performs the best, which is consistent to our intuition. In particular, we emphasize that the best performance happens at $(w^{DG}_{t,1^{\text{st}}}, w^{DG}_{t,2^{\text{nd}}}) = (1.67, 0)$, which implies that DG does not has to be applied on the 2^{nd}-order correction stage of the Heun solver. This reduces the computational burden of calculating $\mathbf{c}_{\boldsymbol{\phi}_\infty}(\tilde{\mathbf{x}}_{t-\Delta t}, t - \Delta t)$ at every denoising step. Figure 21-(b) shows the detailed abalation with respect to $w^{DG}_{t,1^{\text{st}}}$ with $w^{DG}_{t,2^{\text{nd}}} = 0$. From Figure 21-(b), we set $w^{DG}_{t,1^{\text{st}}} = 2$ and $w^{DG}_{t,2^{\text{nd}}} = 0$ by default for the Heun solver. From Figure 21, we denote $w^{DG}_{t,1^{\text{st}}}$ by w^{DG}_t if no confusion arises.

Similar to the unconditional CIFAR-10, conditional CIFAR-10 has a powerful FID improvement. The optimal weight strategy is at $(w^{DG}_{t,1^{\text{st}}}, w^{DG}_{t,2^{\text{nd}}}) = (1, 0)$ from Figure 22-(a), and it has FID gain of 1.66. Figure 22-(b) ablates $w^{DG}_{t,1^{\text{st}}}$ with a fixed $w^{DG}_{t,2^{\text{nd}}} = 0$. The best performance is 1.66 when $w^{DG}_{t,1^{\text{st}}} = 1.1$. However, if we give $w^{DG}_{t,1^{\text{st}}} = 2$ for samples with likelihood ratio less than 0 in every odd denoising steps and apply $w^{DG}_{t,1^{\text{st}}} = 1$ otherwise, we get a better FID of 1.64. For such samples with density-ratio less than 0 in the odd steps, we also make $S_{churn} = 4$ to give small stochasticity to avoid local optimum points. There might be better hyperparameter settings because $S_{churn} = 4$ is set manually without thorough investigation. We call this approach as Adaptive in Table 8.

Table 9: Performance on ImageNet 256x256 with ADM.

Model	CG	DG	EDS (Zheng et al., 2022a)	FID↓	sFID↓	IS↑	Prec↑	Rec↑	F1↑
Validation Data	-	-	-	1.68	3.67	232.21	0.75	0.66	0.70
ADM (Dhariwal & Nichol, 2021)	✗	✗	✗	10.94	6.02	100.98	0.69	**0.63**	0.66
ADM-G++ (cfg=0.10)	✓	✓	✓	**4.45**	**5.38**	**190.71**	**0.76**	0.60	**0.67**
ADM-G (cfg=1.50)	✓	✗	✗	4.59	5.25	186.70	0.82	0.52	0.64
ADM-G (cfg=0.75)	✓	✗	✓	4.01	5.15	217.25	0.82	0.53	0.64
ADM-G++ (cfg=0.25)	✓	✓	✓	3.73	5.03	204.49	0.78	**0.59**	**0.67**
ADM-G++ (cfg=0.75)	✓	✓	✓	**3.18**	**4.53**	255.74	0.84	0.53	0.65

Table 10: Study on w_t^{DG} and w_t^{CG} in ADM-G++ on ImageNet.

Cases	Implications	$w_{t\geq t_0}^{DG}$	$w_{t<t_0}^{DG}$	$w_{t\geq t_0}^{CG}$	$w_{t<t_0}^{CG}$	t_0	EDS	Scaling	FID	sFID	IS	precision	recall
(a)	+ DG	0	0	1.5	1.5	0	✗	✗	4.59	5.25	186.70	0.82	0.52
		1	1	1.5	1.5	0	✗	✗	4.18	4.81	199.94	0.82	0.53
(b)	+ EDS to CG	0	0	1.5	1.5	0	✗	✗	4.59	5.25	186.70	0.82	0.52
		0	0	0.75	0.75	0	✓	✗	4.01	5.15	217.25	0.82	0.53
(c)	+ DG on CG-EDS	0	0	0.75	0.75	0	✓	✗	4.01	5.15	217.25	0.82	0.53
		1	1	0.75	0.75	0	✓	✗	3.69	4.75	215.32	0.82	0.54
(d-1)	+ Adaptive CG	1	1	0.75	0.75	0	✓	✗	3.69	4.75	215.32	0.82	0.54
		1	1	0.75	1.5	650	✓	✗	3.42	4.62	239.64	0.82	0.53
(d-2)	+ Adaptive DG	1	1	0.75	1.5	650	✓	✗	3.42	4.62	239.64	0.82	0.53
		1	1/0.75	0.75	1.5	650	✓	✓	3.18	4.53	255.74	0.84	0.53
(e)	Final Results	0	0	0.75	1.5	650	✓	✗	3.93	5.03	252.56	0.85	0.49
		1	1/0.75	0.75	1.5	650	✓	✓	3.18	4.53	255.74	0.84	0.53
		1/3	1/0.75	0.25	1.5	650	✓	✓	3.73	5.03	204.49	0.78	0.59
		1/3	1/0.75	0.25	1.5	600	✓	✓	4.66	5.52	183.31	0.76	0.61
		1/5	1/0.5	0.1	2	600	✓	✓	4.45	5.38	190.71	0.76	0.60

Adaptive DG is also effective on FFHQ. With $(w_{t,1^{\text{st}}}^{DG}, w_{t,2^{\text{nd}}}^{DG}) = (2, 0)$, EDM-G++ performs 2.04 in FID, but it drops to 1.98 if we apply the Adaptive DG weight of which adaptive strategy is identical to the conditional CIFAR-10 case.

It is worth to note that the score checkpoint of EDM and classifier checkpoint of ADM are trained under different diffusion strategies. The use of distinctive SDEs leads merging two pre-trained models in one sampler being infeasible. To clarify the difference of diffusion mechanisms, we define Weighted VE (WVE) SDE by $\mathbf{x}_t \sim \mathcal{N}(\mathbf{x}_0, \sigma_{WVE}^2(t))$ with $\sigma_{WVE}(t) = \big(\sigma_{min}^{\frac{1}{\rho}} + t(\sigma_{max}^{\frac{1}{\rho}} - \sigma_{min}^{\frac{1}{\rho}})\big)^{\rho}$ for $\sigma_{max} = 80$ and $\sigma_{min} = 0.002$, which is introduced in Karras et al. (2022). On the other hand, Ho et al. (2020) introduce Linear VP (LVP) with a linear scheduling of $\beta(t) = \beta_{min} + t(\beta_{max} - \beta_{min})$ and Nichol & Dhariwal (2021) propose Cosine VP (CVP) with a cosine scheduling of $\beta(t)$. All of EDM checkpoints are trained under WVE, and the classifier checkpoints are trained with LVP for 32x32 and CVP for 64x64. For the brevity, we only consider the case of LVP.

The key to merging two checkpoints from different diffusion strategies is observing that VE/VP-style SDEs are indeed equivalent under scale translations. Concretely, suppose p_t^{VE} and p_t^{VP} are the marginal densities of VE/VP SDEs, respectively. Then, it satisfies that $p_t^{VE}(\mathbf{x}_t) = p_{\tau(t)}^{VP}(\nu_{\tau(t)}\mathbf{x}_t)$ for

$$\begin{aligned} \tau &= \frac{-\beta_{min} + \sqrt{\beta_{min}^2 + 2(\beta_{max} - \beta_{min}) \log 1 + \sigma_{WVE}^2(t)}}{\beta_{max} - \beta_{min}}, \\ \nu_{\tau(t)} &= e^{-\frac{1}{2}\int_0^{\tau(t)} \beta(s) \,\mathrm{d}s}. \end{aligned} \tag{8}$$

With this relations, we put $(\nu_{\tau_{t_i}}\mathbf{x}_{t_i}, \tau_{t_i}) = (\nu_{\tau(t_i)}\mathbf{x}_{t_i}, \tau(t_i))$ to the discriminator at the i-th denoising step of our sampler in our implementation. We reflect this actual implementation in Algorithm 2.

For Soft Truncation-G++ with CelebA, we utilize the pre-trained score model from Kim et al. (2022b) at `https://github.com/kim-dongjun/soft-truncation` and the 64x64 pre-trained classifier model of ADM. We solve the unadjusted/adjusted PFODEs with explicit Runge-Kutta solver of order 5(4) (Dormand & Prince, 1980). Similar to the aforementioned checkpoint mismatch issue, we transform time to align the pre-trained classifier on CVPSDE and the pre-trained score on LVPSDE.

For ADM-G++, we apply the pre-trained checkpoint from Dhariwal & Nichol (2021) at `https://github.com/openai/guided-diffusion`. We exclude the upsampling method from the comparison baseline in order to solely

(a) Discriminator epoch ablation

(b) Precision/Recall trade-off

Figure 23: Ablation study of ADM-G++ on ImageNet 256x256.

compare models synthesized in the same dimension. In this case, as drawing 1,281,167 samples from ADM is too expensive given our computational budget, we train our discriminator with 400,000 samples. Given our budget, it took nearly 1 day to train 2 epochs. In total, we train 10 epochs, and still we achieve SOTA performance with such a small training budget. Instead of PFODE, we sample data in the same way of DDPM (Ho et al., 2020).

Table 10 shows experimental results of ADM in the ImageNet dataset. Cases of (a), (c), and (e) demonstrate the efficacy of the discriminator guidance. Cases of (b) and (d-1,2) are for achieving the SOTA performance with the discriminotor guidance. For (d-2), we multiply to DG weight by the norm ratio of the classifier guidance and the discriminator guidance, so to balance the guidance scale. Also, for the samples of likelihood ratio less than 0 in every odd denoising steps, we set $w_{t<t_0}^{DG}$ by 0.75. Otherwise, $w_{t<t_0}^{DG}=1$. In (e), we apply Adaptive CG/DG to report the performance of ADM-G++. As in (d-2), 1/3 of (e) experiment represents that we boost the DG weight by a factor of 3 for samples of density-ratio less than 0 in every odd denoising steps.

Figure 23-(a) shows the IS-recall curve by discriminator training epochs. We fix all the other hyperparameters except the discriminator network. Figure 23-(a) illustrates that the discriminator guidance has an effect of increasing the recall metric, which implies that a well-trained discriminator facilitates a diverse generation. Due to the diverse generation, the IS metric is decreased. Similarly, Figure 23-(b) shows that ADM-G++ significantly improves the sample diversity with a sacrifice in the precision metric.

For DiT-G++, we apply the pre-trained checkpoint from Peebles & Xie (2022) at `https://github.com/facebookresearch/DiT`. We draw 1,281,167 number of samples from DiT-XL/2-G to train the discriminator, with the classifier guidance scale of 1.5. In sample generation of DiT-XL/2-G++, we choose $w_t^{CG}=1.25$ for $t<t_0$ with $t_0=200$ and $w_t^{CG}=3.0$ otherwise. We use $w_t^{DG}=1$ and do not ablate the weight scale.

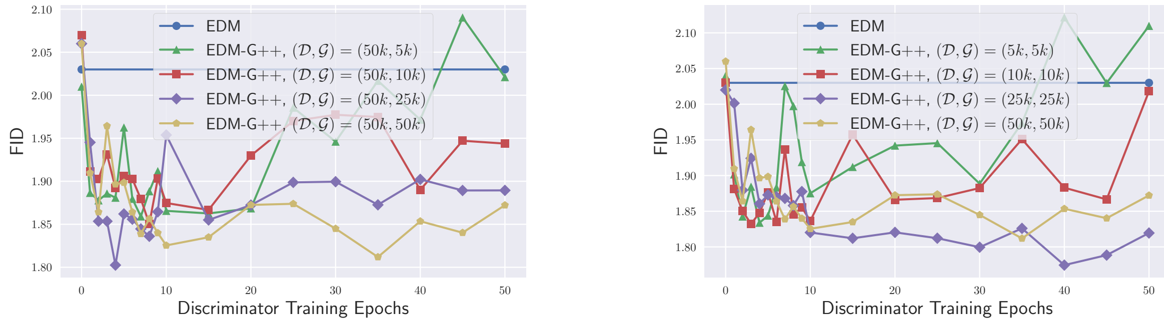
D.2. Further Score Training

D.3. More Ablation Studies

D.3.1. Discriminator Training

Ablation of # Training Data Figure 25 shows the ablation study on the number of training data on CIFAR-10. The discriminator guidance requires many samples from $p_{\boldsymbol{\theta}_\infty}$ to learn the likelihood ratio, so this could arise the computational issue in some cases. For that, we train the discriminator with the full set of real data and a partial set of sample data in Figure 25-(a). In other words, the case of $(\mathcal{D},\mathcal{G})=(50k,25k)$ represents for the case of the full use of real data as $\mathcal{D}$ and the half number of sample data as $\mathcal{G}$ in Algorithm 1. Figure 25-(b), on the other hand, uses the same number of real data for discriminator training. For both experiments, the discriminator suffers from the overffiting issue as the number of generated data decreases.

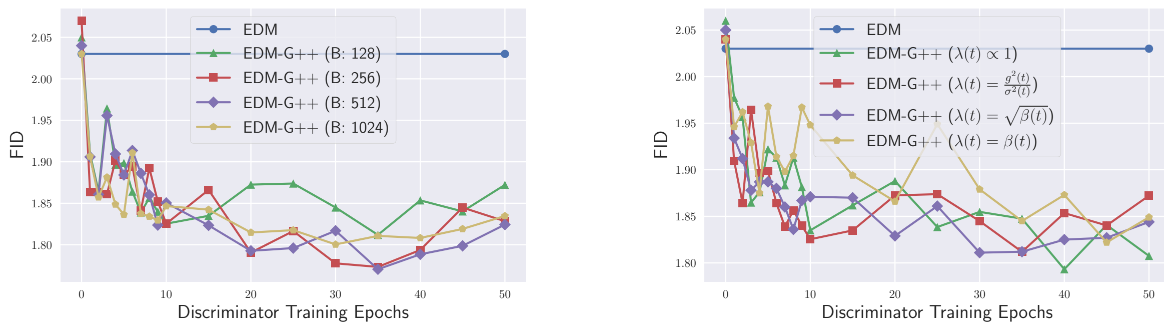
Analogous to Figure 25-(a), Figure 25-(b) shows the another ablation study on the number of training data. The situation is a bit different: in this case, we also reduce the size of real data as well as the sample data. Comparing Figure 25-(b) with Figure 25-(a), it is generally not recommendable in any cases. The overfitting arises faster, leading the performance gets worsened faster.



(a) Full data and partial sample ablation

(b) Partial data and partial sample ablation

Figure 25: Study of EDM on CIFAR-10 with respect to the number of training data. (a) is the case with full real data $\mathcal{D}$ and partial sample data $\mathcal{G}$ and (b) is the case with partial $\mathcal{D}$ and partial $\mathcal{G}$.



(a) Batch size ablation

(b) Temporal weight λ ablation

Figure 26: Study of EDM on CIFAR-10 with respect to (a) the number of training batch size and (b) the temporal weight λ.

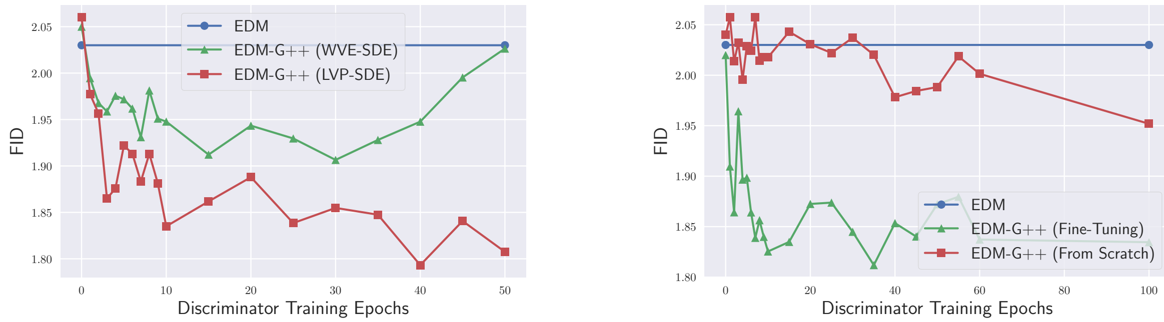
Figure 24: Further score training of a pre-trained score model.

Ablation of # Batch Size Figure 26-(a) illustrates the ablation study on the number of training batch size on CIFAR-10. Figure 26-(a) implies that the number of batch size is not a crucial factor for final performance of discriminator guidance.

Ablation of Temporal Weight λ Figure 26-(b) shows indistinguishable FID by the variance of temporal weighting function λ for $\mathcal{L}_{\phi}$ of Eq. (4). We experiment with a uniform distribution $\lambda(t) \propto 1$, an importance-weighted (Song et al., 2021) distribution $\lambda(t) = \frac{g^2(t)}{\sigma^2(t)}$ that is one of a common practice, and two β-related distributions of $\lambda(t) = \sqrt{\beta(t)}$ and $\lambda(t) = \beta(t)$. We train the discriminator with LVP-SDE. Figure 26-(b) demonstrates that there is no significant difference between the choice of λ.

Ablation of SDE Figure 27-(a) empirically shows that using LVP-SDE as discriminator SDE performs better than using WVE-SDE. This indicates an important fact: while score training is beneficial with WVE-SDE, discriminator training best fits with LVP-SDE. We, therefore, train the discriminator with LVP-SDE as default.

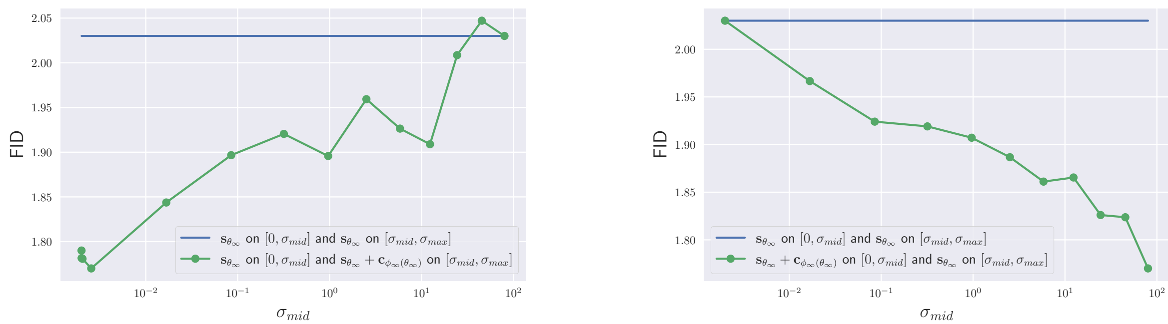
Ablation of Parameter Initialization Figure 27-(b) shows that the fine-tuning performs strictly better than the training from scratch. For the setting of fine-tuning, we fix the pre-trained classifier parameters, and only train the shallow U-Net for discriminator. In contrast, we train all the parameters including that of the latent extractor, and we set fine-tuning as default in our paper.



(a) Forward SDE ablation

(b) Initialization ablation

Figure 27: Study of EDM on CIFAR-10 with respect to (a) the discriminator diffusing method and (b) the discriminator parameter initialization.



(a) Minimum DG scale ablation

(b) Maximum DG scale ablation

Figure 29: Study of EDM on CIFAR-10 for optimal scales to apply DG: (a) minimum scale, (b) maximum scale.

Additionally, we find that Exponential Moving Average (EMA) has nearly no effect on the discriminator training, and we do not apply EMA of our discriminator training.

D.3.2. After Discriminator Training

Ablation of Discriminator Training in Latent Diffusion on CIFAR-10 Figure 28 additionally studies LSGM on CIFAR-10. Figure 28 shows FID by discriminator training. The performance saturates after nearly 300 epochs, and it is because we do not use a pre-trained latent extractor for LSGM. Kim et al. (2022a) clarify that latent diffusion models in general have no diffusion process in the pixel space as long as models use auto-encoder structure to map data to latent. We could detour this problem by defining the diffused data as a decoded diffused latent $\mathbf{x}_t = Dec(\mathbf{z}_t)$, but there is no pre-trained classifier for generic diffusion strategy nor latent diffusion space. We leave it as future work in this direction. In our implementation, we train a U-Net encoder from scratch. This takes a long time to saturate, but FID improves immediately after the discriminator training.

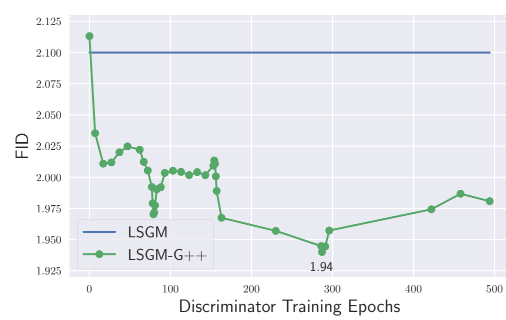


Figure 28: Discriminator epoch ablation.

Ablation of optimal scales on CIFAR-10 Figure 29 studies the minimum/maximum diffusion scales to apply DG. For the experiment in Figure 29-(a), we divide the diffusion scales by σ_{mid}, and denoise with the reverse-time generative process

$$\mathrm{d}\mathbf{x}_t = \left[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + \mathbf{c}_{\boldsymbol{\phi}_\infty})(\mathbf{x}_t, t)\right] \mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

on the large range $[\sigma_{mid}, \sigma_{max}]$, and denoise with

$$\mathrm{d}\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

on the small range $[0, \sigma_{mid}]$. With this ablation study, we could find the optimal minimum stopping scale to apply the diffusion scale. Figure 29-(a) illustrates that the optimal stopping scale near $\sigma_{mid}^* = 0.0025$. This strictly positive optimal scale implies that either the score adjustment becomes inaccurate or the discretization error matters at the range of extremely low scale. In the likelihood ratio community, *density-chasm problem* is a well-known problem (Rhodes et al., 2020; Kato & Teshima, 2021) that depicts a poor density-ratio estimation when the two densities have distinctive supports. It arises from the training-test mismatch: the discriminator perfectly classifies real/fake, but the middle area between the real data support and the fake data support remain unoptimized. Therefore, the density-chasm problem is one of the reason for such a strictly positive optimal scale.

Figure 29-(b) studies the maximum diffusion scale to apply DG. Analogous to the experimental setting of Figure 29-(a), we divide the diffusion scales by σ_{mid}, and denoise with the reverse-time generative process

$$\mathrm{d}\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)\mathbf{s}_{\boldsymbol{\theta}_\infty}(\mathbf{x}_t, t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

on the large range $[\sigma_{mid}, \sigma_{max}]$, and denoise with

$$\mathrm{d}\mathbf{x}_t = \big[\mathbf{f}(\mathbf{x}_t, t) - g^2(t)(\mathbf{s}_{\boldsymbol{\theta}_\infty} + \mathbf{c}_{\boldsymbol{\phi}_\infty})(\mathbf{x}_t, t)\big]\,\mathrm{d}\bar{t} + g(t)\,\mathrm{d}\bar{\mathbf{w}}_t,$$

on the small range $[0, \sigma_{mid}]$. Contrastive to the minimum scale ablation study, the larger the scale DG applied, the better the performance. This means that the actual effect of DG lies in constructing the global shape of the generation, rather than denoising fine-details. In the community of diffusion models, there are only a few works that systematically divide the context generation ability and fine-detail capturing ability of diffusion models. Figure 29 clarify that DG effectively adjusts the context generation ability of diffusion models, rather than cleansing the fine-dust in images.

Ablation of NFE on FFHQ Figure 30 illustrates FID by NFE after discriminator training. For visualization purpose, we select the best hyperparameters to experiment with, except NFE. Similar to NFE ablation on CIFAR-10, the discriminator guidance keeps enhancing FID throughout NFEs.

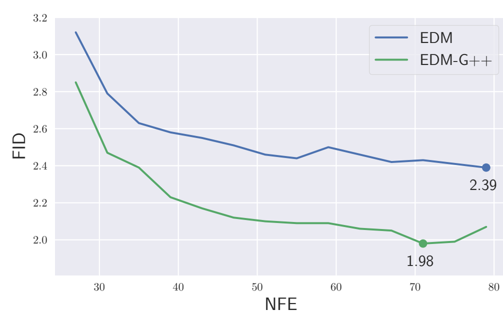


Figure 30: NFE ablation on FFHQ.

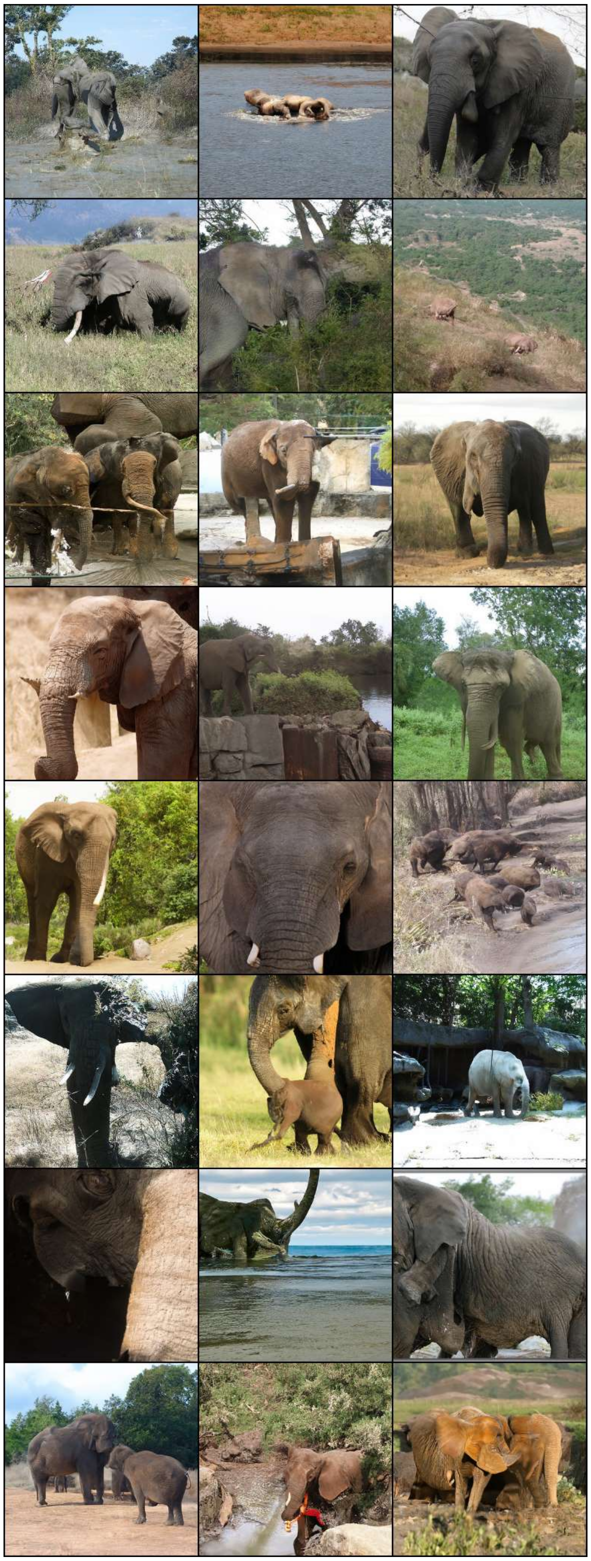
D.4. Uncurated Samples

Figure 31 compares ADM-G++ (cfg=0.10) with the vanilla ADM to illustrate how sample fidelity is improved while keeping the sample diversity. ADM-G++ (cfg=0.10) performs FID of 4.45 and recall of 0.60, and ADM performs FID of 10.94 and recall of 0.63.

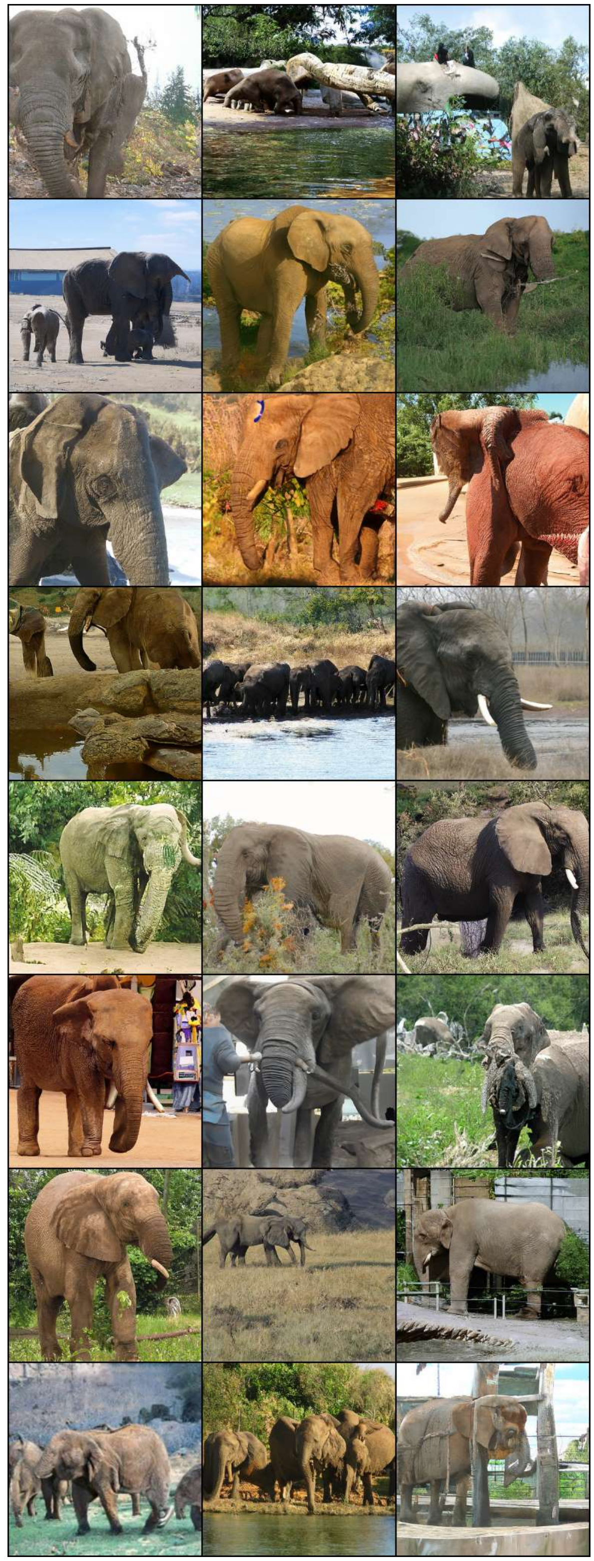
Figure 32 compares ADM-G++ (cfg=0.75) with ADM-G (cfg=1.50). These figures show the discriminator guidance is effective in high-dimensional dataset.

Figures 33 and 34 show uncurated samples from unconditional/conditional CIFAR-10 with EDM-G++.

Figure 35 shows the uncurated samples from I2I translation task.



(a) ADM (FID 10.94 recall 0.63)



(b) ADM-G++ (FID 4.45 recall 0.60)

Figure 31: Uncurated random samples from african elephant class (386) (a) ADM with poor FID (10.94) and good recall (0.63), (b) ADM-G++ with good FID (4.45) and good recall (0.60).

(a) ADM-G (FID 4.59 recall 0.52)

(b) ADM-G++ (FID 3.18 recall 0.53)

Figure 32: Uncurated random samples from african elephant class (386) (a) ADM-G with good FID (4.59) and poor recall (0.52), (b) ADM-G++ with FID (3.18) and moderate recall (0.53).

Figure 33: Uncurated random samples from EDM-G++ on unconditional CIFAR10 (FID: 1.77).

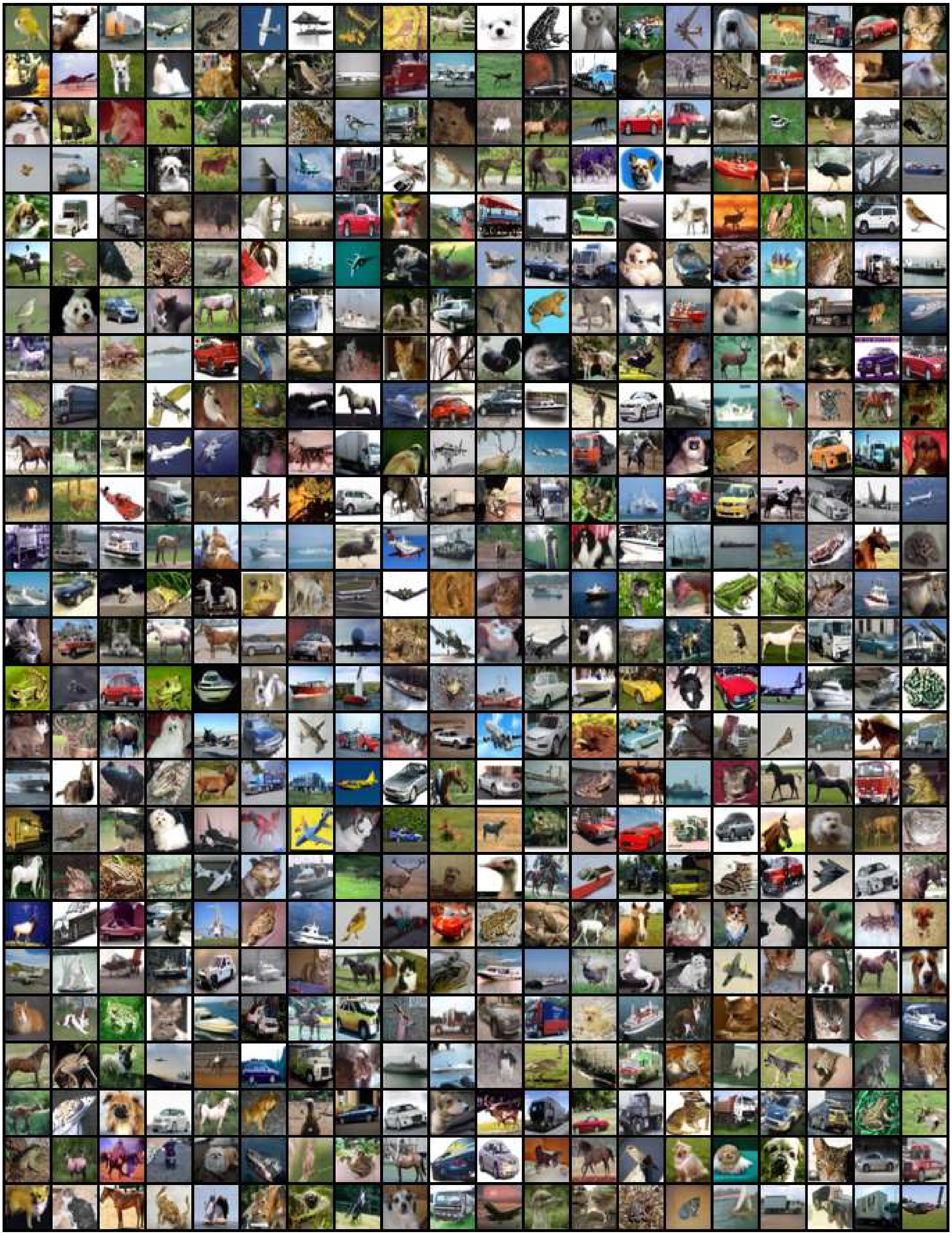


Figure 34: Uncurated random samples from EDM-G++ on conditional CIFAR10 (FID: 1.64).

(a) Cat (Source) (b) SDEdit (FID: 74.02) (c) SDEdit + DG (FID: 61.92)

Figure 35: Uncurated random translated samples from (a) source cat, (b) SDEdit (FID: 74.02, L2: 49.22, PSNR: 19.21, SSIM: 0.42), and (c) SDEdit + DG with $w_t^{DG} = 8$ (FID: 61.92, L2: 50.62, PSNR: 18.94, SSIM: 0.41).