PaGoDA: Progressive Growing of a One-Step Generator from a Low-Resolution Diffusion Teacher

Dongjun Kim[*,†]
Stanford University
CA, USA
dongjun@stanford.edu

Chieh-Hsin Lai[*]
Sony AI
Tokyo, Japan
chieh-hsin.lai@sony.com

Wei-Hsiang Liao
Sony AI

Yuhta Takida
Sony AI

Naoki Murata
Sony AI

Toshimitsu Uesaka
Sony AI

Yuki Mitsufuji
Sony AI, Sony Group Corporation

Stefano Ermon
Stanford University

Abstract

The diffusion model performs remarkable in generating high-dimensional content but is computationally intensive, especially during training. We propose **P**rogressive **G**rowing **o**f **D**iffusion **A**utoencoder (**PaGoDA**), a novel pipeline that reduces the training costs through three stages: training diffusion on downsampled data, distilling the pretrained diffusion, and progressive super-resolution. With the proposed pipeline, PaGoDA achieves a $64\times$ reduced cost in training its diffusion model on $8\times$ downsampled data; while at the inference, with the single-step, it performs state-of-the-art on ImageNet across all resolutions from 64×64 to 512×512, and text-to-image. PaGoDA's pipeline can be applied directly in the latent space, adding compression alongside the pre-trained autoencoder in Latent Diffusion Models (e.g., Stable Diffusion). The code is available at https://github.com/sony/pagoda.

1 Introduction

Diffusion Models (DM) [1, 2], which generate content through gradual denoising, have recently achieved high fidelity in high-dimensional generation [3, 4]. While slow sampling has been improved by distilling trained DMs into single-step generators [5–7], DMs remain computationally intensive, especially at high resolutions, requiring substantial data and GPU resources, thereby limiting large-scale training to a few organizations [8, 9]. This highlights the need for a more efficient pipeline to reduce both training and inference costs while maintaining the quality.

To address these challenges, we present **P**rogressive **G**rowing **o**f **D**iffusion **A**utoencoder (**PaGoDA**), a novel pipeline that significantly reduces costs while achieving competitive quality with one-step sampling. PaGoDA is built on a simple yet effective idea: while diffusion distillation [6] is typically treated as a final stage of the whole pipeline, we explore to have one more stage for the super-resolution after diffusion distillation. This approach led us to design PaGoDA with three distinct stages as below.

[*]Equal contribution
[†]This work was partially done during an internship at Sony AI.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

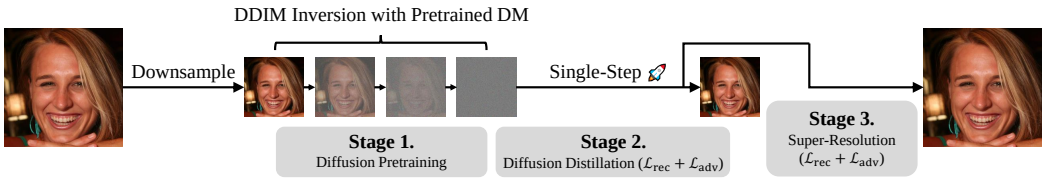


Figure 1: Pipeline overview. PaGoDA deterministically encodes with downsampling followed by DDIM inversion, and constructs its decoder in a progressively growing manner.

PaGoDA's Proposed Training Pipeline

Stage 1. (Pretraining) Train a DM on downsampled data.

Stage 2. (Distillation) Distill the trained DM with DDIM inversion to a one-step generator.

Stage 3. (Super-Resolution) Progressively expand the generator for resolution upsampling.

By adding Stage 3 for the super-resolution after the distillation phase, our approach gains a key advantage: training DM on a low-dimensional, downsampled space rather than directly in the desired high-dimensional space. This dimensional reduction substantially lowers the computational demands of diffusion pretraining by orders of magnitude. For example, an 8×8 downsampling rate reduces the training computation by a factor of $64\times$. Moreover, the computational costs for the distillation and super-resolution stages are relatively minimal compared to the initial diffusion pretraining, making our pipeline highly efficient in terms of overall computation.

Figure 1 provides an overview of our pipeline. We begin with DM trained at base resolution, and generate a dataset of base-resolution data-latent pairs $(\mathbf{x}, \mathbf{z})$, where $\mathbf{x}$ is real data and $\mathbf{z}$ is the latent representation of $\mathbf{x}$, obtained by DDIM inversion [10]. In Stage 2, we train a decoder to map $\mathbf{z}$ back to $\mathbf{x}$, completing the diffusion distillation [6]. In Stage 3, we add ResNet blocks [11] to enhance sample resolution and progressively train these newly added upscaling networks, as visualized in Figure 2. The novel use of DDIM inversion in the distillation process, first introduced in PaGoDA, enables the decoder to be trained with the high-frequency signal from the real data at Stage 3. This integration of DDIM inversion establishes strong connections across stages, creating a cohesive and unified framework.

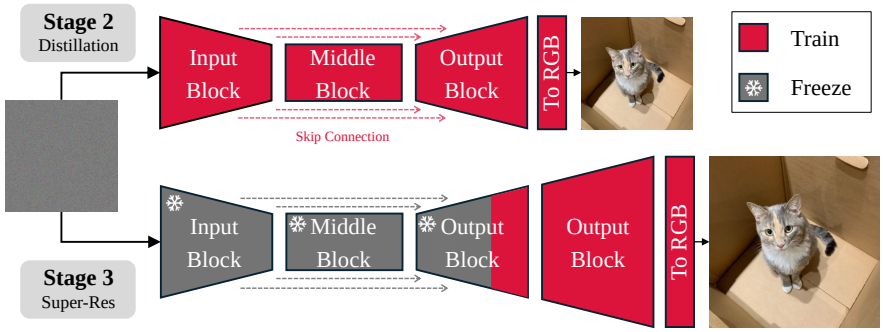


Figure 2: (Top) At Stage 2, PaGoDA learns the one-step generator at a base resolution. (Down) At Stage 3, PaGoDA progressively learns for super-resolution by adding additional network blocks.

In our experiments, we employed the progressively growing generator to upsample from the pre-trained diffusion model's 64×64 resolution to generate samples at 512×512 resolution. Notably, PaGoDA achieved state-of-the-art (SOTA) Fréchet Inception Distances (FID) [12] on ImageNet across all resolutions from 64×64 to 512×512. Additionally, we demonstrated PaGoDA's effectiveness in addressing inverse problems and facilitating controllable generation. However, PaGoDA's potential extends beyond its current application. As PaGoDA being a dimensional reduction technique that operates independently of Latent Diffusion Models (LDM) [3], PaGoDA could be directly applied into the latent space as-is, offering the possibility of further gain on training computes. We leave this exploration as a promising avenue for future research.

2 Preliminary

DM [1] samples from the data distribution p_{data} through an iterative denoising process, beginning from a Gaussian prior distribution p_{prior}. This denoising process attempts to reverse [2] a forward diffusion process. If the forward process is defined by $\mathrm{d}\mathbf{x}_t = \sqrt{2t}\,\mathrm{d}\mathbf{w}_t$ [13], the deterministic counterpart of the denoising (generation) process, known as the probability flow ordinary differential equation (PF-ODE) [2], or DDIM [10], is expressed as

$$\frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = -t\nabla \log p_t(\mathbf{x}_t) \approx -t s_{\boldsymbol{\phi}_0}(\mathbf{x}_t, t),$$

where $s_{\boldsymbol{\phi}_0}(\mathbf{x}_t, t)$ is a neural approximation of $\nabla \log p_t(\mathbf{x}_t)$. Consequently, (deterministic) sample generation from DM is equivalent to solving the PF-ODE (or DDIM) along the trajectory, formally,

$$\mathbf{x}_0^{\text{DDIM}}(\mathbf{x}_T) = \mathbf{x}_T - \int_T^0 t s_{\boldsymbol{\phi}_0}(\mathbf{x}_t, t)\,\mathrm{d}t, \quad \mathbf{x}_T \sim p_{\text{prior}}.$$

Modern solvers of the PF-ODE [10, 14] have significantly accelerated sampling speed, reducing the required network evaluations from hundreds to tens. To further speed up sampling, DMs are distilled with a student model [6] $G_{\boldsymbol{\theta}} : \mathbb{R}^d \to \mathbb{R}^d$ to map from $\mathbf{x}_T$ to $\mathbf{x}_0^{\text{DDIM}}(\mathbf{x}_T)$ by minimizing

$$\mathcal{L}_{\text{dstl}}(G_{\boldsymbol{\theta}}) = \mathbb{E}_{p_{\text{prior}}(\mathbf{x}_T)}\left[\left\|\mathbf{x}_0^{\text{DDIM}}(\mathbf{x}_T) - G_{\boldsymbol{\theta}}(\mathbf{x}_T)\right\|_2^2\right]. \tag{1}$$

We call this DDIM-based approach as the *noise-to-data* distillation.

3 Progressive Growing of Diffusion Autoencoder

3.1 Stage 1: Diffusion Models Trained on Downsampled Data

Training DMs for high-dimensional data generation is primarily feasible for a limited number of well-resourced organizations, largely due to two factors: access to large-scale datasets and substantial computational resources. This centralization of model development underscores the urgent need to democratize access by significantly reducing resource demands during diffusion training. While several strategies [15, 3] have been proposed, our approach, PaGoDA, introduces a paradigm shift by training the DM at a downsampled resolution in Stage 1, rather than at the original full resolution. For instance, training on a d-dimensional downsampled resolution requires approximately 4^n times less computational budget compared to training in the full $4^n d$-dimensional space. In practical terms, when $n = 3$, this translates to training in an 8×8 downsampled space, effectively reducing training costs by a factor of $64\times$, thus making large-scale diffusion training more accessible to a broader range of researchers.

Although this paper does not extend PaGoDA's application to the LDM such as SD, training on a (say) 4×4 downsampled latent space could theoretically reduce the computational cost by $16\times$ compared to full-resolution latent training, further emphasizing PaGoDA's potential for widespread adoption. In the case of generating 1024×1024 images, PaGoDA requires training the diffusion model at only 32×32 resolution, with Stage 3 subsequently upscaling it to the full 128×128 latent space of conventional approaches [8, 9]. This progressive approach illustrates PaGoDA's effectiveness in maintaining model quality while lowering the barriers to high-resolution diffusion training.

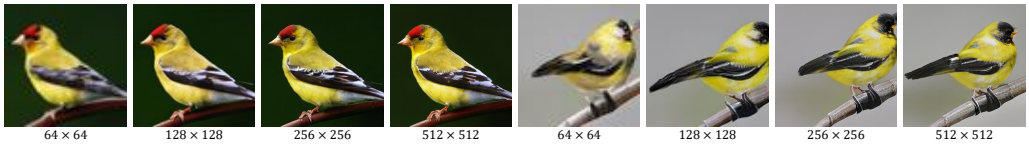
3.2 Stage 2: Diffusion Distillation on Downsampled Data with DDIM Inversion

After pretraining DM on the downsampled space, PaGoDA distills DM to a one-step generator. For distillation, PaGoDA introduces a new loss specifically designed for later usage in super-resolution at Stage 3. In particular, we propose the reconstruction loss (compare it with $\mathcal{L}_{\text{dstl}}$ in Eq. 1 of Section 2)

$$\mathcal{L}_{\text{rec}}(G_{\boldsymbol{\theta}}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)}\left[\left\|\mathbf{x}_0 - G_{\boldsymbol{\theta}}\left(\mathbf{x}_T^{\text{DDIM}^{-1}}(\mathbf{x}_0)\right)\right\|_2^2\right], \tag{2}$$

where $\mathbf{x}_T^{\text{DDIM}^{-1}}(\mathbf{x}_0)$ is now the latent representation of $\mathbf{x}_0$, obtained from DDIM inversion, not from DDIM, i.e., the solution at time T of the PF-ODE starting from $\mathbf{x}_0$ in time forward, defined by

$$\mathbf{x}_T^{\text{DDIM}^{-1}}(\mathbf{x}_0) := \mathbf{x}_0 - \int_0^T t s_{\boldsymbol{\phi}_0}(\mathbf{x}_t, t)\,\mathrm{d}t.$$



(a) Resolution Jump w/ Recon Loss (b) Resolution Jump w/o Recon Loss

Figure 3: Effect of the reconstruction loss in Stage 3. Without the reconstruction loss, the object moves at each resolution jump.

Distillation using $\mathcal{L}_{\text{rec}}$ maps latent representations to real data, following a *data-to-noise* distillation approach. While this method has the potential to improve real data alignment compared to the traditional noise-to-data approach in Eq. 1, we observe a decline in generation quality over iterations. This decline stems from the prior hole problem [16], where the generator's input, $\mathbf{x}_T^{\text{DDIM}^{-1}}$, derived from limited real data, fails to cover the entire prior manifold, leaving certain regions unexplored.

A straightforward strategy like early stopping could alleviate this issue, but it restricts the use of Exponential Moving Average (EMA) in Stage 2. To fundamentally resolve this problem, we propose a solution that maintains generation quality even during prolonged training. In Section 3.4, we provide optimal and stability analysis of Stage 2, guaranteeing that our proposal is stable across training iterations. The key challenge is effectively covering the prior manifold, which we address by introducing an auxiliary adversarial loss, as defined below:

$$\mathcal{L}_{\text{adv}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\log D_{\boldsymbol{\psi}}(\mathbf{x})\big] + \mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\Big[\log\Big(1 - D_{\boldsymbol{\psi}}\big(G_{\boldsymbol{\theta}}(\mathbf{z})\big)\Big)\Big], \tag{3}$$

Here, $D_{\boldsymbol{\psi}}$ is a discriminator that classifies the real and fake samples by maximizing the adversarial loss, and $p_{\text{prior}}(\mathbf{z})$ is the prior distribution.

The second term in $\mathcal{L}_{\text{adv}}$, which involves $G_{\boldsymbol{\theta}}(\mathbf{z})$ with $\mathbf{z}$ sampled from the prior, ensures that the decoder is exposed to the entire support of the prior distribution during training. Overall, we train PaGoDA with the mini-max optimization of the following combined objective:

$$\min_{G_{\boldsymbol{\theta}}} \max_{D_{\boldsymbol{\psi}}} \mathcal{L}_{\text{PaGoDA}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) := \min_{G_{\boldsymbol{\theta}}} \Big[\mathcal{L}_{\text{rec}}(G_{\boldsymbol{\theta}}) + \lambda \max_{D_{\boldsymbol{\psi}}} \mathcal{L}_{\text{adv}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}})\Big]. \tag{4}$$

While PaGoDA incorporates the adversarial loss, the reconstruction loss simultaneously guides the decoder to accurately reconstruct the entire training data. This combination allows the adversarial loss to address underrepresented regions in the prior distribution effectively without compromising sample diversity. In our ImageNet experiments, we found that updating the reconstruction loss with as little as 1% of the data-latent pairs did not affect sample quality and diversity. Exploring the impact of varying the number of data-latent pairs is left as a future work.

3.3 Stage 3: Progressively Growing Decoder for Super-Resolution

Stage 3 trains the super-resolution to generate higher-dimensional data from the downsampled resolution learned in the previous stages. As illustrated in Figure 2, the resolution jump from $\mathbb{R}^d$ to $\mathbb{R}^{4^n d}$ is achieved by freezing most parameters of the distilled model from Stage 2 and training only the final layers, which is augmented with an additional upscaler network (of ResNet blocks [17]). In other words, within the base-resolution U-Net [18], we freeze its input, middle, and output blocks except for the last few layers (previously highest resolution block) during Stage 3 training. Consequently, the unfrozen latter part of the network is trained for super-resolution. We suggest to progressively increasing the resolution by a factor of $2\times$, though larger jumps by factors of $4\times$ or $8\times$ yield comparable performance.

Additionally, the last layer typically converts multi-channel (usually 128 or 256 channels) features to 3-channel RGB output. However, to minimize information loss, we retain these features and pass them directly to the next output block without converting them to 3 channels. This architectural choice, along with progressive training, is heavily inspired by Progressive Growing GAN [19].

In Stage 3, the reconstruction loss from Stage 2 is adapted as

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}_{\text{high}})}\Big[\big\|\mathbf{x}_{\text{high}} - G_{\boldsymbol{\theta}}(\mathbf{x}_T^{\text{DDIM}^{-1}}(\mathbf{x}_0))\big\|_2^2\Big],$$

where $\mathbf{x}_0 \in \mathbb{R}^d$ is the downsampled counterpart of $\mathbf{x}_{\text{high}} \in \mathbb{R}^{4^n d}$. The adversarial loss in this stage is

$$\mathbb{E}_{p_{\text{data}}(\mathbf{x}_{\text{high}})}\left[\log D_{\boldsymbol{\psi}}(\mathbf{x}_{\text{high}})\right] + \mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\left[\log\left(1 - D_{\boldsymbol{\psi}}\left(G_{\boldsymbol{\theta}}(\mathbf{z})\right)\right)\right].$$

Overall, both the reconstruction and adversarial losses are combined to guide training.

Stage 3 employs two key mechanisms to effectively capture high-frequency details while maintaining training stability. First, the reconstruction loss is applied directly to high-dimensional real data, which was not feasible with earlier noise-to-data distillation methods with Eq. 1. As illustrated in Figure 3, $\mathcal{L}_{\text{rec}}$ stabilizes the upscaling process by preventing objects from shifting across resolutions, allowing the added neural network to focus solely on upsampling. Second, the adversarial loss operates directly in high-dimensional space, enabled by the one-step generator trained in Stage 2. This generator is critical; without it, adversarial training in Stage 3 would be infeasible. As shown in Figure 4 tested on ImageNet, the adversarial loss is pivotal for achieving effective upscaling performance.

Figure 4: The adversarial loss makes PaGoDA competitive with GAN-based super-resolution models in Stage 3.

3.4 Optimality Guarantee and Training Stability of PaGoDA Pipeline

When using the conventional $\mathcal{L}_{\text{dstl}}$ for distillation, the optimal student becomes $G_{\boldsymbol{\theta}^*}(\mathbf{x}_T) = \mathbf{x}_0^{\text{DDIM}}(\mathbf{x}_T)$, meaning that the student's samples replicate those of DDIM. As a result, the student's performance is heavily dependent on the teacher's performance. Consequently, the student's generative distribution may diverge from the real data distribution, even when $\mathcal{L}_{\text{dstl}}$ is combined with adversarial loss. In contrast, by using the DDIM inversion-based reconstruction loss proposed in Stage 2, we mathematically prove in Theorem 3.1 that the optimal student's generative distribution aligns with the real data distribution. As visualized in Figure 5, our PaGoDA Stage 2 (red) achieves robust performance even with a weaker teacher, unlike traditional noise-to-data distillation loss $\mathcal{L}_{\text{dst}}$ of Eq. 1, which struggles despite the use of adversarial loss.

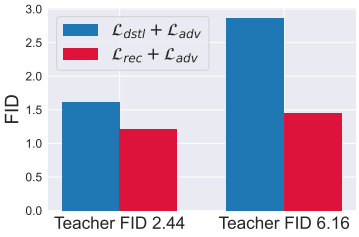


Figure 5: Comparison of $\mathcal{L}_{\text{dstl}}$ and $\mathcal{L}_{\text{rec}}$, both combined with $\mathcal{L}_{\text{adv}}$, using identical hyperparameters. $\mathcal{L}_{\text{rec}}$ shows the robust performance, also supported by Theorem 3.1.

Theorem 3.1. *Let $\lambda > 0$. Suppose $D^*(G) \in \arg\max_D \mathcal{L}_{adv}(G, D)$. If both PaGoDA's reconstruction loss and adversarial loss share a common minimizer G^*, then $p_{G^*}(\mathbf{x}) = p_{data}(\mathbf{x})$. Here, p_{G^*} is the generative distribution learned by optimizing Eq. (4).*

Additionally, Theorem 3.2 shows that PaGoDA's training is stable with the help of reconstruction loss, even with adversarial training. We empirically observe that PaGoDA can be trained effectively without many of the techniques typically used to stabilize GANs [20, 21].

Theorem 3.2. [Informal] *Let E be a fixed deterministic encoder. Suppose that at the generator's equilibria G^* of Eq. (4), $p_{G^*}(\mathbf{x}) = p_{data}(\mathbf{x})$, and $\mathbf{x} = G^*(E(\mathbf{x}))$. Then, under conditions similar to those found in the stability literature for improving GAN [22, 21], training with Eq. (4) is stable (gradient descent locally converges to its equilibria).*

We refer to Theorems B.4 and B.9 for rigorous and extended versions of Theorems 3.1 and 3.2, respectively. All proofs can be found in Appendix B.

4 PaGoDA with Classifier-Free Guidance

In this section, we integrate Classifier-Free Guidance (CFG) [23, 4] into PaGoDA for Text-to-Any generation, with a focus on Text-2-Image. Incorporating CFG alters the sample distribution, necessitating adjustments to the loss functions for Stages 2 and 3. Since previous GAN literature [24–27] has not addressed CFG integration, we introduce the classifier-free guided adversarial loss to accommodate this adaptation.

CFG guides the denoising process by adjusting the conditional score gradient $\nabla \log p_t(\mathbf{x}_t|\mathbf{c})$ into a guided score $\nabla \log p_t(\mathbf{x}_t|\mathbf{c}) + (\omega - 1)\nabla \log p(\mathbf{c}|\mathbf{x}_t)$. This adjustment leads our distillation learning target from $p_{\text{data}}(\mathbf{x}|\mathbf{c})$ to $p_{\text{data}}(\mathbf{x}|\mathbf{c}, \omega)$, defined by

$$p_{\text{data}}(\mathbf{x}|\mathbf{c}, \omega) \propto p_{\text{data}}(\mathbf{x}|\mathbf{c})^{\omega} p_{\text{data}}(\mathbf{x})^{1-\omega},$$

reflecting the influence of guidance strength ω.

4.1 Classifier-Free Guided Adversarial Loss

To describe the classifier-free adversarial loss, we first consider the loss:

$$\mathcal{L}_{\text{adv}}^{\mathbf{c},\omega}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x}|\mathbf{c},\omega)}\Big[\log D_{\boldsymbol{\psi}}(\mathbf{x}, \mathbf{c}, \omega)\Big] + \mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{c},\omega)}\Big[\log\Big(1 - D_{\boldsymbol{\psi}}(\mathbf{x}, \mathbf{c}, \omega)\Big)\Big],$$

where now both generator and discriminator incorporates ω as an additional condition [28], see Eq. (3) for the comparison. From the standard GAN argument [29], this GAN loss guarantees the optimal generator to match to the data distribution, i.e., $p_{G^*}(\mathbf{x}|\mathbf{c}, \omega) = p_{\text{data}}(\mathbf{x}|\mathbf{c}, \omega)$. Hence, the classifier-free adversarial loss could be defined by

$$\begin{aligned}\mathcal{L}_{\text{adv}}^{\text{CFG}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) &:= \mathbb{E}_{p_{\text{data}}(\mathbf{c})\pi(\omega)}\big[\mathcal{L}_{\text{adv}}^{\mathbf{c},\omega}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}})\big] \\ &= \mathbb{E}_{p_{\text{data}}(\mathbf{c})\pi(\omega)p_{\text{data}}(\mathbf{x}|\mathbf{c},\omega)}\Big[\log D_{\boldsymbol{\psi}}\big(\mathbf{x}, \mathbf{c}, \omega\big)\Big] + \mathbb{E}_{p_{\text{data}}(\mathbf{c})\pi(\omega)p_{G_{\boldsymbol{\theta}}}(\mathbf{x}|\mathbf{c},\omega)}\Big[\log\big(1 - D_{\boldsymbol{\psi}}\big(\mathbf{x}, \mathbf{c}, \omega\big)\big)\Big].\end{aligned}$$

A key challenge with $\mathcal{L}_{\text{adv}}^{\mathbf{c},\omega}$ is that sampling from $p_{\text{data}}(\mathbf{x}|\mathbf{c}, \omega)$ is generally infeasible, making it difficult to compute the first term of $\mathcal{L}_{\text{adv}}^{\text{CFG}}$. To address this issue, we leverage the Bayes formula

$$p_{\text{data}}(\mathbf{c})\pi(\omega)p_{\text{data}}(\mathbf{x}|\mathbf{c}, \omega) = p_{\text{data}}(\mathbf{x}, \mathbf{c})p(\omega|\mathbf{x}, \mathbf{c}),$$

where both representations are two different ways to decompose the joint distribution over $(\mathbf{x}, \mathbf{c}, \omega)$, with $\pi(\omega)$ being the prior distribution of the CFG scale ω. From this formula, if we could predict the guidance weight ω by observing $\mathbf{x}$ and $\mathbf{c}$, i.e., if we know $p(\omega|\mathbf{x}, \mathbf{c})$, then sampling $(\mathbf{x}, \mathbf{c}, \omega)$ from $p_{\text{data}}(\mathbf{c})\pi(\omega)p_{\text{data}}(\mathbf{x}|\mathbf{c}, \omega)$ can be alternatively achieved by: 1) sampling $(\mathbf{x}, \mathbf{c})$ from $p_{\text{data}}(\mathbf{x}, \mathbf{c})$, and 2) predicting most likely ω using $p(\omega|\mathbf{x}, \mathbf{c})$.

We approximate $p(\omega|\mathbf{x}, \mathbf{c})$ with a U-Net encoder network with 1-dimensional output, called *CFG weight estimator* $\omega_{\boldsymbol{\phi}}$. The input of $\omega_{\boldsymbol{\phi}}$ network is a single-channel matrix with (i, j)-th value as the multiplication of the i/j-th values of $\mathbf{x}$/$\mathbf{c}$ CLIP embeddings, respectively. As this matrix is high-dimensional, we input the downsampled $64 \times 64 \times 1$ matrix to the U-Net encoder. These CLIP embeddings are also used to condition the network. With DM pretrained at Stage 1, which is supposed to be sufficiently close to the data distribution, we train the CFG weight estimator by minimizing $\mathbb{E}_{p_{\text{prior}}(\mathbf{z})p_{\text{data}}(\mathbf{c})\pi(\omega)}[\|\omega - \omega_{\boldsymbol{\phi}}(\hat{\mathbf{x}}(\mathbf{z}, \mathbf{c}, \omega), \mathbf{c})\|_2^2]$, where $\hat{\mathbf{x}}(\mathbf{z}, \mathbf{c}, \omega)$ is a clean base-resolution sample drawn the teacher diffusion. Then, $\omega_{\boldsymbol{\phi}}(\mathbf{x}, \mathbf{c})$-value becomes the point estimation of $p(\omega|\mathbf{x}, \mathbf{c})$.

4.2 PaGoDA Pipeline with Classifier-Free Guidance

We replace the adversarial loss in Stages 2 and 3 with the proposed classifier-free guided adversarial loss. In Stage 3, we shift the focus from $\mathbf{x} \in \mathbb{R}^d$ to $\mathbf{x}_{\text{high}} \in \mathbb{R}^{4^n d}$ to effectively capture high-frequency details. Additionally, in both Stages 2 and 3, we replace the input of the generator in the reconstruction loss to be classifier-free guided DDIM inversion noise. To enhance text-sample alignment, we further regularize training with CLIP [30] similarity. For training, we use the ViT-L/14 [31] CLIP model pretrained on YFCC100M [32], while for evaluation, we use the ViT-g/14 CLIP model pretrained on LAION-2B [33], minimizing the risk of overfitting.

5 Experiments

5.1 PaGoDA Tested on ImageNet without CFG

We conduct experiments on ImageNet using PaGoDA without CFG to validate the core pipeline described in Section 3, utilizing the discrete time diffusion scheduling proposed by EDM [13]. Before training, we collect DDIM inversion latent representations for all ImageNet data using the Heun method [13] with 40 timesteps (79 NFE). Throughout the experiments, we maintain the batch size

Figure 6: Uncurated samples generated by PaGoDA at resolution 512×512 *without CFG*. Left: class 31 (tree frog); Right: class 33 (loggerhead turtle).

to be 256 for both $\mathcal{L}_{\text{rec}}$ and $\mathcal{L}_{\text{adv}}$ in Stages 2 and 3. We initialize our base resolution generator with the pre-trained diffusion U-Net. Following CTM [7], we implement adaptive weighting [34] with $\lambda = 0.2\frac{\|\nabla_{\boldsymbol{\theta}^l}\mathcal{L}_{\text{rec}}\|_2^2}{\|\nabla_{\boldsymbol{\theta}^l}\mathcal{L}_{\text{adv}}\|_2^2}$, where $\boldsymbol{\theta}^l$ represents the last layer of the generator.

For higher resolution generation, we double the previous resolution by adding two auxiliary ResNet blocks followed by one upsampler ResNet block. The previously trained generator remains frozen, except for the highest-resolution blocks, which are unfrozen. We then train these newly added blocks along with the unfrozen parts, using a fixed GAN weight of $\lambda = 1.0$. Appendix A.1 provides additional details. By freezing part of the trained generator, we achieve greater stability in super-resolution training without adaptive weighting. See Figure 6 for uncurated 512×512 random samples of ImageNet without CFG.

5.1.1 Quantitative Results

Table 1 presents the performance of PaGoDA. Our model consistently outperforms all existing models across all resolutions, achieving SOTA FIDs without the need of CFG and any other stabilization tricks for GAN. Remarkably, PaGoDA's Inception Score (IS) [35] is on par with other diffusion and GAN models that employed CFG, which implies that PaGoDA samples are as distinctive as CFG samples. Also, PaGoDA generates samples as diverse as the real data distribution, evidenced by diversity recall metric [36], where the PaGoDA reports 0.63 for 64×64 resolution (data's recall is 0.67). In contrast, StyleGAN-XL is far behind of PaGoDA in terms of the diversity metric, reporting 0.52 for 64×64 resolution. Note that we used StyleGAN-XL's discriminator in PaGoDA training, implying that the reconstruction loss significantly improves the sample diversity.

5.1.2 Discussion on Base Resolution

When applying PaGoDA pipeline, the choice of downsampled base resolution in Stage 1 will be primarily determined by available computational resources. Thus, we investigate the impact of the base resolution at this section. To understand the impact, we conducted experiments at 32×32 and 64×64 resolutions, as summarized in Table 2. Starting at resolutions below 32×32 imposes excessive complicacy on the upscaling network, while higher resolutions significantly increase the computational costs at the Stage 1. Therefore, our analysis focuses on these two resolutions, balancing between computational efficiency and upscaling feasibility.

We utilized only 1 H100 node with 8 GPUs for diffusion training on 32×32 with 4096 batch size. Also, for 64×64 diffusion, we borrow a pretrained checkpoint [5], which used ≥ 32[3] A100 GPUs to train with 4096 batch size. Results in Table 2 demonstrate that the diffusion model trained in Stage 1 maintains robust performance across both resolutions. Interestingly, the one-step generator distilled

[3]This is an estimate.

Table 1: Experimental results of PaGoDA on ImageNet.

Model	Sampling NFE	Without CFG FID ↓	IS ↑	Recall ↑	With CFG FID	IS	Recall	Without CFG FID	IS	Recall	With CFG FID	IS	Recall
		64 × 64 resolution						**128 × 128 resolution**					
RIN [37]	250	1.23	66.5	-	-	-	-	2.75	144.1	-	-	-	-
simple Diffusion [38]	250	-	-	-	-	-	-	1.91	171.9	-	2.05	189.9	-
VDM++ [39]	79	1.43	63.7	-	-	-	-	1.75	171.1	-	1.78	190.5	-
StyleGAN-XL [40]	1	-	-	-	1.51	**82.35**	0.52	-	-	-	1.81	**200.55**	0.55
CTM [7]	1	1.92	70.38	0.57	-	-	-	-	-	-	-	-	-
PaGoDA (ours)	1	**1.21**	76.47	**0.63**	-	-	-	**1.48**	174.36	**0.61**	-	-	-
		256 × 256 resolution						**512 × 512 resolution**					
DiT-XL [41]	250	9.62	121.5	-	2.27	278.2	-	12.03	105.3	-	3.04	240.8	-
simple Diffusion [38]	250	2.77	211.8	-	2.44	256.3	-	3.54	205.3	-	3.02	248.7	-
VDM++ [39]	250	2.40	225.3	-	2.12	267.7	-	2.99	232.2	-	2.65	278.1	-
EDM2-XXL [42]	63	-	-	-	-	-	-	1.91	-	-	1.81	-	-
StyleGAN-XL [40]	1	-	-	-	2.30	**265.12**	0.53	-	-	-	2.41	**267.75**	0.52
PaGoDA (ours)	1	**1.56**	259.61	**0.59**	-	-	-	**1.80**	251.31	**0.58**	-	-	-

Table 2: Ablation of base resolution.

Model	Base Res	Upscaled Res	NFE	FID	Base Res	Upscaled Res	NFE	FID	Speed [s]	Params
Teacher Diffusion	32 × 32	32 × 32	79	1.75	64 × 64	64 × 64	79	2.44	3.16s	296M
PaGoDA	32 × 32	32 × 32	1	0.79						
	32 × 32	64 × 64	1	1.34	64 × 64	64 × 64	1	1.21	0.040s	296M
	32 × 32	128 × 128	1	1.61	64 × 64	128 × 128	1	1.48	0.041s	299M
	32 × 32	256 × 256	1	1.83	64 × 64	256 × 256	1	1.56	0.044s	301M
					64 × 64	512 × 512	1	1.80	0.046s	302M

in Stage 2 consistently outperforms the teacher model, likely benefiting from the effectiveness of StyleGAN-XL [40], combined with the reconstrcution loss. In Stage 3, the degree of upscaling from the base resolution emerges as the most influential factor for the quality, with upscaling up to 8x showing minimal performance degradation across both tested resolutions.

The upscaler in PaGoDA refines coarse samples generated at lower resolutions, making the pipeline inherently aligned with the scaling laws of smaller resolutions. This design is advantageous, as scaling laws typically worsen with increasing resolution [43], while PaGoDA leverages the more favorable scaling dynamics at lower resolutions to maintain efficiency. Furthermore, the lightweight upscaling module introduces minimal additional latency, keeping inference times nearly identical to those at the base resolution. This practical efficiency makes PaGoDA a promising solution for scalable diffusion model training across various computational settings.

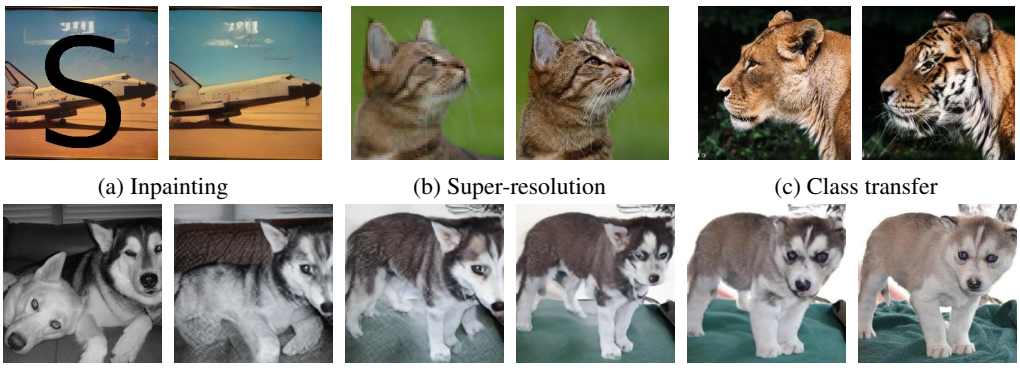
5.1.3 Discussion on Upscaling Capability

In Stage 3, we train the super-resolution module using a combination of reconstruction and adversarial losses. As shown in Figures 3 and 4, we compare PaGoDA's performance to that of StyleGAN-XL. The comparison reveals key insights: 1) PaGoDA maintains consistent object alignment across resolution jumps, largely due to the reconstruction loss, and 2) its performance is strongly influenced by the GAN component, which plays a crucial role in capturing high-frequency details.

Other upsampling methods, such as SD and Cascaded Diffusion Models (CDM) [44] also target high-quality upscaling. While PaGoDA, CDM, and SD share the same goal, they adopt different approaches, making them complementary rather than competing solutions. In fact, their strengths can be combined to enhance overall compression and upscaling performance. For instance, CDM or PaGoDA can be applied to the latent space of SD, integrating their techniques for better results. Despite their compatibility, it is still essential to assess how these methods compare in terms of their upscaling effectiveness. In the following analysis, we break down the upscaling capabilities of PaGoDA, CDM, and SD to understand their respective strengths and potential limitations.

Table 3: Comparison on upsampling.

Model	Resolution	Params	NFE	FID
EDM2	64^2 DM	1.1B	63	1.33
	512^2 LDM	1.1B	63+1	1.96
PaGoDA	64^2 DM (teacher)	0.3B	79	2.44
	$64^2 \to 64^2$	0.3B	1	1.21
	$64^2 \to 512^2$	0.3B	1	1.80



(a) Inpainting (b) Super-resolution (c) Class transfer

(d) Latent interpolation

Figure 8: Controllable generation of PaGoDA with various tasks.

Since PaGoDA is experimented based on EDM [13], we adapted the experimental results from EDM2 [42] to facilitate a direct comparison with PaGoDA in the upscaling performance. EDM2 presents results for both pixel DM and latent DM. In latent diffusion, a $512 \times 512 \times 3$ image is compressed into a $64 \times 64 \times 4$ latent space for training DM, while pixel diffusion operates directly on $64 \times 64 \times 3$ images, sharing the identical network architecture used in its latent DM. As reported in Table 3, EDM2 shows a minor performance decline from 1.33 to 1.96.

Similarly, PaGoDA exhibits a comparable performance drop from 1.21 to 1.80 when upscaling from 64×64 to 512×512. This similarity suggests that PaGoDA's upscaling capacity aligns closely with that of the LDM framework, indicating minimal performance differences even when handling high-resolution data.

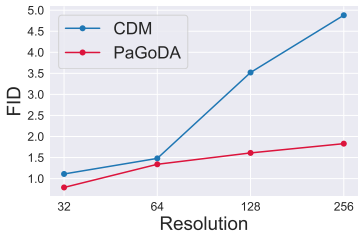


Figure 7: Comparison between PaGoDA and CDM.

Lastly, when comparing PaGoDA to CDM, we observe in Figure 7 that CDM encounters significant performance drops beyond certain dimensional thresholds (128×128), while PaGoDA maintains consistent performance across varying resolutions. This robustness makes PaGoDA a reliable option for high-resolution generation, with its performance remaining steady even as resolution increases.

5.2 Discussion on Controllability

Once we have a trained PaGoDA generator $G_{\boldsymbol{\theta}_0}$, we can utilize it for solving inverse problems [45] and for controllable generation [46] in a training-free manner [47].

Latent Optimization We consider the inverse problem: $\mathbf{y} = \mathcal{A}(\mathbf{x}) + \eta$, where $\mathbf{y}$ represents the observation, and $\mathcal{A} : \mathbb{R}^d \to \mathbb{R}^m$ with $d \geq m$ is a known operator. The restored data $\mathbf{x}$ can be reconstructed by optimizing the latent. Specifically, if $\mathbf{z}^* \in \arg\min_{\mathbf{z}} \|\mathbf{y} - \mathcal{A}(G_{\boldsymbol{\theta}_0}(\mathbf{z}, \mathbf{c}))\|_2^2$, then $G_{\boldsymbol{\theta}_0}(\mathbf{z}^*, \mathbf{c})$ is the best possible estimate of the solution for the inverse problem. Figure 8-(a) displays the outcomes of an inpainting task where latent optimization is employed with Adam optimizer [48].

DDIM Inversion Specific tasks, such as super-resolution illustrated in Figure 8-(b) and class transfer depicted in Figure 8-(c), can be effectively addressed without relying on latent optimization. For these tasks, we apply DDIM inversion to the downsampled observations, then map the DDIM latent back to RGB pixel by feeding the latent into the decoder. Generally, using DDIM inversion yields superior outcomes compared to latent optimization for these types of tasks.

Latent Interpolation Building on techniques from GAN research, we also explored latent interpolation for style mixing. Despite our model's latent dimension being larger than the typical 512-dimensional style vector used in GAN, our observations indicate that latent mixing by slerp operation [49, 20] achieves effective results, as demonstrated in Figure 8-(d).

Table 4: Experimental results on T2I. FID-5K is measured on MSCOCO-2017 [54] validation data. CLIP score is measured by the ViT-g/14 backbone. Our model uses DeepFloy-IF as the pre-trained diffusion.

Model	Params	NFE	Speed [s]	FID ↓	CLIP ↑
SD1.5 [3]	0.9B	50+1	2.59s	19.1	31.3
DeepFloyd-IF [53]	0.9B	27	2.95s	22.3	28.1
Latent Distillation Models based on SD1.5 [3]					
CAD [28]	0.9B	8+1	0.34s	24.2	30.0
PD [55]	0.9B	4+1	0.21s	26.4	30.0
LCM [56]	0.9B	2+1	0.13s	30.4	29.3
InstaFlow [57]	0.9B	1+1	0.09s	23.4	30.4
UFOGen [58]	0.9B	1+1	0.09s	22.5	31.1
Scott [59]	0.9B	1+1	0.09s	21.9	31.2
ADD [26]	0.9B	1+1	0.09s	19.7	32.6
Pixel Distillation Model based on DeepFloyd-IF [53]					
PaGoDA (ours)	0.9B	1	0.05s	20.4	31.2

Table 5: Experimental results on T2I. FID-30K is based on MSCOCO-2014 [54] validation data. Speed is measured on A100.

Model	Params	Speed [s]	FID ↓
eDiff-I [60]	9.1B	32.0s	6.95
LDM [3]	1.5B	9.4s	12.63
Imagen [61]	3.0B	9.1s	7.27
SD1.5 [3]	0.9B	2.9s	9.62
PixArt-α [62]	0.6B	-	10.65
Scott [59]	0.9B	0.13s	12.22
GigaGAN [24]	1.0B	0.13s	9.09
StyleGAN-T [25]	1.0B	0.10s	13.90
InstaFlow [57]	0.9B	0.09s	13.10
UFOGen [58]	0.9B	0.09s	12.78
DMD [63]	0.9B	0.09s	11.49
LAFITE [64]	75M	0.02s	26.94
PaGoDA (ours)	0.9B	0.05s	10.23

5.3 Text-to-Image Generation

We collect the data-latent pairs on the CC12M dataset [50] through DDIM inversion and utilize the filtered COYO-700M [51] dataset for adversarial training. The filtering criteria include only data with CLIP score (measured by ViT-B/32 [52]) higher than 32, and aesthetic score-v2 [33] higher than 5.0. Due to concerns regarding sensitive content in the open-sourced LAION dataset [33], we were unable to conduct large-scale diffusion training for Stage 1. This constraint led us to focus primarily on stages 2 and 3, leveraging pretrained open-source checkpoints. For the pretrained teacher diffusion, we used the DeepFloyd-IF model [53], trained on 64×64 pixel space. For further experimental details, see Appendix A.2.

Table 4 compares our PaGoDA mainly with the distilled models from SD v1.5 on 512×512. One notable observation from the table is that, even though the latent distilled model generates the latent representation in a single step, additional time is required for decoding this latent into image. In contrast, PaGoDA (on pixel teacher) eliminates such decoding step, thereby overcoming the time constraints associated with distilling SD models. For a more detailed breakdown of the time taken by each component, see Figure 9.

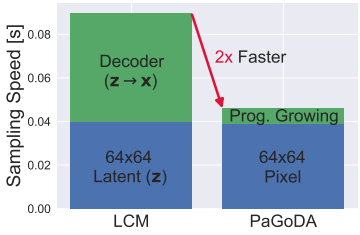


Figure 9: PaGoDA offers faster inference than the one-step LCM.

Returning to the performance results in the table, PaGoDA achieves performance comparable to that of the teacher model. This superior performance is also observed on a different test set as shown in Table 5, further demonstrating PaGoDA's scalability on text-to-image tasks.

6 Conclusion

PaGoDA introduces a training pipeline that can democratize the diffusion training by cutting training budget with orders of magnitudes. The pipeline is consisted of three stages: 1) we pretrain the diffusion models on the downsampled data, 2) we distill the teacher diffusion into a one-step generator on the downsampled data, and 3) we train an upsampler module until we reach to the desired resolution.

Acknowledgement

This project was supported by Sony, ARO (W911NF-21-1-0125), ONR (N00014-23-1-2159), and the CZ Biohub. Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used. We extend our special thanks to our colleagues Takashi Shibuya from Sony AI and Yutong He from Carnegie Mellon University for their invaluable feedback.

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

[2] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations*, 2020.

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.

[5] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. *arXiv preprint arXiv:2303.01469*, 2023.

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

[7] Dongjun Kim, Chieh-Hsin Lai, Wei-Hsiang Liao, Naoki Murata, Yuhta Takida, Toshimitsu Uesaka, Yutong He, Yuki Mitsufuji, and Stefano Ermon. Consistency trajectory models: Learning probability flow ode trajectory of diffusion. In *International Conference on Learning Representations*, 2024.

[8] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024.

[9] Black-Forest. Flux. https://blackforestlabs.ai/announcing-black-forest-labs/, 2024.

[10] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[13] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in Neural Information Processing Systems*, 35:26565–26577, 2022.

[14] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022.

[15] Pablo Pernias, Dominic Rampas, Mats Leon Richter, Christopher Pal, and Marc Aubreville. Würstchen: An efficient architecture for large-scale text-to-image diffusion models. In *The Twelfth International Conference on Learning Representations*, 2023.

[16] Jyoti Aneja, Alex Schwing, Jan Kautz, and Arash Vahdat. A contrastive learning approach for training variational autoencoder priors. *Advances in neural information processing systems*, 34:480–493, 2021.

[17] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. *arXiv preprint arXiv:1605.07146*, 2016.

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.

[19] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017.

[20] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019.

[21] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR, 2018.

[22] Vaishnavh Nagarajan and J Zico Kolter. Gradient descent gan optimization is locally stable. *Advances in neural information processing systems*, 30, 2017.

[23] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021.

[24] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10124–10134, 2023.

[25] Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. In *International conference on machine learning*, pages 30105–30118. PMLR, 2023.

[26] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. *arXiv preprint arXiv:2311.17042*, 2023.

[27] Axel Sauer, Frederic Boesel, Tim Dockhorn, Andreas Blattmann, Patrick Esser, and Robin Rombach. Fast high-resolution image synthesis with latent adversarial diffusion distillation. *arXiv preprint arXiv:2403.12015*, 2024.

[28] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306, 2023.

[29] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

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

[31] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[32] Bart Thomee, David A Shamma, Gerald Friedland, Benjamin Elizalde, Karl Ni, Douglas Poland, Damian Borth, and Li-Jia Li. Yfcc100m: The new data in multimedia research. *Communications of the ACM*, 59(2):64–73, 2016.

[33] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022.

[34] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[35] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[36] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in neural information processing systems*, 32, 2019.

[37] Allan Jabri, David Fleet, and Ting Chen. Scalable adaptive computation for iterative generation. *arXiv preprint arXiv:2212.11972*, 2022.

[38] Emiel Hoogeboom, Jonathan Heek, and Tim Salimans. simple diffusion: End-to-end diffusion for high resolution images. In *International Conference on Machine Learning*, pages 13213–13232. PMLR, 2023.

[39] Diederik P Kingma and Ruiqi Gao. Understanding diffusion objectives as the elbo with simple data augmentation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

[40] Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pages 1–10, 2022.

[41] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.

[42] Tero Karras, Miika Aittala, Jaakko Lehtinen, Janne Hellsten, Timo Aila, and Samuli Laine. Analyzing and improving the training dynamics of diffusion models. *arXiv preprint arXiv:2312.02696*, 2023.

[43] Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative modeling. *arXiv preprint arXiv:2010.14701*, 2020.

[44] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 23(1):2249–2281, 2022.

[45] Hyungjin Chung, Jeongsol Kim, Michael T Mccann, Marc L Klasky, and Jong Chul Ye. Diffusion posterior sampling for general noisy inverse problems. *arXiv preprint arXiv:2209.14687*, 2022.

[46] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023.

[47] Yutong He, Naoki Murata, Chieh-Hsin Lai, Yuhta Takida, Toshimitsu Uesaka, Dongjun Kim, Wei-Hsiang Liao, Yuki Mitsufuji, J Zico Kolter, Ruslan Salakhutdinov, et al. Manifold preserving guided diffusion. In *International Conference on Learning Representations*, 2023.

[48] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[49] Ken Shoemake. Animating rotation with quaternion curves. In *Proceedings of the 12th annual conference on Computer graphics and interactive techniques*, pages 245–254, 1985.

[50] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021.

[51] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022.

[52] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[53] DeepFloyd Lab. If by deepfloyd lab at stabilityai. https://github.com/deep-floyd/IF, 2023.

[54] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer, 2014.

[55] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2021.

[56] Simian Luo, Yiqin Tan, Longbo Huang, Jian Li, and Hang Zhao. Latent consistency models: Synthesizing high-resolution images with few-step inference. *arXiv preprint arXiv:2310.04378*, 2023.

[57] Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *The Twelfth International Conference on Learning Representations*, 2023.

[58] Yanwu Xu, Yang Zhao, Zhisheng Xiao, and Tingbo Hou. Ufogen: You forward once large scale text-to-image generation via diffusion gans. *arXiv preprint arXiv:2311.09257*, 2023.

[59] Hongjian Liu, Qingsong Xie, Zhijie Deng, Chen Chen, Shixiang Tang, Fueyang Fu, Zhengjun Zha, and Haonan Lu. Scott: Accelerating diffusion models with stochastic consistency distillation. *arXiv preprint arXiv:2403.01505*, 2024.

[60] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022.

[61] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022.

[62] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023.

[63] Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. *arXiv preprint arXiv:2311.18828*, 2023.

[64] Yufan Zhou, Ruiyi Zhang, Changyou Chen, Chunyuan Li, Chris Tensmeyer, Tong Yu, Jiuxiang Gu, Jinhui Xu, and Tong Sun. Towards language-free training for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17907–17917, 2022.

[65] Bosco Yung. Open-nsfw 2. https://github.com/bhky/opennsfw2, 2021.

[66] Gant Laborde. https://github.com/GantMan/nsfw_model.

[67] Roman Inflianskas. https://github.com/rominf/profanity-filter/blob/master/profanity_filter/data/en_profane_words.txt.

[68] Jaclyn Brockschmidt. https://github.com/snguyenthanh/better_profanity/blob/master/better_profanity/profanity_wordlist.txt.

[69] Jamie Dubs and Ryan Lewis. https://gist.github.com/ryanlewis/a37739d710ccdb4b406d.

[70] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.

[71] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[72] Shengyu Zhao, Zhijian Liu, Ji Lin, Jun-Yan Zhu, and Song Han. Differentiable augmentation for data-efficient gan training. *Advances in neural information processing systems*, 33:7559–7570, 2020.

[73] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. *arXiv preprint arXiv:1908.03265*, 2019.

[74] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[75] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *Computer Science. https://cdn. openai. com/papers/dall-e-3. pdf*, 2(3):8, 2023.

[76] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[77] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[78] Tim Dettmers, Mike Lewis, Sam Shleifer, and Luke Zettlemoyer. 8-bit optimizers via block-wise quantization. *arXiv preprint arXiv:2110.02861*, 2021.

[79] Adrien Saumard and Jon A Wellner. Log-concavity and strong log-concavity: a review. *Statistics surveys*, 8:45, 2014.

[80] Wenpin Tang and Hanyang Zhao. Contractive diffusion probabilistic models. *arXiv preprint arXiv:2401.13115*, 2024.

[81] Junlong Lyu, Zhitang Chen, and Shoubo Feng. Sampling is as easy as keeping the consistency: convergence guarantee for consistency models. 2023.

[82] Xuefeng Gao, Hoang M Nguyen, and Lingjiong Zhu. Wasserstein convergence guarantees for a general class of score-based generative models. *arXiv preprint arXiv:2311.11003*, 2023.

[83] Bernard A Asner, Jr. On the total nonnegativity of the hurwitz matrix. *SIAM Journal on Applied Mathematics*, 18(2):407–414, 1970.

[84] Nam Parshad Bhatia and Giorgio P Szegö. *Stability theory of dynamical systems*. Springer Science & Business Media, 2002.

[85] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. *Advances in neural information processing systems*, 30, 2017.

[86] David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. In *International Conference on Machine Learning*, pages 354–363. PMLR, 2018.

[87] Ian Gemp and Sridhar Mahadevan. Global convergence to the equilibrium of gans using variational inequalities. *arXiv preprint arXiv:1808.01531*, 2018.

[88] Chuang Wang, Hong Hu, and Yue Lu. A solvable high-dimensional model of gan. *Advances in Neural Information Processing Systems*, 32, 2019.

[89] Chongli Qin, Yan Wu, Jost Tobias Springenberg, Andy Brock, Jeff Donahue, Timothy Lillicrap, and Pushmeet Kohli. Training generative adversarial networks by solving ordinary differential equations. *Advances in Neural Information Processing Systems*, 33:5599–5609, 2020.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We have made our abstract and introduction to accurately reflect the core contribution of the paper.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We have created a separate "Limitations and Broader Impacts" section in the appendix to enumerate potential limitations of our methodology, including the algorithmic, theoretical, and experimental limitations.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Although we omit some of assumptions in the main paper mainly due to page limit, we provide full details of assumptions and complete proof in the appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We disclose all experimental details in the main paper and the appendix, including the hyperaparameters used and the datasets used with their filetering methodologies. For further reproducibility, we plan to release our code upon acceptance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: In the reviewing process, we release our code to the reviewers to regenerate our experimental results. After the acceptance, we plan to release the code to the public.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We faithfully release our hyperparameters and experimental details in the appendix and the main paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We have not reported error bars mainly due to the lack of computational resources.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We explain how much compute resources we used for experiments in the appendix.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: We faithfully follow the code of ethics, suggested by the link above.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the broader impacts as a separate section in the "Limitations and Broader Impacts" of the Appendix C.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.

- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: For the T2I checkpoint release, we plan to use HuggingFace to enroll every users to the system so to control the downloaded user list. Additionally, we prohibited using the LAION dataset [33], which includes NSFW contents. Instead, we used the COYO-700M [51] dataset, a large-scale text-to-image dataset that removes NSFW images by NSFW image detectors [65, 66] and texts that contain NSFW words [67–69].

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have properly credited the original owners of assets by citing them. In the code release, we comply the license and terms of the assets.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: In the supplementary, we include the the details of the dataset/code/model via structured templates.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

Contents

1 Introduction **1**

2 Preliminary **3**

3 Progressive Growing of Diffusion Autoencoder **3**

3.1 Stage 1: Diffusion Models Trained on Downsampled Data 3

3.2 Stage 2: Diffusion Distillation on Downsampled Data with DDIM Inversion 3

3.3 Stage 3: Progressively Growing Decoder for Super-Resolution 4

3.4 Optimality Guarantee and Training Stability of PaGoDA Pipeline 5

4 PaGoDA with Classifier-Free Guidance **5**

4.1 Classifier-Free Guided Adversarial Loss 6

4.2 PaGoDA Pipeline with Classifier-Free Guidance 6

5 Experiments **6**

5.1 PaGoDA Tested on ImageNet without CFG 6

5.1.1 Quantitative Results 7

5.1.2 Discussion on Base Resolution 7

5.1.3 Discussion on Upscaling Capability 8

5.2 Discussion on Controllability 9

5.3 Text-to-Image Generation 10

6 Conclusion **10**

A Experimental Details **24**

A.1 Conditional Generation with ImageNet 24

A.2 Text-to-Image Generation 25

B Theoretical Analysis **28**

B.1 Convergence with PaGoDA's Reconstruction Loss 28

B.1.1 Preliminaries of Convergence Analysis 28

B.1.2 W_2 Bound with PaGoDA's Reconstruction Loss 29

B.1.3 W_1 Bound with PaGoDA's Reconstruction Loss 33

B.2 Optimality analysis 34

B.3 Stability Analysis 35

B.3.1 Preliminaries of Dynamical System 35

B.3.2 Preliminaries for Analysis of PaGoDA Training 36

B.3.3 PaGoDA's Training is Stable 37

B.3.4 Literature on Stability Analysis of Adversarial Training 40

C Limitations and Broader Impacts **42**

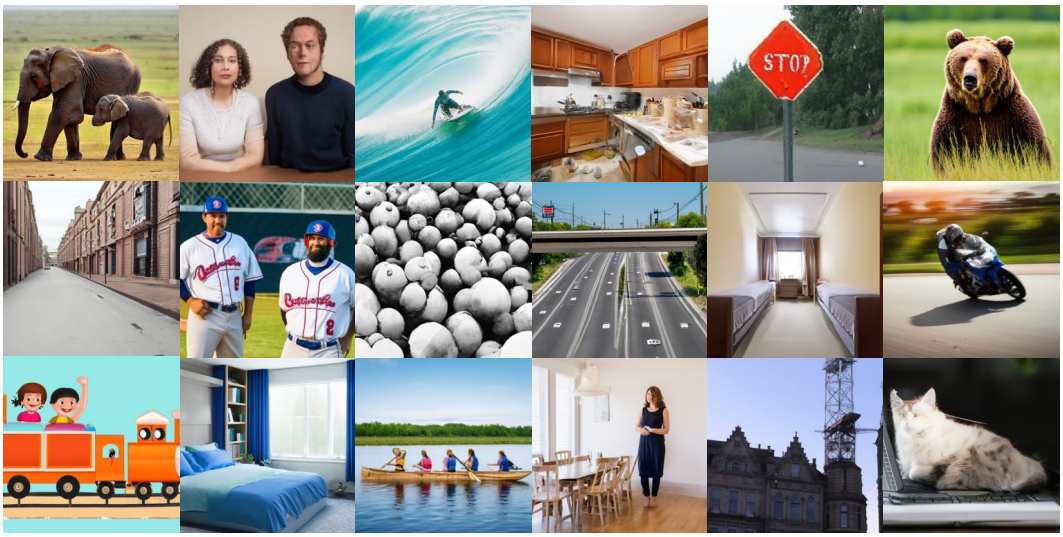


Figure 10: Text-to-image samples from PaGoDA.

A Experimental Details

A.1 Conditional Generation with ImageNet

Throughout the experiments, we omit the class condition $\mathbf{c}$ otherwise mentioned for notational simplicity.

Dataset Construction. We loaded ImageNet2014[4] dataset using center cropping and downsampling using the bicubic algorithm from the PIL python package. To augment the data, we applied a horizontal random flip, and obtained each of latent representations by solving the EDM's 2nd-order ODE sampler (Heun's method) [13] with their suggested diffusion time scheduling and timestep selection. Consequently, in total, we processed approximately 2.5 million data instances forward in time using the PF-ODE to prepare for training. This computational cost of constructing the training dataset is comparable to sampling an equivalent volume of sample from a pre-trained diffusion model.

GAN Details. We adopted the discriminator architecture from StyleGAN-XL. Initially, We loaded DeiT-base [70] and EfficientNet-lite [71] as feature extractors, in line with StyleGAN-XL's setup. When processing real or fake data through the discriminator, we first applied differentiable augmentation (DiffAugment) [72], incorporating three transformations: *Translation*, *Cutout*, and *Color*. Interestingly, we observed no performance differences between the *unconditional* and *conditional* discriminators. We hypothesize that this lack of disparity arises because the discriminator primarily updates the generator to refine high-frequency details, while preserving the low-frequency global semantics due to the reconstruction power. Additionally, we opted not to use additional techniques to tame the GAN training, such as R1 regularization [21] or path length regularization [20] in our GAN training. PaGoDA's training generally remains stable due to its reconstruction loss, which is consistent with our theoretical expectation (Theorem B.9).

We conducted tests on GANs under two distinct scenarios. Initially, following the approach used in Stable Diffusion's VAE training, we introduced both the real data $\mathbf{x}$ and the reconstructed sample $\tilde{\mathbf{x}} = G_{\boldsymbol{\theta}}(E(\mathbf{x}))$ to the discriminator, training it to differentiate between the two while updating the generator to maximize $\log D_{\boldsymbol{\psi}}(\tilde{\mathbf{x}})$. In this setup, as the reconstruction only utilizes the latent representation $E(\mathbf{x})$, the generation quality is not improved.

In the second scenario, adhering to the traditional GAN framework, we trained the discriminator using randomly sampled real data alongside randomly generated fake data $\tilde{\mathbf{x}} = G_{\boldsymbol{\theta}}(\mathbf{z})$ from $\mathbf{z} \sim p_{\text{prior}}(\mathbf{z})$. Then, the endeavor of maximizing $\log D_{\boldsymbol{\psi}}(\tilde{\mathbf{x}})$ now significantly improves the generation quality. Overall, we observed no performance degradation when both types of GAN training were applied to

[4] https://www.image-net.org/index.php

Table 6: Comparison on ImageNet 64×64. We evaluate scores, including Fréchet distance on DINOv2 features [74], based on the statistics released by EDM2. The validation scores are measured by comparing 50k samples and 50k ImageNet validation data.

64×64	Architecture	NFE	$\text{FID}_{\text{InceptionV3}}$		$\text{FD}_{\text{DINOv2}}$	
			vs. Train	vs. Val	vs. Train	vs. Val
Val Data			1.05	-	13.86	-
StyleGAN-XL	GAN	1	2.64	3.52	214.59	220.47
CTM	ADM	1	1.69	2.88	159.67	165.96
EDM	ADM	79	2.51	2.93	112.17	120.58
EDM2-XL	EDM2	63	1.38	2.29	70.31	80.53
PaGoDA	ADM	1	1.01	2.10	70.04	78.44

Table 7: Comparison on ImageNet 512×512. From this result, it would be interesting to experiment PaGoDA on EDM2 architecture for better performance. * the results of ImageNet 256×256.

512×512	Arch	DM	#Params	NFE	$\text{FID}_{\text{InceptionV3}}$	$\text{FD}_{\text{DINOv2}}$
Val Data					1.58	14.13
StyleGAN-XL	GAN	-	0.2B	1	2.41	214.88*
EDM w/o CFG (teacher)	ADM	Latent	0.3B	63	7.24	204.10
EDM2-S w/o CFG	EDM2	Latent	0.3B	63	2.56	68.64
EDM2-S w/ CFG	EDM2	Latent	0.3B	63	2.23	55.23
EDM2-XXL w/ CFG	EDM2	Latent	1.1B	63	1.81	33.09
PaGoDA w/o CFG	ADM	Pixel	0.3B	1	1.80	96.77

- You are LLaVA, a large language and vision assistant trained by UW Madison WAIV Lab.
- You are able to understand the visual content that the user provides, and assist the user with a variety of tasks using natural language.
- You should follow the instructions carefully and explain your answers in detail.
- Given the caption of this image "{text prompt}", describe this image in a very detailed manner

Figure 11: Input prompt of LLaVA to recaption the text-image paired data.

the generator. However, given our limited budget and the goal to develop a generative model rather than a compression model, we opted to proceed solely with the second type of GAN setup.

Reconstruction Details. For the reconstruction loss, we train the generator $G_{\boldsymbol{\theta}}$ by comparing the original data $\mathbf{x} \sim p_{\text{data}}(\mathbf{x})$ and its reconstructed counterpart $G_{\boldsymbol{\theta}}(E(\mathbf{x}, \mathbf{c}), \mathbf{c})$ at the data's resolution, where $E(\mathbf{x}, \mathbf{c})$ is the solution of the DDIM inversion. Since our training occurs in pixel space, we conduct this comparison in the feature space using the Learned Perceptual Image Patch Similarity (LPIPS) metric, and there is no need to develop a new feature extractor in latent space. We experimented with features extracted from DeiT-base [70] and EfficientNet-lite [71]; however, we observed no notable improvement from using LPIPS.

For the training, we use the RAdam [73] with learning rate of 8e-6 for the decoder and 2e-3 for the discriminator, and without weight decay. We use the EMA of 0.999, and all reported FIDs are based on the EMA checkpoint. Until 256^2 resolution, we use only 1 H100 node (with 80Gb memory) to train, and we use 8 A100 nodes (with 40Gb memory, in total $8 \times 8 = 64$ GPUs) to train the 512^2 model. Throughout the experiments, we use the batch size of 256.

For the concerns on the overfitting, we provide additional results in Tables 6 and 7.

A.2 Text-to-Image Generation

Dataset Construction. Due to the presence of inappropriate contents (CSAM) in the LAION dataset [33], we have decided to discontinue its use. Instead, we are now training our model using the CC12M [50] and a filtered version of COYO-700M [51] datasets. For COYO-700M, we apply filters to select only those text-image pairs that meet specific criteria: a CLIP score (measured by ViT-B/32 [52]) above 32.0 and an aesthetic score-v2 [33] higher than 5.0. Additionally, we are enhancing the dataset quality by recaptioning the original text prompts from CC12M, adopting practices similar to those used in DallE-3 [75] and PixArt-α [62]. Specifically, we employ LLaVA-7B [76], a language

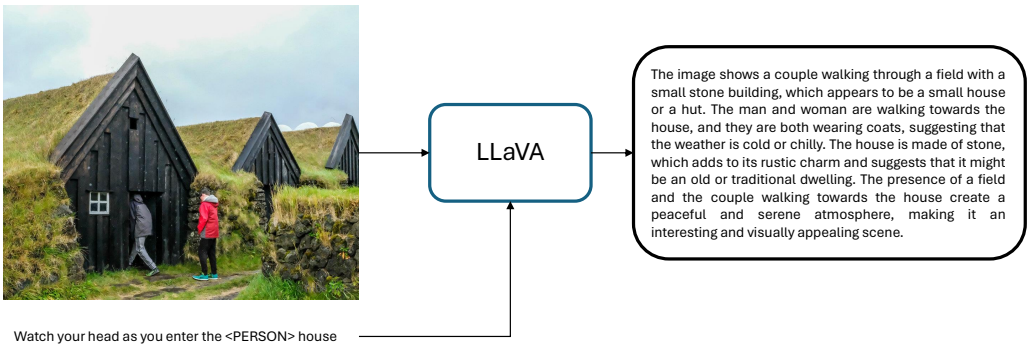
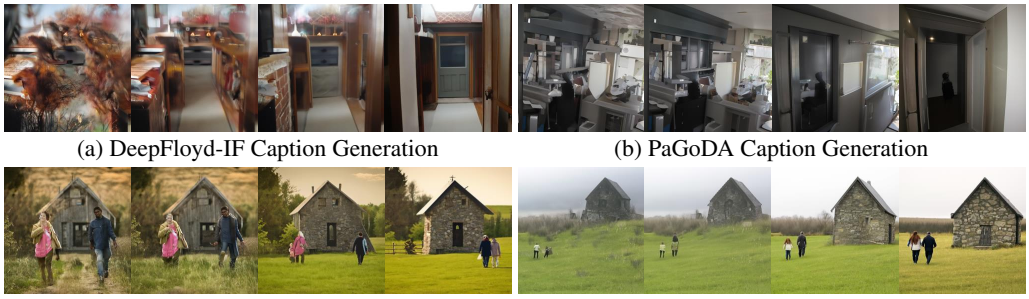


Figure 12: Example of recaptioned image-text pair.



(a) DeepFloyd-IF Caption Generation

(b) PaGoDA Caption Generation

(c) DeepFloyd-IF Reception Generation

(d) PaGoDA Reception Generation

Figure 13: Caption vs. Reception. From left to right, CFG scale increases. The caption and its corresponding reception are given by the exemplary case in Figure 12.

model with vision assistance, to generate descriptions of the images based on the text prompts, thereby ensuring more relevant and accurate text-image pairings.

The input prompt of LLaVA is depicted in Figure 11, where we put text prompt to {`text prompt`}. The output from this recaptioning process adheres to a consistent format, typically beginning with phrases like "`This image features ...`" or "`This image shows ...`". To provide clear demonstration, Figure 13 displays several examples of original captions alongside their recaptioned counterparts.

Interestingly, the recaptioned samples generally outperform the original caption samples. Notably, the recaptioned samples exhibit sufficient quality, particularly when the CFG scale is small, as shown in Figure 14. Therefore, to ensure balanced generation performance across varying CFG scales, we generate samples from the original captions with the CFG scale uniformly sampled from the range $[2, 10]$. For the recaptioned text, we use a CFG scale that follows a truncated Gaussian distribution on the range $[1, 10]$, centered at 2 with a scale of 3. Overall, incorporating these recaptioned texts into the PaGoDA training results in only a marginal improvement in performance metrics such as FID and CLIP. However, it significantly enhances the actual quality of generation, particularly at smaller CFG scales, because the recaptioning provides better-aligned training data.

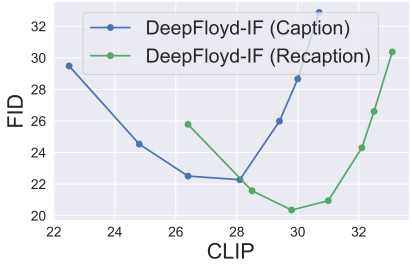


Figure 14: Effect of recaptioning.

Using LLaVA, we reception $\tilde{\mathbf{c}}(\mathbf{x}, \mathbf{c})$ and obtain the DDIM latent representation, $E(\mathbf{x}, \tilde{\mathbf{c}}(\mathbf{x}, \mathbf{c}))$, on the entire CC12M dataset. Then, for the original text $\mathbf{c}$, we have a triplet of $(\text{image}, \text{text}, \text{latent})$ of $(\mathbf{x}, \mathbf{c}, E(\mathbf{x}, \mathbf{c}))$ for one set, and another triplet $(\mathbf{x}, \tilde{\mathbf{c}}(\mathbf{x}, \mathbf{c}), E(\mathbf{x}, \tilde{\mathbf{c}}(\mathbf{x}, \mathbf{c}))$ for recaptioned dataset. When computing $\mathcal{L}_{\text{rec}}$, we mix these triplets and randomly sample from this mixed dataset.

GAN Details. Similar to the ImageNet case, we have adopted the discriminator architecture from StyleGAN-T. In line with StyleGAN-T, we utilize the DINO ViT-S/16 [77] as the feature extractor and apply DiffAugment [72], incorporating *Translation*, *Cutout*, and *Color* transformations. Building

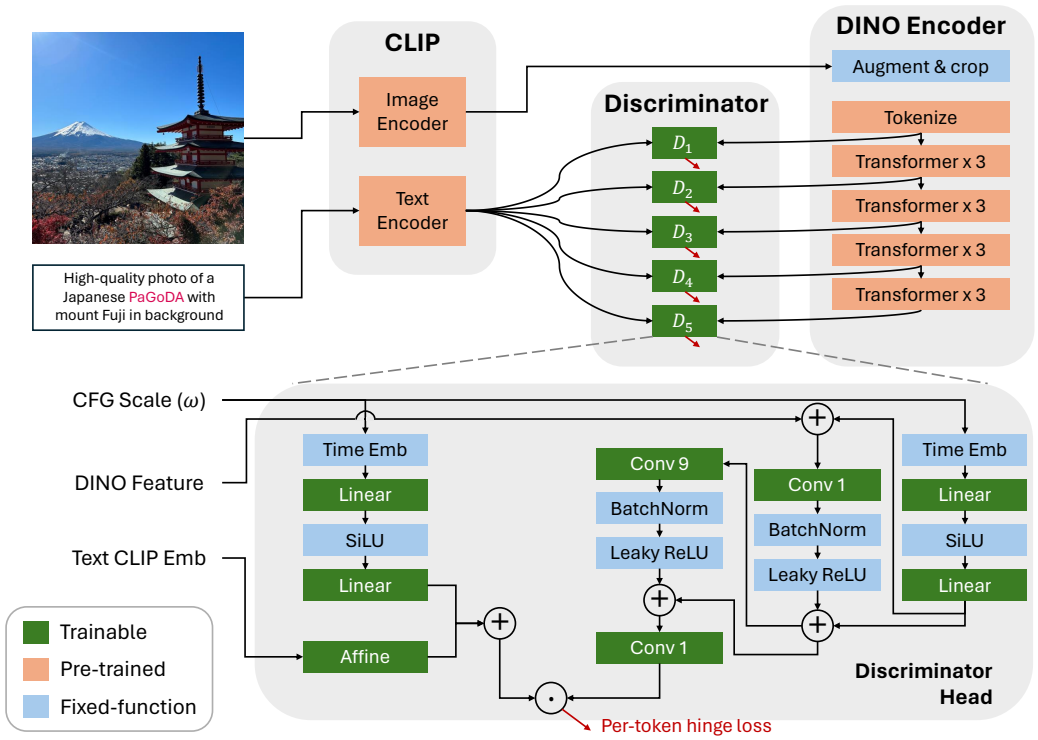


Figure 15: Discriminator architecture.

upon this architecture, we integrate a ω condition into each discriminator head, as illustrated in Figure 15. The inputs for each discriminator head include the DINO feature, text CLIP embedding, and the CFG scale ω, which is scaled by a factor of 100. We handle the CFG scale similarly to the time variable in traditional diffusion U-Net models, incorporating the output CFG embedding into the existing components of the StyleGAN-T discriminator head. We assume both image $\mathbf{x}$ and text $\mathbf{c}$ are related with the CFG scale, thus we designed the discriminator to incorporate ω information into both modules, enhancing the relevance and contextuality of the discrimination process.

Reconstruction Details. In our text-to-image training, we largely adhere to the protocols established for ImageNet training. However, a notable modification involves the decoder network, which now incorporates a ω condition as an auxiliary input. Crucially, this ω condition is processed in decoder in the same way as the time condition in diffusion models. We achieve this by scaling ω by a factor of 100, thus aligning it with the existing time ranges. This method ensures a consistent treatment of the ω parameter, integrating it smoothly into the established model architecture.

CLIP Details. Neither the reconstruction loss nor the GAN loss directly models or maximizes the text-image correlation. To address this, we introduce an additional text-image alignment metric to train our model. Specifically, we employ ViT-L/14 [52] to assess the CLIP value. This regularization significantly enhances PaGoDA's performance, as evidenced in Figure 16 by improving both FID and CLIP scores. These enhancements suggest that not only is the sample quality improving, but also the alignment between text and images is becoming more accurate.

For the training, we use the AdamW8bit optimizer [78] to minimize the required memory with learning rate of 1e-5 for both decoder and discriminator. Similar to the ImageNet experiment, we do not apply the weight decay. In this text-to-image experiment, we do not use EMA, following previous works [61]. In the base resolution, we use the adaptive weighting with $\lambda = 4\frac{\|\nabla_{\theta^l}\mathcal{L}_{\text{rec}}\|_2^2}{\|\nabla_{\theta^l}\mathcal{L}_{\text{adv}}\|_2^2}$. Overall, we use the DeepFloyd-IF-I model with 0.9B number of parameters.

Figure 17 compares PaGoDA with the existing baselines.

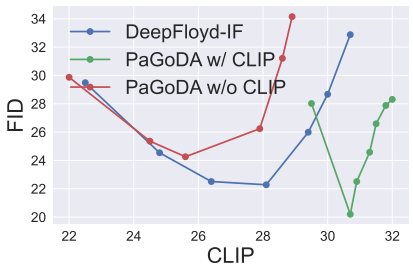


Figure 16: Effect of CLIP regularization.

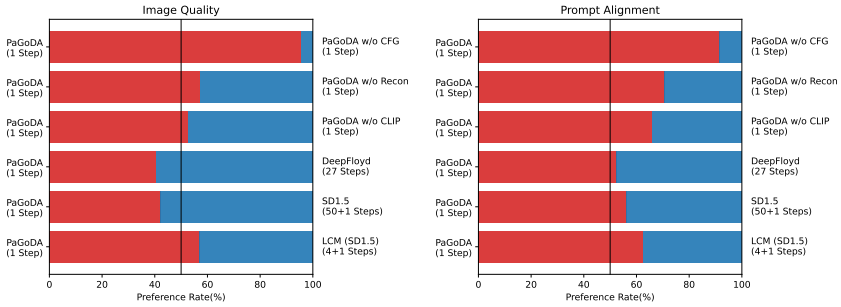


Figure 17: Human evaluation result on T2I with CFG set to be 7 across models.

B Theoretical Analysis

In this section, we present rigorous statements and proofs of all theorems. The theorems are shown for the unconditional generation case (i.e., without the condition $\mathbf{c}$), but the analysis can be extended to the conditional scenario.

B.1 Convergence with PaGoDA's Reconstruction Loss

In Section B.1.1, we introduce the necessary notations and preliminaries. In Sections B.1.2 and B.1.3, we demonstrate that the Wasserstein-1 and Wasserstein-2 discrepancies of the learned density (with PaGoDA's reconstruction loss) from p_{data} are upper bounded by PaGoDA's reconstruction loss and the pre-trained DM's training error. All results are proved for unconditional generation (i.e., without $\mathbf{c}$ as an input), but they can be easily generalized to the conditional case.

B.1.1 Preliminaries of Convergence Analysis

Consider OU process for $t \in [0, T]$, where $T > 0$:

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

Its associated PF-ODE is

$$\mathrm{d}\mathbf{x}_t = \big[-f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\nabla \log p_t(\mathbf{x}_t)\big]\,\mathrm{d}t.$$

We consider $f(t) \equiv 1$ and $g(t) \equiv \sqrt{2}$ for simplicity. That is,

$$\mathrm{d}\mathbf{x}_t = -\mathbf{x}_t \,\mathrm{d}t + \sqrt{2}\,\mathrm{d}\mathbf{w}_t. \tag{5}$$

We recall that PaGoDA's reconstruction loss (unconditional case) is defined as:

$$\mathcal{L}_{\text{rec}}(\boldsymbol{\theta}; \boldsymbol{\phi}_0) := \mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\Big[\big\|\mathbf{x} - G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{z})\big\|_2^2\Big],$$

Here, we use $p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})$ to denote the density obtained by solving the pre-trained teacher DM's empirical PF-ODE forward in time from $t = 0$ to $t = T$:

$$\mathrm{d}\mathbf{x}_t = \big[-f(t)\mathbf{x}_t - \frac{1}{2}g^2(t)\mathbf{s}_{\boldsymbol{\phi}_0}(\mathbf{x}_t, t)\big]\,\mathrm{d}t,$$

where $\mathbf{s}_\phi(\mathbf{x}_t, t)$ indicates the pre-trained DM. We remark that $p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})$ defines a deterministic process.

We take $p_{\text{prior}} := \mathcal{N}\big(\mathbf{0}, (1 - e^{-2T})\mathbf{I}\big)$ as the prior distribution, and define $p_{T,\boldsymbol{\phi}_0} := G_{\boldsymbol{\phi}_0}^{0\to T}\sharp p_{\text{data}}$ as the distribution obtained by solving the teacher-determined empirical PF-ODE forward in time. Let us consider the density obtained by sampling from PaGoDA (trained without GAN) $p_{0,\boldsymbol{\theta}} := G_{\boldsymbol{\theta}}^{T\to 0}\sharp p_{\text{prior}}$. We also let $G^{T\to 0}$ denote the ground truth transition map from T to 0, defined by the PF-ODE.

Conceptually, Theorems B.1 and B.3 demonstrate that

$$W_p(p_{0,\boldsymbol{\theta}}, p_{\text{data}}) \lesssim \mathcal{L}_{\text{rec}}(\boldsymbol{\theta}; \boldsymbol{\phi}_0) + \epsilon_{\text{DM}}, \quad p = 1, 2.$$

This implies that training with PaGoDA's reconstruction loss ensures the learned density $p_{0,\boldsymbol{\theta}} = G_{\boldsymbol{\theta}}^{T\to 0}\sharp p_{\text{prior}}$ is close to p_{data} in Wasserstein distance sense. Moreover, improving the teacher DM to reduce the error ϵ_{DM} is a way to further decrease the discrepancy between $p_{0,\boldsymbol{\theta}}$ and p_{data}.

We remark that the differences between the two theorems primarily lie in the distinct smoothness assumptions on p_{data}.

B.1.2 W_2 Bound with PaGoDA's Reconstruction Loss

Assumption I-1. (i) $m^2 := \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\|\mathbf{x}\|_2^2 < \infty$;

(ii) There is a $\epsilon_{\text{DSM}} > 0$ so that $\sup_{\mathbf{x},t}\|\mathbf{s}_{\boldsymbol{\phi}_0}(\mathbf{x},t) - \nabla \log p_t(\mathbf{x})\|_2^2 \le \epsilon_{\text{DM}}^2$;

(iii) $G_{\boldsymbol{\theta}}^{T\to 0}$ is Lipschitz in $\mathbf{x}$:

$$\Lambda := \sup_{\mathbf{x}\neq\mathbf{y}} \frac{\left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{x}) - G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{y})\right\|_2}{\|\mathbf{x}-\mathbf{y}\|_2} < \infty,$$

for all $\boldsymbol{\theta}$ and T.

(iv) $\log p_{\text{data}}$ is γ-strongly concave with $\gamma > 3/2$:

$$\langle \mathbf{x}-\mathbf{y}, \nabla \log p_{\text{data}}(\mathbf{x}) - \nabla \log p_{\text{data}}(\mathbf{y})\rangle \le -\gamma\|\mathbf{x}-\mathbf{y}\|_2^2,$$

for all $\mathbf{x}$ and $\mathbf{y}$.

Theorem B.1. *Given that Assumption I-1 holds, suppose δ is a positive constant such that $\delta < \frac{e^{-2T}}{3-e^{-2T}}$, and let $h(\gamma,T) := \frac{\gamma}{e^{-2T}+\gamma(1-e^{-2T})} - (1+\delta)$, where it is noted that $h(\gamma,T)$ is also positive. Then*

$$\begin{aligned} W_2(p_{0,\boldsymbol{\theta}}, p_{data}) \le \mathcal{L}_{rec}(\boldsymbol{\theta};\boldsymbol{\phi}_0) &+ \left[\mathbb{E}_{p_{data}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{\frac{1}{2}} \\ &+ \left(\Lambda + e^{-\frac{1}{2}h(\gamma,T)T}\right) W_2\left(p_T, p_{T,\boldsymbol{\phi}_0}\right) \\ &+ \frac{\epsilon_{DM}}{\sqrt{2\delta h(\gamma,T)}}\left(1-e^{-h(\gamma,T)T}\right)^{\frac{1}{2}} + e^{-\frac{T}{2}}m\Lambda. \end{aligned}$$

In particular, if we assume Assumption I-1 (iii) holds also for $G^{T\to 0}$,

$$W_2(p_{0,\boldsymbol{\theta}}, p_{data}) \lesssim \mathcal{L}_{rec}(\boldsymbol{\theta};\boldsymbol{\phi}_0) + \epsilon_{DM} + e^{-\frac{T}{2}}m\Lambda.$$

Here, we use $\lesssim$ to absorb the dependence on the constants T and γ into the inequality.

We present an inequality which is essential for the proof of Theorem B.1.

Lemma B.2 (Proposition 3.5. in [79])**.** *Let P and Q be two distributions on $\mathbb{R}^D$. Suppose that $\log P$ is γ_P-concave and $\log Q$ is γ_Q-concave. Then the convolution of $\log P * Q$ is a $(1/\gamma_P + 1/\gamma_Q)^{-1}$-concave distribution.*

Proof of Theorem B.1. The proof of the theorem is inspired by [80, 81]. Define $p_{T,\boldsymbol{\phi}_0} := G_{\boldsymbol{\phi}_0}^{0\to T}\sharp p_{\text{data}}$, and $p_{0,\boldsymbol{\phi}_0} := G^{T\to 0}\sharp p_{T,\boldsymbol{\phi}_0}$. From the triangle inequality, we have

$$W_2(p_{0,\boldsymbol{\theta}}, p_{\text{data}}) \le \underbrace{W_2(p_{0,\boldsymbol{\theta}}, p_{0,\boldsymbol{\phi}_0})}_{(A)} + \underbrace{W_2(p_{0,\boldsymbol{\phi}_0}, p_{\text{data}})}_{(B)}.$$

For (A), let $\pi(\mathbf{y},\mathbf{z}) \in \Pi\left(p_{\text{prior}}, p_{T,\boldsymbol{\phi}_0}\right)$ be a coupling of p_{prior} and $p_{T,\boldsymbol{\phi}_0}$. Then

$$\begin{aligned} (A) &= W_2\left(G_{\boldsymbol{\theta}}^{T\to 0}\sharp p_{\text{prior}}, G^{T\to 0}\sharp p_{T,\boldsymbol{\phi}_0}\right) \\ &\le \left(\mathbb{E}_{(\mathbf{y},\mathbf{z})\sim\pi}\left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{y}) - G^{T\to 0}(\mathbf{z})\right\|_2^2\right)^{\frac{1}{2}} \\ &\le \underbrace{\left(\mathbb{E}_{(\mathbf{y},\mathbf{z})\sim\pi}\left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{y}) - G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{z})\right\|_2^2\right)^{\frac{1}{2}}}_{(A.1)} + \underbrace{\left(\mathbb{E}_{(\mathbf{y},\mathbf{z})\sim\pi}\left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{z}) - G^{T\to 0}(\mathbf{z})\right\|_2^2\right)^{\frac{1}{2}}}_{(A.2)}. \end{aligned}$$

For (A.1), we can yield

$$
\begin{aligned}
(A.1) &\le \Lambda \min_{\pi \in \Pi\left(p_{\text{prior}}, p_{T,\boldsymbol{\phi}_0}\right)} \left(\mathbb{E}_{(\mathbf{y},\mathbf{z})\sim\pi} \left\|\mathbf{y}-\mathbf{z}\right\|_2^2\right)^{\frac{1}{2}} \\
&= \Lambda W_2\left(p_{\text{prior}}, p_{T,\boldsymbol{\phi}_0}\right) \\
&\le \Lambda W_2\left(p_{\text{prior}}, p_T\right) + \Lambda W_2\left(p_T, p_{T,\boldsymbol{\phi}_0}\right) \\
&\le e^{-\frac{T}{2}} \left(\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \left\|\mathbf{x}_0\right\|_2^2\right)^{\frac{1}{2}} \Lambda + \Lambda W_2\left(p_T, p_{T,\boldsymbol{\phi}_0}\right). \qquad (6)
\end{aligned}
$$

Here, the last inequality is a consequence of the following bound

$$
W_2(p_{\text{prior}}, p_T) \le e^{-\frac{T}{2}} \left(\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \left\|\mathbf{x}_0\right\|_2^2\right)^{\frac{1}{2}},
$$

which holds because p_{prior} is taken as $\mathcal{N}\big(\mathbf{0}, (1-e^{-2T})\mathbf{I}\big)$, and $\mathbf{x}_T \sim p_T$ governed by Eq. (5) admits the expression

$$
\mathbf{x}_T = e^{-T}\mathbf{x}_0 + \int_0^T e^{-(T-s)}\sqrt{2}\,\mathrm{d}\mathbf{w}_s = e^{-T}\mathbf{x}_0 + \mathbf{z}, \quad \mathbf{z} \sim \mathcal{N}\big(\mathbf{0}, (1-e^{-2T})\mathbf{I}\big).
$$

For (A.2), since $p_{T,\boldsymbol{\phi}_0}(\mathbf{z}) = \int p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x}) p_{\text{data}}(\mathbf{x})\,\mathrm{d}\mathbf{x}$, by applying Minkowski inequality we have

$$
\begin{aligned}
(A.2) &= \left(\mathbb{E}_{(\mathbf{y},\mathbf{z})\sim\pi} \left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{z}) - G^{T\to 0}(\mathbf{z})\right\|_2^2\right)^{\frac{1}{2}} \\
&= \left(\mathbb{E}_{\mathbf{z}\sim p_{T,\boldsymbol{\phi}_0}(\mathbf{z})} \left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{z}) - G^{T\to 0}(\mathbf{z})\right\|_2^2\right)^{\frac{1}{2}} \\
&\le \left(\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})} \left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{z}) - \mathbf{x}\right\|_2^2\right)^{\frac{1}{2}} + \left(\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})} \left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right)^{\frac{1}{2}} \\
&= \mathcal{L}_{\text{rec}}(\boldsymbol{\theta};\boldsymbol{\phi}_0) + \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})} \left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{\frac{1}{2}}. \qquad (7)
\end{aligned}
$$

The proof for (B) is motivated by [80]. Consider the following two reverse time PF-ODEs on the interval $[0, T]$

$$
\frac{\mathrm{d}\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}}{\mathrm{d}t} = \hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} + \mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}, T-t), \quad \hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} \sim p_{T,\boldsymbol{\phi}_0}
$$

and

$$
\frac{\mathrm{d}\hat{\mathbf{z}}_t}{\mathrm{d}t} = \hat{\mathbf{z}}_t + \nabla \log p_{T-t}(\hat{\mathbf{z}}_t), \quad \hat{\mathbf{z}}_0 \sim p_T,
$$

with a coupling of $\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} \sim p_{T,\boldsymbol{\phi}_0}$ and $\hat{\mathbf{z}}_0 \sim p_T$ so that $W_2^2(p_{T,\boldsymbol{\phi}_0}, p_T) = \mathbb{E}\left\|\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_0\right\|_2^2$. We notice that $W_2^2(p_{0,\boldsymbol{\phi}_0}, p_{\text{data}}) \le \mathbb{E}\left\|\hat{\mathbf{z}}_{T,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_T\right\|_2^2$. Thus, we need to obtain a upper bound of $\mathbb{E}\left\|\hat{\mathbf{z}}_{T,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_T\right\|_2^2$. Let $u(t) := \mathbb{E}\left\|\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t\right\|_2^2$. Then

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}t}u(t) &= 2\mathbb{E}\langle \hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t, \frac{\mathrm{d}}{\mathrm{d}t}\big(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t\big)\rangle \\
&= 2u(t) + 2\mathbb{E}\Big[\langle \hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t, \mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}, T-t) - \nabla \log p_{T-t}(\hat{\mathbf{z}}_t)\rangle\Big] \\
&= 2u(t) + \underbrace{2\mathbb{E}\Big[\langle \hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t, \mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}, T-t) - \nabla \log p_{T-t}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0})\rangle\Big]}_{(B.1)} \\
&\quad + \underbrace{2\mathbb{E}\Big[\langle \hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t, \nabla \log p_{T-t}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}) - \nabla \log p_{T-t}(\hat{\mathbf{z}}_t)\rangle\Big]}_{(B.2)}. \qquad (8)
\end{aligned}
$$

Let $\delta > 0$, by applying Yang's inequality $ab = \big(\sqrt{2\delta}a\big)\big(\frac{b}{\sqrt{2\delta}}\big) \le \delta a^2 + \frac{b^2}{4\delta}$ to (B.1) for nonnegative a and b, and the Assumption I-1, it becomes

$$
(B.1) \le \delta u(t) + \frac{\epsilon_{\text{DM}}^2}{4\delta}. \qquad (9)
$$

We turn our attention to (B.2). Naively, (B.2) may be naively bounded above by $\mathrm{Lip}\big(\nabla \log p_t(\cdot)\big)u(t)$, where $\mathrm{Lip}\big(\nabla \log p_t(\cdot)\big)$ is the Lipschitz constant of $\nabla \log p_t(\cdot)$ in $\mathbf{x}$. However, we will now derive a sharper bound by incorporating assumptions on the data distribution.

We notice that $p_t(\mathbf{x}_t) = \int p_{t|0}(\mathbf{x}_t|\mathbf{x}_0)p_{\text{data}}(\mathbf{x}_0)\,\mathrm{d}\mathbf{x}_0$, where $p_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}\big(\mathbf{x}_t; e^{-t}\mathbf{x}_0, (1-e^{-2t})\mathbf{I}\big)$ is a transition kernel from 0 to t determined by the forward SDE. Therefore, expressing p_t in convolution form, under Assumption I-1, and leveraging Lemma B.2, we deduce that $\log p_{T-t}$ is a $\gamma/\big(e^{-2(T-t)} + \gamma(1-e^{-2(T-t)})\big)$-strongly concave distribution (see [82]). Hence,

$$(B.2) \le -\frac{\gamma}{e^{-2(T-t)} + \gamma(1-e^{-2(T-t)})}u(t). \tag{10}$$

With the inequalities (9) and (10), we deduce from Eq. (8) that

$$u'(t) \le a(t)u(t) + \frac{\epsilon_{\text{DM}}^2}{2\delta}, \quad \text{where} \quad a(t) := \Big(2 + 2\delta - \frac{2\gamma}{e^{-2(T-t)} + \gamma(1-e^{-2(T-t)})}\Big).$$

By applying Grönwall's inequality, we obtain

$$\begin{aligned}\mathbb{E}\left\|\hat{\mathbf{z}}_{T,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_T\right\|_2^2 &\le e^{A(T)}\mathbb{E}\left\|\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_0\right\|_2^2 + \frac{\epsilon_{\text{DM}}^2}{2\delta}\int_0^T e^{A(T)-A(t)}\,\mathrm{d}t,\\ &= e^{A(T)}W_2^2(p_{T,\boldsymbol{\phi}_0}, p_T) + \frac{\epsilon_{\text{DM}}^2}{2\delta}\int_0^T e^{A(T)-A(t)}\,\mathrm{d}t.\end{aligned} \tag{11}$$

where $A(t) := \int_0^t a(s)\,\mathrm{d}s$.

We aim to find an upper bound for inequality (11) that decays exponentially with respect to T. In $a(t)$, $b(t) := \frac{\gamma}{e^{-2(T-t)}+\gamma(1-e^{-2(T-t)})}$ as a function of t has the derivative as $\frac{2\gamma(\gamma-1)e^{-2(T-t)}}{\left(e^{-2(T-t)}+\gamma(1-e^{-2(T-t)})\right)^2}$. This implies when $\gamma \ge 1$, b's minimum occurs at $b(0) = \frac{\gamma}{\gamma+e^{-2T}(1-\gamma)}$, which implies $a(t) \le 2\big(1+\delta-b(0)\big)$ for all $t \in [0,T]$. Setting $\delta < \frac{e^{-2T}}{3-e^{-2T}}$, which implies $\frac{1}{2} > \frac{\delta}{(1+\delta)e^{-2T}-\delta}$, then $\gamma > \frac{3}{2} = 1 + \frac{1}{2} > 1 + \frac{\delta}{(1+\delta)e^{-2T}-\delta}$ (notice that $(1+\delta)e^{-2T} - \delta > 2\delta$), we can deduce that

$$a(t) \le 1 + \delta - \frac{\gamma}{e^{-2T} + \gamma(1-e^{-2T})} < 0.$$

Let $h(\gamma, T) := \frac{\gamma}{e^{-2T}+\gamma(1-e^{-2T})} - (1+\delta) > 0$. Then we establish that $a(t) \le -h(\gamma,T)$, $A(T) \le -h(\gamma,T)T$, and $A(T) - A(t) \le -h(\gamma,T)t$ which implies $\int_0^T e^{A(T)-A(t)}\,\mathrm{d}t \le 1 - e^{-h(\gamma,T)T}$. By applying the above bounds and inequality (11), (B) becomes

$$\begin{aligned}(B) &\le \left(\mathbb{E}\left\|\hat{\mathbf{z}}_{T,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_T\right\|_2^2\right)^{\frac{1}{2}}\\ &\le \left(e^{-h(\gamma,T)T}W_2^2(p_{T,\boldsymbol{\phi}_0}, p_T) + \frac{\epsilon_{\text{DM}}^2}{2\delta h(\gamma,T)}\big(1-e^{-h(\gamma,T)T}\big)\right)^{\frac{1}{2}}\\ &\le e^{-\frac{1}{2}h(\gamma,T)T}W_2(p_{T,\boldsymbol{\phi}_0}, p_T) + \frac{\epsilon_{\text{DM}}}{\sqrt{2\delta h(\gamma,T)}}\big(1-e^{-h(\gamma,T)T}\big)^{\frac{1}{2}}.\end{aligned} \tag{12}$$

Here, the last inequality is from a simple inequality $\sqrt{a+b} \le \sqrt{a} + \sqrt{b}$ for nonnegative a and b.

By combining inequalities (6), (7), and (12), we obtain

$$\begin{aligned}W_2(p_{0,\boldsymbol{\theta}}, p_{\text{data}}) &\le e^{-\frac{T}{2}}\left(\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)}\left\|\mathbf{x}_0\right\|_2^2\right)^{\frac{1}{2}}\Lambda + \Lambda W_2\big(p_T, p_{T,\boldsymbol{\phi}_0}\big)\\ &\quad + \mathcal{L}_{\text{PaGoDA}}(\boldsymbol{\theta};\boldsymbol{\phi}_0) + \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{\frac{1}{2}}\\ &\quad + e^{-\frac{1}{2}h(\gamma,T)T}W_2(p_{T,\boldsymbol{\phi}_0}, p_T) + \frac{\epsilon_{\text{DM}}}{\sqrt{2\delta h(\gamma,T)}}\big(1-e^{-h(\gamma,T)T}\big)^{\frac{1}{2}}\\ &= \mathcal{L}_{\text{rec}}(\boldsymbol{\theta};\boldsymbol{\phi}_0) + \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{\frac{1}{2}}\end{aligned}$$

$$+ \left(\Lambda + e^{-\frac{1}{2}h(\gamma,T)T}\right) W_2\left(p_T, p_{T,\boldsymbol{\phi}_0}\right)$$
$$+ \frac{\epsilon_{\text{DM}}}{\sqrt{2\delta h(\gamma,T)}}\left(1 - e^{-h(\gamma,T)T}\right)^{\frac{1}{2}} + e^{-\frac{T}{2}} m\Lambda.$$

This shows the first inequality in Theorem B.1.

Now, we show the second inequality in the statement of Theorem B.1. First, we establish an upper bound for $\left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{\frac{1}{2}}$ in terms of ϵ_{DM}. Let $G_{\boldsymbol{\phi}_0}^{0\to T}$ denote the transition map defined by the empirical PF-ODE defined by the teacher $p_{\boldsymbol{\phi}_0}(\mathbf{x}|\mathbf{z})$, and $G^{0\to T}$ denote the ground truth transition map defined by the PF-ODE from 0 to T. Then we have $\mathbf{x} = G^{T\to 0}(G^{0\to T}(\mathbf{x}))$ for all $\mathbf{x} \in \text{supp}(p_{\text{data}})$, and

$$\begin{aligned}\left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{1/2} &= \left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left\|G^{T\to 0}(G^{0\to T}(\mathbf{x})) - G^{T\to 0}(G_{\boldsymbol{\phi}_0}^{0\to T}(\mathbf{x}))\right\|_2^2\right]^{1/2} \\ &\le \Lambda\left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left\|G^{0\to T}(\mathbf{x}) - G_{\boldsymbol{\phi}_0}^{0\to T}(\mathbf{x})\right\|_2^2\right]^{1/2}. \end{aligned} \tag{13}$$

Here, we utilize the assumption that Assumption I-1 (iii) also holds for $G^{T\to 0}$.

Consider the following two forward-time PF-ODEs on the interval $[0, T]$, both starting from $\mathbf{x}_0 \sim p_{\text{data}}$:

$$\frac{\mathrm{d}\mathbf{x}_t}{\mathrm{d}t} = -\mathbf{x}_t - \nabla \log p_t(\mathbf{x}_t), \quad \frac{\mathrm{d}\mathbf{x}_{t,\boldsymbol{\phi}_0}}{\mathrm{d}t} = -\mathbf{x}_{t,\boldsymbol{\phi}_0} - \mathbf{s}_{\boldsymbol{\phi}_0}(\mathbf{x}_{t,\boldsymbol{\phi}_0}, t).$$

By subtracting them and integrating from 0 to t, we obtain

$$\begin{aligned}\left\|\mathbf{x}_t - \mathbf{x}_{t,\boldsymbol{\phi}_0}\right\|_2 &\le \underbrace{\mathbf{x}_0 - \mathbf{x}_{0,\boldsymbol{\phi}_0}}_{\mathbf{0}} + \int_0^t \left\|\left(\mathbf{x}_\tau - \mathbf{x}_{\tau,\boldsymbol{\phi}_0}\right) + \left(\nabla \log p_\tau(\mathbf{x}_\tau) - \mathbf{s}_{\boldsymbol{\phi}_0}(\mathbf{x}_{\tau,\boldsymbol{\phi}_0}, \tau)\right)\right\|_2 \mathrm{d}t \\ &\le \int_0^t \left\|\mathbf{x}_\tau - \mathbf{x}_{\tau,\boldsymbol{\phi}_0}\right\|_2 \mathrm{d}\tau + \epsilon_{\text{DM}} T.\end{aligned}$$

By applying Grönwall's inequality,

$$\left\|\mathbf{x}_t - \mathbf{x}_{t,\boldsymbol{\phi}_0}\right\|_2 \le T e^T \epsilon_{\text{DM}}. \tag{14}$$

Combining the above inequality with inequality (13), it implies

$$\left[\mathbb{E}_{p_{\text{data}}(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2^2\right]^{1/2} \le \Lambda T e^T \epsilon_{\text{DM}}. \tag{15}$$

Next, we derive an upper bound for $W_2(p_{T,\boldsymbol{\phi}_0}, p_T)$ related to ϵ_{DM}. Let $\pi(\hat{\mathbf{z}}, \mathbf{z})$ be a coupling between $\hat{\mathbf{z}} \sim p_{T,\boldsymbol{\phi}_0} = G_{\boldsymbol{\phi}_0}^{0\to T}\sharp p_{\text{data}}$ and $\mathbf{z} \sim p_T = G^{0\to T}\sharp p_{\text{data}}$.

$$W_2^2(p_{T,\boldsymbol{\phi}_0}, p_T) = W_2^2(G_{\boldsymbol{\phi}_0}^{0\to T}\sharp p_{\text{data}}, G^{0\to T}\sharp p_{\text{data}}) \le \mathbb{E}_{\pi(\hat{\mathbf{z}},\mathbf{z})}\left\|\hat{\mathbf{z}} - \mathbf{z}\right\|_2^2 \le \left(T e^T \epsilon_{\text{DM}}\right)^2, \tag{16}$$

where the last inequality is derived from the inequality (14).

Therefore, with the first conclusion of Theorem B.1 and inequalities (15) and (16), we derive

$$W_2(p_{0,\boldsymbol{\theta}}, p_{\text{data}}) \lesssim \mathcal{L}_{\text{rec}}(\boldsymbol{\theta}; \boldsymbol{\phi}_0) + \epsilon_{\text{DM}} + e^{-\frac{T}{2}} m\Lambda.$$

■

The proof can be easily extended in two directions: (1) a more general (VP)-SDE:

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

with $\|f\|_{L^\infty(t;[0,T])}, \|g\|_{L^\infty(t;[0,T])} < \infty$, and (2) truncation at the least time $t = \delta$ (instead of $t = 0$), with an additional argument based on [81]

$$\begin{aligned}W_2(p_\delta, p_{\text{data}}) &\le \left(\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)}\mathbb{E}_{p_{\text{prior}}(\boldsymbol{\xi})}\left\|(1 - e^{-\delta})\mathbf{x}_0 + \sqrt{1 - e^{-2\delta}}\boldsymbol{\xi}\right\|_2^2\right)^{\frac{1}{2}} \\ &\le \left((1 - e^{-\delta})^2 m^2 + (1 - e^{-2\delta})D\right)^{\frac{1}{2}} \\ &\lesssim (\sqrt{D} \vee m)\sqrt{\delta},\end{aligned}$$

where $p_\delta = G^{T\to\delta}\sharp p_{\text{prior}}$.

B.1.3 W_1 Bound with PaGoDA's Reconstruction Loss

Assumption II-1. (i) $m := \mathbb{E}_{p_{\text{data}}(\mathbf{x})} \|\mathbf{x}\|_2 < \infty$;

(ii) There is a $\epsilon_{\text{DSM}} > 0$ so that $\sup_{\mathbf{x},t} \|\mathbf{s}_\phi(\mathbf{x}, t) - \nabla \log p_t(\mathbf{x})\|_2^2 \le \epsilon_{\text{DM}}^2$;

(iii) $G_{\boldsymbol{\theta}}^{T\to 0}$ is Lipschitz in $\mathbf{x}$:

$$\Lambda := \sup_{\mathbf{x}\neq\mathbf{y}} \frac{\left\|G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{x}) - G_{\boldsymbol{\theta}}^{T\to 0}(\mathbf{y})\right\|_2}{\|\mathbf{x}-\mathbf{y}\|_2} < \infty,$$

for all $\boldsymbol{\theta}$ and T.

(iv) $\nabla \log p_t(\cdot)$ is Lipschitz in $\mathbf{x}$ with integrable Lipschitz constant:

$$\Lambda_s(t) := \sup_{\mathbf{x}\neq\mathbf{y}} \frac{\|\nabla \log p_t(\mathbf{x}) - \nabla \log p_t(\mathbf{y})\|_2}{\|\mathbf{x}-\mathbf{y}\|_2} < \infty,$$

and Λ_s is an L^1-integrable function on $(0,\infty)$.

In the following proposition, we prove a variant of Theorem B.1 which does not assume log-concavity of the data density (i.e., Assumption I-1 (iv)).

Theorem B.3 (Variant of Theorem B.1)**.** *Assume that Assumption II-1 holds. Let ν be either the oracle data distribution p_{data} or an empirical distribution $\hat{p}_{data,N} := \frac{1}{N}\sum_{i=1}^N \delta_{\mathbf{x}_i}$, where $\mathbf{x}_i \sim p_{data}$ for $i = 1, \cdots, N$. Let the PaGoDA's reconstruction loss starting from ν be defined as*

$$\mathcal{L}_{rec}(\boldsymbol{\theta}_\nu; \boldsymbol{\phi}_0) := \mathbb{E}_{\nu(\mathbf{x})p_\phi(\mathbf{z}|\mathbf{x})}\left[\left\|\mathbf{x} - G_{\boldsymbol{\theta}_\nu}^{T\to 0}(\mathbf{z})\right\|_2\right].$$

Then we have

$$\begin{aligned} W_1(p_{0,\boldsymbol{\theta}}, \nu) \le\ & \mathcal{L}_{rec}(\boldsymbol{\theta}_\nu; \boldsymbol{\phi}_0) + \mathbb{E}_{\nu(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left[\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2\right] + C_T T \epsilon_{DM} \\ & + \left(C_T + \Lambda\right) W_1(p_{T,\boldsymbol{\phi}_0}, p_T) + e^{-T}\left(\mathbb{E}_{p_{data}(\mathbf{x}_0)} \|\mathbf{x}_0\|_2\right)\Lambda \end{aligned}$$

In particular, if we assume Assumption I-1 (iii) holds also for $G^{T\to 0}$, then for $T = \mathcal{O}\left(\log\left(\frac{m\Lambda}{\epsilon_{DM}}\right)^2\right)$ is sufficiently large, we have

$$W_1(p_{0,\boldsymbol{\theta}}, p_{data}) \lesssim \mathcal{L}_{rec}(\boldsymbol{\theta}; \boldsymbol{\phi}_0) + \epsilon_{DM}.$$

Here, we use $\lesssim$ to absorb the dependence on the constants T and γ into the inequality.

Proof of Theorem B.3. Define $p_{T,\boldsymbol{\phi}_0} := G_{\boldsymbol{\phi}_0}^{0\to T}\sharp\nu$, and $p_{0,\boldsymbol{\phi}_0} := G^{T\to 0}\sharp p_{T,\boldsymbol{\phi}_0}$. From the triangle inequality, we have

$$W_1(p_{0,\boldsymbol{\theta}}, \nu) \le \underbrace{W_1(p_{0,\boldsymbol{\theta}}, p_{0,\boldsymbol{\phi}_0})}_{(A)} + \underbrace{W_1(p_{0,\boldsymbol{\phi}_0}, \nu)}_{(B)}.$$

For (A), by following the similar argument as in Theorem B.1, we can obtain

$$(A) \le e^{-T}\left(\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)} \|\mathbf{x}_0\|_2\right)\Lambda + \Lambda W_1\left(p_T, p_{T,\boldsymbol{\phi}_0}\right) + \mathcal{L}_{\text{PaGoDA}}(\boldsymbol{\theta}_\nu; \boldsymbol{\phi}_0) + \mathbb{E}_{\nu(\mathbf{x})p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left[\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2\right] \tag{17}$$

For (B), by subtracting the following equations and integrating over t from 0 to T,

$$\begin{cases} \frac{\mathrm{d}\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}}{\mathrm{d}t} &= \hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} + \mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}, T-t), \quad \hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} \sim p_{T,\boldsymbol{\phi}_0} \\ \frac{\mathrm{d}\hat{\mathbf{z}}_t}{\mathrm{d}t} &= \hat{\mathbf{z}}_t + \nabla \log p_{T-t}(\hat{\mathbf{z}}_t), \quad \hat{\mathbf{z}}_0 \sim p_T, \end{cases}$$

we will obtain

$$\hat{\mathbf{z}}_{T,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_T = \left(\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_0\right) + \int_0^T \left(\mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0}, T-t) - \nabla \log p_{T-t}(\hat{\mathbf{z}}_t)\right) \mathrm{d}u.$$

Now let $u(t) := \mathbb{E}\left\|\hat{\mathbf{z}}_{t,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_t\right\|_2$. Then

$$\begin{aligned} u(t) \leq\ & u(0) + \mathbb{E}\int_0^t \left\|\mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{\tau,\boldsymbol{\phi}_0}, T-\tau) - \nabla \log p_{T-\tau}(\hat{\mathbf{z}}_\tau)\right\|_2 \mathrm{d}\tau \\ \leq\ & u(0) + \int_0^T \mathbb{E}\left\|\mathbf{s}_{\boldsymbol{\phi}_0}(\hat{\mathbf{z}}_{\tau,\boldsymbol{\phi}_0}, T-\tau) - \nabla \log p_{T-\tau}(\hat{\mathbf{z}}_{\tau,\boldsymbol{\phi}_0})\right\|_2 \mathrm{d}\tau \\ & + \int_0^t \mathbb{E}\left\|\nabla \log p_{T-\tau}(\hat{\mathbf{z}}_{\tau,\boldsymbol{\phi}_0}) - \nabla \log p_{T-\tau}(\hat{\mathbf{z}}_\tau)\right\|_2 \mathrm{d}\tau \\ \leq\ & u(0) + T\epsilon_{\text{DM}} + \int_0^t \Lambda_s(\tau) u(\tau)\, \mathrm{d}\tau, \end{aligned}$$

where $\Lambda_s(t)$ is the Lipschitz constant of $\nabla \log p_t(\cdot)$ in $\mathbf{x}$. By applying integral form of Grönwall's inequality, we get

$$(B) \leq \mathbb{E}\left\|\hat{\mathbf{z}}_{T,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_T\right\|_2 \leq C_T \mathbb{E}\left\|\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_0\right\|_2 + C_T T \epsilon_{\text{DM}} = C_T W_1(p_{T,\boldsymbol{\phi}_0}, p_T) + C_T T \epsilon_{\text{DM}}. \tag{18}$$

where $C_T := \exp\left(\int_0^T \Lambda_s(t)\, \mathrm{d}t\right)$ and the last equality follows from choosing a coupling of $\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} \sim p_{T,\boldsymbol{\phi}_0}$ and $\hat{\mathbf{z}}_0 \sim p_T$ so that $W_1(p_{T,\boldsymbol{\phi}_0}, p_T) = \mathbb{E}\left\|\hat{\mathbf{z}}_{0,\boldsymbol{\phi}_0} - \hat{\mathbf{z}}_0\right\|_2$.

By combining inequalities (17) and (18), we obtain

$$\begin{aligned} W_1(p_{0,\boldsymbol{\theta}}, \nu)\ \leq\ & \mathcal{L}_{\text{rec}}(\boldsymbol{\theta}_\nu; \boldsymbol{\phi}_0) + \mathbb{E}_{\nu(\mathbf{x}) p_{\boldsymbol{\phi}_0}(\mathbf{z}|\mathbf{x})}\left[\left\|\mathbf{x} - G^{T\to 0}(\mathbf{z})\right\|_2\right] + C_T T \epsilon_{\text{DM}} \\ & + \left(C_T + \Lambda\right) W_1(p_{T,\boldsymbol{\phi}_0}, p_T) + e^{-T}\left(\mathbb{E}_{p_{\text{data}}(\mathbf{x}_0)}\left\|\mathbf{x}_0\right\|_2\right)\Lambda. \end{aligned}$$

A similar argument to Theorem B.1 can be applied to obtain the second inequality in the statement of Theorem B.3. ■

B.2 Optimality analysis

In this section, we compare the optimality of the learned distributions resulting from PaGoDA's training and distillation-based training loss, incorporating GAN [7, 6].

PaGoDA's Loss We recall PaGoDA's training objective $\mathcal{L}_{\text{PaGoDA}}$

$$\mathcal{L}_{\text{PaGoDA}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) = \mathcal{L}_{\text{rec}}(G_{\boldsymbol{\theta}}) + \lambda \mathcal{L}_{\text{adv}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}})$$

leverages the reconstruction loss

$$\mathcal{L}_{\text{rec}}(G_{\boldsymbol{\theta}}) = \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\left\|\mathbf{x} - G_{\boldsymbol{\theta}}\left(E(\mathbf{x})\right)\right\|_2^2\right],$$

and adversarial loss

$$\mathcal{L}_{\text{adv}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) = \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\log D_{\boldsymbol{\psi}}(\mathbf{x})\right] + \mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\left[\log\left(1 - D_{\boldsymbol{\psi}}\left(G_{\boldsymbol{\theta}}(\mathbf{z})\right)\right)\right].$$

Knowledge Distillation Loss In the realm of knowledge distillation (KD) methods for DMs, approaches like *local consistency* [5], *global consistency* [6], or *soft consistency* [7] are utilized to learn the noise-to-data trajectory of the teacher DM. Let us consider the global consistency loss as a case study (similar arguments can apply to other distillation objectives), where the teacher's trajectory is obtained by solving its empirical PF-ODE from T to 0. The long jump along the trajectory is represented as $G^{T\to 0}_{\text{teacher}}(\mathbf{z})$, where $\mathbf{z}$ denotes the initial point (noise), T signifies the initial time, and 0 denotes the final time. The output of $G^{T\to 0}_{\text{teacher}}$ corresponds to the estimation of clean data, starting from $\mathbf{z}$.

$$\mathcal{L}_{\text{KD}}(G_{\boldsymbol{\theta}}) := \mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\left[\left\|G^{T\to 0}_{\text{teacher}}(\mathbf{z}) - G_{\boldsymbol{\theta}}\left(\mathbf{z}\right)\right\|_2^2\right].$$

In this context, we abuse the notation $G_{\boldsymbol{\theta}}(\mathbf{z})$ to denote the generator for KD.

The training of KD can also incorporate adversarial loss for enhanced performance [7, 27]. We represent the combined loss as:

$$\mathcal{L}_{\text{KD+GAN}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}) := \mathcal{L}_{\text{KD}}(G_{\boldsymbol{\theta}}) + \mathcal{L}_{\text{adv}}(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}}).$$

Theorem B.4. *Let p_{ϕ_0} be the density determined the teacher DM. Suppose that GAN admits an optimal discriminator D^*.*

- *In PaGoDA, assume that the network parametrized generator class $\{G_{\boldsymbol{\theta}}\}$ is expressive enough so that it can simultaneously optimize both $\mathcal{L}_{rec}(G_{\boldsymbol{\theta}})$ and $\mathcal{L}_{adv}(G_{\boldsymbol{\theta}}; D^*)$ with a same minimizer. Namely, $\arg\min_{\boldsymbol{\theta}}\{\mathcal{L}_{rec}(G_{\boldsymbol{\theta}})\} \cap \arg\min_{\boldsymbol{\theta}}\{\mathcal{L}_{adv}(G_{\boldsymbol{\theta}}; D^*)\} \neq \emptyset$. Then*

$$p_{\boldsymbol{\theta}^*,PaGoDA} := G_{\boldsymbol{\theta}^*,PaGoDA}\sharp p_{prior} = p_{data}.$$

- *In contrast, suppose that $p_{\phi_0} \neq p_{data}$, then under similar conditions for KD+GAN where $\arg\min_{\boldsymbol{\theta}}\{\mathcal{L}_{KD}(G_{\boldsymbol{\theta}})\} \cap \arg\min_{\boldsymbol{\theta}}\{\mathcal{L}_{adv}(G_{\boldsymbol{\theta}}; D^*)\} \neq \emptyset$, there is no minimizer $\boldsymbol{\theta}^*$ so that $p_{\boldsymbol{\theta}^*,KD+GAN} := G_{\boldsymbol{\theta}^*,KD+GAN}\sharp p_{prior} = p_{data}$.*

The first part of the proof of the theorem follows from the following Lemma.

Lemma B.5. *If $\arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\} \neq \emptyset$, then $\arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\} = \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$.*

Proof. First, we prove the relationship $\arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\} \supseteq \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$. Indeed, it holds without additional assumption. Suppose that $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$. Then for any $\boldsymbol{\theta}$, we have $f(\boldsymbol{\theta}) \geq f(\boldsymbol{\theta}^*)$ and $g(\boldsymbol{\theta}) \geq g(\boldsymbol{\theta}^*)$, which implies $f(\boldsymbol{\theta}) + g(\boldsymbol{\theta}) \geq f(\boldsymbol{\theta}^*) + g(\boldsymbol{\theta}^*)$. That is, $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\}$.

On the other hand, suppose that $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\}$. We want to prove that $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$. Let $\boldsymbol{\theta}^*_\cap \in \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$, where we notice that the existence of $\boldsymbol{\theta}^*_\cap$ is guaranteed by the assumption. In particular, we have $f(\boldsymbol{\theta}^*) \geq f(\boldsymbol{\theta}^*_\cap)$ and $g(\boldsymbol{\theta}^*) \geq g(\boldsymbol{\theta}^*_\cap)$. Then for any $\boldsymbol{\theta}$, we have

$$\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\} = f(\boldsymbol{\theta}^*) + g(\boldsymbol{\theta}^*) \geq f(\boldsymbol{\theta}^*_\cap) + g(\boldsymbol{\theta}^*_\cap) \geq \min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\}.$$

Thus, $\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta}) + g(\boldsymbol{\theta})\} = f(\boldsymbol{\theta}^*) + g(\boldsymbol{\theta}^*) = f(\boldsymbol{\theta}^*_\cap) + g(\boldsymbol{\theta}^*_\cap)$ and

$$\big[f(\boldsymbol{\theta}^*) - f(\boldsymbol{\theta}^*_\cap)\big] + \big[g(\boldsymbol{\theta}^*) - g(\boldsymbol{\theta}^*_\cap)\big] = 0.$$

This implies $f(\boldsymbol{\theta}^*) = f(\boldsymbol{\theta}^*_\cap) = \min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\}$ and $g(\boldsymbol{\theta}^*) = g(\boldsymbol{\theta}^*_\cap) = \min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$, as the individual terms are nonnegative. Therefore, $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}}\{f(\boldsymbol{\theta})\} \cap \arg\min_{\boldsymbol{\theta}}\{g(\boldsymbol{\theta})\}$, which concludes the proof.

□

Proof of Theorem B.4. With the lemma above, let $\boldsymbol{\theta}^* \in \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{PaGoDA}}(G_{\boldsymbol{\theta}}, D^*)$. Consequently, $\boldsymbol{\theta}^*$ should also simultaneously minimize both $\mathcal{L}_{\text{rec}}$ and $\mathcal{L}_{\text{adv}}$. Minimizing $\mathcal{L}_{\text{rec}}$ implies that $p_{\boldsymbol{\theta}^*,\text{PaGoDA}} = G_{\boldsymbol{\theta}^*,\text{PaGoDA}}\sharp p_{T,\phi_0}$, where p_{T,ϕ_0} represents the density derived from solving the teacher's empirical PF-ODE forward, starting from p_{data}. On the other hand, optimizing $\mathcal{L}_{\text{adv}}$ implies that $p_{\boldsymbol{\theta}^*,\text{PaGoDA}} = p_{\text{data}}$ by applying Theorem 1 in [29]. This establishes the first part of the theorem.

In the second part, suppose on the contrary that there is a minimizer $\boldsymbol{\theta}^*$ of $\mathcal{L}_{\text{KD+GAN}}$ such that $p_{\boldsymbol{\theta}^*,\text{KD+GAN}} = p_{\text{data}}$. Again, by applying the above lemma, we infer that $\boldsymbol{\theta}^*$ should also minimize $\mathcal{L}_{\text{KD}}$ (and $\mathcal{L}_{\text{adv}}$). This implies that $p_{\boldsymbol{\theta}^*,\text{KD+GAN}} = p_{\phi_0}$. However, this contradicts our assumption that $p_{\text{data}} \neq p_{\phi_0}$. Thus, such a minizer does not exist and the second part of the theorem is proven.

■

We remark that (1) optimality of $\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta})$ may not be unique in $\boldsymbol{\theta}$, and that (2) the first part of the theorem can be directly extended to scenarios involving downsampling in the encoder .

B.3 Stability Analysis

B.3.1 Preliminaries of Dynamical System

To study its convergence and stability, we first introduce the prerequisites for Lyapunov stability [83, 84] in a general setup. Let $\mathcal{F}\colon \Xi \to \Xi$ be a continuously differentiable operator (that is, $\mathscr{C}^1$ operator), where $\Omega \subset \mathbb{R}^N$. We consider the discrete iteration dynamical system defined by

$$\boldsymbol{\xi}_{k+1} = \mathcal{F}(\boldsymbol{\xi}_k) \quad \text{with } \boldsymbol{\xi}_0 \in \Omega.$$

Namely, $\boldsymbol{\xi}_{k+1} = \mathcal{F}^{(k)}(\boldsymbol{\xi}_0) := \underbrace{\mathcal{F} \circ \cdots \circ \mathcal{F}}_{\text{k-copies}}(\boldsymbol{\xi}_0)$. A point $\boldsymbol{\xi}^* \in \Omega$ is called a *fixed point* or *equilibrium* (we use the terms interchangeably) of $\mathcal{F}$ if $\boldsymbol{\xi}^* = \mathcal{F}(\boldsymbol{\xi}^*)$. The stability and convergence analysis focuses on how the dynamical system $\mathcal{F}^{(k)}(\boldsymbol{\xi}_0)$ approaches a fixed point as iterations k are sufficiently large.

Definition B.1. (Stability [84]) Let $\boldsymbol{\xi}^*$ be an equilibrium of the $\mathscr{C}^1$ operator $\mathcal{F}\colon \Omega \to \Omega$. The equilibrium $\boldsymbol{\xi}^*$ is said to be

- *stable* if for every $\epsilon > 0$ there is a $\delta > 0$ so that whenever $\|\boldsymbol{\xi} - \boldsymbol{\xi}^*\|_2 < \delta$, we have $\left\|\mathcal{F}^{(k)}(\boldsymbol{\xi}) - \boldsymbol{\xi}^*\right\|_2 < \epsilon$ for all $k \in \mathbb{N} \cup \{0\}$.
- *asymptotically stable* if $\boldsymbol{\xi}^*$ is stable, and there is a $\delta > 0$ so that whenever $\|\boldsymbol{\xi} - \boldsymbol{\xi}^*\|_2 < \delta$, we have $\lim_{k\to\infty} \left\|\mathcal{F}^{(k)}(\boldsymbol{\xi}) - \boldsymbol{\xi}^*\right\|_2 = 0$.
- *exponentially stable* if $\boldsymbol{\xi}^*$ is asymptotically stable, and there is a $\delta > 0$ and $\alpha, \beta > 0$ so that whenever $\|\boldsymbol{\xi} - \boldsymbol{\xi}^*\|_2 < \delta$, we have $\left\|\mathcal{F}^{(k)}(\boldsymbol{\xi}) - \boldsymbol{\xi}^*\right\|_2 \leq \alpha \|\boldsymbol{\xi} - \boldsymbol{\xi}^*\|_2 e^{-\beta k}$ for all $k \in \mathbb{N} \cup \{0\}$. The largest $\beta > 0$ that satisfies the inequality for exponential stability is referred to as the *rate of convergence*.

Let Γ be a subset of the set of all equilibria. We say the dynamical system $\mathcal{F}^{(k)}$ *locally converges on* Γ if $\mathcal{F}^{(k)}$ is exponentially stable at any point in Γ.

The intuitions of the above stability notions are

- A *stable* equilibrium indicates that if an initialization is within some δ-neighborhood of the equilibrium, the iterations starting from that initialization will always remain within an ϵ-neighborhood of the equilibrium, for any arbitrarily chosen ϵ.
- An *asymptotically stable* equilibrium indicates that iterations starting near the equilibrium not only remain close but ultimately converge to the equilibrium.
- An *asymptotically stable* equilibrium indicates that the iterations not only converge but do so at a rate no slower than the rate $e^{-\beta k}$ with respective to iteration step k.

Analyzing the eigenvalues of the Jacobian $\nabla_{\boldsymbol{\xi}} \mathcal{F}(\boldsymbol{\xi}^*)$ of the operator $\mathcal{F}$ at an equilibrium $\boldsymbol{\xi}^*$ is a crucial tool for studying stability. In principle [83, 84], if we can ensure that the Jacobian of $\mathcal{F}$ at some equilibrium has only eigenvalues with strictly negative real parts, then the dynamical system $\mathcal{F}^{(k)}$ is asymptotically stable at that equilibrium. In particular, we refer to a matrix as a *Hurwitz matrix* if all its eigenvalues have strictly negative real parts.

In the following lemma, we present a necessary condition to ensure that a special class of matrices will be Hurwitz.

Lemma B.6. *(Necessary condition for a Hurwitz matrix [21]) Consider the following matrix $\mathcal{J} \in \mathbb{R}^{(N+M)\times(N+M)}$ with $P \in \mathbb{R}^{N\times N}$, $Q \in \mathbb{R}^{M\times M}$, and $B \in \mathbb{R}^{M\times N}$.*

$$\mathcal{J} = \begin{bmatrix} P & -B^T \\ B & Q \end{bmatrix}.$$

Suppose that B is full rank. Then all eigenvalues of $\mathcal{J}$ have negative real part, if either (1) P is negative definite and Q is negative semi-definite, or (2) P is negative semi-definite and Q is negative definite.

B.3.2 Preliminaries for Analysis of PaGoDA Training

We consider PaGoDA's training, integrating reconstruction and adversarial losses with a weight $\eta > 0$.

$$\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\eta \|\mathbf{x} - G_{\boldsymbol{\theta}}(E(\mathbf{x}))\|_2^2 + f(D_{\boldsymbol{\psi}}(\mathbf{x}))\right] + \mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\left[f(-D_{\boldsymbol{\psi}}(\mathbf{x}))\right] \tag{19}$$

$$= \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\eta \|\mathbf{x} - G_{\boldsymbol{\theta}}(E(\mathbf{x}))\|_2^2 + f(D_{\boldsymbol{\psi}}(\mathbf{x}))\right] + \mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\left[f(-D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))\right]. \tag{20}$$

Here, $f\colon \mathbb{R} \to \mathbb{R}$ is a continuous differentiable function. In the vanilla GAN [29], the f-function is taken as $f(u) := -\log\big(1 + \exp(-u)\big)$, where $f'(u) = \exp(-u)/\big(1 + \exp(-u)\big) > 0$ and

$f''(u) = -\exp(-u)/\big(1+\exp(-u)\big) < 0$ for all $u \in \mathbb{R}$. We maintain the generality of f and will prove the training stability of PaGoDA across a wide class of f.

The velocity field $\mathbf{v}(\boldsymbol{\theta}, \boldsymbol{\psi})$ corresponding to the gradient descent update is

$$\mathbf{v}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \begin{bmatrix} -\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) \\ \nabla_{\boldsymbol{\psi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) \end{bmatrix}.$$

Gradient descent is a special case of fixed-point iteration. Now, we specify the operator $\mathcal{F}$ as an alternative gradient descent operator. That is, we consider $\mathcal{F}_h := \mathcal{F}_{D,h} \circ \mathcal{F}_{G,h}$ with a learning rate $h > 0$. Here,

$$\mathcal{F}_{G,h}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \begin{bmatrix} \boldsymbol{\theta} - h\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) \\ \boldsymbol{\psi} \end{bmatrix} \quad \text{and} \quad \mathcal{F}_{D,h}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \begin{bmatrix} \boldsymbol{\theta} \\ \boldsymbol{\psi} + h\nabla_{\boldsymbol{\psi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) \end{bmatrix}.$$

A point $(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$ is called an *equilibrium* of the system defined by $\mathbf{v}$ if $\mathbf{v}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) = 0$ (equivalently, $\mathcal{F}_h(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) = 0$). We can analyze the learning dynamic via the Jacobian matrix of $\mathbf{v}(\boldsymbol{\theta}, \boldsymbol{\psi})$ which is defined as the following:

$$\mathcal{J}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \begin{bmatrix} -\nabla^2_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) & -\nabla^2_{\boldsymbol{\theta},\boldsymbol{\psi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) \\ \nabla^2_{\boldsymbol{\theta},\boldsymbol{\psi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) & \nabla^2_{\boldsymbol{\psi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) \end{bmatrix}.$$

The following proposition relates Lemma B.8 to the stability of the gradient descent operator $\mathcal{F}_h$, serving as the main tool to prove the training stability of PaGoDA in Theorem B.9.

Lemma B.7. *(Locally stable on manifold – modification of [21]) Suppose that the gradient descent operator $\mathcal{F}_h = \mathcal{F}_h(\boldsymbol{\upsilon}, \boldsymbol{\omega})$ is a $\mathscr{C}^1$ mapping. Let $(\boldsymbol{\upsilon}^*, \boldsymbol{\omega}^*)$ be an equilibrium (fixed point) of $\mathcal{F}_h$. Assume that there is a neighborhood Ω of $\boldsymbol{\omega}^*$ so that $\mathcal{F}_h$ admits equilibrium on $\{\boldsymbol{\upsilon}^*\} \times \Omega$:*

$$\mathcal{F}_h(\boldsymbol{\upsilon}^*, \boldsymbol{\omega}) = (\boldsymbol{\upsilon}^*, \boldsymbol{\omega}) \quad \textit{for all } \omega \in \Omega.$$

If all the eigenvalues of $\mathcal{J} := \nabla_{\boldsymbol{\upsilon}} \mathcal{F}_h(\boldsymbol{\upsilon}^*, \boldsymbol{\omega}^*)$ have negative real parts, then for a sufficiently small learning rate h, the gradient descent iteration defined by $\mathcal{F}_h$ locally converges on $\Gamma := \{(\boldsymbol{\upsilon}^*, \boldsymbol{\omega}) \big| \omega \in \Omega\}$ with a rate of convergence $|\lambda_{max}|$. Here, λ_{max} denotes the eigenvalue of $\mathcal{J}$ with the largest absolute value.

Proof of Lemma B.7. This proposition is followed by Lemma A.5. and Theorem A.3. of [21]. ■

B.3.3 PaGoDA's Training is Stable

Proving PaGoDA's stability involves two steps: First, derive the components of First, deriving the components of $\mathcal{J}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$. Second, verify that these components satisfy Lemma B.6. After these, we can apply Lemma B.7 to conclude PaGoDA's training stability whenever the learning rate $h > 0$ is sufficiently small.

Assumption III-1. (i) E is not an identity map.

(ii) At $\boldsymbol{\theta}^*$, $p_{\boldsymbol{\theta}^*} = p_{\text{data}}$, and $\mathbf{x} = G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))$ for a.e. $\mathbf{x} \in \text{supp}\big(p_{\text{data}}\big)$.

(iii) At $\boldsymbol{\psi}^*$, $D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ and $\nabla_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ for $\mathbf{x} \in \text{supp}(p_{\text{data}})$.

Lemma B.8. *Suppose that Assumption III-1 holds for an equilibrium $(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$. Then the Jacobian at the equilibrium can be computed as*

$$\mathcal{J}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) = \begin{bmatrix} K_{GG} & -K_{DG}^T \\ K_{DG} & K_{DD} \end{bmatrix}.$$

Here, and

$$\begin{aligned} K_{GG} =& -2\eta \mathbb{E}_{p_{data}(\mathbf{x})} \big[\nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))^T \cdot \nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))\big] \\ &+ f'(0) \mathbb{E}_{p_{prior}(\mathbf{z})} \big[\nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(\mathbf{z})^T \cdot \nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(G_{\boldsymbol{\theta}^*}(\mathbf{z})) \cdot \nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(\mathbf{z})\big]. \\ K_{DG} =& -f'(0) \nabla_{\boldsymbol{\theta}} \mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})} \big[\nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})\big]\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} \\ K_{DD} =& 2f''(0) \mathbb{E}_{p_{data}(\mathbf{x})} \big[\nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) \cdot \nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})^T\big]. \end{aligned}$$

Proof of Lemma B.8. We first compute the gradients of $\mathcal{L}$ in terms of $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$, where we utilize the formulations Eqs. (19) and (20), respectively.

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\psi}) =& -2\eta\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\langle \mathbf{x}-G_{\boldsymbol{\theta}}(E(\mathbf{x})), \nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(E(\mathbf{x}))\rangle\big]\\ &-\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\big[f'(-D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))\cdot\nabla_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(\mathbf{z})\big]. \qquad (21)\\ \nabla_{\boldsymbol{\psi}}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\psi}) =& \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[f'(D_{\boldsymbol{\psi}}(\mathbf{x}))\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})\big]-\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[f'(-D_{\boldsymbol{\psi}}(\mathbf{x}))\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})\big]. \qquad (22)\end{aligned}$$

$$\begin{aligned}\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\psi}) =& 2\eta\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\langle\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(E(\mathbf{x})),\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(E(\mathbf{x}))\rangle\big]-2\eta\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\langle\mathbf{x}-G_{\boldsymbol{\theta}}(E(\mathbf{x})),\nabla^2_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(E(\mathbf{x}))\rangle\big]\\ &+\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\big[f''(-D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))\cdot\nabla_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(\mathbf{z})\cdot\nabla_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(\mathbf{z})\big]\\ &-\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\big[f'(-D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(\mathbf{z})^T\cdot\nabla^2_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(\mathbf{z})\big]\\ &-\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\big[f'(-D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z})))\cdot\nabla_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}}(\mathbf{z}))\cdot\nabla^2_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(\mathbf{z})\big].\end{aligned}$$

According to Assumption III-1 (ii) and (iii), we have

$$\begin{aligned}\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^*,\boldsymbol{\psi}^*) =& 2\eta\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))^T\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))\big]\\ &-f'(0)\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\big[\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(\mathbf{z})^T\cdot\nabla^2_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}^*}(\mathbf{z}))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(\mathbf{z})\big].\end{aligned}$$

Thus, we obtain

$$\begin{aligned}K_{GG} =& -\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^*,\boldsymbol{\psi}^*)\\ =& -2\eta\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))^T\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))\big]+f'(0)\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\big[\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(\mathbf{z})^T\cdot\nabla^2_{\mathbf{x}}D_{\boldsymbol{\psi}}(G_{\boldsymbol{\theta}^*}(\mathbf{z}))\cdot\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}^*}(\mathbf{z})\big].\end{aligned}$$

To compute K_{DG}, we first derive $\nabla_{\boldsymbol{\theta}}\mathcal{L}$ from Eq. (19) as

$$\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\psi}) = -2\eta\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\langle\mathbf{x}-G_{\boldsymbol{\theta}}(E(\mathbf{x})),\nabla_{\boldsymbol{\theta}}G_{\boldsymbol{\theta}}(E(\mathbf{x}))\rangle\big]+\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[f(-D_{\boldsymbol{\psi}}(\mathbf{x}))\big].$$

Thus, we can compute

$$\nabla^2_{\boldsymbol{\theta},\boldsymbol{\psi}}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\psi}) = -\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[f'(-D_{\boldsymbol{\psi}}(\mathbf{x}))\cdot\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})\big],$$

and hence,

$$K_{DG} = \nabla^2_{\boldsymbol{\theta},\boldsymbol{\psi}}\mathcal{L}(\boldsymbol{\theta}^*,\boldsymbol{\psi}^*) = -f'(0)\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}^*}(\mathbf{x})\big]\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*}.$$

To compute K_{DD}, we can obtain from Eq. (20) that

$$\begin{aligned}\nabla^2_{\boldsymbol{\psi}}\mathcal{L}(\boldsymbol{\theta},\boldsymbol{\psi}) =& \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[f''(D_{\boldsymbol{\psi}}(\mathbf{x}))\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})\cdot\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})^T\big]\\ &+\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[f''(-D_{\boldsymbol{\psi}}(\mathbf{x}))\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})\cdot\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x})^T\big]\\ &+\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[f'(D_{\boldsymbol{\psi}}(\mathbf{x}))\nabla^2_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x}))\big]-\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[f'(-D_{\boldsymbol{\psi}}(\mathbf{x}))\nabla^2_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}}(\mathbf{x}))\big].\end{aligned}$$

Hence, by using Assumption III-1 (ii) and (iii), we get

$$K_{DD} = \nabla^2_{\boldsymbol{\psi}}\mathcal{L}(\boldsymbol{\theta}^*,\boldsymbol{\psi}^*) = 2f''(0)\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}^*}(\mathbf{x})\cdot\nabla_{\boldsymbol{\psi}}D_{\boldsymbol{\psi}^*}(\mathbf{x})^T\big].$$

■

We consider the following two sets

$$\begin{aligned}\mathcal{M}_G :=& \big\{\boldsymbol{\theta}\big|p_{\boldsymbol{\theta}}=p_{\text{data}},\ \mathbf{x}=G_{\boldsymbol{\theta}}(E(\mathbf{x}))\text{ for a.e. }\mathbf{x}\in\text{supp}\big(p_{\text{data}}\big)\big\}\\ \mathcal{M}_D :=& \big\{\boldsymbol{\psi}\big|S(\boldsymbol{\psi})=0\big\},\end{aligned}$$

where $S(\boldsymbol{\psi}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\,|D_{\boldsymbol{\psi}}(\mathbf{x})|^2+\|\nabla_{\mathbf{x}}D_{\boldsymbol{\psi}}(\mathbf{x})\|_2^2\,\big]$. Also, we let $\mathcal{T}_{\boldsymbol{\psi}^*}\mathcal{M}_D$ denote the tangent space of $\mathcal{M}_D$ at $\boldsymbol{\psi}^*$.

Assumption III-2. (i) The second continuously differentiable function $f:\mathbb{R}\to\mathbb{R}$ satisfies: $f'(0)>0$ and $f''(0)<0$.

(ii) There is a $\delta>0$ so that $\mathcal{M}_G\cap\mathbb{B}_\delta(\boldsymbol{\theta}^*)$ and $\mathcal{M}_D\cap\mathbb{B}_\delta(\boldsymbol{\psi}^*)$ are $\mathscr{C}^1$ manifolds.

(iii) $\nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))^T \cdot \nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(E(\mathbf{x}))$ is positive definite, for all $\mathbf{x} \in \operatorname{supp}\big(p_{\text{data}}\big)$.

(iv) $\partial_{\mathbf{w}} h(\boldsymbol{\psi}^*) \neq 0$ for any $\mathbf{w} \notin \mathcal{T}_{\boldsymbol{\psi}^*}\mathcal{M}_D$, where $h(\boldsymbol{\psi}) := \nabla_{\boldsymbol{\theta}} \mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[D_{\boldsymbol{\psi}}(\mathbf{x})\big]\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*}$.

(v) $\mathbf{w}^T \nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x})\mathbf{w} \geq 0$, for all $\mathbf{w} \notin \mathcal{T}_{\boldsymbol{\theta}^*}\mathcal{M}_G$ and $\mathbf{x} \in \operatorname{supp}(p_{\text{data}})$.
Remark. Two special cases are either (v-1) $\nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ for $\mathbf{x} \in \operatorname{supp}(p_{\text{data}})$, or (v-2) $\mathbf{w}^T \nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x})\mathbf{w} > 0$, for all $\mathbf{w} \notin \mathcal{T}_{\boldsymbol{\theta}^*}\mathcal{M}_G$ and $\mathbf{x} \in \operatorname{supp}(p_{\text{data}})$.

Theorem B.9. *Suppose that Assumptions III-1 and III-2 hold for an equilibrium $(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$ and $\eta > 0$ is sufficiently large. Then the alternative gradient descent iteration $\mathcal{F}_h$ described in Section B.3.2 is locally convergent on $\mathcal{M}_G \times \mathcal{M}_D$ for a sufficiently small learning rate $h > 0$.*

Proof of Theorem B.9. The argument is motivated by [21]. We notice that $\mathcal{M}_G \times \mathcal{M}_D$ is a subset of all equilibria of the operators $\mathcal{F}_h$ (or $\mathbf{v}(\boldsymbol{\theta}, \boldsymbol{\psi})$). This is because that for any $(\boldsymbol{\theta}, \boldsymbol{\psi}) \in \mathcal{M}_G \times \mathcal{M}_D$, we have $p_{\boldsymbol{\theta}} = p_{\text{data}}$, $\mathbf{x} = G_{\boldsymbol{\theta}}(E(\mathbf{x}))$, $D_{\boldsymbol{\psi}}(\mathbf{x}) = 0$, and $\nabla_{\mathbf{x}} D_{\boldsymbol{\psi}}(\mathbf{x}) = \mathbf{0}$ for $\mathbf{x} \in \operatorname{supp}(p_{\text{data}})$. From Eqs. (21) and (22), we then can obtain $\nabla_{\boldsymbol{\theta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) = \nabla_{\boldsymbol{\psi}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\psi}) = 0$, meaning $(\boldsymbol{\theta}, \boldsymbol{\psi})$ is an equilibrium.

Now, we show that the alternating gradient descent converges locally on $\mathcal{M}_G \times \mathcal{M}_D$ by verifying Lemma B.8 is fulfilled, and hence, Lemma B.7 can be applied. Let $(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) \in \mathcal{M}_G \times \mathcal{M}_D$. There is a $\mathscr{C}^1$-diffeomorphism Ψ that transforms a neighborhood of $(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$ onto an open set in $\mathbb{R}^{(N+M)}$ due to Assumption III-2 (ii). More precisely, we can compute the relation of $\mathcal{F}_h$ and $\mathbf{v}$ after the Ψ-reparametrization. Let $\boldsymbol{\zeta} := \Psi(\boldsymbol{\theta}, \boldsymbol{\psi})$, and

$$\begin{aligned}
\mathcal{F}_h^{\Psi}(\boldsymbol{\zeta}) &:= \Psi \circ \mathcal{F}_h \circ \Psi^{-1}(\boldsymbol{\zeta}) \\
\mathbf{v}^{\Psi}(\boldsymbol{\zeta}) &:= \Psi'(\boldsymbol{\theta}, \boldsymbol{\psi}) \cdot \big(\mathbf{v} \circ \Psi^{-1}(\boldsymbol{\zeta})\big).
\end{aligned}$$

Then

$$\begin{aligned}
\nabla_{\zeta} \mathcal{F}_h^{\Psi}(\boldsymbol{\zeta}^*) &= \nabla_{\boldsymbol{\theta},\boldsymbol{\psi}} \Psi(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) \cdot \nabla_{\boldsymbol{\theta},\boldsymbol{\psi}} \mathcal{F}_h(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) \cdot \nabla_{\boldsymbol{\theta},\boldsymbol{\psi}} \Psi(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)^{-1} \\
\nabla_{\zeta} \mathbf{v}^{\Psi}(\boldsymbol{\zeta}^*) &= \nabla_{\boldsymbol{\theta},\boldsymbol{\psi}} \Psi(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) \cdot \nabla_{\boldsymbol{\theta},\boldsymbol{\psi}} \mathbf{v}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) \cdot \nabla_{\boldsymbol{\theta},\boldsymbol{\psi}} \Psi(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)^{-1}.
\end{aligned}$$

We remark that similar matrices have identical ranks and spectrum. Therefore, without loss of the generality, we can assume that $(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) = (\mathbf{0}_N, \mathbf{0}_M) \in \mathbb{R}^N \times \mathbb{R}^M$, and

$$\begin{aligned}
\mathcal{M}_G &= \mathcal{T}_{\boldsymbol{\theta}^*}\mathcal{M}_G = \{0\}^{N_G} \times \mathbb{R}^{N-N_G} \\
\mathcal{M}_D &= \mathcal{T}_{\boldsymbol{\psi}^*}\mathcal{M}_D = \{0\}^{M_D} \times \mathbb{R}^{M-M_D}.
\end{aligned}$$

We write the new parameterizations as $\boldsymbol{\theta} := (\boldsymbol{\upsilon}_G, \boldsymbol{\omega}_G) \in \mathbb{R}^{N_G} \times \mathbb{R}^{N-N_G}$ and $\boldsymbol{\psi} := (\boldsymbol{\upsilon}_D, \boldsymbol{\omega}_D) \in \mathbb{R}^{M_D} \times \mathbb{R}^{M-M_D}$. For simplicity, we write $\mathbf{v}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \mathbf{v}(\boldsymbol{\upsilon}_G, \boldsymbol{\omega}_G, \boldsymbol{\upsilon}_D, \boldsymbol{\omega}_D)$. To apply Lemma B.7, we now aim to show that $\nabla_{(\boldsymbol{\upsilon}_G, \boldsymbol{\upsilon}_D)} \mathbf{v}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$ only admits eigenvalues with negative real parts. From Lemma B.8,

$$\nabla_{(\boldsymbol{\upsilon}_G, \boldsymbol{\upsilon}_D)} \mathbf{v}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*) = \begin{bmatrix} \hat{K}_{GG} & -\hat{K}_{DG}^T \\ \hat{K}_{DG} & \hat{K}_{DD} \end{bmatrix}.$$

Here, $\hat{K}_{GG}$, $\hat{K}_{DG}$, and $\hat{K}_{DD}$ represent submatrices of K_{GG}, K_{DG}, and K_{DD}, respectively, with coordinates $(\boldsymbol{\upsilon}_G, \boldsymbol{\upsilon}_D)$, indicating the Jacobian of $\mathbf{v}$ with derivatives taken along the $\boldsymbol{\upsilon}_G$ and $\boldsymbol{\upsilon}_D$ directions.

First of all, we show that K_{DD} is generally negative semi-definite. Let $\boldsymbol{\xi} \in \mathbb{R}^{(N+M)}$ be any vector. Then

$$\begin{aligned}
\boldsymbol{\xi}^T K_{DD} \boldsymbol{\xi} &= 2f''(0) \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\boldsymbol{\xi}^T \nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) \cdot \nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})^T \boldsymbol{\xi}\big] \\
&= 2f''(0) \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[\big(\nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})^T \boldsymbol{\xi}\big)^T \cdot \nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})^T \boldsymbol{\xi}\big] \leq 0,
\end{aligned}$$

because $f''(0) < 0$ from Assumption III-2 (i). Thus, for any $\hat{\boldsymbol{\xi}}_G \in \mathbb{R}^{N_G}$ and $\hat{\boldsymbol{\xi}}_D \in \mathbb{R}^{M_D}$ if we consider $\hat{\boldsymbol{\xi}} := (\hat{\boldsymbol{\xi}}_G, \hat{\boldsymbol{\xi}}_D)$ in $(\boldsymbol{\upsilon}_G, \boldsymbol{\upsilon}_D)$-coordinate,

$$\hat{\boldsymbol{\xi}}^T \hat{K}_{DD} \hat{\boldsymbol{\xi}} = \boldsymbol{\xi}^T K_{DD} \boldsymbol{\xi} \leq 0,$$

where $\boldsymbol{\xi} := (\hat{\boldsymbol{\xi}}_G, \mathbf{0}_{N-N_G}, \hat{\boldsymbol{\xi}}_D, \mathbf{0}_{M-M_D}) \in \mathbb{R}^{(N+M)}$.

Next, we demonstrate that $\hat{K}_{DG}$ is full rank. We observe that $\hat{\boldsymbol{\xi}}_D \neq 0$ if and only if $\boldsymbol{\xi} \notin \mathcal{T}_{\boldsymbol{\psi}^*}\mathcal{M}_D$. Then, according to Assumption III-2 (iv), we deduce that if $\hat{\boldsymbol{\xi}}_D \neq 0$

$$K_{DG}\boldsymbol{\xi} = -f'(0)\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{G_{\boldsymbol{\theta}^*}}(\mathbf{x})}\left[\nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) \cdot \boldsymbol{\xi}\right]\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} = -f'(0)\partial_{\boldsymbol{\xi}} h(\boldsymbol{\psi}^*) \neq 0.$$

The elements of $K_{DG}\boldsymbol{\xi}$ corresponding to the $\boldsymbol{\upsilon}_D$-coordinates are represented by $\hat{K}_{DG}\hat{\boldsymbol{\xi}}_D$, while those corresponding to the $\boldsymbol{\omega}_D$-coordinates are 0. Therefore, we conclude that $\hat{K}_{DG}\hat{\boldsymbol{\xi}}_D \neq 0$. Consequently, by the rank–nullity theorem, $\hat{K}_{DG}$ is full-rank.

Finally, by using similar arguments by selecting $(\boldsymbol{\upsilon}_G, \boldsymbol{\upsilon}_D)$-coordinate, without loss of generality, we only need to show K_{GG} is negative definite. By applying Assumption III-2 (i) and (v), the following lemma concludes that if $\eta > 0$ is sufficiently large, we can conclude the negative definiteness of $\nabla^2_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$ under Assumption III-2 (v-2).

Lemma B.10. *Let* $\mathbf{A}$ *be positive definite, and* $\mathbf{B}$ *be positive semi-definite. Then there is a* $\eta_{min} > 0$ *so that* $-\eta\mathbf{A} + \mathbf{B}$ *is negative definite for all* $\eta > \eta_{min}$.

The lemma holds because, for positive (semi-) definite matrix $\mathbf{X}$, we generally have

$$\lambda_{\max}(\mathbf{X}) \|\mathbf{w}\|^2 \geq \mathbf{w}^T\mathbf{X}\mathbf{w} \geq \lambda_{\min}(\mathbf{X}) \|\mathbf{w}\|^2,$$

for all $\mathbf{w}$. Here, $\lambda_{\max}(\mathbf{X})$ and $\lambda_{\min}(\mathbf{X})$ denote the maximum and minimum eigenvalues of $\mathbf{X}$, respectively. Thus if select $\eta > \frac{\lambda_{\max}(\mathbf{B})}{\lambda_{\min}(\mathbf{A})}$, then for any $\mathbf{w} \neq \mathbf{0}$, we have

$$\mathbf{w}^T(-\eta\mathbf{A} + \mathbf{B})\mathbf{w} = -\eta\mathbf{w}^T\mathbf{A}\mathbf{w} + \mathbf{w}^T\mathbf{B}\mathbf{w} \leq \big(-\eta\lambda_{\min}(\mathbf{A}) + \lambda_{\max}(\mathbf{B})\big)\|\mathbf{w}\|_2^2 < 0.$$

By applying Lemma B.6, we know that $\nabla_{(\boldsymbol{\upsilon}_G, \boldsymbol{\upsilon}_D)}\mathbf{v}(\boldsymbol{\theta}^*, \boldsymbol{\psi}^*)$ only has eigenvalues with negative real parts. Therefore, with a sufficiently small learning rate $h > 0$, Lemma B.7 guarantees the locally convergence of $\mathcal{F}_h$ on $\mathcal{M}_G \times \mathcal{M}_D$.

■

B.3.4 Literature on Stability Analysis of Adversarial Training

Studying the stability of GAN training from a dynamical systems perspective has been a popular approach [85, 22, 21, 86–89]. Generally, proving or disproving whether adversarial training is stable is challenging. However, [21] provides an example (Dirac-GAN) showing that, in general, GANs are not stable unless additional conditions are imposed.

As a result, researchers have explored additional conditions to stabilize GAN training. Essentially, the goal is to impose extra regularizations on the GAN loss $\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \mathbb{E}_{p_{\text{data}}(\mathbf{x})}\big[f(D_{\boldsymbol{\psi}}(\mathbf{x}))\big] + \mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\big[f(-D_{\boldsymbol{\psi}}(\mathbf{x}))\big]$, or its velocity field $\mathbf{v}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \begin{bmatrix} -\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) \\ \nabla_{\boldsymbol{\psi}}\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) \end{bmatrix}$ to ensure that the resulting Jacobian is Hurwitz. To elaborate further, we revisit the Jacobian $\mathcal{J}_{\text{GAN}}$ of the vanilla GAN, given by $\mathbf{v}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi})$:

$$\mathcal{J}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) := \begin{bmatrix} -\nabla^2_{\boldsymbol{\theta}}\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) & -\nabla^2_{\boldsymbol{\theta},\boldsymbol{\psi}}\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) \\ \nabla^2_{\boldsymbol{\theta},\boldsymbol{\psi}}\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) & \nabla^2_{\boldsymbol{\psi}}\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\psi}) \end{bmatrix} = \begin{bmatrix} K_{GG} & -K_{DG}^T \\ K_{DG} & K_{DD} \end{bmatrix}.$$

Here, we slightly abuse the notation from Section B.3.3 by using K_{ij}, $i, j \in \{D, G\}$, to denote the corresponding components in $\mathcal{J}_{\text{GAN}}$. By similar argument of Lemma B.8, we can obtain (indeed, $\eta = 0$ in Lemma B.8) that

$$\begin{aligned} K_{GG} &= f'(0)\mathbb{E}_{p_{\text{prior}}(\mathbf{z})}\left[\nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(\mathbf{z})^T \cdot \nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(G_{\boldsymbol{\theta}^*}(\mathbf{z})) \cdot \nabla_{\boldsymbol{\theta}} G_{\boldsymbol{\theta}^*}(\mathbf{z})\right]. \\ K_{DG} &= -f'(0)\nabla_{\boldsymbol{\theta}}\mathbb{E}_{p_{G_{\boldsymbol{\theta}}}(\mathbf{x})}\left[\nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})\right]\Big|_{\boldsymbol{\theta}=\boldsymbol{\theta}^*} \\ K_{DD} &= 2f''(0)\mathbb{E}_{p_{\text{data}}(\mathbf{x})}\left[\nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) \cdot \nabla_{\boldsymbol{\psi}} D_{\boldsymbol{\psi}^*}(\mathbf{x})^T\right]. \end{aligned}$$

Conceptually [83, 84], if we can ensure that the Jacobian at some equilibrium has only eigenvalues with strictly negative real parts, then the gradient descent iteration of $\mathcal{L}_{\text{GAN}}$ is asymptotically stable

Table 8: Comparison of various assumptions on stability analysis.

Method	K_{GG}	K_{DD}
[22]'s Vanilla GAN	Both p_{data} and $p_{\boldsymbol{\theta}}$ covers the whole space $\mathbb{R}^D$.	Additional technical assumptions (difficult to verify).
[21]'s Vanilla GAN	• $D_{\boldsymbol{\psi}^*}(\mathbf{x}) = \nabla_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ on $\text{supp}(p_{\text{data}})$. • $\nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x})$ positive definite. This implies K_{GG} is negative definite.	No further assumptions.
[21]'s Regularized GAN	• $D_{\boldsymbol{\psi}^*}(\mathbf{x}) = \nabla_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ on $\text{supp}(p_{\text{data}})$. • $\nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ on $\text{supp}(p_{\text{data}})$. This simply implies $K_{GG} = 0$.	By introducing a regularizer to modify the vector field $\mathbf{v}$ and obtaining a new vector field $\tilde{\mathbf{v}}$, they can determine an L_{DD} so that $\tilde{K}_{DD} := K_{DD} - L_{DD}$ is negative definite. Therefore, it is not vanilla GAN anymore.
PaGoDA	• $D_{\boldsymbol{\psi}^*}(\mathbf{x}) = \nabla_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ on $\text{supp}(p_{\text{data}})$. • $\nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x})$ just need to be positive semi-definite on $\text{supp}(p_{\text{data}})$. Then with $\eta > 0$ chosen to be sufficiently large in PaGoDA, K_{GG} is negative definite.	No further assumptions.

at that equilibrium. Therefore, the objective of many studies [85, 22, 21, 89] is to find conditions to verify Lemma B.6. We focus on discussing the conditions for K_{GG} and K_{DD} to be negative (semi-)definite, as this distinguishes PaGoDA's Theorem B.9 from the existing literature.

Under Assumption III-2 (i) that $f''(0) < 0$, it is worth noting that K_{DD} is generally negative semi-definite without additional conditions. Hence, studies [85, 22, 21] attempted to impose additional regularizers on $\mathcal{J}_{\text{GAN}}$ or $\mathbf{v}_{\text{GAN}}$ to ensure that either K_{DD} is negative definite (as in [22, 21]) or K_{GG} is negative definite (as in [21]). In Table 8, we provide a comparison of the various assumptions, at a high-level, drawn from the literature.

We emphasize that PaGoDA does not require $\nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x})$ to be strictly positive definite, thanks to PaGoDA's reconstruction loss. Specifically, it accommodates the scenario where $\nabla^2_{\mathbf{x}} D_{\boldsymbol{\psi}^*}(\mathbf{x}) = 0$ on $\text{supp}(p_{\text{data}})$. It's noteworthy that this capability enables PaGoDA to address cases where the instability of GAN is demonstrated, as exemplified by examples provided by [21].

C Limitations and Broader Impacts

Limitations. Algorithmically, the reconstruction loss is incompatible with the classifier-free guidance, which requires us to adopt the original distillation loss. However, as reconstruction loss directly uses the real data, it provides additional merit to decoder training, resulting in better performance as evidenced in the experiments. Theoretically, some theoretical assumptions of PaGoDA are challenging to verify in practice. For example, Theorems B.1 and B.3 require certain Lipschitz continuity of the score functions. This assumption is difficult to maintain at $t = 0$ due to the potential concentration of the data manifold in a lower-dimensional space, causing singularity. However, by truncating the PF-ODE solving at $t = \delta$ (for some $\delta > 0$), which is common in practice, this singularity is avoided, making the Lipschitz continuity assumption more feasible. In addition, Theorem B.4's assumption of the existence of a common minimizer can be difficult to verify empirically. However, with proper neural network parametrization and effective optimization, this assumption becomes more feasible. At last, verifying Assumptions III-1 and III-2 concerning the optimal properties of the generator and discriminator $(G_{\boldsymbol{\theta}}, D_{\boldsymbol{\psi}})$ is challenging in practice. These assumptions, essential for general (Lyapunov) stability analysis, are difficult to validate empirically. However, they appear reasonable based on our experimental observations. Last, empirically, PaGoDA's T2I generation capability relies heavily on the scale and quality of the training dataset.

Broader Impacts. PaGoDA, as a general media generative model, carries the risk of producing harmful or inappropriate content, such as deepfake images, graphic violence, or offensive material. To mitigate these risks, we avoid using the LAION dataset [33] in our model training, but robust content filtering and moderation mechanisms are essential to additionally prevent the generation of unethical or harmful media.