TOWARDS TRAINING ONE-STEP DIFFUSION MODELS WITHOUT DISTILLATION

Mingtian Zhang[1,*], **Jiajun He**[2,*], **Wenlin Chen**[2,4,*], **Zijing Ou**[3],
José Miguel Hernández-Lobato[2], **Bernhard Schölkopf**[4], **David Barber**[1]
[1]University College London, [2]University of Cambridge, [3]Imperial College London,
[4]MPI for Intelligent Systems, Tübingen,
m.zhang@cs.ucl.ac.uk jh2383@cam.ac.uk wc337@cam.ac.uk

ABSTRACT

Recent advances in one-step generative models typically follow a two-stage process: first training a teacher diffusion model and then distilling it into a one-step student model. This distillation process traditionally relies on both the teacher model's score function to compute the distillation loss and its weights for student initialization. In this paper, we explore whether one-step generative models can be trained directly without this distillation process. First, we show that the teacher's score function is not essential and propose a family of distillation methods that achieve competitive results without relying on score estimation. Next, we demonstrate that initialization from teacher weights is indispensable in successful training. Surprisingly, we find that this benefit is not due to improved "input-output" mapping but rather the learned feature representations, which dominate distillation quality. Our findings provide a better understanding of the role of initialization in one-step model training and its impact on distillation quality.

1 INTRODUCTION

Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020; Song & Ermon, 2019) have achieved remarkable success across various domains (Rombach et al., 2022; Li et al., 2022; Poole et al., 2022a; Ho et al., 2022; Hoogeboom et al., 2022; Liu et al., 2023), with several approaches enhancing generation speed (Jolicoeur-Martineau et al., 2021; Liu et al., 2022; Lu et al., 2022; Wang et al., 2021; De Bortoli et al., 2021; Xiao et al., 2021; Wang et al., 2022; Bao et al., 2022; Bekas et al., 2007; Ou et al., 2025). Recently, distillation techniques have gained popularity for one-step generation, achieving state-of-the-art results (Zhou et al., 2024b). These methods fall into two categories: trajectory-based distillation (Salimans & Ho, 2022; Berthelot et al., 2023; Song et al., 2023; Heek et al., 2024; Kim et al., 2023; Li & He, 2024), which integrates multi-step training with distillation, and score-based distillation (Luo et al., 2024; Salimans et al., 2024; Xie et al., 2024; Zhou et al., 2024b), which first pre-trains a diffusion teacher model and then distils it into a one-step model.

In this paper, we focus on the latter score-based strategy, as it provides a simpler training scheme. Specifically, we investigate whether a one-step model can be effectively trained without relying on a pre-trained first-stage teacher model. In the following sections, we first introduce the two-stage distillation method and then explore (1) whether a one-step model can be trained without using the teacher's scores and (2) whether it can be trained without initializing with the teacher's weights.

1.1 BACKGROUND OF SCORE-BASED DISTILLATION

Given data samples $\{x^{(1)}, \ldots, x^{(N)}\} \sim p_d(x_0)$, we define a one-step implicit model (Goodfellow et al., 2014; Huszár, 2017; Zhang et al., 2020) as $q_\theta(x_0) = \int \delta(x_0 - g_\theta(z))p(z)dz$ to match the data distribution $p_d(x_0)$. Inspired by diffusion models, one can use a set of (scaled) Gaussian convolution kernels $\mathcal{K} = \{k_1, \cdots, k_T\}$ where $k_t(x_t|x_0) = \mathcal{N}(x_t|\alpha_t x_0, \sigma_t^2 I)$ and define the Diffusive KL

[*]Equal contribution.

divergence between $q_\theta(x_0)$ and $p_d(x_0)$ as

$$\mathrm{DiKL}_{\mathcal{K}}(q_\theta(x_0)||p_d(x_0)) \equiv \sum_{t=1}^{T} w(t)\mathrm{KL}(q_\theta(x_t)||p_d(x_t)), \tag{1}$$

where $q_\theta(x_t) = \int q_\theta(x_0)k_t(x_t|x_0)dx_0$ and $p_d(x_t) = \int p_d(x_0)k_t(x_t|x_0)dx_0$. In addition to the diffusion distillation (Luo et al., 2024; Xie et al., 2024), this divergence has successfully been used in 3D generative models (Poole et al., 2022b; Wang et al., 2024) or training neural samplers He et al. (2024). For a single Gaussian kernel, the divergence was previously known as *Spread KL divergence* (Zhang et al., 2020; 2019). It is straightforward to show that it is a valid divergence, i.e., $\mathrm{DiKL}_{\mathcal{K}}(q_\theta||p_d) = 0 \Leftrightarrow q_\theta = p_d$, see Zhang et al. (2020) for a proof.

The gradient of θ is derived as follows, considering a single Gaussian kernel for simplicity:

$$\nabla_\theta \mathrm{KL}(q_\theta(x_t)||p_d(x_t)) = \int q_\theta(x_t)\left(\nabla_{x_t}\log q_\theta(x_t) - \nabla_{x_t}\log p_d(x_t)\right)\frac{\partial x_t}{\partial \theta}dx_t. \tag{2}$$

However, both $\nabla_{x_t}\log q_\theta(x_t)$ and $\nabla_{x_t}\log p_d(x_t)$ are not directly accessible. Fortunately, since we have access to samples of p_d and $\nabla_{x_t}\log p_d(x_t)$ remains fixed, we can approximate it once with *denoising score matching* (DSM) (Vincent, 2011) using a score network $s^{p_d}_{\psi_1}(x_t,t) \approx \nabla_{x_t}\log p_d(x_t)$:

$$\mathcal{L}_{\mathrm{DSM}}(\psi_1) = \iint \frac{1}{2}\|s^{p_d}_{\psi_1}(x_t,t) - \nabla_{x_t}\log k_t(x_t|x_0)\|_2^2 p_d(x_0)p(x_t|x_0)dx_t dx_0. \tag{3}$$

To approximate the score of the student model, we note that since we can efficiently sample from the student model, we can approximate $\nabla_{x_t}\log q_\theta(x_t)$ using a score network $s^{q_\theta}_{\psi_2}(x_t,t) \approx \nabla_{x_t}\log q_\theta(x_t)$, trained with the following DSM loss:

$$\mathcal{L}_{\mathrm{DSM}}(\psi_2) = \iint \frac{1}{2}\|s^{q_\theta}_{\psi_2}(x_t,t) - \nabla_{x_t}\log k_t(x_t|x_0)\|_2^2 q_\theta(x_0)p(x_t|x_0)dx_t dx_0. \tag{4}$$

Thus, the gradient with respect to θ is estimated as follows, a method known as Variational Score Distillation (VSD) (Poole et al., 2022a; Wang et al., 2024; Luo et al., 2024):

$$\nabla_\theta \mathrm{DiKL}(q_\theta(x_0)||p_d(x_0)) \approx \sum_{t=1}^{T} w(t)\int q_\theta(x_t)\big(s^{q_\theta}_{\psi_2}(x_t,t) - s^{p_d}_{\psi_1}(x_t,t)\big)\frac{\partial x_t}{\partial \theta}dx_t. \tag{5}$$

However, unlike $\nabla_{x_t}\log p_d(x_t)$, which remains fixed, $\nabla_{x_t}\log q_\theta(x_t)$ dynamically changes during training. Therefore, we need to update the score network $s^{q_\theta}_{\psi_2}(x_t,t) \approx \nabla_{x_t}\log q_\theta(x_t)$ at each gradient step when optimizing θ. The full training procedure is detailed in Algorithm 1.

Algorithm 1 Score-based Distillation of One-Step Generative Models

Require: Data samples $\{x^{(1)},\ldots,x^{(N)}\} \sim p_d(x_0)$

Stage 1: Train a multi-step teacher diffusion model

1: Train teacher score network $s^{p_d}_{\psi_1}(x_t,t)$ using DSM until convergence

Stage 2: Train a one-step student generative model

2: Initialize the student network with the teacher's score network $g_{\theta_{\text{init}}}(\cdot) \equiv s^{p_d}_{\psi_1}(\cdot, t = t_{\text{init}})$

3: **for** each training iteration **do**

4: Train student score network $s^{q_\theta}_{\psi_2}(x_t,t)$ using DSM

5: Estimate the DiKL gradient with score network $s^{q_\theta}_{\psi_2}(x_t,t)$ and $s^{p_d}_{\psi_1}(x_t,t)$

6: Update one-step generator's parameters θ with the estimated DiKL gradient

7: **end for**

2 Training One-Step Model without Teacher's Score

In Algorithm 1, the DiKL gradient estimation relies on the difference score difference, $s^{q_\theta}_{\psi_1}(x_t,t) - s^{p_d}_{\psi_2}(x_t,t)$. To eliminate the dependency on the teacher's score network, we observe that the score difference can be computed via the gradient of this ratio: $\nabla_{x_t}\log q_\theta(x_t) - \nabla_{x_t}\log p_d(x_t) =$

$\nabla_{x_t} \log(q_\theta(x_t)/p_d(x_t))$ Rather than estimating the two scores separately, we can directly estimate the density ratio between the student and teacher models using class-ratio estimation (Sugiyama et al., 2012; Qin, 1998; Gutmann & Hyvärinen, 2010). Specifically, we first denote distributions $q_\theta(x_t)$ and $p_d(x_t)$ as two conditional distributions $m(x_t|y=0)$ and $m(x_t|y=1)$, respectively. With Bayes' rule, we can transform the ratio estimation as a binary classification problem:

$$\frac{q_\theta(x_t)}{p_d(x_t)} \equiv \frac{m(x_t|y=0)}{m(x_t|y=1)} = \frac{p(y=0|x_t)\cancel{m(x_t)}}{\cancel{p(y=0)}} \bigg/ \frac{p(y=1|x_t)\cancel{m(x_t)}}{\cancel{p(y=1)}} = \frac{p(y=0|x_t)}{p(y=1|x_t)}. \tag{6}$$

where the mixture distribution $m(x) \equiv m(x_t|y=1)p(y=1) + m(x_t|y=0)p(y=0)$ and the Bernoulli prior distribution $p(y)$ can be simply set as a uniform prior $p(y=1) = p(y=0) = 0.5$. In practice, we sample a batch of data from the p_d and q_θ and with the labels $y=0$ and $y=1$, we train a neural network $c_\eta(x_t, t)$ classifier that conditional on t to learn the probability of $y=1$ given x_t, $c^*(x_t, t) = p(y=1|x_t, t)$. The log-ratio can be estimated by

$$\nabla_{x_t} \log(q_\theta(x_t)/p_d(x_t)) \approx \nabla_{x_t} \log(1 - c_\eta(x_t, t))/c_\eta(x_t, t) = \nabla_{x_t} \text{logit}(1 - c_\eta(x_t, t)). \tag{7}$$

We can then plug in this estimator to Equation 2 to form the DiKL gradient estimation:

$$\nabla_\theta \text{DiKL}(q_\theta(x_0)||p_d(x_0)) \approx \sum_{t=1}^{T} w(t) \int q_\theta(x_t) \nabla_{x_t} \text{logit}(1 - c_\eta(x_t, t)) \frac{\partial x_t}{\partial \theta} dx_t, \tag{8}$$

In addition to the DiKL, we can use the learned classifier function c_η to define alternative learning objectives. For instance, replacing the logit function with the logarithm yields an objective that minimizes the probability of generated samples being classified as fake. This formulation aligns with the GAN (Goodfellow et al., 2014; Nowozin et al., 2016) across different diffusion timesteps, which is equivalent to minimizing the diffusive JS divergence:

$$\nabla_\theta \text{DiJS}(q_\theta(x_0)||p_d(x_0)) \approx \sum_{t=1}^{T} w(t) \int q_\theta(x_t) \nabla_{x_t} \log(1 - c_\eta(x_t, t)) \frac{\partial x_t}{\partial \theta} dx_t. \tag{9}$$

This objective was first used in DiffusionGAN (Wang et al., 2022) and has also shown promise in one-step video generation (Lin et al., 2025) from a recent concurrent work. However, unlike DiffusionGAN, which heavily depends on the StyleGAN2 architecture (Karras et al., 2020) with gradient penalty (Arjovsky et al., 2017), our method is compatible with a UNet (Ronneberger et al., 2015) generator without requiring additional GAN techniques, while still maintaining stable training.

Alternatively, rather than minimizing the probability that generated images are classified as fake as used in GAN, we can maximize the probability that they are classified as real. We refer to this approach as *Diffusive Realness Maximization (DiRM)*, and define the loss gradient as

$$\nabla_\theta \text{DiRM}(\theta) \approx -\sum_{t=1}^{T} w(t) \int q_\theta(x_t) \nabla_{x_t} \log(c_\eta(x_t, t)) \frac{\partial x_t}{\partial \theta} dx_t. \tag{10}$$

We implement the proposed methods using the EDM (Karras et al., 2022) codebase (see Algorithm 2 for training details). Our discriminator employs the encoder part of the U-Net, outputting a logit scalar at half the size of the score network, which utilizes a full UNet. The generator is initialized with EDM pre-training, and experiments are conducted on unconditional CIFAR-10 (Krizhevsky et al., 2009). Additional details are provided in Appendix B. As shown in Table 1, DiJS, without teacher score estimation, outperforms DiKL and DiRM and remains competitive with state-of-the-art one-step generation methods.

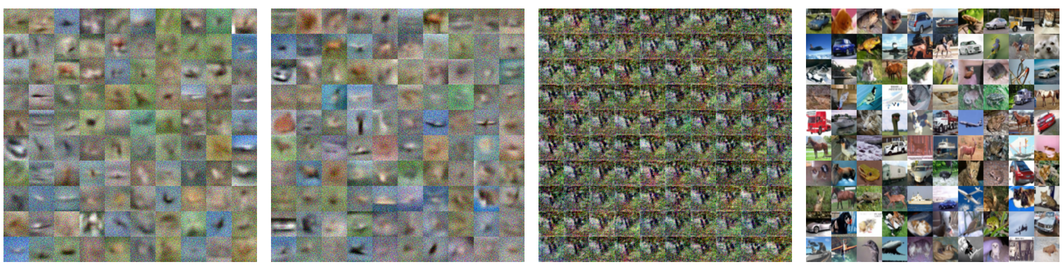
Table 1: Sample quality on CIFAR-10.

METHOD	NFE (↓)	FID (↓)	IS (↑)
Accelerated Diffusion models			
EDM (Karras et al., 2022)	35	2.04	9.84
DDIM (Song et al., 2020)	10	8.23	-
DPM-solver-fast (Lu et al., 2022)	10	4.70	-
AMED-plugin (Zhou et al., 2024c)	5	6.61	-
iCT (Song & Dhariwal, 2024)	1	2.83	9.54
CTM (Kim et al., 2023)	1	1.98	-
BCM (Li & He, 2024)	1	3.10	9.45
sCT (Lu & Song, 2025)	1	2.97	-
Score-based Distillation			
Diff-Instruct (Luo et al., 2024)	1	4.53	-
SID ($\alpha = 1$) (Zhou et al., 2024b)	1	2.03	10.02
SIDA ($\alpha = 1$) (Zhou et al., 2024a)	1	1.52	10.32
SID^{2}A ($\alpha = 1$) (Zhou et al., 2024a)	1	1.40	10.19
Diff-GAN (Wang et al., 2022)	1	3.19	-
Score-free / Class-ratio-based Distillation (Ours)			
DiRM	1	4.87	9.85
DiKL	1	3.81	9.90
DiJS	1	2.39	9.93

3 TRAINING ONE-STEP MODEL WITHOUT TEACHER'S WEIGHTS

In previous results, student models were initialized from the teacher's weights. Training from random initialization led to mode collapse, see Figure 1c for an example of mode collapse. One possible explanation is that mode collapse arises from the training objectives (RKL or JS divergence), a phenomenon also observed in GAN literature Goodfellow et al. (2014). To understand why the teacher's weights help prevent mode collapse in student model training, we investigate two hypotheses:

Function Space Hypothesis: *Weight initialization provides a more structured latent-to-output functional mapping—i.e., different locations in the latent space are initially mapped to distinct images, preventing mode collapse.* This hypothesis arises from visualizing initialized samples (see Figure 1a), which show that initialization already induces diverse mappings, with the second stage primarily refining these into sharper images. Although intuitive, our findings surprisingly show that functional initialization alone is insufficient to prevent mode collapse. To show this, instead of training the teacher model across different timesteps t and selecting the t_{init} for initialization, we only pre-train the teacher model at the target timestep t_{init} and use its weight to initialize the one-step model. This setup ensures identical latent-to-output mappings for the student model at initialization, see Figure 1b. However, with this initialization, the student model still exhibits mode collapse early in second-stage training, which suggests that the functional mapping perspective alone does not fully explain one-step model training.



(a) Single-level DSM Init. (b) Multi-level DSM Init. (c) Collapsed Samples (d) DiJS Samples

Figure 1: Sample visualizations of different methods, see Appendix B for full images visualizations.

Table 2: FID scores for different initialization methods on various datasets.

Initialization	Initialization Dataset	FID
No initialization	-	collapsed
Single-level DSM	full CIFAR-10	collapsed
Multi-level DSM	10 classes in CIFAR-100	collapsed
	50 classes in CIFAR-100	6.20
	90 classes in CIFAR-100	6.01
	full CIFAR-10	2.39

Feature Space Hypothesis: *Weight initialization provides a rich set of multi-level features learned in training the diffusion, which help prevent mode collapse.* To verify this hypothesis and isolate the role of learned features from functional mapping effects, we pre-trained the teacher model on CIFAR-100 while excluding any classes that overlap with CIFAR-10. This ensures that the second-stage generation targets are absent during pre-training, allowing us to focus solely on the contribution of learned features. We then trained the teacher model using progressively larger subsets of CIFAR-100 with (10, 50, 90) classes, creating a setting with increasing feature diversity. Table 2 shows the FID scores of one-step model on CIFAR-10 with different numbers of CIFAR-100 classes used for initialization. We find that when the teacher model is trained on only 10 classes, mode collapse still occurs. However, as the number of training classes increases, the model no longer collapses, indicating that feature richness plays a crucial role in preventing mode collapse. Nevertheless, despite mitigating mode collapse, this initialization strategy achieves an FID of 6.01, which is significantly worse than the 2.39 FID obtained when using CIFAR-10 as the pre-training dataset. This suggests that while feature richness is essential for stabilizing training, functional mapping initialization remains important for achieving higher sample quality.

4 Conclusion and Discussion

In this paper, we investigate training a one-step diffusion model without a pre-trained teacher and propose score-estimation-free methods for training one-step generative models. Additionally, our study identifies key pre-training components, highlighting the role of feature richness in preventing mode collapse and the necessity of functional mapping for high-quality samples. Future work could explore unsupervised or self-supervised pre-training, in addition to diffusion pre-training, to enhance feature diversity and improve one-step models across modalities like images, audio, or videos.

Acknowledgments

MZ and DB acknowledge funding from AI Hub in Generative Models, under grant EP/Y028805/1. JH is supported by the University of Cambridge Harding Distinguished Postgraduate Scholars Programme. ZO is supported by the Lee Family Scholarship. JMHL acknowledges support from a Turing AI Fellowship under grant EP/V023756/1.

References

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pp. 214–223. PMLR, 2017.

Fan Bao, Chongxuan Li, Jun Zhu, and Bo Zhang. Analytic-dpm: an analytic estimate of the optimal reverse variance in diffusion probabilistic models. *arXiv preprint arXiv:2201.06503*, 2022.

Costas Bekas, Effrosyni Kokiopoulou, and Yousef Saad. An estimator for the diagonal of a matrix. *Applied numerical mathematics*, 57(11-12):1214–1229, 2007.

David Berthelot, Arnaud Autef, Jierui Lin, Dian Ang Yap, Shuangfei Zhai, Siyuan Hu, Daniel Zheng, Walter Talbott, and Eric Gu. Tract: Denoising diffusion models with transitive closure time-distillation. *arXiv preprint arXiv:2303.04248*, 2023.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. *Advances in Neural Information Processing Systems*, 34:17695–17709, 2021.

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.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 297–304, 2010.

Jiajun He, Wenlin Chen, Mingtian Zhang, David Barber, and José Miguel Hernández-Lobato. Training neural samplers with reverse diffusive kl divergence. *arXiv preprint arXiv:2410.12456*, 2024.

Jonathan Heek, Emiel Hoogeboom, and Tim Salimans. Multistep consistency models. *arXiv preprint arXiv:2403.06807*, 2024.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022.

Emiel Hoogeboom, Vıctor Garcia Satorras, Clément Vignac, and Max Welling. Equivariant diffusion for molecule generation in 3d. In *International conference on machine learning*, pp. 8867–8887. PMLR, 2022.

Ferenc Huszár. Variational inference using implicit distributions. *arXiv preprint arXiv:1702.08235*, 2017.

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

Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020.

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.

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. *arXiv preprint arXiv:2310.02279*, 2023.

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

Liangchen Li and Jiajun He. Bidirectional consistency models. *arXiv preprint arXiv:2403.18035*, 2024.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343, 2022.

Shanchuan Lin, Xin Xia, Yuxi Ren, Ceyuan Yang, Xuefeng Xiao, and Lu Jiang. Diffusion adversarial post-training for one-step video generation. *arXiv preprint arXiv:2501.08316*, 2025.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D Plumbley. Audioldm: Text-to-audio generation with latent diffusion models. *arXiv preprint arXiv:2301.12503*, 2023.

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

Cheng Lu and Yang Song. Simplifying, stabilizing and scaling continuous-time consistency models. In *International Conference on Learning Representations*, 2025.

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.

Weijian Luo, Tianyang Hu, Shifeng Zhang, Jiacheng Sun, Zhenguo Li, and Zhihua Zhang. Diff-instruct: A universal approach for transferring knowledge from pre-trained diffusion models. *Advances in Neural Information Processing Systems*, 36, 2024.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. *Advances in neural information processing systems*, 29, 2016.

Zijing Ou, Mingtian Zhang, Andi Zhang, Tim Z Xiao, Yingzhen Li, and David Barber. Improving probabilistic diffusion models with optimal covariance matching. *International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=fV0t65OBUu`.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022a.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *The Eleventh International Conference on Learning Representations*, 2022b.

Jing Qin. Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, 85(3):619–630, 1998.

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*, pp. 10684–10695, 2022.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Tim Salimans, Thomas Mensink, Jonathan Heek, and Emiel Hoogeboom. Multistep distillation of diffusion models via moment matching. *arXiv preprint arXiv:2406.04103*, 2024.

Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pp. 2256–2265. PMLR, 2015.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

Yang Song and Prafulla Dhariwal. Improved techniques for training consistency models. In *The Twelfth International Conference on Learning Representations*, 2024.

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

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

Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori. *Density ratio estimation in machine learning*. Cambridge University Press, 2012.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011.

Gefei Wang, Yuling Jiao, Qian Xu, Yang Wang, and Can Yang. Deep generative learning via schrödinger bridge. In *International conference on machine learning*, pp. 10794–10804. PMLR, 2021.

Zhendong Wang, Huangjie Zheng, Pengcheng He, Weizhu Chen, and Mingyuan Zhou. Diffusion-gan: Training gans with diffusion. *arXiv preprint arXiv:2206.02262*, 2022.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems*, 36, 2024.

Zhisheng Xiao, Karsten Kreis, and Arash Vahdat. Tackling the generative learning trilemma with denoising diffusion gans. *arXiv preprint arXiv:2112.07804*, 2021.

Sirui Xie, Zhisheng Xiao, Diederik P Kingma, Tingbo Hou, Ying Nian Wu, Kevin Patrick Murphy, Tim Salimans, Ben Poole, and Ruiqi Gao. Em distillation for one-step diffusion models. *arXiv preprint arXiv:2405.16852*, 2024.

Mingtian Zhang, Thomas Bird, Raza Habib, Tianlin Xu, and David Barber. Variational f-divergence minimization. *arXiv preprint arXiv:1907.11891*, 2019.

Mingtian Zhang, Peter Hayes, Thomas Bird, Raza Habib, and David Barber. Spread divergence. In *International Conference on Machine Learning*, pp. 11106–11116. PMLR, 2020.

Mingyuan Zhou, Huangjie Zheng, Yi Gu, Zhendong Wang, and Hai Huang. Adversarial score identity distillation: Rapidly surpassing the teacher in one step. *arXiv preprint arXiv:2410.14919*, 2024a.

Mingyuan Zhou, Huangjie Zheng, Zhendong Wang, Mingzhang Yin, and Hai Huang. Score identity distillation: Exponentially fast distillation of pretrained diffusion models for one-step generation. In *Forty-first International Conference on Machine Learning*, 2024b.

Zhenyu Zhou, Defang Chen, Can Wang, and Chun Chen. Fast ode-based sampling for diffusion models in around 5 steps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7777–7786, 2024c.

A ALGORITHM

Algorithm 2 Score-free Training of One-Step Generative Models

Require: Data samples $\{x^{(1)}, \ldots, x^{(N)}\} \sim p_d(x_0)$

——— Stage 1: Train a multi-step teacher diffusion model ———

1: Train teacher score network $s_{\psi_2}^{p_d}(x_t, t)$ using Eq. 3 until convergence

——— Stage 2: Train a one-step student generative model ———

2: Initialize the student network with the teacher's score network $g_{\theta_{\text{init}}}(\cdot) \equiv s_{\psi_1}^{p_d}(\cdot, t = t_{\text{init}})$
3: **for** each training iteration **do**
4: Estimate the ratio r_η using Eq. 8 or Eq. 9 or Eq. 10
5: Estimate the DiKL gradient (Eq. 5) with the ratio network r_η
6: Update one-step generator's parameters θ with the estimated DiKL gradient
7: **end for**

B EXPERIMENTAL SETUP AND ADDITIONAL RESULTS

We conduct all our experiments on a single Nvidia H100-80GB GPU. The generator is initialized using the EDM pre-trained model from `https://nvlabs-fi-cdn.nvidia.com/edm/pretrained/edm-cifar10-32x32-uncond-vp.pkl`. We adopt the variance-exploding (VE) parameterization, consistent with EDM Karras et al. (2022) for the corresponding settings. Additionally, we apply non-leaky data augmentation Karras et al. (2020).

Our training setup includes a batch size of 64, an exponential moving average (EMA) decay of 0.5, a learning rate of 0.00001, and a fixed timestep $t_{\text{fix}} = 2.5$ with weight function $w(t) = \sigma_t^2$.

For each generator update, we take one gradient step for ratio estimation to ensure efficient training. We observed that multiple-step updates can accelerate generator convergence without introducing instability—unlike GANs, where multiple ratio updates often cause training instability. However, multiple ratio steps significantly slow down the overall training process. Therefore, we use a single-step gradient update in all our experiments, which is consistent with the settings in Luo et al. (2024); Zhou et al. (2024b) and leave the exploration of multi-step ratio estimation for future work.

Figure 2: Visualization of the samples from the multi-level DSM Initialization

Figure 3: Visualization of the samples from the single-level DSM Initialization

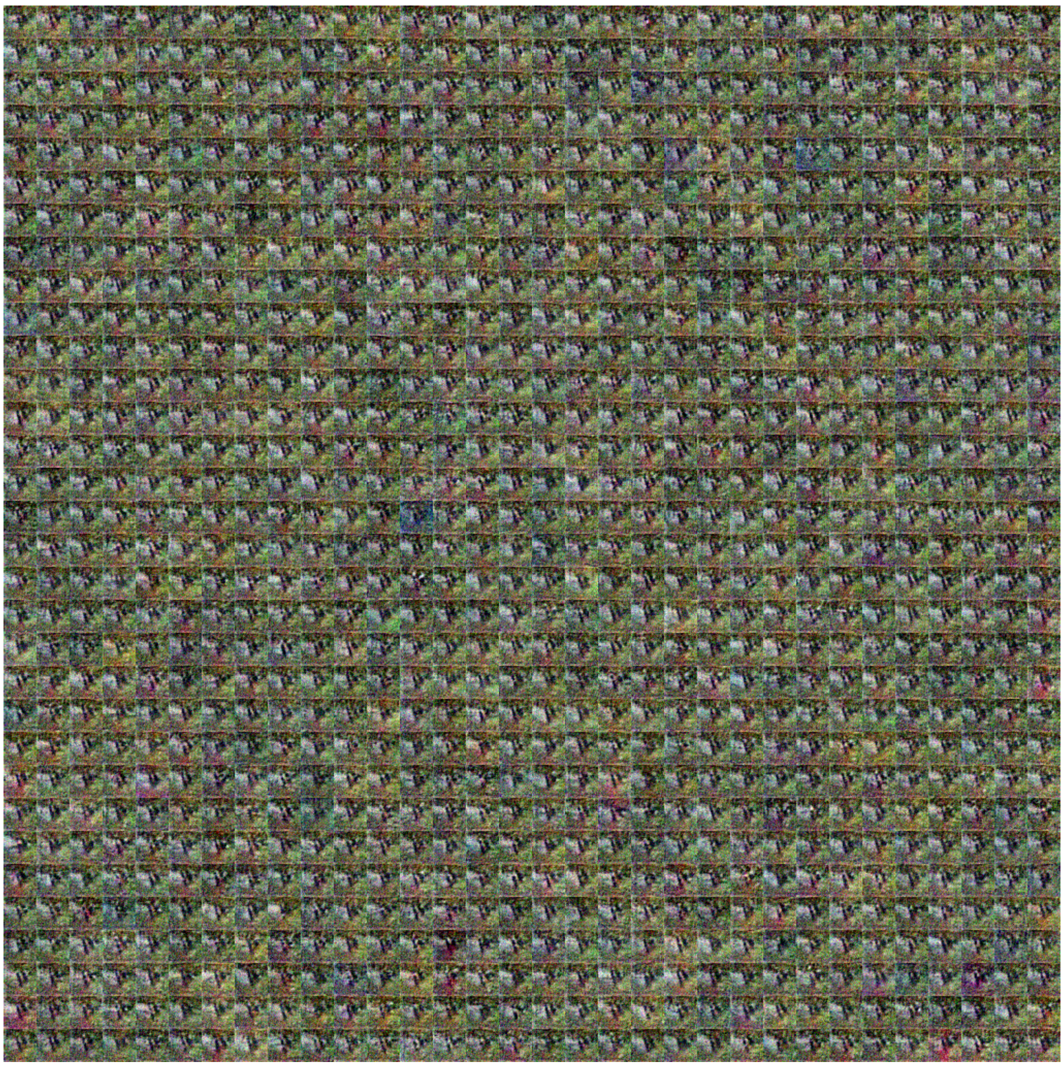


Figure 4: Visualization of the collapsed samples

Figure 5: Visualization of the DiJS samples (FID=2.39, IS=9.93)