ScaleDreamer: Scalable Text-to-3D Synthesis with Asynchronous Score Distillation

Zhiyuan Ma[1,2], Yuxiang Wei[1,5], Yabin Zhang[1],
Xiangyu Zhu[3,4], Zhen Lei[1,2,3,4*], and Lei Zhang[1*]

[1] The Hong Kong Polytechnic University, PolyU
[2] Center for Artificial Intelligence and Robotics, HKISI CAS
[3] State Key Laboratory of Multimodal Artificial Intelligence Systems, CASIA
[4] School of Artificial Intelligence, University of Chinese Academy of Sciences, UCAS
[5] Harbin Institute of Technology, HIT

Abstract. By leveraging the text-to-image diffusion prior, score distillation can synthesize 3D contents without paired text-3D training data. Instead of spending hours of online optimization per text prompt, recent studies have been focused on learning a text-to-3D generative network for amortizing multiple text-3D relations, which can synthesize 3D contents in seconds. However, existing score distillation methods are hard to scale up to a large amount of text prompts due to the difficulties in aligning pretrained diffusion prior with the distribution of rendered images from various text prompts. Current state-of-the-arts such as Variational Score Distillation finetune the pretrained diffusion model to minimize the noise prediction error so as to align the distributions, which are however unstable to train and will impair the model's comprehension capability to numerous text prompts. Based on the observation that the diffusion models tend to have lower noise prediction errors at earlier timesteps, we propose Asynchronous Score Distillation (ASD), which minimizes the noise prediction error by shifting the diffusion timestep to earlier ones. ASD is stable to train and can scale up to 100k prompts. It reduces the noise prediction error without changing the weights of pre-trained diffusion model, thus keeping its strong comprehension capability to prompts. We conduct extensive experiments using different text-to-3D architectures, including Hyper-iNGP and 3DConvNet. The results demonstrate ASD's effectiveness in stable 3D generator training, high-quality 3D content synthesis, and its superior prompt-consistency, especially under large prompt corpus. Code is available at `https://github.com/theEricMa/ScaleDreamer`.

Keywords: Text-to-3D · Score Distillation · Diffusion Model

1 Introduction

Text-to-3D aims to generate realistic 3D contents from the given textual descriptions [40], which is particularly useful in many applications such as virtual

* Corresponding authors.

reality [61] and game design [23]. The main challenge of this task, however, lies in how to generate high-quality 3D contents conditioned on the abstract and diverse textual descriptions. Many existing text-to-3D methods [11, 12, 26–29, 32, 35, 37, 40, 42, 58, 59, 75] are optimization-based ones, which distill the guidance from the powerful pretrained text-to-image diffusion models [5, 11, 26, 32, 42, 44, 73] via score distillation [40, 59, 62, 72]. In general, these methods employ the KL divergence to reduce the discrepancy between the distribution of rendered images and the desired image distribution embedded in the 2D diffusion prior, while they differ in how to use the pretrained diffusion prior to model the distribution of rendered images. Extensive efforts have been made to explore prompt-specific optimization of various 3D representations, including implicit radiance fields [40], explicit radiance fields [29, 37, 59], DmTets [56, 74] and 3D Gaussians [9]. Typically, tens of minutes to hours are needed to optimize a single 3D representation for one prompt to achieve the desired result.

Compared to the aforementioned optimization-based text-to-3D methods, learning-based methods [6, 21, 31, 36, 43, 54, 65] can largely reduce the computational cost by training a text-conditioned 3D generative network. With the availability of 3D object collections [10, 63, 71], a deep network can be trained in a supervised manner so that 3D outputs can be generated in several seconds. Unfortunately, the size of existing text-3D datasets is far from sufficient compared to text-image datasets [45], limiting the text-to-3D generation performance of trained models. Inspired by the optimization-based text-to-3D methods that use pretrained 2D diffusion models, efforts have been made to train text-to-3D networks by using 2D diffusion models as supervisors [33, 41, 65] without using text-3D pairs. For example, a text-conditioned 3D hyper-network is trained in ATT3D [33] via Score Distillation Sampling (SDS) [40]. Nevertheless, this method suffers from numerical instability, which has been observed in subsequent studies [41, 65] that apply SDS to different 3D generator networks.

Despite the success of score distillation in optimization-based text-to-3D generation [40, 59, 72], its application to learning-based text-to-3D frameworks is rather limited because of the unstable training or unsatisfactory results. We argue that the primary challenge lies in how to efficiently and effectively leverage the pretrained 2D diffusion prior to represent the distribution of images rendered by the 3D generator. For example, SDS [40] forces the rendered images to adhere to the Dirac distribution, which causes numerical instability in 3D generator training [33, 65]. Variational Score Distillation (VSD) [59] finetunes the 2D diffusion prior for distribution alignment via minimizing the noise prediction error. However, the finetuning changes the pretrained diffusion network and hurts its comprehension capability to numerous text prompts, leading to mode collapse when the size of text prompts is extended.

To address the above mentioned issues, we propose Asynchronous Score Distillation (ASD). Like VSD, ASD aims to minimize the noise prediction error. Different from VSD, ASD does not finetune the pretrained 2D diffusion network; instead, it achieves the goal by shifting the diffusion timestep. This is based on the observation that diffusion networks will have smaller noise prediction errors

in earlier timesteps [67]; therefore, we can shift the timestep to an earlier step to achieve a similar goal to VSD, *i.e.*, reducing the noise prediction error. In this way, the diffusion network can be frozen in training and its strong text comprehension capability can be well-preserved. The shifted timesteps can be well sampled from a pre-defined range for most prompts. To evaluate the performance of ASD, we conduct extensive experiments by using two types of generator architectures, *i.e.* Hyper-iNGP [33] and 3DConv-Net [4], across various prompt corpus sizes. We conduct extensive experiments to evaluate the superiority of ASD to previous methods, including the stable training of 3D generators, the production of high-quality 3D outputs, the high content fidelity to input prompts, as well as its scalability to larger corpus sizes, *e.g.*, **100k** prompts. We also show that ASD can work with other 2D diffusion models such as MVDream [48], and can be used to train more 3D generators such as Triplane-Transformer [17].

2 Literature Review

2.1 Text-to-3D with Score Distillation

Text-to-3D takes text description, a.k.a. text prompt y, as input, and outputs 3D representation θ that renders high-fidelity images at any camera view π. Thanks to the powerful text-to-image diffusion models [32, 42, 44, 48, 73], we can optimize θ to align with y by computing the objective $\mathcal{L}(\boldsymbol{x}, y)$ on the rendered image $\boldsymbol{x} = g(\theta, \pi)$ from camera view π. Through differential rendering, θ can be updated with the gradient $\nabla_\theta \mathcal{L}(\theta, y) = \frac{\partial \mathcal{L}(\boldsymbol{x}, y)}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial \theta}$. This technique is generally termed as score distillation. Unlike data-driven techniques [6, 21, 31, 43, 54], score distillation approaches [8, 19, 24, 29, 40, 53, 59, 72] can produce high-quality 3D content without the need for 3D training datasets.

Prompt-Specific Text-to-3D. Existing score distillation methods [40, 59, 72] were originally developed to output a single 3D result θ for a single text prompt y via online optimization: $\min_\theta \mathbb{E}_{\pi, \boldsymbol{x}=g(\theta,\pi)}[\mathcal{L}(\boldsymbol{x}, y)]$. The utilized 3D representations, *e.g.*, NeRF [39, 40], DmTet [47, 74], and 3D Gaussian [20, 30, 51, 53, 57, 70], are not designed to render scenes from varying text prompts. Therefore, the optimization has to be conducted again for newly provided text prompts. The optimization process typically costs tens of minutes to hours.

Prompt-Amortized Text-to-3D. To mitigate the computational costs in prompt-specific methods, recent studies [25, 33, 41, 65] have attempted to use score distillation to train a text-to-3D generator $\theta = \mathcal{G}(y)$, aiming to generate multiple 3D representations from a set of text prompts $S_y = \{y\}$. These methods can generate 3D results from queried text prompt in seconds. As proposed by ATT3D [33], the 3D generator training is performed by minimizing $\min_{\mathcal{G}} \mathbb{E}_{\pi, y \in S_y, \boldsymbol{x}=g(\mathcal{G}(y),\pi)}[\mathcal{L}(\boldsymbol{x}, y)]$ over all text prompts. Unlike data-driven approaches [17, 52, 66], score distillation bypasses the scarcity of text-3D data pairs because the 2D diffusion prior can offer the guidance to align the 3D output with the input text prompt. However, its application is currently restricted to training the 3D generator within a limited range of text prompts.

2.2 Representative Score Distillation Methods

Denote by ϕ the 2D diffusion prior [44,48] and by $p^{\phi}(\boldsymbol{x} \mid y)$ the text-conditioned image distribution embedded within ϕ, the objectives of most existing score distillation methods can be generally concluded as minimizing the objective $\mathcal{L}(\theta, y) = \mathbb{E}_{\pi,t,\boldsymbol{\epsilon},\boldsymbol{x}=g(\theta,\pi)}\left[\omega(t) D_{\mathrm{KL}}\left(q_t^{\theta}(\boldsymbol{x}_t \mid \pi) \| p_t^{\phi}(\boldsymbol{x}_t \mid y^{\pi})\right)\right]$, where D_{KL} denotes KL divergence, $q_t^{\theta}(\boldsymbol{x}_t \mid \pi)$ denotes the distribution of images $\boldsymbol{x}$ rendered at camera view π at diffusion timestep t [15], and the same for $p_t^{\phi}(\boldsymbol{x}_t \mid y)$. $\omega(t)$ is a timestep-dependent weight [40]. y^{π} denotes the view-dependent strategy [44] or view-awareness [42,48] to prompt the different camera views [40]. To minimize this objective, the gradient w.r.t. θ can be calculated as per [59]:

$$\nabla_{\theta}\mathcal{L}(\theta, y) = \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\underbrace{-\sigma_t \nabla_{\boldsymbol{x}_t} \log p_t^{\phi}(\boldsymbol{x}_t \mid y^{\pi})}_{\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t, y^{\pi})} - \underbrace{\left(-\sigma_t \nabla_{\boldsymbol{x}_t} \log q_t^{\theta}(\boldsymbol{x}_t \mid \pi)\right)}_{\boldsymbol{\epsilon}_{\theta}(\boldsymbol{x}_t; t, \pi, y)}\right)\frac{\partial \boldsymbol{x}}{\partial \theta}\right], \quad (1)$$

where the first term $-\sigma_t \nabla_{\boldsymbol{x}_t} \log p_t^{\phi}(\boldsymbol{x}_t \mid y^{\pi})$ corresponds to the score function [50] of the desired image distribution, and it can be achieved by predicting the noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$ in the noisy image $\boldsymbol{x}_t = \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\epsilon}$ using the pretrained 2D diffusion model $\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t, y^{\pi})$ [44,48]. Existing score distillation methods [40,59,72] mainly differ in how to model $-\sigma_t \nabla \boldsymbol{x}_t \log q_t^{\theta}(\boldsymbol{x}_t \mid \pi)$, which corresponds to the score function of the distribution of rendered images $q^{\theta}(\boldsymbol{x} \mid \pi)$. We denote this term in Eq. 1 as $\boldsymbol{\epsilon}_{\theta}(\boldsymbol{x}_t; t, \pi, y)$ in the following context, since it represents a diffusion model that corresponds to θ. A summary of the objectives of major score distillation methods is shown in Tab. 1.

The objective of **Score Distillation Sampling** (SDS) [40] is $\nabla_{\theta}\mathcal{L}_{\mathrm{SDS}}(\theta, y) \triangleq \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t, y^{\pi}) - \boldsymbol{\epsilon}\right)\frac{\partial \boldsymbol{x}}{\partial \theta}\right]$, which approximates the term $\boldsymbol{\epsilon}_{\theta}(\boldsymbol{x}_t; t, \pi, y)$ in Eq. 1 as the ground-truth noise $\boldsymbol{\epsilon}$. That is, SDS assumes that $q^{\theta}(\boldsymbol{x} \mid \pi)$ adheres to a Dirac distribution $\delta(\boldsymbol{x} - g(\theta, \pi))$ [59], which is characterized by a non-zero density at the singular point of $\boldsymbol{x} = g(\theta, \pi)$ and zero density everywhere else. However, updating θ under the Dirac distribution might be troublesome [59]. We may need to set the CFG (Classifier Free Guidance) [16] as high as 100 for model convergence, which will produce excessively large gradients and lead to unstable optimization. This problem is alleviated by **Classifier Score Distillation** (CSD) [72], which uses the classifier component [16] in SDS as the objective: $\nabla_{\theta}\mathcal{L}_{\mathrm{CSD}}(\theta, y) \triangleq \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t, y^{\pi}) - \boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t)\right)\frac{\partial \boldsymbol{x}}{\partial \theta}\right]$. CSD can be regraded as straightforwardly using the unconditional term of the diffusion prior $\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t)$ to represent $\boldsymbol{\epsilon}_{\theta}(\boldsymbol{x}_t; t, \pi, y)$ in Eq. 1. Unfortunately, in the case of prompt-amortized training, this term may not provide effective gradient, because $\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t)$ is unconditional to the provided text-prompts. In contrast, **Variational Score Distillation** (VSD) [59] models $\boldsymbol{\epsilon}_{\theta}(\boldsymbol{x}_t; t, \pi, y)$ with another text-aware diffusion model $\boldsymbol{\epsilon}_{\phi'}(\mathbf{x}_t; t, \pi, y)$, leading to $\nabla_{\theta}\mathcal{L}_{\mathrm{VSD}}(\theta, y) \triangleq \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t, y^{\pi}) - \boldsymbol{\epsilon}_{\phi'}(\mathbf{x}_t; t, \pi, y)\right)\frac{\partial \boldsymbol{x}}{\partial \theta}\right]$, where $\boldsymbol{\epsilon}_{\phi'}(\mathbf{x}_t; t, \pi, y)$ is achieved by finetuning the pretrained 2D diffusion prior $\boldsymbol{\epsilon}_{\phi}(\boldsymbol{x}_t; t, y^{\pi})$ to align with the rendered image distribution $q^{\theta}(\boldsymbol{x} \mid \pi)$ via parameter efficient adaptation [18]. In

Method	Gradient of $\mathcal{L}(\boldsymbol{x}, y)$ w.r.t. $\boldsymbol{x} = g(\theta, \pi)$
SDS [40]	$\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}\right)\right]$
CSD [72]	$\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t, t\right)\right)\right]$
VSD [59]	$\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}_{\phi'}\left(\boldsymbol{x}_t; t, \pi, y\right)\right)\right]$
ASD (Ours)	$\mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_{t+\Delta t}; t + \Delta t, y^\pi\right)\right)\right]$

Table 1: Objectives of representative score distillation methods. ASD introduces Δt alongside t to align with the rendered image distribution $q^\theta(\boldsymbol{x} \mid \pi)$.

practice, this is conducted by alternatively optimizing θ and finetuning ϕ with the noise prediction objective $\|\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y\right) - \boldsymbol{\epsilon}\|_2^2$ [15] such that:

$$\mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon}_{\phi'}\left(\boldsymbol{x}_t; t, \pi, y\right) - \boldsymbol{\epsilon}\|_2^2\right] \leq \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\|\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}\|_2^2\right]. \tag{2}$$

The above equation reveals that **a better alignment with the distribution of $q^\theta(\boldsymbol{x} \mid \pi)$ can be achieved by a more accurate noise prediction**.

While VSD achieves state-of-the-art results in prompt-specific text-to-3D [14, 59], it changes the diffusion prior's parameters by alternately optimizing θ and finetuning ϕ. This forms a bi-level optimization, known to be problematic in generative adversarial training [55], and may be troublesome for training prompt-amortized text-to-3D models, because the change of pre-trained diffusion model might impairs its comprehension capability on a wide range of text-prompts. In specific, the pre-trained 2D diffusion model may have to sacrifice its generation capability in order to align with the distribution of rendered images, making it fail to produce good gradient for training the 3D generator.

3 Asynchronous Score Distillation (ASD)

3.1 Objective of ASD

From the above discussions in Sec. 2.2, it can be seen that one key issue in VSD is to minimize the noise prediction error so that the model output can be aligned with the desired distribution of rendered images. VSD achieves this goal via finetuning the pre-trained 2D diffusion model, which however sacrifices its comprehension capability on text prompts. One interesting question is: can we minimize the noise prediction error without changing the pre-trained diffusion network weights? Fortunately, we find that this is possible and in this section we present a new objective function to achieve this goal.

Recall that diffusion models solve the stochastic differential equation [50] via reversing the noise added along different stages, a.k.a. diffusion timestep $t \in \{T_{\max}, \ldots, T_{\min}\}$ via $\boldsymbol{x}_t = \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\epsilon}$ [15]. The influence of the noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$ on the image $\boldsymbol{x}$ is incrementally reduced as the process progresses from the initial timestep $T_{\max}$ to the final timestep $T_{\min}$, which is controlled by the scalars α_t and σ_t. Consequently, the diffusion model's noise prediction accuracy will vary with the timestep t, at which the identical noise $\boldsymbol{\epsilon}$ is added. To evaluate this, we consider a diffusion model with fixed image $\boldsymbol{x}$, noise $\boldsymbol{\epsilon}$ and condition y, but varied timestep t. We denote such a diffusion model as $\boldsymbol{\epsilon}(t)$ and explore how its prediction error, denoted by $e(t)$=$\|\boldsymbol{\epsilon}(t) - \boldsymbol{\epsilon}\|_2^2$, changes with t.

The model $\boldsymbol{\epsilon}(t)$ can be a pre-trained 2D diffusion model (such as Stable Diffusion [44]). We denote by $\boldsymbol{\epsilon}_{PT}(t)$ such a model, and investigate the behaviour of its noise prediction error, denoted by $e_{PT}(t)$. In Fig. 1, we plot the curve (*i.e.*, the blue colored curve) of $e_{PT}(t)$ versus t. We use a corpus with 15 text prompts from Magic3D [40] to draw this curve. For each prompt y, we generate 16 images with VSD [59]. Then for each image $\boldsymbol{x}$, we apply one instance of Gaussian noise $\boldsymbol{\epsilon}$ and conduct a single diffusion step with 100 distinct timesteps. The average noise reconstruction error is then calculated for these timesteps across all prompts and images. We can see from the curve of $e_{PT}(t)$ that earlier diffusion timesteps (*e.g.*, timestep 600) will have lower noise prediction error than later timesteps (*e.g.*, timestep 200). Such a trend holds for almost every image sample $\boldsymbol{x}$ and noise sample $\boldsymbol{\epsilon}$ because the well-trained diffusion model is frozen in our case. Since the noise prediction error declines from $T_{\min}$ (*i.e.*, late diffusion timestep) to $T_{\max}$ (*i.e.*, early diffusion timestep), we can conclude that for a given timestep t and a timestep shift $0 \leq \Delta t \leq T_{\max} - t$, the following inequality holds:

$$\mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon}_{\phi}\left(\boldsymbol{x}_{t+\Delta t}; t+\Delta t, y^{\pi}\right)-\boldsymbol{\epsilon}\right\|_2^2\right] \leq \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\left\|\boldsymbol{\epsilon}_{\phi}\left(\boldsymbol{x}_{t}; t, y^{\pi}\right)-\boldsymbol{\epsilon}\right\|_2^2\right], \quad (3)$$

which implies that **more accurate noise predictions can be achieved at earlier diffusion timesteps**.

The above property of diffusion models has also been observed by Yang *et al.* [68], who indicated that as the timestep shifts from $T_{\max}$ towards $T_{\min}$, the variance in noise prediction increases, as evidenced by the rising Lipschitz constants, which suggests an increased instability in noise prediction and larger noise prediction errors. Such a behavior can be observed in both $\boldsymbol{\epsilon}$-prediction and $\boldsymbol{v}$-prediction models, as well as in 2D and 3D diffusion models (please refer to *supplementary material* for details). This can be intuitively explained as follows. When $t \rightarrow T_{\max}$, $\boldsymbol{x}_t = \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\epsilon} \rightarrow \boldsymbol{\epsilon}$, then it is easier to achieve $\boldsymbol{\epsilon}_{\phi}\left(\boldsymbol{x}_t; t, y^{\pi}\right) \approx \boldsymbol{\epsilon}$ because the model can manage to copy the input as the output.

The similarity between Eq. 3 and the fine-tuning objective of VSD in Eq. 2 inspires us to investigate whether simply shifting earlier the timestep could fulfill the fine-tuning requirements of VSD without modifying the pre-trained 2D diffusion network parameters. Specifically, we employ the pretrained 2D diffusion model with shifted timestep to approximate the diffusion model of rendered images in Eq. 1 as $\boldsymbol{\epsilon}_{\theta}\left(\boldsymbol{x}_t; t, \pi, y\right) \triangleq \boldsymbol{\epsilon}_{\phi}\left(\boldsymbol{x}_{t+\Delta t}; t+\Delta t, y^{\pi}\right)$, resulting in the following Asynchronous Score Distillation (ASD) objective function:

$$\nabla_{\theta}\mathcal{L}_{\mathrm{ASD}}(\theta, y) \triangleq \mathbb{E}_{\pi,t,\boldsymbol{\epsilon}}\left[\omega(t)\left(\boldsymbol{\epsilon}_{\phi}\left(\boldsymbol{x}_t; t, y^{\pi}\right)-\boldsymbol{\epsilon}_{\phi}\left(\boldsymbol{x}_{t+\Delta t}; t+\Delta t, y^{\pi}\right)\right)\frac{\partial \boldsymbol{x}}{\partial \theta}\right]. \quad (4)$$

We can see that rather than iteratively fine-tuning the diffusion network as in VSD, ASD achieves similar goal by shifting the timestep t with an interval Δt in each step, which is much more efficient. One key variable introduced in ASD is the timestep shift Δt, which will be discussed in the next subsection.

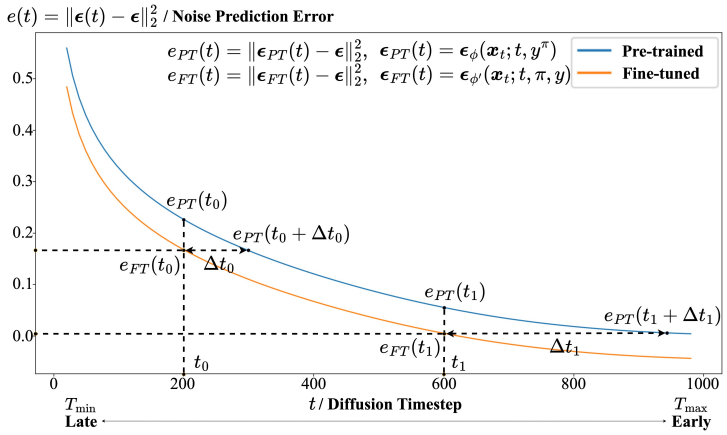

Fig. 1: Illustration of the noise prediction error of the pre-trained 2D diffusion model $\epsilon_{PT}(t)$ and that of the fine-tuned 2D diffusion model $\epsilon_{FT}(t)$. We can see that the curve of $e_{FT}(t)$ is positioned under that of $e_{PT}(t)$, and we can shift the timestep of $\epsilon_{PT}(t)$ to $\epsilon_{PT}(t + \Delta t)$ to approximate the noise prediction error of $\epsilon_{FT}(t)$.

3.2 The Setting of Timestep Shift Δt

Before discussing how to set the timestep shift Δt, let's plot another curve, *i.e.*, the noise prediction error of $\epsilon_\theta(x_t; t, \pi, y)$ w.r.t. timestep t. Actually, in the process of generating x with VSD, we will have the fine-tuned model $\epsilon_{\phi'}(\mathbf{x}_t; t, \pi, y)$ as the by-product, which is used to represent $\epsilon_\theta(x_t; t, \pi, y)$ in Eq. 1. Therefore, with fixed x, ϵ and y, the noise prediction error of the fine-tuned diffusion model, denoted by $\epsilon_{FT}(t)$, can be calculated as $e_{FT}(t) = \|\epsilon_{\phi'}(t) - \epsilon\|_2^2$.

The curve of $e_{FT}(t)$ w.r.t. t (*i.e.*, the yellow curve) is plotted in Fig. 1 by using the same data as in plotting $e_{PT}(t)$. We can see that the curve of $e_{FT}(t)$ is positioned under $e_{PT}(t)$ because $e_{FT}(t)$ is obtained by the fine-tuned diffusion model ϵ_{FT}. However, as mentioned in Sec. 2.2, this fine-tuning changes the weights of pre-trained diffusion model and might damage its ability in comprehending text-image pairs. Therefore, we propose to fix the pre-trained model $\epsilon_{PT}(t)$ but shift it to $\epsilon_{PT}(t + \Delta t)$ to approximate the desired $\epsilon_{FT}(t)$. Referring to Fig. 1, we could shift $\epsilon_{PT}(t)$ to an **earlier** timestep to achieve this goal. For example, at timestep t_0 and with a time shift $\Delta t_0 > 0$, we can use $\epsilon_{PT}(t_0 + \Delta t_0)$ to approximate the noise prediction error of $\epsilon_{FT}(t_0)$.

On the other hand, the magnitude of Δt will vary with t. Let's come to another timestep t_1 in Fig. 1, where t_1 is earlier than t_0. Because the decreasing speeds of both e_{PT} and e_{FT} will be reduced with t going to $T_{\max}$, the magnitude of Δt_1 will be increased to approximate $e_{FT}(t_1)$. In other words, the magnitude of Δt should grow when t goes from $T_{\min}$ to $T_{\max}$. We heuristically set this relationship as $\Delta t = \eta(t - T_{\min})$, where $\eta \in [0, 1]$ is a hyper-parameter that controls the length of shift range. Finally, it should be pointed out that the curves in Fig. 1 will vary a little for different training iterations, rendered images x and text prompts y. Therefore, Δt should fall into some range $S(t)$. In practice, we set $\Delta t \sim S(t) = \mathcal{U}[0, \eta(t - T_{\min})]$, which follows a uniform distribution within 0

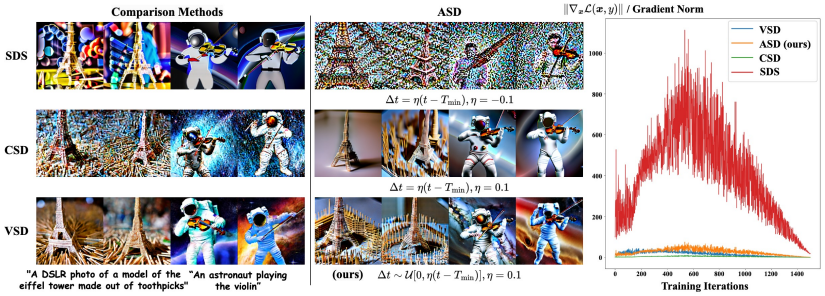

Fig. 2: Left and middle: 2D toy examples by SDS [40], CSD [72], VSD [59] and our proposed ASD. **Right:** Gradient norms generated by different methods.

Algorithm 1 Asynchronous Score Distillation (ASD)

Input: 3D representation θ; Text prompt y; Hyperparamter η; 2D diffusion prior $\boldsymbol{\epsilon}_\phi$

while *not converged* **do**
  Sample a camera pose π
  Render an image $\boldsymbol{x} = g(\theta, \pi)$
  Sample a timestep $t \sim \mathcal{U}[T_{\min}, T_{\max}]$, Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
  Sample a timestep shift $\Delta t \sim S(t) = \mathcal{U}\left[0, \eta\left(t - T_{\min}\right)\right]$
  $\boldsymbol{x}_t \leftarrow \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\epsilon}$, $\boldsymbol{x}_{t+\Delta t} \leftarrow \alpha_{t+\Delta t} \boldsymbol{x} + \sigma_{t+\Delta t} \boldsymbol{\epsilon}$
  Update θ with $\Delta\theta \leftarrow \omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_{t+\Delta t}; t + \Delta t, y^\pi\right)\right) \frac{\partial \boldsymbol{x}}{\partial \theta}$
end

and $\eta(t - T_{\min})$. The pseudo-code of ASD is summarized in **Alg. 1**, which can be applied to both prompt-specific and prompt-amortized text-to-3D tasks.

2D toy experiments. To verify the proposed timestep shift strategy, we follow the paradigm in [59] to test SDS, CSD, VSD and our ASD on 2D toy examples. The left column of Fig. 2 shows the results of SDS, CSD, VSD, and the middle column shows the results of ASD with different sampling strategies of Δt. One can see that the proposed sampling strategy $\Delta t \sim S(t) = \mathcal{U}\left[0, \eta\left(t - T_{\min}\right)\right]$ yields similar results to VSD [59]. Besides, we show the gradient norm produced by these score distillation methods in the right column of Fig. 2. One can see that the range of gradient norm produced by ASD is similar to that of VSD. However, the gradient norm of SDS is more than 10 times larger than ASD and VSD because it needs to set CFG=100 for convergence [40, 59, 72]. Such a large gradient may result in training instability. We append more 2D results in the *supplementary material* to further validate our proposed sampling strategy.

Text-to-3D Synthesis with ASD. As a score distillation method, ASD is open to the selection of 3D generator architectures [4, 17, 22, 33, 39]. The general pipeline of ASD for text-to-3D synthesis is shown in Fig. 3. It takes a rendered image as input and diffuses it in two timesteps t and $t + \Delta t$. The noise prediction difference is used as the gradient to optimize the 3D representation of generator. In this work, in addition to prompt-specific generation, as done in most existing score distillation works [16, 28, 40, 59, 64], we focus more on prompt-amortized text-to-3D and conduct thorough experiments to evaluate the effectiveness of

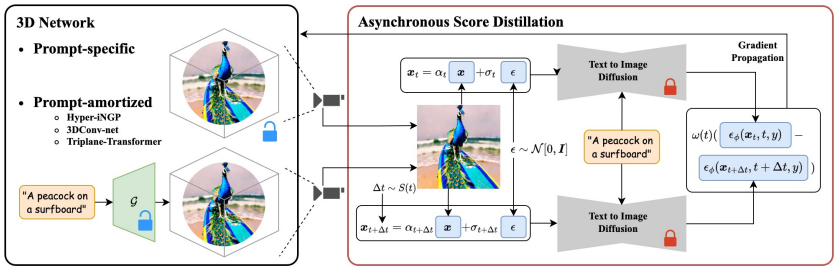

Fig. 3: Overview of Asynchronous Score Distillation (ASD). As illustrated in the left sub-figure, ASD can be employed for prompt-specific generation by optimizing 3D representations for each prompt, as well as for prompt-amortized generation by training a text-to-3D generator. The right sub-figure depicts how ASD uses the difference in noise predictions at asynchronous timesteps to update the 3D network parameters.

ASD with representative 3D generator architectures, including **Hyper-iNGP** and **3DConv-net**, by using **Stable Diffusion** as the 2D diffusion model. Hyper-iNGP is adopted by ATT3D [33], which integrates a prompt-agnostic hash-grid spatial encoding [39] with prompt-conditioned decoding layers to output color and density. 3DConv-net [4] is a 3D generator that maps the provided condition to voxel using 3D convolution. They are chosen in our experiments because they represent two groups of 3D generators that inject the condition through implicit mapping function [3,21] or explicit space representation [7,17,46,49]. The results are shown in Sec. 4. We also conduct experiments to demonstrate that ASD can work with other 2D diffusion models such as MVDream [48], and can be used to train other 3D generator architectures such as Triplane-Transformer [17]. More details of these experiments can be found in the *supplementary material*.

4 Experiments

4.1 Experimental Settings

Comparison Methods. We compare ASD with state-of-the-art score distillation methods, including SDS [40], CSD [72] and VSD [59]. We adhere to their official codes for training prompt-amortized text-to-3D networks. For example, the CFG [16] values for SDS, CSD and VSD are configured to 100, 1, and 7.5, respectively. In addition, we compare with existing prompt-amortized method ATT3D [33] (whose code is not released yet) by replicating its reported results.

Implementation Details. We employ VolSDF [69] to render images from the 3D generators. For Stable Diffusion, we employ SD-v2.1-base [1] for all score distillation methods for fair comparison. As configured in VSD [59], we set the CFG value as 7.5 for the pre-trained diffusion model in ASD, and 1 for the diffusion model of rendered images. The resolution of rendered images by Hyper-iNGP is set to 256×256, while that of 3DConv-net and Triplane-Transformer is set to 64×64 for GPU memory considerations. Other details are in the *supplementary material*.

Prompt Corpus. To thoroughly evalutate the capability of ASD in prompt-amortized text-to-3D synthesis, we employ multiple datasets encompassing a

range of text prompt quantities. **MG15** includes 15 prompts from Magic3D [29]; **DF415** comprises 415 prompts from DreamFusion [40]; and **AT2520** contains 2520 compositional prompts of animals from ATT3D [33]. DL17k contains 17k compositional prompts of human with daily activities, proposed by [25]. While AT2520 and DL17k provide a larger number of prompts than DF415, the prompt diversity of them is relatively low due to the predefined templates.

To test ASD's performance with an even larger scale of prompts, we introduce a novel prompt corpus named **CP100k**. This corpus consists of 100,000 text prompts filtered from the image descriptions collected by Cap3D [34], which was developed to test text-to-image model performance. To the best of our knowledge, it is the first time to evaluate score distillation methods on such a scale of text prompts. Meanwhile, it should be clarified that this work is focused on examining the score distillation performance rather than prompt generalization, so the test prompts share the same distribution as training prompts. More details of the prompt corpus are in *supplementary material*.

Evaluation Metrics. We render 120 surrounding view images as the 3D synthesis result from each prompt. Similar to previous text-to-3D works [25, 33, 33, 40], we compute the CLIP recall, *i.e.*, the classification accuracy by applying CLIP model to the rendered images to predict the correct text prompt, as one performance metric, denoted by "R@1". Additionally, we calculate the CLIP text-image similarity between generated images and input prompts as another metric [54, 60], denoted by "Sim".

4.2 Evaluation Results

Results with iNGP/Hyper-iNGP as 3D Representation. The iNGP [39] architecture is designed for prompt-specific text-to-3D generation. Hyper-iNGP has the same spatial encoding as iNGP except that the weights of the decoding layer depend on the text prompt. To eliminate the effect caused by architecture difference as much as possible, we adopt iNGP for prompt-specific text-to-3D tasks, and Hyper-iNGP for prompt-amortized tasks. Our experiments are carried out on the MG15 dataset. For prompt-specific tasks, we optimize an individual iNGP [39] for each MG15 prompt; while for the prompt-amortized tasks, we train a single Hyper-iNGP [33] across all MG15 prompts. We also compare our results with ATT3D [33], which is among the first to apply Hyper-iNGP to prompt-amortized text-to-3D tasks. ATT3D employs SDS for training and uses soft-shading [40] (denoted as * in Tab. 2) for rendering.

The qualitative and quantitative results are shown in Fig. 4 and Tab. 2, respectively. We can see that the existing methods suffer from performance decrease when transiting from prompt-specific to prompt-amortized tasks, as evidenced by the decreased CLIP similarity and recall in Tab. 2. It is worth mentioning that training Hyper-net with SDS requires turning on the spectral normalization [38] in the linear layers, otherwise the training will fail due to numerical instability. This observation is consistent with what reported in ATT3D [33]. This is because SDS suffers from large gradient norm (please also refer to Fig. 2 and the discussions therein), which makes Hyper-iNGP hard to

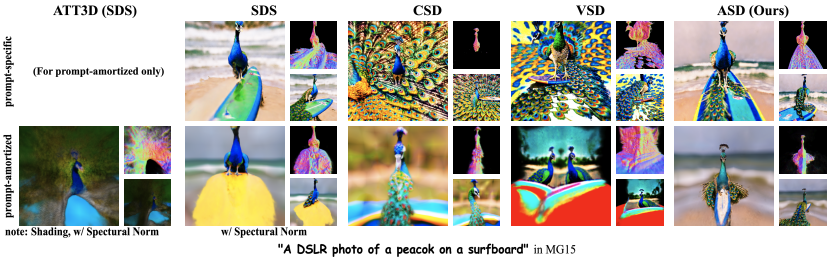

Fig. 4: Qualitative comparison on prompt-specific (with iNGP as the 3D representation) and prompt-amortized (with Hyper-iNGP as the 3D generator) text-to-3D results by SDS [40], CSD [72], VSD [59], ATT3D [33] and our ASD methods.

Reference	Method	Sim ↑	R@1 ↑	Method	Sim ↑	R@1 ↑
ATT3D [33]	-	-	-	Hyper-iNGP* + SDS	0.195	0.468
DreamFusion [40]	iNGP + SDS	0.288	**1.000**	Hyper-iNGP + SDS	0.257	0.918
Classifier [16]	iNGP + CSD	0.280	0.936	Hyper-iNGP + CSD	0.264	0.972
ProlificDreamer [59]	iNGP + VSD	0.276	0.932	Hyper-iNGP + VSD	0.259	0.987
Ours	iNGP + ASD	**0.289**	**1.000**	Hyper-iNGP + ASD	**0.284**	**1.000**

Table 2: Quantitative comparison on prompt-specific (with iNGP as the 3D representation) and prompt-amortized (with Hyper-iNGP as the 3D generator) text-to-3D results by SDS [40], CSD [72], VSD [59], ATT3D [33] and our ASD methods.

converge. As can be seen in Fig. 4, ATT3D results in wrong geometry by using soft shading and SDS for training. For CSD, we see that it fails to optimize the full geometry, as shown by the shrunk peacock in both prompt-amortized and prompt-amortized results. For VSD, it tends to generate content drifts [48], resulting in repetitive patterns and abnormal geometry. It may fail to generate reasonable contents in both prompt-specific and prompt-amortized tasks. In contrast, our proposed ASD works very stable across the two tasks, yielding not only outstanding quantitative scores but also high quality 3D contents.

Results with 3DConv-net as 3D Generator. The issues of existing score distillation methods either persist or become more pronounced when replacing Hyper-iNGP to 3DConv-net as the 3D generator. We find that training SDS with 3DConv-net always fails within several thousand iterations, even using spectral or other normalization techniques. This issue stems from that deeper network is more sensitive to large gradients [13] caused by SDS. Therefore, we only compare the results of other methods in Fig. 5. We see that CSD outputs acceptable results on AT2520, but its results on DF415, which has more varied prompts, are consistently smaller than anticipated. Such a phenomenon has been observed when Hyper-iNGP is used as the generator, which underlines CSD's inability to reliably guide the 3D generator to produce geometries aligned with the text prompts. As for VSD, it leads to rather abnormal results, failing to match the text prompts. This can be attributed to its fine-tuning of the pre-trained 2D diffusion model, which severely compromises VSD's text-image comprehending ability. In comparison, our proposed ASD, with 3DConv-net as the generator,

Fig. 5: Qualitative comparison among CSD [72], VSD [59] and our ASD (with 3DConv-net as generator) on AT2520 and DF415 corpuses. SDS is not compared because it encounters numerical instability in this experiment.

Method	DF415		AT2520		CP100k	
	Sim ↑	R@1 ↑	Sim ↑	R@1 ↑	Sim ↑	R@1 ↑
SDS	×	×	×	×	×	×
CSD	0.176	0.062	0.279	0.037	0.195	0.108
VSD	0.158	0.002	0.115	0.001	0.103	0.000
ASD (ours)	**0.237**	**0.276**	**0.285**	**0.058**	**0.199**	**0.117**

Table 3: Quantitative comparison on prompt-amortized text-to-3D with 3DConv-net as generator. Symbol × denotes that the training fails due to numerical instability.

yields improved outcomes, as evidenced by the visual results in Fig. 5 and the enhanced metric scores in Tab. 3.

Scalability. In this section, we evaluate the scalability of competing methods by using as many as 100k prompts in the CP100k dataset with 3DConv-net as the generator. The results are shown in Fig. 6 and Tab. 3. Due to the issue of numerical instability, SDS is not involved in this experiment. We can see that the outcomes of CSD are significantly diminished with uniformly small-sized shapes across all prompts. There is also a lack of variety since most outputs exhibit similar patterns. The results of VSD are also degenerated, displaying almost identical and anomalous outcomes for the text prompts. This resembles the phenomenon of mode collapse often encountered in bi-level optimization [55], which also highlights the importance of fixing the 2D diffusion model when training with such a large number of text prompts. In comparison, ASD is able to produce much higher quality outcomes across the text prompts, showcasing its capability in large-scale training with numerous text prompts as inputs.

In addition to the above comparisons, we also evaluate the effectiveness of ASD when using other 2D diffusion models. We employ MVDream [48] to illustrate ASD's generality to diffusion prior models. Besides, we compare against data-driven methods to prove that ASD is advantageous in tackling diverse input of text prompts. Please see *supplementary material* for detail.

4.3 Ablation Study

In this section, we perform ablation studies to evaluate the settings of timestep shift $\Delta t \sim S(t) = \mathcal{U}\left[0, \eta\left(t - T_{\text{min}}\right)\right]$ from several aspects. The qualitative and

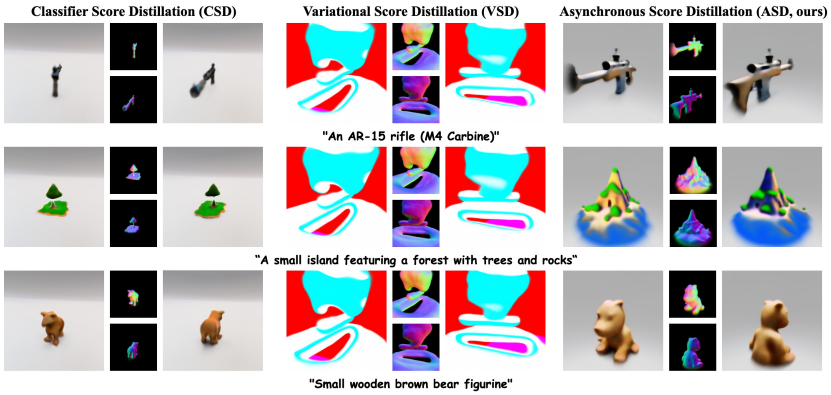

Fig. 6: The scalability comparison with CSD [72] and VSD [59] on CP100k corpus.

quantitative results are shown in Fig. 7 and Tab. 4, respectively. We provide extensive ablation studies in the *supplementary material* to show that the proposed strategy is also effective when using MVDream as the diffusion model.

Importance of Timestep Shift. We use $\eta = 0$ (*i.e.*, no timestep shift) as a baseline to evaluate the necessity of introducing timestep shift Δt. From Fig. 7 and Tab. 4, we see that while it can generate plausible results, the model is prone to generating shapes that do not make sense, such as the so-called Janus problem [2]. Examples include a frog with an extra eye, robot face with block-like features, and a peacock with tails at both the front and back. This is because the non-shifted diffusion model will align more with the 2D image distribution, tending to generate redundant contents and unreasonable geometry along the training. By introducing a timestep shift, our proposed ASD demonstrates advantages in achieving more coherent and visually pleasing results.

Range of Timestep Shift. By setting $\eta = 0.2$, we allow Δt to be sampled from a large range. However, this might not be a good choice. In the extreme case, for any timestep t we can set a large interval Δt such that $t + \Delta t = T_{\max}$, then the noise prediction becomes $\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; t, y^\pi) \approx \boldsymbol{\epsilon}$, so that ASD is degraded to SDS, which cannot perform well under CFG=7.5 [40]. In practice, we find a larger η tends to result 3D contents with larger size and rounded shapes, *e.g.*, the peacock with closer views, or the frog with larger size, as shown in Fig. 7. Therefore, we set $\eta = 0.1$ in all our experiments.

Deterministic or Random Shift. If we set $\Delta t = \eta\,(t - T_{\min})$, it assumes that the diffusion model of rendered images can be approximated by the pre-trained one with a fixed and deterministic timestep shift. As shown in Fig. 7 and Tab. 4, it reduces the chance to generate correct geometry and colors. Randomly sampling Δt in a range is more effective, which is adopted in our method.

5 Conclusion and Limitations

In this paper, we presented Asynchronous Score Distillation (ASD), a novel score distillation method that can assist 2D diffusion prior in training 3D generators

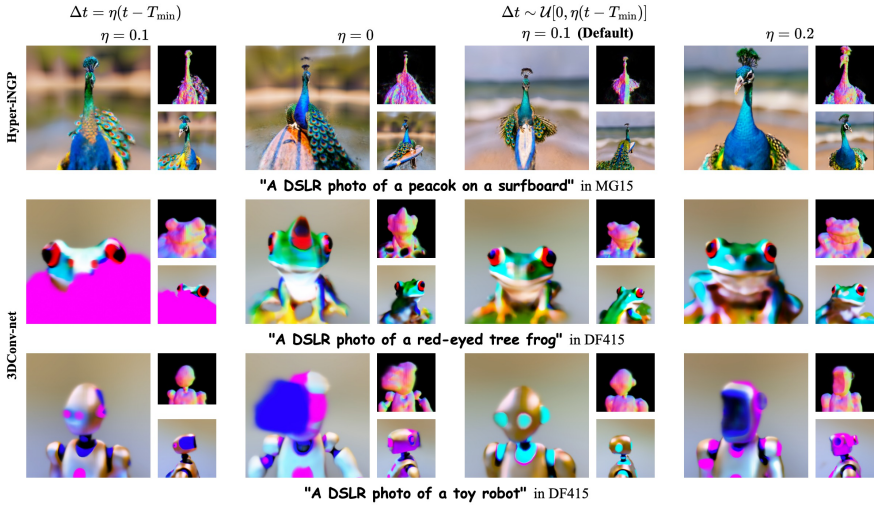

Fig. 7: The qualitative results of the ablation study on the timestep interval Δt.

	Param	Sim ↑	R@1 ↑
$\Delta t = \eta(t - T_{\min})$	$\eta = 0.1$	0.214	0.178
	$\eta = 0.2$	0.214	0.180
$\Delta t \sim \mathcal{U}[0, \eta(t - T_{\min})]$	$\eta = 0$	0.235	0.267
	$\eta = 0.1$	**0.237**	**0.276**
	$\eta = 0.2$	0.229	0.237

Table 4: The quantitative results of the ablation study on the timestep interval Δt.

with a scalable size of text prompts. By shifting the diffusion timestep to earlier stages, our ASD can effectively predict the noise prediction error to align the diffusion model with the distribution of rendered images, while preserving the superior text comprehension capability of pre-trained models, thus facilitating stable training with high-fidelity generation results. Our extensive experiments revealed that ASD performed consistently well on datasets of various sizes, being able to manage as much as 100k prompts. Though ASD has shown improvements over earlier score distillation approaches, there remain some limitations.

For man-made objects that have very regular shapes, such as chairs or airplanes, the performance of our model will lag behind those data-driven methods, which benefit from an abundance of relevant data. We foresee opportunities to combine the advantages of data-driven and score distillation methodologies to improve text-to-3D capabilities in a more comprehensive manner in the future research.

Acknowledgement

This work is supported in part by the Beijing Science and Technology Plan Project Z231100005923033, and the InnoHK program.

References

1. Stable-diffusion-v2.1-base. https://huggingface.co/stabilityai/stable-diffusion-2-1-base
2. Armandpour, M., Zheng, H., Sadeghian, A., Sadeghian, A., Zhou, M.: Re-imagine the negative prompt algorithm: Transform 2d diffusion into 3d, alleviate janus problem and beyond. arXiv preprint arXiv:2304.04968 (2023)
3. Babu, S., Liu, R., Zhou, A., Maire, M., Shakhnarovich, G., Hanocka, R.: Hyperfields: Towards zero-shot generation of nerfs from text. arXiv preprint arXiv:2310.17075 (2023)
4. Bahmani, S., Park, J.J., Paschalidou, D., Yan, X., Wetzstein, G., Guibas, L., Tagliasacchi, A.: Cc3d: Layout-conditioned generation of compositional 3d scenes. arXiv preprint arXiv:2303.12074 (2023)
5. Balaji, Y., Nah, S., Huang, X., Vahdat, A., Song, J., Kreis, K., Aittala, M., Aila, T., Laine, S., Catanzaro, B., et al.: ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. arXiv preprint arXiv:2211.01324 (2022)
6. Cao, Z., Hong, F., Wu, T., Pan, L., Liu, Z.: Large-vocabulary 3d diffusion model with transformer. arXiv preprint arXiv:2309.07920 (2023)
7. Chan, E.R., Lin, C.Z., Chan, M.A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L.J., Tremblay, J., Khamis, S., et al.: Efficient geometry-aware 3d generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16123–16133 (2022)
8. Chen, R., Chen, Y., Jiao, N., Jia, K.: Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. arXiv preprint arXiv:2303.13873 (2023)
9. Chen, Z., Wang, F., Liu, H.: Text-to-3d using gaussian splatting. arXiv preprint arXiv:2309.16585 (2023)
10. Deitke, M., Liu, R., Wallingford, M., Ngo, H., Michel, O., Kusupati, A., Fan, A., Laforte, C., Voleti, V., Gadre, S.Y., et al.: Objaverse-xl: A universe of 10m+ 3d objects. arXiv preprint arXiv:2307.05663 (2023)
11. Ding, L., Dong, S., Huang, Z., Wang, Z., Zhang, Y., Gong, K., Xu, D., Xue, T.: Text-to-3d generation with bidirectional diffusion using both 2d and 3d priors. arXiv preprint arXiv:2312.04963 (2023)
12. Guo, P., Hao, H., Caccavale, A., Ren, Z., Zhang, E., Shan, Q., Sankar, A., Schwing, A.G., Colburn, A., Ma, F.: Stabledreamer: Taming noisy score distillation sampling for text-to-3d. arXiv preprint arXiv:2312.02189 (2023)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
14. He, Y., Bai, Y., Lin, M., Zhao, W., Hu, Y., Sheng, J., Yi, R., Li, J., Liu, Y.J.: T^3bench: Benchmarking current progress in text-to-3d generation. arXiv preprint arXiv:2310.02977 (2023)
15. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
16. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
17. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: Lrm: Large reconstruction model for single image to 3d. arXiv preprint arXiv:2311.04400 (2023)

18. Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685 (2021)
19. Huang, Y., Wang, J., Shi, Y., Qi, X., Zha, Z.J., Zhang, L.: Dreamtime: An improved optimization strategy for text-to-3d content creation. arXiv preprint arXiv:2306.12422 (2023)
20. Jiang, L., Wang, L.: Brightdreamer: Generic 3d gaussian generative framework for fast text-to-3d synthesis. arXiv preprint arXiv:2403.11273 (2024)
21. Jun, H., Nichol, A.: Shap-e: Generating conditional 3d implicit functions. arXiv preprint arXiv:2305.02463 (2023)
22. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics (ToG) **42**(4), 1–14 (2023)
23. Koster, R.: Theory of fun for game design. " O'Reilly Media, Inc." (2013)
24. Lee, K., Sohn, K., Shin, J.: Dreamflow: High-quality text-to-3d generation by approximating probability flow. arXiv preprint arXiv:2403.14966 (2024)
25. Li, M., Zhou, P., Liu, J.W., Keppo, J., Lin, M., Yan, S., Xu, X.: Instant3d: Instant text-to-3d generation. arXiv preprint arXiv:2311.08403 (2023)
26. Li, W., Chen, R., Chen, X., Tan, P.: Sweetdreamer: Aligning geometric priors in 2d diffusion for consistent text-to-3d. arXiv preprint arXiv:2310.02596 (2023)
27. Li, Z., Chen, Y., Zhao, L., Liu, P.: Mvcontrol: Adding conditional control to multi-view diffusion for controllable text-to-3d generation. arXiv preprint arXiv:2311.14494 (2023)
28. Liang, Y., Yang, X., Lin, J., Li, H., Xu, X., Chen, Y.: Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. arXiv preprint arXiv:2311.11284 (2023)
29. Lin, C.H., Gao, J., Tang, L., Takikawa, T., Zeng, X., Huang, X., Kreis, K., Fidler, S., Liu, M.Y., Lin, T.Y.: Magic3d: High-resolution text-to-3d content creation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 300–309 (2023)
30. Lin, Y., Clark, R., Torr, P.: Dreampolisher: Towards high-quality text-to-3d generation via geometric diffusion. arXiv preprint arXiv:2403.17237 (2024)
31. Liu, Y.T., Luo, G., Sun, H., Yin, W., Guo, Y.C., Zhang, S.H.: Pi3d: Efficient text-to-3d generation with pseudo-image diffusion. arXiv preprint arXiv:2312.09069 (2023)
32. Liu, Z., Li, Y., Lin, Y., Yu, X., Peng, S., Cao, Y.P., Qi, X., Huang, X., Liang, D., Ouyang, W.: Unidream: Unifying diffusion priors for relightable text-to-3d generation. arXiv preprint arXiv:2312.08754 (2023)
33. Lorraine, J., Xie, K., Zeng, X., Lin, C.H., Takikawa, T., Sharp, N., Lin, T.Y., Liu, M.Y., Fidler, S., Lucas, J.: Att3d: Amortized text-to-3d object synthesis. arXiv preprint arXiv:2306.07349 (2023)
34. Luo, T., Rockwell, C., Lee, H., Johnson, J.: Scalable 3d captioning with pretrained models. Advances in Neural Information Processing Systems **36** (2024)
35. Ma, Y., Fan, Y., Ji, J., Wang, H., Sun, X., Jiang, G., Shu, A., Ji, R.: X-dreamer: Creating high-quality 3d content by bridging the domain gap between text-to-2d and text-to-3d generation. arXiv preprint arXiv:2312.00085 (2023)
36. Mercier, A., Nakhli, R., Reddy, M., Yasarla, R., Cai, H., Porikli, F., Berger, G.: Hexagen3d: Stablediffusion is just one step away from fast and diverse text-to-3d generation. arXiv preprint arXiv:2401.07727 (2024)

37. Metzer, G., Richardson, E., Patashnik, O., Giryes, R., Cohen-Or, D.: Latent-nerf for shape-guided generation of 3d shapes and textures. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12663–12673 (2023)
38. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. arXiv preprint arXiv:1802.05957 (2018)
39. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (ToG) **41**(4), 1–15 (2022)
40. Poole, B., Jain, A., Barron, J.T., Mildenhall, B.: Dreamfusion: Text-to-3d using 2d diffusion. arXiv preprint arXiv:2209.14988 (2022)
41. Qian, G., Cao, J., Siarohin, A., Kant, Y., Wang, C., Vasilkovsky, M., Lee, H.Y., Fang, Y., Skorokhodov, I., Zhuang, P., et al.: Atom: Amortized text-to-mesh using 2d diffusion. arXiv preprint arXiv:2402.00867 (2024)
42. Qiu, L., Chen, G., Gu, X., Zuo, Q., Xu, M., Wu, Y., Yuan, W., Dong, Z., Bo, L., Han, X.: Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. arXiv preprint arXiv:2311.16918 (2023)
43. Ren, X., Huang, J., Zeng, X., Museth, K., Fidler, S., Williams, F.: Xcube: Large-scale 3d generative modeling using sparse voxel hierarchies. arXiv preprint (2023)
44. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)
45. Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M., Coombes, T., Katta, A., Mullis, C., Wortsman, M., 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)
46. Schwarz, K., Sauer, A., Niemeyer, M., Liao, Y., Geiger, A.: Voxgraf: Fast 3d-aware image synthesis with sparse voxel grids. Advances in Neural Information Processing Systems **35**, 33999–34011 (2022)
47. Shen, T., Gao, J., Yin, K., Liu, M.Y., Fidler, S.: Deep marching tetrahedra: a hybrid representation for high-resolution 3d shape synthesis. Advances in Neural Information Processing Systems **34**, 6087–6101 (2021)
48. Shi, Y., Wang, P., Ye, J., Long, M., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3d generation. arXiv preprint arXiv:2308.16512 (2023)
49. Sitzmann, V., Thies, J., Heide, F., Nießner, M., Wetzstein, G., Zollhofer, M.: Deepvoxels: Learning persistent 3d feature embeddings. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2437–2446 (2019)
50. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020)
51. Tang, B., Wang, J., Wu, Z., Zhang, L.: Stable score distillation for high-quality 3d generation. arXiv preprint arXiv:2312.09305 (2023)
52. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: Lgm: Large multi-view gaussian model for high-resolution 3d content creation. arXiv preprint arXiv:2402.05054 (2024)
53. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. arXiv preprint arXiv:2309.16653 (2023)
54. Tang, Z., Gu, S., Wang, C., Zhang, T., Bao, J., Chen, D., Guo, B.: Volumediffusion: Flexible text-to-3d generation with efficient volumetric encoder. arXiv preprint arXiv:2312.11459 (2023)

55. Thanh-Tung, H., Tran, T.: Catastrophic forgetting and mode collapse in gans. In: 2020 international joint conference on neural networks (ijcnn). pp. 1–10. IEEE (2020)
56. Tsalicoglou, C., Manhardt, F., Tonioni, A., Niemeyer, M., Tombari, F.: Textmesh: Generation of realistic 3d meshes from text prompts. arXiv preprint arXiv:2304.12439 (2023)
57. Vilesov, A., Chari, P., Kadambi, A.: Cg3d: Compositional generation for text-to-3d via gaussian splatting. arXiv preprint arXiv:2311.17907 (2023)
58. Wang, H., Du, X., Li, J., Yeh, R.A., Shakhnarovich, G.: Score jacobian chaining: Lifting pretrained 2d diffusion models for 3d generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12619–12629 (2023)
59. Wang, Z., Lu, C., Wang, Y., Bao, F., Li, C., Su, H., Zhu, J.: Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. arXiv preprint arXiv:2305.16213 (2023)
60. Wei, Y., Zhang, Y., Ji, Z., Bai, J., Zhang, L., Zuo, W.: Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. arXiv preprint arXiv:2302.13848 (2023)
61. Wohlgenannt, I., Simons, A., Stieglitz, S.: Virtual reality. Business & Information Systems Engineering **62**, 455–461 (2020)
62. Wu, R., Sun, L., Ma, Z., Zhang, L.: One-step effective diffusion network for real-world image super-resolution. arXiv preprint arXiv:2406.08177 (2024)
63. Wu, T., Zhang, J., Fu, X., Wang, Y., Ren, J., Pan, L., Wu, W., Yang, L., Wang, J., Qian, C., et al.: Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 803–814 (2023)
64. Wu, Z., Zhou, P., Yi, X., Yuan, X., Zhang, H.: Consistent3d: Towards consistent high-fidelity text-to-3d generation with deterministic sampling prior. arXiv preprint arXiv:2401.09050 (2024)
65. Xie, K., Lorraine, J., Cao, T., Gao, J., Lucas, J., Torralba, A., Fidler, S., Zeng, X.: Latte3d: Large-scale amortized text-to-enhanced3d synthesis. arXiv preprint arXiv:2403.15385 (2024)
66. Xu, Y., Tan, H., Luan, F., Bi, S., Wang, P., Li, J., Shi, Z., Sunkavalli, K., Wetzstein, G., Xu, Z., et al.: Dmv3d: Denoising multi-view diffusion using 3d large reconstruction model. arXiv preprint arXiv:2311.09217 (2023)
67. Yang, Z., Feng, R., Zhang, H., Shen, Y., Zhu, K., Huang, L., Zhang, Y., Liu, Y., Zhao, D., Zhou, J., et al.: Eliminating lipschitz singularities in diffusion models. arXiv preprint arXiv:2306.11251 (2023)
68. Yang, Z., Feng, R., Zhang, H., Shen, Y., Zhu, K., Huang, L., Zhang, Y., Liu, Y., Zhao, D., Zhou, J., et al.: Lipschitz singularities in diffusion models. In: The Twelfth International Conference on Learning Representations (2023)
69. Yariv, L., Gu, J., Kasten, Y., Lipman, Y.: Volume rendering of neural implicit surfaces. Advances in Neural Information Processing Systems **34**, 4805–4815 (2021)
70. Yi, T., Fang, J., Wu, G., Xie, L., Zhang, X., Liu, W., Tian, Q., Wang, X.: Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. arXiv preprint arXiv:2310.08529 (2023)
71. Yu, X., Xu, M., Zhang, Y., Liu, H., Ye, C., Wu, Y., Yan, Z., Zhu, C., Xiong, Z., Liang, T., et al.: Mvimgnet: A large-scale dataset of multi-view images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9150–9161 (2023)

72. Yu, X., Guo, Y.C., Li, Y., Liang, D., Zhang, S.H., Qi, X.: Text-to-3d with classifier score distillation. arXiv preprint arXiv:2310.19415 (2023)
73. Zhao, M., Zhao, C., Liang, X., Li, L., Zhao, Z., Hu, Z., Fan, C., Yu, X.: Efficientdreamer: High-fidelity and robust 3d creation via orthogonal-view diffusion prior. arXiv preprint arXiv:2308.13223 (2023)
74. Zhao, R., Wang, Z., Wang, Y., Zhou, Z., Zhu, J.: Flexidreamer: Single image-to-3d generation with flexicubes. arXiv preprint arXiv:2404.00987 (2024)
75. Zhou, L., Shih, A., Meng, C., Ermon, S.: Dreampropeller: Supercharge text-to-3d generation with parallel sampling. arXiv preprint arXiv:2311.17082 (2023)