VividDreamer: Invariant Score Distillation for Hyper-Realistic Text-to-3D Generation

Wenjie Zhuo[1,2], Fan Ma[2], Hehe Fan[2], and Yi Yang[1,2†]

[1] State Key Laboratory of Brain-machine Intelligence, Zhejiang University
[2] ReLER Lab, CCAI, Zhejiang University
wenjiezhuo@zju.edu.cn

Abstract. This paper presents Invariant Score Distillation (ISD), a novel method for high-fidelity text-to-3D generation. ISD aims to tackle the over-saturation and over-smoothing problems in Score Distillation Sampling (SDS). In this paper, SDS is decoupled into a weighted sum of two components: the reconstruction term and the classifier-free guidance term. We experimentally found that over-saturation stems from the large classifier-free guidance scale and over-smoothing comes from the reconstruction term. To overcome these problems, ISD utilizes an invariant score term derived from DDIM sampling to replace the reconstruction term in SDS. This operation allows the utilization of a medium classifier-free guidance scale and mitigates the reconstruction-related errors, thus preventing the over-smoothing and over-saturation of results. Extensive experiments demonstrate that our method greatly enhances SDS and produces realistic 3D objects through single-stage optimization.

Keywords: SDS · Text-to-3D Generation · Invariant Score Distillation

1 Introduction

This paper considers text-to-3D content generation [3, 19, 37, 40, 53, 55]. It is a fundamental task in computer vision with the potential to benefit various downstream applications. Benefiting from large-scale text-image pairs, there recently emerges a new 3D generation method, called 2D lifting methods. The main idea of these methods is to utilize text-to-image generative models [13,24,42,47,49] to guide the 3D creation. Early methods like DreamFields [18] and CLIP-forge [46] use the CLIP [24, 42] as guidance. Recently, inspired by the tremendous success of diffusion models [13, 47, 49] in the realm of image generation and conditional image generation [36, 43, 44, 56], DreamFusion [40] introduces a loss based on probabilistic density distillation [38], called Score Distillation Sampling (SDS). They utilize pre-trained diffusion priors to optimize 3D representations [22,34,35] by steering their rendered images toward regions with higher probability density,

†Corresponding author.
Our code is available at https://github.com/SupstarZh/VividDreamer.



Fig. 1: Examples generated by our framework. Our methods can generate detailed and high-fidelity 3D objects from a wide range of textual prompts.

conditioned on the text. This technology has sparked an abundance of subsequent research efforts. For example, Liu *et al.* [28] propose Zero123, which introduces SDS into the single-image reconstruction task [9, 27, 30, 41]. However, despite its widespread impact, the 3D objects generated by SDS frequently suffer from over-saturation and over-smoothing problems [1, 53, 63].

In this paper, we take an in-depth analysis of these issues. We decompose the SDS into two components, one is the reconstruction term: $\epsilon_\phi(z_t; y, t) - \epsilon$, and the other is the classifier-free guidance term: $\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$. We use each term individually as loss for text-to-image generation, and find two problems: 1) using the reconstruction term alone not only lacks semantic alignment functionality but also deteriorates the originally high-quality images. 2) using the classifier-free guidance term alone suffers from serious over-saturation (as shown in Fig. 2). SDS avoids problem 1 by assigning a large guidance scale (100.0) to the classifier-free guidance term, but this exacerbates problem 2.

To overcome these problems, we further analyze the causes of problem 1 and focus on optimizing the reconstruction term. We find the function of this item is actually to reconstruct the noise image in single-step, and this leads to problem 1 from two aspects: 1) At the beginning of the generation, reconstructing semantic-free images hinders semantic alignment. 2) When the image is aligned with the condition, the inconsistency of the single-step reconstruction destroys the image and causes over-smoothing. To address these, we propose the Invariant Score Distillation (ISD) method, which replaces the reconstruction item with

the invariant score term. This term no longer reconstructs the noisy image, but aligns the noisy image to the previous time step state. This brings two benefits: Firstly, the invariant score prior can be continuously refined during the generation process to provide more details for the results while avoiding errors caused by single-step reconstruction, avoiding over-smoothing and details lost problems. Secondly, the introduction of invariant score prior allows us to use a conventional guidance scale (7.5) for text-to-3D generation, avoiding the over-saturation problem. We also conduct extensive experiments to verify the effectiveness of our method. The results show that our method can generate realistic and detailed 3D objects. Compared with existing methods, our method shows strong competitiveness both qualitatively and quantitatively.

Overall, our contributions can be summarized as follows:

(1) We decompose the SDS into a classifier-free guidance term and a reconstruction term, and identify the over-saturation comes from the large classifier-free guidance scale, and the over-smoothing stems from the reconstruction term.

(2) We propose a novel Invariant Score Distillation (ISD) to overcome the aforementioned issues. ISD replaces the reconstruction term with an invariant score prior derived from DDIM, avoiding the inherent problems of SDS.

(3) We verify the effectiveness of our method through quantitative and qualitative experiments. The results show that our method can generate high-fidelity 3D objects in single-stage optimization and is highly competitive compared with existing methods.

2 Related Work

2.1 Text to image Generation

Recently, significant progress has been made in text-to-image generation [24, 42, 43, 61, 64]. Among these, diffusion models [13, 20, 32, 47, 49] are particularly notable for their outstanding performance. These models, once trained, are capable of transforming Gaussian noise into high-definition, realistic samples through multiple iterations. To enable conditional image generation, Dhariwal *et al*. [8] introduces classifier guidance. This approach requires training an additional noisy image classifier, thereby needing extra gradient guidance during sampling, and limiting generated image variety. To address these issues, Ho *et al*. proposes classifier-free guidance [14], eliminating the need for an explicit classifier. This method trains the diffusion model in both conditioned and unconditioned modes, it modifies the model by Eq. 1:

$$\tilde{\epsilon}_\phi(z_t; y, t) = (1 + w)\epsilon_\phi(z_t; y, t) - w\epsilon_\phi(z_t; y = \varnothing, t) \tag{1}$$

where $\varnothing$ is an empty text prompt representing the unconditional case, and w is a guidance scale, used to control the influence of conditional information. Building upon this methodology, the introduction of models such as GLIDE [36], Imagen [44], and Stable Diffusion [43] have propelled diffusion models to excel in the text-to-image generation domain. Recently, the influence of these models has also extended to text-to-3D generation.

2.2 Text to 3D Generation

Text-to-3D generation refers to creating 3D assets based on text descriptions. Current works can be categorized into two main types: (i) Native 3D methods that directly train a new diffusion model with 3D data [15,19,23,37,60] and (ii) 2D lifting methods that use the pre-trained 2D text-to-image diffusion models to guide the 3D assets generation [4,10,17,21,25,26,40,51–54,57,59,62,63]. Upon completion of pre-training, the former can rapidly generate 3D-consistent assets. However, due to the scarcity and expense of 3D datasets [2,6,7], the performance and diversity of current 3D diffusion models are significantly constrained.

2D lifting methods leverage models pre-trained on rich 2D text-image datasets to guide the generation of 3D assets [22,34,35]. Dreamfusion [40] proposes Score Distillation Sampling (SDS) to utilize diffusion models for guiding the generation of 3D assets. Despite the popularity of this approach, empirical observations reveal that SDS often suffers from over-saturation and over-smoothing issues [4, 17, 26, 53, 59, 63]. To address these challenges, Magic3D [26] adopts a two-stage method by combining NeRF [34] with mesh fine-tuning, Fantasia3D [4] takes a unique approach by decoupling geometry and appearance, facilitating the generation of explicit meshes and textures. Additionally, some methods refine the loss function to promote the quality of created 3D models. HiFA [63] introduces an image-level loss term based on the original SDS. ProlificDreamer [53] proposes VSD, it replaces the random Gaussian noise with a learned score function of noisy rendered images conditioned on the camera pose c. NFSD [21] decomposes components of SDS and provides experimental improvements to the loss function, while CSD [59] analyzes the components of the SDS and directly uses the classifier score term as guidance. Recently, with the rise of the new rendering technology 3DGS [22], a new batch of excellent works has emerged [5,25,50,58].

3 Preliminaries

3.1 Score Distillation Sampling

Score Distillation Sampling (SDS) [40] utilizes a pre-trained text-to-image diffusion model ϕ to guide the 3D representation parameterized by θ. Specifically, for a given camera pose π, we denote $x = g(\theta; \pi)$ as the image rendered from a differentiable rendering function g. SDS ensures that the images obtained from any viewpoint via the differentiable rendering function g are aligned with the textual prompt y. The form of SDS loss is as follows:

$$\nabla_\theta \mathcal{L}_{SDS} = \mathbf{E}_{t,\epsilon,\pi}[w(t)(\epsilon_\phi(z_t; y, t) - \epsilon)\frac{\partial x}{\partial \theta}] \tag{2}$$

where $w(t)$ is a weighting function, and z_t represents a noisy version of x. In practice, SDS employs Classifier-free guidance [14], therefore, the $\epsilon_\phi(z_t; y, t)$ in Eq. 2 is $\epsilon_\phi(z_t; y, t) + w(\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t))$, where w represents the guidance scale. Combine them, the final form of SDS is:

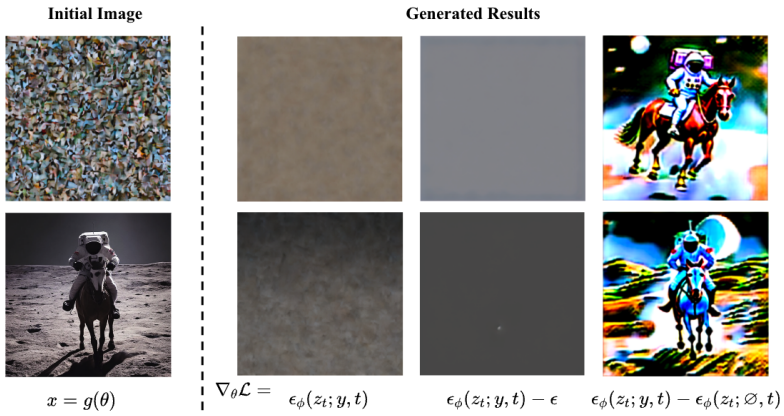


Fig. 2: We separately utilize each term as the loss function to generate images in 2D experiments using SDS, with the prompt "An astronaut riding a horse". In the first line, we use a randomly noise as initialization, in the second line, we use a high-quality image sampled from Stable Diffusion 2-1 base [43] as initialization. The results show that regardless of the initialization method, using each term alone cannot generate realistic and detailed samples.

$$\nabla_\theta \mathcal{L}_{SDS} = \mathbf{E}_{t,\epsilon,\pi}[w(t)(\epsilon_\phi(z_t; y, t) + w(\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)) - \epsilon)\frac{\partial x}{\partial \theta}] \quad (3)$$

In this form, SDS consists of two parts. The first part is the original $\epsilon_\phi(z_t; y, t) - \epsilon$, which is proportional to $\nabla_{z_t} \log p(z_t|y)$. Its role is to restore z_t back to its pre-noised state x according to the condition y. The second part is introduced by Classifier-free guidance [14], $\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$, which is proportional to $\nabla_{z_t} \log p(z_t|y) - \nabla_{z_t} \log p(z_t)$. According to Bayes' rule, this term can further be proportional to $\nabla_{z_t} \log p(y|z_t)$. It acts as an implicit noisy image classifier.

4 Methodology

4.1 Anatomy of SDS

Based on the analysis in Sec. 3.1, we seek to explore why SDS suffers from over-smoothing and over-saturation. To this end, we conduct an experiment to investigate the role of each term in the generation process of SDS. Specifically, following our previous analysis, we divide the SDS loss into two components: one is the classifier-free guidance term $\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$, and the other is the reconstruction term $\epsilon_\phi(z_t; y, t) - \epsilon$. We use a randomly generated noise and a high-quality image sampled from Stable Diffusion [43] as our initialization images, respectively. Then we use each term as the loss function for image generation.

As demonstrated in Fig. 2, we find that although using $\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$ alone can generate images aligned with the textual condition y, it suffers from

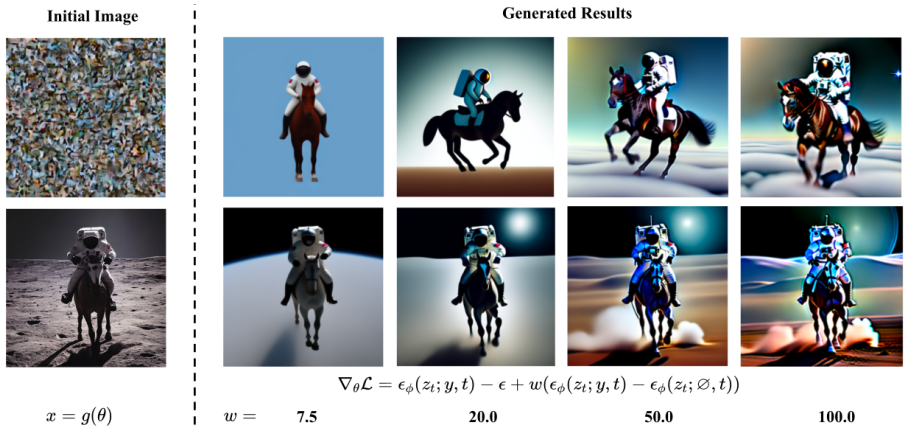


Fig. 3: In order to show the relationship between over-saturation and guidance scale w in SDS, we use different guidance scales for text-to-image generation. We find that as w increases, the over-saturation of the generated images gradually becomes serious, while as w decreases, the images gradually become over-smoothing.

serious over-saturation and distortion, this cannot be avoided even when using a high-quality image as initialization. To further investigate the relationship between over-saturation and the classifier-free guidance term, we conduct another experiment. As shown in Fig. 3, we find that as w increases, the images generated by SDS gradually become full of details (such as the texture of the horse, the details of the astronaut), but the over-saturation problem is also gradually amplified. Therefore, we speculate that the over-saturation phenomenon in SDS comes from the larger guidance scale. This phenomenon is also reasonable because Stable Diffusion [43] has not used classifier-free guidance term alone for generating during pre-training. Therefore, this item itself cannot generate real samples, and amplifying its weight have a negative impact on generated results.

Additionally, we discover that using $\epsilon_\phi(z_t; y, t) - \epsilon$ alone is incapable of generating images. Even in cases where a high-quality image is used for initialization, using $\epsilon_\phi(z_t; y, t) - \epsilon$ alone reverts it back into a semantically meaningless noise image. Based on this phenomenon, we further analyze $\epsilon_\phi(z_t; y, t) - \epsilon$. According to the DDIM sampling [47], we know diffusion models predict the x_0^t at time step t with the following formula:

$$x_0^t = \frac{z_t - \sqrt{1-\alpha_t}\epsilon_\phi(z_t; y, t)}{\sqrt{\alpha_t}} \tag{4}$$

In the case of SDS, we know that z_t is obtained by $z_t = \sqrt{\alpha_t}x + \sqrt{1-\alpha_t}\epsilon$. Therefore, the Eq. 4 can be written as:

$$x = x_0^t + \frac{\sqrt{1-\alpha_t}}{\sqrt{\alpha_t}}(\epsilon_\phi(z_t; y, t) - \epsilon) \tag{5}$$

In this form, we find that $\epsilon_\phi(z_t; y, t) - \epsilon$ is actually related to the x_0^t predicted at time step t, meaning its function is to reconstruct z_t back to its pre-noised state x. However, this presents two problems. One is the single-step reconstruction error caused by diffusion models [29, 45, 48], they can not reconstruct the image well when t is large. This is why using $\epsilon_\phi(z_t; y, t) - \epsilon$ alone causes an originally high-quality image to become noise (as shown in the third column of Fig. 2). Another is that when the original image x has noise, reconstructing it back to its original state still retains the noise (as shown in the third column in Fig. 5). This directly leads to the following problems in SDS: at the beginning (when x lacks semantic information), $\epsilon_\phi(z_t; y, t) - \epsilon$ inhibits $\epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$ from aligning x with the text condition, and in the later stage of SDS (when x already meets the text condition), $\epsilon_\phi(z_t; y, t) - \epsilon$ degrades x into noise. Therefore, these two issues ultimately lead to over-smoothing and loss of detail in SDS optimization.

Based on the above analysis, we know that SDS assigns a large guidance scale to the classifier-free guidance term to mitigate the over-smoothing and details lost problems brought by the reconstruction term. However, this exacerbates the over-saturation problem brought by the classifier-free guidance term.

4.2 Invariant Score Distillation

To overcome the aforementioned problems, *i.e.*, the over-smoothing and details lost problems brought by the reconstruction term and the over-saturation problem brought by the large classifier-free guidance scale. We introduce an invariant score term derived from the DDIM sampling process to replace the original reconstruction term. This term no longer reconstructs the noisy image to its original state in a single step but aligns the noisy image to the previous time step state, avoiding the error of single-step reconstruction.

According to the DDIM sampling formula, we can get the relationship between x_0^t predicted at the current time step and x_0^{t-c} predicted at the previous interval time step $t - c$:

$$x_0^{t-c} = x_0^t - \frac{\sqrt{1-\alpha_{t-c}}\epsilon_\phi(z_{t-c}; t-c) - \sqrt{1-\alpha_{t-c}-\sigma_t^2}\epsilon_\phi(z_t; t) + \sigma_t\epsilon}{\sqrt{\alpha_{t-c}}} \tag{6}$$

We extract the interval sampling term about the noise prediction and get $\delta_{inv} = \epsilon_\phi(z_{t-c}; y, t-c) - \sqrt{1-\sigma_t^2}\epsilon_\phi(z_t; y, t) - \sigma_t\epsilon$. Following the setting of DDIM, we set $\sigma_t = 0$, and derive our invariant score term as follows:

$$\delta_{inv} = \epsilon_\phi(z_{t-c}; y, t-c) - \epsilon_\phi(z_t; y, t) \tag{7}$$

where c is the sampling step interval. Introducing this item brings two benefits: Firstly, by continuously aligning x_0^t with the previous time step during the optimization process, the consistency of the final generated results is enhanced, thus overcoming the over-smoothing problems. Secondly, since this invariant score prior is related to $x_0^{t-c} - x_0^t$, and we know that x_0^{t-c} predicted at the previous time step always has more details than x_0^t at the current time step, so $x_0^{t-c} - x_0^t$

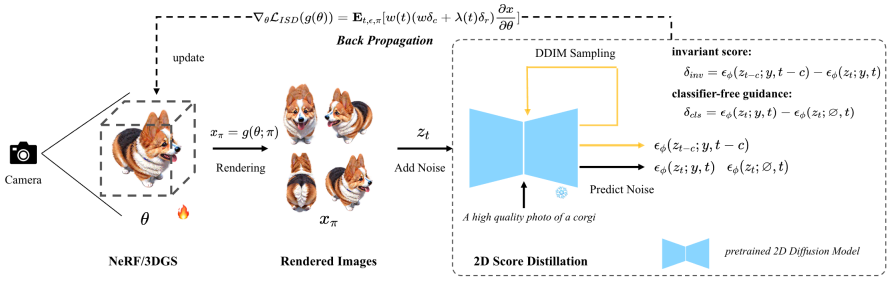


Fig. 4: Overview of ISD for text-to-3D generation. We aim to optimize a 3D model θ using a pretrained text-to-image diffusion model. To achieve this, we render a 2D rendered image from $g(\theta, \pi)$ at a random pose π, and then employ a diffusion model ϵ_ϕ to do the 2D Score Distillation. In particular, given a rendered image x_π, we first add noise to it to obtain z_t, and utilize the diffusion model to estimate the noise. In our framework, there are three noise predictions: $\epsilon_\phi(z_t, y, t)$, $\epsilon_\phi(z_t, \varnothing, t)$ and $\epsilon_\phi(z_{t-c}, y, t-c)$. We utilize them separately to compute the classifier-free guidance term δ_{cls} and the invariant score term δ_{inv}, and then utilize our proposed ISD for optimization.

remains increased detail and removed the original noise (as shown in the last column of Fig. 5), solving the original problem of details lost. Combing this term with the classifier-free guidance term $\delta_{cls} = \epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$, we obtain our final form of ISD, the gradient is expressed as follows:

$$\nabla_\theta \mathcal{L}_{ISD}(g(\theta)) = \mathbf{E}_{t,\epsilon}[w(t)(\lambda(t)\delta_{inv} + w\delta_{cls})\frac{\partial x}{\partial \theta}] \tag{8}$$

where $\lambda(t)$ and $w(t)$ is weighting function, we keep the value of $w(t)$ consistent with SDS. In addition to the advantages discussed above, we found that ISD also allows us to assign a normal weight (7,5) to the classifier terms, thus avoiding the problem of over-saturation. In Fig. 4, we present an overview of our framework. Our framework can optimize a Neural Radiance Field (NeRF) [34, 35] or 3D Gaussian Splatting [22] using 2D text-to-image diffusion models. In Fig. 1 and Fig. 6, we show our proposed ISD can generate high-fidelity and detailed 3D models in a single-stage optimization.

4.3 Connection with other methods

In this section, we discuss the connection of our proposed Invariant Score Distillation with several methods.

Compared with SDS. SDS [40] is proposed to distill 2D diffusion priors for text-to-3D generation. The objective function is given by Eq. 3. We find that it can be expressed as a special case of the residual term we introduced. Considering Eq. 6 with $\sigma_t = \sqrt{1-\alpha_{t-1}}$, we observe that in this case, the formula reduces to:

$$x_0^{t-1} = x_0^t + \frac{\sqrt{1-\alpha_{t-1}}}{\sqrt{\alpha_{t-1}}}(\epsilon - \epsilon_\phi(z_{t-1}; t-1)) \tag{9}$$

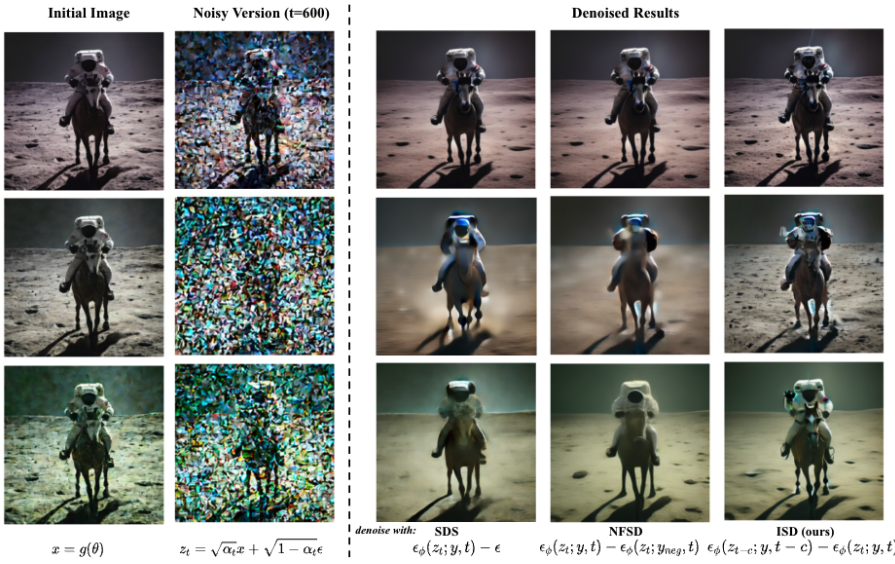


Fig. 5: We use different noisy versions of a clear image as our initial images and add noise to each image with $t = 600$. We use different residual terms for denoising. The results show that when we continuously add noise to the initial image, our method can always restore more details than baseline methods

In this situation, we find that SDS is actually using the residual term $\epsilon_\phi(z_{t-1}; t-1) - \epsilon$ to continuously optimize x. This is equivalent to using random noise $\epsilon \sim N(0, I)$ instead of $\epsilon_\phi(z_t; t)$ to obtain z_{t-1} after predicting x_0^t at each step. This approach can cause significant interference in generating results.

Compared with VSD. VSD [53] is proposed to alleviate the over-smoothing and over-saturated problems in SDS. In VSD, in addition to the original diffusion model $\epsilon_\phi(z_t; y, t)$, they also train an additional LoRA [16] ϵ_{lora}. During the optimization process, they initialize the LoRA with the weights of $\epsilon_\phi(z_t; y, t)$ and fine-tuned with the rendered images from the current scene states. They replace the noise term ϵ in Eq. 3 with ϵ_{lora}. The gradient is expressed as follows:

$$\nabla_\theta \mathcal{L}_{VSD} = \mathbf{E}_{t,\epsilon,\pi}[w(t)(\tilde{\epsilon}_\phi(z_t; y, t) - \epsilon_{lora}(z_t; y, t))\frac{\partial x}{\partial \theta}] \tag{10}$$

Actually, the residual term of VSD is $\epsilon_\phi(z_t; y, t) - \epsilon_{lora}(z_t; y, t)$. Since the LoRA model is initialized with the weights of $\epsilon_\phi(z_t; y, t)$ and uses random noise ϵ as the learning target. Therefore, $\epsilon_{lora}(z_t; y, t)$ can be written as a linear combination of $\epsilon_\phi(z_t; y, t)$ and ϵ. *i.e.*, $\epsilon_{lora}(z_t; y, t) = \sqrt{1-\sigma_t^2}\epsilon_\phi(z_t; y, t) + \sigma_t\epsilon$. This is still consistent with the general form of the residual term we provided in Eq. 6.

Compared with NFSD. NFSD [21] decomposes the loss function into three terms and omits the denoising term $\delta_N = \epsilon_\phi(z_t; y, t) - \epsilon$. The final form is:

$$\nabla_\theta \mathcal{L}_{NFSD} = \mathbf{E}_{t,\epsilon,\pi}[w(t)(\delta_D + w\delta_{cls})\frac{\partial x}{\partial \theta}] \tag{11}$$

where δ_{cls} has the same form as ours, and δ_D is a piecewise function. When $t > 200$, $\delta_D = \epsilon_\phi(z_t; \varnothing, t) - \epsilon_\phi(z_t; y_{neg}, t)$. We take an item $\delta_{cls} = \epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; \varnothing, t)$ and add it to δ_D, which turns the original w in Eq. 11 to $w - 1$. At this time, δ_D becomes $\delta_D = \epsilon_\phi(z_t; y, t) - \epsilon_\phi(z_t; y_{neg}, t)$. Since y_{neg} describes the disordered and chaotic direction. Therefore we can regard $\epsilon_\phi(z_t; y_{neg}, t)$ as the noisy version of $\epsilon_\phi(z_t; y, t)$, *i.e.*, $\epsilon_\phi(z_t; y_{neg}, t) \approx \epsilon_\phi(z_{t+c}; y, t + c)$.

To further verify the advantages of our method, we conduct another experiment. We use different noisy versions of an image to simulate x in SDS. Then we imitate the process of SDS, add noise to these images, and restore them with different residual terms. As shown in Fig 5, we verified that the image after SDS denoising has the problem of losing details and retaining noise, and the ISD we proposed can not only supplement the details, but also remove the noise originally covered by the image.

5 Experiments

5.1 Implementation Details

We implement Invariant Score Distillation (ISD) using the threestudio [11] framework. For all results, we set the diffusion step interval c as 20 and $\lambda(t)$ as $\sqrt{(\beta_{t-c}/\alpha_{t-c})/(\beta_t/\alpha_t)}$. In addition, we sample time steps $t \sim \mathcal{U}(0.02, 0.98)$ at first 5000 steps, and after then, we follow the annealed time schedule proposed by ProlificDreamer [53] and sample time steps $t \sim \mathcal{U}(0.02, 0.50)$. For text-to-NeRF, we use Stable Diffusion 2-1-base [43] as our pretrained text-to-image diffusion model with guidance scale $w = 7.5$. We set the rendering resolution as 512x512 and optimized all 3D models for 25000 iterations using AdamW [31] optimizer with a learning rate of 0.01. We set the camera distance range as $[1.5, 2.0]$ and fovy range as $[40, 70]$, we found that these two parameters are highly related to the "Janus" problem. When set too low, it leads to a severe "Janus" problem. For text-to-3DGS, we set the rendering resolution as 512x512 and optimized all 3D models for 5000 iterations. We set the camera distance range as $[3.5, 5.0]$. For other training hyperparameters, we follow the original 3DGS [22] paper. For the head generation, we add a description of hair in the prompt to avoid multi-face issues and utilize Point-E [37] as the 3D prior model with the prompt "a man head". We conduct all experiments on a single A6000 GPU.

5.2 Text-to-3D Generation

We present some representative results using ISD in Fig. 1 and 6, including both text-to-3DGS and text-to-NeRF (bottom two lines in Fig. 6). In general, ISD can generate 3D assets aligned well with textual prompts and possess a realistic and detailed appearance. For text-to-3DGS, we show that ISD can generate highly intricate details. For text-to-NeRF, we also show that ISD can effectively generate long text descriptions of content.



Fig. 6: Examples generated by ISD. We provide examples of text-to-3DGS (the top two rows) and text-to-NeRF (the bottom two rows). ISD is capable of generating meticulously detailed and realistic textures, and it is also effective for long text descriptions.

Qualitative Comparisons. In Fig. 7, we compare our generation results with Dreamfusion [40], Maigc3D [26], ProlificDreamer [53] and NFSD [21]. For DreamFusion, ProlificDreamer, and Magic3D, we take the results provided in ProlificDreamer. For NFSD, we directly use the results provided in their paper. As can be seen, our method achieves comparable or better results, while being significantly simpler than most of these methods. For example, observing the roof of the cottage, the results generated by our method have better detail and realism than other methods and are comparable to the result of ProlificDreamer. However, our method requires only minor changes and only single-stage optimization, compared to ProlificDreamer, we have great advantages in efficiency. Moreover, for a fair comparison, we apply ISD and SDS [40] to 3DGS [22] technology. As can be seen in Fig. 8, although the SDS results are much improved compared to the original version, our method can generate more realistic and detailed 3D objects. For example, in the bagel example, ISD produced a more detailed salmon texture.



Fig. 7: Qualitative comparisons to baselines for text-to-3D generation. Our method can generate 3D objects that align well with input text prompts with realistic and detailed appearances.



Fig. 8: Qualitative comparisons to baselines for text-to-3D generation. We applied SDS and ISD to 3DGS for comparison. Although the quality generated by SDS has been greatly improved, ISD can still produce more detailed and realistic objects.

Quantitative Evaluations. Following previous works [18,33,40,59], we quantitatively evaluate the quality of generated 3D assets utilizing CLIP Score [12,42]

Table 1: User study results. We provide two attributes to measure the quality of 3D generation. Users choose the one they like most among the results we provide, the results show more users prefer our results.

Methods	Prompt Alignment (%) ↑	Image Quality (%) ↑
DreamFusion [40]	10.74	-
Magic3D [26]	10.44	4.97
ProlificDreamer [53]	20.37	22.37
NFSD [21]	17.53	10.82
Ours	**40.92**	**61.84**

Table 2: Quantitative comparisons to baselines for text-to-3D generation, evaluated by CLIP Score and CLIP R-Precision.

Methods	CLIP Score (↑)	CLIP R-Precision (%) ↑ R@1	R@5	R@10
DreamFusion [40]	67.22	75.79	91.70	97.42
Magic3D [26]	75.73	77.31	92.74	96.93
ProlificDreamer [53]	78.90	78.64	93.44	97.42
NFSD [21]	75.27	76.93	92.74	97.27
Ours	**81.06**	**82.30**	**95.08**	**97.93**

and CLIP R-Precision [39]. We generate 3D objects using 50 distinct textual prompts. For each object, images are rendered from four different views (front, back, left, right). The resulting CLIP Score is calculated as the average semantic similarity between the textual prompt and the images from these four angles. We use the CLIP ViT-B/32 [42] model to extract text and image features, and the results are shown in Tab. 2. Our approach significantly outperforms DreamFusion, Magic3D, ProlificDreamer, and NFSD in terms of CLIP Score and CLIP R-Precision, indicating better alignment between the generated results and input text prompts.

User Study. We follow previous works [21, 26, 53, 59] and conducted a user study comparing rendered images obtained by Dreamfusion [40], Magic3D [26], ProlificDreamer [53], NFSD [21] and our ISD method. We enlist 47 participants and ask them to choose the image that is most aligned with the given prompt, and the highest quality image. The result is presented in Tab. 1, all methods provide results that are aligned with the prompt and our method attains the highest score. In terms of the rendered image quality, our method obtains significantly better scores compared with the competing methods.

5.3 Ablation Study

Ablation on time step interval c. In the first line of Fig. 9, we show the results generated using different c. We find that when c is small (*e.g.*, 1), the generated results lack details and suffer from over-saturation. And when c is large (*e.g.*, 50), the results have rich details, but there are problems such as blur artifacts. The reason here is that when c gradually decreases, ISD gradually degenerates into only using the classifier-free guidance term to optimize the 3D model, and when c is large, the increased variance will destroy the quality of the generated results and cause local over-smoothing. Therefore, we ultimately choose $c = 20$ to avoid the problems of over-saturation and over-smoothing. The results show that when $c = 20$, ISD can produce highly detailed and high-fidelity results.

Ablation on weighting function $\lambda(t)$. In the second line of Fig. 9, we show the results generated using different $\lambda(t)$. We set $\lambda(t)$ according to SDS [40],

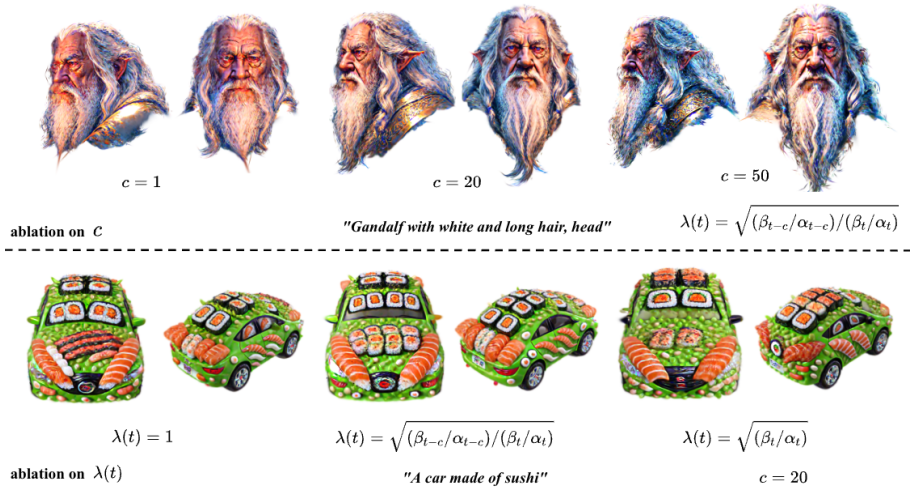


Fig. 9: ISD with different c (the first line) and $\lambda(t)$ (the second line). The results show that c affects the details and quality of the generated results, and $\lambda(t)$ affects the richness of detail and alignment with the prompt.

Magic3D [26], and the derivation of Eq. 15. This coefficient determines the size of parameter weights at different time steps. The difference in parameter weights also has a certain impact on the generated results. As shown in Fig. 9, we find that when $\lambda(t) = 1$ and $\lambda(t) = \sqrt{(\beta_{t-c}/\alpha_{t-c})/(\beta_t/\alpha_t)}$, where $\beta_t = 1 - \alpha_t$, the sushi car is full of details, but we find that when $\lambda(t) = \sqrt{(\beta_{t-c}/\alpha_{t-c})/(\beta_t/\alpha_t)}$, the generated object is more consistent with the text description.

6 Conclusion

In this paper, we present an analysis of the over-saturation and over-smoothing problems in Score Distillation Sampling. We find that over-saturation comes from the larger classifier-free guidance scale, while over-smoothing comes from the noise preservation of the reconstruction term and the single-step reconstruction error. To overcome these problems, we propose Invariant Score Distillation (ISD), a novel framework for text-to-3D generation. ISD replaces the reconstruction term with an invariant score term derived from DDIM sampling, and improves the problem of the reconstruction term in two aspects: 1) invariant score term no longer reconstructs the noisy but aligns the noisy image to its previous time step state, avoiding the single-step reconstruction error. 2) the invariant score term can remove the original noise in the image and add details, avoiding the details lost problem. Moreover, this improvement also allows us to use a normal classifier-free guidance scale, which avoids over-saturation problems. Extensive experiments demonstrate that our method greatly enhances SDS and produces realistic 3D objects through single-state optimization.

Acknowledgments

This work is supported by the Major program of the National Natural Science Foundation of China (T2293720/T2293723), the National Natural Science Foundation of China (U2336212), the National Natural Science Foundation of China (62293554), the Fundamental Research Funds for the Central Universities (No. 226-2024-00058), the Fundamental Research Funds for the Zhejiang Provincial Universities (226-2024-00208), the "Leading Goose" R&D Program of Zhejiang Province under Grant 2024C01101, and the China Postdoctoral Science Foundation (524000-X92302).

References

1. 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)
2. Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: ShapeNet: An Information-Rich 3D Model Repository. Tech. Rep. arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago (2015)
3. Chen, G., Wang, W.: A survey on 3d gaussian splatting. arXiv preprint arXiv:2401.03890 (2024)
4. 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)
5. Chen, Z., Wang, F., Liu, H.: Text-to-3d using gaussian splatting (2023)
6. 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. Advances in Neural Information Processing Systems **36** (2024)
7. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13142–13153 (2023)
8. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. Advances in neural information processing systems **34**, 8780–8794 (2021)
9. Gao, J., Fu, Y., Wang, Y., Qian, X., Feng, J., Fu, Y.: Mind-3d: Reconstruct high-quality 3d objects in human brain (2024), `https://arxiv.org/abs/2312.07485`
10. Gao, J., Qian, X., Wang, Y., Xiao, T., He, T., Zhang, Z., Fu, Y.: Coarse-to-fine amodal segmentation with shape prior (2023), `https://arxiv.org/abs/2308.16825`
11. Guo, Y.C., Liu, Y.T., Shao, R., Laforte, C., Voleti, V., Luo, G., Chen, C.H., Zou, Z.X., Wang, C., Cao, Y.P., Zhang, S.H.: threestudio: A unified framework for 3d content generation. `https://github.com/threestudio-project/threestudio` (2023)
12. Hessel, J., Holtzman, A., Forbes, M., Bras, R.L., Choi, Y.: Clipscore: A reference-free evaluation metric for image captioning (2022)
13. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)

14. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
15. 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)
16. 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 (2021)
17. 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)
18. Jain, A., Mildenhall, B., Barron, J.T., Abbeel, P., Poole, B.: Zero-shot text-guided object generation with dream fields (2022)
19. Jun, H., Nichol, A.: Shap-e: Generating conditional 3d implicit functions. arXiv preprint arXiv:2305.02463 (2023)
20. Karras, T., Aittala, M., Aila, T., Laine, S.: Elucidating the design space of diffusion-based generative models. Advances in Neural Information Processing Systems **35**, 26565–26577 (2022)
21. Katzir, O., Patashnik, O., Cohen-Or, D., Lischinski, D.: Noise-free score distillation. arXiv preprint arXiv:2310.17590 (2023)
22. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4) (July 2023), `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`
23. Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., Bi, S.: Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model (2023)
24. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation (2022)
25. Liang, Y., Yang, X., Lin, J., Li, H., Xu, X., Chen, Y.: Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching (2023)
26. 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)
27. Liu, M., Xu, C., Jin, H., Chen, L., Varma T, M., Xu, Z., Su, H.: One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. Advances in Neural Information Processing Systems **36** (2024)
28. Liu, R., Wu, R., Hoorick, B.V., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object (2023)
29. Liu, X., Zhang, X., Ma, J., Peng, J., 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)
30. Liu, Y., Lin, C., Zeng, Z., Long, X., Liu, L., Komura, T., Wang, W.: Syncdreamer: Generating multiview-consistent images from a single-view image. arXiv preprint arXiv:2309.03453 (2023)
31. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization (2019)
32. Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., Zhu, J.: 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)
33. Luo, T., Rockwell, C., Lee, H., Johnson, J.: Scalable 3d captioning with pretrained models (2023)

34. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis (2020)
35. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Trans. Graph. **41**(4), 102:1–102:15 (Jul 2022). https://doi.org/10.1145/3528223.3530127, https://doi.org/10.1145/3528223.3530127
36. Nichol, A., Dhariwal, P., Ramesh, A., Shyam, P., Mishkin, P., McGrew, B., Sutskever, I., Chen, M.: Glide: Towards photorealistic image generation and editing with text-guided diffusion models. arXiv preprint arXiv:2112.10741 (2021)
37. Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., Chen, M.: Point-e: A system for generating 3d point clouds from complex prompts. arXiv preprint arXiv:2212.08751 (2022)
38. van den Oord, A., Li, Y., Babuschkin, I., Simonyan, K., Vinyals, O., Kavukcuoglu, K., van den Driessche, G., Lockhart, E., Cobo, L.C., Stimberg, F., Casagrande, N., Grewe, D., Noury, S., Dieleman, S., Elsen, E., Kalchbrenner, N., Zen, H., Graves, A., King, H., Walters, T., Belov, D., Hassabis, D.: Parallel wavenet: Fast high-fidelity speech synthesis (2017)
39. Park, D.H., Azadi, S., Liu, X., Darrell, T., Rohrbach, A.: Benchmark for compositional text-to-image synthesis. In: Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1) (2021)
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., Mai, J., Hamdi, A., Ren, J., Siarohin, A., Li, B., Lee, H.Y., Skorokhodov, I., Wonka, P., Tulyakov, S., et al.: Magic123: One image to high-quality 3d object generation using both 2d and 3d diffusion priors. arXiv preprint arXiv:2306.17843 (2023)
42. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021)
43. 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)
44. Saharia, C., Chan, W., Saxena, S., Li, L., Whang, J., Denton, E.L., Ghasemipour, K., Gontijo Lopes, R., Karagol Ayan, B., Salimans, T., et al.: Photorealistic text-to-image diffusion models with deep language understanding. Advances in Neural Information Processing Systems **35**, 36479–36494 (2022)
45. Salimans, T., Ho, J.: Progressive distillation for fast sampling of diffusion models. arXiv preprint arXiv:2202.00512 (2022)
46. Sanghi, A., Chu, H., Lambourne, J.G., Wang, Y., Cheng, C.Y., Fumero, M., Malekshan, K.R.: Clip-forge: Towards zero-shot text-to-shape generation (2022)
47. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
48. Song, Y., Dhariwal, P., Chen, M., Sutskever, I.: Consistency models (2023)
49. 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)
50. 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)
51. 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)
52. Wang, X., Wang, Y., Ye, J., Wang, Z., Sun, F., Liu, P., Wang, L., Sun, K., Wang, X., He, B.: Animatabledreamer: Text-guided non-rigid 3d model generation and reconstruction with canonical score distillation. arXiv preprint arXiv:2312.03795 (2023)
53. 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. Advances in Neural Information Processing Systems **36** (2024)
54. Wu, Z., Zhou, P., Yi, X., Yuan, X., Zhang, H.: Consistent3d: Towards consistent high-fidelity text-to-3d generation with deterministic sampling prior (2024)
55. Xu, Y., Yang, Z., Yang, Y.: Seeavatar: Photorealistic text-to-3d avatar generation with constrained geometry and appearance. arXiv preprint arXiv:2312.08889 (2023)
56. Yang, Z., Chen, G., Li, X., Wang, W., Yang, Y.: Doraemongpt: Toward understanding dynamic scenes with large language models (exemplified as a video agent) (2024)
57. Ye, J., Liu, F., Li, Q., Wang, Z., Wang, Y., Wang, X., Duan, Y., Zhu, J.: Dreamreward: Text-to-3d generation with human preference. arXiv preprint arXiv:2403.14613 (2024)
58. 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)
59. 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)
60. Zeng, X., Vahdat, A., Williams, F., Gojcic, Z., Litany, O., Fidler, S., Kreis, K.: Lion: Latent point diffusion models for 3d shape generation (2022)
61. Zhou, D., Li, Y., Ma, F., Zhang, X., Yang, Y.: Migc: Multi-instance generation controller for text-to-image synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 6818–6828 (2024)
62. Zhou, Z., Ma, F., Fan, H., Yang, Y.: Headstudio: Text to animatable head avatars with 3d gaussian splatting. arXiv preprint arXiv:2402.06149 (2024)
63. Zhu, J., Zhuang, P.: Hifa: High-fidelity text-to-3d with advanced diffusion guidance. arXiv preprint arXiv:2305.18766 (2023)
64. Zhuo, W., Sun, Y., Wang, X., Zhu, L., Yang, Y.: Whitenedcse: Whitening-based contrastive learning of sentence embeddings. In: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 12135–12148 (2023)