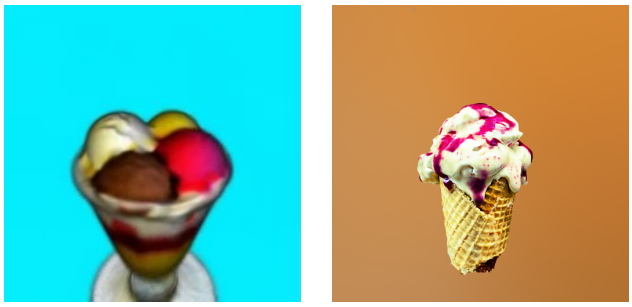
Adversarial Score Distillation: When score distillation meets GAN -Supplemental Material-

A. Quantitative Evaluations and User Studies for 3D Generation

We quantitatively compare our ASD with DreamFusion [9] (SDS) and ProlificDreamer [11] (VSD) using CLIP score, following previous works [7, 9]. We evaluate the similarities between text prompts and randomly rendered views of the 3D NeRFs using three variants of CLIP [10] (*i.e.*, CLIP B/32, CLIP B/16, and CLIP B/14). The CLIP score may not honestly reflect the quality of 3D results. For example, the CLIP score of Figure 1(a) is higher than Figure 1(b) even though its quality is significantly worse, which does not align with human perceptual judgments. Several existing methods [2, 11] choose to only conduct user studies for performance evaluation.

Method	CLIP B/32↑	CLIP B/16↑	CLIP L/14↑
DreamFusion (SDS)	0.3282	0.3290	0.2889
ProlificDreamer (VSD)	0.3203	0.3258	0.2789
Ours (ASD)	0.3314	0.3376	0.2854

Table 1. CLIP Score by computing the similarity between multiple randomly rendered views of the 3D model and the given text prompt.



(a) CLIP score: 0.3401 (b) CLIP score: 0.3069

Figure 1. Illustration of CLIP scores that conflict with human perceptual judgments. We use CLIP B/16 to compute scores here.

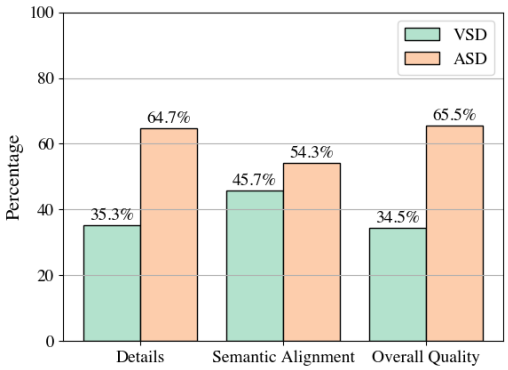


Figure 2. User evaluation between VSD and ASD.

Therefore, we also present user studies in Figure 2. We compare 3D NeRFs generated by VSD and ASD with different prompts in terms of details, semantic alignment, and overall quality. Figure 2 demonstrates that ASD leads to better details, which is consistent with our observation that VSD's results are slightly more saturated and therefore have worse details. The semantic alignment scores of VSD and ASD are similar. In general, more users prefer the results generated by ASD.

B. Derivation of Adversarial Score Distillation Loss and Gradients

We define a discriminator in the form as

$$D(x_t; y) = \log \frac{p(y|x_t)}{p(\phi|x_t)} \tag{1}$$

Based on WGAN [1, 12], we have the generator loss as

$$\mathcal{L}_G = \mathbb{E}_{t,\epsilon}[-D(x_t^g; y)] = \mathbb{E}_{t,\epsilon}[-\log \frac{p(y|x_t)}{p(\phi|x_t)}] \leq \mathbb{E}_{t,\epsilon}[-\log \frac{p(y|x_t)^\lambda}{p(\phi|x_t)}] := \mathcal{L}'_G \tag{2}$$

where $x_t^g = \alpha_t g(\theta, c) + \sigma_t \epsilon$ denotes the generated sample with noise. According to the Bayesian rule, we can do the transformation $\nabla_{x_t} \log p(y|x_t) = -1/\sigma_t(\epsilon_{x_t;y,t} - \epsilon_{x_t,t})$ similar to CFG [4]. Thus, we have

$$\begin{aligned}
\nabla_\theta \mathcal{L}'_G &= \nabla_\theta \mathbb{E}_{t,\epsilon}[\log p(\phi|x_t^g) - \lambda \log p(y|x_t^g)] \\
&= \mathbb{E}_{t,\epsilon}[\omega(t)(-\Big(\epsilon_{x_t^g,\phi,t} - \epsilon_{x_t^g,t}\Big) + \lambda\Big(\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,t}\Big))\frac{\partial x_0^g}{\partial \theta}] \\
&= \mathbb{E}_{t,\epsilon}[\omega(t)(\epsilon_{x_t^g,t} + \lambda(\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,t}) - \epsilon_{x_t^g,\phi,t})\frac{\partial x_0^g}{\partial \theta}]
\end{aligned} \tag{3}$$

where $\omega(t)$ is a weight based on the time t.

Then we have the discriminator loss as

$$\begin{aligned}
\mathcal{L}_D &= \mathbb{E}_{t,\epsilon}[D(x_t^g; y) - D(x_t^r; y) + \tau \|\nabla_{\tilde{x}_t} D(\tilde{x}_t; y)\|_2^2] \\
&= \mathbb{E}_{t,\epsilon}[\log \frac{p(y|x_t^g)}{p(\phi|x_t^g)} - \log \frac{p(y|x_t^r)}{p(\phi|x_t^r)} + \tau \|\nabla_{\tilde{x}_t} \log \frac{p(y|\tilde{x}_t)}{p(\phi|\tilde{x}_t)}\|_2^2]
\end{aligned} \tag{4}$$

where $x_t^r = \alpha_t x_0^r + \sigma_t \epsilon$ is the noisy real sample based on the real sample x_0^r and the time t. In Eq. (4), $p(y|x_t^r)$ and $p(y|x_t^g)$ are based on the given pretrained diffusion model, so they are not optimizable. ϕ in $p(\phi|x_t^g)$ and $p(\phi|x_t^r)$ is optimizable, which can be implemented using an textual-inversion embedding [3] or LoRA [6]. Thus, $\mathcal{L}_D$ is a loss function only about parameters ϕ, its gradient can be derived as

$$\begin{aligned}
&\nabla_\phi \mathcal{L}_D = \nabla_\phi \mathbb{E}_{t,\epsilon}[\log p(\phi|x_t^r) - \log p(\phi|x_t^g) + \tau \|\nabla_{\tilde{x}_t} \log \frac{p(y|\tilde{x}_t)}{p(\phi|\tilde{x}_t)}\|_2^2] \\
&= \nabla_\phi \mathbb{E}_{t,\epsilon}[\Big(\log p(x_t^r|\phi) + \log p(\phi) - \log p(x_t^r)\Big) - \Big(\log p(x_t^g|\phi) + \log p(\phi) - \log p(x_t^g)\Big) + \tau \|\nabla_{\tilde{x}_t} \log \frac{p(y|\tilde{x}_t)}{p(\phi|\tilde{x}_t)}\|_2^2] \\
&= \nabla_\phi \mathbb{E}_{t,\epsilon}[\log \frac{p(x_t^r|\phi)}{p(x_t^g|\phi)} + \tau \|\nabla_{\tilde{x}_t}(\Big(\log p(\tilde{x}_t|y) + \log p(y) - \log p(\tilde{x}_t)\Big) - \Big(\log p(\tilde{x}_t|\phi) + \log p(\phi) - \log p(\tilde{x}_t)\Big))\|_2^2] \\
&= \nabla_\phi \mathbb{E}_{t,\epsilon}[\log p(x_t^r|\phi) - \log p(x_t^g|\phi) + \tau \|\nabla_{\tilde{x}_t} \log p(\tilde{x}_t|y) - \nabla_{\tilde{x}_t} \log p(\tilde{x}_t|\phi)\|_2^2] \\
&\approx \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 - \|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \eta \|\epsilon_{x_t^r,y,t} - \epsilon_{x_t^r,\phi,t}\|_2^2 + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2]
\end{aligned} \tag{5}$$

where we use $-\log p(x_0) \approx \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t,\phi,t} - \epsilon\|_2^2]$, according to [5, 8]. $\tilde{x}$ comes from a mixture distribution $\tilde{\mu}$, so we can sampling both real or fake data points [12] which means $\tilde{x}_t$ can be replaced by a combination of x_t^r and x_t^g. η and γ are adjustable hyperparameters, to keep the penalty term positive, they are subject to $\gamma \geq -\eta \|\epsilon_{x_t^r;y,t} - \epsilon_{x_t^r;\phi,t}\|_2^2 / \|\epsilon_{x_t^g;y,t} - \epsilon_{x_t^g;\phi,t}\|_2^2$. However, in Eq. (5), the noisy real sample x_t^r is unavailable for text-conditioned cases. For these cases, we can use the upper bound of $\mathcal{L}_D$, (*i.e.* $\mathcal{L}'_D$), which does not contain x_t^r terms. When $\eta = 1/2$, based on triangle and Cauchy–Schwarz inequality, we have:

$$\begin{aligned}
&\nabla_\phi \mathcal{L}'_D = \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2] \\
&= \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 - \|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + (\|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \|\epsilon_{x_t^r,y,t} - \epsilon\|_2^2) + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2] \\
&= \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 - \|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \frac{1}{2}(1^2 + 1^2)(\|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \|\epsilon_{x_t^r,y,t} - \epsilon\|_2^2) + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2] \\
&\geq \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 - \|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \frac{1}{2}(\|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2 + \|\epsilon_{x_t^r,y,t} - \epsilon\|_2)^2 + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2] \\
&\geq \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 - \|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \frac{1}{2}(\|\epsilon_{x_t^r,\phi,t} - \epsilon + \epsilon - \epsilon_{x_t^r,y,t}\|_2)^2 + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2] \\
&= \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g,\phi,t} - \epsilon\|_2^2 - \|\epsilon_{x_t^r,\phi,t} - \epsilon\|_2^2 + \frac{1}{2}\|\epsilon_{x_t^r,y,t} - \epsilon_{x_t^r,\phi,t}\|_2^2 + \gamma \|\epsilon_{x_t^g,y,t} - \epsilon_{x_t^g,\phi,t}\|_2^2] = \nabla_\phi \mathcal{L}_D
\end{aligned} \tag{6}$$

In practice, we find $\eta = 1/2$, $\gamma \in [-1, 0)$ works for most cases.

C. Explanation of the workflow

To get a clearer picture of the optimization workflow, we further provide an algorithm description in Algorithm 1.

Algorithm 1 Adversarial Score Distillation for 3D tasks

Input: Prompt y, optimizable textual-inversion embedding or LoRA ϕ, and pretrained diffusion model.

repeat
  Sample a camera pose c.
  Differentiable render the 3D structure θ at pose c to get a 2D image $x_0 = g(\theta, c)$.
  Random select a timestep t and add random noise ϵ to get $x_t^g = \alpha_t x_0 + \sigma_t \epsilon$.
  Predict noise $\epsilon_{x_t^g;\phi,t}$, $\epsilon_{x_t^g;y,t}$ and $\epsilon_{x_t^g;t}$ using the pretrained diffusion model.
  (Update G) Update θ with the gradient $\nabla_\theta \mathcal{L}'_G : \mathbb{E}_{t,\epsilon}[\omega(t)(\epsilon_{x_t^g;t} + \lambda(\epsilon_{x_t^g;y,t} - \epsilon_{x_t^g;t}) - \epsilon_{x_t^g;\phi,t})\frac{\partial x_0^g}{\partial \theta}]$
  (Update D) Update ϕ with the gradient $\nabla_\phi \mathcal{L}'_D : \nabla_\phi \mathbb{E}_{t,\epsilon}[\|\epsilon_{x_t^g;\phi,t} - \epsilon\|_2^2 + \gamma\|\epsilon_{x_t^g;y,t} - \epsilon_{x_t^g;\phi,t}\|_2^2]$
until converged

Output: 3D structure θ.

D. Multi-view visualization

We exhibit some examples in Figure 3. The code and project page will be released which will contain rotating videos of the generated 3D models.

E. Additional Experimental Results

More examples in both 2D distillation and text-to-3D tasks are shown in Figure 4. In the text-to-3D task, we only generate 3D NeRFs from scratch for simplicity, which is the first stage of VSD.

References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International conference on machine learning*, pages 214–223. PMLR, 2017. 1

[2] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22246–22256, 2023. 1

[3] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *International Conference on Learning Representations*, 2022. 2

[4] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 2

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 2

[6] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021. 2

[7] Ajay Jain, Ben Mildenhall, Jonathan T Barron, Pieter Abbeel, and Ben Poole. Zero-shot text-guided object generation with dream fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 867–876, 2022. 1

[8] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *International Conference on Learning Representations*, 2014. 2

[9] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. In *International Conference on Learning Representations*, 2022. 1

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1

[11] Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Advances in Neural Information Processing Systems*, 36, 2024. 1

[12] Jiqing Wu, Zhiwu Huang, Janine Thoma, Dinesh Acharya, and Luc Van Gool. Wasserstein divergence for gans. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 1, 2

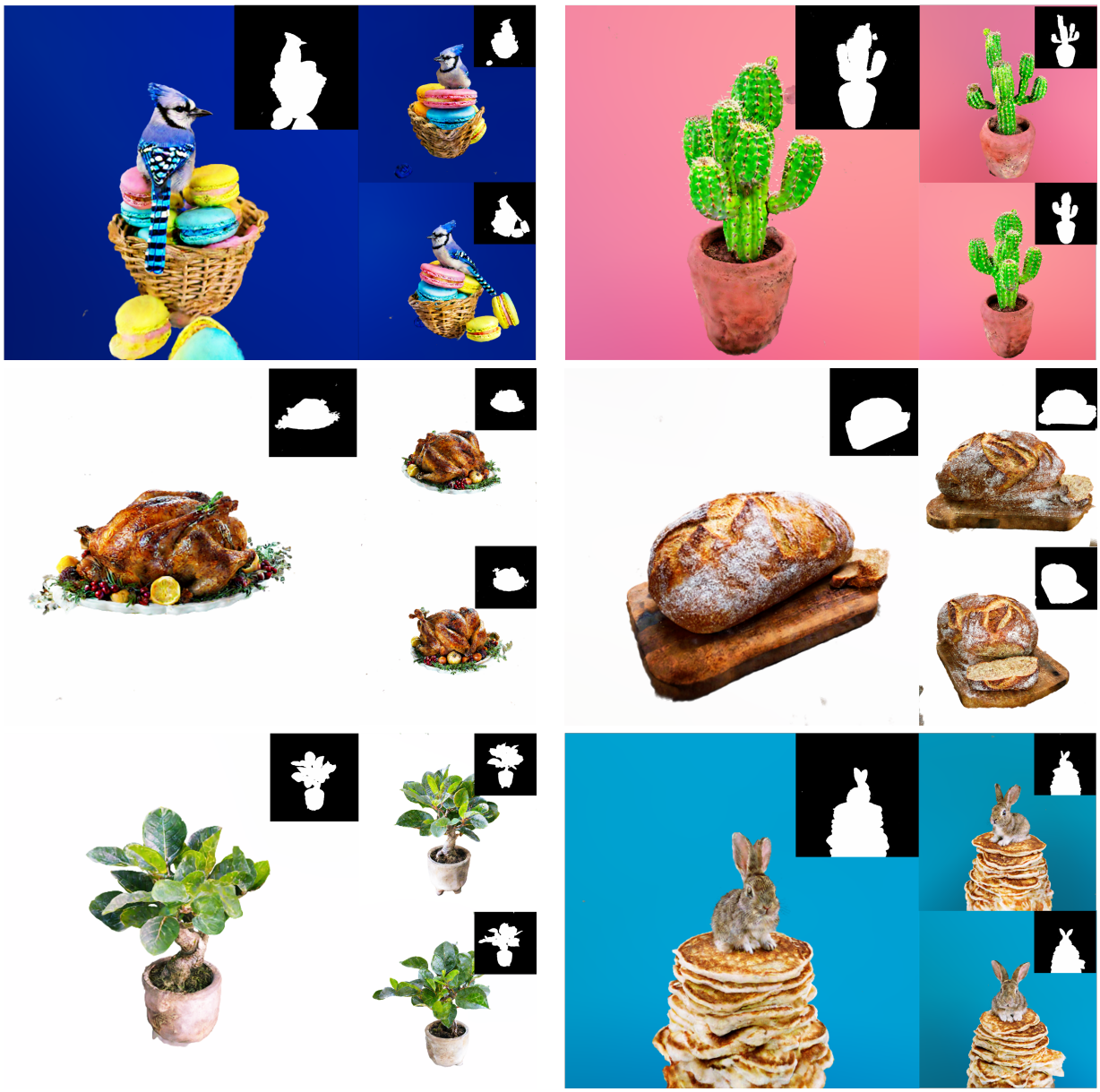
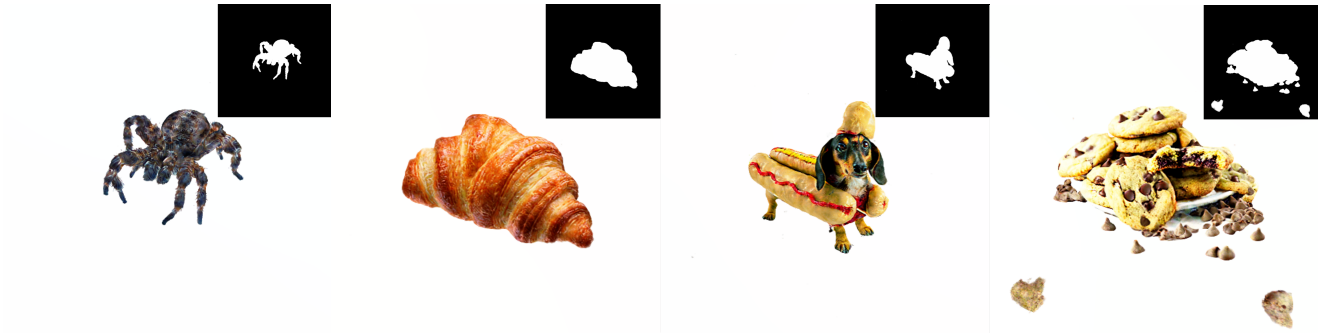


Figure 3. Multi-view visualizations of generated 3D results.

a castle in the middle of a marsh | *a boy portrait with sunglasses* | *a cozy living room with a painting of a corgi* [...] | *a group of elephants walking in muddy wate*

a photograph of a hamster | *a red fire hydrant spraying water* | *a small kitchen with a low ceiling* | *cliffs at day time*



a tarantula, highly detailed | *a delicious croissant* | *a dachsund dressed up* [...] | [...] *chocolate chip cookies*

a baby bunny sitting on [...] | *a blue jay standing on* [...] | [...] *a goose made out of gold* | *a plush dragon toy*

Figure 4. More examples generated by ASD in both 2D and 3D.