SegmentDreamer: Towards High-fidelity Text-to-3D Synthesis with Segmented Consistency Trajectory Distillation

Supplementary Material

To provide proofs of SegmentDreamer and visual comparisons of state-of-the-arts, this supplementary material includes the following contents:

- Section A: Guided Consistency Sampling Loss
- Section B: Flaws in the Consistency Function of Guided Consistency Sampling
- Section C: Oversaturation and Artifacts in GCS
- Section D: How to Connect SCTD with SDS
- Section E: Computation Reduction Trick
- Section F: Proof of Upper Bound of Distillation Error
- Section G: Visual Comparisons of State-of-the-arts

A. Guided Consistency Sampling Loss

The Guided Consistency Sampling (GCS) loss [2] is composed of a compact consistency loss $\mathcal{L}_{\text{CC}}$, a conditional guidance loss $\mathcal{L}_{\text{CG}}$, and a pixel-wise constraint loss $\mathcal{L}_{\text{CP}}$, which are defined by

$$\begin{aligned}
&\mathcal{L}_{\text{CC}}(\boldsymbol{\theta}) = \mathbb{E}[\big\|\boldsymbol{G}_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}_t^{\boldsymbol{\Phi}}, t, e, \emptyset) - \boldsymbol{G}_{\boldsymbol{\theta}}(\hat{\mathbf{z}}_s^{\boldsymbol{\Phi}}, s, e, \emptyset)\big\|_2^2],\\
&\mathcal{L}_{\text{CG}}(\boldsymbol{\theta}) = \mathbb{E}[\big\|\boldsymbol{F}_{\boldsymbol{\theta}}(\mathbf{z}_e, e, \emptyset) - \boldsymbol{F}_{\boldsymbol{\theta}}(\boldsymbol{G}_{\theta}(\tilde{\mathbf{z}}_t^{\boldsymbol{\Phi}}, t, e, \mathbf{y}), e, \emptyset)\big\|_2^2],\\
&\mathcal{L}_{\text{CP}}(\boldsymbol{\theta}) = \mathbb{E}[\big\|\boldsymbol{D}(F_{\theta}(\mathbf{z}_e, e, \emptyset)) - \boldsymbol{D}(\boldsymbol{F}_{\boldsymbol{\theta}}(\boldsymbol{G}_{\theta}(\tilde{\mathbf{z}}_t^{\boldsymbol{\Phi}}, t, e, \mathbf{y}), e, \mathbf{y}))\big\|_2^2],
\end{aligned} \tag{1}$$

where $e < s < t$, $\tilde{\mathbf{z}}_t^{\boldsymbol{\Phi}}$ is estimated by the following trajectory: $\mathbf{z}_e = \alpha_t \mathbf{z}_0 + \sigma_t \boldsymbol{\epsilon}^* \rightarrow \tilde{\mathbf{z}}_s^{\boldsymbol{\Phi}} = \boldsymbol{\Phi}(\mathbf{z}_e, e, s, \emptyset) \rightarrow \tilde{\mathbf{z}}_t^{\boldsymbol{\Phi}} = \boldsymbol{\Phi}(\tilde{\mathbf{z}}_s^{\boldsymbol{\Phi}}, s, t, \emptyset)$, $\hat{\mathbf{z}}_s^{\boldsymbol{\Phi}} = \boldsymbol{\Phi}(\tilde{\mathbf{z}}_t^{\boldsymbol{\Phi}}, t, s, \mathbf{y})$, and $\boldsymbol{D}$ denotes the VAE decoder.

B. Flaws in the Consistency Function of Guided Consistency Sampling

As we know, given a well-trained diffusion model $\boldsymbol{\phi}$, there exists an exact solution from timestep t to e [5]:

$$\boldsymbol{G}(\mathbf{z}_t, t, e, \mathbf{y}) = \frac{\alpha_e}{\alpha_t}\mathbf{z}_t + \alpha_s \int_{\lambda_t}^{\lambda_e} e^{-\lambda} \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_{t_\lambda(\lambda)}, t_\lambda(\lambda), \mathbf{y}) \mathrm{d}\lambda, \tag{2}$$

where $\lambda_t = \ln \frac{\alpha_t}{\sigma_t}$ and t_λ denotes the inverse function of λ_t. Inspired by [8], we find Eq. (6) suggests that GCS aims to optimize a 3D representation $\boldsymbol{\theta}$ such that $\boldsymbol{G}_{\boldsymbol{\theta}}(\mathbf{z}_t, t, e, \emptyset) = \boldsymbol{G}_{\boldsymbol{\theta}}(\hat{\mathbf{z}}_s^{\boldsymbol{\Phi}}, s, e, \emptyset)$ for $\forall t, s, e \in [0, T]$ where $t > s > e$. This implies $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y})$, as defined in Eq. (5), is not an approximation but an exact solution learning related to the 3D representation $\boldsymbol{\theta}$, *i.e.*, $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y}) = \frac{\int_{\lambda_t}^{\lambda_e} e^{-\lambda} \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}^{\mathbf{c}}_{t_\lambda(\lambda)}, t_\lambda(\lambda), \mathbf{y}) \mathrm{d}\lambda}{\int_{\lambda_t}^{\lambda_e} e^{-\lambda} \mathrm{d}\lambda}$. However, dropping the target timestep e in $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y})$, as GCS does, is problematic.

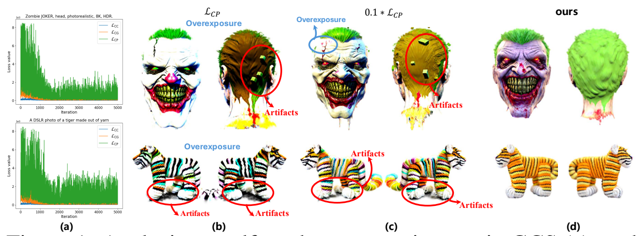

Figure 1. Analysis on self- and cross-consistency in GCS (a), and visual analysis of oversaturation and artifacts (b)-(d).

Suppose we predict both $\mathbf{z}_e$ and $\mathbf{z}_{e'}$ from $\mathbf{z}_t$ where $t > e' > e$, we must have

$$\begin{aligned}
\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y}) &= \frac{\int_{\lambda_t}^{\lambda_{e'}} e^{-\lambda} \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_{t_\lambda(\lambda)}, t_\lambda(\lambda), \mathbf{y}) \mathrm{d}\lambda}{\int_{\lambda_t}^{\lambda_{e'}} e^{-\lambda} \mathrm{d}\lambda}\\
\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y}) &= \frac{\int_{\lambda_t}^{\lambda_e} e^{-\lambda} \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_{t_\lambda(\lambda)}, t_\lambda(\lambda), \mathbf{y}) \mathrm{d}\lambda}{\int_{\lambda_t}^{\lambda_e} e^{-\lambda} \mathrm{d}\lambda}.
\end{aligned} \tag{3}$$

Clearly, without the target timestep in $\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y})$, optimizing a 3D representation $\boldsymbol{\theta}$ to satisfy both conditions in Eq. (3) for all intervals $[e, t]$ is invalid. For GCS, as the number of training steps increases, the above unreasonable phenomena occur frequently, potentially resulting in poor distillation results.

C. Oversaturation and Artifacts in GCS

Fig. 1 shows that the cross-consistency loss $\mathcal{L}_{\text{CG}} + \mathcal{L}_{\text{CP}}$ (with $\mathcal{L}_{\text{CP}}$ dominating) greatly exceeds the self-consistency loss $\mathcal{L}_{\text{CC}}$. As shown in Fig. 1, we scale down $\mathcal{L}_{\text{CP}}$ **(c)** and observe that the oversaturation **(b)** is alleviated, but undesirable geometry persists (red circle). This suggests that such oversaturation is brought from an excessively large value of cross-consistency loss, *i.e.,* the "excessive conditional guidance," which has been analyzed in Sec. 4.1. In contrast, our $\mathcal{L}_{\text{SCTD}}$ **(d)** successfully addresses the above issues.

D. How to Connect SCTD with SDS?

As described in Sec. 4.3, given any subtrajectory $[s_m, s_{m+1}]$, $\boldsymbol{G}_{\boldsymbol{\theta}}(\mathbf{z}_t, t, s_m, \mathbf{y}) := \boldsymbol{G}^m_{\boldsymbol{\theta}}(\mathbf{z}_t, t, \mathbf{y}) = \frac{\alpha_{s_m}}{\alpha_t}\mathbf{z}_t - \alpha_{s_m} \boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y}) \int_{\lambda_t}^{\lambda_{s_m}} e^{-\lambda} \mathrm{d}\lambda$. Then, we have

$$\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t, t, \mathbf{y}) = \frac{\boldsymbol{G}^m_{\boldsymbol{\theta}}(\mathbf{z}_t, t, \mathbf{y}) - \frac{\alpha_{s_m}}{\alpha_t}\mathbf{z}_t}{\alpha_{s_m} \int_{\lambda_t}^{\lambda_{s_m}} e^{-\lambda} \mathrm{d}\lambda}, \tag{4}$$

where $\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \boldsymbol{\epsilon}$. In this case, $\mathbf{y}$ can represent any prompt embedding, including $\emptyset$. According to Eq. (7), $\mathcal{L}_{\text{SDS}}$

can be further transformed into:

$$\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}=\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})+ \boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})+\omega(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\emptyset)), \tag{5}$$

where $\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s=\boldsymbol{\Phi}(\mathbf{z}_t,t,s,\mathbf{y})$. By substituting Eq. (4) into Eq. (5), we obtain

$$\begin{aligned}\mathcal{L}_{\text{SDS}}(\boldsymbol{\theta})=&\mathbb{E}_t[b(t)||\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\mathbf{z}_t,t,\mathbf{y})\\&+\omega(\boldsymbol{G}^m_{\boldsymbol{\theta}}(\mathbf{z}_t,t,\emptyset)-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\mathbf{z}_t,t,\mathbf{y}))-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})\\&+\frac{\alpha_{s_m}}{\alpha_t}\mathbf{z}_t-\alpha_{s_m}\int_{\lambda_t}^{\lambda_{s_m}}e^{-\lambda}\mathrm{d}\lambda\boldsymbol{\epsilon}||^2_2],\end{aligned} \tag{6}$$

where $b(t)=\frac{\omega(t)}{(\alpha_{s_m}\int_{\lambda_t}^{\lambda_{s_m}}e^{-\lambda}\mathrm{d}\lambda)^2}$. Furthermore,

$$\begin{aligned}&\frac{\alpha_{s_m}}{\alpha_t}\mathbf{z}_t-\alpha_{s_m}\int_{\lambda_t}^{\lambda_{s_m}}e^{-\lambda}\mathrm{d}\lambda\boldsymbol{\epsilon}\\&=\frac{\alpha_{s_m}}{\alpha_t}(\alpha_t\mathbf{z}^{\mathbf{c}}_0+\sigma_t\boldsymbol{\epsilon})+\alpha_{s_m}(e^{\lambda_{s_m}-\lambda_t}-1)\boldsymbol{\epsilon}\\&=\alpha_{s_m}\mathbf{z}_0+\sigma_{s_m}\boldsymbol{\epsilon}\\&=\mathbf{z}_{s_m}\end{aligned} \tag{7}$$

Substituting Eq. (7) into Eq. (6), we can obtain Eq. (8). Proof is completed.

E. Computation Reduction Trick

Eq. (7) can be transformed into

$$\begin{aligned}\hat{\boldsymbol{\epsilon}}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}&=\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}+\omega(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\emptyset))\\&=\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\emptyset)-\boldsymbol{\epsilon}+(\omega+1)(\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\mathbf{y})-\boldsymbol{\epsilon}_{\boldsymbol{\phi}}(\mathbf{z}_t,t,\emptyset))\end{aligned} \tag{8}$$

Based on Eq. (8), we can readily derive Eq. (9) by following the procedure described in App. D.

F. Proof of Upper Bound of Distillation Error

Before proving Theorem 1, we first give the following lemma:

Lemma 1. *Given a sub-trajectory* $[s_m,s_{m+1}]$*, let* $\Delta t=max_{t,s\in[s_m,s_{m+1})}\{|t-s|\}$*. We assume* $\boldsymbol{G}^m_{\boldsymbol{\theta}}$ *satisfies the Lipschitz condition and the ODE solver has local error uniformly bounded by* $O(t-s)^{p+1}$ *with* $p\geq 1$*. If* $\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,\emptyset)=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\emptyset)$ *for* $\forall t,s\in[s_m,s_{m+1}]$*, we have*

$$\begin{aligned}&\sup_{t,s\in[s_m,s_{m+1})}||\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,\mathbf{y})-\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,s_m,\mathbf{y})||\\&=O((\Delta t)^p)(s_{m+1}-s_m).\end{aligned} \tag{9}$$

Proof. The proof is based on [2, 7–9]. Let

$$\boldsymbol{e}_{n-1}:=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,s_m,\mathbf{y}), \tag{10}$$

where $\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s=\boldsymbol{\Phi}(\mathbf{z}^{\mathbf{c}}_{s_m},s_m,s,\mathbf{y})$. According to the condition, we have

$$\begin{aligned}\boldsymbol{e}_n&=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,\mathbf{y})-\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,s_m,\mathbf{y})\\&=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})\\&+\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,s_m,\mathbf{y})\\&=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})+\boldsymbol{e}_{n-1},\end{aligned} \tag{11}$$

Provided that $\boldsymbol{G}^m_{\boldsymbol{\theta}}$ satisfies L-Lipschitz condition, we have

$$\begin{aligned}&||e_n||=||e_{n-1}+\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})||\\&\leq||e_{n-1}||+||\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})-\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,\mathbf{y})||\\&\leq\boldsymbol{e}_{n-1}+L||\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_s-\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s||\\&\leq\boldsymbol{e}_{n-1}+L\cdot O((t-s)^{p+1})\\&\leq\boldsymbol{e}_{n-1}+L(t-s)\cdot O((\Delta t)^p).\end{aligned} \tag{12}$$

Besides, according to the boundray condition,

$$\begin{aligned}\boldsymbol{e}_{s_m}&=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_{s_m},s_m,\mathbf{y})-\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_{s_m},s_m,s_m,\mathbf{y})\\&=\hat{\mathbf{z}}^{\boldsymbol{\Phi}}_{s_m}-\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_{s_m}=\mathbf{z}^{\mathbf{c}}_{s_m}-\mathbf{z}^{\mathbf{c}}_{s_m}=0,\end{aligned} \tag{13}$$

Therefore,

$$\begin{aligned}&||\boldsymbol{e}_n||\leq||\boldsymbol{e}_{s_m}||+L\sum_{t_i,t_{i-1}\in[s_m,s_{m+1}]}(t_i-t_{i-1})O((\Delta t)^p)\\&=O((\Delta t)^p)\cdot(s_{m+1}-s_m).\end{aligned} \tag{14}$$

The proof is completed. □

According to Eq. (12), one can ideally optimize a 3D model $\boldsymbol{\theta}$ such that $\mathbf{z}_{s_m}=\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,\mathbf{y})$. In this case, we have $\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_s,s,s_m,\mathbf{y})=\mathbf{z}^{data}_{s_m}$, where $\mathbf{z}^{data}_{s_m}=\alpha_{s_m}\mathbf{z}^{data}+\sigma_{s_m}\boldsymbol{\epsilon}^*$. Based on this, we have

$$||\boldsymbol{G}^m_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,\mathbf{y})-\boldsymbol{\Phi}(\tilde{\mathbf{z}}^{\boldsymbol{\Phi}}_t,t,s_m,\mathbf{y})||=||\mathbf{z}_{s_m}-\mathbf{z}^{data}_{s_m}||=||\mathbf{z}_0-\mathbf{z}^{data}||. \tag{15}$$

Since we use a first-order ODE solver to implement $\boldsymbol{\Phi}$, we have

$$\sup_{t,s\in[s_m,s_{m+1})}||\mathbf{z}_0-\mathbf{z}^{data}||=\mathcal{O}(\Delta t)(s_{m+1}-s_m). \tag{16}$$

The proof is completed.

G. Visual Comparisons of State-of-the-arts

We also present additional visual comparisons with state-of-the-art methods, as shown in Fig. 2 and Fig. 3. We provide additional qualitative comparisons against DreamFusion [6], LucidDreamer [3], Consistent3D [9], ConnectCD [2], Magic3D [4], Fantasia3D [1], and CSD [10]. As shown, our method clearly outperforms others visually.

Figure 2. Additional qualitative comparisons with DreamFusion [6], LucidDreamer [3], Consistent3D [9], and ConnectCD [2]. CFG scales are set to 100, 7.5, 20∼40, 7.5, 7.5, respectively. Our approach yields results with high quality. Please zoom in for details.

References

[1] 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 International Conference on Computer Vision (ICCV)*, pages 22246–22256, 2023. 2

[2] Zongrui Li, Minghui Hu, Qian Zheng, and Xudong Jiang. Connecting consistency distillation to score distillation for text-to-3d generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 274–291. Springer, 2025. 1, 2, 3

Figure 3. Additional qualitative comparisons with Magic3D [6], Fantasia3D [3], and CSD [9]. Our approach yields results with high quality. Please zoom in for details.

Table 1. 40 prompts for evaluation.

Column 1	**Column 2**
1. A cat with a mullet	21. A blue motorcycle
2. A pig wearing a backpack	22. Michelangelo style statue of an astronaut
3. A DSLR photo of an origami crane	23. A DSLR photo of a chow chow puppy
4. A photo of a mouse playing the tuba	24. A DSLR photo of cats wearing eyeglasses
5. An orange road bike	25. A red panda
6. A ripe strawberry	26. A DSLR photo of an elephant skull
7. A DSLR photo of the Imperial State Crown of England	27. An amigurumi bulldozer
8. A photo of a wizard raccoon casting a spell	28. A typewriter
9. A DSLR photo of a corgi wearing a top hat	29. A red-eyed tree frog, low poly
10. A rabbit, animated movie character, high-detail 3D model	30. A DSLR photo of a chimpanzee wearing headphones
11. A panda rowing a boat	31. A robot made out of vegetables
12. A highly detailed sand castle	32. A DSLR photo of a red rotary telephone
13. A DSLR photo of a chimpanzee dressed like Henry VIII king of England	33. A DSLR photo of a blue lobster
14. A photo of a skiing penguin wearing a puffy jacket, highly realistic DSLR photo	34. A DSLR photo of a squirrel flying a biplane
15. A blue poison-dart frog sitting on a water lily	35. A DSLR photo of a baby dragon hatching out of a stone egg
16. A DSLR photo of a bear dressed in medieval armor	36. A DSLR photo of a bear dancing ballet
17. A DSLR photo of a squirrel dressed like a clown	37. A plate of delicious tacos
18. A plush toy of a corgi nurse	38. A DSLR photo of a car made out of cheese
19. A humanoid robot playing the violin	39. A yellow school bus
20. A DSLR photo of a bear dressed as a lumberjack	40. A DSLR photo of a shiny beetle

[3] Yixun Liang, Xin Yang, Jiantao Lin, Haodong Li, Xiaogang Xu, and Yingcong Chen. Luciddreamer: Towards high-fidelity text-to-3d generation via interval score matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6517–6526, 2024. 2, 3, 4

[4] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 300–309, 2023. 2

[5] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, pages 5775–5787, 2022. 1

[6] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Milden-

hall. Dreamfusion: Text-to-3d using 2d diffusion. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–10, 2022. 2, 3, 4

[7] Yang Song, Prafulla Dhariwal, Mark Chen, and Ilya Sutskever. Consistency models. In *Proceedings of the International Conference on Machine Learning (ICLR)*, 2023. 2

[8] Fu-Yun Wang, Zhaoyang Huang, Alexander William Bergman, Dazhong Shen, Peng Gao, Michael Lingelbach, Keqiang Sun, Weikang Bian, Guanglu Song, Yu Liu, et al. Phased consistency model. In *Proceedings of the International Conference on Neural Information Processing Systems (NeurIPS)*, 2024. 1

[9] Zike Wu, Pan Zhou, Xuanyu Yi, Xiaoding Yuan, and Hanwang Zhang. Consistent3d: Towards consistent high-fidelity text-to-3d generation with deterministic sampling prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9892–9902, 2024. 2, 3, 4

[10] Xin Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Song-Hai Zhang, and Xiaojuan Qi. Text-to-3d with classifier score distillation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2024. 2