Posterior Distillation Sampling

Juil Koo Chanho Park Minhyuk Sung
KAIST
{63days,charlieppark,mhsung}@kaist.ac.kr

Abstract

We introduce Posterior Distillation Sampling (PDS), a novel optimization method for parametric image editing based on diffusion models. Existing optimization-based methods, which leverage the powerful 2D prior of diffusion models to handle various parametric images, have mainly focused on generation. Unlike generation, editing requires a balance between conforming to the target attribute and preserving the identity of the source content. Recent 2D image editing methods have achieved this balance by leveraging the stochastic latent encoded in the generative process of diffusion models. To extend the editing capabilities of diffusion models shown in pixel space to parameter space, we reformulate the 2D image editing method into an optimization form named PDS. PDS matches the stochastic latents of the source and the target, enabling the sampling of targets in diverse parameter spaces that align with a desired attribute while maintaining the source's identity. We demonstrate that this optimization resembles running a generative process with the target attribute, but aligning this process with the trajectory of the source's generative process. Extensive editing results in Neural Radiance Fields and Scalable Vector Graphics representations demonstrate that PDS is capable of sampling targets to fulfill the aforementioned balance across various parameter spaces.

1. Introduction

Diffusion models [10, 33–36] have recently led to rapid development in text-conditioned generation and editing across diverse domains, including 2D images [8, 12, 17, 37, 40], 3D objects [15, 16, 18, 23], and audio [5, 11, 43]. Among them, in particular, 2D image diffusion models [4, 21, 27–29] have demonstrated their powerful generative prior aided by Internet-scale image and text datasets [2, 30, 31]. Nonetheless, this rich 2D generative prior has been confined to pixel space, limiting their broader applicability. A pioneer work overcoming this limitation, DreamFusion [24], has introduced Score Distillation Sampling (SDS). It leverages the generative prior of text-to-image diffusion models to synthesize 3D scenes represented by Neural Radiance Fields (NeRFs) [22] from texts. Beyond NeRF representations [3, 19, 26, 32, 38, 39, 45], SDS has been widely applied to various parameter spaces, where images are not represented by pixels but specific parameterizations, such as texture [1, 20], material [42] and Scalable Vector Graphics (SVGs) [13, 14, 41].

While SDS [24] has achieved great advances in generating parametric images, editing is also an essential element for full freedom in handling visual content. Editing differs from generation in that it requires considerations of both the target text and the original source content, thereby emphasizing two key aspects: (1) alignment with the target text prompt and (2) preservation of the source content's identity. To extend SDS, which lacks the latter aspect, Hertz *et al.* [7] propose Delta Denoising Score (DDS). DDS reduces the noisy gradients inherent in SDS, leading to better-maintaining background details and sharper editing outputs. However, the optimization function of DDS still lacks an explicit term for identity preservation.

To address the absence of preserving the source's identity in SDS [24] and DDS [7], we turn our attention to a recent 2D image editing method [12, 40] based on diffusion models, known as stochastic diffusion inversion. Their primary objective is to compute the stochastic latent of an input image within the generative process of diffusion models. Once the stochastic latent of a source image is computed, the source image can be edited by running a generative process with new conditions, such as new target text prompts, while feeding the source's stochastic latent into the process.

To extend the editing capabilities of the stochastic diffusion inversion method from pixel space to parameter space, we reformulate this method into an optimization form named Posterior Distillation Sampling (PDS). Unlike SDS [24] and DDS [7], which match two noise variables, PDS aims to match the stochastic latents of the source and the optimized target.

Our extensive editing experiment results, including NeRF editing (Section 4.1) and SVG editing (Section 4.2), demonstrate the versatility of our method for parametric image editing. In NeRF editing, we are the first to produce large geometric changes or to add objects to arbitrary regions without specifying local regions to be edited. Fig-

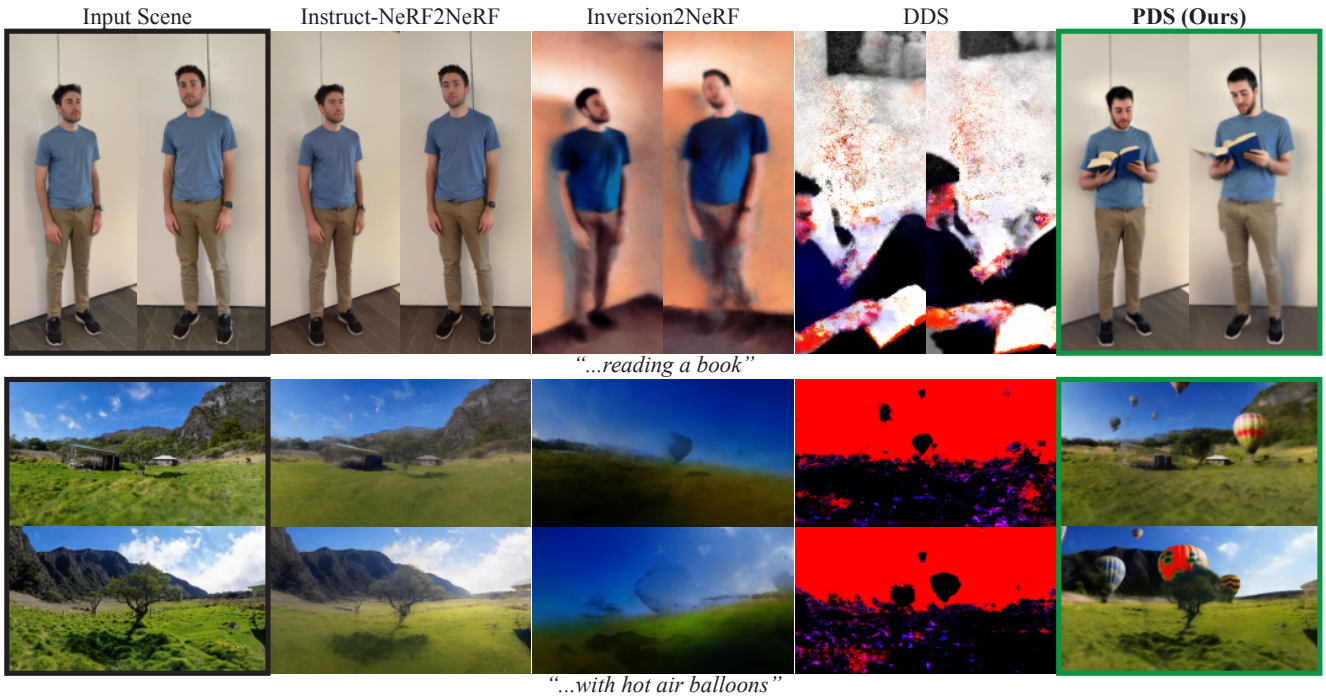

Figure 1. **A comparison of 3D scene editing between PDS and other baselines.** Given input 3D scenes on the left, PDS, marked by green boxes on the rightmost side, successfully performs complex editing, such as geometric changes and adding objects, according to the input texts. On the other hand, the baselines either fail to change the input 3D scenes or produce results that greatly deviate from the input scenes, losing their identity.

ure 1 shows these examples. Qualitative and quantitative comparisons of SVG editing with other optimization methods, namely SDS [24] and DDS [7], have demonstrated that PDS produces only the necessary changes to source SVGs, effectively aligning them with the target prompts.

2. Preliminaries

Score Distillation Sampling (SDS) [24]. Score Distillation Sampling (SDS) [24] is proposed to generate parametric images by leveraging the 2D prior of pre-trained text-to-image diffusion models. Given an input data $\mathbf{x}_0$ and a text prompt y, the training objective function of diffusion models is to predict injected noise $\boldsymbol{\epsilon}$ using a noise predictor $\boldsymbol{\epsilon}_\phi$:

$$\mathcal{L}(\mathbf{x}_0) = \mathbb{E}_{t\sim\mathcal{U}(0,1),\boldsymbol{\epsilon}_t}\left[w(t)\|\boldsymbol{\epsilon}_\phi(\mathbf{x}_t, y, t) - \boldsymbol{\epsilon}_t\|_2^2\right], \quad (1)$$

where $w(t)$ is a weighting function and $\mathbf{x}_t$ results from the forward process of diffusion models:

$$\mathbf{x}_t := \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t, \quad \boldsymbol{\epsilon}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}) \quad (2)$$

with variance schedule variables $\bar{\alpha}_t := \prod_{s=1}^{t}\alpha_s$. When the input data $\mathbf{x}_0$ is generated by a differentiable image generator $\mathbf{x}_0 = g(\theta)$, parameterized by θ, SDS updates θ by back-propagating the gradient of Equation 1 while omitting the U-Net jacobian term $\frac{\partial\boldsymbol{\epsilon}_\phi}{\partial\mathbf{x}_t}$ for computation efficiency:

$$\nabla_\theta\mathcal{L}_{\text{SDS}}(\mathbf{x}_0 = g(\theta)) = \mathbb{E}_{t,\boldsymbol{\epsilon}_t}\left[w(t)(\boldsymbol{\epsilon}_\phi(\mathbf{x}_t, y, t) - \boldsymbol{\epsilon}_t)\frac{\partial\mathbf{x}_0}{\partial\theta}\right], \quad (3)$$

where we denote a noise prediction of diffusion models with classifier-free guidance [9] by $\boldsymbol{\epsilon}_\phi$ for simplicity. Through this optimization process, SDS is capable of generating a parametric image which conforms to the input text prompt y.

Delta Denoising Score (DDS) [7]. Even though SDS has been widely used for various parametric images, its optimization is designed for generation, thus it does not reflect one of the key aspects of editing: preserving the source identity.

To extend SDS to editing, Hertz *et al.* [7] have proposed Delta Denoising Score (DDS). Given source data $\mathbf{x}^{\text{src}}$ and its corresponding text prompt y^{src}, the goal of DDS is to synthesize new target data $\mathbf{x}^{\text{tgt}}$ that is aligned with a target text prompt y^{tgt}. In the SDS formula 3, DDS replaces randomly sampled noise $\boldsymbol{\epsilon}$ with a noise prediction given a

source data-text pair $\epsilon_\phi(\mathbf{x}_t^{\text{src}}, y^{\text{src}}, t)$:

$$\nabla_\theta \mathcal{L}_{\text{DDS}} = \mathbb{E}_{t,\boldsymbol{\epsilon}_t}\left[w(t)\left(\boldsymbol{\epsilon}_\phi(\mathbf{x}_t^{\text{tgt}}, y^{\text{tgt}}, t) - \boldsymbol{\epsilon}_\phi(\mathbf{x}_t^{\text{src}}, y^{\text{src}}, t)\right)\frac{\partial \mathbf{x}_0^{\text{tgt}}}{\partial \theta}\right], \tag{4}$$

where the same noise $\boldsymbol{\epsilon}_t$ is shared for $\mathbf{x}_t^{\text{src}}$ and $\mathbf{x}_t^{\text{tgt}}$:

$$\begin{aligned}\boldsymbol{\epsilon}_t &\sim \mathcal{N}(\mathbf{0}, \mathbf{I}),\\ \mathbf{x}_t^{\text{src}} &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0^{\text{src}} + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t,\\ \mathbf{x}_t^{\text{tgt}} &= \sqrt{\bar{\alpha}_t}\mathbf{x}_0^{\text{tgt}} + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t.\end{aligned} \tag{5}$$

While DDS extends SDS for editing tasks, it lacks an explicit term in its optimization to preserve the identity of the source. As a result, DDS is still prone to produce editing results that significantly deviate from the source.

Stochastic Latent in Generative Process. To achieve both conformity to the text and preservation of the source's identity, we turn our attention to the rich information encoded in the stochastic generative process of DDPM [10]. When $\beta_t := 1 - \alpha_t$ are small, it is well-known that the posterior of the forward process also follows a Gaussian distribution according to a property of Gaussians. The forward process posteriors are represented as:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\boldsymbol{\mu}(\mathbf{x}_t, \mathbf{x}_0), \sigma_t \mathbf{I}), \tag{6}$$

where $\sigma_t := \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t}\beta_t$ and the posterior mean $\boldsymbol{\mu}$ is a linear combination of $\mathbf{x}_0$ and $\mathbf{x}_t$: $\boldsymbol{\mu}(\mathbf{x}_t, \mathbf{x}_0) := \gamma_t \mathbf{x}_0 + \delta_t \mathbf{x}_t$ with $\gamma_t := \frac{\sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)}{1-\bar{\alpha}_t}$ and $\delta_t := \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t}$.

Since $\mathbf{x}_0$ is unknown during a generative process, we approximate $\mathbf{x}_0$ with a one-step denoised estimate as follows:

$$\tilde{\mathbf{x}}_0(\mathbf{x}_t, y; \boldsymbol{\epsilon}_\phi) := \frac{1}{\sqrt{\bar{\alpha}_t}}(\mathbf{x}_t - \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_\phi(\mathbf{x}_t, y, t)). \tag{7}$$

Consequently, one step of the generative process is represented as follows:

$$\mathbf{x}_{t-1} = \boldsymbol{\mu}_\phi(\mathbf{x}_t, y; \boldsymbol{\epsilon}_\phi) + \sigma_t \mathbf{z}_t, \quad \mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{8}$$

where $\boldsymbol{\mu}_\phi(\mathbf{x}_t, y; \boldsymbol{\epsilon}_\phi) = \gamma_t \tilde{\mathbf{x}}_0(\mathbf{x}_t, y; \boldsymbol{\epsilon}_\phi) + \delta_t \mathbf{x}_t$.

Using Equation 8, one can compute stochastic latent $\tilde{\mathbf{z}}_t$ that captures the structural details of $\mathbf{x}_0$. This involves computing $\mathbf{x}_t$ and $\mathbf{x}_{t-1}$ via the forward process and then rearranging Equation 8 as follows:

$$\tilde{\mathbf{z}}_t(\mathbf{x}_0, y; \boldsymbol{\epsilon}_\phi) = \frac{\mathbf{x}_{t-1} - \boldsymbol{\mu}_\phi(\mathbf{x}_t, y; \boldsymbol{\epsilon}_\phi)}{\sigma_t}. \tag{9}$$

Several recent works [12, 40], known as DDPM inversion, have utilized the stochastic latent for image editing tasks. To edit an image using $\tilde{\mathbf{z}}_t$, they first pre-compute $\tilde{\mathbf{z}}_t$ of the source image across all t in the generative process. They then run a new generative process with a new target prompt while incorporating the pre-computed $\tilde{\mathbf{z}}_t$ of the source into the process instead of randomly sampled noise $\mathbf{z}_t$.

3. Posterior Distillation Sampling

Here, we introduce Posterior Distillation Sampling (PDS), a novel optimization function designed for parametric image editing.

Our objective is to synthesize $\mathbf{x}_0^{\text{tgt}}$ that is aligned with y^{tgt} while it retains the identity of $\mathbf{x}_0^{\text{src}}$. To achieve this, we employ the stochastic latent $\tilde{\mathbf{z}}_t$ in our optimization. For simplicity, we denote the stochastic latents of the source and the target as follows:

$$\tilde{\mathbf{z}}_t^{\text{src}} := \tilde{\mathbf{z}}_t(\mathbf{x}_0^{\text{src}}, y^{\text{src}}; \boldsymbol{\epsilon}_\phi) \tag{10}$$

$$\tilde{\mathbf{z}}_t^{\text{tgt}} := \tilde{\mathbf{z}}_t(\mathbf{x}_0^{\text{tgt}}, y^{\text{tgt}}; \boldsymbol{\epsilon}_\phi). \tag{11}$$

Using the stochastic latents, we define a novel objective function as follows:

$$\mathcal{L}_{\tilde{\mathbf{z}}_t}(\mathbf{x}_0^{\text{tgt}} = g(\theta)) := \mathbb{E}_{t,\boldsymbol{\epsilon}_{t-1},\boldsymbol{\epsilon}_t}\left[\|\tilde{\mathbf{z}}_t^{\text{tgt}} - \tilde{\mathbf{z}}_t^{\text{src}}\|_2^2\right], \tag{12}$$

where, similar to Equation 5, $\tilde{\mathbf{z}}_t^{\text{src}}$ and $\tilde{\mathbf{z}}_t^{\text{tgt}}$ share the same noises, denoted by $\boldsymbol{\epsilon}_{t-1}$ and $\boldsymbol{\epsilon}_t$, when computing their respective $\mathbf{x}_{t-1}$ and $\mathbf{x}_t$.

Rather than matching noise variables as in SDS [24] and DDS [7], we match the stochastic latents of the source and the target via the optimization. By taking the gradient of $\mathcal{L}_{\tilde{\mathbf{z}}_t}$ with respect to θ and ignoring the U-Net jacobian term as previous works [7, 24, 38], one can obtain PDS as follows:

$$\nabla_\theta \mathcal{L}_{\text{PDS}} := \mathbb{E}_{t,\boldsymbol{\epsilon}_t,\boldsymbol{\epsilon}_{t-1}}\left[w(t)(\tilde{\mathbf{z}}_t^{\text{tgt}} - \tilde{\mathbf{z}}_t^{\text{src}})\frac{\partial \mathbf{x}_0^{\text{tgt}}}{\partial \theta}\right]. \tag{13}$$

Expanding Equation 13, the following detailed formulation is derived:

$$\nabla_\theta \mathcal{L}_{\text{PDS}} := \mathbb{E}_{t,\boldsymbol{\epsilon}_t,\boldsymbol{\epsilon}_{t-1}}\left[(\psi(t)(\mathbf{x}_0^{\text{tgt}} - \mathbf{x}_0^{\text{src}}) + \chi(t)(\hat{\boldsymbol{\epsilon}}_t^{\text{tgt}} - \hat{\boldsymbol{\epsilon}}_t^{\text{src}}))\frac{\partial \mathbf{x}_0^{\text{tgt}}}{\partial \theta}\right], \tag{14}$$

where $\hat{\boldsymbol{\epsilon}}_t^{\text{src}} := \boldsymbol{\epsilon}_\phi(\mathbf{x}_t^{\text{src}}, y^{\text{src}}, t)$ and $\hat{\boldsymbol{\epsilon}}_t^{\text{tgt}} := \boldsymbol{\epsilon}_\phi(\mathbf{x}_t^{\text{tgt}}, y^{\text{tgt}}, t)$. We leave a more detailed derivation to the **supplementary material**.

4. Experiment Results

4.1. NeRF Editing

We evaluate our method against three baselines: Instruct-NeRF2NeRF (IN2N) [6], DDS [7] and Inversion2NeRF

Table 1. **A quantitative comparison of NeRF editing between ours and other baselines.** Ours outperforms the baselines quantitatively. **Bold** indicates the best result for each column.

Methods	CLIP [25] Score ↑	User Preference Rate (%) ↑
IN2N [6]	0.2280	27.71
DDS [7]	0.2210	13.71
Inv2N	0.2232	9.24
PDS (Ours)	**0.2477**	**49.33**

Table 2. **A quantitative comparison of SVG editing between SDS [24], DDS [7] and PDS.** Ours outperforms the others in LPIPS [44] while achieving a CLIP [25] score that is on par with the others. **Bold** indicates the best result for each column.

Methods	CLIP [25] Score ↑	LPIPS [44] ↓	User Preference Rate (%) ↑
SDS [24]	**0.2606**	0.4855	30.83
DDS [7]	0.2460	0.5982	20.24
PDS (Ours)	0.2504	**0.3121**	**48.94**

(Inv2N). Similar to IN2N [6], Inv2N is also based on Iterative DU, which performs editing within 2D space, but employs DDPM inversion [12] for 2D editing. Figure 1 presents the qualitative comparisons of NeRF editing. Notably, as depicted in row 1, our method is the only one that makes large geometric changes in 3D scenes from the input text, folding the man's arms to create natural poses of him reading a book. In contrast, Iterative-DU-based methods like IN2N [6] and Inv2N fail to produce the right edits in 3D space. DDS [7] produces the outputs that completely lose the identity of the input scenes, focusing solely on conforming to the input texts. Row 2 of Figure 1 shows the editing scenario of adding objects in an outdoor scene without specifying local regions, which also leads to large variations. Here, our method successfully adds the objects in the input scene, maintaining their background details. On the other hand, the baselines either fail to add the objects in 3D space or produce outputs that significantly deviate from the original scenes. We provide more qualitative results in the **supplementary material**.

To further assess the perceptual quality of the editing results, we conduct a user study compared to the baselines. As illustrated in Table 1, our editing results are most preferred over the baselines in human evaluation by a large margin: 49.33% (Ours) vs. 27.71% (IN2N [6], the second best). See the **supplementary material** for a more detailed user study setup.

For a quantitative evaluation, we measure CLIP [25] Score that measures the similarity between edited 2D renderings and target text prompts in CLIP [25] space. As shown in Table 1, ours outperforms the baselines quantitatively.

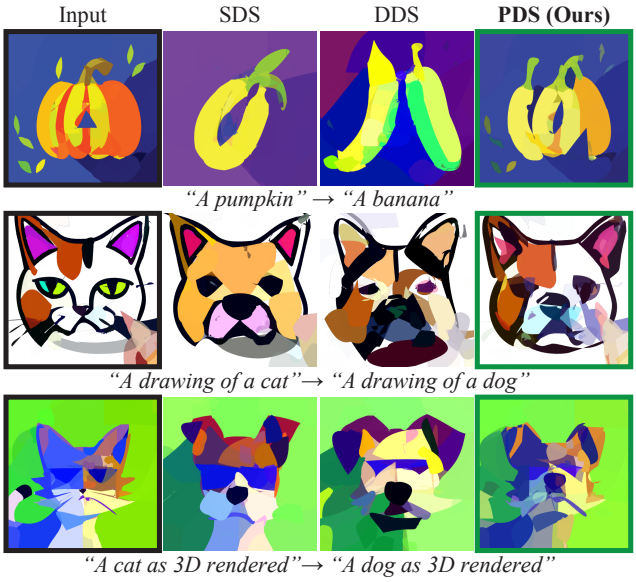

Figure 2. **A qualitative comparison of SVG editing using three different optimization methods: SDS [24], DDS [7] and PDS.** PDS makes changes according to input text while most preserving the structural semantics of the input SVGs.

4.2. SVG Editing

Qualitative results of SVG editing are shown in Figure 2. It demonstrates that while all the methods effectively change input SVGs according to the target text prompts, ours best preserves the structural semantics of the input SVGs. This is particularly evident in row 2 of Figure 2, where ours maintains the overall color pattern of the input SVG.

The trends from the qualitative results are mirrored in our quantitative results. As seen in Table 2, ours significantly surpasses the others in LPIPS [44] by a large margin, which measures the fidelity to the input SVG, while our CLIP score is on par with the others. This demonstrates that our method introduces only minimal necessary changes to meet the described attributes in the target text prompts.

We further provide a user study result of SVG editing in Table 2. We use the same user study setup used in NeRF editing (Section 4.1). Consistent with the qualitative and quantitative results, ours are most preferred in human evaluation.

5. Conclusion

We propose Posterior Distillation Sampling (PDS), an optimization method for parametric image editing. PDS matches the stochastic latents of the source and the target to fulfill both conformity to the target text and preservation of the source identity in parameter space. We demonstrate the versatility of PDS in parametric image editing through a comparative analysis between ours and other optimization methods and extensive experiments across various parameter spaces.

References

[1] Anonymous. Learning pseudo 3D guidance for view-consistent 3D texturing with 2D diffusion. In *Submitted to The Twelfth International Conference on Learning Representations*, 2023. under review. 1

[2] Minwoo Byeon, Beomhee Park, Haecheon Kim, Sungjun Lee, Woonhyuk Baek, and Saehoon Kim. Coyo-700m: Image-text pair dataset. https://github.com/kakaobrain/coyo-dataset, 2022. 1

[3] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3D: Disentangling geometry and appearance for high-quality text-to-3D content creation. In *ICCV*, 2023. 1

[4] DeepFloyd. Deepfloyd if. https://www.deepfloyd.ai/deepfloyd-if/. 1

[5] Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. Text-to-audio generation using instruction tuned llm and latent diffusion model. *arXiv preprint arXiv:2304.13731*, 2023. 1

[6] Ayaan Haque, Matthew Tancik, Alexei Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-NeRF2NeRF: Editing 3D scenes with instructions. In *ICCV*, 2023. 3, 4

[7] Amir Hertz, Kfir Aberman, and Daniel Cohen-Or. Delta denoising score. In *ICCV*, 2023. 1, 2, 3, 4

[8] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-or. Prompt-to-prompt image editing with cross-attention control. In *ICLR*, 2023. 1

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

[10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 1, 3

[11] Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models. *arXiv preprint arXiv:2301.12661*, 2023. 1

[12] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly DDPM noise space: Inversion and manipulations. *arXiv preprint arXiv:2304.06140*, 2023. 1, 3, 4

[13] Shir Iluz, Yael Vinker, Amir Hertz, Daniel Berio, Daniel Cohen-Or, and Ariel Shamir. Word-as-image for semantic typography. *ACM TOG*, 2023. 1

[14] Ajay Jain, Amber Xie, and Pieter Abbeel. Vectorfusion: Text-to-svg by abstracting pixel-based diffusion models. In *CVPR*, 2023. 1

[15] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 1

[16] Juil Koo, Seungwoo Yoo, Minh Hieu Nguyen, and Minhyuk Sung. SALAD: Part-level latent diffusion for 3d shape generation and manipulation. In *ICCV*, 2023. 1

[17] Yuseung Lee, Kunho Kim, Hyunjin Kim, and Minhyuk Sung. Syncdiffusion: Coherent montage via synchronized joint diffusions. In *NeurIPS*, 2023. 1

[18] Muheng Li, Yueqi Duan, Jie Zhou, and Jiwen Lu. Diffusion-sdf: Text-to-shape via voxelized diffusion. In *CVPR*, 2023. 1

[19] 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 *CVPR*, 2023. 1

[20] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3D shapes and textures. In *CVPR*, 2023. 1

[21] Midjourney. Midjourney. https://www.midjourney.com/. 1

[22] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV*, 2020. 1

[23] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts. *arXiv preprint arXiv:2212.08751*, 2022. 1

[24] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3D using 2D diffusion. In *ICLR*, 2023. 1, 2, 3, 4

[25] 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 *ICML*, 2021. 4

[26] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3D: Subject-driven text-to-3D generation. In *ICCV*, 2023. 1

[27] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 1

[28] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022.

[29] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *NeurIPS*, 2022. 1

[30] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 1

[31] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 1

[32] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. MVDream: Multi-view diffusion for 3D generation. *arXiv preprint arXiv:2308.16512*, 2023. 1

[33] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*, 2015. 1

[34] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2021.

[35] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, 2019.

[36] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *ICLR*, 2021. 1

[37] Bram Wallace, Akash Gokul, and Nikhil Naik. EDICT: Exact diffusion inversion via coupled transformations. In *CVPR*, 2023. 1

[38] Haochen Wang, Xiaodan Du, Jiahao Li, Raymond A Yeh, and Greg Shakhnarovich. Score jacobian chaining: Lifting pretrained 2D diffusion models for 3D generation. In *CVPR*, 2023. 1, 3

[39] 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. In *NeurIPS*, 2023. 1

[40] Chen Henry Wu and Fernando De la Torre. A latent space of stochastic diffusion models for zero-shot image editing and guidance. In *ICCV*, 2023. 1, 3

[41] Ximing Xing, Chuang Wang, Haitao Zhou, Jing Zhang, Qian Yu, and Dong Xu. Diffsketcher: Text guided vector sketch synthesis through latent diffusion models. In *NeurIPS*, 2023. 1

[42] Xudong Xu, Zhaoyang Lyu, Xingang Pan, and Bo Dai. Matlaber: Material-aware text-to-3D via latent BRDF auto-encoder. *arXiv preprint arXiv:2308.09278*, 2023. 1

[43] Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. Diffsound: Discrete diffusion model for text-to-sound generation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2023. 1

[44] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 4

[45] Joseph Zhu and Peiye Zhuang. HiFA: High-fidelity text-to-3D with advanced diffusion guidance. *arXiv preprint arXiv:2305.18766*, 2023. 1