Supplementary of LN3Diff: Scalable Latent Neural Fields Diffusion for Speedy 3D Generation

Yushi Lan[1], Fangzhou Hong[1], Shuai Yang[2], Shangchen Zhou[1], Xuyi Meng[1], Bo Dai[3], Xingang Pan[1], and Chen Change Loy[1]

[1] S-Lab, Nanyang Technological University, Singapore
[2] Wangxuan Institute of Computer Technology, Peking University
[3] Shanghai Artificial Intelligence Laboratory

In this supplementary material, we provide additional details regarding the implementations and additional results. We also discuss the limitations of our model.

Broader Social Impact. In this paper, we introduce a new latent 3D diffusion model designed to produce high-quality textures and geometry using a single model. As a result, our approach has the potential to be applied to generating DeepFakes or deceptive 3D assets, facilitating the creation of falsified images or videos. This raises concerns as individuals could exploit such technology with malicious intent, aiming to spread misinformation or tarnish reputations.

A Implementation details

A.1 Training details

Diffusion. We mainly adopt the diffusion training pipeline implementation from ADM [4], continuous noise schedule from LSGM [22] with the spatial transformer attention implementation from LDM [16]. For ShapeNet and FFHQ dataset, we adopt U-Net [17] architecture and list the hyperparameters in Tab. 1. For Objaverse dataset, we adopt DiT-L [14] architecture with cross attention design, as proposed in PixArt [3]. The diffusion transformer is built with 24 layers with 16 heads and 1024 hidden dimension, which result in 458M parameters.

VAE Architecture. For the convolutional encoder $\mathcal{E}_{\phi}$, we adopt a lighter version of LDM [16] encoder with channel 64 and 1 residual blocks for efficiency. When training on Objaverse with $V = 6$, we incorporate 3D-aware attention [18] in the middle layer of the convolutional encoder. For convolutional upsampler $\mathcal{D}_U$, we further half the channel to 32. All other hyper-parameters remain at their default settings. Regarding the transformer decoder $\mathcal{D}_T$, we employ the DiT-L/2 architecture, and overall saved VAE model takes around 1.5 GiB storage. The input dimension of z to the MLP in each DiT block is $h \times w \times c$ for self-plane attention, and $h \times w \times 3 \times c$ in cross-plane attention. When ablating the 3D-aware attention in Tab.3, we adopt channel-wise concatenated latent $h \times w \times (3c)$ for model input, as in SSDNeRF. Note that we trade off a smaller model with faster training speed due to the overall compute limit, and a heavier model would

Table 1: Hyperparameters and architecture of diffusion model ϵ_θ.

Diffusion Model Details	
Learning Rate	$2e-5$
Batch Size	96
Optimizer	AdamW
Iterations	500K
U-Net base channels	320
U-Net channel multiplier	1, 1, 2, 2, 4, 4
U-Net res block	2
U-Net attention resolutions	4,2,1
U-Net Use Spatial Transformer	True
U-Net Learn Sigma	False
U-Net Spatial Context Dim	768
U-Net attention head channels	64
U-Net pred type	v
U-Net norm layer type	GroupNorm
Noise Schedules	Linear
CFG Dropout prob	15%
CLIP Latent Scaling Factor	18.4

certainly empower better performance [14, 23]. We ignore the plucker camera condition for the ShapeNet and FFHQ dataset, over which we find raw RGB input already yields good enough performance.

A.2 Data and Baseline Comparison

Training data. For ShapeNet, following GET3D [8], we use the blender to render the multi-view images from 50 viewpoints for all ShapeNet datasets with foreground mask. Those camera points sample from the upper sphere of a ball with a 1.2 radius. For Objaverse, we use a high-quality subset from the pre-processed rendering from G-buffer Objaverse [15] for experiments. Since G-buffer Objaverse splits the subset into 10 general categories, we use all the 3D instances except from "Poor-quality": Human-Shape, Animals, Daily-Used, Furniture, Buildings&Outdoor, Transportations, Plants, Food and Electronics. The ground truth camera pose, rendered multi-view images and depth maps are used for stage-1 VAE training.

Evaluation. The 2D metrics are calculated between 50k generated images and all available real images. Furthermore, for comparison of the geometrical quality, we sample 4096 points from the surface of 5000 objects and apply the Coverage Score (COV) and Minimum Matching Distance (MMD) using Chamfer Distance

(CD) as follows:

$$
\begin{aligned}
CD(X,Y) &= \sum_{x\in X} \underset{y\in Y}{min}\, ||x-y||_2^2 + \sum_{y\in Y} \underset{x\in X}{min}\, ||x-y||_2^2, \\
COV(S_g,S_r) &= \frac{|\{\arg\min_{Y\in S_r} CD(X,Y) | X\in S_g\}|}{|S_r|}\quad , \\
MMD(S_g,S_r) &= \frac{1}{|S_r|}\sum_{Y\in S_r} \underset{X\in S_g}{min}\, CD(X,Y)
\end{aligned}
\tag{1}
$$

where $X \in S_g$ and $Y \in S_r$ represent the generated shape and reference shape.

Note that we use 5k generated objects S_g and all training shapes S_r to calculate COV and MMD. For fairness, we normalize all point clouds by centering in the original and recalling the extent to [-1,1]. Coverage Score aims to evaluate the diversity of the generated samples, and MMD is used for measuring the quality of the generated samples. 2D metrics are evaluated at a resolution of 128 $\times$ 128. Since the GT data contains intern structures, we only sample the points from the outer surface of the object for results of all methods and ground truth.

For FID/KID evaluation, since different methods have their unique evaluation settings, we standardize this process by re-rendering each baseline's samples using a fixed upper-sphere ellipsoid camera pose trajectory of size 20. With $2.5K$ sampled 3D instances for each method, we recalculate FID@50K/KID@50K, ensuring a fair comparison across all methods.

Details about Baselines. We reproduce EG3D, GET3D, and SSDNeRF on our ShapeNet rendering using their officially released codebases. In the case of RenderDiffusion, we use the code and pre-trained model shared by the author for ShapeNet experiments. Regarding FFHQ dataset, due to the unavailability of the corresponding inference configuration and checkpoint from the authors, we incorporate their unconditional generation and monocular reconstruction results as reported in their paper. For DiffRF, given the absence of the public code, we reproduce their method with Plenoxel [7] and ADM [4].

B More Results

B.1 More Qualitative 3D Generation Results

We include more uncurated samples generated by our method on ShapeNet in Fig. 1, and on FFHQ in Fig. 2. For Objaverse, we include its qualitative evaluation against state-of-the-art generic 3D generative models (Shape-E [11] and Point-E [13]in Fig. 3, along with the quantitative benchmark in Tab. **??** in the main paper. We use CLIP-precision score in DreamField [9] to evaluate the text-3D alignment. As can be seen, LN3Diff shows more geometry and appearance details with higher CLIP scores against Shape-E and Point-E.

B.2 More Monocular 3D Reconstruction Results

We further benchmark the generalization ability of our stage-1 monocular 3D reconstruction VAE. For ShapeNet, we include the quantitative evaluation in

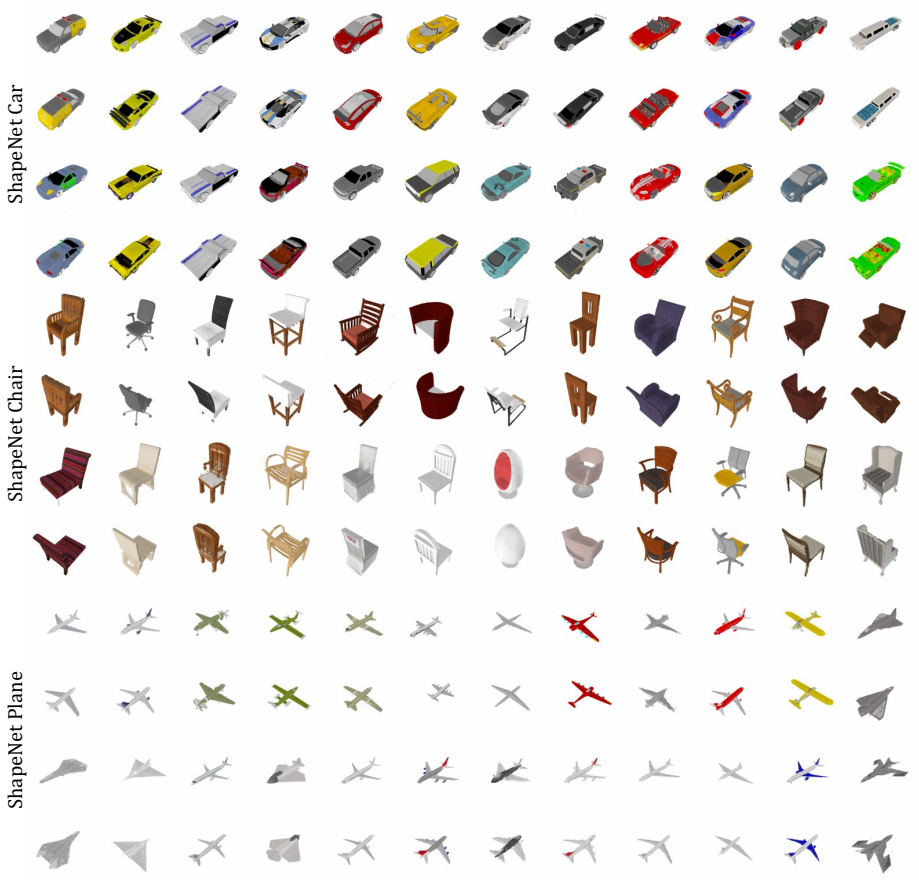

Fig. 1: Unconditional 3D Generation by LN3Diff (Uncurated). We showcase uncurated samples generated by LN3Diff on ShapeNet three categories. We visualize two views for each sample. Better zoom in.

Tab. 2. Our method achieves a comparable performance with monocular 3D reconstruction baselines. Note that strictly saying, our stage-1 VAE shares a similar setting with Pix2NeRF [1], whose encoder also has a latent space for generative modeling. Other reconstruction-specific methods like PixelNeRF [24] do not have these requirements and can leverage some designs like pixel-aligned features and long-skip connections to further boost the reconstruction performance. We include their performance mainly for reference and leave training the stage-1 VAE model with performance comparable with those state-of-the-art 3D reconstruction models for future work.

Besides, we visualize LN3Diff's stage-1 monocular VAE reconstruction performance over our Objaverse split in Fig. 5. As can be seen, though only one view is provided as the input, our monocular VAE reconstruction can

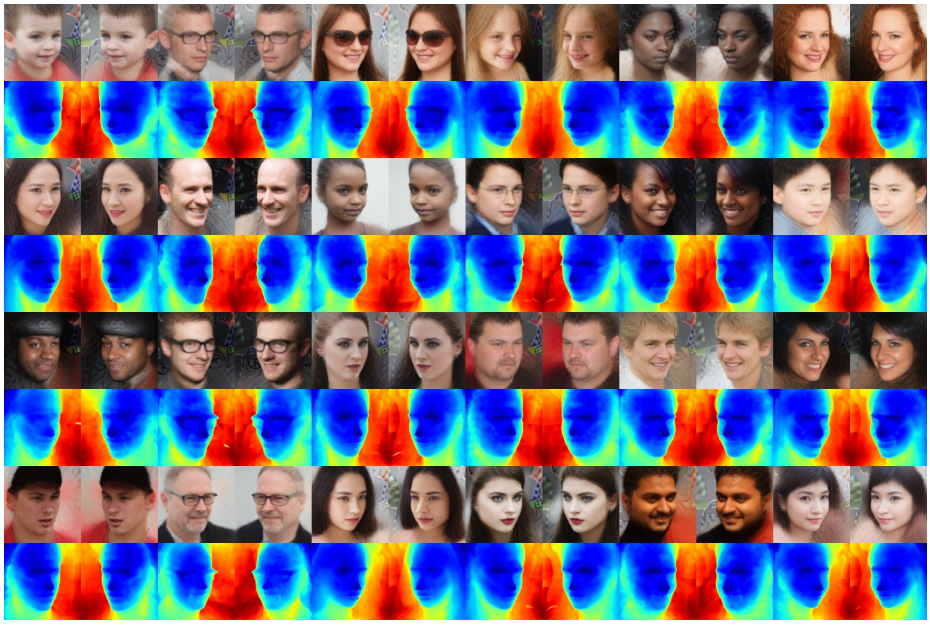

Fig. 2: Unconditional 3D Generation by LN3Diff (Uncurated). We showcase uncurated samples generated by LN3Diff on FFHQ. We visualize two views for each sample along with the extracted depth. Better zoom in.

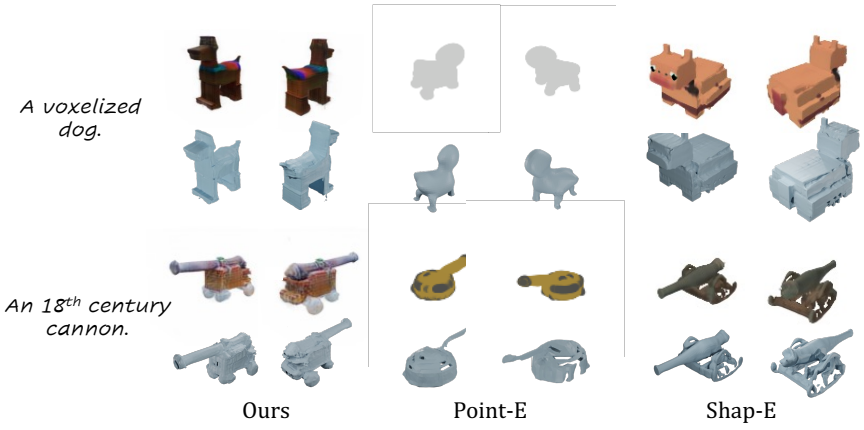

Fig. 3: Qualitative Comparison of Text-to-3D We showcase uncurated samples generated by LN3Diff on ShapeNet three categories. We visualize two views for each sample. Better zoom in.

Table 2: Quantitative results on ShapeNet-SRN [2, 19] chairs evaluate on 128×128. Legend: * – requires test time optimization. Note that our stage-1 VAE shares the same setting only with Pix2NeRF [24], which also has an explicit latent space for generative learning. Other baselines are included for reference.

Method	PSNR ↑	SSIM ↑
GRF [21]	21.25	0.86
TCO [20]	21.27	0.88
dGQN [6]	21.59	0.87
ENR [5]	22.83	-
SRN* [19]	22.89	0.89
CodeNeRF* [10]	22.39	0.87
PixelNeRF [24]	**23.72**	**0.91**
Pix2NeRF [1] conditional	18.14	0.84
Ours	20.91	0.89

Fig. 4: Limitation analysis. We showcase the deficiency to generate composed 3D scenes by LN3Diff. As shown here, the prompt **Two** chair yields similar results with **A** chair.

yield high-quality and view-consistent 3D reconstruction with a detailed depth map. Quantitatively, the novel-view reconstruction performance over our whole Objaverse dataset achieves an average PSNR of 26.14. This demonstrates that our latent space can be treated as a compact proxy for efficient 3D diffusion training.

C Limitation and Failure Cases

We have included a brief discussion of limitations in the main submission. Here we include more details along with the visual failure cases for a more in-depth analysis of LN3Diff's limitations and future improvement directions.

C.1 VAE Limitations

We have demonstrated that using a monocular image as encoder input can achieve high-quality 3D reconstruction. However, we noticed that for some challenging cases with diverse color and geometry details, the monocular encoder leads to blurry artifacts. As labeled in Fig. 5, our method with monocular input may yield floating artifacts over unseen viewpoints. We hypothesize that these artifacts are largely due to the ambiguity of monocular input and the use of regression loss

(L2/LPIPS) during training. These observations demonstrate that switching to a multi-view encoder is necessary for better performance.

Besides, since our VAE requires plucker camera condition as input, the pre-trained VAE method cannot be directly applied to the unposed dataset. However, we believe this is not a research issue at the current time, considering the current methods still perform lower than expected on existing high-quality posed 3D datasets like Objaverse.

C.2 3D Diffusion Limitations

As one of the earliest 3D diffusion models that works on Objaverse, our method still suffers from several limitations that require investigation in the future. **(1)** The support of image-to-3D on Objaverse. Currently, we leverage $\text{CLIP}_{\text{text}}$ encoder with the 77 tokens as the conditional input. However, unlike 2D AIGC with T2I models [16], 3D content creation can be greatly simplified by providing easy-to-get 2D images. An intuitive implementation is by using our ShapeNet 3D diffusion setting, which provides the final normalized CLIP text embeddings as the diffusion condition. However, as shown in the lower half of Fig. 4 in the main submission, the CLIP encoder is better at extracting high-level semantics rather than low-level visual details. Therefore, incorporating more accurate image-conditioned 3D diffusion design like ControlNet [25] to enable monocular 3D reconstruction and control is worth exploring in the future. **(2)** Compositionality. Currently, our method is trained on object-centric dataset with simple captions, so the current model does not support composed 3D generation. For example, the prompt "Two yellow plastic chair with armchests" will still yield one chair, as visualized in Fig. 4. **(3)** UV map. To better integrate the learning-based method into the gaming and movie industry, a high-quality UV texture map is required. A potential solution is to disentangle the learned geometry and texture space and build the connection with UV space through dense correspondences [12].

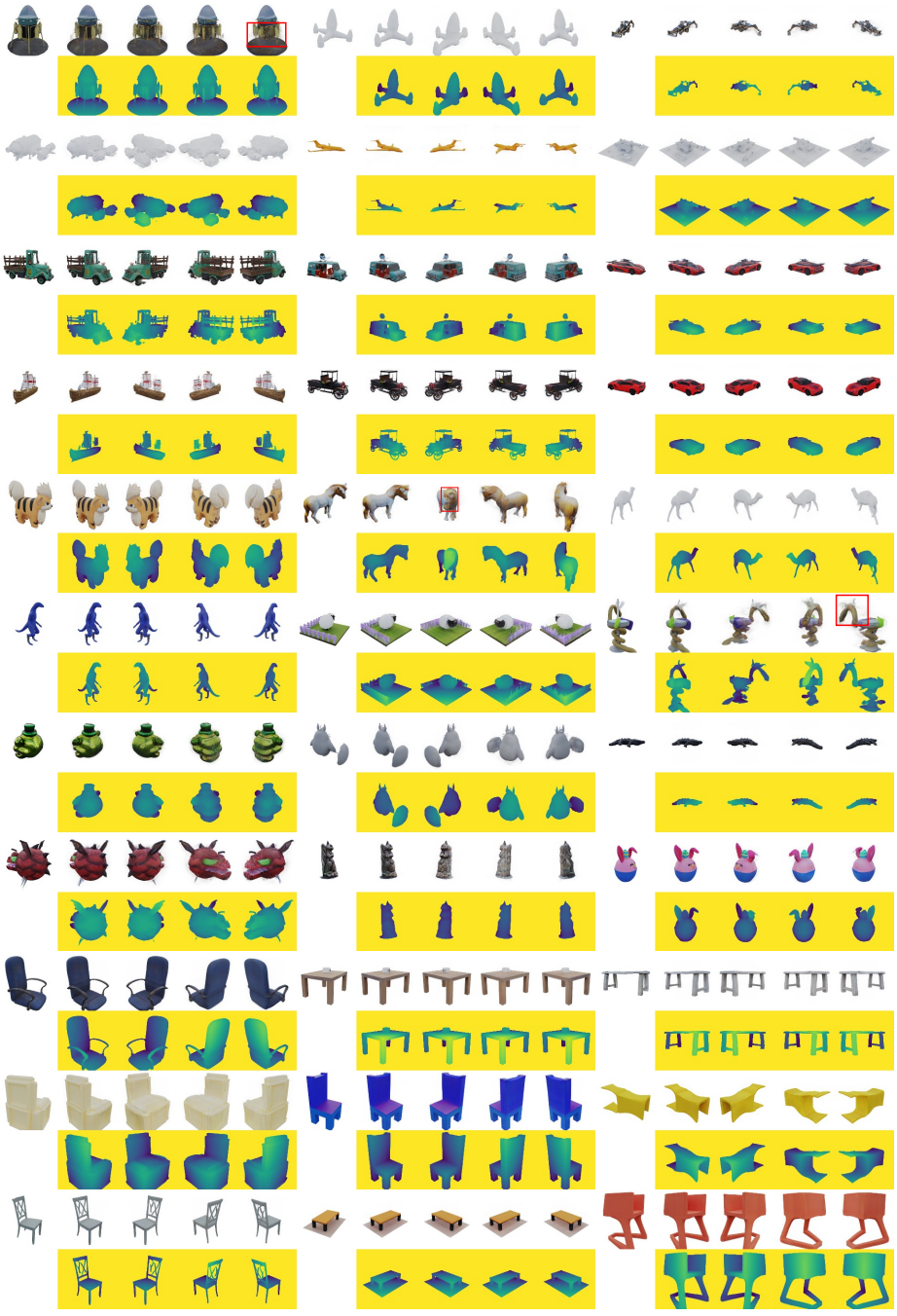

Fig. 5: Monocular 3D Reconstruction by LN3Diff stage-1 VAE on Objaverse (Uncurated). We showcase uncurated samples monocular-reconstructed by LN3Diff on Objaverse. From left to right, we visualize the input image, four reconstructed novel views with the corresponding depth maps. Artifacts are labeled in Red. Better zoom in.

References

1. Cai, S., Obukhov, A., Dai, D., Van Gool, L.: Pix2NeRF: Unsupervised Conditional p-GAN for Single Image to Neural Radiance Fields Translation. In: CVPR (2022) 4, 6
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. arXiv preprint arXiv:1512.03012 (2015) 6
3. Chen, J., Yu, J., Ge, C., Yao, L., Xie, E., Wu, Y., Wang, Z., Kwok, J., Luo, P., Lu, H., Li, Z.: Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis (2023) 1
4. Dhariwal, P., Nichol, A.: Diffusion models beat gans on image synthesis. NeurIPS (2021) 1, 3
5. Dupont, E., Martin, M.B., Colburn, A., Sankar, A., Susskind, J., Shan, Q.: Equivariant neural rendering. In: International Conference on Machine Learning. pp. 2761–2770. PMLR (2020) 6
6. Eslami, S.M.A., Rezende, D.J., Besse, F., Viola, F., Morcos, A.S., Garnelo, M., Ruderman, A., Rusu, A.A., Danihelka, I., Gregor, K., Reichert, D.P., Buesing, L., Weber, T., Vinyals, O., Rosenbaum, D., Rabinowitz, N.C., King, H., Hillier, C., Botvinick, M.M., Wierstra, D., Kavukcuoglu, K., Hassabis, D.: Neural scene representation and rendering. Science **360**, 1204 – 1210 (2018) 6
7. Fridovich-Keil and Yu, Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. In: CVPR (2022) 3
8. Gao, J., Shen, T., Wang, Z., Chen, W., Yin, K., Li, D., Litany, O., Gojcic, Z., Fidler, S.: Get3D: A generative model of high quality 3D textured shapes learned from images. In: NeurIPS (2022) 2
9. Jain, A., Mildenhall, B., Barron, J.T., Abbeel, P., Poole, B.: Zero-shot text-guided object generation with dream fields. In: CVPR (2022) 3
10. Jang, W., Agapito, L.: Codenerf: Disentangled neural radiance fields for object categories. In: ICCV. pp. 12949–12958 (2021) 6
11. Jun, H., Nichol, A.: Shap-E: Generating conditional 3D implicit functions. arXiv preprint arXiv:2305.02463 (2023) 3
12. Lan, Y., Loy, C.C., Dai, B.: DDF: Correspondence distillation from nerf-based gan. IJCV (2022) 7
13. Nichol, A., Jun, H., Dhariwal, P., Mishkin, P., Chen, M.: Point-e: A system for generating 3d point clouds from complex prompts (2022) 3
14. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: ICCV (2023) 1, 2
15. Qiu, L., Chen, G., Gu, X., zuo, Q., Xu, M., Wu, Y., Yuan, W., Dong, Z., Bo, L., Han, X.: Richdreamer: A generalizable normal-depth diffusion model for detail richness in text-to-3d. arXiv preprint arXiv:2311.16918 (2023) 2
16. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022) 1, 7
17. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015) 1
18. Shi, Y., Wang, P., Ye, J., Mai, L., Li, K., Yang, X.: Mvdream: Multi-view diffusion for 3D generation. arXiv:2308.16512 (2023) 1
19. Sitzmann, V., Zollhöfer, M., Wetzstein, G.: Scene Representation Networks: Continuous 3D-structure-aware neural scene representations. In: NeurIPS (2019) 6

20. Tatarchenko, M., Dosovitskiy, A., Brox, T.: Multi-view 3d models from single images with a convolutional network (2016) 6
21. Trevithick, A., Yang, B.: GRF: Learning a general radiance field for 3D scene representation and rendering. In: ICCV (2021) 6
22. Vahdat, A., Kreis, K., Kautz, J.: Score-based generative modeling in latent space. In: NeurIPS (2021) 1
23. Xu, Y., Tan, H., Luan, F., Bi, S., Wang, P., Li, J., Shi, Z., Sunkavalli, K., Wetzstein, G., Xu, Z., Zhang, K.: DMV3D: Denoising multi-view diffusion using 3D large reconstruction model. In: ICLR (2024) 2
24. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: PixelNeRF: Neural radiance fields from one or few images. In: CVPR (2021) 4, 6
25. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: ICCV (2023) 7