DiffRF: Rendering-Guided 3D Radiance Field Diffusion - Supplementary Document -

Norman Müller[1,2] Yawar Siddiqui[1,2] Lorenzo Porzi[2] Lorenzo Porzi[2] Samuel Rota Bulò[2]
Peter Kontschieder[2] Matthias Nießner[1]

Technical University of Munich[1] Meta Reality Labs Zurich[2]

Appendix

In this supplementary document, we discuss additional details about our method, the data used for training and evaluation, and show further qualitative results. We also refer to our for a comprehensive overview with further qualitative results.

A. Additional qualitative results

We provide additional qualitative results on PhotoShape Chairs [6] in Fig. 1 as well as on ABO Tables [3] in Fig. 2. Furthermore, we provide a qualitative comparison of our method when removing the rendering loss in Fig. 3 and Fig. 4.

B. Implementation detail

Architecture We base our architecture on 2D U-Net structure from [4]: For this, we replace the 2D convolutions in the ResNet and attention blocks with corresponding 3D convolutions, preserving kernel sizes and strides. Furthermore, we use 3D average pooling layers instead of 2D in the downsampling steps. Our U-Net consists of 4 scaling blocks with two ResNet blocks per scale, where we linearly increase the initial feature channel dimension of 64 to 256. We use skip attention blocks at the scaling factors 2, 4, and 8 with 32 channels per head.

Training details We train all models with a batch size of 8 and use the Adam optimizer with an initial learning of 10^{-4}. We apply a linear beta scheduling from 0.0015 to 0.05 at 1000 timesteps. From 4 random training views at a resolution of 128×128, we sample 8192 random pixels for the rendering supervision (with 92 z-steps for volumetric rendering) and weight the rendering loss with $\omega_t = \bar{\alpha}_t^2$. We train for 3.0m iterations with a decaying LR scheduling for 10^{-4} to 10^{-6} at a voxel grid resolution of 32 on 2 GPUs on every data set.

Sampling time We perform DDPM sampling for 1000 iterations leading to a run time of 48.6s per sample on an NVIDIA RTX 2080 TI. Once synthesized, our explicit representation enables rendering at 128×128 resolution with over 380 FPS.

C. Data

Radiance Field Generation For PhotoShape Chairs, we render the provided 15,576 chairs using Blender Cycles from 200 views on an Archimedean spiral at a fixed radius of 2.5 units with pitch starting from $-20°$ to $60°$. For ABO Tables, we use the provided 91 renderings with 2-3 different environment map settings per object, resulting in 1676 tables. For PhotoShape Chairs, we hold out 10% of the samples for testing based on shape ids selected randomly, whereas for ABO Tables, we use the official data split. We fit explicit voxel grids at a resolution of 32^3 using volumetric rendering with spherical harmonics of degree 2 for an initial fit. We then fine-tune our representations for spherical harmonics of degree 0, which we found to lead to sharper geometry compared to directly optimizing density and color features. We furthermore bound the feature space to $[-1, 1]$ which we found to stabilize the sampling process noticeably affecting the rendering quality.

Evaluation For image quality evaluation, we calculate FID and IS by sampling 10k views by rendering 1000 samples from 10 random views at a resolution of 128×128. We follow [10] and evaluate the geometric quality by computing the Coverage Score (COV) and Minimum Matching Distance (MMD) using Chamfer Distance (CD)

$$\mathrm{CD}(X,Y) = \sum_{x \in X} \min_{y \in Y} ||x-y||_2^2 + \sum_{y \in Y} \min_{x \in X} ||x-y||_2^2,$$

$$\mathrm{COV}(S_g, S_r) = \frac{|\{\arg\min_{Y \in S_r} CD(X,Y) | X \in S_g\}|}{|S_r|},$$

$$\mathrm{MMD}(S_g, S_r) = \frac{1}{|S_r|} \sum_{Y \in S_r} \min_{X \in S_g} CD(X,Y),$$

on a reference set S_r (the test samples) and a generated set S_g twice as large as the reference set. We extract meshes using marching cubes [5] and sample 2048 points on the faces.

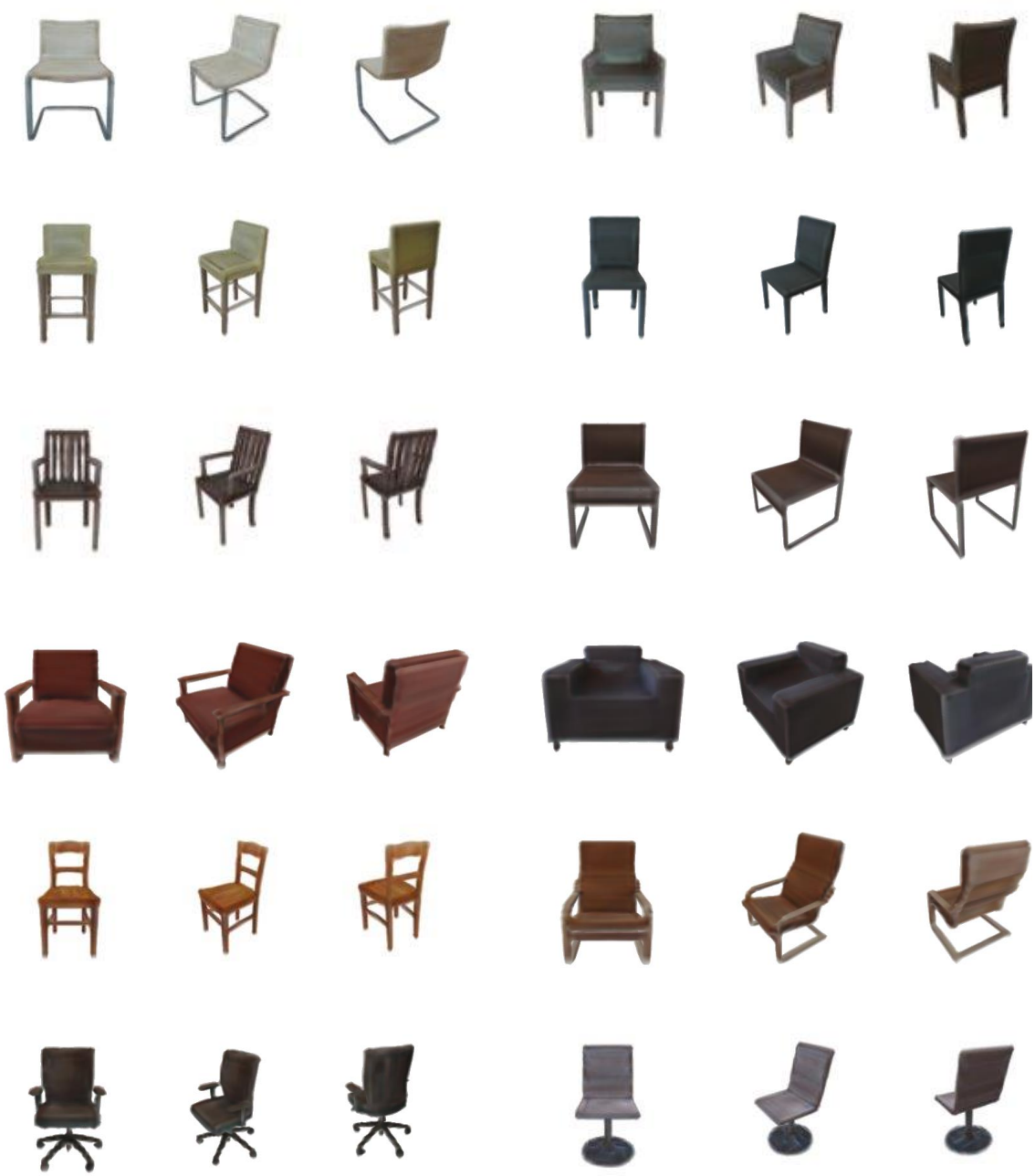

Figure 1. Additional qualitative sampling results on PhotoShape Chairs [6].

To account for potentially different scaling of the samples produced by the 3D-aware GAN models, we normalize all point clouds by centering in the origin and an-isotropic scaling of the extent to $[-1, 1]$.

For evaluation of the masked radiance field completion, we additionally compute a masked peak signal-to-noise ratio (mPSNR): Given a binary mask m of the input radiance field f^{in}, we compute for each corresponding input image the non-masked area by depth-based projection into the image plane using depth estimated from the input radiance field. We then compute the mPSNR by averaging the PSNR on the non-masked pixels for all evaluation views (we choose 10 views randomly).

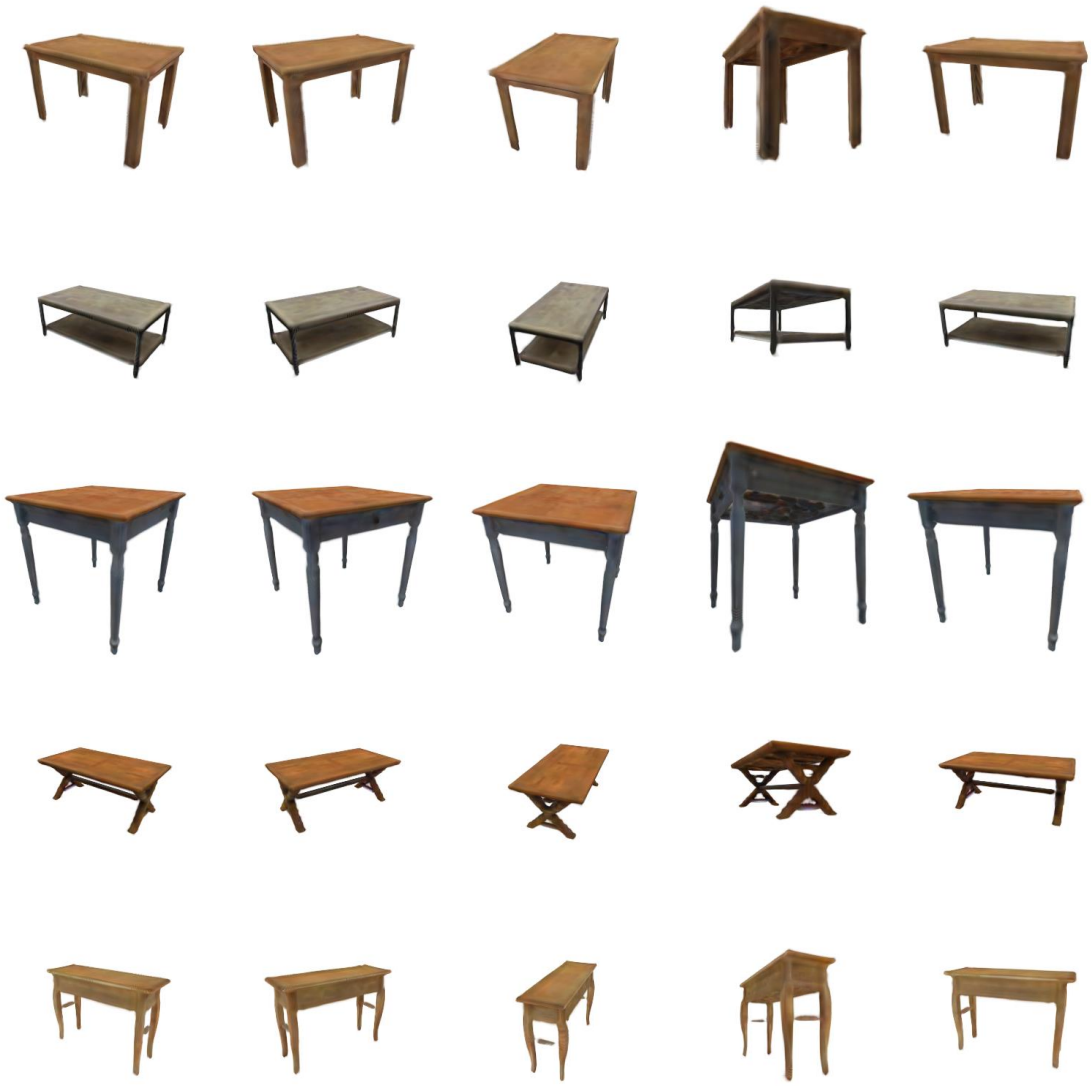

Figure 2. Additional qualitative sampling results on ABO Tables [3].

D. Conditional sampling

Masked completion Since the generator of EG3D [1] is trained via 2D discriminator guidance, we perform 3D masked completion via GAN inversion. For this, we start from a random initial latent code and repeat the following steps for 200 iterations on each masked sample: We render the current synthesized sample from 8 views and project the 3D input mask onto the synthesized views using the predicted depths. On the remaining non-masked regions, we compute the photometric error with the input images. We use the Adam optimizer with a learning rate of 10^{-2} with a small L_2 regularization term on the code (weighted with 5×10^{-2}) in order to update the latent code.

Image-to-Volume Synthesis Given a posed and segmented image, we condition our trained radiance field diffusion model by steering the sampling processing similar to the Classifier Guidance formulation from [4]: During sampling time, for each time step t, we gradually update the predicted denoised field $\tilde{f}_0^t$ towards minimizing the photometric error obtained from comparing the rendering $\tilde{I}_t$ from a given pose with the foreground-masked target image I. For this, we compute the gradient $\nabla_{\tilde{f}_0^t}(\tilde{I}_t, I)$ on the current denoising estimate by volumetric rendering and steer the

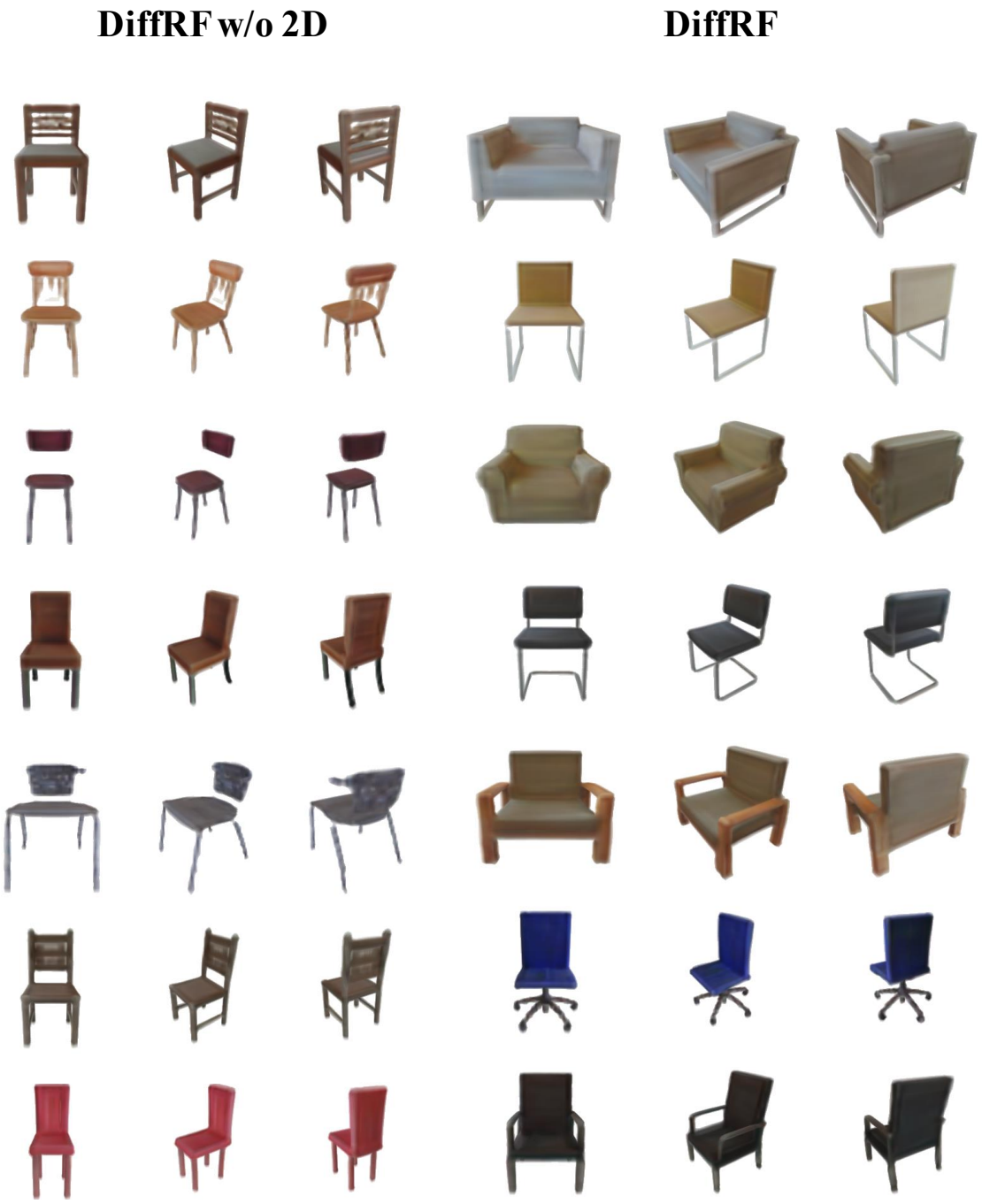

Figure 3. Qualitative comparison on PhotoShape Chairs [6] when removing the 2D rendering loss.

sampling process by $\tilde{f}_0^t \leftarrow \tilde{f}_0^t - \lambda \nabla_{\tilde{f}_0^t}(\tilde{I}_t, I)$ with a small guidance weight λ.

E. CLIP conditioning

Following related work [2, 9, 11], we additionally augment our model to condition on embeddings derived from text or single-image encodings obtained from CLIP ViT-B/32 [7] using cross-attention layers. For training, we use random single training views encoded by the frozen CLIP model to condition the denoiser. Here, we adapt the cross-attention mechanism from [4] for the 3D U-Net and do not train the image encoder in order to preserve the image-text-correspondence of CLIP. We show examples on

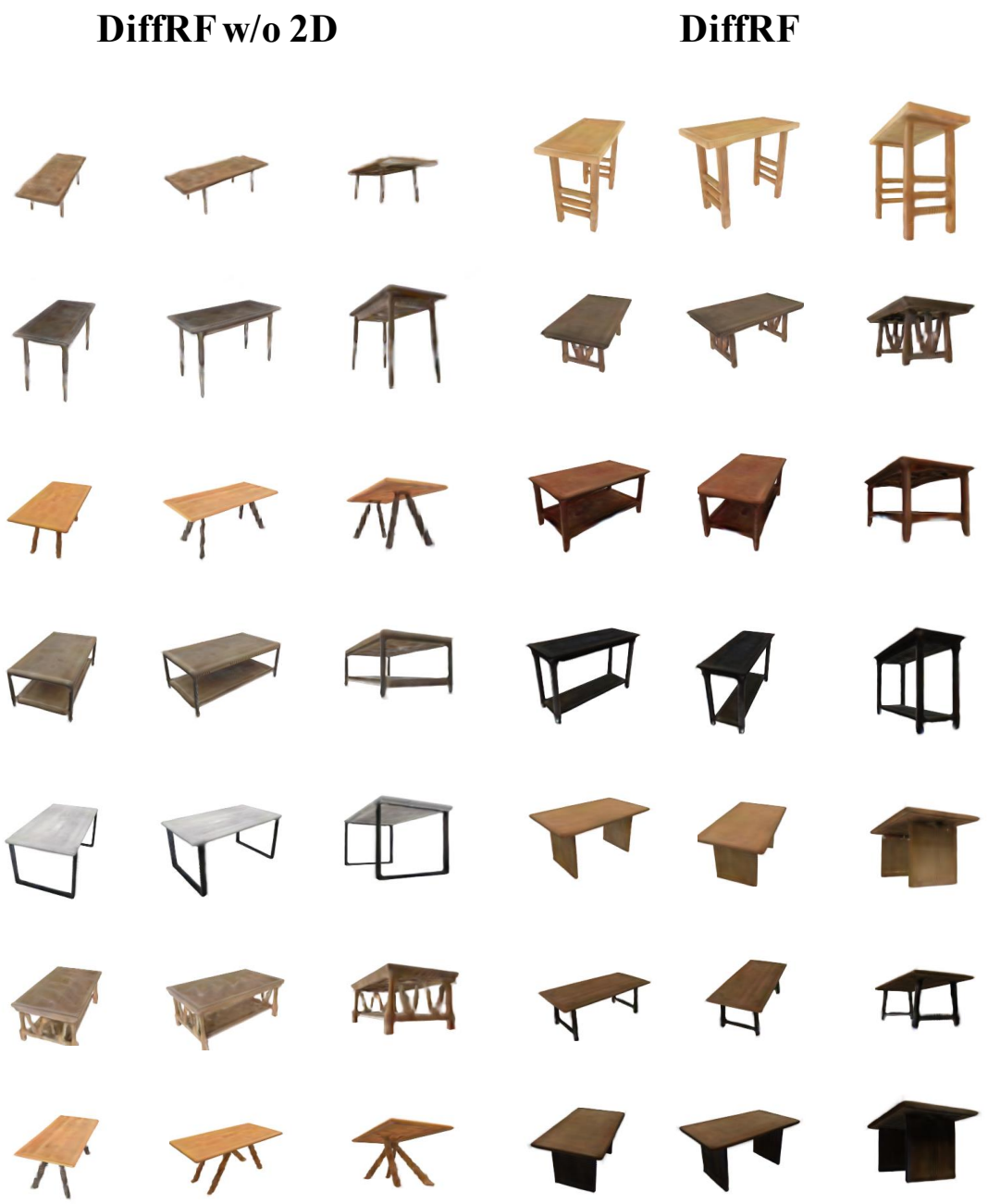

Figure 4. Qualitative comparison on ABO Tables [3] when removing the 2D rendering loss.

single-image PhotoShape samples in Fig. 7 as well as on real-world image from the Pix3D dataset [8] in Fig. 6.

As these codes have strong correspondences to text samples by design, we can guide the sampling process by text prompts, examples are shown in Fig. 5 without the need for training on text-radiance field pairs.

References

[1] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *CVPR*, 2022. 3

[2] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tuyakov, Alex

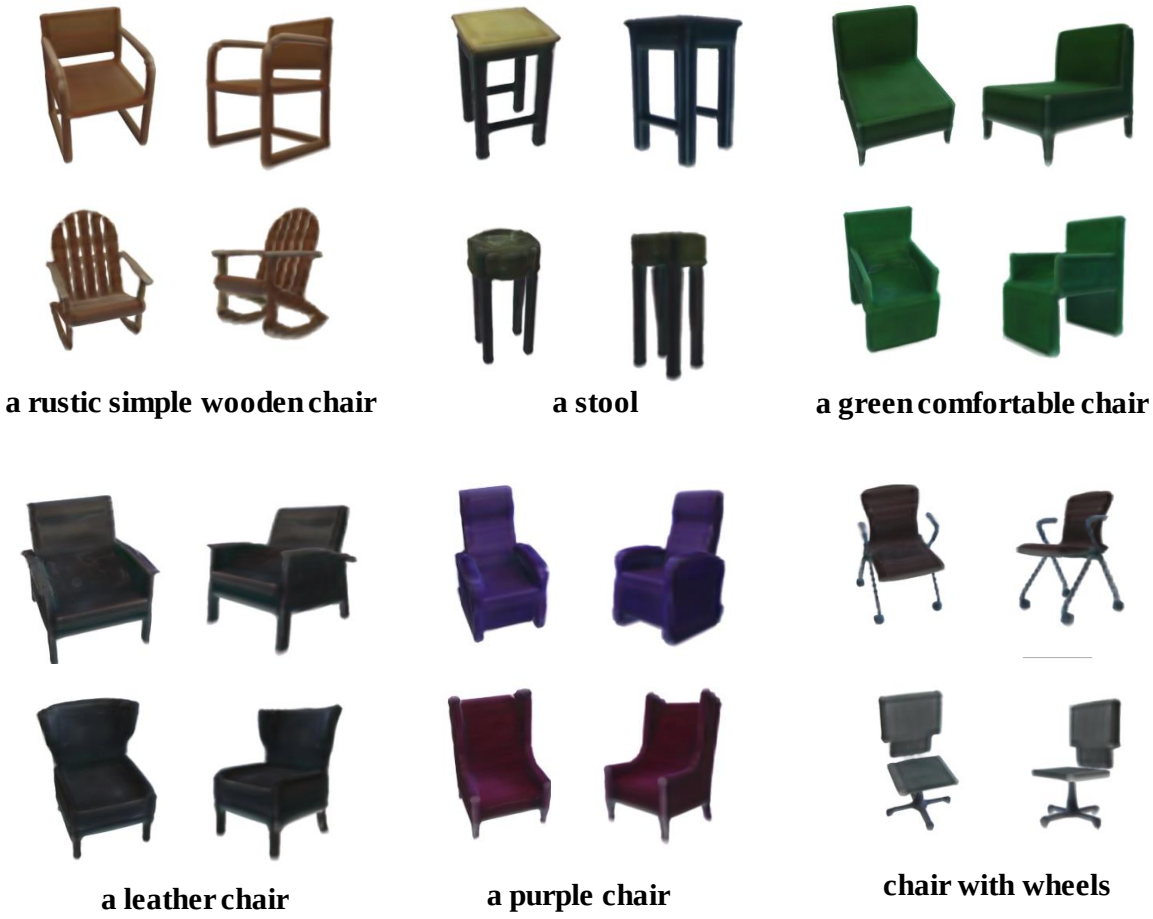

Figure 5. Text-conditional inference using CLIP-embeddings trained on PhotoShape Chairs [6].

Schwing, and Liangyan Gui. SDFusion: Multimodal 3d shape completion, reconstruction, and generation. *arXiv*, 2022. 4

[3] Jasmine Collins, Shubham Goel, Kenan Deng, Achleshwar Luthra, Leon Xu, Erhan Gundogdu, Xi Zhang, Tomas F Yago Vicente, Thomas Dideriksen, Himanshu Arora, et al. Abo: Dataset and benchmarks for real-world 3d object understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21126–21136, 2022. 1, 3, 5

[4] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 1, 3, 4

[5] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987. 1

[6] Keunhong Park, Konstantinos Rematas, Ali Farhadi, and Steven M. Seitz. Photoshape: Photorealistic materials for large-scale shape collections. *ACM Trans. Graph.*, 37(6), Nov. 2018. 1, 2, 4, 6, 8

[7] 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. 4

[8] Xingyuan Sun, Jiajun Wu, Xiuming Zhang, Zhoutong Zhang, Chengkai Zhang, Tianfan Xue, Joshua B Tenenbaum, and William T Freeman. Pix3d: Dataset and methods for single-image 3d shape modeling. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2974–2983, 2018. 5, 7

[9] Tengfei Wang, Bo Zhang, Ting Zhang, Shuyang Gu, Jianmin Bao, Tadas Baltrusaitis, Jingjing Shen, Dong Chen, Fang Wen, Qifeng Chen, et al. Rodin: A generative model for sculpting 3d digital avatars using diffusion. *arXiv preprint arXiv:2212.06135*, 2022. 4

[10] Guandao Yang, Xun Huang, Zekun Hao, Ming-Yu Liu, Serge Belongie, and Bharath Hariharan. Pointflow: 3d point cloud generation with continuous normalizing flows. In *Proceedings of the IEEE/CVF International Conference on Com-*

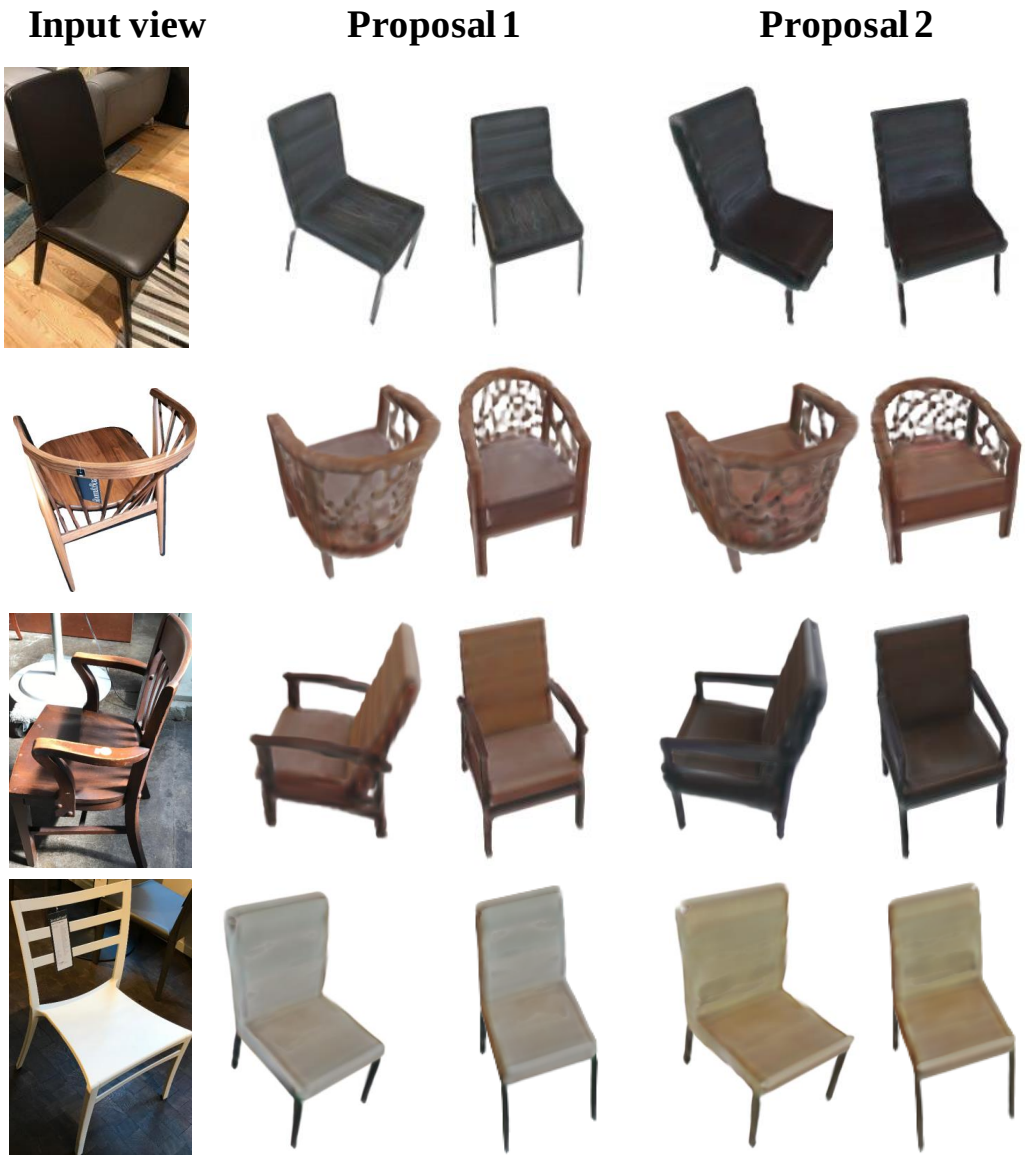

Figure 6. Image-conditional inference using CLIP-embeddings on Pix3D [8] images.

puter Vision, pages 4541–4550, 2019. 1

[11] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. *arXiv preprint arXiv:2210.06978*, 2022. 4

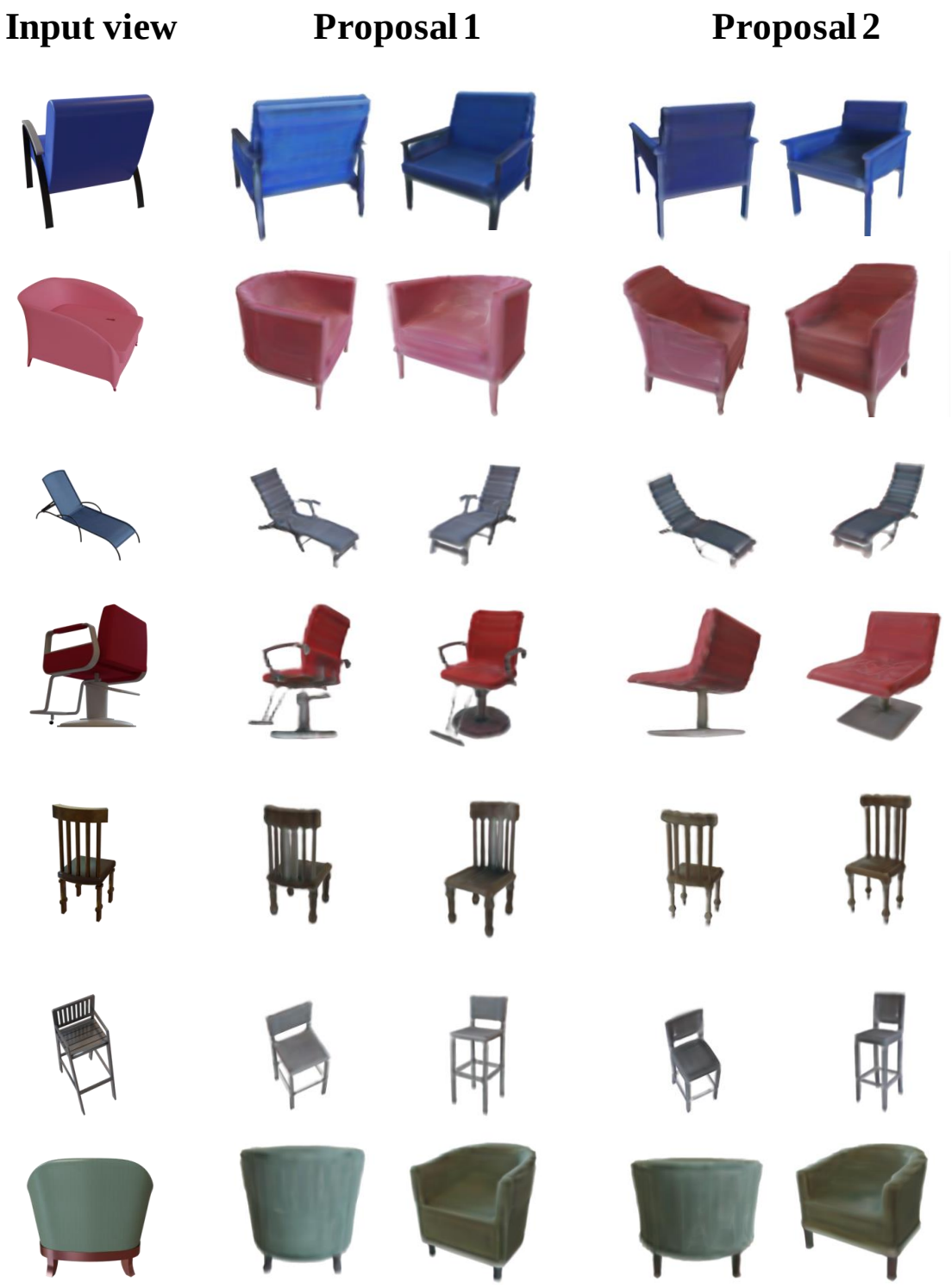

Figure 7. Image-conditional inference using single-view CLIP-embeddings on PhotoShape Chairs [6].