UVGS: Reimagining Unstructured 3D Gaussian Splatting using UV Mapping

Supplementary Material

Our supplementary material contains a wide range of information that cover implementation details for our networks and training procedures, as well as a large variety of qualitative results.

Supplementary Video: We refer the interested reader to the supplementary video where we provide an overview of how our proposed approach works as well as a plethora of qualitative results across different tasks.

1. Spherical Mapping

Spherical Mapping: Spherical mapping [43] is a fundamental technique in computer graphics that is used to project 3D meshes onto a 2D map generally for texture mapping, where a 2D image is wrapped around a 3D object, such as a cylinder or a sphere. However, cylindrical mapping fails to capture the top and bottom parts of the object in the same UV map, and can introduce distortions for objects that extend far in the Z-direction. Hence we opted for spherical mapping the process of which involves converting 3D Cartesian coordinates (x, y, z) into spherical coordinates (ρ, θ, ϕ) and then mapping these onto a 2D plane. Algorithm [1] explains spherical unwrapping in detail for a single layer(K=1). The same process can be repeated for multiple layers, by keeping a track of opacity values.

Thresholding Opacity 3DGS use multiple points with varying opacity values to represent an object from any specific viewpoint. However, it is oftentimes noticed that many of these points have very low opacity values and do not contribute to the object's overall representation or appearance. We filter these points using a threshold opacity value with no impact on the object's overall geometry and representation to reduce the number of tractable primitives.

Dynamic GS Selection and Multiple Layers When projecting 3DGS points to UV maps using spherical mapping, multiple points may map to the same pixel in UV space as shown in Fig. 3. The two 3DGS points (g_1) and (g_2) map to the same pixel on UV map (P_a) causing many-to-one mapping. However, the UV map can only hold a single 3DGS primitive at any given pixel. To address this, we propose a Dynamic Selection approach where each UV pixel retains the 3DGS attributes with the highest opacity intersecting the same ray from the centroid to the farthest 3DGS primitive along the ray. Using the same example in Fig. 3, if opacity o_1 of Gaussian g_1 is less than opacity o_2 of g_2. then only g_2 attributes will be stored in the UV map at pixel P_a. Through multiple testing, we observed that this method helps retain the overall geometry and appearance of the 3DGS object while resolving many-to-one mapping issues with minimal quality loss.

This method with single layer is applicable to most of the objects in our dataset. However, this might fail in the case of more complex objects or real-world scene representation. There could be multiple layers of Gaussians holding higher opacity and contributing to the overall scene's appearance or geometry, and even partial or full occlusions. To better represent such objects and scenes and to prove the effectiveness of UVGS, we stack multiple layers of UV maps, where each UVGS layer holds the 3DGS primitives of the top-K^{th} opacity value intersecting the same ray. This can be accomplished by inscribing the 3DGS object inside multiple spheres where each sphere maps the 3DGS attribute corresponding to the top-K^{th} opacity value along the same ray. To show the effectiveness of proposed UVGS maps in capturing the intricacies of a complex real-world scene, we use a 12 layer UVGS map to reconstruct the real-world 3D scenes. The results are presented in Fig. 8. We also com-

Algorithm 1 Spherical Unwrapping for UVGS map (K=1).

Require: $3DGS \in \mathbb{R}^{n\times 14},\ (M, N) \in \mathbb{Z},\ K = 1$
Ensure: $position(\sigma), color(c), scale(s) \in \mathbb{R}^{n\times 3}$
Ensure: $rotation(r) \in \mathbb{R}^{n\times 4},\ opacity(o) \in \mathbb{R}^{n\times 1}$
Extract $xyz(\sigma),\ opac(o)$ from $3DGS$
Spherical radius, $r \leftarrow \sqrt{x^2 + y^2 + z^2}$
Azimuthal Angle, $\theta \leftarrow tan^{-1}(y, x)$
Polar Angle, $\phi \leftarrow cos^{-1}(z, r)$
$(\theta,\ \phi) \leftarrow (\text{deg}(\theta) + 180,\ \text{deg}(\phi))$
$\theta_{UV} \leftarrow \text{round}((\theta/360) \times M)$
$\phi_{UV} \leftarrow \text{round}((\phi/180) \times N)$
Initialize $UV_{map} \leftarrow \text{zeros}(M, N, 14)$
Initialize $UV_{opac} \leftarrow \text{zeros}(height, width)$
for all (t, P, xyz, o) in $(\theta_{UV}, \phi_{UV}, 3DGS, opac)$ **do**
  if $0 \le P < height$ and $0 \le t < width$ **then**
    if $UV_{map}[P, t]$ is 0 **then**
      $UV_{map}[P, t] \leftarrow 3DGS[ind]$
      $UV_{opac}[P, t] \leftarrow o$
    else
      if $o > UV_{opac}[P, t]$ **then**
        $UV_{map}[P, t] \leftarrow 3DGS[ind]$
      end if
    end if
  end if
end for
return UV_{map}

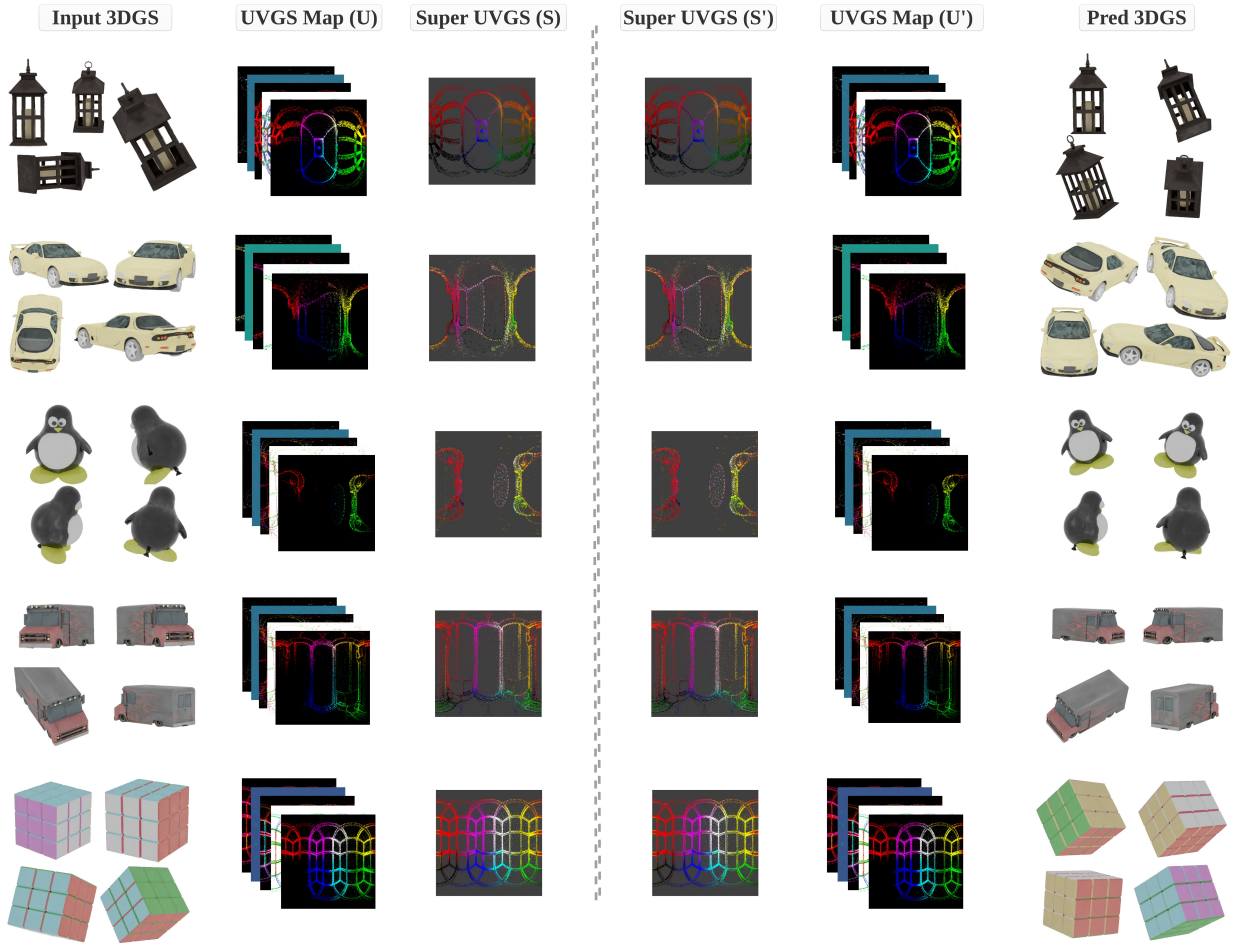

Figure 1. In this figure, we show the qualitative results of reconstructing 3DGS object using pretrained Image Autoencoder (A) via Super UVGS. We obtain UVGS maps (U) through spherical projection of 3DGS objects, followed by using forward mapping network to get Super UVGS (S). A pretrained AE is used to reconstruct Super UVGS (S'), which can be converted to UVGS maps (U') through inverse mapping network. At last, through inverse spherical mapping, we can get predicted 3DGS object which has the same appearance and geometry as the input object with minimal loss.

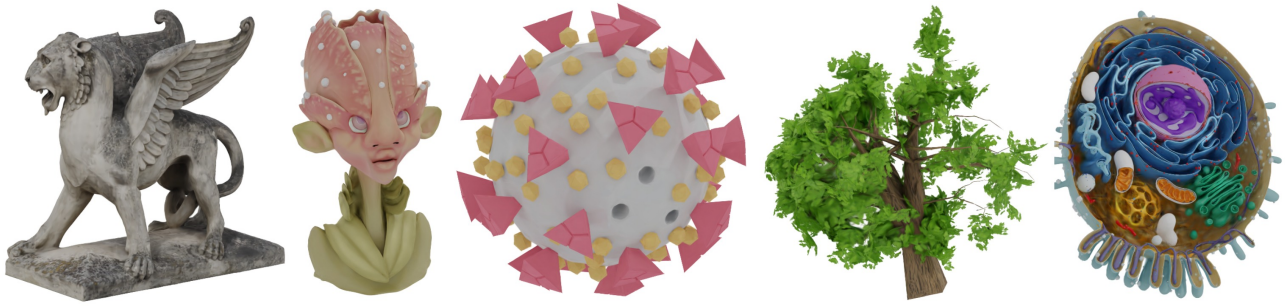

Figure 2. Complex object reconstructions (K=4) using pretrained image-based autoencoder.

pare the effect of increasing the number of UVGS layers in representing a real-world 3D scene in Fig. 3 In future work, we want to extend this ability for potentially many applications in 3D dynamic scene reconstructions using video diffusion models, and the segmentation or tracking of objects in 3DGS scenes as the features in the UVGS maps can be easily processed with the neural networks and tracked over time.

2. Mapping Networks

Forward Mapping Details: This process is defined as:

$$f^f_{map} = [\,[\phi^f_P(\sigma)]\,[\phi^f_T([r,s])]\,[\phi^f_A[o,c]]\,] \tag{3}$$

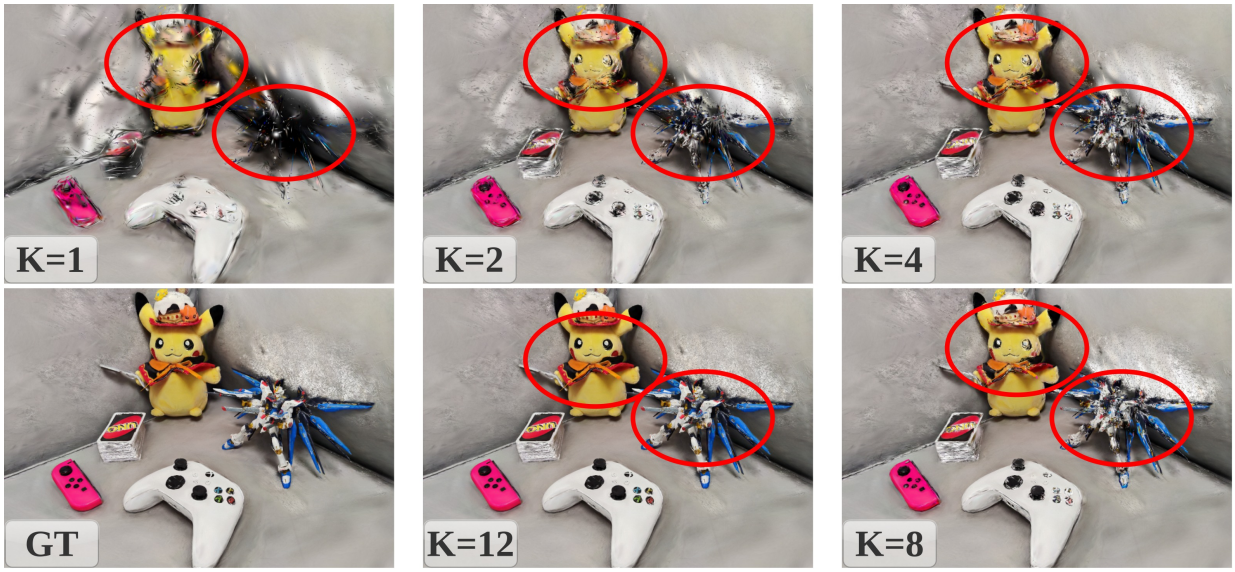

Figure 3. Reconstruction of a real-world scene for different K values. Smaller K results in many-to-one issue, hence lacking details.

The central branch (ϕ_C^f) is composed of $2L$ hidden Convolution layers. The first L hidden convolution layers increase the feature dimension at each step, while the last L layers does the inverse and squeezes the high-dimensional feature maps to 3 channels to output Super UVGS image $S \in \mathbb{R}^{M \times N \times 3}$.

$$S = tanh(\ \phi_C^f[f_{map}^f]\ ) \ \in \ \mathbb{R}^{(H,W,3)} \tag{4}$$

Each CNN layer is followed by a batch normalization layer and ReLU activation both in multi-branch and central branch modules. The last layer of central branch is activated using $tanh$ to ensure the Super UVGS doesn't take any ambiguous value resulting in gradient explosion or undesired artifacts. The obtained Super UVGS S representation squeezes all the 3DGS attributes to a 3 dimensional image while also maintaining local and global structural correspondence among them.

Inverse Mapping: The first L layers in the Central branch increases the feature dimension and the last L layers reduces them to obtain a combined feature map.

$$f_{map}^i = \phi_C^i(S)$$

The final layer is a set of 3 branches projecting the features to position, translation, and appearance attributes, respectively.

$$f_\sigma^i = [\phi_P^i(f_{map}^i)]$$
$$f_{r,s}^i = [\phi_T^i(f_{map}^i)]$$
$$f_{o,c}^i = [\phi_A^i(f_{map}^i)]$$

Similar to the forward mapping network, each layer in the central branch and attribute specific branches is followed by batch normalization and $Relu(.)$ activation. The last set of branch layers are activated using $tanh(.)$ to prevent ambiguous values resulting in gradient explosion or reconstruction artifacts.

$$\hat{U} = tanh(\ [[f_\sigma^i]\ [f_{r,s}^i]\ [f_{o,c}^i]]\ )$$

Losses Details: We used MSE to focus on pixel-wise difference during the training. We solely used MSE for a few iterations to make the mapping networks learn the overall structural representation of the UVGS map using:

$$\mathcal{L}_{mse} = \frac{1}{n}\sum_{i=1}^{n}(U_i - \hat{U}_i)^2. \tag{5}$$

After training the model for few iterations using MSE, we introduce the LPIPS loss giving same weight to both MSE and LPIPS over a few iterations. We observed that increasing the weight value of LPIPS over the iterations resulted in better and faster convergence results.

$$\mathcal{L}_{lpips} = \sum_l w_l \left\| \phi_l(x) - \phi_l(y) \right\|^2, \tag{6}$$

where $\phi_l(x)$ and $\phi_l(y)$ are feature maps extracted from pretrained layers of AlexNet[64].

Mapping Training Details: Before training the models, we normalized the different attributes in UVGS to $[-1, 1]$ using the same normalization functions as used in 3DGS paper[19]. The normalized UVGS maps are used to train the multi-branch forward and reverse mapping networks using MSE and LPIPS loss. We trained the mapping networks on $8 \times A100\ (80GB)$ GPUs with a Batch Size of 96 for 120 hours using Adam optimizer with a learning rate of $6e-5$ and set $\beta_1 = 0.5$ and $\beta_2 = 0.9$ with weight decay of 0.01.

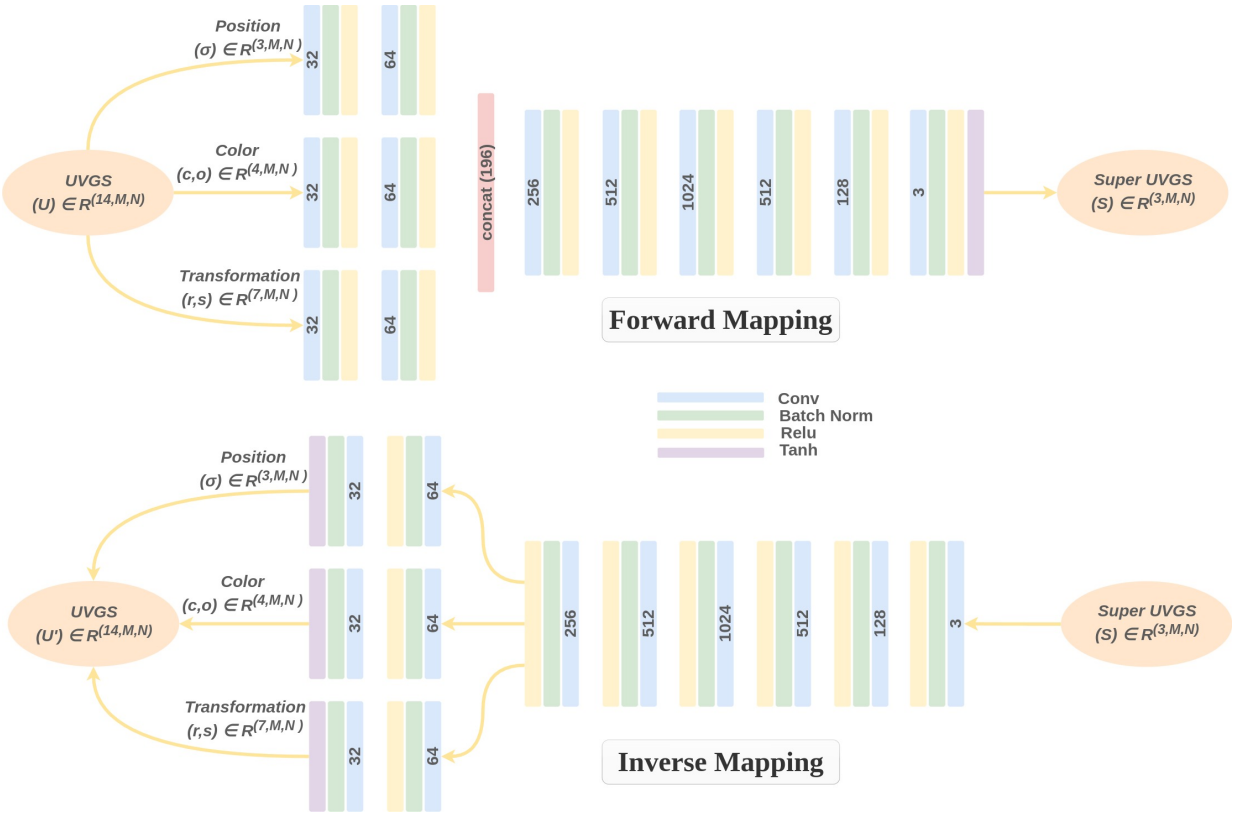

Figure 4. Forward Mapping Network for UVGS to Super UVGS mapping. The inverse mapping network follows just the inverse of this architecture with each attribute-specific branch now followed by $tanh()$ at the end.

We set the λ for LPIPS loss to be 0 for the first 24 hours of training and gradually increased it from 1 to 10 for the remaining training in a step of 1.

2.1. Interpolation with UVGS

We show that the proposed SuperUVGS representation can be used to perform local editing and interpolation directly in the UV domain. We can perform edits like swapping the parts of one object from the other, cropping the 3D object, or merging two objects together simply with the SuperUVGS images without any learning based method. The results are demonstrated in Fig 5.

3. LDM - Unconditional and Conditional Generation

Caption Generation To generate the relevant text captions for the objects in our dataset for conditional generation, we leverage CLIP [37], BLIP2 [21], and GPT4 [1] very similar to [25]. Specifically, we use BLIP2 to generate N different captions for randomly selected 20 views from the 88 rendered views for each object in the dataset. CLIP encoders are used to encode and calculate the cosine similarity between the N generated caption per view and the corresponding 20 views. The caption with max similarity is assigned to that particular view, resulting in 20 different captions for the same object. We now use GPT4 to extract a single caption distilling all the given 20 descriptions. We found that the resulting captions were very appropriate to the input objects, and thus we directly used them for conditional generation.

Figure 5. Linear interpolation between two 3DGS objects using SuperUVGS representation.

LDMs [17, 41] use pretrained VAEs [12] to convert the original image $x \in R^{H\times W\times 3}$ into a compact latent representation $z \in R^{h\times w\times c}$, where the forward and reverse diffusion processes are applied [41]. The VAE decoder then converts the compact latent representation back to pixels.

Table 4. We compare the FID and KID of unconditional generation using the current SOTA methods on 20K randomly generated samples from each method and ours. We also compare our method against SOTA text-conditioned generation frameworks on CLIP Score for 10K generated objects from each method.

Unconditional Generation			**Text-Conditioned Generation**	
Method	**FID ↓**	**KID ↓**	**Method**	**CLIP Score ↑**
Get3D [14]	53.17	4.19	DreamGaussian [48]	28.51
DiffTF [3]	84.57	8.73	Shap. E [7]	30.53
EG3D [4]	74.51	6.62	LGM [50]	30.74
GaussianCube	34.67	3.72	GaussianCube [61]	30.34
UVGS (Ours)	**26.20**	**3.24**	**UVGS (Ours)**	**32.62**

The objective function in latent diffusion model can be written as:

$$\mathbb{L}_{LDM} := \mathbb{E}_{\epsilon(x),\epsilon\sim\mathbb{N}(0,1),t}[||\epsilon - \epsilon_\theta(z_t, t)||_2^2] \quad (7)$$

where, $\mathbb{N}(0,1)$ is the Normal distribution, and t is the number of time steps and z_t is the noisy sample after t time steps.

Training was done using AdamW optimizer with a learning rate of $1e-4$ for 75 epochs on $8 \times A100\ (80GB)$ GPUs.

Once trained, we can randomly sample new high-quality 3DGS assets from the learned generative model.

To allow generation of objects from text, we also trained a conditional LDM, where we use Stable Diffusion (SD) [41] pipeline as it can use text prompt conditioning to guide the image generation through cross-attention. Similar to unconditional LDM, we use pretrained SD's VAE for mapping the Super UVGS image to a latent space and back to the reconstructed Super UVGS. The text prompts are given to a pretrained CLIP [37] text encoder to generate a text embedding $c_t \in \mathbb{R}^{77\times768}$, which is then passed to the UNet encoder of SD for cross-attention. We used a set of CLIP encoder and BLIP2 [21], and GPT4 [1] to generate captions for our dataset. The overall objective function for conditional LDM now becomes:

$$\mathbb{L}^C_{LDM} := \mathbb{E}_{\epsilon(x),\epsilon\sim\mathbb{N}(0,1),t,c_t}[||\epsilon - \epsilon_\theta(z_t, t, c_t)||_2^2] \quad (8)$$

where, $\epsilon_\theta(\cdot, t)$ is a time-conditional U-Net [42] model, $\mathbb{N}(0,1)$ is the Normal distribution, z_t is the latent code, and c_t is the text embedding. Training was done using AdamW optimizer with a learning rate of $1e-4$ for 50 epochs on $8 \times A100\ (80GB)$ GPUs. Once trained, this conditional LDM can use used to generate text-conditioned Super UVGS images, which can later be mapped to high-quality 3DGS objects.

4. Comparison with Baselines

We compare the generational capabilities of our method against various conditional and unconditional SOTA 3D object generation method on ShapeNet-cars dataset. Specifically, we used the methods using multiview rendering for optimization, like DiffTF [3] and Get3D [14]. We also compared our approach again the current SOTA methods trying to give structural representation to Gaussians, including GaussianCube [61] and TriplaneGaussian [69]. We also compared against general purpose SOTA large 3D content generation models like DreamGaussian [48], LGM [50], and EG3D [4].

To compare the quality of our generation results, as a standard practice, we use FID and KID for unconditional generation, and Clip Score for text-conditioned generation. Table 2 quantitatively compares the unconditional and conditional generation results of our method again various SOTA methods. From this table, it can be seen that our method performs a good job in unconditional generation of good quality 3D assets. The main reason behind this is the learned Super UVGS representation which not only maintains the appearance of the 3DGS object, but also serves as a proxy for geometrical shape by encoding all the 3DGS attributes into the same coherent feature space. Table 4 compares the CLIP Score of our text-conditioned generation results and the current SOTA methods. The unconditional and conditional qualitative comparison results are presented in Fig. 7 and Fig. 6, respectively.

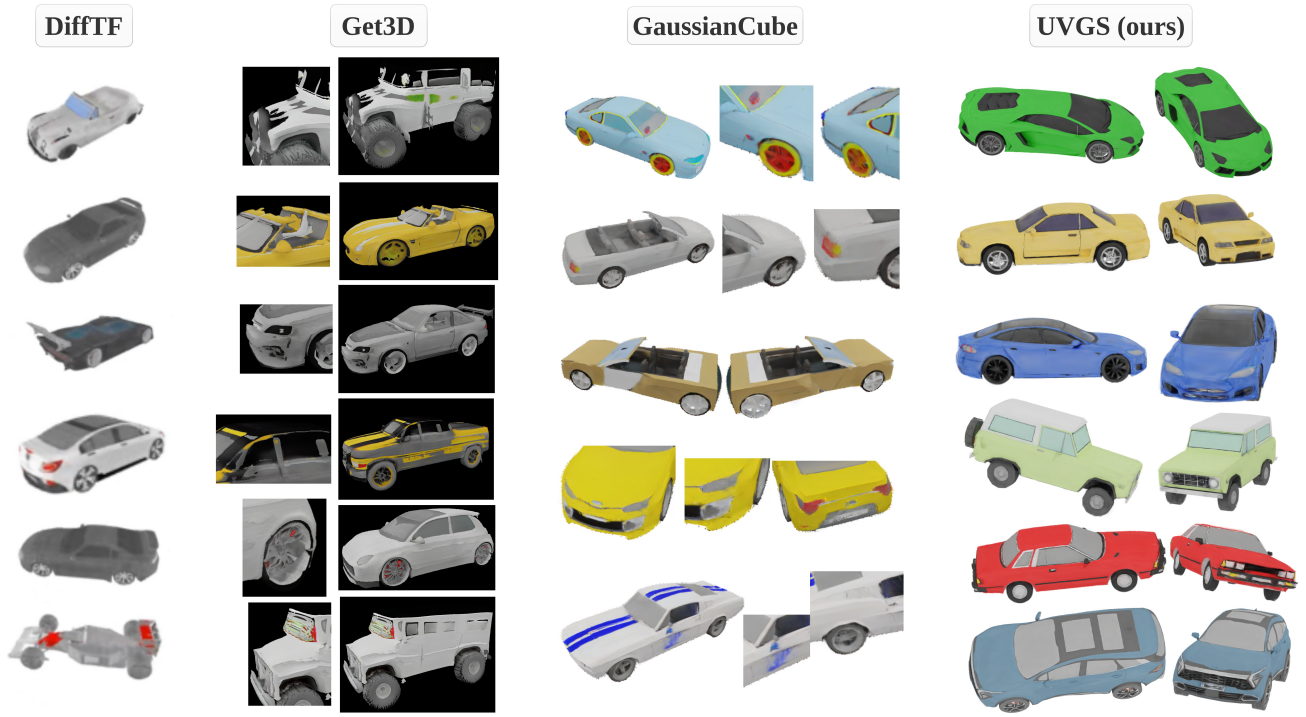

Figure 6. Here we show more comparison of unconditional 3D asset generation on the cars category with SOTA methods. Figure shows that DiffTF [3] produces low-quality, low-resolution cars lacking detail. While Get3D [14] achieve higher resolution, it suffers from 3D inconsistency, numerous artifacts, and lack richness in 3D detail. Similar issues are found in GaussianCube [61] along with symmetric inconsistency in the results. In contrast, our method generates high-quality, high-resolution objects that are 3D consistent with sharp, well-defined edges. The top three rows show the unconditional generation results of our method using ShapeNet dataset, while the bottom 3 show from Objaverse dataset.

Figure 7. Text-conditioned generation results on various baselines and the proposed method. Our method not only generates high-quality assets for simpler objects, but also for complicated objects with intricate geometries like *the wheel* or *the airplane*.

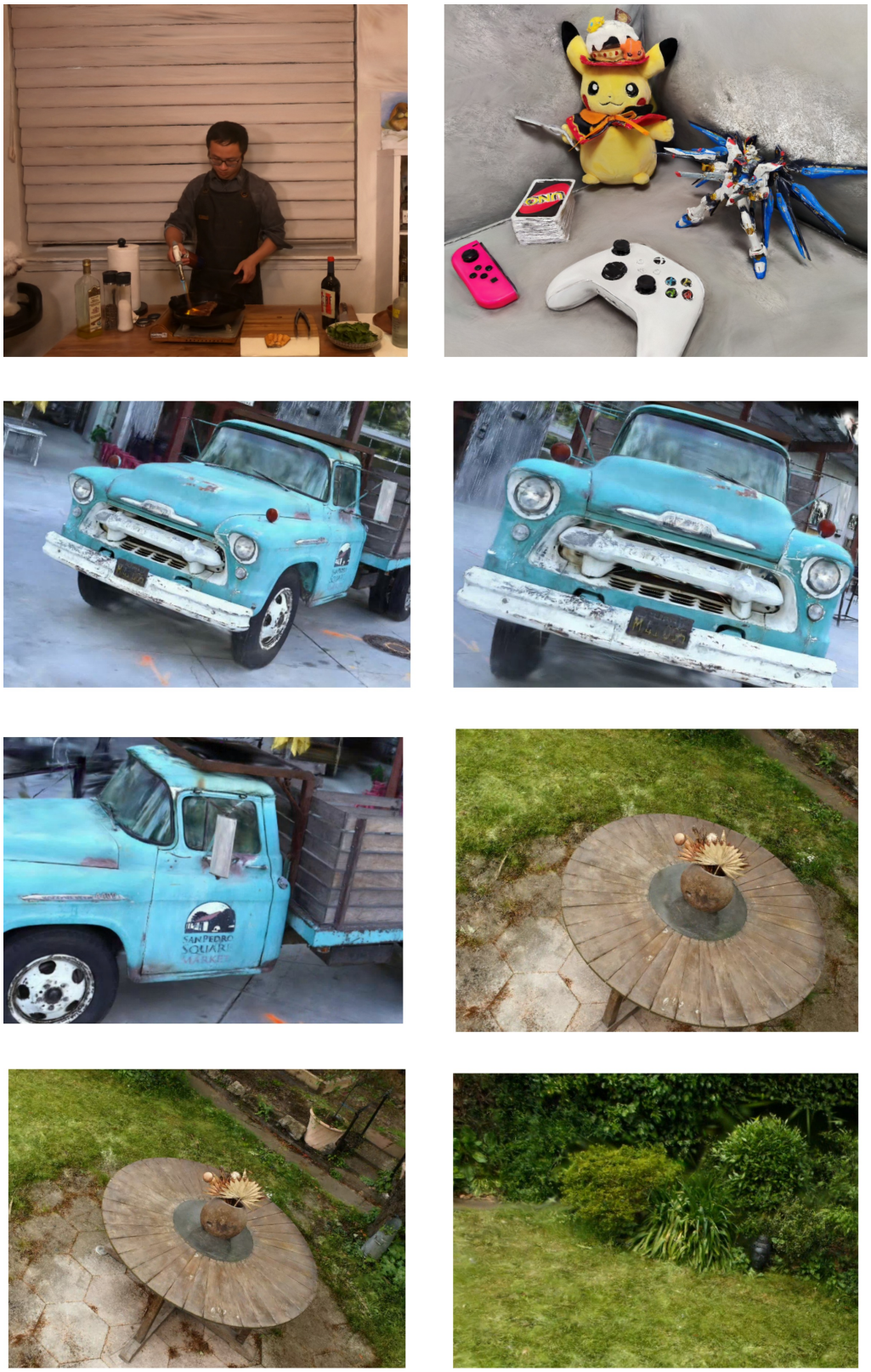

Figure 8. To show the effectiveness of proposed UVGS maps in capturing the intricacies of a complex real-world scene, we used a 12 layer UV map to reconstruct the 3D scenes.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 12, 13

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5855–5864, 2021. 2

[3] Ziang Cao, Fangzhou Hong, Tong Wu, Liang Pan, and Ziwei Liu. Large-vocabulary 3d diffusion model with transformer. *arXiv preprint arXiv:2309.07920*, 2023. 3, 6, 7, 8, 13, 14

[4] Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas J Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16123–16133, 2022. 2, 6, 8, 13

[5] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 3, 6

[6] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. In *European conference on computer vision*, pages 333–350. Springer, 2022. 2

[7] Minghao Chen, Junyu Xie, Iro Laina, and Andrea Vedaldi. Shap-editor: Instruction-guided latent 3d editing in seconds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26456–26466, 2024. 8, 13

[8] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21401–21412, 2024. 3

[9] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4456–4465, 2023. 2

[10] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13142–13153, 2023. 3, 5, 8

[11] Slava Elizarov, Ciara Rowles, and Simon Donné. Geometry image diffusion: Fast and data-efficient text-to-3d with image-based surface representation. *arXiv preprint arXiv:2409.03718*, 2024. 2

[12] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021. 12

[13] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5501–5510, 2022. 2

[14] Jun Gao, Tianchang Shen, Zian Wang, Wenzheng Chen, Kangxue Yin, Daiqing Li, Or Litany, Zan Gojcic, and Sanja Fidler. Get3d: A generative model of high quality 3d textured shapes learned from images. *Advances In Neural Information Processing Systems*, 35:31841–31854, 2022. 3, 6, 7, 8, 13, 14

[15] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv preprint arXiv:2303.05371*, 2023. 2

[16] Xianglong He, Junyi Chen, Sida Peng, Di Huang, Yangguang Li, Xiaoshui Huang, Chun Yuan, Wanli Ouyang, and Tong He. Gvgen: Text-to-3d generation with volumetric representation. In *European Conference on Computer Vision*, pages 463–479. Springer, 2025. 2, 3

[17] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3, 7, 12

[18] Han Huang, Yulun Wu, Junsheng Zhou, Ge Gao, Ming Gu, and Yu-Shen Liu. Neusurf: On-surface priors for neural surface reconstruction from sparse input views. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2312–2320, 2024. 2

[19] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1, 2, 6, 11

[20] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 7

[21] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 12, 13

[22] Jiahe Li, Jiawei Zhang, Xiao Bai, Jin Zheng, Xin Ning, Jun Zhou, and Lin Gu. Dngaussian: Optimizing sparse-view 3d gaussian radiance fields with global-local depth normalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20775–20785, 2024. 2

[23] Shujuan Li, Junsheng Zhou, Baorui Ma, Yu-Shen Liu, and Zhizhong Han. Neaf: Learning neural angle fields for point normal estimation. In *Proceedings of the AAAI conference on artificial intelligence*, pages 1396–1404, 2023. 2

[24] 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*, pages 300–309, 2023. 2

[25] Tiange Luo, Chris Rockwell, Honglak Lee, and Justin Johnson. Scalable 3d captioning with pretrained models. *Advances in Neural Information Processing Systems*, 36, 2024. 12

[26] Baorui Ma, Haoge Deng, Junsheng Zhou, Yu-Shen Liu, Tiejun Huang, and Xinlong Wang. Geodream: Disentangling 2d and geometric priors for high-fidelity and consistent 3d generation. *arXiv preprint arXiv:2311.17971*, 2023. 2

[27] Qi Ma, Yue Li, Bin Ren, Nicu Sebe, Ender Konukoglu, Theo Gevers, Luc Van Gool, and Danda Pani Paudel. Shapesplat: A large-scale dataset of gaussian splats and their self-supervised pretraining. *arXiv preprint arXiv:2408.10906*, 2024. 2, 3

[28] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673, 2023. 2

[29] 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. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2

[30] Chaerin Min and Srinath Sridhar. Genheld: Generating and editing handheld objects. *arXiv preprint arXiv:2406.05059*, 2024. 2

[31] Chaerin Min, Sehyun Cha, Changhee Won, and Jongwoo Lim. Tsdf-sampling: Efficient sampling for neural surface field using truncated signed distance field. *arXiv preprint arXiv:2311.17878*, 2023. 2

[32] Yuxuan Mu, Xinxin Zuo, Chuan Guo, Yilin Wang, Juwei Lu, Xiaofeng Wu, Songcen Xu, Peng Dai, Youliang Yan, and Li Cheng. Gsd: View-guided gaussian splatting diffusion for 3d reconstruction. *arXiv preprint arXiv:2407.04237*, 2024. 2, 3, 7

[33] Norman Müller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner. Diffrf: Rendering-guided 3d radiance field diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4328–4338, 2023. 2

[34] Francesco Palandra, Andrea Sanchietti, Daniele Baieri, and Emanuele Rodolà. Gsedit: Efficient text-guided editing of 3d objects via gaussian splatting. *arXiv preprint arXiv:2403.05154*, 2024. 2

[35] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 3

[36] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 2

[37] 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. 12, 13

[38] Aashish Rai and Srinath Sridhar. Egosonics: Generating synchronized audio for silent egocentric videos. *arXiv preprint arXiv:2407.20592*, 2024. 7

[39] Aashish Rai, Hiresh Gupta, Ayush Pandey, Francisco Vicente Carrasco, Shingo Jason Takagi, Amaury Aubel, Daeil Kim, Aayush Prakash, and Fernando De la Torre. Towards realistic generative 3d face models. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3738–3748, 2024. 2

[40] 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 *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2349–2359, 2023. 2

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3, 7, 12, 13

[42] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmen-

tation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015. 13

[43] Li Shen and Fillia Makedon. Spherical mapping for processing of 3d closed surfaces. *Image and vision computing*, 24 (7):743–761, 2006. 2, 9

[44] J Ryan Shue, Eric Ryan Chan, Ryan Po, Zachary Ankner, Jiajun Wu, and Gordon Wetzstein. 3d neural field generation using triplane diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20875–20886, 2023. 2

[45] Jingxiang Sun, Bo Zhang, Ruizhi Shao, Lizhen Wang, Wen Liu, Zhenda Xie, and Yebin Liu. Dreamcraft3d: Hierarchical 3d generation with bootstrapped diffusion prior. *arXiv preprint arXiv:2310.16818*, 2023. 2

[46] Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10208–10217, 2024. 2, 3, 7

[47] Fariborz Taherkhani, Aashish Rai, Quankai Gao, Shaunak Srivastava, Xuanbai Chen, Fernando De la Torre, Steven Song, Aayush Prakash, and Daeil Kim. Controllable 3d generative adversarial face model via disentangling shape and appearance. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 826–836, 2023. 2

[48] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023. 2, 3, 6, 8, 13

[49] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22819–22829, 2023. 2

[50] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2025. 6, 7, 8, 13

[51] Zhicong Tang, Shuyang Gu, Chunyu Wang, Ting Zhang, Jianmin Bao, Dong Chen, and Baining Guo. Volumediffusion: Flexible text-to-3d generation with efficient volumetric encoder. *arXiv preprint arXiv:2312.11459*, 2023. 2

[52] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 7

[53] 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4563–4573, 2023. 2

[54] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3295–3306, 2023. 2

[55] Yuxuan Wang, Xuanyu Yi, Zike Wu, Na Zhao, Long Chen, and Hanwang Zhang. View-consistent 3d editing with gaussian splatting. In *European Conference on Computer Vision*, pages 404–420. Springer, 2025. 2

[56] 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. 2

[57] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20310–20320, 2024. 2

[58] Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. *arXiv preprint arXiv:2403.14621*, 2024. 2, 3

[59] Xingguang Yan, Han-Hung Lee, Ziyu Wan, and Angel X Chang. An object is worth 64x64 pixels: Generating 3d object via image diffusion. *arXiv preprint arXiv:2408.03178*, 2024. 2

[60] Taoran Yi, Jiemin Fang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. *arXiv preprint arXiv:2310.08529*, 2023. 2

[61] Bowen Zhang, Yiji Cheng, Jiaolong Yang, Chunyu Wang, Feng Zhao, Yansong Tang, Dong Chen, and Baining Guo. Gaussiancube: Structuring gaussian splatting using optimal transport for 3d generative modeling. *arXiv preprint arXiv:2403.19655*, 2024. 2, 3, 6, 7, 8, 13, 14

[62] Kai Zhang, Sai Bi, Hao Tan, Yuanbo Xiangli, Nanxuan Zhao, Kalyan Sunkavalli, and Zexiang Xu. Gs-lrm: Large reconstruction model for 3d gaussian splatting. In *European Conference on Computer Vision*, pages 1–19. Springer, 2025. 2, 3

[63] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 7

[64] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6, 11

[65] 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. 5

[66] Junsheng Zhou, Yu-Shen Liu, and Zhizhong Han. Zero-shot scene reconstruction from single images with deep prior assembly. *arXiv preprint arXiv:2410.15971*, 2024. 2

[67] Junsheng Zhou, Weiqi Zhang, and Yu-Shen Liu. Diffgs: Functional gaussian splatting diffusion. *arXiv preprint arXiv:2410.19657*, 2024. 2, 3

[68] Junsheng Zhou, Weiqi Zhang, Baorui Ma, Kanle Shi, Yu-Shen Liu, and Zhizhong Han. Udiff: Generating conditional unsigned distance fields with optimal wavelet diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21496–21506, 2024. 2

[69] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10324–10335, 2024. 2, 3, 7, 13

[70] Zi-Xin Zou, Zhipeng Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Yan-Pei Cao, and Song-Hai Zhang. Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10324–10335, 2024. 2