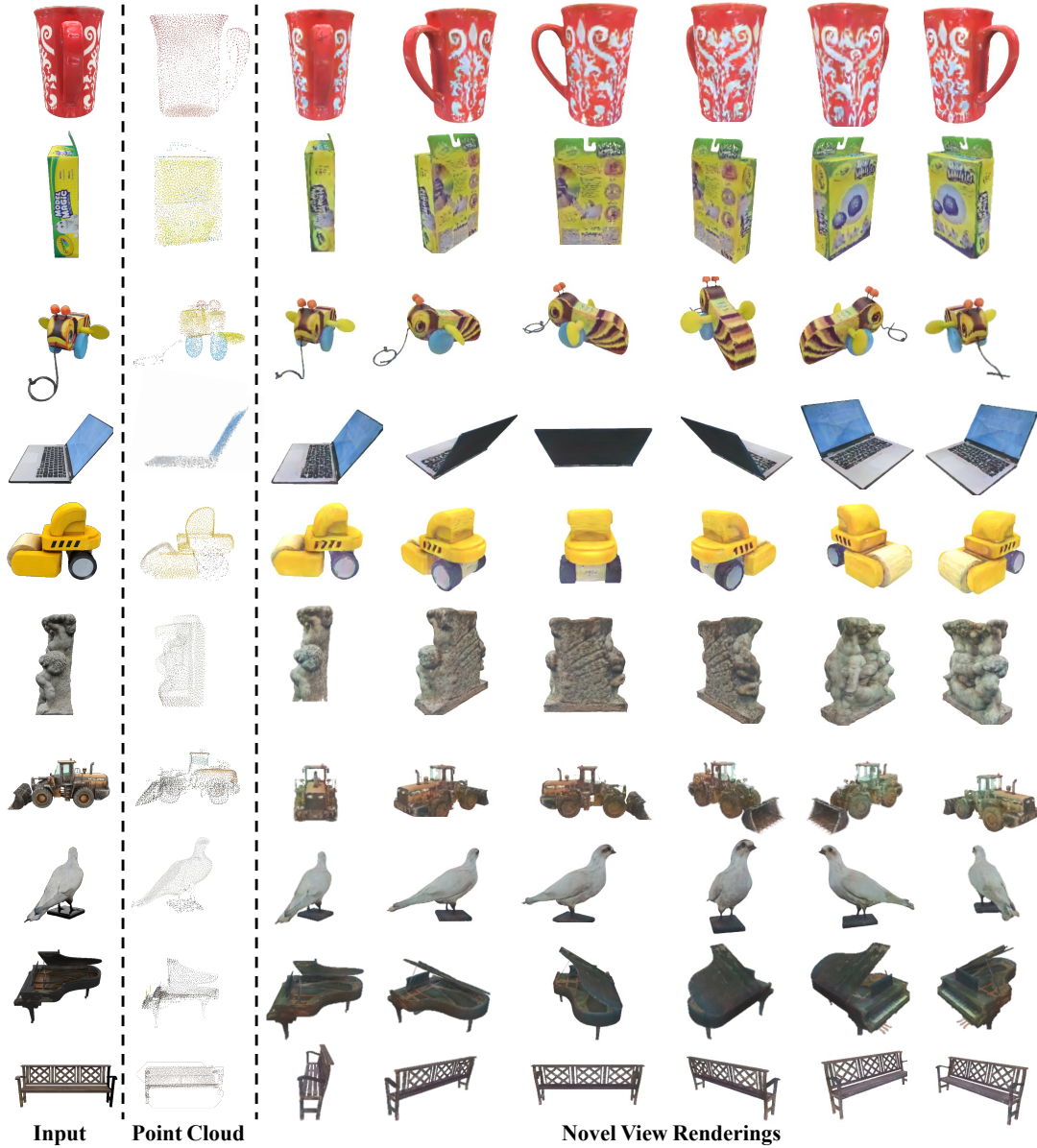


# Supplementary Materials: Large Point-to-Gaussian Model for Image-to-3D Generation

Anonymous Authors



**Figure 1: More Qualitative results. Our method produces high-fidelity generation results from a single-view image.**

## A ARCHITECTURE DETAILS

The proposed Point-to-Gaussian Generator is based on [3], which utilizes an encoder-decoder architecture and takes the point cloud generated from pretrained 3D diffusion models as inputs, and outputs 3D Gaussians for splatting. Details of the network architecture are presented in Table 1.

## B MULTI-VIEW IMAGE INPUT DETAILS

Leveraging existing image diffusion models, our Point-to-Gaussian Generator can also support cross modality enhancement with multiple images. Specifically, we employ MVDream [5] to initially convert the single image input into four consistent images, and subsequently extract image features from four distinct views using

Modules	Details	
Point Cloud Upsampler	SPD Layer [7]	3 layers with upscale factor of [2, 1, 2]
APP Block	Attention Layers	2 layers with channel width 256, # heads 4
	Projection Layers	Image feature 768, raster point radius 0.0075, points per pixel 1
	Point Feature Extractor Layers	Point-based layer and voxel-based layer
	Cross View Layers	Attention layer if multi-view else Identity layer
Point to Gaussian Encoder	APP Block	1 block per layer
	Point cloud scales	Point cloud with 4 scales of [1024, 256, 64, 16]
	Feature dims	Out feature dims (64, 128, 256, 512) with voxel resolutions (32, 16, 8)
Point to Gaussian Decoder	Normal Block	Point Feature Extractor Layers <b>ONLY</b>
	Point cloud scales	Point cloud with 4 scales of [64, 256, 1024, 16384]
	Feature dims	Out feature dims (256, 256, 128, 64) with voxel resolutions (8, 8, 16, 32)
Multi Linear Heads	Scale activation	Softplus
	Rotation activation	Normalize
	Opacity activation	Sigmoid
	Position offset activation	Clamp
	Color activation	Sigmoid

Table 1: Details of our Point-to-Gaussian Generator architecture.

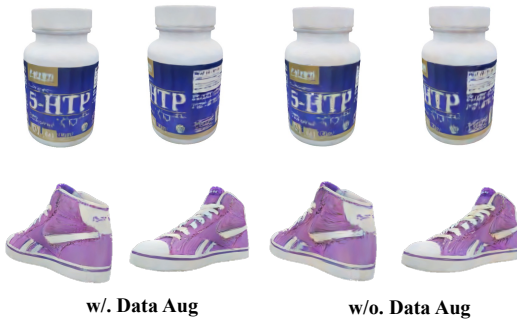


Figure 2: Visualization of rendered images with and without data augmentation. Our model achieves richer textures and more refined details with data augmentation.

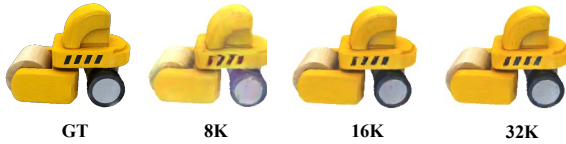


Figure 3: Visualization with varying numbers of 3D Gaussians.

the pretrained DINOv2 [4]. The framework for multi-view input is akin to that for single-image, with the exception of an additional cross-view layer to fuse the features from different views. Drawing inspiration from [5, 6], we utilize the self-attention mechanism for cross-view feature fusion. In detail, we flatten image features from four views and concatenate them along the sequence length to perform attention in all views.

## C MORE VISUALIZATIONS

We present additional visualization results from the Objaverse [1] and Google Scanned Objects [2] datasets in this subsection.

## D ADDITIONAL ABLATION STUDY

### D.1 Point Cloud Upsampling Rates

A substantial quantity of 3D Gaussians can more effectively capture the details of an object, albeit at the cost of increased overhead. In this section, we conduct an additional ablation study on the number of 3D Gaussians. Specifically, we control the number of 3D Gaussians by adjusting the upsampling rates of the point cloud. The experimental results are presented in Fig. 3, which reveals that employing more 3D Gaussians can enhance the clarity and richness of the texture. 8K Gaussians results in a significantly inferior outcome compared to 16K, but the improvement is limited from 16K to 32K. Therefore, we opt to use 16K Gaussians to strike a balance between performance and overhead.

### D.2 Data Augmentation

The point cloud utilized for inference is generated by the pretrained 3D diffusion model, which may differ from the ground truth point cloud employed during training. To mitigate the gap in data distributions, we implement data augmentation to perturb the training data during the training process. In this section, we ablate the role of data augmentation, with the results displayed in Fig. 2. We can see that the application of data augmentation increases the robustness of noisy point cloud inputs, ultimately yielding better texture rendering results.

## REFERENCES

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

- [2] Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, ThomasB. McHugh, and Vincent Vanhoucke. 2022. Google Scanned Objects: A High-Quality Dataset of 3D Scanned Household Items. (Apr 2022).
- [3] Zhijian Liu, Haotian Tang, Yujun Lin, and Song Han. 2019. Point-voxel cnn for efficient 3d deep learning. *Advances in neural information processing systems* 32 (2019).
- [4] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193* (2023).
- [5] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. 2023. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512* (2023).
- [6] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. 2024. LGM: Large Multi-View Gaussian Model for High-Resolution 3D Content Creation. *arXiv preprint arXiv:2402.05054* (2024).
- [7] Peng Xiang, Xin Wen, Yu-Shen Liu, Yan-Pei Cao, Pengfei Wan, Wen Zheng, and Zhizhong Han. 2021. Snowflakenet: Point cloud completion by snowflake point deconvolution with skip-transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*. 5499–5509.

291  
292  
293  
294  
295  
296  
297  
298  
299  
300  
301  
302  
303  
304  
305  
306  
307  
308  
309  
310  
311  
312  
313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348