EVPGS: Enhanced View Prior Guidance for Splatting-based Extrapolated View Synthesis

Supplementary Material

In this **supplementary document**, we first provide a detailed overview of our Merchandise3D dataset, including its configuration and capture process, in Sec.A. Next, we elaborate on further experimental details, including dataset specifics and training implementation, in Sec.B. Subsequently, we present extended results for EVPGS, covering more quantitative and qualitative comparison experiments in Sec.C. Finally, we provide an example for the real-world application of EVPGS in Sec.D.

A. Our Merchandise3D Dataset

To adequately represent both real-world applications and the challenges posed by EVS, our Merchandise3D dataset comprises real-world merchandise objects with diverse structures and textures. The dataset has 7 objects instances, each represented in the form of 100 camera views, where each view is a triplet that contains an RGB image, an object mask, and the corresponding camera parameters. As shown in Figure D left, we select 50 views captured from nearly horizontal angles as the training set and another 50 views, elevated by approximately 40 degrees, as the testing set. We instantiate each of these objects in Figure A. Each object is captured originally in the format of a video and we release these object videos as well. The resolution of videos in our dataset is 1280×720. We describe our video capturing and processing pipeline as follows.

When capturing video for each object, we place the object on a motorized rotating turntable and keep recording until the object is rotated for two rounds, which comprise one round from a horizontal viewpoint followed by another round from an elevated viewpoint looking downwards at approximately 40 degrees. We slowly lift the camera after the first round and keep the camera fixed otherwise.

When processing each video, we sample 100 frames on average and solve the camera parameters for these frames using HLOC [8]. To boost the accuracy of HLOC, we introduce additional textures to the scene by placing graffiti sheets under the object when capturing the video. Each graffiti we select has non-repeating local patterns across the graffiti as to avoid confusing HLOC. When using multiple graffiti sheets, we use a different graffiti on each sheet for the same reason. Since we focus the view synthesis task on objects, we also provide the object mask of each frame obtained using the off-the-shelf Segment Anything Model (SAM) [6], while we treat the graffiti as background.

Figure A. **Our Merchandise3D dataset**, which comprises real-world merchandise objects with diverse structures and textures. To boost the accuracy of HLOC [8], we place graffiti sheets under each object during capturing as explained in Section A. We focus on EVS for objects in this work, and the graffiti is not a part of object to be reconstructed.

B. Experiments Details

B.1. Datasets

Details: For the two public datasets, DTU [4] and Synthetic-NeRF [7], we utilize the ground truth masks provided by the datasets to segment objects from the background for the EVS task. For the DTU [4] dataset, we follow [2, 11, 12] to evaluate the proposed method on 14 selected scenes. The selected scan IDs are 24, 37, 40, 55, 63, 65, 83, 97, 105, 106, 110, 114, 118, and 122. Scan 69 was excluded from our experiments as it lacks the ground truth masks necessary for segmenting the object from the background. To improve experimental efficiency, the image resolution is resized to 777×851. For the Synthetic-NeRF [7] dataset, all eight synthetic objects are included in our experiments. The training images keep the original resolution of 800×800. For the Merchandise3D dataset, please refer to Sec. A

Visualization: As shown in Table 2 of the main paper, each dataset organized following the EVS scenario exhibits drastic view coverage difference between the training and testing splits. We visualize some examples from each dataset in Figure D to showcase the difficulty of the EVS problem.

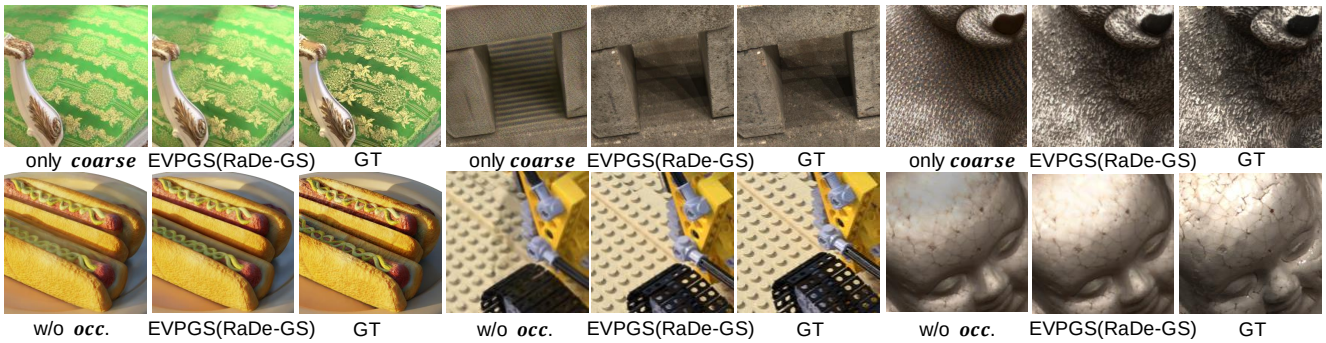

Figure B. **More ablation study results**. Complementary to Figure 6 in the main paper, we present additional qualitative ablation study results for the other two simplified version of EVPGS, i.e. *only coarse* and *full w/o occ.* **Top row:** In comparison to the simplified EVPGS with only the coarse stage, the full EVPGS can better reconstruct fine details. **Bottom row:** In comparison to the simplified EVPGS without the occlusion-aware reprojection strategy, the full EVPGS produces renderings clear of corruptions caused by occlusion.

B.2. Training

Coarse Stage: As mentioned in Sec. 4 in the main paper, we use 3DGS [5], Mip-Splatting [10], 2DGS [2], GOF [11] and RaDe-GS [12] as alternative backbones for our EVPGS framework. When pre-training these GS-based models, we adhere to the training configurations specified in their respective original papers. For our appearance regularization (Sec. 3.2 of the main paper), we utilize the Stable-Diffusion-2.1 model directly, without any additional fine-tuning on our dataset while using empty text prompts. For our geometry regularization (Sec. 3.2 of the main paper), we use the *pytorch3d* toolbox to rasterize the depth map from reconstructed mesh. We set $\lambda_a = 1e-7$ and $\lambda_g = 1e-1$ in Eq. 8 of the main paper.

Fine stage: For our View Prior Refinement strategy (Section 3.4 of the main paper), we select the parameters that bring the best performance for EVPGS. We select $w_h = 0.8$ and $w_l = 0.5$.

C. More Results

C.1. Computational Cost

We compute the training time of RaDe-GS [12], VEGS*[3], and EVPGS(RaDe-GS) to assess the efficiency of our framework. For all three methods, we train each for a total of 30k iterations. Specifically, for EVPGS(RaDe-GS), we follow the coarse-to-fine training process: 20k iterations for pretraining, 1k iterations for the coarse stage, and 9k iterations for the fine stage. Table A presents the training time of each method across different datasets. To mitigate artifacts and recover high-frequency details in extrapolated views, EVPGS incorporates several strategies that increase training time compared to the baseline RaDe-GS. While VEGS* is also a GS-based method specifically designed for the EVS problem, EVPGS is both more efficient and achieves superior performance. As shown in Table A and Table 1 of the main paper, EVPGS effectively balances performance and computational cost.

Training Time	DTU	Merchandise3D	Synthetic-NeRF
RaDe-GS [12]	~13.5m	~15m	~14.2m
VEGS* [3]	~52.5m	~43.3m	~35m
EVPGS(RaDe-GS)	~25m	~26m	~22.5m

Table A. **Computational Cost Comparison of RaDe-GS [12], VEGS* [3] and EVPGS(RaDe-GS).** We compare the training time of these three methods. Compared to the RaDe-GS baseline, our EVPGS achieves a good balance between performance and computational cost. Additionally, EVPGS is more efficient than VEGS* in handling the EVS problem.

C.2. Comparing with Sparse-view GS Methods

To assess the applicability of sparse-view GS methods in EVS scenarios, we select the state-of-the-art sparse-view GS method MVPGS [9] as a baseline and integrate it into our EVPGS framework. We uniformly sample 9 training views per scene from our EVS training split (Sec. 4.1 of the main paper) to train MVPGS. As shown in Table B, MVPGS performs poorly with the EVS task, while our EVPGS framework still provides a slight performance improvement.

C.3. Results on real-life dataset

EVPGS is primarily designed for object-centric scenes. To further evaluate its generalizability, we tested it on real-world scenes from the Mip-NeRF360 [1] dataset. We created training and testing splits following the EVS setting, resulting in an average pitch angle difference of 17.01° between splits across scenes, compared to 4.10° in the original Mip-NeRF360 splits. Notably, since the reconstructed mesh after the pretraining stage is not available in Mip-NeRF360, we used the EVPGS variant $full\ w/o\ mesh$ (Section 4.3 of the main paper) for evaluation. As shown in Table C and

	DTU			Merchandise3D			Synthetic-NeRF		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
MVPGS [9]	21.032	0.7761	0.1705	11.067	0.7505	0.2693	13.929	0.7008	0.3613
RaDe-GS [12]	25.778	0.8882	0.0753	23.559	0.9134	0.0645	27.531	0.9190	0.0530
EVPGS(MVPGS)	21.471	0.7877	0.1643	11.133	0.7519	0.2670	13.965	0.7017	0.3598
EVPGS(RaDe-GS)	**26.488**	**0.8991**	**0.0670**	**25.136**	**0.9267**	**0.0496**	**27.849**	**0.9243**	**0.0498**

Table B. **Quantitative evaluation of our EVPGS framework with MVPGS [9].** We evaluate EVPGS integrated with the state-of-the-art sparse-view method MVPGS [9]. The results indicate that sparse-view methods struggle to handle the EVS scenario effectively. However, our EVPGS framework still provides a slight improvement over MVPGS.

	Mip-NeRF360		
	PSNR ↑	SSIM ↑	LPIPS ↓
RaDe-GS	19.066	0.5679	0.2253
EVPGS(RaDe-GS)*	**19.446**	**0.5757**	**0.2091**

Table C. **Quantitative evaluation of our EVPGS framework on the Mip-NeRF360 [1] dataset.** We compare EVPGS(RaDe-GS) variant $full\ w/o\ mesh$ (denoted as EVPGS(RaDe-GS)*) with the RaDe-GS baseline, demonstrating that our EVPGS framework also achieves improved performance in outdoor scenes.

Figure C. **Qualitative Results on Mip-NeRF360 [1] dataset.** The result of EVPGS(RaDe-GS) variant $full\ w/o\ mesh$ (denoted as EVPGS(RaDe-GS)*) show fewer artifacts and preserve more details compared to the RaDe-GS baseline.

Figure C, EVPGS achieves a performance boost comparable to that observed in object-centric scenes, demonstrating its potential in scene-level reconstruction.

C.4. More Qualitative Results on Ablation Study

In the main paper, we presented the qualitative results of our ablation study in Figure 6, showcasing two simplified versions of EVPGS: *only fine* and *full w/o ref.*. In addition to Figure 6, we conduct further qualitative experiments on the DTU [4] and Synthetic-NeRF [7] datasets to evaluate two other simplified versions of EVPGS: *only coarse* and *full w/o occ.*. These experiments assess the effectiveness of our fine stage and occlusion-aware module. We present the results in Figure B, which demonstrate that the fine stage (Section 3.3 of the main paper) enhances the reconstruction of fine details, as evidenced by comparing the full EVPGS with *only coarse*. Additionally, the occlusion-aware reprojection strategy (Section 3.3 of the main paper) effectively mitigates image corruption caused by occlusions, as shown by the comparison between the full EVPGS and *full w/o occ.*.

C.5. More Qualitative Results on EVPGS

In addition to the qualitative results in Figure 4 and Figure 5 of the main paper where we compared our overall EVPGS with the other methods on the Merchandise3D, DTU [4], and Synthetic-NeRF [7] datasets, we conduct further qualitative comparison on more object instances from these three datasets. We present the additional comparison results in Figure E and Figure F for Merchandise3D, Figure G and Figure H for DTU [4], Figure I and Figure J for Synthetic-NeRF [7].These results highlight the intricate structures and fine details accurately reconstructed by our EVPGS framework across all three datasets, showcasing the effectiveness of EVPGS in addressing the EVS task.

D. Application

Our EVPGS framework enables a practical application for free-view merchandise exhibition, allowing users to effortlessly showcase any object they desire. When creating attractive merchandise videos using the conventional commercial technologies, it often requires a professional photographer to capture the object from a diversity of viewpoints along a set of planned camera paths, which can make the filming process labor-intensive and costly. With EVPGS, users only need to capture a simple circular sequence of images around the object using a smartphone. Then our EVPGS generates high-quality extrapolated views, enabling the creation of engaging videos from various perspectives with minimal effort. EVPGS not only reduces the burden of photography but also ensures display of the object with good realism and high-quality details, providing an efficient and effective solution for merchandise display. We provide several merchandise display videos with artistically designed camera paths in our project page. **We encourage the readers to watch the videos in our project page to gain a better grasp of the capabilities of our EVPGS in real-world applications.**

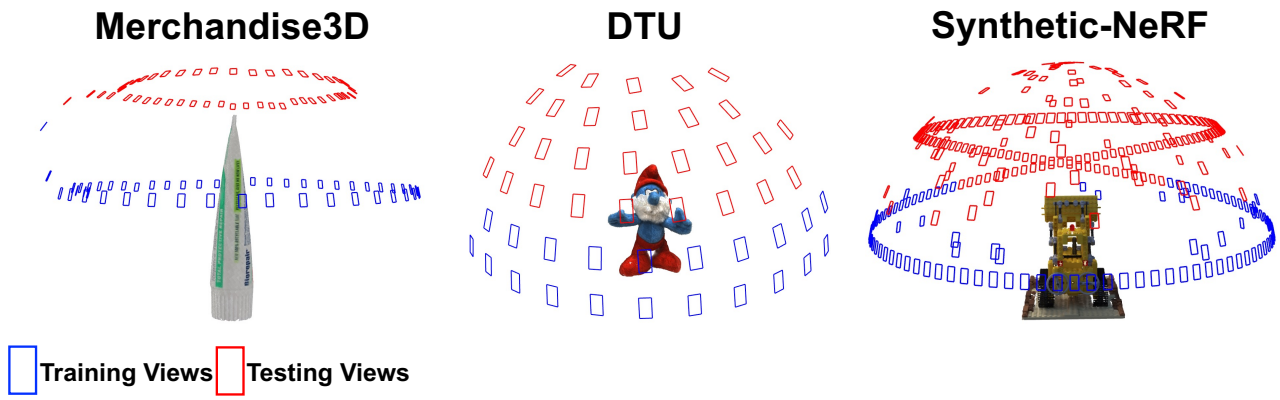

Figure D. **Visualization of the substantial view coverage disparity between the training and testing splits in our EVS scenario.** The quantitative average angle differences are provided in Table 2 of the main paper.

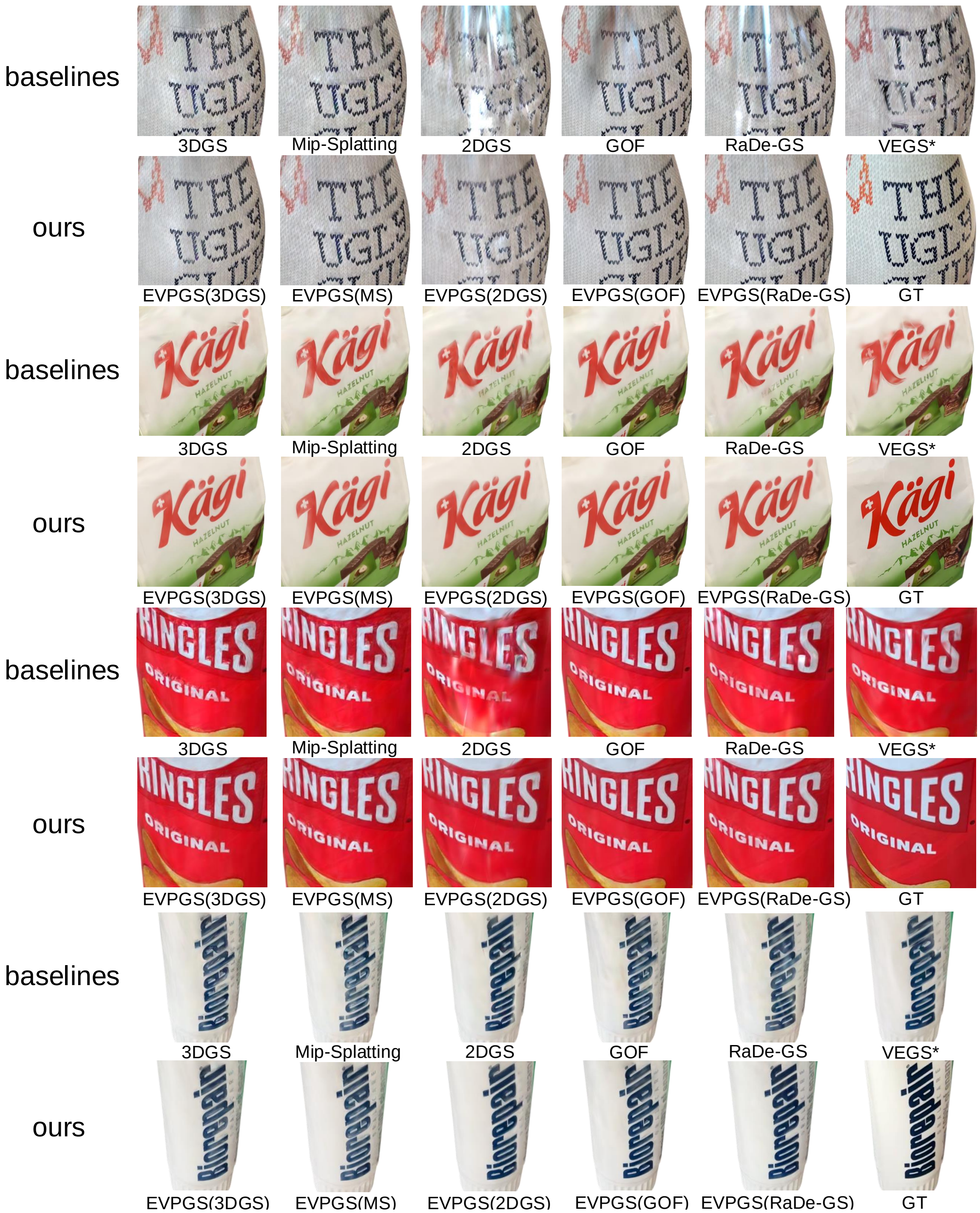

Figure E. **More qualitative comparison on our Merchandise3D dataset.**

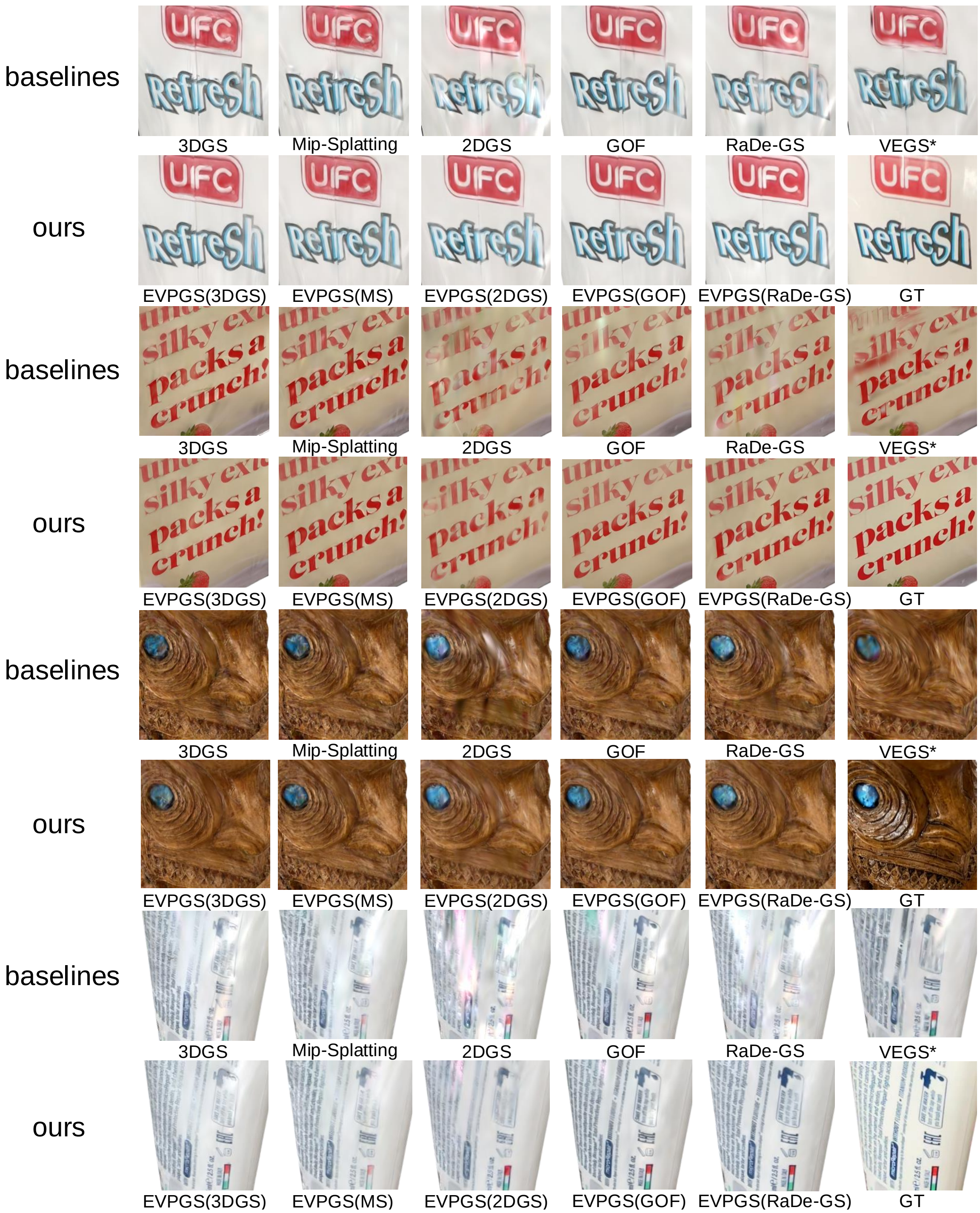

Figure F. **More qualitative comparison on our Merchandise3D dataset.**

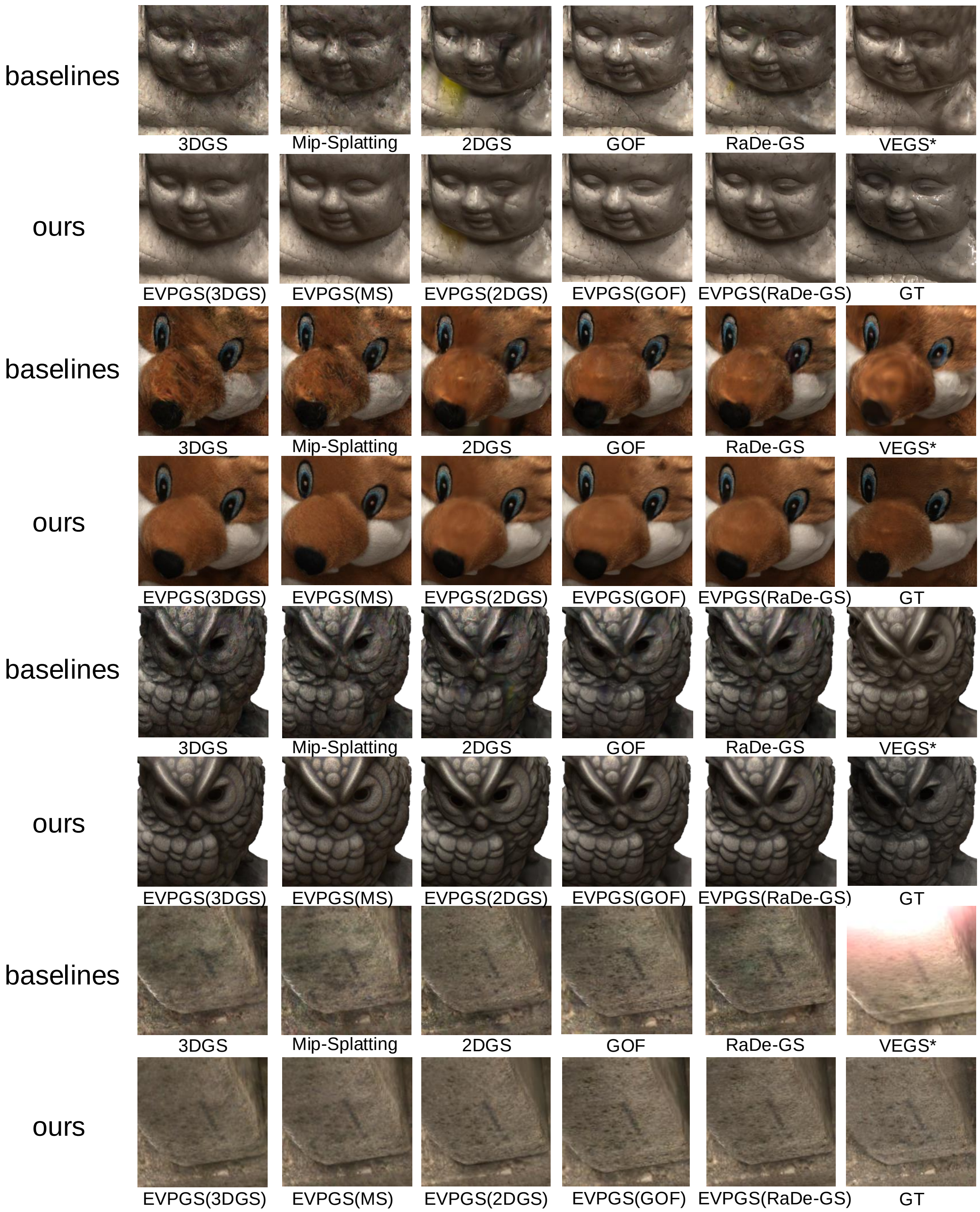

Figure G. **More qualitative comparison on the DTU [4] dataset**.

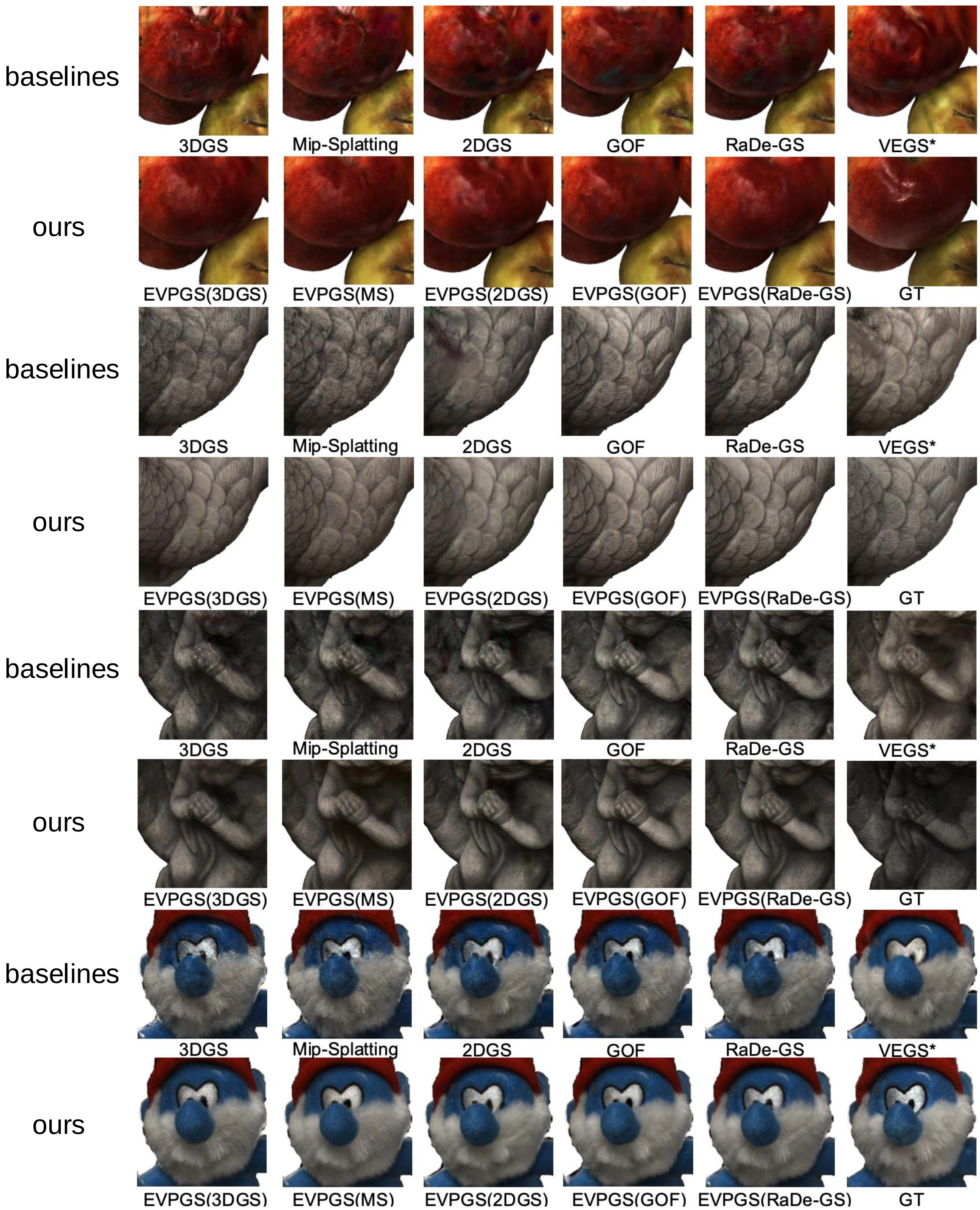

Figure H. **More qualitative comparison on the DTU [4] dataset**.

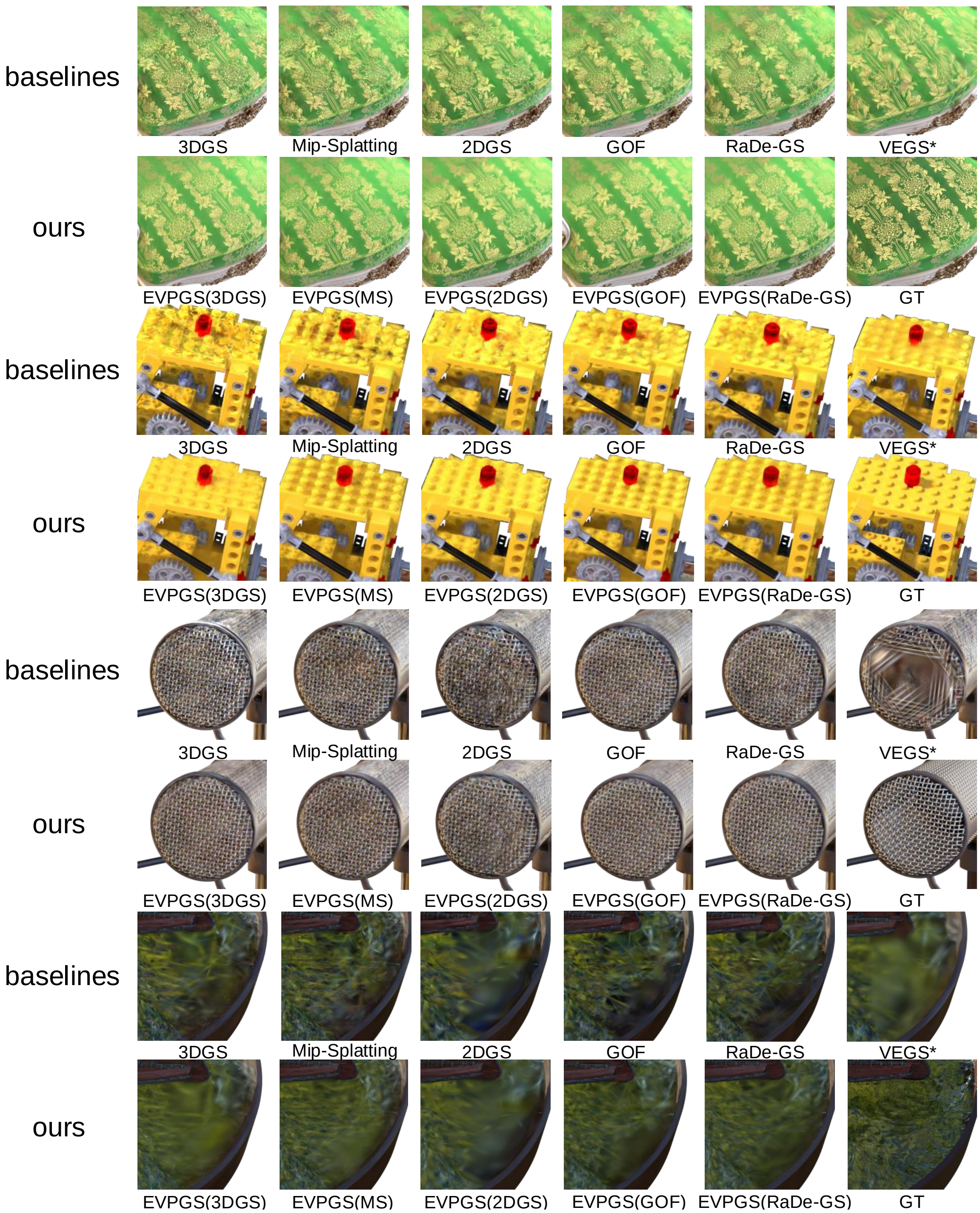

Figure I. **More qualitative comparison on the Synthetic-NeRF [7] dataset**.

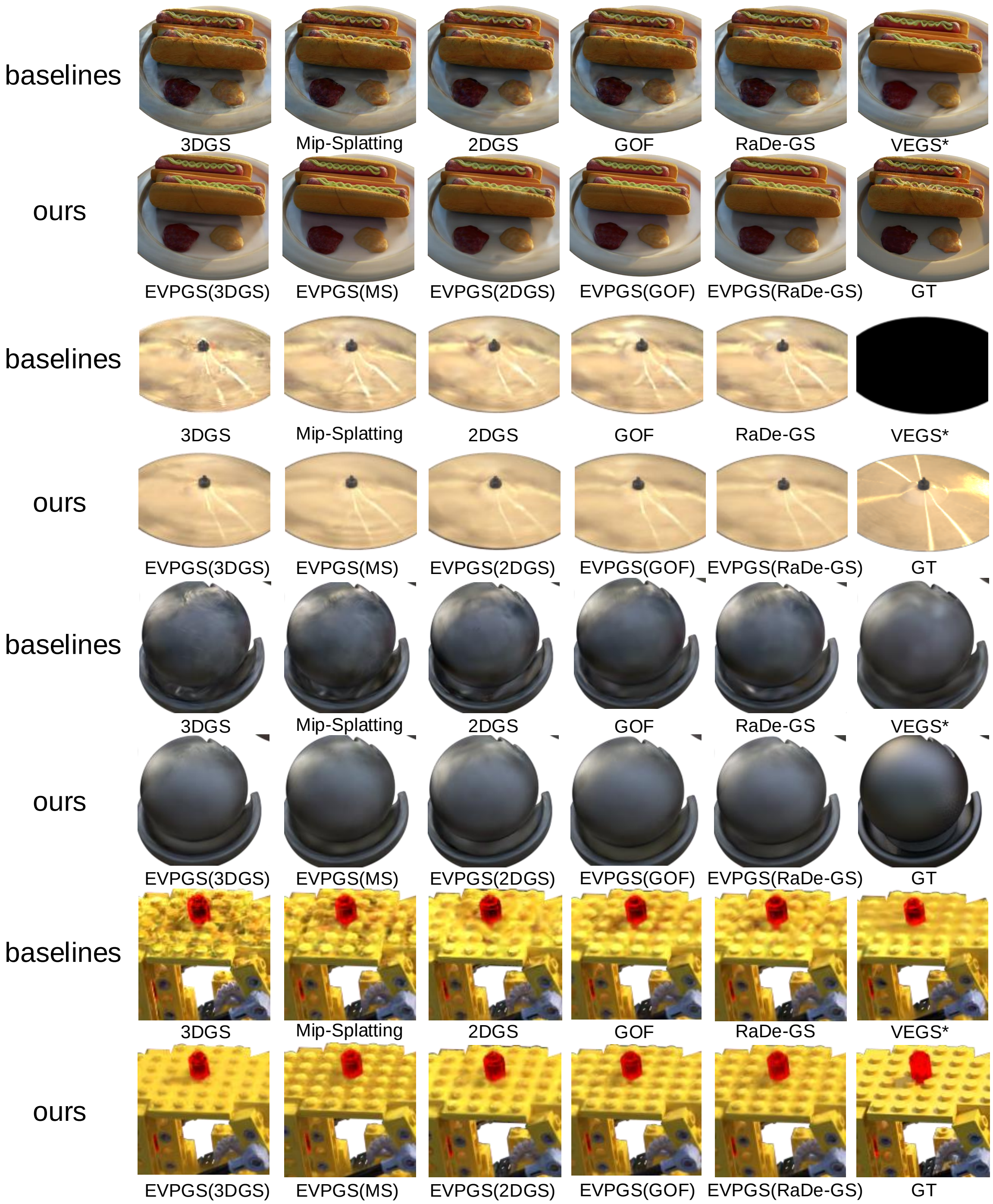

Figure J. **More qualitative comparison on the Synthetic-NeRF [7] dataset**.

References

[1] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022. 2, 3

[2] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 1, 2

[3] Sungwon Hwang, Min-Jung Kim, Taewoong Kang, Jayeon Kang, and Jaegul Choo. Vegs: View extrapolation of urban scenes in 3d gaussian splatting using learned priors, 2024. 2

[4] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 1, 3, 7, 8

[5] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering, 2023. 2

[6] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollar, and Ross Girshick. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4015–4026, 2023. 1

[7] 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, 2020. 1, 3, 9, 10

[8] Paul-Edouard Sarlin, Cesar Cadena, Roland Siegwart, and Marcin Dymczyk. From coarse to fine: Robust hierarchical localization at large scale. In *CVPR*, 2019. 1

[9] Wangze Xu, Huachen Gao, Shihe Shen, Rui Peng, Jianbo Jiao, and Ronggang Wang. Mvpgs: Excavating multi-view priors for gaussian splatting from sparse input views, 2024. 2, 3

[10] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19447–19456, 2024. 2

[11] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 2024. 1, 2

[12] Baowen Zhang, Chuan Fang, Rakesh Shrestha, Yixun Liang, Xiaoxiao Long, and Ping Tan. Rade-gs: Rasterizing depth in gaussian splatting, 2024. 1, 2, 3