GSRecon: Efficient Generalizable Gaussian Splatting for Surface Reconstruction from Sparse Views

Supplementary Material

1. More Implementation Details

1.1. Network Details

We construct cascade cost volume to capture the scene geometry, thereby providing geometric cues for subsequent Gaussian generation. Specifically, given the image features $\hat{\boldsymbol{F}}$ that capture cross-view correlation through a feature matching transformer [1], we regard each input view as the reference view and others as source views to construct cost volume that encodes the feature matching scores via differentiable warping. The cost volume $\boldsymbol{C}_i \in \mathbb{R}^{1\times D\times H\times W}$ is obtained as:

$$\boldsymbol{C}_{ij} = < \hat{\boldsymbol{F}}_i, \hat{\boldsymbol{F}}_{j\to i} >, \tag{1}$$

where $\hat{\boldsymbol{F}}_i$ is the feature map of i-th reference view, $\hat{\boldsymbol{F}}_{j\to i}$ denotes the warped j-th view feature, and $< \cdot >$ denotes the inner-product of feature vectors at corresponding pixel. To aggregate the cost volume of different source views, we generate a pixel-wise weight map for each cost volume, indicating the reliability of each pixel. Specifically, we first apply a tiny convolution network with three convolution layers on the cost volumes, obtaining the weight volume. We chose the maximum weight along the depth dimension to generate the final weight map. The aggregated cost is calculated by:

$$\boldsymbol{C}_i = \sum_j \boldsymbol{W}_j \boldsymbol{C}_{ij}, \tag{2}$$

where $\boldsymbol{W}_j$ is the weight map of $\boldsymbol{C}_{ij}$. After that, the cost volume is regularized by a 3D CNN to obtain the geometry volume $\boldsymbol{V}_i$ and probability volume $\boldsymbol{P}_i$. The depth map is generated by the winner-take-all strategy and is utilized to generate subsequent cascade cost volume by redefining the depth samples.

2. More Experiments

2.1. Evaluation on *normal* set

In the main paper, we have presented the results of the average Chamfer distance for all scenes of each method on the *Favorable* and *Unfavorable* sets. Here, we report the evaluation results on *normal* image set in Table 1. It can be observed that our method significantly outperforms the previous state-of-the-art method, UFORecon, on the challenging *Unfavorable* set. On the *Favorable* set with small viewpoint variation, our method still exceeds UFORecon by 4%.

2.2. Evaluation on pixel-NeRF set

We note that some methods, such as NeuSurf [3], evaluate their performance on the pixel-NeRF image set with significant viewpoint variations. Here, we also provide evaluation results on the pixel-NeRF image set in Table 2. Notably, NeuSurf is a per-scene optimization method that requires tens of minutes of training to reconstruct a single scene, thus it cannot generalize directly to new scenes. Nonetheless, our generalizable method still outperforms NeuSurf by a large margin.

2.3. Effect of Random Set Training Strategy

To verify the robustness of our method against unseen and challenging view combinations, we select the closest viewpoints during training and use viewpoint combinations with different variations during testing. UFORecon further utilizes a random set training strategy to enhance the performance on the *Unfavorable* set by randomly selecting views during the training phase. To analyze the impact of this strategy, we employ this strategy to baseline models and report the result in Table 3. As demonstrated in the table, most methods benefit from this strategy and exhibit improvements on the *Unfavorable* set. Nonetheless, this strategy inevitably results in a slight decline in performance on the *Favorable* set due to the instability caused by the large viewpoint variations in the training set.

2.4. Impact of the Number of Input Views

We investigate the impact of viewpoint density on our method by varying the number of source views. The results are reported in Table 4, it can be observed that performance gradually improves with increasing viewpoint density. Utilizing more views alleviates challenges associated with reconstructing difficult areas, such as those that are occluded or less visible across views.

2.5. Effect of Normal Loss

To align the Gaussian representations with the ground truth surface, we employ a normal loss to bring the normals of the Gaussian primitives closer to the surface normals. To investigate the impact of the normal loss, we present the performance of retraining our model with and without using normal loss in Table 5. It can be observed that using normal loss during training yields better performance.

Table 1. Quantitative results of sparse *Normal* (medium viewpoint variation) image set reconstruction on 15 testing scenes of DTU dataset. We report the chamfer distance of each scene, the lower the better. The best performance is in boldface and the second best is underlined.

Methods	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
2D GS [2]	3.56	3.38	3.47	1.24	2.70	2.91	2.31	1.88	2.44	1.59	2.60	4.71	0.87	2.62	1.90	2.78
VolRecon [6]	2.63	4.22	2.89	2.49	2.93	2.50	1.68	1.84	2.02	1.76	2.35	2.64	1.16	2.17	1.76	2.34
ReTR [4]	2.06	3.72	2.54	2.51	1.75	2.11	1.49	1.57	1.74	1.35	1.88	2.05	1.00	1.74	1.48	1.93
UFORecon [5]	1.30	2.59	1.51	1.39	**1.04**	**1.28**	0.80	1.37	1.16	0.95	0.98	0.90	0.54	1.06	1.08	1.20
GSRecon (ours)	**1.06**	**2.19**	**1.49**	**1.04**	1.05	1.31	**0.73**	**1.26**	**1.05**	**0.79**	**0.88**	**0.84**	**0.46**	**0.98**	**0.95**	**1.07**

Table 2. Quantitative results of pixel-NeRF (medium viewpoint variation) image set reconstruction on 15 testing scenes of DTU dataset. We report the chamfer distance of each scene, the lower the better. The best performance is in boldface.

Methods	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
SparseNeuS_ft	4.81	5.56	5.81	2.68	3.30	3.88	2.39	2.91	3.08	2.33	2.64	3.12	1.74	3.55	2.31	3.34
MonoSDF	3.47	3.61	2.10	1.05	2.37	1.38	1.41	1.85	1.74	1.10	1.46	2.28	1.25	1.44	1.45	1.86
NeuSurf	**1.35**	3.25	2.50	**0.80**	**1.21**	2.35	**0.77**	**1.19**	1.20	1.05	**1.05**	1.21	**0.41**	**0.80**	**1.08**	1.35
VolRecon [6]	3.05	4.45	3.36	3.09	2.78	3.68	3.01	2.87	3.07	2.55	3.07	2.77	1.59	3.44	2.51	3.02
UFORecon [5]	1.51	2.58	**1.76**	1.35	1.52	**1.80**	1.05	1.57	0.95	1.36	1.15	0.93	0.65	1.24	1.21	1.37
Ours	1.43	**2.28**	1.96	0.94	**1.31**	1.93	0.82	1.25	**0.84**	**0.87**	1.11	**0.73**	0.52	1.10	1.16	**1.21**

Table 3. Effect of adopting random set training strategy. We present the mean chamfer distance on all testing scenes. (*) indicates that the model adopts the random set training strategy.

Method	Favorable	Unfavorable
VolRecon [6]	1.42	3.18
VolRecon* [6]	2.74	3.88
ReTR [4]	1.17	2.94
ReTR* [4]	1.62	2.88
UFORecon [5]	0.99	1.56
UFORecon* [5]	1.01	1.28
GSRecon (ours)	0.95	1.30
GSRecon* (ours)	0.97	1.16

Table 4. Impact of the number of input views.

Number of views	VolRecon	UFORecon	Ours
2	1.72	1.15	1.15
3	1.38	1.00	0.95
4	1.35	0.97	0.94
5	1.33	0.96	0.93

2.6. More Qualitative Results

We provide additional visual comparisons to demonstrate the superiority of our method. Since UFORecon is the first paper focusing on large baseline viewpoint reconstruction, we show a qualitative comparison between our method and UFORecon across *Favorable*, *Normal*, and *Unfavorable* sets in Figure 2 and 3. We then provide more qualitative results on a variety of scenes from the BlendedMVS dataset in Figure 1. The qualitative results demonstrate that our method can produce more accuracy and complete surface than other methods under various view combinations across different scenarios.

Table 5. Effect of normal loss.

Method	Favorable	Unfavorable
w/o $\mathcal{L}_n$	0.98	1.38
w/ $\mathcal{L}_n$	0.95	1.30

3. Limitation

Our method constructs a cost volume for each view, so the memory overhead increases with the number of views, which makes it unable to perform reconstruction from dense views. A feasible approach to tackle this issue is to perform progressive reconstruction, similar to [7].

References

[1] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *CVPR*, pages 8585–8594, 2022. 1

[2] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically

Figure 1. More qualitative comparison of different methods using 3 unfavorable views on BlendedMVS dataset.

accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[3] 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 *AAAI*, pages 2312–2320, 2024. 1

[4] Yixun Liang, Hao He, and Yingcong Chen. Retr: Modeling rendering via transformer for generalizable neural surface reconstruction. In *NeurIPS*, pages 62332–62351, 2023. 2

[5] Youngju Na, Woo Jae Kim, Kyu Beom Han, Suhyeon Ha, and Sung-eui Yoon. Uforecon: Generalizable sparse-view surface reconstruction from arbitrary and unfavorable data sets. *CVPR*, 2024. 2

[6] Yufan Ren, Tong Zhang, Marc Pollefeys, Sabine Süsstrunk, and Fangjinhua Wang. Volrecon: Volume rendering of signed ray distance functions for generalizable multi-view reconstruction. In *CVPR*, pages 16685–16695, 2023. 2

[7] Jiaming Sun, Yiming Xie, Linghao Chen, Xiaowei Zhou, and Hujun Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *CVPR*, pages 15598–15607, 2021. 2

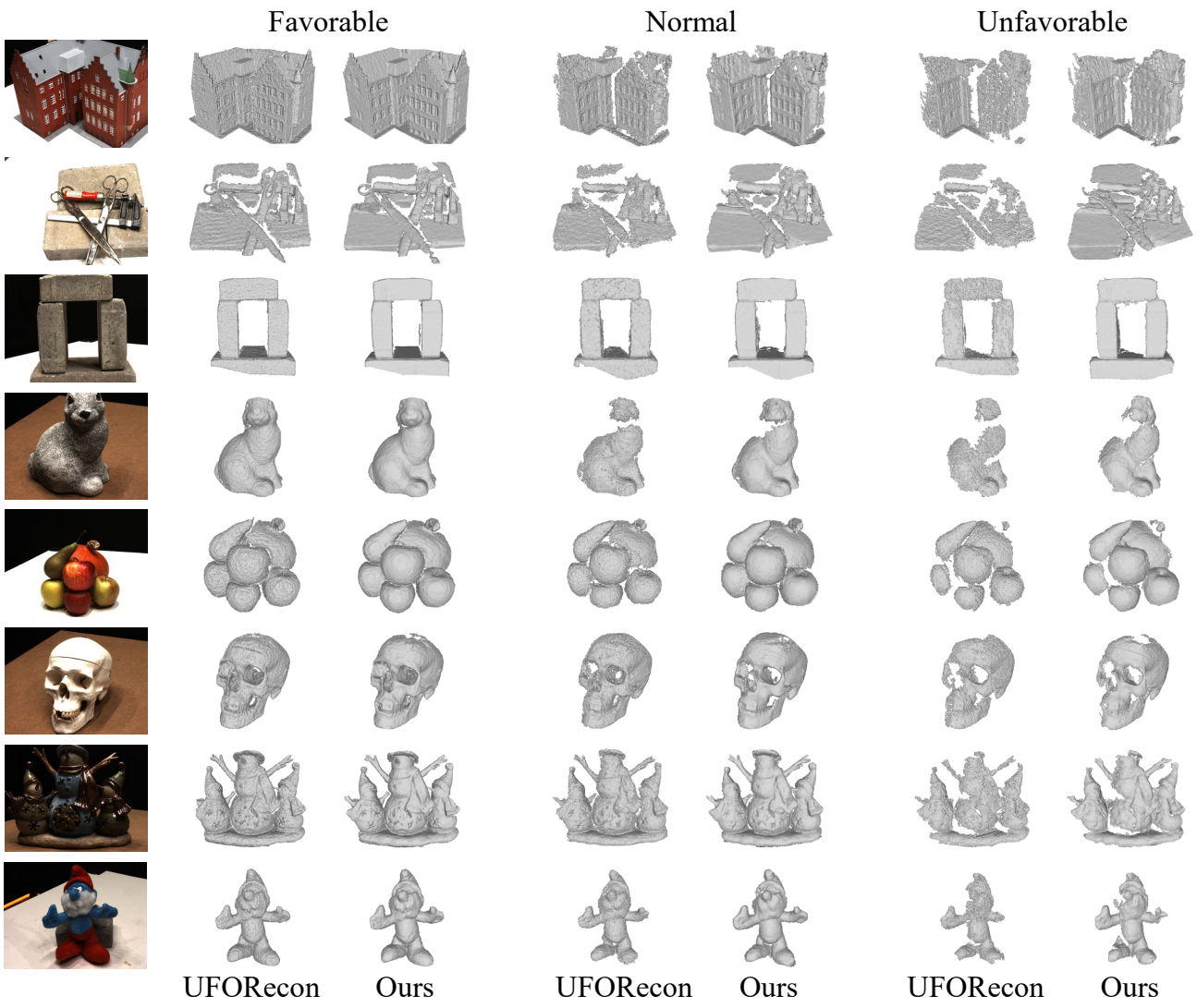

Figure 2. Reconstruction results across various view-combination sets.

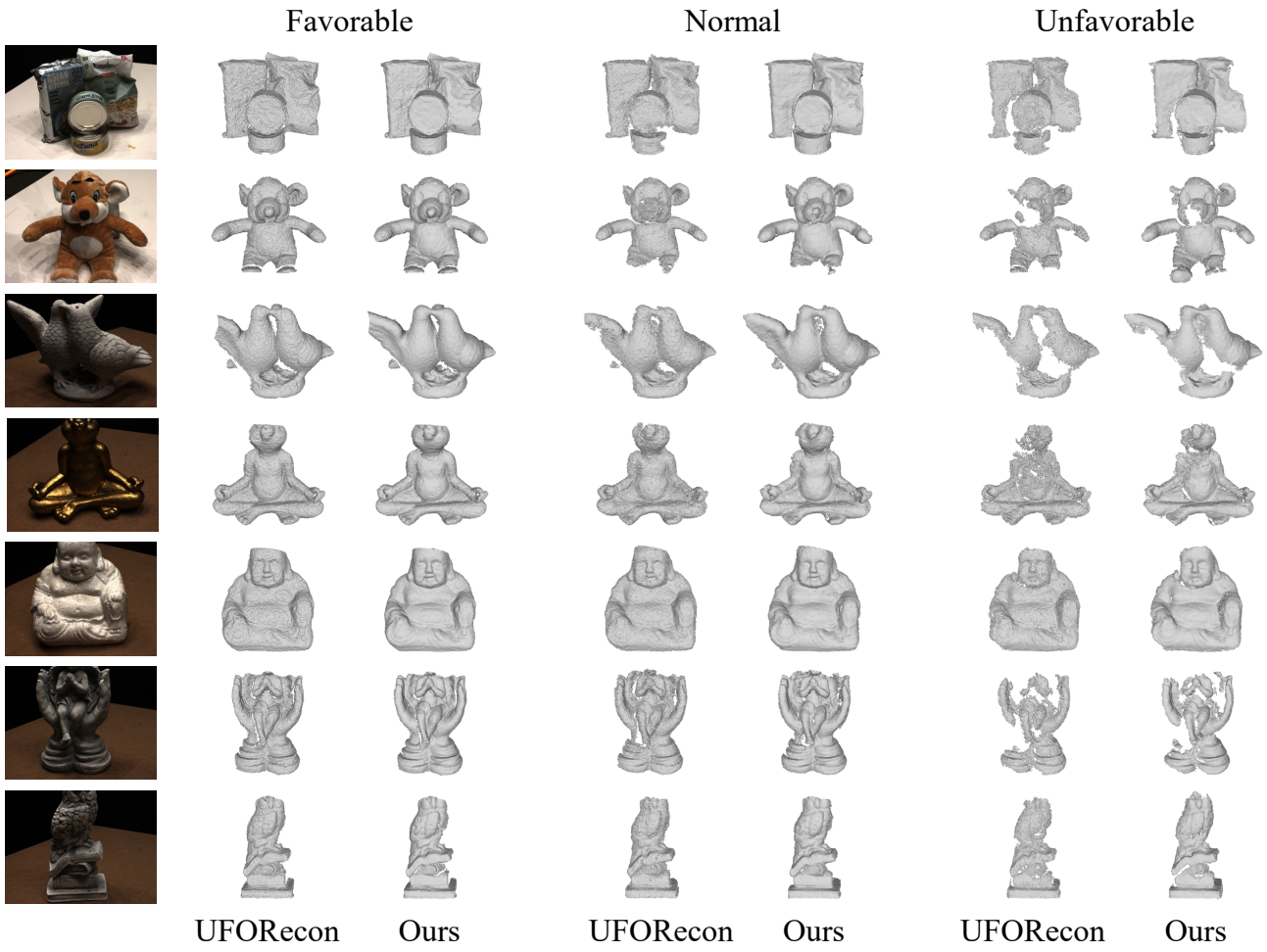

Figure 3. Reconstruction results across various view-combination sets.