

SUPPLEMENTARY DOCUMENT FOR ENHANCING SPARSE-VIEW 3D GAUSSIAN SPLATTING WITH GUIDANCE OF NORMALS PRIORS AND DENSE POINT INITIALIZATION

Yi-Huang Hsieh I-Chen Lin

College of Computer Science, National Yang Ming Chiao Tung University

This supplementary document provides additional information that supports the findings presented in the main paper, including additional ablation study, experiments, discussions and visual comparisons.

1. ADDITIONAL EXPERIMENTS

1.1. Efficiency

Table 1 compares the computational efficiency of our method with related representative frameworks, using random initialization and COLMAP MVS initialization, based on their released codes. We measured execution time across three stages: initialization, training, and rendering.

1.2. Comparison on T&T

The T&T dataset consists of large-scale real-world scenes captured with an industrial laser scanner. Unlike object-centric or room-scale scenes, these large-scale scenes comprise complex spatial layouts, requiring methods to handle long-range visibility and substantial variations in geometric complexity. Due to these properties, T&T is difficult under sparse-view conditions since the differences in angles between each input view are larger than those of regular datasets. To verify the effectiveness of our method on large-scale scenes, we conducted experiments on the T&T dataset using the same split approach as LLFF without downsampling images. Table 2 presents quantitative results with three training views. As shown in Fig. 1, while CoR-GS and DropGaussian fail to preserve fine details due to their respective regularization strategies, our approach successfully balances

Method	Initialization	Training	Rendering
DNGaussian	0.094	93.282	15.363
CoR-GS	1690.17	270.962	7.220
Ours	62.59	74.775	3.873

Table 1: Comparison of per-scene execution time (in seconds) of DNGaussian [1], CoR-GS [2], and our method on LLFF dataset.

Method	PSNR	SSIM	LPIPS	AVGE
DNGaussian	17.46	0.529	0.472	0.183
CoR-GS	19.22	0.656	0.340	0.137
DropoutGS	17.82	0.543	0.474	0.181
DropGaussian	19.64	0.658	0.316	0.129
Ours	21.73	0.742	0.249	0.098

Table 2: Quantitative evaluation on T&T with three training views. Cell colors denote **best**, **second-best**, and **third-best** results. Since this dataset is difficult and has not been evaluated by related sparse-view methods, the scores are obtained from models trained by ourselves with released codebases.

global structure reconstruction and local detail preservation in both foreground and background objects.

2. ADDITIONAL ABLATION STUDY

2.1. Different Initialization Strategies

We evaluated our framework using random initialization, COLMAP MVS initialization [3], and our DFtri initialization. Table 3 presents the quantitative results. Initialization with DFtri achieves the best performance in terms of PSNR, SSIM, and LPIPS scores compared to both baselines. This advantage stems from the ability of DFtri to form denser point clouds with improved coverage of co-visible regions. The study demonstrates the effectiveness of leveraging dense cor-

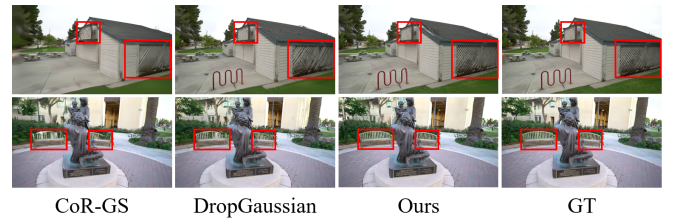


Fig. 1: Qualitative comparisons of novel view synthesis on T&T with three input views.

Method	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	AVGE $\downarrow$
w/ random init.	16.16	0.450	0.368	0.195
w/ MVS init.	20.58	0.665	0.231	0.111
w/ DFtri	21.18	0.757	0.161	0.090

Table 3: Quantitative results of the ablation study on initialization strategies. We performed our framework with three different initialization methods on LLFF dataset with three input views.



Fig. 2: Visual comparison of COLMAP MVS and our DFtri on LLFF with three input views.

respondences from the pre-trained feature matching model. Fig. 2 illustrates the substantial difference in point density and coverage between COLMAP MVS and our DFtri.

2.2. Ablation Study of the Primary Components

To evaluate the effectiveness of our designs, we conducted ablation studies on the DTU dataset with three input views. Quantitative results are presented in Table 4. We use vanilla 3DGS with random initialization as the baseline. With dense point initialization providing comprehensive coverage of co-visible areas, even the vanilla 3DGS can achieve impressive performance. Incorporating normal priors yields further performance gains, confirming that normal priors provide essential geometric information for learning accurate geometric structure.

Normal priors shows their effectiveness in handling texture-poor scenes in the DTU dataset, where DFtri faces challenges. Fig. 3 demonstrates that 3DGS struggles to reconstruct textureless regions even with our improved initialization. The incorporation of normal priors significantly enhances reconstruction quality in these challenging areas by providing crucial geometric guidance.

2.3. Impact of Taking Confidence Maps as Regional Weights

In our normal reconstruction loss functions, we introduce per-pixel weights to adjust the influences of individual pixels. We thought that the predictions of deep-learning-based normal estimators tend to be unstable or inaccurate at edge regions. It

Components		PSNR	SSIM	LPIPS	AVGE
DFtri	Norm.P.				
		15.28	0.681	0.257	0.167
	✓	16.60	0.735	0.226	0.144
✓		19.38	0.842	0.126	0.088
✓	✓	19.91	0.860	0.115	0.082

Table 4: Quantitative results of ablation studies on our proposed components on DTU with three input views. Norm.P. stands for normal priors.

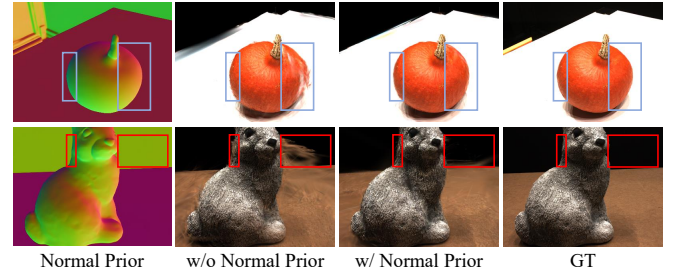


Fig. 3: Qualitative results of the ablation study on normal priors.

should benefit the performance if we apply spatially-varying weights to adjust the contributions of normals in different areas.

We take the confidence values as the corresponding weights for the surface normal reconstruction loss $\mathcal{L}_{\text{snorm}}$ and improve the performance. On the other hand, since the error distribution of surface depth gradients may not be consistent with those of normal predictions, we found that applying the aforementioned confidences as weights for the depth-to-normal reconstruction loss $\mathcal{L}_{\text{dnorm}}$ did not improve the performance compared to its equal-weight counterpart $\mathcal{L}_{\text{dnorm}}^*$. Table 5 compares four settings regarding applying confidence weights or identical weights to the surface normal loss and depth-to-normal loss.

Settings	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	AVGE $\downarrow$
$\mathcal{L}_{\text{snorm}}^* + \mathcal{L}_{\text{dnorm}}^*$	21.00	0.747	0.172	0.094
$\mathcal{L}_{\text{snorm}} + \mathcal{L}_{\text{dnorm}}^*$	21.18	0.757	0.161	0.090
$\mathcal{L}_{\text{snorm}}^* + \mathcal{L}_{\text{dnorm}}$	20.85	0.741	0.175	0.096
$\mathcal{L}_{\text{snorm}} + \mathcal{L}_{\text{dnorm}}$	21.12	0.753	0.161	0.091

Table 5: Quantitative results of impact of confidence weights on LLFF dataset with three input views. * denotes that the loss function is applied without taking the confidence maps as weights. We adopt $\mathcal{L}_{\text{snorm}} + \mathcal{L}_{\text{dnorm}}^*$ in our framework.

Method(Data)	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	AVGE $\downarrow$
Ours (LLFF)	21.18	0.757	0.161	0.090
Our+D(LLFF)	21.31	0.757	0.164	0.090
Ours (DTU)	19.91	0.860	0.115	0.082
Our+D(DTU)	18.59	0.844	0.135	0.095
Ours (T&T)	21.73	0.742	0.249	0.098
Our+D(T&T)	21.55	0.739	0.258	0.101

Table 6: Quantitative results of our proposed method (Ours) and our framework with an additional depth prior (Our+D) on LLFF, DTU and T&T datasets with three input views.

2.4. Study on the Effect of Adding Depth Prior to Our Framework

Our method proposes using normal priors instead of depth prior to guide the 3DGS reconstruction under sparse-view conditions. We also conducted an experiment, in which the monocular depth is added as an additional prior to our framework. In this additional experiment, we incorporated the depth-prior-related losses proposed in DNGaussian [1] to our framework.

As shown in Table 6, for the LLFF dataset, addition depth prior slightly increased PSNR but degraded the LPIPS score. For the DTU, the performance apparently decreased when the depth prior was included. For T&T dataset, the results of our proposed method were still better than results by our framework with additional depth prior. Based on this study, we believe that incorporating normal priors is adequate to obtain geometric information and beneficial for sparse-view 3DGS estimation.

3. STUDY ON CONDITIONS OF SIX AND NINE INPUT VIEWS

Our method is designed for sparse view conditions. In the main paper, we have compared its capability with related methods for three input views. This section presents quantitative comparisons across all the three datasets using six-view and nine-view settings. For qualitative comparisons, we mainly show six-view results, as nine-view reconstructions generally achieve adequate quality across all methods, making visual differences less distinguishable. The six-view setting represents a middle-ground in which more geometric cues are revealed compared to the three-view setting, and it remains challenging compared to the nine-view condition. For each dataset, we show synthesis results by our method and the other two available best-performing related methods.

3.1. Comparison on LLFF with More Input Views

As shown in Table 7, we achieve the best performance across all metrics with six input views and the best SSIM, LPIPS,

Method	6-View				9-View			
	PSNR	SSIM	LPIPS	AVGE	PSNR	SSIM	LPIPS	AVGE
DNGaussian	22.18	0.755	0.198	0.088	23.17	0.788	0.180	0.077
FSGS	24.20	0.811	0.173	-	25.32	0.856	0.136	-
CoR-GS	24.49	0.837	0.115	0.060	26.06	0.874	0.089	0.046
FewViewGS	24.35	0.826	0.126	-	25.90	0.868	0.095	-
DropoutGS	23.35	0.791	0.177	0.076	24.33	0.825	0.160	0.066
DropGaussian	24.74	0.837	0.117	-	26.21	0.874	0.088	-
Ours	24.93	0.852	0.096	0.054	26.02	0.877	0.079	0.045

Table 7: Quantitative comparisons of conditions with six and nine input views on LLFF dataset.

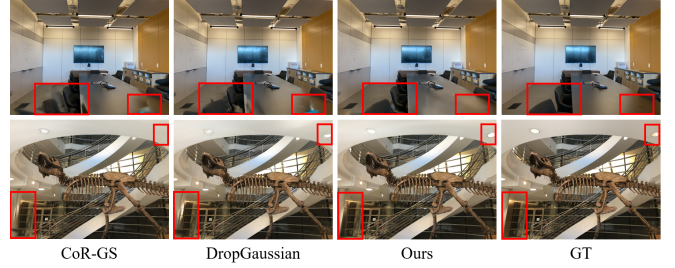


Fig. 4: Qualitative comparison among CoR-GS [2], DropGaussian [4], and our method on LLFF dataset with six input views.

and AVGE scores with nine input views. In the nine-view setting, CoR-GS [2] and DropGaussian [4] achieve higher PSNR scores than our method. This can be attributed to their prior-free nature, which becomes advantageous with increased input views as they avoid potential errors introduced by priors. Nevertheless, our method maintains higher perceptual quality, achieving the best SSIM and LPIPS scores while remaining competitive in PSNR. As shown in Fig. 4, our method can reconstruct more accurate geometry and capture finer scene details compared to CoR-GS and DropGaussian.

3.2. Comparison on DTU with More Input Views

Table 8 presents the quantitative results. Our method achieves the best LPIPS scores in the six-view setting and the second-best in the nine-view setting, demonstrating superior perceptual quality. The SSIM differences between our method and the top-performing approaches are marginal in both settings. Notably, we obtain the best PSNR score with six input views.

Method	6-View				9-View			
	PSNR	SSIM	LPIPS	AVGE	PSNR	SSIM	LPIPS	AVGE
DNGaussian	22.60	0.864	0.130	0.068	25.17	0.907	0.104	0.048
FSGS	22.54	0.896	0.100	0.065	24.95	0.926	0.076	0.046
CoR-GS	24.51	0.917	0.068	0.046	27.18	0.947	0.045	0.029
FewViewGS	24.33	0.920	0.069	-	27.31	0.953	0.041	-
DropoutGS	23.76	0.882	0.133	0.062	25.63	0.914	0.115	0.046
DropGaussian	18.93	0.831	0.179	0.106	22.66	0.899	0.118	0.061
Ours	24.55	0.916	0.068	0.045	26.90	0.947	0.044	0.030

Table 8: Quantitative comparison of conditions with six and nine input views on DTU dataset.

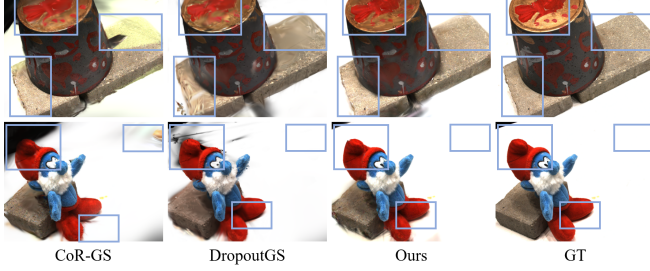


Fig. 5: Qualitative comparison among CoR-GS [2], DropoutGS [5], and our method on DTU with six input views.

Method	6-View				9-View			
	PSNR	SSIM	LPIPS	AVGE	PSNR	SSIM	LPIPS	AVGE
DNGaussian	20.78	0.645	0.427	0.131	21.30	0.660	0.428	0.125
CoR-GS	26.28	0.852	0.172	0.055	28.50	0.894	0.131	0.040
DropoutGS	22.22	0.678	0.407	0.114	22.49	0.680	0.414	0.113
DropGaussian	26.67	0.856	0.170	0.053	28.60	0.895	0.137	0.040
Ours	27.58	0.876	0.137	0.045	29.47	0.909	0.104	0.034

Table 9: Quantitative comparison of conditions with six and nine input views on T&T.

As shown in Figure 5, CoR-GS and DropoutGS suffer from blurry and floating artifacts despite the additional input views. These issues might be inherent to their regularization mechanisms. In contrast, our method successfully learns accurate textures and geometric structures while minimizing floaters, thereby achieving superior quality for novel view synthesis.

3.3. Comparison on T&T with More Input Views

Table 9 presents our quantitative results, demonstrating that our method achieves the best performance across all metrics and view settings. Fig. 6 shows that our method can generate more accurate geometry and details for large-scale outdoor and indoor scenes with significant angular variations between input views.

4. LIMITATIONS AND FUTURE WORK

While our method advances on benchmark datasets including LLFF, DTU, and T&T through dense point initialization and

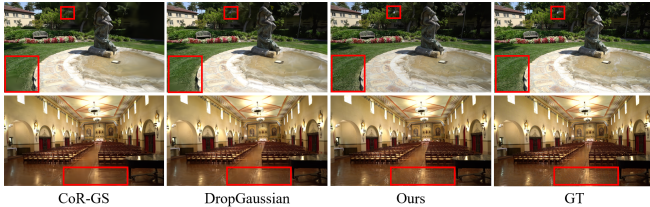


Fig. 6: Qualitative comparison among CoR-GS [2], DropGaussian [4], and our method on T&T with six input views.



Fig. 7: Failure cases where novel views are far from the input views.

normal-based optimization, several limitations in our work reveal opportunities for future research.

Our framework makes use of pre-trained models for dense feature matching and monocular normal map prediction. Inadequate priors predicted from these models would degrade the synthesis quality or produce artifacts. We believe that our proposed framework and results can be further improved by introducing more advanced predictors of feature correspondences and normal maps. For novel views that are far from the input training views, our approach still struggles to reconstruct them and tends to produce artifacts since there is no sufficient geometric cues from the inputs, as shown in Fig. 7. We notice that generative priors are extremely helpful in these scenarios. Therefore, we will explore the possibility of incorporating powerful generative models to further enhance these views.

5. REFERENCES

- [1] 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*, 2024, pp. 20775–20785.
- [2] Jiawei Zhang, Jiahe Li, Xiaohan Yu, Lei Huang, Lin Gu, Jin Zheng, and Xiao Bai, “Cor-gs: sparse-view 3d gaussian splatting via co-regularization,” in *European Conference on Computer Vision*. Springer, 2024, pp. 335–352.
- [3] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang, “Fsgs: Real-time few-shot view synthesis using gaussian splatting,” in *European conference on computer vision*. Springer, 2024, pp. 145–163.
- [4] Hyunwoo Park, Gun Ryu, and Wonjun Kim, “Dropgaussian: Structural regularization for sparse-view gaussian splatting,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 21600–21609.
- [5] Yexing Xu, Longguang Wang, Minglin Chen, Sheng Ao, Li Li, and Yulan Guo, “Dropouts: Dropping out gaussians for better sparse-view rendering,” in *Proceedings of the Computer Vision and Pattern Recognition Conference*, 2025, pp. 701–710.