

CoR-GS: Sparse-View 3D Gaussian Splatting via Co-Regularization

Jiawei Zhang¹, Jiahe Li¹, Xiaohan Yu³, Lei Huang², Lin Gu^{4,5}, Jin Zheng^{1*},
and Xiao Bai^{1*}

¹ School of Computer Science and Engineering, State Key Laboratory of
Complex & Critical Software Environment, Jiangxi Research Institute,
Beihang University

² SKLCCSE, Institute of Artificial Intelligence, Beihang University

³ School of Computing, Macquarie University

⁴ RIKEN AIP

⁵ The University of Tokyo

Abstract. 3D Gaussian Splatting (3DGS) creates a radiance field consisting of 3D Gaussians to represent a scene. With sparse training views, 3DGS easily suffers from overfitting, negatively impacting rendering. This paper introduces a new co-regularization perspective for improving sparse-view 3DGS. When training two 3D Gaussian radiance fields, we observe that the two radiance fields exhibit point disagreement and rendering disagreement that can unsupervisedly predict reconstruction quality, stemming from the randomness of densification implementation. We further quantify the two disagreements and demonstrate the negative correlation between them and accurate reconstruction, which allows us to identify inaccurate reconstruction without accessing ground-truth information. Based on the study, we propose CoR-GS, which identifies and suppresses inaccurate reconstruction based on the two disagreements: (i) Co-pruning considers Gaussians that exhibit high point disagreement in inaccurate positions and prunes them. (ii) Pseudo-view co-regularization considers pixels that exhibit high rendering disagreement are inaccurate and suppress the disagreement. Results on LLFF, Mip-NeRF360, DTU, and Blender demonstrate that CoR-GS effectively regularizes the scene geometry, reconstructs the compact representations, and achieves state-of-the-art novel view synthesis quality under sparse training views. Project page: <https://jiaw-z.github.io/CoR-GS>

Keywords: 3d gaussian splatting · sparse-view novel view synthesis

1 Introduction

Obtaining 3D representations from 2D images has long been a topic of interest. 3D Gaussian Splatting (3DGS) [15] creates unstructured radiance fields consisting of a set of 3D Gaussians to represent the scene and has achieved high-quality

* Corresponding author: Xiao Bai and Jin Zheng

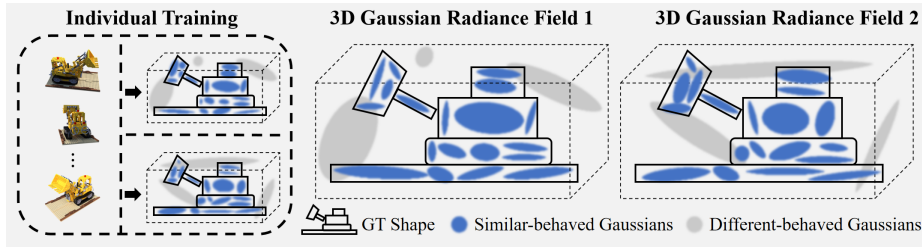


Fig. 1: Illustration of how the different behaviors between two 3D Gaussian radiance fields correlated to construction quality. Gaussians with different behaviors tend to not fit the ground-truth shape well. Therefore, inaccurate reconstructions can be identified by measuring the differences without accessing ground-truth information.

novel view synthesis in real time. However, when only sparse views are available, the decrease in training constraints makes 3DGS prone to overfit training views [40, 51], resulting in unrealistic novel view synthesis.

3DGS initializes 3D Gaussians with a sparse point cloud. Then an interleaved optimization/density control of 3D Gaussians is performed to achieve an accurate radiance field representation of the scene. Under sparse training views, we observe that although two 3D Gaussian radiance fields are trained to represent the same scene, they exhibit differences in both Gaussian positions and rendered pixels. Their differences significantly increase during density control, which involves creating new Gaussians and initializing their positions by sampling from a normal distribution. With sparse training views, the optimization can struggle to correct Gaussians to accurately represent the scene due to the ambiguities of 3D to 2D projection, resulting in an accumulation of differences.

In this paper, we reveal the link between the different behaviors of two 3D Gaussian radiance fields and their reconstruction quality. Specifically, we propose point disagreement and rendering disagreement to indicate the differences and measure them quantitatively. The point disagreement indicates the differences in the Gaussian position, which is evaluated on the registration between Gaussians’ point cloud representations. The rendering disagreement indicates the differences in their rendered pixels. We compare the rendered images to ground-truth test views to measure the reconstruction quality of 3D Gaussian radiance fields. We also utilize a 3D Gaussian field trained with dense views as ground truth to evaluate Gaussian positions and rendered depth maps to provide a more comprehensive assessment of reconstruction quality. More details are provided in Sec. 3. The experimental results demonstrate a negative correlation between the two disagreements and accurate reconstruction. This allows us to unsupervisedly identify inaccurate reconstruction by comparing two 3D Gaussian radiance fields, as illustrated in Fig. 1.

Based on our study, we propose CoR-GS, which trains two 3D Gaussian radiance fields with the same views and conducts co-regularization during training. It improves sparse-view 3DGS by identifying and suppressing inaccurate recon-

struction based on the point disagreement and rendering disagreement. CoR-GS implies co-pruning to suppress point disagreement. Co-pruning treats two 3D Gaussian radiance fields as two point clouds, performing point-wise matching between them. It considers Gaussians that do not have nearby matching points in the opposite point cloud as outliers and prunes them. To suppress rendering disagreement, CoR-GS implies pseudo-view co-regularization. It samples online pseudo views by interpolating training views and considers pixels that exhibit high rendering disagreement are inaccurately rendered. To suppress the inaccurate rendered results, it computes the differences of rendered pixels as a regularization term combined with the training-view loss.

Integrating co-pruning and pseudo-view co-regularization, CoR-GS reconstructs coherent and compact geometry and achieves state-of-the-art sparse-view rendering performance on LLFF, Mip-NeRF360, DTU, and Blender datasets. The experiments demonstrate our method’s universal ability to regularize sparse-view 3DGS in various scene situations.

Our main contributions are the following:

- We propose point disagreement and rendering disagreement to measure the differences between two 3D Gaussian radiance fields for the same scene and demonstrate the negative correlation between the two kinds of disagreement and accurate reconstruction. The two agreements can be used to access the reconstruction quality without ground-truth information.
- We propose co-pruning and pseudo-view co-regularization for suppressing the point disagreement and rendering disagreement, respectively. We demonstrate that suppressing the two disagreements leads to more accurate 3D Gaussian radiance fields for representing the scene.
- Equipped with co-pruning and pseudo-view co-regularization, CoR-GS reconstructs coherent and compact geometry and achieves competitive quality across multiple benchmarks compared to the state-of-the-art methods.

We hope the observations and discussions in this paper can stimulate further thinking on the randomness of 3D Gaussian radiance fields.

2 Related Work

2.1 Radiance Fields

Radiance Fields are employed for reconstructing 3D scenes and synthesizing novel views. Neural Radiance Fields (NeRFs) [24] have experienced significant advancements, which learn the neural volumetric representations to represent 3D scenes and render images through volume rendering. Since then, a great number of studies aim to improve the rendering quality [2, 3] and efficiency [5, 10, 12, 21, 25, 32, 43] for NeRFs. Recent advancements in unstructured radiance fields, as demonstrated by studies like [7, 15, 41], employ a collection of Gaussians to represent 3D scenes. Notably, 3D Gaussian Splatting (3DGS) [15] represents scenes using a set of anisotropic 3D Gaussians, and renders images

via differentiable splitting. 3DGS has demonstrated remarkable success in reconstructing high-quality complex scenes in real time. Despite the tremendous success of 3DGS in many 3D tasks [19, 22, 33, 38], the behavior of 3DGS with sparse views is less investigated, posing an open problem in the field.

2.2 Novel View Synthesis with Sparse Views

Novel view synthesis aims to generate unseen views of objects or scenes from a set of given images [1, 49]. While given sparse views, many methods tend to reconstruct degraded scenes and render unrealistic novel views. Many studies have explored regularizing NeRFs with sparse views [9, 16, 26, 42]. Some methods focus on designing generative models and pre-train them on extensive datasets [6, 8, 17, 44, 50]. Conversely, others [9, 13, 27, 31, 35, 39] utilize the external knowledge from pre-trained models to regularize the training process.

With the advancements of 3DGS, recent studies [18, 40, 51] have observed the issue of 3DGS to synthesize unrealistic novel views in sparse-view scenarios. These methods rely on predictions of pre-trained depth estimators as regularization to correct the reconstructed geometry. However, external supervision can introduce additional noise, negatively impacting the reconstruction. Our paper introduces a new co-regularization perspective for sparse-view 3DGS by suppressing the disagreement between two 3D Gaussian radiance fields.

2.3 Prediction Agreement

The agreement of predictions of two neural networks has been explored in many tasks [4, 11, 20, 30, 37, 45, 46], especially for semi-supervised learning [30] and learning with noisy labels [11]. The typical pipeline involves simultaneously training two neural networks. They leverage the agreement of different networks' predictions to pseudo-label unlabeled data [20] or clean noise from labeled annotations [11], thereby training networks with improved generalization capabilities. Our distinction from previous studies lies in our investigation of disagreement of unstructured 3D Gaussian radiance fields instead of generalized neural networks, where we consider Gaussian positions and rendered results.

3 Point Disagreement and Rendering Disagreement

3.1 Preliminary and Definition

3D Gaussian Splatting. 3D Gaussian splatting [15] represents a scene with a set of 3D Gaussians. The i -th Gaussian primitive can be described as $\theta_i = \{\mu_i, s_i, q_i, \alpha_i, f_i\}$, where $\mu_i \in \mathbb{R}^3$ is the center, $s_i \in \mathbb{R}^3$ is the scaling factor, $q_i \in \mathbb{R}^4$ is the rotation, $\alpha_i \in \mathbb{R}$ is the opacity for rendering and $f_i \in \mathbb{R}^K$ is the K -dimensional color feature. In the three-dimensional space, the influence of the i -th Gaussian primitive on the position x is defined as $G_i(x)$:

$$G_i(x) = e^{-\frac{1}{2}(x-\mu_i)^T \Sigma_i^{-1}(x-\mu_i)}, \quad (1)$$

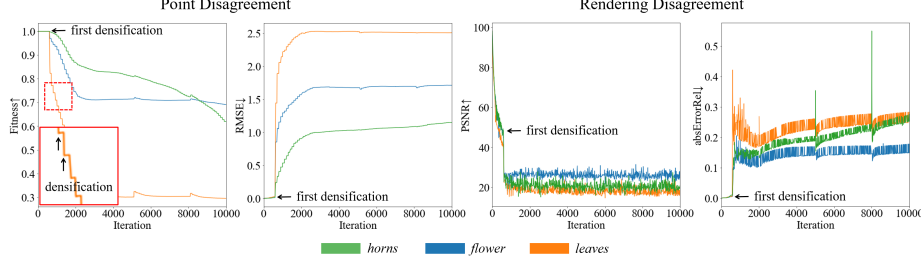


Fig. 2: The recorded different behaviors of two 3d Gaussian radiance fields during training. The point disagreement and rendering disagreement increases during training, especially during densification.

where the covariance matrix Σ is calculated from the scale s and rotation q . For color rendering, 3D Gaussian Splatting orders all the Gaussians that contribute to each 2D-plane pixel and renders through alpha-blending. For optimization, 3DGS optimizes the parameters θ for Gaussians through gradient descent and conducts densification to identify and duplicate active Gaussians. 3DGS [15] suggests utilizing the point cloud from COLMAP [28, 29] or other SfMs for initialization. However, the number of initial points is limited under sparse views, thus we follow FSGS [51] to conduct a stereo-fusion-based initialization.

Point disagreement. When considering the 3-axis position of Gaussians, Gaussians in two radiance fields can be regarded as two point clouds. We evaluate their differences using Fitness and root mean square error (RMSE), which are used to evaluate the registration between point clouds [48]. Fitness calculates the overlapping area between Gaussians concerning a max distance $\tau = 5$ for correspondence. RMSE computes the average distance of correspondent points.

Rendering disagreement. We evaluate the differences by considering both rendered images and depth maps. We use PSNR to compute differences between two rendered images. For depth maps, we use Relative Absolute Error (absErrorRel), which is commonly used to evaluate the accuracy of depth values [34].

Evaluation for reconstruction quality. The most popular method to evaluate the reconstruction is to compare rendered images with ground-truth images at test views. We follow this evaluation and use the PSNR metric. To provide a more comprehensive evaluation, we obtain the ground-truth Gaussian positions and depth maps by training a 3D Gaussian radiance field with dense views for each scene. We use Fitness, RMSE, and absErrorRel for evaluation.

3.2 Empirical Study

Two 3D Gaussian radiance fields trained with the same sparse views can exhibit different behaviors. We simultaneously train two 3D Gaussian radiance fields and record the disagreements during training. Since ground-truth supervision is directly imposed on training views, we evaluate the rendering disagreement at unseen views. As shown in Fig. 3, the two 3D Gaussian radiance

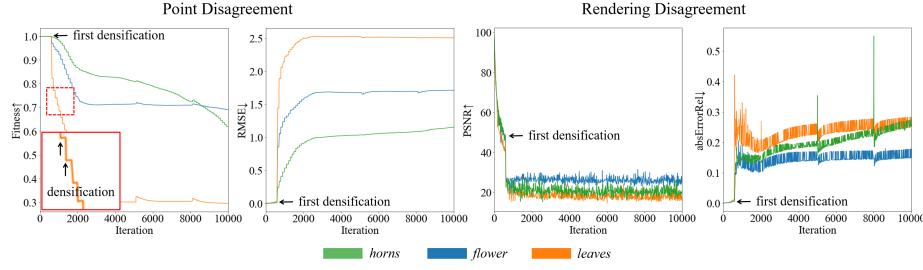


Fig. 3: The recorded different behaviors of two 3d Gaussian radiance fields during training. The point disagreement and rendering disagreement increases during training, especially during densification.

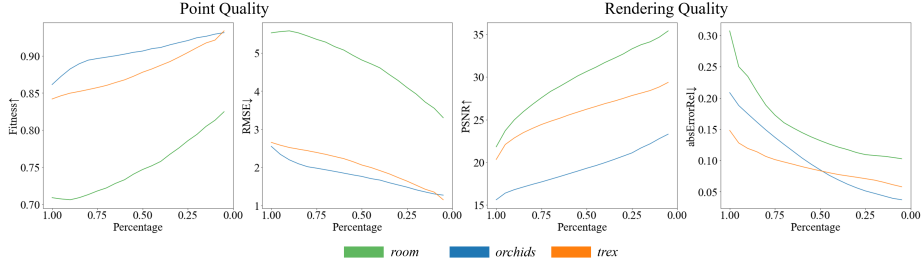


Fig. 4: The correlation between the two disagreements and reconstruction quality. The x-axis percentage represents we mask out the percentage of regions with the highest disagreement scores. With the reduction of regions with higher disagreement scores, the reconstruction quality averaging the remaining regions continuously improves.

fields exhibit different behaviors. In particular, the two disagreements grow significantly during densification, which creates new Gaussians but locates them blindly to the scene geometry. Therefore, the randomness during densification can be a source of error geometry in sparse-view 3DGS.

Point disagreement and Rendering disagreement are negatively correlated with accurate reconstruction of the scene. Based on observing different behaviors between two 3D Gaussian radiance fields, we further investigate whether their disagreements imply certain correlations concerning the reconstructed geometry quality. We mask out areas with a certain percentage with the highest disagreement scores across two 3D Gaussian radiance fields and calculate the reconstruction quality of the remaining regions. As shown in Fig. 4, with the reduction of regions with higher disagreement scores, the reconstruction quality of the remaining regions continuously improves. This demonstrates a negative correlation between the disagreement and accurate reconstruction. Therefore, we can identify the inaccurate reconstruction based on the disagreements even without ground-truth information. We provide the test-view visualization of rendered results with the corresponding error maps in Fig. 5. We

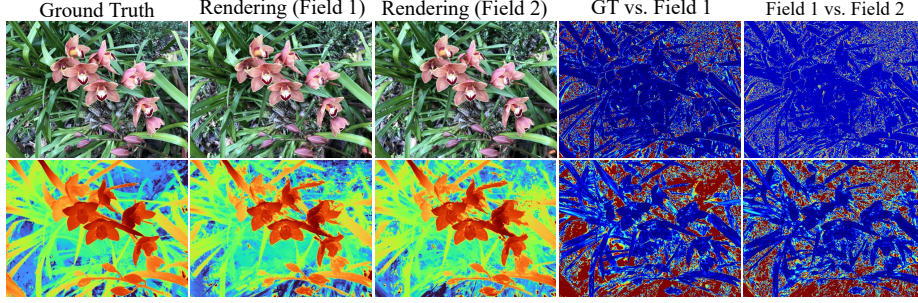


Fig. 5: Test-view visualization of rendered images and depth maps, and the corresponding error maps. The disagreed regions between the rendered results of two 3D Gaussian radiance fields tend to be inaccurate compared to ground truth.

can observe that the disagreed regions between the rendered results of two 3D Gaussian radiance fields tend to be inaccurate compared to ground truth.

4 Method

CoR-GS identifies and suppresses the inaccurate reconstruction based on both point disagreement and rendering disagreement. We simultaneously train two 3D Gaussian radiance fields $\Theta^1 = \{\theta_i^1 | i = 1, 2, \dots, N^1\}$ and $\Theta^2 = \{\theta_i^2 | i = 1, 2, \dots, N^2\}$, where N^1 and N^2 are the number of Gaussians. After training, we keep one 3D Gaussian radiance field for inference. In the following content, we omit the process for the second field Θ^2 , which is in the same manner for Θ^1 .

4.1 Co-pruning

The sampling implementation in densification creates new Gaussians blind to the geometry and can be hard to correct with sparse views. Based on the point disagreement, co-pruning identifies and prunes Gaussians that are located in inaccurate positions. We first find the matching correspondence $f : \Theta^1 \rightarrow \Theta^2$ based on their nearest Gaussians in the other set:

$$f(\theta_i^1) = \text{KNN}(\theta_i^1, \Theta^2). \quad (2)$$

We set a permissible maximum distance $\tau = 5$ to calculate the non-matching mask M :

$$M_i = \begin{cases} 1, & \sqrt{(\theta_{xi}^1 - f(\theta_i^1)_x)^2 + (\theta_{yi}^1 - f(\theta_i^1)_y)^2 + (\theta_{zi}^1 - f(\theta_i^1)_z)^2} > \tau, \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where $\theta_{xi}^1, \theta_{yi}^1, \theta_{zi}^1$ represent the 3-axis positions of Gaussians. After the calculation of the non-matching mask for both sets of Gaussians, we prune all Gaussians

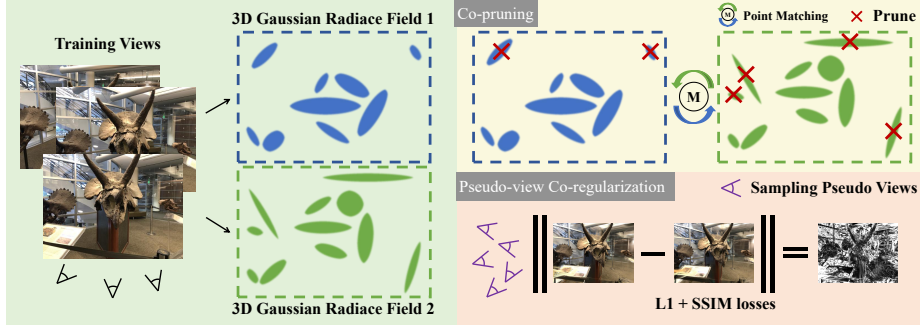


Fig. 6: Overview of CoR-GS. We train two 3D Gaussian radiance fields simultaneously and regularize them by suppressing point disagreement and rendering disagreement.

that are marked as non-matching, which are considered in inaccurate positions. We perform co-pruning every certain number (we set it to 5) of the optimization/density control interleaves, as we aim to prune Gaussians that optimization cannot handle.

4.2 Pseudo-view Co-regularization

Sampling pseudo views. The online pseudo view is sampled from the two nearest training views in Euclidean space, following previous studies [51].

$$P' = (t + \epsilon, q). \quad (4)$$

Here, $t \in P$ denotes the camera location of training views, ϵ is the random noise sampled from a normal distribution and q is a quaternion representing the rotation averaged from the two training cameras.

Color co-regularization. At the sampled pseudo view, we render two images I'^1 and I'^2 using each of the two Gaussian primitive sets Θ^1 and Θ^2 , respectively. The color reconstruction loss is a combination of L1 reconstruction loss and a D-SSIM term with the balance weight $\lambda = 0.2$:

$$\mathcal{R}_{\text{pcolor}} = (1 - \lambda)\mathcal{L}_1(I'^1, I'^2) + \lambda\mathcal{L}_{\text{D-SSIM}}(I'^1, I'^2). \quad (5)$$

At the training view, we render the image I^1 from the primitive set Θ^1 and supervise it with ground truth I^* :

$$\mathcal{L}_{\text{color}} = (1 - \lambda)\mathcal{L}_1(I^1, I^*) + \lambda\mathcal{L}_{\text{D-SSIM}}(I^1, I^*). \quad (6)$$

The final training loss is a combination of the ground truth supervised loss at the training view and the color co-regularization term at the pseudo view:

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \lambda_p \mathcal{R}_{\text{pcolor}}, \quad (7)$$

where λ_p is the balance weight and we set it to 1.0.

Table 1: Quantitative results on LLFF with 3, 6, 9 training views. The best, second-best, and third-best entries are marked in red, orange, and yellow, respectively.

| Method | PSNR \uparrow | | | SSIM \uparrow | | | LPIPS \downarrow | | | AVGE \downarrow | | |
|-----------------|-----------------|--------|--------|-----------------|--------|--------|--------------------|--------|--------|-------------------|--------|--------|
| | 3-view | 6-view | 9-view | 3-view | 6-view | 9-view | 3-view | 6-view | 9-view | 3-view | 6-view | 9-view |
| Mip-NeRF [2] | 16.11 | 22.91 | 24.88 | 0.401 | 0.756 | 0.826 | 0.460 | 0.213 | 0.160 | 0.215 | 0.090 | 0.066 |
| DietNeRF [13] | 14.94 | 21.75 | 24.28 | 0.370 | 0.717 | 0.801 | 0.496 | 0.248 | 0.183 | 0.240 | 0.105 | 0.073 |
| RegNeRF [26] | 19.08 | 23.10 | 24.86 | 0.587 | 0.760 | 0.820 | 0.336 | 0.206 | 0.161 | 0.149 | 0.086 | 0.067 |
| FreeNeRF [42] | 19.63 | 23.73 | 25.13 | 0.612 | 0.779 | 0.827 | 0.308 | 0.195 | 0.160 | 0.134 | 0.075 | 0.064 |
| SparseNeRF [35] | 19.86 | - | - | 0.624 | - | - | 0.328 | - | - | 0.127 | - | - |
| 3DGS [15] | 19.22 | 23.80 | 25.44 | 0.649 | 0.814 | 0.860 | 0.229 | 0.125 | 0.096 | 0.120 | 0.066 | 0.051 |
| FSGS [51] | 20.43 | 24.09 | 25.31 | 0.682 | 0.823 | 0.860 | 0.248 | 0.145 | 0.122 | 0.104 | 0.066 | 0.054 |
| CoR-GS (ours) | 20.45 | 24.49 | 26.06 | 0.712 | 0.837 | 0.874 | 0.196 | 0.115 | 0.089 | 0.101 | 0.060 | 0.046 |

5 Experiments

5.1 Setup

Datasets. We conducted experiments on four datasets: LLFF [23], Mip-NeRF360 [3], DTU [14], and Blender [24]. Our experimental setup aligns with previous studies [26, 35, 42, 51], adopting the same data split, with downsampling rates of 8, 4, 4, and 2 for LLFF, Mip-NeRF360, DTU, and Blender, respectively. More details are in the Materia. To focus on the target object and eliminate background noise during evaluation, we apply object masks similar to prior works [26] for DTU. Camera poses are assumed to be known based on calibration or other established methods, following the conventions of sparse-view settings.

Comparison Baselines. We take current SOTA methods Mip-NeRF [2], DietNeRF [13], RegNeRF [26], FreeNeRF [42], SparseNeRF [35], and FSGS [51] for comparisons. For most methods, we directly report their best published quantitative results. For vanilla 3DGS, we report our implemented results.

Evaluation Metrics. We present quantitative evaluations of the reconstruction performance through reporting PSNR, SSIM [36], and LPIPS [47] scores. Additionally, we calculate an Average Error (AVGE) [26], derived from the geometric mean of $\text{MSE} = 10^{-\text{PSNR}/10}$, $\sqrt{1 - \text{SSIM}}$, and LPIPS.

Implementations. We conduct training for 10k iterations on the LLFF, DTU, and Blender datasets and follow 3DGS [15] to train models for 30k iterations on the MipNeRF dataset. Following FSGS [51], we initialize 3DGS and CoR-GS with fused stereo point cloud from sparse views.

5.2 Comparison

LLFF. We provide the quantitative results on the LLFF dataset in Tab. 1. Our method consistently achieves the best performance across PSNR, SSIM, LPIPS, and AVGE metrics with 3, 6, and 9 training views. The depth-supervised FSGS [51] enhances the performance of 3DGS in scenarios with 3 and 6 views. However, with 9 training views, 3DGS can already reconstruct the scene structure well, which limits the effectiveness of external depth supervision. Our method

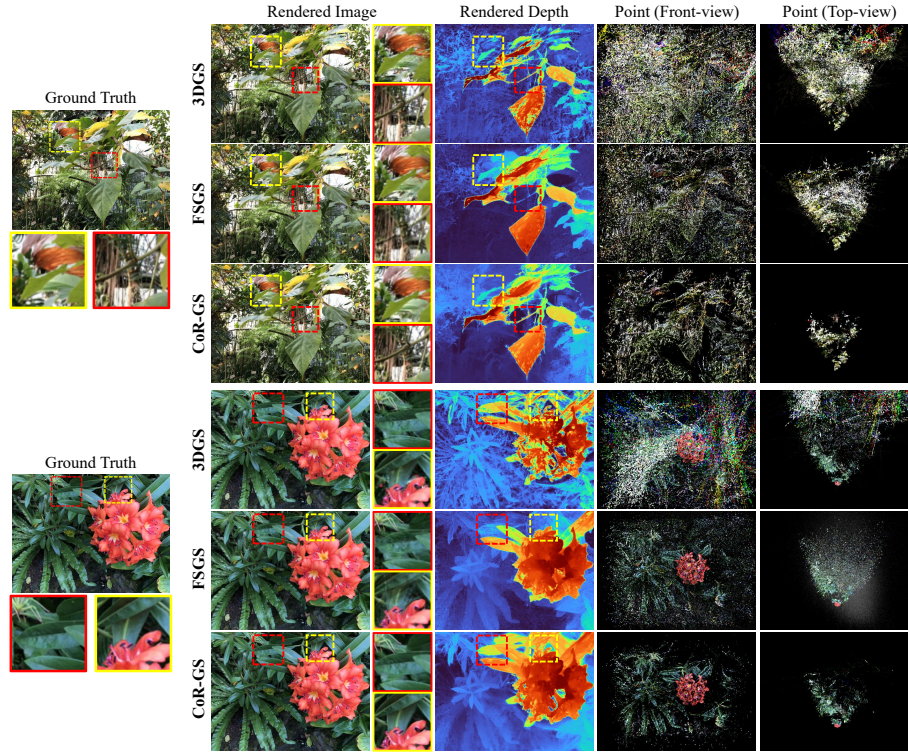
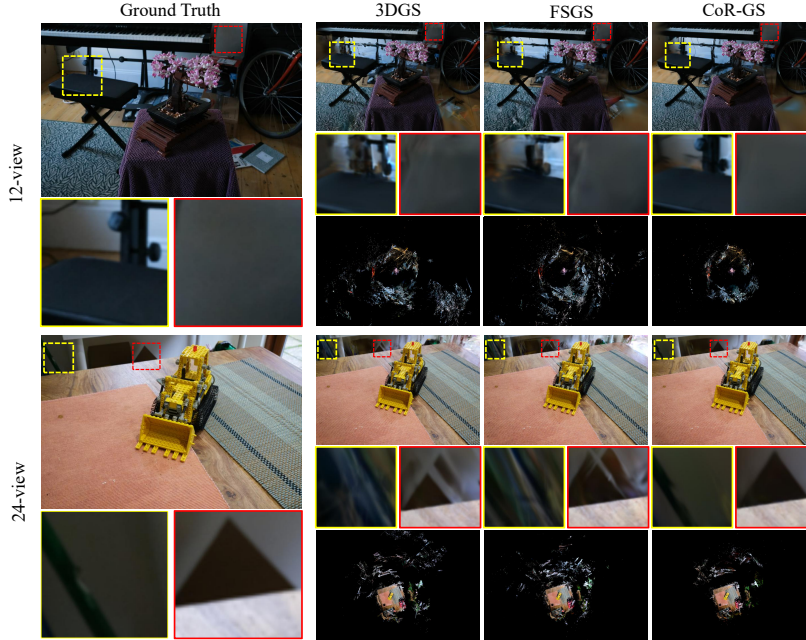


Fig. 7: Qualitative comparison on LLFF. We provide visualizations of Gaussian primitive positions from both the frontal and top views. 3DGS [15] reconstructed incorrect geometry, reflected in the rendered depth maps and the positions of 3D Gaussians. FSGS [51] is affected by noises in the external depth and also struggles to reconstruct compact Gaussians. Conducting co-regularization between different primitive sets, our CoR-GS reconstructs the compact scene geometry.

conducts co-regularization to avoid reconstructing incorrect geometry, leading to improvements across all metrics with various training views. We provide quantitative visualizations of rendered images and depth at novel views and Gaussian points in Fig. 7. From the rendered depth maps, we observe that vanilla 3DGS can recover certain scene structures but exhibits geometric errors, particularly evident in the unrealistic parts of rendered images. For the point representations, many Gaussians are dispersed throughout space and far from the reconstructed scene. These Gaussians, when observed from novel views, can result in unrealistic rendering images. With depth supervision, FSGS can correct the incorrect geometry of 3DGS; however, noises in external depth maps also adversely affect the geometry. Similarly, relying on depth supervision makes it challenging to directly constrain Gaussians. It can be seen that the Gaussians’ dispersion of FSGS is still not compact. Our method effectively assists Gaussians in recon-

Table 2: Quantitative results on Mip-NeRF360. The best are marked in **bold**.

| Method | 12-view | | | | 24-view | | | |
|---------------|-----------------|-----------------|--------------------|-------------------|-----------------|-----------------|--------------------|-------------------|
| | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | AVGE \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | AVGE \downarrow |
| 3DGS [15] | 18.52 | 0.523 | 0.415 | 0.167 | 22.80 | 0.708 | 0.276 | 0.096 |
| FSGS [51] | 18.80 | 0.531 | 0.418 | 0.163 | 23.28 | 0.715 | 0.274 | 0.093 |
| CoR-GS (ours) | 19.52 | 0.558 | 0.418 | 0.151 | 23.39 | 0.727 | 0.271 | 0.091 |

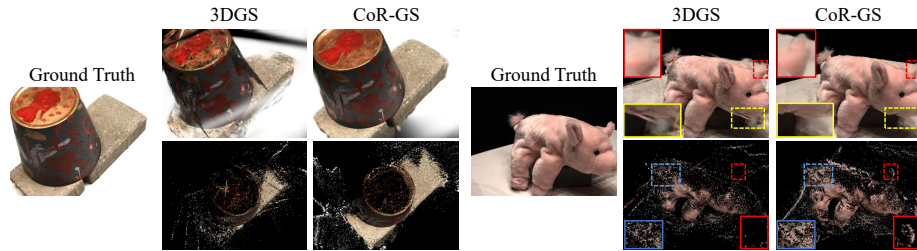
**Fig. 8:** Qualitative comparison on Mip-NeRF360. All three 3DGS-based methods perform well in reconstructing central objects. Our method demonstrates an advantage in non-central regions without sufficient co-observation.

structing coherent and compact geometry, resulting in higher-quality novel view rendering.

Mip-NeRF360. The quantitative results on the Mip-NeRF360 dataset with sparse training views are reported in Table 2. Our method achieves the best performance across PSNR, SSIM, and AVGE with 12 and 24 training views. Vanilla 3DGS achieves the best LPIPS scores with 12 training views. This is because for 360-degree panoramic scenes, 12 views cause many regions to lack co-visibility constraints. In these areas, the pseudo-view co-regularization tends to produce a smoother effect. When using 24 training views, where co-visible regions increase, our method achieves the best LPIPS score. We provide qualitative visualizations in Fig. 8. From rendered images, we observe that all methods can achieve good reconstruction for central objects observed from multiple training views. The ad-

Table 3: Quantitative results on DTU with 3, 6, 9 training views. The best, second-best, and third-best entries are marked in red, orange, and yellow, respectively.

| Method | PSNR \uparrow | | | SSIM \uparrow | | | LPIPS \downarrow | | | AVGE \downarrow | | |
|-----------------|-----------------|--------|--------|-----------------|--------|--------|--------------------|--------|--------|-------------------|--------|--------|
| | 3-view | 6-view | 9-view | 3-view | 6-view | 9-view | 3-view | 6-view | 9-view | 3-view | 6-view | 9-view |
| DietNeRF [13] | 11.85 | 20.63 | 23.83 | 0.633 | 0.778 | 0.823 | 0.314 | 0.201 | 0.173 | 0.243 | 0.101 | 0.068 |
| RegNeRF [26] | 18.89 | 22.20 | 24.93 | 0.745 | 0.841 | 0.884 | 0.190 | 0.117 | 0.089 | 0.112 | 0.071 | 0.047 |
| Mip-NeRF [2] | 9.10 | 16.84 | 23.56 | 0.578 | 0.754 | 0.877 | 0.348 | 0.197 | 0.100 | 0.311 | 0.144 | 0.057 |
| FreeNeRF [42] | 19.92 | 23.25 | 25.38 | 0.787 | 0.844 | 0.888 | 0.182 | 0.137 | 0.096 | 0.098 | 0.068 | 0.046 |
| SparseNeRF [35] | 19.55 | - | - | 0.769 | - | - | 0.201 | - | - | 0.102 | - | - |
| 3DGS [15] | 17.65 | 24.00 | 26.85 | 0.816 | 0.907 | 0.942 | 0.146 | 0.076 | 0.049 | 0.108 | 0.050 | 0.032 |
| CoR-GS (ours) | 19.21 | 24.51 | 27.18 | 0.853 | 0.917 | 0.947 | 0.119 | 0.068 | 0.045 | 0.087 | 0.046 | 0.029 |

**Fig. 9:** Qualitative results on DTU. CoR-GS reconstructs more complete objects.

vantage of our method is evident in its ability to reconstruct clearer structures in non-central regions. From Gaussian positions, we observe that despite training with 360-degree views, 3DGS and FSGS still reconstruct dispersed Gaussians in the distance. The dispersed Gaussians have a negative impact when observing non-central areas in novel views. In contrast, our method reconstructs more compact representations. The results demonstrate that CoR-GS remains well-suited for reconstructing full 360° unbounded scenes with sparse training views.

DTU. The quantitative results on the DTU dataset are reported in Table 3. Our method achieves the best in SSIM, LPIPS, and AVGE with 3, 6, and 9 training views. However, with only 3 views, we observe that 3DGS-based methods get lower PSNR scores compared to NeRF-based methods. This is caused by that 3DGS simply renders the black background in invisible areas beyond training views and the 3 training views of DTU cause lots of invisible areas. NeRFs achieve better PSNR scores due to their interpolation nature for invisible areas. With the increase in training views, we see that 3DGS-based methods get better PSNR scores compared to NeRFs. In the qualitative visualizations in Figure 9, our method renders more complete objects and reconstructs more compact Gaussians compared to 3DGS.

Blender. The quantitative results on the Blender dataset with 8 surrounding training views are reported in Tab. 4. Our method gets the best scores in SSIM and LPIPS, and second in PSNR. The results demonstrate that CoR-GS is also applicable for reconstructing complex objects.

Table 4: Quantitative results on Blender with 8 training views. The best, second-best, and third-best entries are marked in red, orange, and yellow, respectively.

| Method | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow |
|-----------------|-----------------|-----------------|--------------------|
| RegNeRF [26] | 23.86 | 0.852 | 0.105 |
| FreeNeRF [42] | 24.26 | 0.883 | 0.098 |
| SparseNeRF [35] | 24.04 | 0.876 | 0.113 |
| 3DGS [15] | 23.20 | 0.870 | 0.104 |
| FSGS [51] | 24.64 | 0.895 | 0.095 |
| CoR-GS (ours) | 24.43 | 0.896 | 0.084 |

Table 5: Efficiency comparison. Our method achieves the best novel view synthesis and the least inference cost.

| Method | Points | FPS | Inference | | | Train | |
|-----------------|--------------------------------------|------------|-----------------|-----------------|--------------------|---------------|------------|
| | | | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | Time | GPU Mem |
| FreeNeRF [42] | - | 0.09 | 19.63 | 0.612 | 0.308 | 2.3h | 4×48 GB |
| SparseNeRF [35] | - | 0.09 | 19.86 | 0.624 | 0.328 | 1.5h | 32GB |
| 3DGS [15] | 1.16×10^5 | 318 | 19.22 | 0.649 | 0.229 | 2.5min | 2GB |
| CoR-GS (ours) | 7.85×10^4 | 349 | 20.45 | 0.712 | 0.196 | 6min | 3GB |

Efficiency. In Tab. 5. We conduct an efficiency comparison of both training and inference on the LLFF 3-view setting with an RTX 3090 Ti GPU. Compared to NeRFs, 3DGS has significant advantages of efficiency. Compared to vanilla 3DGS, our method introduces some additional training costs due to training two 3D Gaussian radiance fields and rendering pseudo views but remains much more efficient than NeRFs. Our method is the most efficient during inference. Due to reconstructing more compact representations, our method reduces the number of Gaussians by 33% and thus improves inference speed.

5.3 Ablation Study

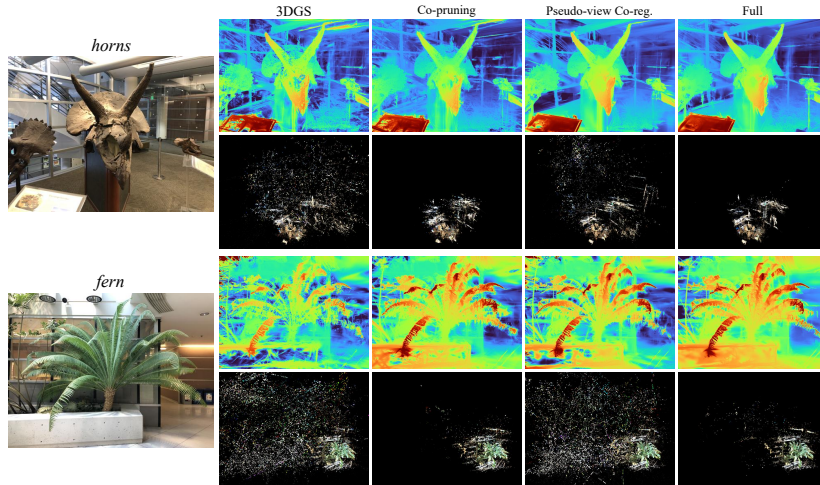
We ablate the effectiveness of suppressing the point disagreement and rendering disagreement. The quantitative results are reported in Tab. 6. The results indicate that suppressing each disagreement is individually beneficial for sparse-view 3DGS, and the combination of both achieves the best performance. We further provide the qualitative visualizations in Fig. 10 to better illustrate their effects.

Co-pruning. Co-pruning improves the geometric structure, reflected in more reasonable depth rendering and more compact Gaussians. However, its drawback is that the rendered depth lacks smoothness, especially evident in the floor part of the horns scene. This is because co-pruning mainly operates on non-matching Gaussians that are far from reconstructed scenes while lacking sufficient constraints on matching Gaussians that are located in reasonable positions.

Pseudo-view Co-regularization. Solely with pseudo-view co-regularization is effective in constraining Gaussians to obtain reasonable depth maps due to inaccurately located Gaussians can be identified through rendering disagreement. However, due to the lack of direct constraints on Gaussians, the representations are not compact enough, leading to some geometric errors that cannot be

Table 6: Ablation study of CoR-GS with 3 training views.

| Co-Pruning | CoR-GS | | LLFF | | | DTU | | |
|------------|-------------|---------|-----------------|-----------------|--------------------|-----------------|-----------------|--------------------|
| | Pseudo-view | Co-reg. | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow | PSNR \uparrow | SSIM \uparrow | LPIPS \downarrow |
| | | | 19.22 | 0.649 | 0.229 | 17.65 | 0.816 | 0.146 |
| ✓ | | | 19.62 | 0.673 | 0.217 | 18.59 | 0.836 | 0.127 |
| | | ✓ | 20.26 | 0.706 | 0.198 | 18.56 | 0.849 | 0.128 |
| ✓ | | ✓ | 20.45 | 0.712 | 0.196 | 19.21 | 0.853 | 0.119 |

**Fig. 10:** Visualization results for ablation. Co-pruning prunes Gaussians far from the reconstructed scene for compact representations, while pseudo-view co-regularization helps correct nearby Gaussians, complementing each other in our method.

effectively corrected, especially evident in the fern scene. The full set of CoR-GS compresses two kinds of disagreement, reconstructing more accurate scene geometry and compact representations.

6 Conclusion

This paper introduces a new co-regularization perspective for improving sparse-view 3DGS. We observe the two 3D Gaussian radiance fields exhibit different behaviors for the same scene with sparse training views and propose point disagreement and rendering disagreement to quantitatively indicate the differences. We further demonstrate the negative correlation between the two disagreements and accurate reconstruction, which allows us to identify inaccurate reconstruction unsupervisedly. Based on the study, we propose CoR-GS, which improves sparse-view 3DGS by depressing the two kinds of disagreement. We validate the effectiveness of CoR-GS across various datasets, achieving state-of-the-art novel view synthesis in the sparse-view setting.

Acknowledgements

In this work, we are supported by the National Natural Science Foundation of China (Grant No.62276016, 62372029, 62106012), the Fundamental Research Funds for the Central Universities. Dr. Lin Gu was supported by JST Moonshot R&D Grant Number JPMJMS2011, Japan.

References

1. Avidan, S., Shashua, A.: Novel view synthesis in tensor space. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. pp. 1034–1040. IEEE (1997)
2. Barron, J.T., Mildenhall, B., Tancik, M., Hedman, P., Martin-Brualla, R., Srinivasan, P.P.: Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 5855–5864 (2021)
3. Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P.: Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5470–5479 (2022)
4. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory. pp. 92–100 (1998)
5. Chen, A., Xu, Z., Geiger, A., Yu, J., Su, H.: Tensorf: Tensorial radiance fields. In: Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXXII. pp. 333–350. Springer (2022)
6. Chen, A., Xu, Z., Zhao, F., Zhang, X., Xiang, F., Yu, J., Su, H.: Mvsnerf: Fast generalizable radiance field reconstruction from multi-view stereo. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14124–14133 (2021)
7. Chen, Z., Li, Z., Song, L., Chen, L., Yu, J., Yuan, J., Xu, Y.: Neurbf: A neural fields representation with adaptive radial basis functions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4182–4194 (2023)
8. Cong, W., Liang, H., Wang, P., Fan, Z., Chen, T., Varma, M., Wang, Y., Wang, Z.: Enhancing nerf akin to enhancing llms: Generalizable nerf transformer with mixture-of-view-experts. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3193–3204 (2023)
9. Deng, K., Liu, A., Zhu, J.Y., Ramanan, D.: Depth-supervised nerf: Fewer views and faster training for free. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12882–12891 (2022)
10. Fridovich-Keil, S., Yu, A., Tancik, M., Chen, Q., Recht, B., Kanazawa, A.: Plenoxels: Radiance fields without neural networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5501–5510 (2022)
11. Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., Sugiyama, M.: Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems* **31** (2018)
12. Hu, W., Wang, Y., Ma, L., Yang, B., Gao, L., Liu, X., Ma, Y.: Tri-miprf: Tri-mip representation for efficient anti-aliasing neural radiance fields. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 19774–19783 (2023)

13. Jain, A., Tancik, M., Abbeel, P.: Putting nerf on a diet: Semantically consistent few-shot view synthesis. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 5885–5894 (2021)
14. Jensen, R., Dahl, A., Vogiatzis, G., Tola, E., Aanaes, H.: Large scale multi-view stereopsis evaluation. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 406–413 (2014)
15. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (ToG)* **42**(4), 1–14 (2023)
16. Kim, M., Seo, S., Han, B.: Infonerf: Ray entropy minimization for few-shot neural volume rendering. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 12912–12921 (2022)
17. Kulháněk, J., Derner, E., Sattler, T., Babuška, R.: Viewformer: Nerf-free neural rendering from few images using transformers. In: *European Conference on Computer Vision (ECCV)* (2022)
18. Li, J., Zhang, J., Bai, X., Zheng, J., Ning, X., Zhou, J., Gu, L.: 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*. pp. 20775–20785 (2024)
19. Li, J., Zhang, J., Bai, X., Zheng, J., Ning, X., Zhou, J., Gu, L.: Talkinggaussian: Structure-persistent 3d talking head synthesis via gaussian splatting. *arXiv preprint arXiv:2404.15264* (2024)
20. Li, J., Socher, R., Hoi, S.C.: Dividemix: Learning with noisy labels as semi-supervised learning. *arXiv preprint arXiv:2002.07394* (2020)
21. Liu, L., Gu, J., Zaw Lin, K., Chua, T.S., Theobalt, C.: Neural sparse voxel fields. *Advances in Neural Information Processing Systems* **33**, 15651–15663 (2020)
22. Luiten, J., Kopanas, G., Leibe, B., Ramanan, D.: Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. *arXiv preprint arXiv:2308.09713* (2023)
23. Mildenhall, B., Srinivasan, P.P., Ortiz-Cayon, R., Kalantari, N.K., Ramamoorthi, R., Ng, R., Kar, A.: Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. *ACM Transactions on Graphics (TOG)* **38**(4), 1–14 (2019)
24. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* **65**(1), 99–106 (2021)
25. Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics (ToG)* **41**(4), 1–15 (2022)
26. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N.: Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5480–5490 (2022)
27. Roessle, B., Barron, J.T., Mildenhall, B., Srinivasan, P.P., Nießner, M.: Dense depth priors for neural radiance fields from sparse input views. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 12892–12901 (2022)
28. Schönberger, J.L., Frahm, J.M.: Structure-from-Motion Revisited. In: *Conference on Computer Vision and Pattern Recognition (CVPR)* (2016)
29. Schönberger, J.L., Zheng, E., Pollefeys, M., Frahm, J.M.: Pixelwise View Selection for Unstructured Multi-View Stereo. In: *European Conference on Computer Vision (ECCV)* (2016)

30. Sindhwani, V., Niyogi, P., Belkin, M.: A co-regularization approach to semi-supervised learning with multiple views. In: *Proceedings of ICML workshop on learning with multiple views*. vol. 2005, pp. 74–79. Citeseer (2005)
31. Song, J., Park, S., An, H., Cho, S., Kwak, M.S., Cho, S., Kim, S.: Därf: Boosting radiance fields from sparse inputs with monocular depth adaptation (2023)
32. Sun, C., Sun, M., Chen, H.T.: Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5459–5469 (2022)
33. Tang, J., Ren, J., Zhou, H., Liu, Z., Zeng, G.: Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653* (2023)
34. Uhrig, J., Schneider, N., Schneider, L., Franke, U., Brox, T., Geiger, A.: Sparsity invariant cnns. In: *International Conference on 3D Vision (3DV)* (2017)
35. Wang, G., Chen, Z., Loy, C.C., Liu, Z.: Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. pp. 9065–9076 (October 2023)
36. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing* **13**(4), 600–612 (2004)
37. Wei, H., Feng, L., Chen, X., An, B.: Combating noisy labels by agreement: A joint training method with co-regularization. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. pp. 13726–13735 (2020)
38. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X.: 4d gaussian splatting for real-time dynamic scene rendering. *arXiv preprint arXiv:2310.08528* (2023)
39. Wynn, J., Turmukhambetov, D.: Diffusionerf: Regularizing neural radiance fields with denoising diffusion models. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 4180–4189 (2023)
40. Xiong, H., Muttukuru, S.: Sparsegs: Real-time 360° sparse view synthesis using gaussian splatting (10 2023)
41. Xu, Q., Xu, Z., Philip, J., Bi, S., Shu, Z., Sunkavalli, K., Neumann, U.: Point-nerf: Point-based neural radiance fields. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 5438–5448 (2022)
42. Yang, J., Pavone, M., Wang, Y.: Freenerf: Improving few-shot neural rendering with free frequency regularization. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 8254–8263 (2023)
43. Yu, A., Li, R., Tancik, M., Li, H., Ng, R., Kanazawa, A.: Plenotrees for real-time rendering of neural radiance fields. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. pp. 5752–5761 (2021)
44. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 4578–4587 (2021)
45. Yu, X., Han, B., Yao, J., Niu, G., Tsang, I., Sugiyama, M.: How does disagreement help generalization against label corruption? In: *International Conference on Machine Learning*. pp. 7164–7173. PMLR (2019)
46. Zhang, J., Li, J., Huang, L., Yu, X., Gu, L., Zheng, J., Bai, X.: Robust synthetic-to-real transfer for stereo matching. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. pp. 20247–20257 (2024)
47. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. pp. 586–595 (2018)

- 48. Zhou, Q.Y., Park, J., Koltun, V.: Open3d: A modern library for 3d data processing. arXiv preprint arXiv:1801.09847 (2018)
- 49. Zhou, T., Tulsiani, S., Sun, W., Malik, J., Efros, A.A.: View synthesis by appearance flow. In: Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14. pp. 286–301. Springer (2016)
- 50. Zhou, Z., Tulsiani, S.: Sparsefusion: Distilling view-conditioned diffusion for 3d reconstruction. In: CVPR (2023)
- 51. Zhu, Z., Fan, Z., Jiang, Y., Wang, Z.: Fsgs: Real-time few-shot view synthesis using gaussian splatting (2023)