DP-GS: Depth-prior & Perception-guided Gaussian Splatting for Sparse-view Novel View Synthesis

Bowen Gao, Zhicheng Lu, Mingyi He, Yuchao Dai†
School of Electronics and Information, Northwestern Polytechnical University and
Shaanxi Key Laboratory of Information Acquisition and Processing, Xi'an, Shaanxi, 710129, China

***Abstract*—Sparse-view novel view synthesis (NVS) is crucial for practical applications such as AR/VR, robotics, and large-scale scene reconstruction, where capturing dense multi-view images is often impractical. 3D Gaussian Splatting (3DGS) offers explicit and efficient scene representations for real-time NVS. However, its performance significantly degrades under sparse-view settings, resulting in incomplete geometry and severe texture artifacts. To overcome these limitations, we propose DP-GS (Depth-prior & Perception-guided Gaussian Splatting), a unified framework designed for sparse-view NVS. DP-GS integrates depth regularization derived from monocular depth estimation to improve geometric completeness. Moreover, we introduce a point-level dropout mechanism to suppress unreliable points, and integrate perceptual optimization guided by diffusion model to enhance texture fidelity and structural consistency. Extensive experiments on LLFF and Mip-NeRF360 datasets demonstrate that DP-GS consistently outperforms existing 3DGS methods under sparse view settings.**

I. Introduction

Novel view synthesis (NVS) aims to generate photorealistic images from novel viewpoints given a limited set of inputs. This task is crucial for AR/VR, autonomous driving, and digital content creation. In practice, capturing dense multi-view data is often impractical, making sparse-view NVS a highly challenging yet important problem [1].

Neural radiance fields (NeRF) [2] achieve impressive quality by learning volumetric scene representations but rely on dense calibrated views and long optimization times. 3D Gaussian Splatting (3DGS) [3] improves efficiency and enables real-time rendering via explicit Gaussian primitives. Nevertheless, 3DGS still fails to maintain geometric completeness and texture consistency when input views are sparse [4].

Recent approaches [4], [5] have explored incorporating depth priors to mitigate geometry fragmentation. But these methods often apply depth as auxiliary supervision only, lacking globally aligned constraints, which leads to incomplete geometry under sparse settings[6]. Also, original 3DGS framework prunes points using only a simple opacity threshold. This strategy provides limited control over point reliability and fails to effectively suppress floating artifacts or unstable regions.

For appearance modeling, SparseGS [1] introduces score distillation sampling (SDS) [7] combined with depth cues to improve visual fidelity. However, these approaches typically rely on loose constraints without enforcing explicit structural consistency across views, often leading to local texture inconsistencies and incomplete suppression of floating artifacts, especially under highly sparse input conditions.

To address these limitations, we propose DP-GS (Depth-prior & Perception-guided Gaussian Splatting), which extends 3DGS by integrating depth priors, point-level dropout, and perceptual optimizations. We incorporate depth regularization predicted by Depth Anything v2 [8] and introduces a point-level dropout mechanism to filter unreliable points dynamically. For the appearance module, we leverage perceptual guidance (SDS and structural consistency loss) to enhance texture fidelity and structural consistency. Guided by a diffusion model, this approach does not rely on ground-truth novel-view images as supervision. Instead, it leverages the perceptual understanding capability of diffusion models to provide high-level semantic guidance, simultaneously enhancing geometry and appearance. By integrating these components, DP-GS achieves high-fidelity and structurally consistent NVS under very few input views. Our main contributions are summarized as follows:

(1) A depth-prior regularization strategy and point-level dropout mechanism are introduced into 3DGS to enhance geometric consistency and suppress unreliable points under sparse-view conditions.

(2) A perceptual and structural guidance module is incorporated, integrating SDS and structural consistency loss to improve texture coherence and generalization to unseen viewpoints.

(3) Experiments on LLFF and Mip-NeRF360 datasets demonstrate superior performance over baseline methods, validating the effectiveness of the proposed DP-GS framework for sparse-view NVS.

II. Related Work

A. Traditional novel view synthesis methods

Before the emergence of neural network-based methods, novel view synthesis was primarily addressed using geometry-based multi-view reconstruction and image-based interpolation techniques. Classical Structure from Motion (SfM) [9] and Multi-View Stereo (MVS) [10], [11] approaches estimate dense point clouds or meshes via feature matching and reprojection. However, these methods heavily rely on dense and well-distributed viewpoints, often failing under sparse or low-texture conditions. Recent works such as LoopRefine [12] introduce loop consistency to refine camera poses and mitigate

† Corresponding author (daiyuchao@nwpu.edu.cn)
This research was supported in part by the National Natural Science Foundation of China (62271410, 12150007).

979-8-3315-7206-8/25/$31.00

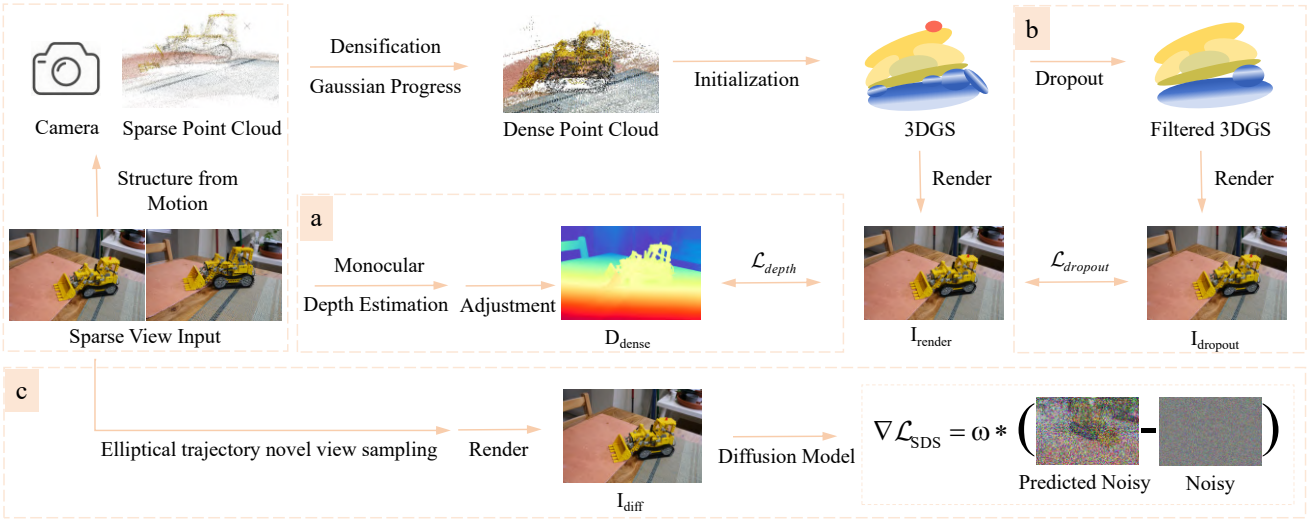

Fig. 1. Overview of the proposed DP-GS framework. The framework integrates MOGP-based point cloud densification and depth-prior geometric regularization (a) to enhance structural completeness. A point-level dropout mechanism (b) further suppresses unreliable points. Finally, perceptual optimization guided by SDS and structural consistency (c) improves texture fidelity and cross-view coherence under sparse inputs.

drift, but they still struggle to produce complete and accurate geometry in extremely sparse capture setups.

B. Neural radiance fields

NeRF [2] achieves high-quality novel view synthesis by implicitly modeling volumetric density and radiance. However, they require dense inputs and long optimization times, making them unsuitable for sparse-view scenarios. To address this, DS-NeRF [5] incorporates depth supervision as explicit geometric constraints, MixNeRF [13] uses data mixing and regularization to enhance generalization and RegNeRF [14] introduces various regularization techniques to mitigate overfitting and improve texture coherence in novel views. Although these methods improve sparse-view performance to some extent, they remain fundamentally limited by implicit volumetric representations and still struggle with reliable geometry and consistent appearance under very few views.

C. 3D Gaussian Splatting

3DGS [3] offers explicit, efficient novel view synthesis with real-time high-fidelity rendering using anisotropic 3D Gaussians. Lu et al. [15] further extended it to dynamic scenes via geometry-aware deformable Gaussians. Beyond dense-view scenarios, recent efforts have adapted 3DGS to sparse-view conditions by introducing various priors and constraints. TGS [16] proposes an image-driven point cloud densification scheme from single images, but it largely depends on the quality of the initial coarse reconstruction and lacks explicit appearance constraints. GP-GS [17] uses multi-output Gaussian processes to predict 3D coordinates and colors, while it primarily focuses on point cloud densification without explicit texture consistency considerations. DepthGS [4] enforces depth supervision and smoothness regularization to stabilize Gaussians, yet it does not address appearance coherence across novel views. SparseGS [1] integrates Diffusion with SDS, using perceptual gradients from a generative model to optimize Gaussians without ground-truth supervision, improving visual fidelity under sparse inputs. DIG3D [18] relies purely on image-based perceptual signals to initialize Gaussian points, which may struggle to maintain accurate geometry under severe sparsity.

III. Prerequisite

A. 3D Gaussian Splatting

3DGS [3] is an explicit rasterization-based method for real-time radiance field rendering using 3D Gaussian distributions, enabling efficient photorealistic scene synthesis from sparse image observations. The inputs to 3DGS include a set of images of a static scene and a sparse point cloud obtained via camera alignment. On these sparse points, a set of 3D Gaussian distributions are constructed, parameterized by position $\boldsymbol{x}$, opacity α, covariance matrix $\boldsymbol{\Sigma}$, and Spherical Harmonic coefficients to capture view-dependent colors. The parameters are optimized through an adaptive density control algorithm. The Gaussian distribution is defined as:

$$G(\boldsymbol{x}) = e^{-\frac{1}{2}\boldsymbol{x}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{x}}. \tag{1}$$

For covariance matrix optimization, $\boldsymbol{\Sigma}$ can be decomposed using a scaling matrix $\boldsymbol{S}$ and a rotation matrix $\boldsymbol{R}$, as

$$\boldsymbol{\Sigma} = \boldsymbol{R}\boldsymbol{S}\boldsymbol{S}^T\boldsymbol{R}^T. \tag{2}$$

To enable independent optimization, these are further represented as a 3D scaling vector $\boldsymbol{s}$ and a quaternion $\boldsymbol{r}$ for rotation.

A key factor enabling 3DGS to achieve high rendering speed is its fast differentiable rasterizer. The overall acceleration is achieved by chunking the Gaussian splats and allowing approximate opacity blending. The rasterizer uses fast inverse passes to traverse accumulated opacity values, supporting efficient gradient-based optimization without restrictions. For

each pixel, the final color C is computed by blending all overlapping Gaussians as

$$C = \sum_{i \in N_{cov}} c_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \tag{3}$$

where N_{cov} denotes Gaussians covering the pixel, α_i is the effective opacity (combined with the 2D projected Gaussian density), and c_i represents the color contribution.

B. Diffusion Models and Score Distillation Sampling

Diffusion models approximate data distributions by gradually adding Gaussian noise to samples and learning to reverse this process via denoising. In the forward process, a clean data point $\mathbf{x}_0$ is progressively perturbed:

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}\left(\mathbf{x}_t; \sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t \mathbf{I}\right), \tag{4}$$

where β_t denotes the noise schedule. The reverse process aims to reconstruct clean samples:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}\left(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \Sigma_\theta(\mathbf{x}_t, t)\right), \tag{5}$$

where μ_θ is predicted by the neural network with parameters θ.

SDS [7] utilizes a pre-trained diffusion model to guide 3D parameter optimization without explicit 3D supervision. The score function is defined as:

$$s_\phi(\mathbf{x}_t, t, c) = \nabla_{\mathbf{x}_t} \log p_\phi(\mathbf{x}_t|c), \tag{6}$$

where c denotes an optional conditioning input. The gradient for updating 3D parameters θ is:

$$\nabla_\theta L_{\text{SDS}} = \mathbb{E}_{t,\epsilon}\left[s_\phi(\mathbf{x}_t, t, c) \cdot \nabla_\theta \mathbf{x}\right]. \tag{7}$$

This formulation provides perceptual gradients that enhance realism and structural consistency.

C. MOGP-based Point Cloud Extension

GP-GS [17] proposes a point cloud densification strategy based on a Multi-Output Gaussian Process (MOGP) to address sparse and incomplete reconstructions. It models the mapping from 2D pixel space $\mathbf{x}_i = (u, v, d)$ to 3D geometry and color $\mathbf{y}_i = (x, y, z, r, g, b)$ as

$$\mathbf{Y} \sim \mathcal{MOGP}\big(\mathbf{m}(\mathbf{x}), K(\mathbf{x}, \mathbf{x}')\big), \tag{8}$$

where $\mathbf{m}(\mathbf{x})$ is the mean function and $K(\mathbf{x}, \mathbf{x}')$ is the covariance kernel. GP-GS adopts the Matern kernel:

$$k(\mathbf{x}, \mathbf{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\frac{|\mathbf{x}-\mathbf{x}'|}{l}\right)^\nu K\nu\left(\sqrt{2\nu}\frac{|\mathbf{x}-\mathbf{x}'|}{l}\right), \tag{9}$$

where l is the length scale, ν controls smoothness, $K\nu$ is the modified Bessel function, and $\Gamma(\cdot)$ is the gamma function.

Model parameters θ_i are optimized by maximizing the marginal likelihood with L_2 regularization:

$$\mathcal{L}_{total} = -\log p(\mathbf{Y}|\mathbf{X}) + \lambda \sum |\theta_i|^2, \tag{10}$$

where λ is the regularization weight.

Predicted uncertainty is used to filter unreliable points, and reliable predictions are merged with the original sparse points to form the densified set.

IV. Method

In this section, we present DP-GS, a unified framework designed to improve geometry completeness and appearance consistency under sparse-view settings.

As illustrated in Fig. 1, DP-GS consists of three core modules: depth-prior regularization, point-level dropout and perceptual optimization. It operates on a set of sparse view images $\{I_i \in \mathbb{R}^{H\times W\times 3}\}_{i=1}^N$ and produces synthesized novel view images from arbitrary target viewpoints. Initial sparse point cloud is obtained using SfM and further densified via MOGP to improve geometric coverage. Monocular depth maps predicted by Depth Anything v2 [8] are then projected and scale-aligned to constrain the 3D geometry. A point-level dropout mechanism is introduced to suppress unreliable Gaussians during optimization. Novel views sampled along elliptical camera paths are refined using high-level perceptual guidance. This guidance leverages semantic gradients from a pre-trained diffusion model via SDS loss, enabling the model to capture global appearance semantics and achieve realistic, coherent textures. Detailed formulations for each module are described in the following subsections.

A. Depth-prior Regularization

DP-GS initializes geometry using a sparse point cloud $\mathcal{P}_{sparse}$ estimated via SfM, which is often incomplete and misaligned under sparse-view conditions. To provide stronger geometric supervision, monocular depth maps D_{dense} are predicted using Depth Anything v2, chosen for its better generalization and stability.

A linear correction aligns D_{dense} to the SfM-derived sparse depths D_{sparse} by estimating a scaling coefficient and an offset based on median and median absolute deviation statistics. The corrected depth is defined as:

$$D^*_{\text{dense}} = \text{scale}_i \cdot D_{\text{dense}} + \text{offset}_i. \tag{11}$$

A confidence check excludes images with extreme scale deviations, and valid pixels are further filtered by a binary mask m_i based on depth consistency thresholds to exclude unreliable regions.

The depth alignment loss encourages Gaussian points to match the corrected depth map:

$$\mathcal{L}_{depth} = \frac{1}{N}\sum_{i=1}^{N} m_i \cdot |D_{\text{render}}(u_i, v_i) - D^*_{\text{dense}}(u_i, v_i)|, \tag{12}$$

where $D_{\text{render}}(u_i, v_i)$ denotes the rendered depth at pixel (u_i, v_i) from the Gaussian splatting model, and D^*_{dense} is the corrected depth map.

An exponential decay adjusts the depth weight as

$$\lambda_{\text{depth}}(t) = \lambda_0 \cdot \exp\left(-\gamma \cdot \frac{t}{T}\right), \tag{13}$$

where λ_0 is the initial weight, t is the current step, T is the total steps, and γ controls the decay rate. A larger γ causes faster decay, reducing depth supervision in later stages.

B. Point-level Dropout

Despite depth-prior regularization, sparse-view reconstructions can still exhibit geometric ambiguities such as floating artifacts and noisy points. We introduce a point-level dropout mechanism to dynamically filter unstable Gaussians using multi-factor reliability cues.

A relative depth residual e_p is computed by comparing the rendered depth D_{render} with the aligned predicted depth D^*_{dense}:

$$e_p(u,v) = \frac{|D_{\text{render}}(u,v) - D^*_{\text{dense}}(u,v)|}{D^*_{\text{dense}}(u,v) + \delta}. \tag{14}$$

An initial uncertainty score s_j^{base} is derived from the scale $\|\mathbf{s}_j\|$ and opacity α_j:

$$s_j^{\text{base}} = \exp(-10\|\mathbf{s}_j\|)\,(1-\alpha_j). \tag{15}$$

By combining the depth residual e_j and a binary reliability mask r_j, constructed by thresholding aligned depth residuals and filtering low-opacity points, the final dropout activation score with scene-level adjustment factor ρ is defined as:

$$s_j^{\text{dropout}} = \rho \cdot (s_j^{\text{base}} + w_1 e_j + w_2(1-r_j)), \tag{16}$$

Points are stochastically deactivated by comparing s_j^{dropout} to a uniform sample:

$$\text{mask}_j = \mathbb{I}[\text{Uniform}(0,1) > s_j^{\text{dropout}}]. \tag{17}$$

Finally, we enforce a consistency-based dropout loss:

$$L_{dropout} = \|I_{\text{render}} - I_{\text{dropout}}\|_1 + \lambda(1 - \text{SSIM}(I_{\text{render}}, I_{\text{dropout}})), \tag{18}$$

where I_{dropout} denotes the rendering after masking.

C. Perceptual Optimization

While pixel-wise losses (e.g., L1, SSIM) enforce local consistency, they lack high-level semantic understanding, often resulting in oversmoothed or structurally inconsistent results. In contrast, perceptual guidance refers to using high-level semantic cues from pre-trained models to guide optimization beyond low-level details. In our framework, we leverage SDS derived from diffusion models as a perceptual signal, enabling the system to "understand" texture and structure from a global semantic perspective without relying on ground-truth novel views. A structural consistency loss further improves texture geometric consistency across sparse-view reconstructions.

Novel viewpoints are sampled along an elliptical trajectory defined in the $\mathbf{e}_1$-$\mathbf{e}_2$ plane, where $\mathbf{e}_1$, $\mathbf{e}_2$, and $\mathbf{e}_3$ are obtained via Principal Component Analysis (PCA) on camera centers, and $\mathbf{c}_{mean}$ is the ellipse center. The base path is

$$\mathbf{p}(\theta) = \mathbf{c} + a\cos\theta\,\mathbf{e}_1 + b\sin\theta\,\mathbf{e}_2, \tag{19}$$

with an added height perturbation $z(\theta) = A\cos(\theta+\phi)$ along $\mathbf{e}_3$ to encourage 3D diversity:

$$\mathbf{p}(\theta) = \mathbf{c} + a\cos\theta\,\mathbf{e}_1 + b\sin\theta\,\mathbf{e}_2 + z(\theta)\mathbf{e}_3. \tag{20}$$

For each $\mathbf{p}(\theta)$, a view matrix pointing to $\mathbf{c}_{mean}$ is built. During later training stages, random samples along this path are used for novel view rendering and SDS supervision.

Pre-trained model (Stable Diffusion-2.1) provides the score function $\nabla_{\mathbf{x}} \log p_\phi(\mathbf{x}|c)$, where c denotes optional conditioning. The SDS gradient is given by:

$$\nabla_\theta \mathcal{L}_{SDS} = \mathbb{E}_{t,\epsilon}\left[\nabla_{\mathbf{x}} \log p_\phi(\mathbf{x}|c) \cdot \nabla_\theta \mathbf{x}\right], \tag{21}$$

providing high-level perceptual gradients that complement geometric signals.

To enforce cross-view structural coherence, a structural consistency loss based on LPIPS is used:

$$L_{struct} = \text{LPIPS}(I_{\text{render}}, I_{\text{syn}}), \tag{22}$$

where I_{render} is the Gaussian-rendered image from training pose and I_{syn} is the synthesized novel view rendered image from a new viewpoint sampled along the elliptical trajectory.

The overall perceptual loss is defined as:

$$\mathcal{L}_{diff} = \lambda_1 \mathcal{L}_{SDS} + \lambda_2 \mathcal{L}_{struct}, \tag{23}$$

where $\lambda_1 = 0.001$ and $\lambda_2 = 0.01$.

This formulation empirically improves texture preservation and perceptual realism (see Sec. V-D), though it introduces additional computational overhead due to diffusion inference.

D. Overall Objective

The final objective function of the proposed framework integrates multiple loss terms, including photometric reconstruction, depth supervision, point-level dropout regularization, and perceptual guidance components. The overall loss is formulated as:

$$\mathcal{L} = \mathcal{L}_{color} + \lambda_{depth} \cdot \mathcal{L}_{depth} + \lambda_{drop} \cdot \mathcal{L}_{dropout} + \mathcal{L}_{diff}, \tag{24}$$

where λ_{depth} is exponentially decayed from 1 to 0.02 (as defined in Eq. 13), and λ_{drop} is set to 0.1 during the first half of training and 0.2 during the second half.

The photometric reconstruction loss $\mathcal{L}_{color}$ ensures consistency between rendered and ground truth images in both color and structure. It is defined as a hybrid term:

$$\mathcal{L}_{color} = 0.8 \cdot \mathcal{L}_1 + 0.2 \cdot (1 - \text{SSIM}), \tag{25}$$

where L_1 denotes pixel-wise absolute error and SSIM is the structural similarity index.

V. Experiment

A. Dataset

Experiments are conducted on the LLFF [19] and Mip-NeRF360 [20] datasets to evaluate geometric fidelity and view synthesis quality under sparse-view settings.

LLFF contains forward-facing indoor and outdoor scenes with complex geometric and appearance variations. Following standard sparse-view protocols, 3, 6, 9 and 12 views are used as inputs, and the remaining views are reserved for evaluation. Mip-NeRF360 includes large-scale unbounded outdoor scenes with rich geometric details and challenging illumination conditions. We adopt configurations with 12 and 24 training views and use the remaining views for testing, following the setup in [1].

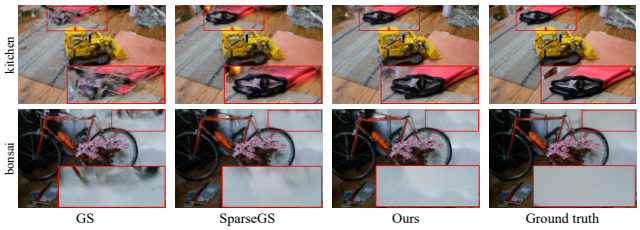

Fig. 2. **Qualitative evaluation on Mip-NeRF360 dataset.** Our method better preserves global scene completeness and sharp background structures, especially in complex areas like "kitchen" and "bonsai". Compared to 3DGS and SparseGS, DP-GS reduces floating artifacts and enhances fine texture details, leading to more realistic and stable results under sparse inputs.

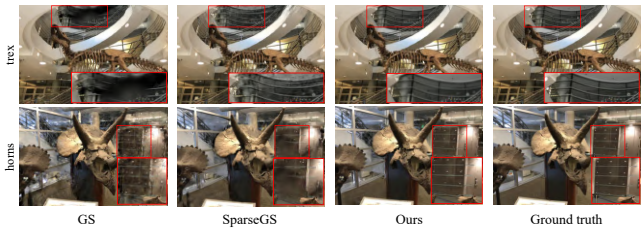

Fig. 3. **Qualitative evaluation on LLFF dataset.** DP-GS reconstructs sharper object contours and preserves thin structures, as seen in "trex" and "horn". The method mitigates background noise and over-smoothing common in 3DGS and SparseGS, resulting in improved texture coherence and better geometric fidelity.

TABLE I
RESULTS ON MIP-NERF360 UNDER 12/24 INPUT-VIEW

models	12-view			24-view		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS	15.7619	0.3996	0.4683	20.0520	0.5963	0.3337
SparseGS(local)	17.1044	0.4668	0.4374	21.0598	0.6277	0.3312
Ours	17.5222	0.5068	0.4223	21.4934	0.6453	0.3126

TABLE II
RESULTS ON LLFF UNDER 3/6 INPUT-VIEW

models	3-view			6-view		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS	11.9889	0.3078	0.5346	18.4149	0.6320	0.3004
SparseGS(local)	15.3106	0.4517	0.4158	20.7469	0.6963	0.2519
Ours	16.9160	0.4941	0.3738	23.1249	0.7586	0.1929

B. Experiment Details

All experiments are conducted on an NVIDIA RTX 3090 GPU. Input images are downsampled to 1/4 resolution on Mip-NeRF360 and 1/8 on LLFF. Gaussian initialization comes from SfM, while depth priors are predicted by Depth Anything v2 and filtered with confidence masks (Sec. IV-A).

We train each model for 10k iterations using the Adam optimizer. The initial learning rates are set as follows: 1.6×10^{-4} for position parameters, 2.5×10^{-3} for color features, 0.05 for opacity, 5×10^{-3} for scale, and 1×10^{-3} for rotation. Point cloud densification is performed every 100 iterations, with opacity reset every 3k iterations, and densification activated from iteration 500 until the final 10k iteration. For the point-level dropout, we set $\delta = 1 \times 10^{-6}$, $w_1 = 0.6$ and $w_2 = 0.3$. The scene-level factor ρ is dynamically computed as the scale ratio, clipped to $[0.3, 3.0]$. For SDS guidance, we follow the setting of SparseGS [1], using the same timestep distribution and noise schedule, while other hyperparameters remain unchanged.

Evaluation metrics include PSNR, SSIM, and LPIPS. We evaluate configurations with 3, 6, 9, 12, and 24 input views to analyze performance under different sparsity levels, aligning with prior works [1] for fair comparison. We locally reimplemented SparseGS under our experimental protocol, where input views are randomly sampled with a stride of 8, while all other settings follow the original SparseGS.

C. Comparison Study

We evaluate our method on LLFF and Mip-NeRF360 datasets to validate its performance under sparse-view configurations. The LLFF dataset includes forward-facing scenes with diverse indoor and outdoor content, where limited input views pose challenges for geometry completeness and texture consistency. In contrast, Mip-NeRF360 comprises large-scale unbounded outdoor scenes with complex occlusions and illumination variations, posing significant challenges for sparse-view reconstruction fidelity.

Quantitative results are summarized in Tables I, II, and III, showing that DP-GS outperforms 3DGS and SparseGS(local) baselines across all evaluation metrics. Qualitative comparisons illustrating improvements in geometric fidelity and texture coherence are presented in Fig. 2 and Fig. 3.

On Mip-NeRF360, similar trends are observed under 12-view and 24-view settings, demonstrating improved geometric stability and texture coherence in unbounded scenes. On LLFF, DP-GS achieves consistent improvements in PSNR and SSIM while reducing LPIPS, indicating better geometric fidelity and perceptual quality even with as few as 3 or 6 input views.

TABLE III
RESULTS ON LLFF UNDER 9/12 INPUT-VIEW

models	9-view			12-view		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
3DGS	22.2857	0.7563	0.2099	23.6173	0.8084	0.1697
SparseGS(local)	23.2760	0.7890	0.1856	24.2362	0.8195	0.1684
Ours	24.7944	0.8217	0.1500	26.3290	0.8640	0.1180

These gains can be attributed to the combination of depth regularization, point-level dropout, and perceptual optimization guided by SDS and structural consistency, which together mitigate geometric fragmentation and improve texture detail.

D. Ablation Study

We perform ablation experiments on the Mip-NeRF360 dataset under the 24-view setting to validate the individual contributions of each proposed component. Quantitative results are summarized in Table IV, while qualitative visualizations illustrating geometric completeness and texture coherence are presented in Fig. 4.

Depth Regularization. Depth-prior regularization constrains the Gaussian points to align with predicted depth, improving geometric completeness. Removing this module leads to geometric distortion and floating artifacts, causing a

w/o Depth Regularization w/o Dropout w/o Perceptual Guidance Ours

Fig. 4. **Ablation results on Mip-NeRF360 (24-view).** Without depth regularization, geometry becomes incomplete and floating artifacts appear. Disabling dropout leads to more noise and unstable surfaces. Omitting perceptual guidance results in over-smoothed and blurry textures. Our full model combines all components, achieving sharper geometry and finer texture details.

TABLE IV
ABLATION STUDIES

depth regularization	dropout	perceptual guidance	PSNR↑	SSIM↑	LPIPS↓
	✓	✓	20.6421	0.6194	0.3340
✓		✓	21.2577	0.6364	0.3172
✓	✓		21.1813	0.6373	0.3183
✓	✓	✓	21.4934	0.6453	0.3126

We ablate our components on the Mip-NeRF360 under 24-view setting.

PSNR drop from 21.49 dB to 20.64 dB and an LPIPS increase from 0.313 to 0.334.

Point-level Dropout. The dropout mechanism filters unreliable points with high depth residuals or low opacity. Disabling this module results in increased surface noise and reduced robustness, reflected by a PSNR decrease from 21.49 dB to 21.26 dB and an SSIM decrease from 0.645 to 0.636.

Perceptual Guidance. The perceptual optimization module (SDS and structural consistency) enhances texture detail and cross-view coherence. Without this component, the model produces smoother yet less detailed textures, causing PSNR to decrease from 21.493 to 21.18.

VI. CONCLUSIONS

This paper proposes DP-GS, a Depth-prior and Perception-guided Gaussian Splatting framework for sparse-view NVS. By integrating depth regularization, point-level dropout, and diffusion-based perceptual optimization, DP-GS achieves improved geometric completeness and texture fidelity. Experiments on LLFF and Mip-NeRF360 show consistent gains in PSNR, SSIM, and LPIPS over baselines. In future work, we aim to adapt DP-GS for dynamic or deformable scenes, improve its real-time capabilities, and investigate broader applications in large-scale, unbounded environments.

REFERENCES

[1] H. Xiong, S. Muttukuru, R. Upadhyay, P. Chari, and A. Kadambi, "Sparsegs: Real-time 360° sparse view synthesis using gaussian splatting," *arXiv preprint arXiv:2312.00206*, 2023.

[2] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," in *ECCV*, 2020, pp. 405–421.

[3] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis, "3d gaussian splatting for real-time radiance field rendering," *TOG*, vol. 42, no. 4, pp. 1–14, 2023.

[4] J. Chung, J. Oh, and K. M. Lee, "Depth-regularized optimization for 3d gaussian splatting in few-shot images," in *CVPR*, 2024, pp. 811–820.

[5] K. Deng, A. Liu, J.-Y. Zhu, and D. Ramanan, "Depth-supervised nerf: Fewer views and faster training for free," in *CVPR*, 2022, pp. 12 882–12 891.

[6] Z. Zhang, W. Hu, Y. Lao, T. He, and H. Zhao, "Pixel-gs: Density control with pixel-aware gradient for 3d gaussian splatting," in *ECCV*, 2024, pp. 326–342.

[7] B. Poole, A. Jain, J. T. Barron, and B. Mildenhall, "Dreamfusion: Text-to-3d using 2d diffusion," *arXiv preprint arXiv:2209.14988*, 2022.

[8] L. Yang, B. Kang, Z. Huang, *et al.*, "Depth anything v2," *arXiv preprint arXiv:2406.09414*, 2024.

[9] J. L. Schonberger and J.-M. Frahm, "Structure-from-motion revisited," in *CVPR*, 2016, pp. 4104–4113.

[10] D. Rückert, L. Franke, and M. Stamminger, "Adop: Approximate differentiable one-pixel point rendering," *TOG*, vol. 41, no. 4, pp. 1–14, 2022.

[11] K.-A. Aliev, A. Sevastopolsky, M. Kolos, D. Ulyanov, and V. Lempitsky, "Neural point-based graphics," in *ECCV*, 2020, pp. 696–712.

[12] Z. Wang, H. Deng, J. Shi, *et al.*, "Looprefine: Deep camera pose estimation with loop consistency," *IEEE RAL*, vol. 10, no. 8, pp. 8003–8010, 2025.

[13] S. Seo, D. Han, Y. Chang, and N. Kwak, "Mixnerf: Modeling a ray with mixture density for novel view synthesis from sparse inputs," in *CVPR*, 2023, pp. 20 659–20 668.

[14] M. Niemeyer, J. T. Barron, B. Mildenhall, M. S. M. Sajjadi, A. Geiger, and N. Radwan, "Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs," in *CVPR*, 2022, pp. 5480–5490.

[15] Z. Lu, X. Guo, L. Hui, *et al.*, "3d geometry-aware deformable gaussian splatting for dynamic view synthesis," in *CVPR*, 2024, pp. 8900–8910.

[16] Z. Zou, Z. Yu, Y. Guo, *et al.*, "Triplane meets gaussian splatting: Fast and generalizable single-view 3d reconstruction with transformers," in *CVPR*, 2024, pp. 10 324–10 335.

[17] Z. Guo, J. Su, S. Wang, *et al.*, "Gp-gs: Gaussian processes for enhanced gaussian splatting," *arXiv preprint arXiv:2502.02283*, 2025.

[18] J. Wu, K. Liu, H. Gao, X. Jiang, and L. Zhang, "Dig3d: Marrying gaussian splatting with deformable transformer for single image 3d reconstruction," *arXiv preprint arXiv:2404.16323*, 2024.

[19] B. Mildenhall, P. P. Srinivasan, R. Ortiz-Cayon, *et al.*, "Local light field fusion: Practical view synthesis with prescriptive sampling guidelines," *TOG*, vol. 38, no. 4, pp. 1–14, 2019.

[20] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, and P. Hedman, "Mip-nerf 360: Unbounded anti-aliased neural radiance fields," in *CVPR*, 2022, pp. 5470–5479.