Eve3D: Elevating Vision Models for Enhanced 3D Surface Reconstruction via Gaussian Splatting

Jiawei Zhang[1*] **Youmin Zhang**[2†] **Fabio Tosi**[4] **Meiying Gu**[1] **Jiahe Li**[1] **Xiaohan Yu**[5]
Jin Zheng[1,3] **Xiao Bai**[1‡] **Matteo Poggi**[4]
[1]School of Computer Science and Engineering, State Key Laboratory of Complex Critical Software Environment, Jiangxi Research Institute, Beihang University
[2]Rawmantic AI [3]State Key Laboratory of Virtual Reality Technology and Systems, Beijing
[4]University of Bologna [5]Macquarie University

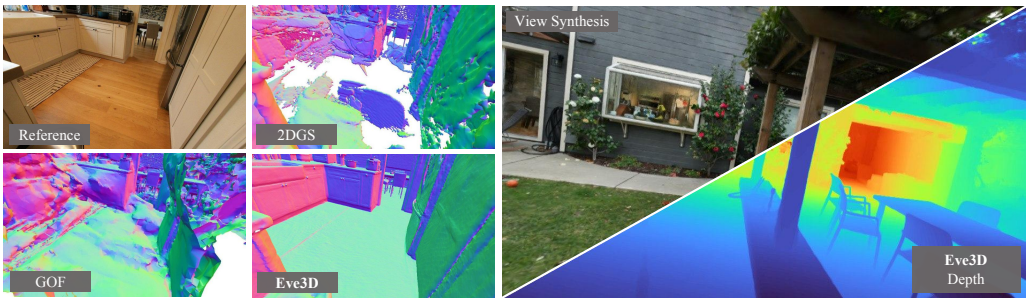


Figure 1: **Geometric Reconstruction and Rendering.** Examples from a challenging scene [11] showing our Eve3D method's accurate geometry and appearance. Left: Normal maps highlight superior surface reconstruction over 2DGS [18] and GOF [55], particularly on flat surfaces and object boundaries. Right: Photorealistic view synthesis alongside its detailed depth map rendering.

Abstract

We present Eve3D, a novel framework for dense 3D reconstruction based on 3D Gaussian Splatting (3DGS). While most existing methods rely on imperfect priors derived from pre-trained vision models, Eve3D fully leverages these priors by jointly optimizing both them and the 3DGS backbone. This joint optimization creates a mutually reinforcing cycle: the priors enhance the quality of 3DGS, which in turn refines the priors, further improving the reconstruction. Additionally, Eve3D introduces a novel optimization step based on bundle adjustment, overcoming the limitations of the highly local supervision in standard 3DGS pipelines. Eve3D achieves state-of-the-art results in surface reconstruction and novel view synthesis on the Tanks & Temples, DTU, and Mip-NeRF360 datasets. while retaining fast convergence, highlighting an unprecedented trade-off between accuracy and speed.

1 Introduction

Dense 3D scene reconstruction is a crucial task in computer vision and graphics, supporting applications ranging from virtual reality and simulating environments to robotic navigation. Recent breakthroughs in this field draw inspiration from the adjacent literature concerning novel view

[*]This work was conducted during Jiawei Zhang's research internship at Rawmantic AI.
[†]Project Lead
[‡]Corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

synthesis [32, 22], which faced a remarkable revolution in the last few years. Initially, Neural Radiance Fields (NeRFs) [32] were adapted to encode dense 3D surfaces within MLP weights [42, 25]; however, despite their compact representation, these approaches impose prohibitive computational demands, often requiring hundreds of hours of processing to reconstruct a single scene. More recently, 3D Gaussian Splatting [22] (3DGS) has gained significant attention in the graphics community, characterized by an optimized rasterizer that facilitates real-time rendering, establishing it as the current preferred alternative to NeRFs. This has led to the development of numerous 3DGS-based approaches [18, 55, 6, 29] that deliver promising results and high-fidelity scene reconstructions. However, despite the efficiency and flexibility of 3DGS and the latest advances, the framework continues to face significant limitations that hinder its broader application for 3D reconstruction task.

First and foremost, 3DGS in its original formulation is unsuited for this purpose, as Gaussian primitives rarely fit accurately to true surface geometries [16]. This limitation derives from two critical factors: the inherent independence of Gaussian primitives, which operate without contextual awareness, and the sole reliance on image reconstruction losses during training. While this approach excels at novel view synthesis, it often fails at properly modeling the real 3D geometry of the observed scene. Second, the accuracy of 3DGS heavily relies on a proper initialization strategy for seeding Gaussian primitives. Despite its efficiency and widespread adoption, COLMAP's [35] reliance on local feature makes it vulnerable to challenging scenarios such as textureless regions and repetitive patterns, resulting in structural distortions, blurring, and under-reconstructed areas.

To address these shortcomings, recent 3D reconstruction pipelines built upon 3DGS [18, 55, 6, 29] have incorporated additional scene *priors* from pre-trained vision models [48, 58, 4, 5, 12, 1, 15, 53]. Among these supplementary signals, *depth* and *surface normals* have proven most effective, significantly enhancing reconstruction quality in traditionally challenging scenarios such as textureless regions. On the one hand, although vision models can reason both locally and globally over their inputs, they are constrained by the limited amount of images they can process simultaneously – e.g., typically a single frame [48, 12, 1, 15, 53], a stereo pair [58, 44, 3], or at most about ten images [4, 5]. This limitation restricts these models to analyzing only localized portions of a scene when extracting priors. Furthermore, despite their training on large-scale datasets ranging from hundreds of thousands [4, 5] to several million images [48], the priors they predict still suffer from inaccuracies. These dual constraints create sub-optimal supervision for 3DGS, tampering with the optimization process and ultimately limiting the final 3D reconstruction accuracy.

In this paper, we introduce **Eve3D**, a novel 3D reconstruction framework built upon 3D Gaussian Splatting, specifically designed to overcome these limitations. Firstly, Eve3D unlocks the full potential of vision models by rendering synthetic rectified stereo pairs through 3DGS and applying state-of-the-art stereo foundation models to obtain geometric priors. These priors are then refined through joint optimization alongside the 3DGS model itself. This joint optimization strategy works by back-propagating gradients through both the depth rendered by 3DGS and the predicted depth priors, with the latter treated as learnable parameters throughout the process. Secondly, we propose a local bundle adjustment strategy that maintains global consistency across co-visible frames during each forward optimization. This approach overcomes 3DGS's inherent limitation in simultaneously rasterizing and optimizing multiple frames – a constraint imposed by computational complexity. This allows Eve3D to achieve unprecedented accuracy, as demonstrated in Fig. 1.

Eve3D is trained and evaluated over popular benchmarks for 3D reconstruction, including DTU [9] and Tanks and Temples [24], achieving state-of-the-art accuracy on both datasets. In particular, on the latter, this is done in just 20 minutes, while it takes only ~ 1 GPU hours to push the accuracy to the upper bound. In summary, the main contributions of this paper are:

- We introduce Eve3D, a novel framework for dense 3D reconstruction based on 3DGS, setting a new state-of-the-art in the field while maintaining efficient training time.
- We develop an improved supervision paradigm for 3DGS used to train Eve3D, which treats external priors as learnable parameters and optimizes them jointly with the 3DGS model during training.
- We introduce a local bundle adjustment to better enforce multi-view consistency during each optimization step, overcoming one of the main limitations of 3DGS-based frameworks.

2 Related Work

Neural Scene Representations. Neural scene representations have revolutionized 3D reconstruction and rendering. NeRF [32] pioneered this direction with continuous volumetric functions modeled by MLPs, though at high computational cost. More recently, 3DGS [22] introduced an explicit representation using 3D Gaussian primitives with efficient rasterization-based rendering, achieving superior quality and real-time performance. Despite successful applications in large-scale scene reconstruction [23, 28, 26], SLAM systems [47, 21, 37, 63, 40], dynamic scene modeling [49, 50, 46, 27], AI-generated content [38, 8, 34, 61], and autonomous driving [62, 30, 60], both NeRF and 3DGS focus primarily on appearance rather than geometry, resulting in poorly defined surfaces.

Neural Surface Reconstruction. To address these limitations, several methods have extended neural rendering for accurate surface reconstruction. NeuS [42] and VolSDF [52] represent surfaces as zero-level sets of signed distance functions without mask supervision. Neuralangelo [25] enhances detail with multi-resolution hash grids and numerical gradients, while NeuralWarp [10] improves consistency through patch warping. These implicit approaches produce high-quality surfaces but require significant computational resources and long optimization times.

Surface Reconstruction with Gaussian Splatting. 3DGS's efficiency has inspired many surface reconstruction methods. SuGaR [16] introduced regularization for surface-aligned Gaussians, enabling Poisson reconstruction. 2DGS [18] uses planar disks for surface modeling, while Gaussian Surfels [9] treats local z-axis as normal direction. GS2Mesh [45] extracts meshes using TSDF fusion on depth maps from a pre-trained stereo model, while StereoGS [33] employs self-improving depth supervision with virtual stereo pairs. Some approaches combine Gaussians with implicit fields: GOF [55] derives an opacity field, while GSDF [54] and 3DGSR [31] integrate 3DGS with signed distance functions. Others focus on geometric constraints, like PGSR [6] with unbiased depth rendering, DN-Splatter [41] with depth/normal priors, and VCR-GauS [7] with view-consistent depth-normal regularization. GS-Pull [57] aligns Gaussians to a neural SDF's zero-level set. Despite progress, current approaches have limitations with geometric priors - either using external vision model priors that may be inconsistent [48, 58, 4, 5, 12, 1, 15, 53], or applying heuristic constraints that struggle with complex geometries. Our work jointly optimizes Gaussians and priors in a unified framework, introducing non-local information flow for better geometric consistency.

3 Method

3.1 Framework Overview

Scene Representation. As illustrated in Fig. 2, our backbone is built over 3D Gaussian Splatting, which models the scene as a set of Gaussian primitives, each one defined as:

$$\mathcal{G}_i(\boldsymbol{x}|\boldsymbol{\mu}_i, \boldsymbol{\Sigma_i}) = e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\mu}_i)} \tag{1}$$

with $\boldsymbol{\mu}_i \in \mathbb{R}^3$ and $\boldsymbol{\Sigma}_i \in \mathbb{R}^{3\times3}$ being the center and 3D covariance matrix, respectively. The latter can be decomposed into scaling and rotation matrices $\boldsymbol{S}_i\, \boldsymbol{R}_i \in \mathbb{R}^{3\times3}$:

$$\boldsymbol{\Sigma}_i = \boldsymbol{R}_i \boldsymbol{S}_i \boldsymbol{S}_i^\top \boldsymbol{R}_i^\top \tag{2}$$

To better fit surfaces [18, 9], we enforce Gaussians to be flat by minimizing the minimal factor in $\boldsymbol{S}_i$. The pixel-wise color $\boldsymbol{C} \in \mathbb{R}^3$ is rendered through α-blending:

$$\boldsymbol{C} = \sum_{i\in N} T_i \alpha_i \boldsymbol{c}_i, \quad T_i = \prod_{j=1}^{i-1}(1-\alpha_i), \tag{3}$$

where α and $\boldsymbol{c}_i \in \mathbb{R}^3$ are the alpha and view-dependent color. Similarly, properties such as depth and surface normals can be rendered. Following [6], normals $\boldsymbol{N}$ are derived from the minimum scale factor direction $\boldsymbol{n}_i$ and camera rotation $\boldsymbol{R}_c$. Depth $\boldsymbol{D}$ are obtained through unbiased rendering [6], calculated as the intersection between the ray and the plane defined by rendered normals and the distance map $\boldsymbol{\mathcal{D}}$, where $d_i = (\boldsymbol{R}_c^T(\boldsymbol{\mu}_i - \boldsymbol{T}_c))^T(\boldsymbol{R}_c^T\boldsymbol{n}_i)$ represents the distance from the plane to the camera center $\boldsymbol{T}_c$. The two geometric properties follow:

$$\boldsymbol{N} = \sum_{i\in N} \boldsymbol{R}_c^T \boldsymbol{n}_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \quad \boldsymbol{\mathcal{D}} = \sum_{i\in N} d_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j). \tag{4}$$

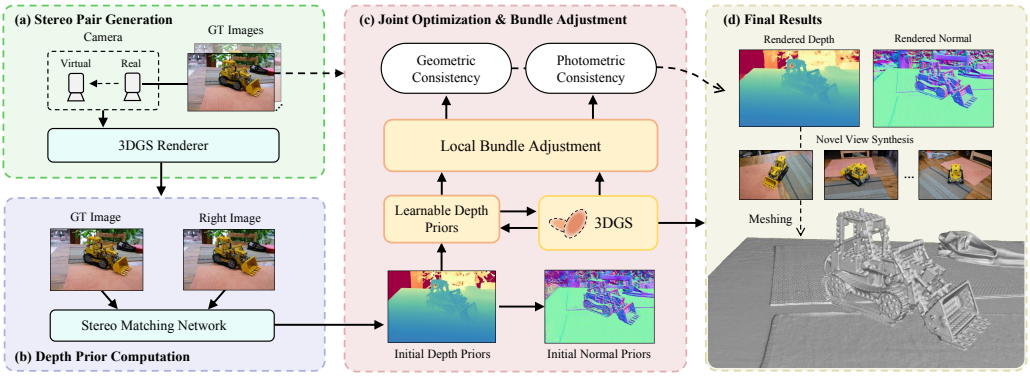


Figure 2: **Overview of our Framework.** (a) We generate stereo pairs by rendering from a virtual camera using 3DGS. (b) A pre-trained stereo network extracts initial depth priors. (c) Our core contribution: joint optimization of learnable depth priors and 3DGS, enhanced by local bundle adjustment for geometric and photometric consistency. (d) Final results showing high-quality mesh reconstruction, accurate depth/normal maps, and high-fidelity novel view synthesis.

While color can be supervised with real images, depth and normals can be supervised using *priors* from pre-trained vision models [48, 58, 4, 5, 12, 1, 15, 53] for better optimization.

Priors Computation. Unlike approaches supervising 3DGS with ill-posed monocular depth models [48, 12], we use binocular stereo matching [39] for more geometrically consistent priors. Since our method operates in a standard multi-view setting with a single moving camera rather than a stereo rig, we exploit 3DGS's view synthesis to render rectified stereo pairs [33, 45]. Specifically, for a given camera pose $\mathbf{P}_i$, we define a virtual right camera with pose $\mathbf{P}_i^r$, at a distance b as:

$$\mathbf{P}_i^r = \begin{pmatrix} \mathbb{I} & \mathbf{t} \\ 0 & 1 \end{pmatrix} \times \mathbf{P}_i \qquad \text{with} \quad \mathbf{t} = \begin{pmatrix} b & 0 & 0 \end{pmatrix}^\top \tag{5}$$

We render a virtual right image and form a stereo pair with the input left image $(\mathbf{I}_i, \mathbf{I}_i^r)$. We then apply a pre-trained state-of-the-art stereo model – FoundationStereo [44] in our implementation – to predict a disparity map, which is converted to depth $\boldsymbol{D}^*$ through triangulation. We also derive a confidence mask $\boldsymbol{M}^c$ by computing the consistency check between disparity maps for left and right views. While effective for assisting 3DGS optimization [33] and recovering meshes [45], we argue that explicitly addressing the noisy nature of these priors is critical to fully exploit their potential.

3.2 Prior-Involved Local Bundle Adjustment

Multi-view consistency is vital for determining accurate surfaces. However, extracting accurate and detailed surfaces from either Gaussians or vision models presents challenges without multi-view constraints. This difficulty arises because 3DGS captures depth information through image rendering supervision, while vision models inherently contain noise and have limited view inputs. To address this issue, we have developed a local bundle adjustment algorithm that utilizes both rendered and prior depth maps, enhancing their multi-view consistency to represent surfaces more accurately. In multi-view pairs, we employ priors at source views, which allows us to incorporate more viewpoints in a single loop by eliminating the need for depth rendering. For the reference view, we select either the rendered or prior depth map, depending on the stage of joint optimization (Sec 3.3).

For view $\boldsymbol{V}_i$ at the current iteration, we build a factor-graph $(\mathcal{V}, \mathcal{E})$ to perform local bundle adjustment. To balance reconstruction quality and efficiency, rather than optimizing all frames together, we only select views that overlap with the current view i. Specifically, following standard MVS methods [51, 4, 5], we determine neighboring views by computing an overlapping score and select only the top K frames $\{\boldsymbol{V}_j\}_{j=1}^K$. We construct the graph by adding edge connections between the current view $\boldsymbol{V}_i$ and each of its neighboring views $\boldsymbol{V}_j$. The underlying optimization principle enforces both geometric and photometric consistency.

Geometric Consistency. Given learnable depth $\hat{\boldsymbol{D}}_i$ of the current view $\boldsymbol{V}_i$, we first convert it into a normal map $\boldsymbol{\mathcal{N}}_{\hat{\boldsymbol{D}}_i}$ with finite differences as in [18], and derive the distance map $\hat{\boldsymbol{\mathcal{D}}}_i$ as:

$$\hat{\boldsymbol{\mathcal{D}}}_i(\boldsymbol{p}) = \hat{\boldsymbol{D}}_i(\boldsymbol{p})\boldsymbol{\mathcal{N}}^T_{\hat{\boldsymbol{D}}_i}(\boldsymbol{p})\boldsymbol{K}_i^{-1}\hat{\boldsymbol{p}}, \tag{6}$$

where $\boldsymbol{p}$ is the 2D position on the image plane, $\hat{\boldsymbol{p}}$ denotes its homogeneous coordinate, and $\boldsymbol{K}_i$ is the camera intrinsic matrix. We then map the full set of pixel coordinates $\boldsymbol{P}_i$ from the current view $\boldsymbol{V}_i$ to the neighboring view $\boldsymbol{V}_j$ through the homography matrix $\boldsymbol{H}_{ij}$:

$$\hat{\boldsymbol{P}}_j = \boldsymbol{H}_{ij}\boldsymbol{P}_i, \quad \boldsymbol{H}_{ij} = \boldsymbol{K}_j(\boldsymbol{R}_{ij} - \frac{\boldsymbol{T}_{ij}\boldsymbol{\mathcal{N}}^T_{\hat{\boldsymbol{D}}_i}}{\hat{\boldsymbol{\mathcal{D}}}_i})\boldsymbol{K}_i^{-1} \tag{7}$$

where $\boldsymbol{R}_{ij}$ and $\boldsymbol{T}_{ij}$ are the relative rotation and translation from view $\boldsymbol{V}_i$ to the neighboring view $\boldsymbol{V}_j$. Similarly, for pixels in the neighboring view $\boldsymbol{V}_j$, we derive the surface normal and distance map from its learnable depth $\hat{\boldsymbol{D}}_j$ to compute the homography matrix $\boldsymbol{H}_{ji}$. By enforcing geometry consistency, we aim at minimizing the projection error $\min \boldsymbol{\Phi}_{ij}$, where

$$\boldsymbol{\Phi}_{ij} = \| \boldsymbol{P}_i - \boldsymbol{H}_{ji}\boldsymbol{H}_{ij}\boldsymbol{P}_i \| . \tag{8}$$

Photometric Consistency. This constraint is based on plane patches. For pixels $\boldsymbol{P}_i$ in the current view $\boldsymbol{V}_i$, we map a 7×7 pixel patch centered at each pixel $\boldsymbol{p} \in \boldsymbol{P}_i$ to the neighboring view using the homography matrix $\boldsymbol{H}_{ij}$. We aim to minimize the photometric error to zero:

$$\boldsymbol{\Psi}_{ij} = \left(1 - \mathrm{NCC}\left(\boldsymbol{I}_i\left(\boldsymbol{P}_i\right), \boldsymbol{I}_j\left(\boldsymbol{H}_{ij}\boldsymbol{P}_i\right)\right)\right), \tag{9}$$

Objective. To account for occlusions between views and noise in depth estimates, following [6], we model the confidence map to weight the error function $\boldsymbol{W}_{ij}$ and the overall cost function to refine depth maps of the current view and its neighbors is defined as:

$$\mathcal{L}_{lba} = \lambda_{lba} \sum_{(s,t)\in\mathcal{E}} \boldsymbol{W}_{st}\left(\lambda_g\boldsymbol{\Phi}_{st} + \lambda_p\boldsymbol{\Psi}_{st}\right). \tag{10}$$

3.3 Joint Optimization of 3DGS and Priors

Regardless of their source, all model-generated priors inevitably contain noise and inaccuracies. Rather than treating them as rigid supervision, we jointly optimize both the 3DGS model and the associated guidance. This creates a mutually beneficial relationship, refining priors through multi-view consistency while providing improved supervision for the 3DGS representation.

Parameterized Prior. Given a prior depth map $\boldsymbol{D}^*$, we initialize a set of learnable parameters $\hat{\boldsymbol{D}}$ with its values. The joint optimization follows a two-phase schedule. Initially (i.e., before iteration T_{joint}), we use the original $\boldsymbol{D}^*$ to supervise 3DGS and avoid early convergence to poor local minima. Later, once the 3DGS-rendered depth becomes sufficiently reliable, we switch to optimizing $\hat{\boldsymbol{D}}$ jointly with 3DGS parameters through backpropagation, allowing the model to refine both the scene representation and the the guidance signal simultaneously without degrading their quality.

Local Bundle Adjustment Pre-training. Neural networks that generate prior depth estimates often do so at very sparse viewpoints, resulting in insufficient multi-view consistency. To address this limitation, before starting the joint optimization (T_{joint}), we apply our proposed local bundle adjustment between the prior depths of different viewpoints, which will provide a multi-view consistent prior initialization before starting the joint optimization of 3DGS and prior depth.

Confidence Mask Update. As the quality of depth prior gradually improves through the joint optimization and local bundle adjustment, the initial confidence mask $\hat{\boldsymbol{M}}^c$ may become outdated. Concurrently, the confidence map computed during local bundle adjustment reflects the quality of the latest depth prior. Therefore, we update the mask $\hat{\boldsymbol{M}}^c$ as:

$$\hat{\boldsymbol{M}}^c_i = \hat{\boldsymbol{M}}^c_i \vee (\boldsymbol{W}_{ij} > 0) \quad \text{for } j \in \{1, 2, \cdots, K\} \tag{11}$$

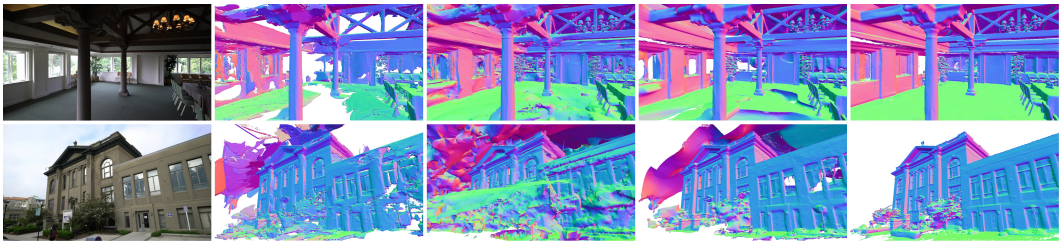
For pixels with value 0 in $\hat{\boldsymbol{M}}^c$, i.e., low-quality depth prior, it remains possibile for them to be adjusted to a more accurate position by 3DGS:

$$\mathcal{L}_{pull} = \lambda_{pull}(\sim \hat{\boldsymbol{M}}^c)||\hat{\boldsymbol{D}} - \boldsymbol{D}^{\text{detach}}||_1, \tag{12}$$

where $\boldsymbol{D}^{\text{detach}}$ denotes the gradient of rendered depth $\boldsymbol{D}$ is detached. When such pixels attain a sufficiently accurate state and are subsequently classified as confident by local bundle adjustment, they become eligible to participate in joint optimization and local bundle adjustment, ultimately contributing to accurate surface reconstruction.

Table 1: **Quantitative results of F1 Score on Tanks and Temples.** , , indicate the absolute, second, and third bests respectively.

	Method	Barn	Caterpillar	Courthouse	Ignatius	Meetingroom	Truck	Mean ↑	Time
Implicit	NeuS [42]	0.29	0.29	0.17	0.83	0.24	0.45	0.38	>24h
	Geo-Neus [14]	0.33	0.26	0.12	0.72	0.20	0.45	0.35	>24h
	Neuralangelo [25]	0.70	0.36	0.28	0.89	0.32	0.48	0.50	>128h
Explicit	3DGS [22]	0.13	0.08	0.09	0.04	0.01	0.19	0.09	20m
	SuGaR [16]	0.14	0.16	0.08	0.33	0.15	0.26	0.19	2h
	DN-Splatter [41]	0.15	0.11	0.07	0.18	0.01	0.20	0.12	1h
	GSurfels [9]	0.24	0.22	0.07	0.39	0.12	0.24	0.21	15m
	2DGS [18]	0.36	0.23	0.13	0.44	0.16	0.26	0.30	34m
	GOF [55]	0.51	0.41	0.28	0.68	0.28	0.58	0.46	2h
	PGSR [6]	0.66	0.44	0.20	0.81	0.33	0.66	0.52	45m
	GS-Pull [57]	0.60	0.37	0.16	0.71	0.22	0.52	0.43	38m
	Eve3D-*fast* (Ours)	0.69	0.44	0.34	0.82	0.41	0.62	0.56	20m
	Eve3D (Ours)	0.70	0.48	0.35	0.83	0.46	0.66	0.58	1.2h



Reference Image 2DGS [18] GOF [55] PGSR [6] **Eve3D (Ours)**

Figure 3: **Qualitative Comparison on Tanks and Temples.** We visualize the surface normal of reconstructed 3D meshes for comparison.

3.4 Training Loss

With the prior depth, we apply a single-view prior loss $\mathcal{L}_{depth}$ to regularize the depth and normal from 3D Gaussians in confident regions weighted by $\hat{\boldsymbol{M}}^c$:

$$\mathcal{L}_{prior}^{depth} = \hat{\boldsymbol{M}}^c \odot \begin{cases} ||\boldsymbol{D}^* - \boldsymbol{D}||_1, & \text{iter} < \text{T}_{\text{joint}}, \\ ||\hat{\boldsymbol{D}} - \boldsymbol{D}||_1, & \text{otherwise.} \end{cases} \tag{13}$$

To strengthen the impact on plane surface reconstruction, we also jointly optimize the normals obtained from 3DGS and the depth prior in confident regions as follows:

$$\mathcal{L}_{prior}^{normal} = \hat{\boldsymbol{M}}^c \odot \begin{cases} \Gamma(\boldsymbol{\mathcal{N}}_{\boldsymbol{D}^*}, \boldsymbol{\mathcal{N}}_{\boldsymbol{D}}) + \Gamma(\boldsymbol{\mathcal{N}}_{\boldsymbol{D}^*}, \boldsymbol{N}), & \text{iter} < \text{T}_{\text{joint}}, \\ \Gamma(\boldsymbol{\mathcal{N}}_{\hat{\boldsymbol{D}}}, \boldsymbol{\mathcal{N}}_{\boldsymbol{D}}) + \Gamma(\boldsymbol{\mathcal{N}}_{\hat{\boldsymbol{D}}}, \boldsymbol{N}), & \text{otherwise,} \end{cases} \tag{14}$$

where $\Gamma(\boldsymbol{A}, \boldsymbol{B})$ denotes the pixel-wise cosine distance $(1 - \boldsymbol{A} \cdot \boldsymbol{B})$ between normal maps. In addition, we also use ground-truth color loss $\mathcal{L}_c$ [22], depth-normal consistency loss $\mathcal{L}_{dn}$ to encourage consistency between rendered depth and rotational normal [6, 18, 55], and scale loss $\mathcal{L}_s$ [6] to encourage Gaussians to flatten to planes. The final training loss is:

$$\mathcal{L} = \mathcal{L}_c + \mathcal{L}_{dn} + \mathcal{L}_s + \mathcal{L}_{prior} + \mathcal{L}_{lba} + \mathcal{L}_{pull}. \tag{15}$$

4 Experiments

4.1 Experimental Setup

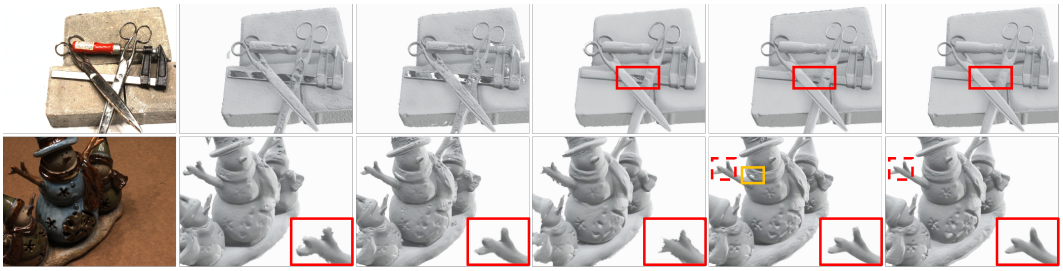
Datasets. We evaluate on real-world datasets, including object-centric, indoor, and outdoor scenes. For 3D reconstruction, we use large-scale scenes from Tanks and Temples [24] and 15 object-centric scenes from DTU [20]. For novel view synthesis, we use the Mip-NeRF360 dataset [2].

Evaluation Metrics. Following established protocols, we assess reconstruction accuracy using Chamfer Distance (CD) on DTU [9] and F-score on Tanks and Temples [24], employing the official evaluation scripts. For novel view synthesis evaluation on Mip-NeRF360, we use standard rendering quality metrics: PSNR, SSIM [43], and LPIPS [56].

Baselines. We compare against state-of-the-art methods from two categories: (1) implicit NeRF-based approaches including NeRF [32], VolSDF [52], NeuS [42], Geo-Neus [14], NeuralWarp [10], and Neuralangelo [25]; and (2) explicit 3DGS-based frameworks including 3DGS [22], SuGaR [16], DN-Splatter [41], GSurfels [9], 2DGS [18], GOF [55], GS2Mesh [45], PGSR [6], and GS-Pull [57].

Table 2: **Quantitative Results (Chamfer Distance) on DTU.** , , indicate the absolute, second, and third bests respectively.

	Method	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean ↓	Time
Implicit	NeRF [32]	1.90	1.60	1.85	0.58	2.28	1.27	1.47	1.67	2.05	1.07	0.88	2.53	1.06	1.15	0.96	1.49	> 12h
	VolSDF [52]	1.14	1.26	0.81	0.49	1.25	0.70	0.72	1.29	1.18	0.70	0.66	1.08	0.42	0.61	0.55	0.86	>12h
	NeuS [42]	1.00	1.37	0.93	0.43	1.10	0.65	0.57	1.48	1.09	0.83	0.52	1.20	0.35	0.49	0.54	0.84	>12h
	NeuralWarp [10]	0.49	0.71	0.38	0.38	0.79	0.81	0.82	1.20	1.06	0.68	0.66	0.74	0.41	0.63	0.51	0.68	>10h
	Neuralangelo [25]	0.37	0.72	0.35	0.35	0.87	0.54	0.53	1.29	0.97	0.73	0.47	0.74	0.32	0.41	0.43	d0.61	>12h
Explicit	3DGS [22]	2.14	1.53	2.08	1.68	3.49	2.21	1.43	2.07	2.22	1.75	1.79	2.55	1.53	1.52	1.50	1.96	12m
	SuGaR [16]	1.47	1.33	1.13	0.61	2.25	1.71	1.15	1.63	1.62	1.07	0.79	2.45	0.98	0.88	0.79	1.33	1h
	DN-Splatter [41]	1.60	2.03	1.42	1.44	2.37	2.11	1.62	1.95	1.88	1.48	1.63	1.82	1.20	1.50	1.40	1.70	30m
	GSurfels [9]	0.66	0.93	0.54	0.41	1.06	1.14	0.85	1.29	1.53	0.79	0.82	1.58	0.45	0.66	0.53	0.88	11m
	2DGS [18]	0.48	0.91	0.39	0.39	1.01	0.83	0.81	1.36	1.27	0.76	0.70	1.40	0.40	0.76	0.52	0.80	20m
	GOF [55]	0.50	0.82	0.37	0.37	1.12	0.74	0.73	1.18	1.29	0.68	0.77	0.90	0.42	0.66	0.49	0.74	2h
	GS2Mesh [45]	0.59	0.79	0.70	0.38	0.78	1.00	0.69	1.25	0.96	0.59	0.50	0.68	0.37	0.50	0.46	0.68	20m
	PGSR [6]	0.36	0.57	0.38	0.33	0.78	0.58	0.50	1.08	0.63	0.59	0.46	0.54	0.30	0.38	0.34	0.52	30m
	GS-Pull [57]	0.51	0.56	0.46	0.39	0.82	0.67	0.85	1.37	1.25	0.73	0.54	1.39	0.35	0.88	0.42	0.75	22m
	Eve3D (Ours)	0.33	0.47	0.32	0.33	0.73	0.58	0.44	1.00	0.62	0.54	0.43	0.45	0.29	0.38	0.32	0.48	15m



Reference Image 2DGS [18] GOF [55] GS2Mesh [45] PGSR [6] **Eve3D (Ours)**

Figure 4: **Qualitative Comparison on the DTU Dataset.** Visual comparison of 3D meshes reconstructed by our approach versus previous methods.

Implementation Details. Training Eve3D occurs in two phases: first, we pre-train a vanilla 3DGS model using [13], which we use to render pseudo stereo views that feed into [44] to predict dense depth maps $\boldsymbol{D}^*$ as priors. We use the pretrained model without fine-tuning, ensuring zero-shot generalization with no overlap between its training data and our evaluation datasets. We employ a left-right consistency check with a 3-pixel threshold to estimate confidence masks $\boldsymbol{M}^{\boldsymbol{c}}$. We set T_{joint} to 7000, introduce depth prior supervision after the first 500 iterations, and apply local bundle adjustment from the beginning of training. We also built the Eve3D-*fast* variant, reducing the total iterations from 30k to 5k and setting T_{joint} to 1000. Following PGSR [6], we set $\lambda_{dn} = 0.015$, $\lambda_s = 100.0$, and $\lambda_c = 1.0$. For our proposed losses, we use $\lambda_{prior} = 0.05$, $\lambda_{pull} = 0.05$, and $\lambda_{lba} = 0.15$. These hyperparameters remain fixed across all datasets without tuning. Finally, we use $K = 4$ neighboring views for local bundle adjustment across all experiments.

4.2 Evaluation Against State-of-the-Art

Tanks and Temples. The Tanks and Temples dataset [24] collects scenes in a surround manner, comprising six diverse environments that include both indoor and outdoor scenarios with varying scales and lighting conditions. Tab. 1 presents our quantitative evaluation based on F1-score (the higher the better). Eve3D achieves superior performance with an average of 0.58, outperforming both implicit methods and other explicit approaches. Our method achieves the highest ranking in five scenes (Barn, Caterpillar, Courthouse, Meeting Room, and Truck) and ranks second in one scene (Ignatius). The most significant improvements appear in challenging scenarios: Eve3D achieves an F1 Score of 0.46 in Meetingroom (compared to 0.33 from PGSR) and 0.35 in Courthouse (compared to 0.28 from the best competitors). Despite these substantial quality improvements, Eve3D (trained for 30K iterations) maintains a reasonable 1.2-hour total training time (including 3DGS pretraining, stereo rendering, and depth prediction) – significantly faster than implicit methods (>24h) while delivering superior reconstruction quality. Our Eve3D-*fast* variant, trained for only 5K iterations (20 minutes total time), still achieves second-best reconstruction quality with an average F1-score of 0.56, setting an unprecedented trade-off between accuracy and speed. Fig. 3 shows Eve3D addresses standard 3DGS limitations in scenes with varied lighting and complex architecture, producing more accurate flat surfaces and detailed structures in both indoor and outdoor settings.

Table 3: **Quantitative Comparisons of Novel View Synthesis on the Mip-NeRF360 Dataset.** , , indicate the absolute, second, and third bests, respectively.

	Outdoor Scenes			Indoor Scenes		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
NeRF [32]	21.46	0.458	0.515	26.84	0.790	0.370
Deep Blending [17]	21.54	0.524	0.364	26.40	0.844	0.261
Instant NGP	22.90	0.566	0.371	29.15	0.880	0.216
MERF	23.19	0.616	0.343	27.80	0.855	0.271
BakedSDF	22.47	0.585	0.349	27.06	0.836	0.258
MipNeRF360	24.47	0.691	0.283	31.72	0.917	0.180
SuGaR [16]	22.93	0.629	0.356	29.43	0.906	0.225
3DGS [22]	24.64	0.731	0.234	30.41	0.920	0.189
2DGS [18]	24.34	0.717	0.246	30.40	0.916	0.195
GOF [55]	24.76	0.742	0.225	30.80	0.928	0.167
PGSR [6]	24.76	0.752	0.203	30.36	0.934	0.147
Eve3D (Ours)	24.99	0.758	0.203	30.42	0.930	0.157

Reference Image | 2DGS [18] | GOF [55] | PGSR [6] | **Eve3D (Ours)**

Figure 5: **Qualitative Comparison on the Mip-NeRF360 Dataset.** Visual comparison of 3D meshes reconstructed by our approach versus previous methods.

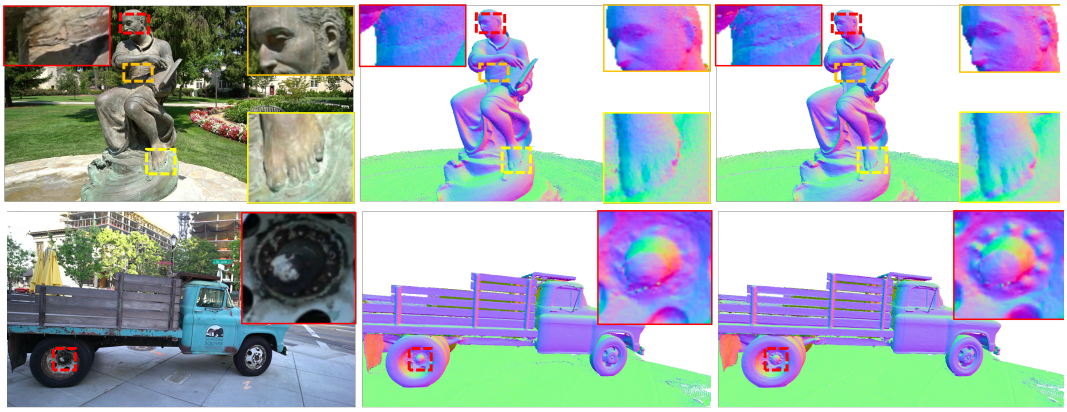
DTU. Tab. 2 shows our Chamfer Distance evaluation (lower is better) on DTU [20]. Eve3D achieves the best average performance with a score of 0.48, outperforming both the best implicit method (Neuralangelo at 0.61) and the previous best explicit approach (PGSR at 0.52). Our method demonstrates remarkable consistency, achieving the best performance in any scenes except one, where it is the second-best. Importantly, our method maintains an efficient 15-minute total training time (including all preprocessing steps), comparable to vanilla 3DGS while delivering largely superior reconstruction quality. Qualitative comparisons in Fig. 4 show that while existing 3DGS methods produce good object meshes, they struggle with sparse viewpoints and inconsistent multi-view images. Our method enhances robustness in these difficult scenarios, yielding more complete, detailed meshes with better preserved fine structures.

Mip-NeRF360. For validating rendering quality, we evaluate on the Mip-NeRF360 dataset [2] following 3DGS's standard protocol. Tab. 3 shows our results using standard metrics. Eve3D achieves excellent synthesis across outdoor and indoor scenes. For outdoor scenes, Eve3D delivers best performance on all metrics: PSNR (24.99), SSIM (0.758), and LPIPS (0.203), outperforming both NeRF-based approaches and recent 3DGS methods. In indoor scenes, instead, our method achieves the second-highest SSIM (0.930) and LPIPS (0.157) scores, while maintaining competitive PSNR (30.42, third-best after MipNeRF360 and GOF). This demonstrates that our geometry-aware optimization also enhances rendering quality. Fig. 5 further highlights this qualitatively.

4.3 Ablation Study

We ablate the key components of Eve3D on the Tanks and Temples [24] dataset in Tab. 4.

Single-view Prior Loss. Due to the lack of explicit geometric constraints, the baseline model struggles to reconstruct accurate surfaces relying solely on RGB supervision. When a single-view depth prior is introduced to constrain the rendered depth and normals of 3D Gaussians, the surface reconstruction performance improves from 0.340 to 0.463. We also ablate the use of single-view prior loss in a joint optimization setting, where the prior depth maps are treated as learnable parameters. This leads to a further improvement from 0.523 to 0.539. Please refer to the supplementary material for experiments using different types of single-view priors derived from other vision models.



Reference w/o Joint Optimization w/ Joint Optimization

Figure 6: **Qualitative Mesh Comparison.** Results without (w/o) and with (w/) joint optimization.

Local Bundle Adjustment. Local bundle adjustment applies explicit multi-view consistency constraints on the rendered depth and prior depth maps. Solely with local bundle adjustment, the performance is improved significantly from 0.340 to 0.523. In this configuration, prior depths guide the multi-view consistency but remain non-learnable. Enabling joint optimization, which treats priors as learnable parameters $\hat{\boldsymbol{D}}$, yields further improvement from 0.523 to 0.539, demonstrating that allowing the model to refine priors through backpropagation enhances both prior quality and 3DGS reconstruction. When using both single-view prior loss and local bundle adjustment, jointly optimizing 3DGS and depth priors also has significant effects, improving the F1 score from 0.553 to 0.581.

Table 4: **Component Contribution.** Evaluation of each module's effect on reconstruction quality.

Single-view Prior Loss	Local Bundle Adjustment	Joint Optimization	Evaluation P ↑	R ↑	F1 ↑
			0.297	0.418	0.340
✓			0.431	0.519	0.463
✓		✓	0.437	0.523	0.467
	✓		0.483	0.594	0.523
	✓	✓	0.504	0.605	0.539
✓	✓		0.531	0.595	0.553
✓	✓	✓	0.553	0.631	0.581

Joint Optimization. We analyze the impact of each joint optimization component in Tab 5. The local bundle adjustment pre-training plays an important role in initializing multi-view consistent priors; removing it leads to a performance drop from 0.581 to 0.574. Disabling the confidence mask update causes a slight drop to 0.578, highlighting the benefit of including more prior regions—when validated by geometry checks—during joint optimization. Fig 6 compares results with and without joint optimization. Jointly optimizing Gaussians and priors enables reconstruction of fine details, while disabling it leads to over-smoothed surfaces due to excessive reliance on prior supervision. Furthermore, this joint strategy is also especially effective for recovering geometry details that are very ambiguous from single-view visual clues, such as the dark areas in the Truck.

Table 5: **Component Contribution – Joint Optimization.** Evaluation of each module's effect.

Methods	P ↑	R ↑	F1 ↑
Joint Optimization (Full)	0.553	0.631	0.581
w/o LBA Pre-training	0.549	0.618	0.574
w/o Confidence Mask Update	0.552	0.626	0.578

In Tab. 6, we compare the FoundationStereo [44] prior depths before and after joint optimization with Eve3D. We evaluate depth accuracy on the Tanks and Temples dataset using ground-truth depth maps provided by the RobustMVD benchmark. Among the four scenes available in RobustMVD, we use the three that overlap with our experimental setup: Barn, Courthouse, and Ignatius.

Since the camera poses estimated by COLMAP are not aligned with the ground-truth poses—leading to inconsistencies in depth scale—we perform mesh-to-mesh alignment between the reconstructed and ground-truth geometry to obtain accurate scaling and alignment information. We then compute the relative depth error to quantify performance.

As shown in the comparison, in all evaluated scenes, the Eve3D rendering results consistently outperform the initial priors in depth accuracy. Furthermore, our proposed joint optimization strategy improves the accuracy of the initial depth priors.

Table 6: **Impacts of Joint Optimization.** We evaluate the accuracy of the learned depth priors before and after joint optimization using ground-truth depth maps from the Tanks and Temples dataset, provided by the Robust Multi-view Depth (RobustMVD) benchmark [36]. Relative error rates are reported to quantify the improvement in depth estimation.

Methods	Barn (%)	Courthouse (%)	Ignatius (%)
FoundationStereo [44]	1.84	12.25	1.43
Optimized FoundationStereo	1.51	11.96	1.00
Eve3D Depth	1.48	11.79	0.80

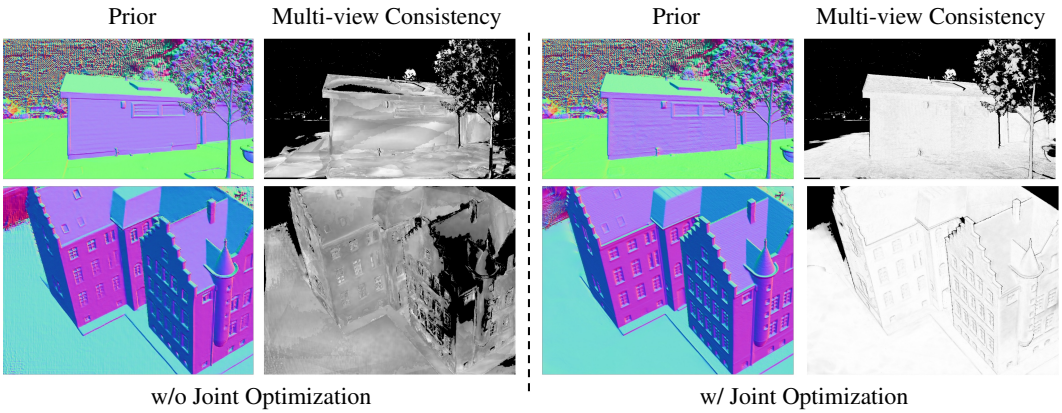


Figure 7: **Visualization of the multi-view consistency without and with the joint optimization.** The consistency of current-view depth priors is measured by computing the pixel-wise reprojection differences with the eight nearest neighbor views, followed by a consistency measurement using an exponential decay function, $\exp(-d)$, where d denotes the reprojection error. This weighting emphasizes geometrically consistent regions and suppresses unreliable estimates. Joint optimization significantly improves the multi-view consistency.

Additionally, in Fig 7, we highlight another key property of the priors: multi-view consistency. While initial priors provide reasonable but coarse estimates of depth and normals, they often exhibit inconsistency across different viewpoints. This inconsistency introduces noise, which can degrade the quality of supervision during Gaussian optimization. With our proposed joint optimization strategy, the multi-view consistency of priors is significantly improved, resulting in more stable and accurate supervision signals.

5 Conclusion

We presented Eve3D, a novel framework for dense surface reconstruction based on 3DGS. Our approach jointly optimizes both self-derived stereo depth priors and the 3DGS representation, establishing a mutually beneficial relationship in which each component improves the other. Our local bundle adjustment strategy ensures global consistency across view-overlapping frames, effectively compensating for the local supervision limitations inherent in 3DGS. Extensive experiments on Tanks & Temples, DTU, and Mip-NeRF360 demonstrate that Eve3D achieves state-of-the-art performance in both surface reconstruction and novel view synthesis, while training in as little as 15-20 minutes for our fast version, and $\sim$1 GPU hours for our base approach.

Limitations. Eve3D sets a new state-of-the-art, yet with some trade-offs. Its primary constraint is the reliance of a vision foundation model for stereo depth estimation. This choice is motivated by the unpaired accuracy of the estimated depth priors compared to alternative monocular [48] or multi-view stereo [19] solutions, as discussed in the supplementary material. However, it requires rendering stereo images from the model itself – an overhead that could be avoided if a multi-view stereo foundation model could achieve comparable accuracy.

Acknowledgments and Disclosure of Funding

This work is supported by Rawmantic AI, and the National Natural Science Foundation of China 62276016, 62372029.

References

[1] Gwangbin Bae and Andrew J Davison. Rethinking inductive biases for surface normal estimation. In *CVPR*, pages 9535–9545, 2024.

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5855–5864, 2021.

[3] Luca Bartolomei, Fabio Tosi, Matteo Poggi, and Stefano Mattoccia. Stereo anywhere: Robust zero-shot deep stereo matching even where either stereo or mono fail. *arXiv preprint arXiv:2412.04472*, 2024.

[4] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer: Multi-view stereo by learning robust image features and temperature-based depth. *arXiv preprint arXiv:2208.02541*, 2022.

[5] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer++: Revealing the devil in transformer's details for multi-view stereo. *arXiv preprint arXiv:2401.11673*, 2024.

[6] Danpeng Chen, Hai Li, Weicai Ye, Yifan Wang, Weijian Xie, Shangjin Zhai, Nan Wang, Haomin Liu, Hujun Bao, and Guofeng Zhang. Pgsr: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics*, 2024.

[7] Hanlin Chen, Fangyin Wei, Chen Li, Tianxin Huang, Yunsong Wang, and Gim Hee Lee. Vcr-gaus: View consistent depth-normal regularizer for gaussian surface reconstruction. *Advances in Neural Information Processing Systems*, 37:139725–139750, 2024.

[8] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *CVPR*, pages 21401–21412, 2024.

[9] Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using gaussian surfels. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024.

[10] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *CVPR*, pages 6260–6269, 2022.

[11] Daniel Duckworth, Peter Hedman, Christian Reiser, Peter Zhizhin, Jean-François Thibert, Mario Lučić, Richard Szeliski, and Jonathan T. Barron. Smerf: Streamable memory efficient radiance fields for real-time large-scene exploration, 2023.

[12] Ainaz Eftekhar, Alexander Sax, Jitendra Malik, and Amir Zamir. Omnidata: A scalable pipeline for making multi-task mid-level vision datasets from 3d scans. In *ICCV*, pages 10786–10796, 2021.

[13] Guangchi Fang and Bing Wang. Mini-splatting2: Building 360 scenes within minutes via aggressive gaussian densification. *arXiv preprint arXiv:2411.12788*, 2024.

[14] Qiancheng Fu, Qingshan Xu, Yew Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *NeurIPS*, 35:3403–3416, 2022.

[15] Xiao Fu, Wei Yin, Mu Hu, Kaixuan Wang, Yuexin Ma, Ping Tan, Shaojie Shen, Dahua Lin, and Xiaoxiao Long. Geowizard: Unleashing the diffusion priors for 3d geometry estimation from a single image. In *ECCV*, pages 241–258. Springer, 2024.

[16] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *CVPR*, 2024.

[17] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel Brostow. Deep blending for free-viewpoint image-based rendering. *ACM Transactions on Graphics (ToG)*, 37(6):1–15, 2018.

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

[19] Sergio Izquierdo, Mohamed Sayed, Michael Firman, Guillermo Garcia-Hernando, Daniyar Turmukhambetov, Javier Civera, Oisin Mac Aodha, Gabriel J. Brostow, and Jamie Watson. MVSAnywhere: Zero shot multi-view stereo. In *CVPR*, 2025.

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

[21] Nikhil Keetha, Jay Karhade, Krishna Murthy Jatavallabhula, Gengshan Yang, Sebastian Scherer, Deva Ramanan, and Jonathon Luiten. Splatam: Splat track & map 3d gaussians for dense rgb-d slam. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21357–21366, 2024.

[22] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG*, 42(4):139–1, 2023.

[23] Bernhard Kerbl, Andreas Meuleman, Georgios Kopanas, Michael Wimmer, Alexandre Lanvin, and George Drettakis. A hierarchical 3d gaussian representation for real-time rendering of very large datasets. *ACM TOG*, 43(4):1–15, 2024.

[24] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM TOG*, 36(4):1–13, 2017.

[25] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *CVPR*, 2023.

[26] Jiaqi Lin, Zhihao Li, Xiao Tang, Jianzhuang Liu, Shiyong Liu, Jiayue Liu, Yangdi Lu, Xiaofei Wu, Songcen Xu, Youliang Yan, et al. Vastgaussian: Vast 3d gaussians for large scene reconstruction. In *CVPR*, pages 5166–5175, 2024.

[27] Youtian Lin, Zuozhuo Dai, Siyu Zhu, and Yao Yao. Gaussian-flow: 4d reconstruction with dynamic 3d gaussian particle. In *CVPR*, pages 21136–21145, 2024.

[28] Yang Liu, Chuanchen Luo, Lue Fan, Naiyan Wang, Junran Peng, and Zhaoxiang Zhang. Citygaussian: Real-time high-quality large-scale scene rendering with gaussians. In *ECCV*, pages 265–282. Springer, 2024.

[29] Yang Liu, Chuanchen Luo, Zhongkai Mao, Junran Peng, and Zhaoxiang Zhang. Citygaussianv2: Efficient and geometrically accurate reconstruction for large-scale scenes. *arXiv preprint arXiv:2411.00771*, 2024.

[30] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pages 800–809. IEEE, 2024.

[31] Xiaoyang Lyu, Yang-Tian Sun, Yi-Hua Huang, Xiuzhe Wu, Ziyi Yang, Yilun Chen, Jiangmiao Pang, and Xiaojuan Qi. 3dgsr: Implicit surface reconstruction with 3d gaussian splatting. *ACM Transactions on Graphics (TOG)*, 43(6):1–12, 2024.

[32] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[33] Sadra Safadoust, Fabio Tosi, Fatma Güney, and Matteo Poggi. Self-evolving depth-supervised 3d gaussian splatting from rendered stereo pairs. In *BMVC*, 2024.

[34] Shunsuke Saito, Gabriel Schwartz, Tomas Simon, Junxuan Li, and Giljoo Nam. Relightable gaussian codec avatars. In *CVPR*, pages 130–141, 2024.

[35] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, pages 4104–4113, 2016.

[36] Philipp Schröppel, Jan Bechtold, Artemij Amiranashvili, and Thomas Brox. A benchmark and a baseline for robust multi-view depth estimation. In *2022 International Conference on 3D Vision (3DV)*, pages 637–645. IEEE, 2022.

[37] Edgar Sucar, Shikun Liu, Joseph Ortiz, and Andrew J Davison. imap: Implicit mapping and positioning in real-time. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6229–6238, 2021.

[38] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023.

[39] Fabio Tosi, Luca Bartolomei, and Matteo Poggi. A survey on deep stereo matching in the twenties. *International Journal of Computer Vision (IJCV)*, 2025.

[40] Fabio Tosi, Youmin Zhang, Ziren Gong, Erik Sandström, Stefano Mattoccia, Martin R Oswald, and Matteo Poggi. How nerfs and 3d gaussian splatting are reshaping slam: a survey. *arXiv preprint arXiv:2402.13255*, 4:1, 2024.

[41] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splatter: Depth and normal priors for gaussian splatting and meshing. *arXiv preprint arXiv:2403.17822*, 2024.

[42] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. 2021.

[43] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[44] Bowen Wen, Matthew Trepte, Joseph Aribido, Jan Kautz, Orazio Gallo, and Stan Birchfield. Foundation-stereo: Zero-shot stereo matching. *CVPR*, 2025.

[45] Yaniv Wolf, Amit Bracha, and Ron Kimmel. Gs2mesh: Surface reconstruction from gaussian splatting via novel stereo views. In *ECCV*, pages 207–224. Springer, 2024.

[46] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *CVPR*, pages 20310–20320, 2024.

[47] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19595–19604, 2024.

[48] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *NeurIPS*, 37:21875–21911, 2025.

[49] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023.

[50] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *CVPR*, pages 20331–20341, 2024.

[51] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, pages 767–783, 2018.

[52] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *NeurIPS*, 2021.

[53] Chongjie Ye, Lingteng Qiu, Xiaodong Gu, Qi Zuo, Yushuang Wu, Zilong Dong, Liefeng Bo, Yuliang Xiu, and Xiaoguang Han. Stablenormal: Reducing diffusion variance for stable and sharp normal. *ACM TOG*, 43(6):1–18, 2024.

[54] Mulin Yu, Tao Lu, Linning Xu, Lihan Jiang, Yuanbo Xiangli, and Bo Dai. Gsdf: 3dgs meets sdf for improved rendering and reconstruction. *arXiv preprint arXiv:2403.16964*, 2024.

[55] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM TOG*, 43(6):1–13, 2024.

[56] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

[57] Wenyuan Zhang, Yu-Shen Liu, and Zhizhong Han. Neural signed distance function inference through splatting 3d gaussians pulled on zero-level set. *arXiv preprint arXiv:2410.14189*, 2024.

[58] Haoliang Zhao, Huizhou Zhou, Yongjun Zhang, Jie Chen, Yitong Yang, and Yong Zhao. High-frequency stereo matching network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1327–1336, 2023.

[59] Haoliang Zhao, Huizhou Zhou, Yongjun Zhang, Jie Chen, Yitong Yang, and Yong Zhao. High-frequency stereo matching network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1327–1336, 2023.

[60] Hongyu Zhou, Jiahao Shao, Lu Xu, Dongfeng Bai, Weichao Qiu, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugs: Holistic urban 3d scene understanding via gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21336–21345, 2024.

[61] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3dgs: Supercharging 3d gaussian splatting to enable distilled feature fields. In *CVPR*, pages 21676–21685, 2024.

[62] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. In *CVPR*, pages 21634–21643, 2024.

[63] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12786–12796, 2022.

[64] Yiming Zuo, Willow Yang, Zeyu Ma, and Jia Deng. Omni-dc: Highly robust depth completion with multiresolution depth integration. *arXiv preprint arXiv:2411.19278*, 2024.