STRATGS: LAYERED SURFACE RECOVERY FOR TRANSPARENT OBJECTS FROM GAUSSIAN SPLATTING

Anonymous authors
Paper under double-blind review

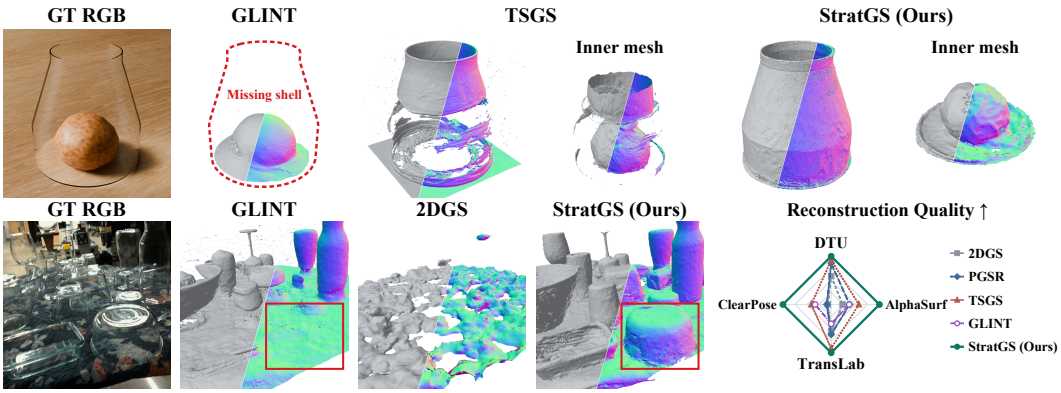

Figure 1: **Reconstructing transparent surfaces and the structures behind them.** StratGS recovers transparent boundaries and visible interior structures in challenging nested and real scenes. Diagonally split views show meshes on the left and normal maps on the right. It achieves the lowest mean geometric error among the compared methods on αSurf, TransLab, and ClearPose. The radar quality score is the best plotted error divided by each method's error, so higher is better.

ABSTRACT

Reconstructing transparent and semi-transparent scenes requires recovering multiple surfaces along a viewing ray. In nested scenes, dominant shell contributions can suppress the learning of interior structures. Reducing their contributions to a single depth can discard weak surface evidence or place geometry between distinct interfaces. We present StratGS, a Gaussian Splatting framework that combines nested-scene optimization with multiple-surface mesh reconstruction. After learning the whole scene, our nested refinement fixes the shell and background Gaussians and caps the shell's rendering contribution while optimizing the interior representation. Range-Aware Depth Extraction (RADE) retains local depth estimates from multiple search ranges for independent multi-view reconstruction. Independent TSDF integration produces fine meshes for all queries and a coarse candidate from the widest query using a broader truncation distance. To control the addition of coarse geometry, Fusion Using Scale-aware Exclusion (FUSE) preserves all fine triangles as a fixed coverage reference and clips the coarse mesh to retain fragments beyond that region. To evaluate coverage of transparent enclosures and the objects inside them, we introduce TransNest, eight synthetic multi-view scenes with separate ground-truth geometry for both regions. On existing benchmarks, the resulting meshes achieve mean Chamfer distances of **0.52** on αSurf and **1.553** mm on TransLab, and a mean one-way $\mathrm{CD}_{\mathrm{s}\to\mathrm{d}}$ of **3.276** mm on ClearPose. These are the lowest mean errors among the compared methods. On TransNest, StratGS achieves the lowest CD on all eight scenes and the lowest mean ground-truth-to-mesh distances for both enclosures and interiors.

1 INTRODUCTION

3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) enables efficient novel-view synthesis with high visual quality. Building on this representation, methods such as 2DGS, GOF, and PGSR (Huang

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

et al., 2024; Yu et al., 2024b; Chen et al., 2025) have advanced explicit surface reconstruction on opaque objects. Extending these advances to scenes with transparent and semi-transparent surfaces remains challenging. These surfaces reveal structures behind them while contributing their own appearance, complicating the correspondence between image observations and physical geometry. Reconstructing such scenes requires recovering the transparent surfaces together with the structures visible through them.

This difficulty arises in many spatial arrangements, including objects viewed through glass, partially overlapping transparent objects, and nested structures. In nested scenes, the outer shell can acquire disproportionately large rendering weights during joint optimization, suppressing the image contribution and training gradients of the interior. Consequently, structures visible through the shell may remain poorly represented even before depth extraction. At reconstruction time, averaging Gaussian contributions across depths can place the estimated surface between physical interfaces; selecting only the dominant contribution can instead discard evidence of other surfaces. Recovering multiple surfaces therefore requires both learning their geometry and retaining their distinct depth evidence.

Existing approaches improve transparent geometry through surface priors or explicit light-transport modeling. TSGS (Li et al., 2025b) learns geometry using normal and de-lighting priors, then fixes opacity while refining appearance with anisotropic specular modeling. Its two-stage training separates geometry learning from appearance refinement, while its first-surface extractor returns one depth per ray. GLINT (Na et al., 2026) reconstructs a primary interface and models reflected and transmitted radiance separately, improving the representation of transparent appearance and geometry. Nested reconstruction further requires improving interior learning while preserving the shell. The missing transparent surfaces and fragmented geometry in Figure 1 illustrate the difficulty of retaining geometry at different depths.

At the reconstruction stage, TSPE-GS (Xu et al., 2026) extracts multiple depth layers and integrates them through progressive TSDF fusion. It freezes voxels supporting the outer reconstruction before integrating subsequent layers. This protects existing surfaces during fusion, but does not change how the interior is learned. Depth extraction depends on the learned representation containing sufficient evidence for the surfaces to reconstruct. Addressing weak interior representations therefore requires intervention during training, before depth extraction and fusion.

These challenges call for learning the interior without sacrificing the shell and preserving both during reconstruction. We introduce **StratGS**, a framework with three complementary components (Figure 2): (a) **Nested-scene optimization** freezes the shell and background Gaussians and limits shell contributions while refining the interior in a second stage. (b) **Range-Aware Depth Extraction (RADE)** retains depth candidates from multiple search ranges so that weaker surface evidence can contribute to reconstruction. (c) **Coverage-constrained fusion (FUSE)** preserves fine reconstructions and selectively supplements uncovered regions with coarse geometry, improving coverage while retaining surface detail.

Evaluating nested reconstruction requires measuring whether both the transparent enclosure and its contents are recovered. We therefore construct TransNest, eight synthetic laboratory and robotic scenes with calibrated multi-view images and separate ground-truth meshes for enclosure and interior surfaces. These annotations allow coverage of each region to be evaluated against the reconstructed mesh. Across αSurf (Wu et al., 2025b), TransLab (Li et al., 2025b), and ClearPose (Chen et al., 2022b), StratGS achieves the lowest mean geometric reconstruction error among the compared methods. Relative to the strongest baseline on each benchmark, it reduces mean CD by approximately **15.8%** on αSurf and **9.2%** on TransLab, and mean one-way $\mathrm{CD}_{\mathrm{s}\to\mathrm{d}}$ by **58.8%** on ClearPose. On TransNest, StratGS achieves the lowest CD on all eight scenes and the lowest mean ground-truth-to-mesh distances for both enclosures and interiors, showing improved recovery of both parts of the nested geometry.

Our contributions are:

1. A two-stage nested-scene optimization strategy that freezes shell and background Gaussians, limits shell contributions, and refines only the interior representation.

2. Multiple-surface mesh reconstruction through range-aware depth extraction and coverage-constrained fusion: independent depth queries produce fine meshes, whose fixed triangle surfaces guide the selection and clipping of coarse geometry for supplementation.

3. TransNest, eight synthetic laboratory and robotic scenes with calibrated multi-view images and separate ground-truth meshes for measuring enclosure and interior coverage in nested reconstruction.

2 Related Work

Neural Surface Reconstruction and Fusion. NeRF Mildenhall et al. (2020) and its anti-aliased variants such as Mip-NeRF Barron et al. (2021), Mip-NeRF 360 Barron et al. (2022), and Zip-NeRF Barron et al. (2023) optimize radiance fields from posed images. Hash grids and tensor factorizations accelerate this family Müller et al. (2022); Chen et al. (2022a), and reflectance-aware radiance fields improve glossy appearance Verbin et al. (2022). VolSDF Yariv et al. (2021), NeuS Wang et al. (2021), and Neuralangelo Li et al. (2023) adapt differentiable volume rendering to surface extraction. For transparent objects, differentiable refraction tracing Lyu et al. (2020), implicit refraction rendering Gao et al. (2023), NU-NeRF Sun et al. (2024), α-NeuS Zhang et al. (2024b), αSurf Wu et al. (2025b), NEMTO Wang et al. (2023), and differentiable neural surface refinement Deng et al. (2024) model refraction, opacity, or transparent light transport with neural implicit fields. These methods provide important transparent-surface priors, but they rely on dense volumetric sampling or explicit refractive modeling rather than the sorted Gaussian depth–weight evidence available in a splatting renderer. For depth-based surface construction, TSDF fusion Curless & Levoy (1996); Newcombe et al. (2011) and multi-resolution reconstruction Steinbrücker et al. (2013) aggregate evidence across views. We integrate range-restricted depths independently with a narrow TSDF truncation distance and use the resulting fine meshes as a fixed coverage reference for supplementation from a candidate reconstructed with a broader truncation distance.

Surface Reconstruction from 3D Gaussian Splatting. 3D Gaussian Splatting Kerbl et al. (2023) represents scenes as collections of anisotropic Gaussians optimized via differentiable rasterization. While originally designed for novel-view synthesis, several works have adapted it for surface reconstruction. SuGaR Guédon & Lepetit (2024) regularizes Gaussians to lie on surfaces and extracts meshes via Poisson reconstruction. 2D Gaussian Splatting Huang et al. (2024) replaces 3D Gaussians with oriented 2D disks, enabling direct depth and normal rendering. GOF Yu et al. (2024b) introduces Gaussian opacity fields for level-set extraction. PGSR Chen et al. (2025) adds multi-view geometric consistency losses. RaDe-GS Zhang et al. (2024a) proposes rasterized depth-normal rendering with geometric regularization. 3DGSR Lyu et al. (2024) couples Gaussian optimization with an implicit SDF to improve mesh extraction. GaussianPro Cheng et al. (2024) improves Gaussian densification via progressive propagation, while Spec-Gaussian Yang et al. (2024), GaussianShader Jiang et al. (2024), 3DGS-DR Ye et al. (2024), and Textured Gaussians Chao et al. (2025) increase view-dependent or spatially varying appearance capacity. GGGS Zhang et al. (2026) treats Gaussian primitives as stochastic solids for geometry-grounded depth rendering. Other recent variants improve aliasing, depth/normal regularization, geometry-field splatting, volumetric transmittance, order-independent transparency, and secondary-ray rendering Yu et al. (2024a); Turkulainen et al. (2025); Jiang et al. (2025); Talegaonkar et al. (2025); Müller et al. (2025); Moenne-Loccoz et al. (2024); Wu et al. (2025a); Sun et al. (2025). Transparent or reflective GS applications have also begun to appear in object-specific settings Li et al. (2025a); Kim et al. (2025b); Huang et al. (2025); Zeng et al. (2026). Layered representations address complementary settings: Multi-Layer GS Kleinbeck et al. (2025) builds anatomical layers, LayerGS Xu & Dingliana (2027) separates body and garment layers, and Free360 Bao et al. (2025) uses spatial layers for sparse, unposed 360-degree view synthesis. Their layers follow anatomy, human templates, or scene partitions. Our query indices instead refer to search ranges within the rendering contributions of an optimized Gaussian representation.

Transparent and Semi-Transparent Object Reconstruction. Refraction and transmission complicate transparent-surface reconstruction. Classical methods rely on specialized capture or physical priors Miyazaki et al. (2004); Ihrke et al. (2010); Wu et al. (2018), while learning-based work studies transparent-object perception and depth or normal estimation Sajjan et al. (2020); Fang et al. (2022); Chen et al. (2022b); Li et al. (2026a;b). Among Gaussian methods, TSGS Li et al. (2025b) uses transparent-aware training and window-based first-surface extraction, whereas TSPE-GS Xu et al. (2026) retains multiple depth layers for progressive TSDF fusion. Geometry Gaussians and TRAN-D further target geometry recovery through Gaussian representations Zhou & Lähner (2026); Kim et al. (2025a). Other approaches model reflection and refraction for transparent-scene render-

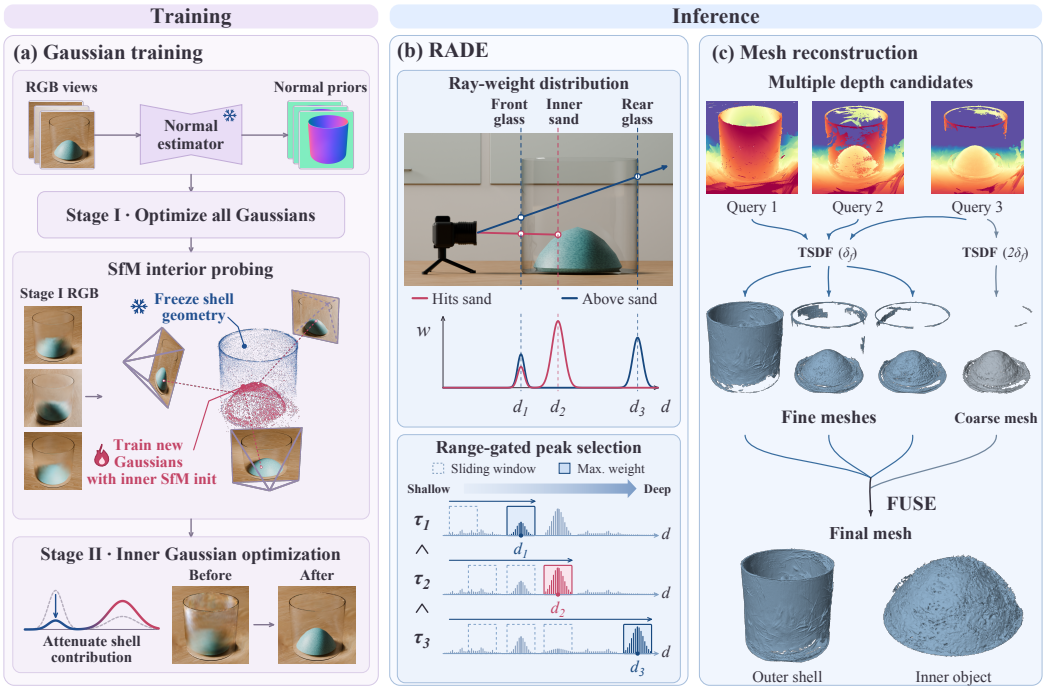

Figure 2: **Method overview.** (a) Multi-view images and normal priors guide Gaussian optimization. For nested scenes, Stage II freezes the shell and background Gaussians, caps shell contributions, and optimizes only SfM-initialized interior Gaussians with full-image L1+SSIM. (b) RADE uses three thresholds $\tau_1 < \tau_2 < \tau_3$ to query progressively broader ray ranges and aggregate local rendering contributions into depth candidates. (c) Each query's multi-view depth maps are independently integrated into a fine mesh (blue). The widest query additionally yields a coarse mesh (gray). FUSE retains the fine geometry and adds uncovered coarse regions.

ing Huang et al. (2025); Zeng et al. (2026); Yoon et al. (2026); Na et al. (2026). StratGS focuses on surface extraction from multiple ray ranges. It retains their fine triangles as a fixed reference and clips a coarse candidate to select additions outside the reference coverage radius.

3 Method

Overview. Given calibrated multi-view images, StratGS first learns a Gaussian representation of the scene. For nested scenes, a second stage fixes the shell and background Gaussians and caps the shell's rendering contribution while optimizing the interior representation (Figure 2(a)). From the optimized representation, RADE preserves multiple depth candidates along each ray instead of reducing its evidence to a single depth. Their depth maps are integrated independently into fine surface candidates, while broader integration support provides a coarse candidate for additional coverage. FUSE combines the fine geometry with selected coarse regions to form the final mesh (Figure 2).

3.1 Gaussian Representation and Optimization

Representation and image formation. We represent the scene with 3D Gaussians (Kerbl et al., 2023), each parameterized by a center $\boldsymbol{\mu}_i$, covariance $\boldsymbol{\Sigma}_i = \mathbf{R}_i \,\mathrm{diag}(\mathbf{s}_i^2)\mathbf{R}_i^\top$, opacity, and spherical-harmonic coefficients for view-dependent color c_i. Here $\mathbf{R}_i$ and $\mathbf{s}_i$ describe its orientation and scales. For a pixel, the projected Gaussians are composited front to back:

$$\hat{C} = \sum_i w_i c_i, \qquad w_i = \alpha_i T_i, \qquad T_i = \prod_{j<i}(1-\alpha_j), \tag{1}$$

where α_i is the Gaussian's effective opacity at the pixel.

Normal and depth rendering. Following TSGS (Li et al., 2025b), we flatten each Gaussian along its smallest-scale axis, which defines the normal of a local surface plane through its center. Let $\mathbf{n}_i$ be its camera-facing normal and b_i its unsigned plane distance from the camera. For pixel $p = (u, v)$ with ray $\mathbf{q}(p) = \mathbf{K}^{-1}(u, v, 1)^\top$, we render its normal and depth using the same compositing weights:

$$\mathbf{n}_r(p) = \sum_i w_i(p)\mathbf{n}_i, \qquad D(p) = \frac{\sum_i w_i(p) b_i}{-\mathbf{n}_r(p)^\top \mathbf{q}(p)}. \tag{2}$$

Here $\mathbf{K}$ denotes the camera intrinsics.

Training objective. We adopt the training objective of TSGS (Li et al., 2025b) to optimize the Gaussian parameters through image fitting and geometric regularization:

$$\mathcal{L} = \mathcal{L}_{\mathrm{RGB}} + \lambda_p \mathcal{L}_{\mathrm{prior}} + \lambda_d \mathcal{L}_{\mathrm{dn}} + \lambda_f \mathcal{L}_{\mathrm{flat}} + \lambda_g \mathcal{L}_{\mathrm{geo}} + \lambda_c \mathcal{L}_{\mathrm{NCC}}. \tag{3}$$

The image loss $\mathcal{L}_{\mathrm{RGB}}$ combines L1 error and SSIM. The normal-prior loss $\mathcal{L}_{\mathrm{prior}}$ aligns the rendered normal map $\mathbf{n}_r$ with normal priors predicted by TransNormal-2 (Li et al., 2026b), while $\mathcal{L}_{\mathrm{dn}}$ encourages agreement between $\mathbf{n}_r$ and normals derived from the rendered depth map D. The flattening term $\mathcal{L}_{\mathrm{flat}}$ penalizes the smallest Gaussian scale to favor local surface geometry. Across views, $\mathcal{L}_{\mathrm{geo}}$ penalizes round-trip reprojection error and $\mathcal{L}_{\mathrm{NCC}}$ compares image patches warped by the local planes. Together, these terms fit appearance and constrain surface geometry; weights and supervision details are given in the supplementary material.

Nested-scene refinement. We observe that, during joint optimization, shell Gaussians can accumulate disproportionately large rendering weights. This suppresses the interior's contribution to the image and weakens the gradients needed to learn its appearance. Since the shell is semi-transparent, both the shell and interior should contribute to the observed image. We therefore use two-stage refinement to learn the interior while controlling the shell's contribution. The first stage optimizes a Gaussian representation of the entire scene using the objective in Eq. (3). Starting from this representation, we use the learned geometry together with the SfM point cloud to distinguish the shell, interior, and background regions for second-stage refinement. We retain the learned Gaussians for the shell $\mathcal{G}_s$ and background $\mathcal{G}_b$, and initialize the interior Gaussians $\mathcal{G}_{\mathrm{in}}$ from SfM points in the interior region.

In the second stage, we attenuate $\mathcal{G}_s$ by capping its total rendering contribution along each ray:

$$\widetilde{w}_i^s = w_i^s \min\big(1, \beta / \textstyle\sum_j w_j^s\big), \tag{4}$$

where w_i^s is the compositing weight computed from $\mathcal{G}_s$ alone and β bounds its total contribution. We freeze the geometry, opacity, and spherical-harmonic colors of $\mathcal{G}_s$ and $\mathcal{G}_b$ and optimize only $\mathcal{G}_{\mathrm{in}}$ with the L1+SSIM RGB term in Eq. (3), computed over the full image jointly rendered from all three groups. The following reconstruction stages operate on the fixed, optimized representation.

3.2 Range-Aware Depth Extraction

Motivation. In a transparent scene, a ray may receive contributions from several surfaces at different depths. Collapsing these contributions into one estimate can mix separated surfaces, while selecting only the strongest local contribution can miss a weaker surface.

Range-aware queries. For each ray, let d_i denote its intersection depth with Gaussian i's local plane and w_i the corresponding rendering weight. We sort the depths as $d_1 \le \cdots \le d_N$, retaining their original rendering weights. Each query uses a threshold τ on accumulated rendering weight. The search starts at the first contributing Gaussian and ends when the accumulated weight first exceeds τ. A smaller τ restricts the search to shallower evidence, while a larger τ admits deeper contributions.

Window aggregation. Within each query range, a compact window identifies concentrated surface evidence. We select the window with the largest total weight and estimate its depth by weighted averaging:

$$(l^*, r^*) = \underset{\substack{1 \le l \le r \le N \\ \sum_{j<r} w_j \le \tau \\ d_r - d_l \le h}}{\arg\max} \sum_{i=l}^{r} w_i, \qquad \hat{d}_\tau = \frac{\sum_{i=l^*}^{r^*} w_i d_i}{\sum_{i=l^*}^{r^*} w_i}. \tag{5}$$

Here h bounds the depth span of the aggregation window. Using several search ranges allows weaker contributions to produce a candidate without competing with all contributions along the ray.

Multiple depth candidates. For the configuration illustrated in Figure 2(b), we empirically select three thresholds $\tau_1 < \tau_2 < \tau_3$, yielding three depth candidates from progressively broader search ranges. Each threshold produces a separate depth map for every view, preserving alternatives for subsequent reconstruction.

3.3 Mesh Reconstruction and Fusion

Motivation. We reconstruct surfaces by independently integrating each query's depth maps across calibrated views (Figure 2(c)). A narrow TSDF truncation distance preserves local detail but can leave gaps where observations are sparse or inconsistent. A broader support can connect observations across these gaps, although it may also smooth or thicken existing surfaces. We retain both forms of evidence so that broader coverage can supplement the fine reconstructions.

Fine and coarse meshes. For each threshold $\tau \in \mathcal{T}$, we integrate its depth maps from all calibrated views into an independent TSDF volume with truncation distance δ_f and extract a fine mesh M_τ^f using Marching Cubes (Curless & Levoy, 1996; Lorensen & Cline, 1987). The depth maps from the widest search range additionally supply a coarse mesh M^c, reconstructed with a broader truncation distance $\delta_c > \delta_f$. Both use the same voxel resolution; fine and coarse refer to integration support. Independent volumes prevent observations from different search ranges from being averaged into a single surface estimate.

Coverage-constrained fusion. The fine candidates preserve detailed surfaces, while the coarse candidate can cover regions missing from them. Adding the entire coarse mesh can duplicate or thicken surfaces already represented by the fine candidates. FUSE instead retains all fine meshes as a fixed reference M_f and uses the coarse candidate to supplement uncovered regions. Exclusion applies only to the coarse candidate, preserving nearby fine surfaces that may correspond to distinct interfaces. For a point $\mathbf{x}$ on the coarse surface, its unsigned distance to the fine reference determines whether that region is already covered:

$$d_f(\mathbf{x}) = \min_{\mathbf{y} \in |M_f|} \|\mathbf{x} - \mathbf{y}\|_2, \qquad S_c = \{\mathbf{x} \in |M^c| : d_f(\mathbf{x}) > r\}. \tag{6}$$

Here $|M|$ denotes the surface of a mesh and r is a coverage radius scaled to the reconstruction resolution. The region S_c contains the coarse geometry beyond the fine reference's coverage. The reference remains fixed during selection, so newly accepted coarse regions do not change which other regions are retained.

Final mesh. We refine and clip coarse triangles that cross the coverage boundary, forming a supplementary mesh M_{add} that approximates S_c. The final reconstruction combines this mesh with the fine reference:

$$M_{\text{out}} = M_f \oplus M_{\text{add}}, \tag{7}$$

where $\oplus$ denotes mesh concatenation. This produces the final mesh shown in Figure 2(c), preserving the fine geometry while adding selected coarse fragments. More implementation details are provided in the supplementary material.

4 Experiments

4.1 Experimental Setup

Datasets and baselines. We evaluate the complete mesh pipeline on eight αSurf scenes (Wu et al., 2025b), eight TransLab scenes (Li et al., 2025b), and four ClearPose scene sets (Chen et al., 2022b). To assess enclosure and interior recovery in nested scenes, we construct TransNest, eight synthetic laboratory and robotic scenes with calibrated views and separate ground-truth meshes for the two regions. Each scene provides 375 training and 57 test views; scene construction and annotations are described in the supplementary material. We also evaluate opaque-object reconstruction on fifteen DTU scans (Jensen et al., 2014). We compare with Plenoxels (Fridovich-Keil et al., 2022), Mip-NeRF 360 (Barron et al., 2022), NeuS (Wang et al., 2021), HF-NeuS (Wang et al., 2022),

Neuralangelo (Li et al., 2023), NeRRF (Chen et al., 2023), αSurf (Wu et al., 2025b), and NU-NeRF (Sun et al., 2024). Our Gaussian-based baselines include SuGaR (Guédon & Lepetit, 2024), GOF (Yu et al., 2024b), 2DGS (Huang et al., 2024), PGSR (Chen et al., 2025), RaDe-GS (Zhang et al., 2024a), GGGS (Zhang et al., 2026), TSGS (Li et al., 2025b), 2DGS+TSPE (Huang et al., 2024; Xu et al., 2026), TSPE-GS (Xu et al., 2026), and GLINT (Na et al., 2026), which models secondary light transport with ray tracing. We follow the evaluation protocols of TSGS (Li et al., 2025b). Additional evaluation and baseline implementation details are provided in the supplementary material.

4.2 Quantitative Results

Table 1: **Geometry and rendering on transparent-scene benchmarks.**

(a) Geometry results on αSurf.
Per-scene and mean CD↓ over eight scenes. Non-Gaussian baselines are listed first, followed by Gaussian-based methods. Across all three benchmarks, shading marks the best, second-best, and third-best results in each column.

Method	case	bot.	cof.	din.	dbl.	mnk.	leaf	kit.	Mean
Plen (σ=30)	1.20	0.71	0.71	0.44	0.56	1.08	1.02	0.38	0.76
Mip360 (σ=50)	4.01	2.32	2.10	5.22	2.90	4.39	1.19	2.38	3.06
NeuS	5.09	2.27	1.07	1.19	0.87	5.92	1.95	0.39	2.34
HFS	5.09	1.49	2.84	0.85	—	3.22	8.68	—	3.70
N-angelo	2.07	0.40	0.80	0.48	0.51	2.46	1.70	0.58	1.13
NeRRF	1.27	3.72	0.82	0.57	0.83	2.91	3.49	3.09	2.09
αSurf	0.84	0.65	0.72	0.37	0.51	0.78	0.87	0.26	0.62
RaDe-GS	0.49	0.80	0.98	0.57	0.63	3.00	0.85	0.44	0.97
2DGS	7.82	2.44	1.16	0.81	0.76	2.43	0.93	1.34	2.21
PGSR	1.60	2.97	0.65	0.54	0.51	3.39	0.75	0.56	1.37
2DGS+TSPE	0.72	1.45	0.99	1.45	1.12	1.47	0.72	0.70	1.08
TSPE-GS	0.48	0.76	1.04	0.51	0.56	1.66	0.76	0.51	0.82
TSGS	0.87	0.73	0.71	0.53	0.78	2.23	0.88	0.59	0.91
GGGS	0.87	5.58	0.78	0.60	1.25	4.58	0.86	0.77	1.91
GLINT	1.54	1.36	0.85	0.50	0.62	4.03	1.77	0.53	1.40
StratGS (Ours)	**0.34**	**0.52**	**0.49**	**0.44**	**0.37**	**0.84**	**0.71**	**0.47**	**0.52**

(b) ClearPose. Geometry and rendering results.

Method	Geometry: $CD_{s\to d}$ (mm, ↓)					Rendering		
	Set 1	Set 2	Set 3	Set 4	**Mean**	PSNR↑	SSIM↑	LPIPS↓
2DGS	30.24	50.76	47.82	62.13	47.74	16.96	0.475	0.503
PGSR	19.27	59.18	49.99	15.66	36.03	15.69	0.400	0.525
NU-NeRF	27.90	12.00	10.60	17.40	17.00	17.78	0.599	0.458
TSGS	4.40	7.54	11.14	8.78	7.96	18.22	0.608	0.391
GLINT	6.52	11.38	13.22	8.34	9.87	17.33	0.572	0.415
GGGS	39.06	60.96	59.06	87.33	61.60	17.07	0.424	0.495
StratGS (Ours)	**3.43**	**3.58**	**3.36**	**2.74**	**3.28**	**18.33**	**0.620**	**0.382**

(c) TransLab. CD↓ (mm) over eight scenes.

Method	01	02	03	04	05	06	07	08	Mean
SuGaR	37.20	10.30	8.00	6.50	8.60	46.20	14.00	7.40	17.30
2DGS	1.80	3.52	2.80	2.80	2.75	4.42	1.62	3.06	2.85
GOF	4.79	10.40	2.72	3.14	10.00	18.40	9.97	4.88	8.04
PGSR	1.89	3.65	2.22	2.20	2.71	1.67	2.15	3.52	2.50
TSPE-GS†	3.68	32.10	4.88	5.68	5.55	16.50	5.40	4.70	9.81
GGGS	2.04	45.00	2.00	2.23	34.80	11.70	7.40	4.96	13.76
GLINT	3.87	3.68	2.53	2.46	3.40	5.89	5.53	3.82	3.90
TSGS	1.59	2.13	1.50	1.59	1.65	1.36	1.70	2.15	1.71
StratGS (Ours)	**1.38**	**2.17**	**1.37**	**1.41**	**1.37**	**1.35**	**1.56**	**1.81**	**1.55**

Transparent-object geometry. StratGS achieves the lowest mean error in Table 1. On αSurf, mean CD is **0.52**, compared with 0.91 for TSGS and 0.82 for TSPE-GS. On TransLab, mean CD is **1.553** mm, compared with 1.71 mm for TSGS. On ClearPose, mean $CD_{s\to d}$ is **3.276** mm, compared with 7.96 mm for TSGS, with lower errors on all four scene sets. The rule is unchanged across these three benchmarks. The threshold sets and TSDF truncation distances are given in the supplementary material.

GLINT yields mean CD 1.398 on αSurf and 3.896 mm on TransLab, and one-way $CD_{s\to d}$ 9.87 mm on ClearPose.

Nested-scene reconstruction. On TransNest, StratGS achieves the lowest CD on all eight scenes (Table 2), with a mean of **3.970** mm compared with 8.499 mm for TSGS, 17.106 mm for TSPE-GS, and 25.992 mm for SuGaR. Its mean F1 at 5 mm reaches **0.834**, compared with 0.611 for TSGS. The outer and inner GT-to-mesh distances are 2.623 mm and 4.580 mm, respectively, the lowest among the compared methods, indicating better coverage of both the transparent enclosures and their contents. The TransNest training configuration is given in the supplementary material, with training ablations reported below.

Opaque-object reconstruction. On DTU, our opaque-scene configuration achieves mean CD 0.47, compared with 0.51 for TSGS and 0.55 for PGSR (Table 3).

Rendering quality. The ClearPose Gaussian-rendering comparison in Table 1 reports PSNR 18.33, SSIM 0.620, and LPIPS 0.382. Total training and mesh extraction time is reported in the supplementary material.

Table 2: **Geometry on TransNest.** Columns 01–08 report per-scene CD; the remaining metrics are averaged over the eight scenes. CD and regional distances are in millimeters. Outer and Inner measure distances from the respective ground-truth surfaces to the reconstructed mesh.

Method	CD (mm)↓									F1@5mm↑	GT-to-mesh↓	
	01	02	03	04	05	06	07	08	**Mean**		Outer	Inner
SuGaR	33.108	28.990	31.231	13.637	18.931	37.561	27.806	16.677	25.992	0.347	56.090	6.276
2DGS	20.830	17.469	19.384	49.402	14.602	49.171	16.015	12.378	24.906	0.302	46.063	29.918
GOF	20.653	17.070	21.038	9.825	11.794	36.313	16.504	12.490	18.211	0.371	38.237	6.944
PGSR	9.545	14.962	8.261	9.171	8.253	18.320	10.624	13.235	11.547	0.373	22.783	7.232
TSPE-GS	21.747	21.865	20.041	8.485	12.716	10.207	21.851	19.934	17.106	0.334	33.763	7.759
TSGS	9.175	10.399	7.726	7.050	7.339	8.564	9.922	7.818	8.499	0.611	7.686	12.317
GLINT	14.133	17.160	30.730	10.317	8.882	32.497	12.791	12.936	17.431	0.446	24.766	28.207
StratGS (Ours)	**6.155**	**4.580**	**3.554**	**2.834**	**3.271**	**3.228**	**5.040**	**3.097**	**3.970**	**0.834**	**2.623**	**4.580**

Table 3: **Chamfer Distance (↓) on DTU (2014) (15 scans).** Columns report per-scan CD and the mean; lower is better. TSPE-GS† is reproduced using open-source code.

Method		24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	**Mean**
SuGaR	(2024)	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	1.32
2DGS	(2024)	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
GOF	(2024b)	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
PGSR	(2025)	0.35	0.55	0.40	0.33	0.82	0.55	0.48	1.10	0.70	0.60	0.52	0.68	0.30	0.45	0.37	0.55
TSPE-GS	(2026)	0.44	0.74	0.35	0.41	0.81	0.74	0.69	1.22	1.19	0.63	0.60	0.88	0.41	0.66	0.47	0.68
TSPE-GS†	(2026)	0.45	0.73	0.39	0.39	0.87	0.71	0.70	1.13	0.93	0.67	1.68	0.93	0.38	0.68	0.47	0.74
TSGS	(2025b)	0.34	0.53	0.37	0.41	0.79	0.55	0.48	1.05	0.62	0.59	0.42	0.49	0.31	0.37	0.35	0.51
StratGS (Ours)		**0.38**	**0.47**	**0.34**	**0.34**	**0.65**	**0.56**	**0.42**	**0.94**	**0.55**	**0.52**	**0.44**	**0.40**	**0.29**	**0.36**	**0.35**	**0.47**

Ablation Studies. We evaluate training and reconstruction components on TransLab and nested-scene refinement on TransNest, averaging over eight scenes in each dataset.

Table 4: **TransLab ablations** report mean mesh CD (mm) over eight scenes.

Configuration	CD (mm)↓
w/o normal prior	1.64
w/o depth–normal consistency	1.67
w/o RADE	1.60
w/o FUSE	1.56
Full method (Ours)	**1.55**

Table 5: **TransNest refinement variants.** Eight-scene means with the same four-candidate mesh extraction as Table 2. Distances are in millimeters.

Setting	CD↓	F1@5mm↑	GT-to-mesh (mm)↓		
			All	Outer	Inner
w/o shell attenuation	4.150	0.8341	3.403	2.629	7.349
w/o nested refinement	6.545	0.7911	8.062	4.684	26.122
Full refinement (Ours)	**3.970**	**0.8344**	**2.924**	**2.623**	**4.580**

Training constraints. We separately remove the normal-prior loss and the depth–normal consistency loss, retraining each variant from the same initialization and applying the full reconstruction pipeline. Both removals increase CD, supporting these constraints on TransLab (Table 4).

Depth candidates and fusion. Using the full-objective models from the training ablation, we replace three query thresholds with one (w/o RADE) while retaining both TSDF scales and FUSE. We then replace FUSE with direct concatenation of the same fine and coarse candidates. Both changes increase mean CD (Table 4), with a larger increase from using a single threshold.

Nested-scene refinement. Table 5 compares nested-scene refinement with variants without shell attenuation or nested refinement, using the same four-candidate mesh extraction. Full refinement achieves a mean CD of 3.970 mm and an inner GT-to-mesh distance of 4.580 mm, matching Table 2. All variants use the full-image L1+SSIM RGB loss described in the supplementary material. Optimizing the Stage-I Gaussians without nested refinement yields 6.545 mm and 26.122 mm, respectively. The largest difference is in interior coverage when the refinement stage is omitted.

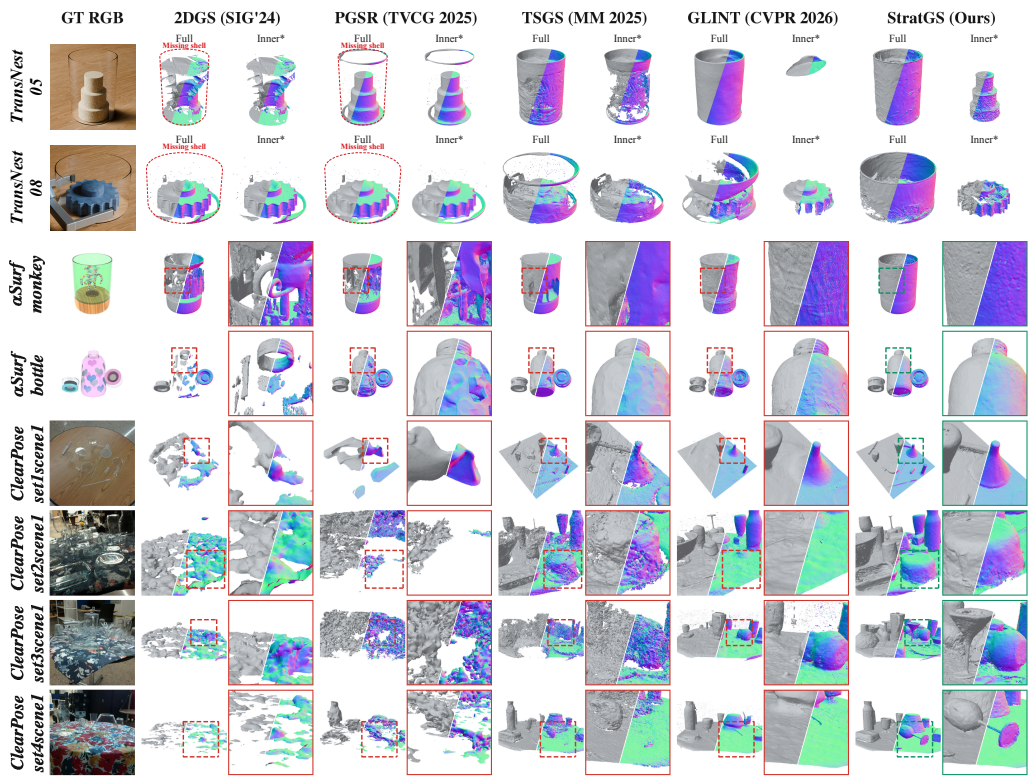

Figure 3: **Qualitative reconstruction comparisons across eight scenes.** The top two rows show TransNest 05 and 08 as Full/Inner pairs; TSGS and StratGS use TransNormal-2 priors. Red dashed GT shell outlines highlight missing sidewalls in 2DGS and PGSR. The remaining six rows show Full/Zoom comparisons on αSurf and ClearPose. 🔍 *Please zoom in for details.*

4.3 Qualitative Results

Qualitative comparison. Figure 3 compares reconstructed surfaces across eight scenes from TransNest, αSurf, and ClearPose. The first two rows show full surfaces and additional inner-surface displays; the remaining rows use matched views and close-ups to reveal missing surfaces and incomplete objects that are less apparent in aggregate metrics.

5 Conclusion

StratGS reconstructs transparent scenes by retaining depth alternatives until mesh reconstruction and selecting supplementary geometry according to spatial coverage. RADE queries multiple search ranges and integrates each query into an independent fine mesh. FUSE preserves these meshes as a fixed reference and clips a coarse candidate, obtained with a broader TSDF truncation distance, to supplement regions beyond their coverage radius. To evaluate the recovery of nested geometry, we introduce TransNest: eight synthetic laboratory and robotic scenes with transparent enclosures containing opaque objects. Calibrated multi-view images and separate ground-truth meshes enable evaluation of enclosure and interior coverage within the reconstructed mesh. On αSurf, TransLab, and ClearPose, StratGS achieves the lowest mean errors among the compared methods: **0.52**, **1.553** mm, and **3.276** mm under their respective metrics. On TransNest, it achieves the lowest CD on all eight scenes, with a mean of **3.970** mm, and the lowest mean distances from both enclosure and interior ground-truth surfaces to the reconstructed mesh.

AI USE STATEMENT

We used generative AI tools, including GPT-6-astra, to assist with method and experiment design, coding, result analysis, and manuscript preparation. GPT-6-astra also assisted with Blender scripting for the manually designed TransNest scenes. The authors guided the work and take responsibility for the final content.

ETHICS STATEMENT

This work studies object reconstruction using existing benchmarks and synthetic scenes, without new human-subject studies or collection of personal data. TransNest depicts laboratory apparatus and static robotic setups created in Blender. Use and redistribution of existing benchmark data remain subject to their original licenses. Reconstructed transparent boundaries can be incomplete, so downstream use in robotic manipulation requires validation under the intended operating conditions.

REPRODUCIBILITY STATEMENT

The reconstruction pipeline is specified in Section 3. The supplementary material provides the training schedules, geometry conventions, and dataset-specific extraction and fusion settings. The construction of TransNest, its camera sampling and train–test split, and its annotations are described in the supplementary material. The supplementary material documents benchmark-specific evaluation, sampling, and aggregation protocols.

REFERENCES

Chong Bao, Xiyu Zhang, Zehao Yu, Jiale Shi, Guofeng Zhang, Songyou Peng, and Zhaopeng Cui. Free360: Layered gaussian splatting for unbounded 360-degree view synthesis from extremely sparse and unposed views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16377–16387, 2025.

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 *ICCV*, 2021.

Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-NeRF 360: Unbounded anti-aliased neural radiance fields. In *CVPR*, 2022.

Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Zip-NeRF: Anti-aliased grid-based neural radiance fields. In *ICCV*, 2023.

Brian Chao, Hung-Yu Tseng, Lorenzo Porzi, Chen Gao, Tuotuo Li, Qinbo Li, Ayush Saraf, Jia-Bin Huang, Johannes Kopf, Gordon Wetzstein, and Changil Kim. Textured gaussians for enhanced 3D scene appearance modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8964–8974, 2025.

Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. TensoRF: Tensorial radiance fields. In *ECCV*, 2022a.

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*, 31(9):6100–6111, 2025. doi: 10.1109/TVCG.2024.3494046.

Xiaotong Chen, Huijie Zhang, Zeren Yu, Anthony Opipari, and Odest Chadwicke Jenkins. ClearPose: Large-scale transparent object dataset and benchmark. In *ECCV*, 2022b.

Xiaoxue Chen, Junchen Liu, Hao Zhao, Guyue Zhou, and Ya-Qin Zhang. NeRRF: 3D reconstruction and view synthesis for transparent and specular objects with neural refractive-reflective fields. arXiv preprint arXiv:2309.13039, 2023.

Kai Cheng, Xiaoxiao Long, Kaizhi Yang, Yao Yao, Wei Yin, Yuexin Ma, Wenping Wang, and Xuejin Chen. GaussianPro: 3D gaussian splatting with progressive propagation. In *ICML*, 2024.

Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH*, 1996.

Weijian Deng, Dylan Campbell, Chunyi Sun, Shubham Kanitkar, Matthew E. Shaffer, and Stephen Gould. Differentiable neural surface refinement for modeling transparent objects. In *CVPR*, 2024.

Hongjie Fang, Hao-Shu Fang, Sheng Xu, and Cewu Lu. TransCG: A large-scale real-world dataset for transparent object depth completion and a grasping baseline. *IEEE Robotics and Automation Letters*, 2022.

Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *CVPR*, 2022.

Fangzhou Gao, Lianghao Zhang, Li Wang, Jiamin Cheng, and Jiawan Zhang. Transparent object reconstruction via implicit differentiable refraction rendering. In *SIGGRAPH Asia*, 2023. doi: 10.1145/3610548.3618236.

Antoine Guédon and Vincent Lepetit. SuGaR: Surface-aligned gaussian splatting for efficient 3D mesh reconstruction and high-quality mesh rendering. In *CVPR*, 2024.

Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2D gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH*, 2024.

Letian Huang, Dongwei Ye, Jialin Dan, Chengzhi Tao, Huiwen Liu, Kun Zhou, Bo Ren, Yuanqi Li, Yanwen Guo, and Jie Guo. TransparentGS: Fast inverse rendering of transparent objects with gaussians. *ACM Transactions on Graphics*, 44(4):1–17, 2025. doi: 10.1145/3730892.

Ivo Ihrke, Kiriakos N. Kutulakos, Hendrik P. A. Lensch, Marcus Magnor, and Wolfgang Heidrich. Transparent and specular object reconstruction. *Computer Graphics Forum*, 29(8):2400–2426, 2010.

Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, 2014.

Kaiwen Jiang, Venkataram Sivaram, Cheng Peng, and Ravi Ramamoorthi. Geometry field splatting with gaussian surfels. In *CVPR*, 2025.

Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. GaussianShader: 3D gaussian splatting with shading functions for reflective surfaces. In *CVPR*, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):139:1–139:14, 2023. doi: 10.1145/3592433.

Jeongyun Kim, Seunghoon Jeong, Giseop Kim, Myung-Hwan Jeon, Eunji Jun, and Ayoung Kim. 2D gaussian splatting-based sparse-view transparent object depth reconstruction via physics simulation for scene update. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27927–27936, 2025a.

Jeongyun Kim, Jeongho Noh, Dong-Guw Lee, and Ayoung Kim. TranSplat: Surface embedding-guided 3D gaussian splatting for transparent object manipulation. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3190–3196. IEEE, 2025b. doi: 10.1109/ICRA55743.2025.11127360.

Constantin Kleinbeck, Hannah Schieber, Klaus Engel, Ralf Gutjahr, and Daniel Roth. Multi-layer gaussian splatting for immersive anatomy visualization. *IEEE Transactions on Visualization and Computer Graphics*, 31(5):2353–2363, 2025. doi: 10.1109/TVCG.2025.3549882.

Congcong Li, Jin Wang, Xiaomeng Wang, Xingchen Zhou, Wei Wu, Yuzhi Zhang, and Tongyi Cao. Car-GS: Addressing reflective and transparent surface challenges in 3D car reconstruction. arXiv preprint arXiv:2501.11020, 2025a.

Mingwei Li, Pu Pang, Hehe Fan, Hua Huang, and Yi Yang. TSGS: Improving gaussian splatting for transparent surface reconstruction via normal and de-lighting priors. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 7220–7229. Association for Computing Machinery, 2025b. doi: 10.1145/3746027.3754548.

Mingwei Li, Hehe Fan, and Yi Yang. TransNormal: Dense visual semantics for diffusion-based transparent object normal estimation, 2026a. URL https://arxiv.org/abs/2602.00839.

Mingwei Li, Yi Yang, and Hehe Fan. TransNormal-2: Geometry-grounded rectified flow with edge-aware decoding for precise normal estimation, 2026b. URL https://arxiv.org/abs/2609.06665.

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.

William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *ACM SIGGRAPH Computer Graphics*, 21(4):163–169, 1987.

Jiahui Lyu, Bojian Wu, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Differentiable refraction-tracing for mesh reconstruction of transparent objects. *ACM Transactions on Graphics*, 39(6):1–13, 2020. doi: 10.1145/3414685.3417815.

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*, 43(6):1–12, 2024. doi: 10.1145/3687952.

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. In *ECCV*, 2020.

Daisuke Miyazaki, Masataka Kagesawa, and Katsushi Ikeuchi. Transparent surface modeling from a pair of polarization images. *IEEE TPAMI*, 26(1):73–82, 2004.

Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 3D gaussian ray tracing: Fast tracing of particle scenes. *ACM Transactions on Graphics*, 2024.

Jan U. Müller, Robin Tim Landsgesell, Leif Van Holland, Patrick Stotko, and Reinhard Klein. Moment-based 3D gaussian splatting: Resolving volumetric occlusion with order-independent transmittance. arXiv preprint arXiv:2512.11800, 2025.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Transactions on Graphics*, 41(4):102:1–102:15, 2022. doi: 10.1145/3528223.3530127.

Youngju Na, Jaeseong Yun, Soohyun Ryu, Hyunsu Kim, Sung-Eui Yoon, and Suyong Yeon. GLINT: Modeling scene-scale transparency via gaussian radiance transport. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026. URL https://arxiv.org/abs/2603.26181.

Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. KinectFusion: Real-time dense surface mapping and tracking. In *ISMAR*, pp. 127–136, 2011. doi: 10.1109/ISMAR.2011.6092378.

Shreeyak S. Sajjan, Matthew Moore, Mike Pan, Ganesh Nagaraja, Johnny Lee, Andy Zeng, and Shuran Song. ClearGrasp: 3D shape estimation of transparent objects for manipulation. In *ICRA*, 2020.

Frank Steinbrücker, Christian Kerl, Jürgen Sturm, and Daniel Cremers. Large-scale multi-resolution surface reconstruction from RGB-D sequences. In *ICCV*, pp. 3264–3271, 2013. doi: 10.1109/ICCV.2013.405.

Jia-Mu Sun, Tong Wu, Ling-Qi Yan, and Lin Gao. NU-NeRF: Neural reconstruction of nested transparent objects with uncontrolled capture environment. *ACM Transactions on Graphics*, 43 (6):1–14, 2024. doi: 10.1145/3687757.

Xin Sun, Iliyan Georgiev, Yun Fei, and Miloš Hašan. Stochastic ray tracing of transparent 3D gaussians. In *Eurographics Symposium on Rendering*, 2025. doi: 10.2312/sr.20251191.

Chinmay Talegaonkar, Yash Belhe, Ravi Ramamoorthi, and Nicholas Antipa. Volumetrically consistent 3D gaussian rasterization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10953–10963, 2025.

Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. DN-Splatter: Depth and normal priors for gaussian splatting and meshing. In *WACV*, 2025.

Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-NeRF: Structured view-dependent appearance for neural radiance fields. In *CVPR*, 2022.

Dongqing Wang, Tong Zhang, and Sabine Süsstrunk. NEMTO: Neural environment matting for novel view and relighting synthesis of transparent objects. In *ICCV*, 2023.

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. In *NeurIPS*, 2021.

Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. HF-NeuS: Improved surface reconstruction using high-frequency details. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1966–1978. Curran Associates, Inc., 2022. doi: 10.52202/068431-0143.

Bojian Wu, Yang Zhou, Yiming Qian, Minglun Gong, and Hui Huang. Full 3D reconstruction of transparent objects. *ACM Transactions on Graphics*, 37(4):103:1–103:11, 2018. doi: 10.1145/3197517.3201286.

Qi Wu, Janick Martinez Esturo, Ashkan Mirzaei, Nicolas Moenne-Loccoz, and Zan Gojcic. 3DGUT: Enabling distorted cameras and secondary rays in gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26036–26046. IEEE/CVF, 2025a. doi: 10.1109/CVPR52734.2025.02425.

Tianhao Wu, Hanxue Liang, Fangcheng Zhong, Gernot Riegler, Shimon Vainer, Jiankang Deng, and A. Cengiz Öztireli. αSurf: Implicit surface reconstruction for semi-transparent and thin objects with decoupled geometry and opacity. In *International Conference on 3D Vision (3DV)*, pp. 44–63. IEEE, 2025b. doi: 10.1109/3DV66043.2025.00011.

Yinghan Xu and John Dingliana. LayerGS: Decomposition and inpainting of layered 3D human avatars via 2D gaussian splatting. In *Pattern Recognition (ICPR 2026)*, volume 16814 of *Lecture Notes in Computer Science*, pp. 270–284. Springer Nature Switzerland, 2027. doi: 10.1007/978-3-032-31654-7_19. First online August 2026.

Zhiyuan Xu, Min Nan, Yuhang Guo, and Tong Wei. TSPE-GS: Probabilistic depth extraction for semi-transparent surface reconstruction via 3D gaussian splatting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(14):11478–11486, 2026. doi: 10.1609/aaai.v40i14.38130.

Ziyi Yang, Xinyu Gao, Yang-Tian Sun, Yi-Hua Huang, Xiaoyang Lyu, Wen Zhou, Shaohui Jiao, Xiaojuan Qi, and Xiaogang Jin. Spec-Gaussian: Anisotropic view-dependent appearance for 3D gaussian splatting. In *Advances in Neural Information Processing Systems*, volume 37, pp. 61192–61216. Curran Associates, Inc., 2024. doi: 10.52202/079017-1956.

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

Keyang Ye, Qiming Hou, and Kun Zhou. 3D gaussian splatting with deferred reflection. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, SIGGRAPH '24, pp. 1–10. ACM, 2024. doi: 10.1145/3641519.3657456.

Yeonghun Yoon, Hojoon Jung, Jaeyoon Lee, Taegwan Kim, Gyuhyun Kim, and Jongwon Choi. Through the water: Refractive gaussian splatting for water surface scenes. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(14):12081–12089, 2026. doi: 10.1609/aaai.v40i14.38197.

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-Splatting: Alias-free 3D gaussian splatting. In *CVPR*, 2024a.

Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 43(6):1–13, 2024b. doi: 10.1145/3687937.

Kunnong Zeng, Chensheng Peng, Yichen Xie, Masayoshi Tomizuka, and Cem Yuksel. RT-GS: Gaussian splatting with reflection and transmittance primitives. arXiv preprint arXiv:2604.00509, 2026.

Baowen Zhang, Chuan Fang, Rakesh Shrestha, Yixun Liang, Xiaoxiao Long, and Ping Tan. RaDe-GS: Rasterizing depth in gaussian splatting. arXiv preprint arXiv:2406.01467, 2024a. URL https://arxiv.org/abs/2406.01467.

Baowen Zhang, Chenxing Jiang, Heng Li, Shaojie Shen, and Ping Tan. Geometry-grounded gaussian splatting. *arXiv preprint arXiv:2601.17835*, 2026.

Haoran Zhang, Junkai Deng, Xuhui Chen, Fei Hou, Wencheng Wang, Hong Qin, Chen Qian, and Ying He. From transparent to opaque: Rethinking neural implicit surfaces with α-NeuS. In *Advances in Neural Information Processing Systems*, volume 37, pp. 28078–28102, 2024b.

Hongyu Zhou and Zorah Lähner. Geometry gaussians: Decoupling appearance and geometry in gaussian splatting. arXiv preprint arXiv:2606.05124, 2026. URL https://arxiv.org/abs/2606.05124.