DOI: 10.1111/cgf.70347

EUROGRAPHICS 2026 / B. Masia and J. Thies
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 45 (2026), Number 2

GS-2M: Material-aware Gaussian Splatting for High-fidelity Mesh Reconstruction

D. M. Nguyen[1], M. Avenhaus[2], and T. Lindemeier[2]

[1]Norwegian University of Science and Technology, Norway
[2]Carl Zeiss AG, Germany

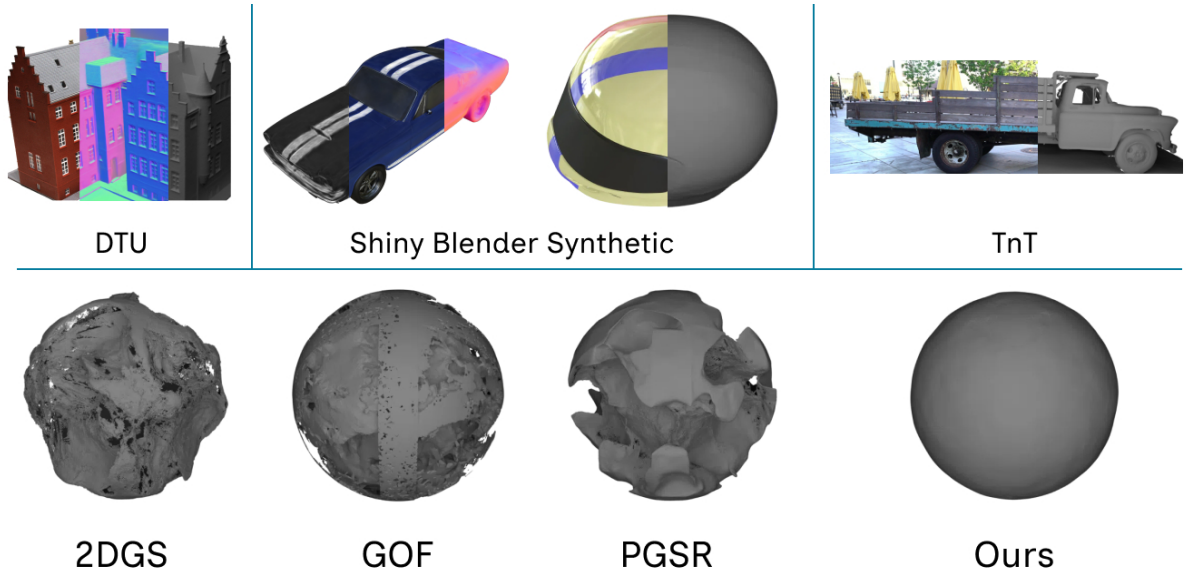

Figure 1: *Our approach reinforces mesh reconstruction by incorporating material decomposition into a joint optimization framework, producing high-fidelity triangle meshes even for reflective surfaces. We validate the effectiveness of our method (top row) with the DTU benchmark (left), the Shiny Blender Synthetic dataset (middle), and the TanksAndTemples dataset (right). We also provide qualitative comparisons (bottom row) between our method and state-of-the-art surface reconstruction methods to highlight the challenge of recovering triangle meshes for reflective objects.*

Abstract

We propose a material-aware optimization framework for high-fidelity mesh reconstruction from multi-view images based on 3D Gaussian Splatting, referred to as GS-2M. Previous works handle these tasks separately and struggle to reconstruct highly reflective surfaces, often relying on priors from external models to enhance the decomposition results. Conversely, our method addresses these two problems by jointly optimizing attributes relevant to the quality of rendered depth and normals, maintaining geometric details while being resilient to reflective surfaces. Although contemporary works effectively solve these tasks together, they often employ sophisticated neural components to learn scene properties, which hinders their performance at scale. To further eliminate these neural components, we propose a novel roughness supervision strategy based on multi-view photometric variation. When combined with a carefully designed loss and optimization process, our unified framework produces reconstruction results comparable to state-of-the-art methods, delivering accurate triangle meshes even for reflective surfaces. We validate the effectiveness of our approach with widely used datasets from previous works and qualitative comparisons with state-of-the-art surface reconstruction methods. Project page: https://ndming.github.io/publications/gs2m/.

CCS Concepts

• ***Computing methodologies*** → *Point-based Rendering;* • ***Geometry modeling*** → *3D Reconstruction; Scene Reconstruction;*

1. Introduction

Reconstructing triangle meshes from multi-view images is a highly relevant problem within the visual computing domain, allowing the acquisition of 3D models from photos without the need for laborious manual work. With the advent of Neural Radiance Fields (NeRF) [MST*20], numerous neural implicit surface reconstruction methods have been introduced [YGKL21, WLL*21, ZYL*22, DBD*22, WHH*23, LME*23]; however, they often require hours of training on high-end GPUs to achieve sufficient output quality, hindering their applicability in practice. While methods exist to reduce training time [MESK22, WHH*23], they are limited and often come at the expense of reconstruction performance.

Most recently, 3D Gaussian splatting (3DGS) [KKLD23] has emerged as an explicit alternative for representing radiance fields, achieving state-of-the-art (SoTA) rendering quality in novel view synthesis (NVS) while enabling real-time rasterization. Subsequent works in the domain gracefully inherit its computational advantage and adapt the method to various vision-based tasks, including surface reconstruction and scene decomposition from multi-view images. Despite producing high-quality meshes, SoTA explicit surface reconstruction methods often struggle to recover objects exhibiting reflection. In particular, they either rely solely on view-dependent radiance functions, i.e, spherical harmonics [DXX*24, HYC*24, YSG24, GGM*25], incorporate a small multi-layer perceptron (MLP) model for exposure compensation [CLY*25], or normal priors from pretrained models [WLW*24]. These limited appearance modelings may perform well for scenes with diffuse surfaces, yet suffer from those exhibiting highly varying photometric details, as shown in Figure 2. Conversely, 3DGS-based methods with sophisticated appearance modeling excel in decomposing material properties, yet little work has been demonstrated for their capability of complementing mesh reconstruction tasks for reflective surfaces. Recent SoTA explicit methods in material decomposition realized the significance of the underlying surface geometry and and brought these tasks together by borrowing components from neural-based approaches such as SDF backbones [ZWY25], geometry segmentation priors [LHG*25], or tensor factorization [ZCW*25] to maintain high-fidelity surface geometry. Despite delivering faithful decomposition results with geometrically feasible meshes, they inherit the performance penalty carried over by said neural components, hindering their runtime at scale. Moreover, these methods still demonstrate limited capability of reconstructing surfaces with sharp edges and geometric details, making them unfavorable for recovering diffuse but highly intricate objects.

To this end, we propose a material-aware optimization framework for high-fidelity mesh reconstruction from multi-view images based on 3D Gaussian Splatting, referred to as GS-2M. Our goal is to derive a solution that maintains the reconstruction quality to be on par with current SoTA surface reconstruction methods while simultaneously handling reflective objects, as demonstrated in Figure 1. By incorporating material parameters into the training pipeline, we identify the appearance properties of the underlying object and eliminate geometric artifacts caused by view-dependent effects, producing watertight meshes and smoother surfaces. To further eliminate neural components in this joint optimization framework, we propose a novel roughness supervision strategy based on multi-view photometric variation, completely independent of priors from pre-trained models. This allows us to scale the runtime performance of our framework without being constrained by external factors. Finally, we validate our method with the DTU [JDV*14] and TanksAndTemples (TnT) [KPZK17] benchmarks to demonstrate its ability to maintain reconstruction performance, and the Shiny Blender Synthetic dataset [VHM*22] for qualitative comparisons with SoTA surface reconstruction methods. To summarize, our contributions are:

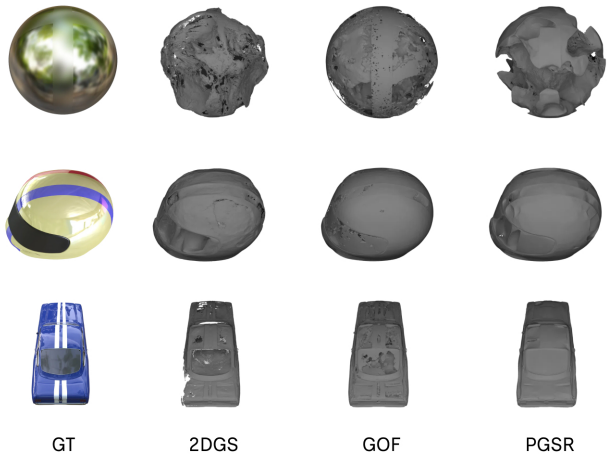

Figure 2: *The reconstructed meshes of reflective objects taken from the Shiny Blender Synthetic dataset [VHM*22], experimented on recent SoTA surface reconstruction methods: 2DGS [HYC*24], GOF [YSG24], and PGSR [CLY*25]. Due to the lack of appearance modeling, these methods often sacrifice geometric details for view-dependent effects caused by highly specular materials, resulting in non-watertight or distorted meshes.*

- A novel framework jointly optimizing 3D Gaussians for both mesh reconstruction and material decomposition from multi-view images, delivering comparable mesh reconstruction quality to SoTA, while being resilient to reflective surfaces.
- A roughness supervision strategy based on multi-view photometric variation, eliminating the dependence on neural components for appearance modeling.
- An integration of occlusion-aware check and multi-view normal consistency for SoTA mesh reconstruction methods, improving NVS performance while maintaining the reconstruction quality.

2. Related Work

In this section, we review recent methods based on 3DGS for mesh reconstruction and material decomposition from multi-view posed images. For completeness, classical and neural implicit reconstruction approaches are also discussed in Section 2.1 and 2.2, respectively. Section 2.3 then extensively reviews the recent SoTA neural explicit (3DGS-based) works that inspired GS-2M's design.

2.1. Traditional methods

Reconstruction of 3D geometry from photographs is an ill-posed problem that relies on certain assumptions about the underlying scene [FH15]. Among these cues, stereo correspondence has exhibited reasonable robustness and propelled a class of reconstruction algorithms known as multi-view stereo (MVS) [SCD*06]. MVS shares largely the same principle as GS-2M and other 3DGS-based methods: reconstruct the 3D geometry of the scene from multi-view images. Structure-from-Motion (SfM) [SSS06] is widely adopted to recover camera intrinsics and poses for each image, with an initial sparse pointcloud as a byproduct. From there, 3D meshes can be reconstructed via voxel-based optimization [SD99,SMP07], feature point growing [FP10,WYJT10], or depth fusion [SZFP16,XCW18, HSG24]. Like other traditional methods, their performance is affected by inconsistent appearance across input images, making it challenging to accurately capture complete geometric representations due to the ambiguities in the correspondence. By contrast, we model the appearance properties of the target object with a handful of material parameters as part of our pipeline, making it resilient to abrupt photometric changes across views.

2.2. Neural implicit methods

As NeRF is not mainly designed for surface reconstruction, radiance fields are replaced with implicit surfaces [NMOG20, OPG21] or signed distance functions (SDFs) [YKM*20] to better define isosurfaces from a volume density. These representations are later reparameterized [WLL*21, YGKL21] to be compatible with neural volume rendering as employed by NeRF. To further increase the reconstruction quality, auxiliary information is baked into the training pipeline as priors, such as patch warping with co-visibility masks [DBD*22], sparse SfM points [FXOT22a, ZYL*22], semantic segmentation [GPL*22], monocular depth [SCW*22], or monocular geometric features [YPN*22]. On the other hand, Neuralangelo [LME*23] leverages hash encodings as introduced by InstantNGP [MESK22] to eliminate the need for auxiliary guidance, while HF-NeuS [WSW22] adopts coarse-to-fine optimization for improved surface details. As previously stated, these implicit methods are SoTA in mesh reconstruction but demand expensive training resources, and attempts have been made to speed up their training time [WHH*23, YHR*23, LYZ*24]. However, these enhancements often come at the expense of reconstruction quality, while neural explicit methods retain sufficient performance without requiring high computing resources.

Neural implicit approaches for material decomposition often stem from inverse rendering frameworks. Early methods only consider direct lighting [BBJ*21, BJB*21, ZLW*21] to trade quality for computation speed. Subsequent works [SDZ*21, YZL*22, JLX*23] introduce indirect lighting but produce inferior results on highly reflective objects. To combat this, Ref-NeRF [VHM*22] decomposes colors into diffuse and specular terms, while NeRO [LWL*23] designs a novel light representation based on the split-sum approximation [KG13] of the rendering equation. Most recently, TensoSDF [LWZW24] achieves SoTA decomposition performance by expressing geometry as an implicit SDF and incorporating roughness-aware radiance fields.

2.3. Neural explicit methods

SuGaR [GL24] was among the first to adapt 3DGS for mesh reconstruction. By minimizing the SDF derived from the Gaussians and the scene density function, they encourage points to better align with the underlying surface. However, SuGaR only approximates the planarity of Gaussians with their minimum scaling axes, resulting in insufficient constraints on their overall shape during training. To address this loose approximation, 2D Gaussian Splatting (2DGS) [HYC*24] and GaussianSurfels [DXX*24] replace 3D Gaussian points with planar primitives. In particular, they collapse 3D Gaussians to planar ellipses to maximize the alignment between points and the object surface. Despite achieving view-consistent geometry, 2DGS and GaussianSurfels rely on mean/median or blended z-depth in camera space, which may cause ambiguity or biased depth rendering. At the other extreme, Gaussian Opacity Fields (GOF) [YSG24] constructs an opacity field from the trained Gaussians to extract the underlying surface by identifying its level set, drawing inspiration from ray-traced volume rendering of 3D Gaussians. MILo [GGM*25] later incorporates the mesh extraction step into the optimization process of GOF, simultaneously minimizing the image-based and surface consistency losses. However, GOF and MILo suffer from the expensive computational resources required for their sophisticated mesh extraction and training algorithms. PGSR [CLY*25] and GausSurf [WLW*24] are recent SoTA explicit methods in the field, with the former introducing unbiased depth rendering and multi-view constraints imposed for geometric and photometric consistency, while the latter borrowing the patch-matching technique from MVS to refine the rendered depth across views. As stated before, they both struggle to reconstruct reflective objects due to the lack of appearance modeling, producing distorted meshes even for simple scenes.

To address view-dependent issues stemming from highly specular surfaces, [JTL*24] introduces simplified shading functions to capture light-surface interactions, while [YHZ24] employs deferred rendering with a trainable reflection strength parameter. These ideas are quickly adopted in the literature and further enhanced for scene relighting [BZZ*24, WSL*25, SWW*25] and inverse rendering frameworks [YGL*24, LZF*24, ZZWZ25]. To further factorize lighting, most works employ differential environment cubemaps [LHK*20a] and sample them during training at various mip levels. This helps separate lighting from the object's intrinsic appearance, allowing for more accurate material decomposition. GS-ROR^2 [ZWY25], GlossyGS [LHG*25], and Ref-GS [ZCW*25] are SoTA works in this context by also realizing the significance of the underlying geometry to the decomposition performance. In particular, GS-ROR^2 jointly optimizes an SDF backbone for robust geometry, GlossyGS incorporates micro-facet features and priors to supervise neural materials, and Ref-GS builds upon 2DGS with tensorial factorization for material decomposition.

3. Method

To set up the necessary terminology and notations, we provide a prelude to 3DGS in Section 3.1. Section 3.2 discusses in detail the construct of GS-2M, including the choice of point primitive, definitions of depth and normals, and augmented material parameters.

3.1. Preliminary

To represent a scene, [KKLD23] initializes n anisotropic 3D Gaussians, $\{\mathcal{G}_0, \mathcal{G}_1, \mathcal{G}_2, \cdots, \mathcal{G}_{n-1}\}$, where each Gaussian $\mathcal{G}_i$ is parameterized by a center $\mu_i \in \mathbb{R}^3$, a scaling vector $\mathbf{s}_i \in \mathbb{R}^3$, and a quaternion $\mathbf{q}_i \in \mathbb{R}^4$. These are learnable parameters, with $\mathbf{s}_i$ and $\mathbf{q}_i$ further defining a scaling matrix $S_i \in \mathbb{R}^{3\times3}$ and a rotation matrix $R_i \in \mathrm{SO}(3)$. Evaluating $\mathcal{G}_i$ at a position $\mathbf{x} \in \mathbb{R}^3$ thus follows $\mathcal{G}_i(\mathbf{x}) = e^{-\frac{1}{2}(\mathbf{x}-\mu_i)^\top \Sigma_i^{-1}(\mathbf{x}-\mu_i)}$, where $\Sigma_i = R_i S_i S_i^\top R_i^\top$ is the 3D covariance matrix of $\mathcal{G}_i$. Suppose there's a camera $\mathcal{C}$ in the scene whose viewing transform is $V \in \mathbb{R}^{4\times4}$, we can apply EWA volume splatting [ZPvBG01] to find the 2D projection $\mathcal{G}_i^{2D}$ of $\mathcal{G}_i$ as seen by $\mathcal{C}$. Given the projected 2D Gaussians in image space, the alpha value α_i of any $\mathcal{G}_i^{2D}$ can be evaluated for any pixel $\mathbf{p} \in \mathbb{R}^2$ by evaluating $\mathcal{G}_i^{2D}(\mathbf{p})$, multiplied with a learnable opacity value ϕ_i corresponding to $\mathcal{G}_i$. From there, Equation 1 formulates the α-blending process to compute the rendered image $\hat{\mathcal{I}}$, where per pixel linear color $\hat{\mathcal{I}}(\mathbf{p}) \in \mathbb{R}^3$ is the alpha-composite value of the view-dependent color $\mathbf{c}_i \in \mathbb{R}^3$ evaluated from $\mathcal{G}_i$'s learnable spherical harmonics (SH) coefficients. Note that i in the equation indexes over the set $G_\mathbf{p}$ of 2D splats that contribute meaningful values to pixel $\mathbf{p}$ in front-to-back order.

$$\hat{\mathcal{I}}(\mathbf{p}) = \sum_{i \in G_\mathbf{p}} T_i \alpha_i \mathbf{c}_i, \quad T_i = \prod_{j=0}^{i-1}(1-\alpha_j) \tag{1}$$

Since most learnable parameters are randomly initialized, $\hat{\mathcal{I}}$ is not yet similar to the ground-truth image $\mathcal{I}$ captured under the same viewpoint. A loss function $\mathcal{L}$ can therefore be defined to measure how far $\hat{\mathcal{I}}$ is to $\mathcal{I}$, and a training process is then established to minimize $\mathcal{L}$ by optimizing all learnable parameters contributing to the rasterization of $\hat{\mathcal{I}}$. 3DGS uses a simple RGB photometric loss $\mathcal{L}_{\text{rgb}}$ composed of pixel-wise mean absolute error between $\hat{\mathcal{I}}$ and $\mathcal{I}$, weighted sum with an SSIM term, detailed in the supplementary.

3.2. GS-2M

As an overview, we construct GS-2M from PGSR [CLY*25] to maintain SoTA reconstruction performance. In particular, we employ unbiased depth rendering [CLY*25, HSG24] and multi-view constraints [CLY*25, HSG24, WLW*24] to encourage high-fidelity mesh reconstruction. For material decomposition, we draw inspiration from [JTL*24, LZF*24, ZWY25] and introduce two additional per-Gaussian learnable parameters with lighting captured via differential environment cubemaps. As stated before, we propose a novel roughness supervision strategy to help supervise material parameters without relying on neural components. In short, we leverage the existing multi-view construct of PGSR and measure photometric variation across views using the normalized cross correlation (NCC) loss applied to warped patches.

Unbiased depth rendering and normal as shortest axis

As highlighted in recent works [HSG24, CLY*25], blending z-depth in camera space results in biased depth. Concretely, nearby pixels observing the same Gaussian $\mathcal{G}_i$ have similar z values even though their true depth values are different, as illustrated in Figure 3. The unbiased depth is instead identified based on the hypothetical plane perpendicular to the normal $\mathbf{n}_i \in \mathbb{R}^3$ of $\mathcal{G}_i$. Following previous works [YHZ24, JTL*24, LZF*24, CLY*25, ZWY25], we assign the orientation axis (a column vector of R_i) corresponding to the shortest scaling direction (a scalar in $\mathbf{s}_i$) as $\mathbf{n}_i$, normalized and flipped to face $\mathcal{C}$. From there, we can compute the distance value $\bar{d}_i = \mu_i' \cdot (R_i \mathbf{n}_i)$ from C to $\mathcal{G}_i$, where μ_i' is the mean μ_i of $\mathcal{G}_i$ transformed to camera space. By replacing $\mathbf{c}_i$ in Equation 1 with $\bar{d}_i$ and $\mathbf{n}_i$, respectively, we obtain the distance map $\bar{\mathcal{D}}$ and normal map $\mathcal{N}$ defined for all $\mathbf{p}$. The unbiased depth map $\mathcal{D}$ is finally computed by rescaling the distance map with the inverted cosine of the angle between the normal and ray directions. Equation 2 details this operation, where $\tilde{\mathbf{p}}$ is the homogeneous version of $\mathbf{p}$, i.e $\tilde{\mathbf{p}} = [\mathbf{p}, 1]^\top$, and $K \in \mathbb{R}^{3\times3}$ is the intrinsic of $\mathcal{C}$.

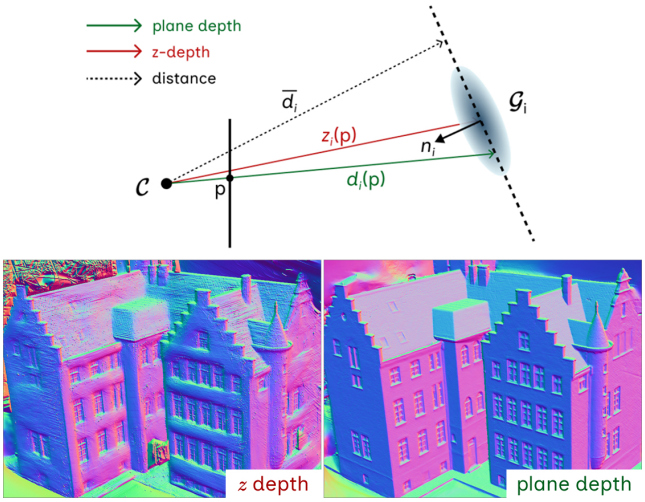

Figure 3: *Comparing depth quality when training with z-depth (left) and plane depth (right). We compare normal maps derived from rendered depth maps via a Sobel-like operator for better visualization. Training with z-depth results in biased and noisy depth values, while plane depth allows for more accurate and consistent distributions of Gaussians to capture geometric details.*

$$\mathcal{D}(\mathbf{p}) = \frac{\bar{\mathcal{D}}(\mathbf{p})}{\mathcal{N}(\mathbf{p}) \cdot (K^{-1}\tilde{\mathbf{p}})} \tag{2}$$

The use of plane depth requires planar or near-planar Gaussians, for which we employ the plane loss term $\mathcal{L}_{\text{plane}}$ to penalize one of the scaling scalars of all $\mathbf{s}_i$. We further impose the depth-normal constraint $\mathcal{L}_{\text{dn}}$ as used in previous works [HYC*24, HSG24, DXX*24, YSG24, CLY*25, WLW*24, JTL*24, LZF*24, ZWY25] to encourage their consistency. Both are defined in Equation 3 and further detailed in the supplementary.

$$\mathcal{L}_{\text{plane}} = \frac{\lambda_{\text{plane}}}{|\mathcal{V}|}\sum_{\mathbf{s}} ||\min(\mathbf{s}_i)||_1, \quad \mathcal{L}_{\text{dn}} = \frac{\lambda_{\text{dn}}}{|\mathcal{N}|}\sum_{\mathbf{p}} |\nabla\mathcal{I}(\mathbf{p})|^2 ||\mathbf{n}_\mathbf{p} - \hat{\mathbf{n}}_\mathbf{p}||_1 \tag{3}$$

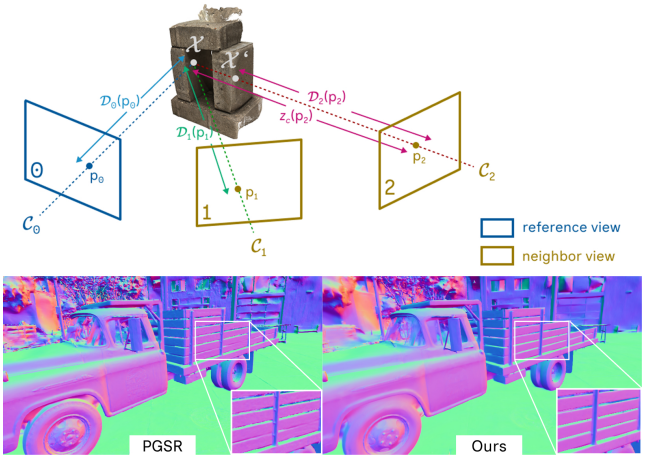

Figure 4: *Filtering invalid correspondences in neighboring views (top) and enhanced multi-view constraints with normal consistency (bottom). Correspondence pixel* $\mathbf{p}_2$ *of neighbor view* $\mathcal{C}_2$ *is excluded from multi-view loss calculations because its depth value is less than the z-coordinate of the camera-space point* $\mathcal{X}$ *back-projected from* $\mathbf{p}_0$ *in the reference view. We also sample values from normal maps rendered at reference and neighboring views to provoke multi-view normal consistency in high-frequency regions.*

Multi-view normal consistency and occlusion-aware filtering

As stated previously, we adopt PGSR [CLY*25]'s multi-view loss $\mathcal{L}_{\text{mv}}$—a weighted sum of multi-view geometric L_{g} and multi-view photometric L_{p} terms. However, PGSR only minimizes the reprojection error between sampled pixels in the reference view and those found via forward–backward projection of a neighboring view, resulting in underconstrained supervision for L_{g}. Inspired by [HSG24], we further encourage multi-view normal consistency by minimizing the difference in normal directions between the reference and neighboring views at sampled points in world space. This helps us achieve more consistent geometry in regions with high-frequency textures, as demonstrated in Figure 4. Please refer to the supplementary for the modified L_{g} term and details of $\mathcal{L}_{\text{mv}}$.

Another improvement we apply on top of PGSR is robust occlusion-aware filtering, as introduced in [HSG24]. Specifically, PGSR only approximates invalid correspondences by thresholding large reprojection noises, which is unreliable and ambiguous. We instead explicitly detect and reject these invalid pixels by comparing their depth values in the neighbor view's rendered depth map with z-coordinates of back-projected points in the same camera space, as shown in Figure 4.

Material modeling and differential environment lighting

We introduce two additional per-Gaussian learnable parameters—albedo $\mathbf{a}_i \in \mathbb{R}^3$ and roughness $\rho_i \in [0,1]$—to decompose intrinsic properties of surface appearance, allowing the optimization process to treat smooth (reflective) and rough (diffuse) regions differently. To learn these parameters, we employ a physically based rendering (PBR) pipeline, where the BRDF is formulated with the Cook-Torrance microfacet model [CT82, WMLT07]. In essence, the shading model combines diffuse and specular reflection, with the specular component governed by microfacet theory. We provide the mathematical construct of the BRDF in the supplementary.

Similar to depth and normal maps, we also render the albedo map $\mathcal{A}$ and roughness map $\mathcal{R}$ by α-blending per-Gaussian albedo and roughness parameters. Additionally, the Cook-Torrance shading model depends on a metallic fraction value between 0 and 1, for which we approximate from the roughness map as $\mathcal{M} = 1 - \mathcal{R}$. Finally, we composite all G-buffer renders into a PBR image $\bar{\mathcal{I}}$ (not to be confused with the rasterized image $\hat{\mathcal{I}}$) via a deferred rendering step [YHZ24, WSL*25]. Concretely, we perform deferred physically based shading in image space by feeding the rendered G-buffer channels (albedo, roughness, depth, and normals) into a Cook–Torrance BRDF, from which we separately compute the diffuse and specular lighting components before compositing them into the final PBR image. The RGB photometric loss can therefore be replaced with the PBR photometric loss $\mathcal{L}_{\text{pbr}}$ to supervise all components contributing to the final PBR composite. Similar to $\mathcal{L}_{\text{rgb}}$, $\mathcal{L}_{\text{pbr}}$ consists of the pixel-wise L_1 term and the SSIM term, applied on $\bar{\mathcal{I}}$ and $\mathcal{I}$, as defined in Equation 4.

$$\mathcal{L}_{\text{pbr}} = (1 - \lambda_{\text{SSIM}})L_1 + \lambda_{\text{SSIM}} L_{\text{D-SSIM}} \tag{4}$$

Per [Kaj86]'s rendering equation, the PBR compositing of the G-buffer requires incoming irradiance of the environment lighting. As with previous works [LZF*24, ZWY25], we employ a differential environment cubemap [LHK*20b] to learn incoming radiance, and prefilter it at various mip levels to sample irradiance values at runtime. As diffuse shading is independent of the viewing direction, we sample prefiltered lighting values from the base level of the cubemap. For the specular component, however, we adopt [KG13]'s split-sum approximation to help with the computation of the view-dependent integral, combining a constant roughness-aware lighting term and a BRDF's response term. The former is similar to irradiance sampled from a prefiltered cubemap, but at the mip level corresponding to the supplied roughness value. The latter, on the other hand, is pre-computed and stored in a 2D lookup texture (LUT) for fast retrieval of BRDF's responses at runtime. Please refer to the supplementary for the mathematical constructs of the rendering equation and lighting computation in our PBR pipeline.

Multi-view roughness supervision

Training with $\mathcal{L}_{\text{pbr}}$ alone to supervise appearance parameters is extremely under-constrained, resulting in noisy lighting and incoherent decomposition results. Previous works [ZCW*25, LHG*25, ZWY25] thus exploit neural components in terms of encoders–decoders or pretrained priors to learn scene parameters. We instead propose a lightweight roughness supervision strategy based solely on multi-view photometric variation, eliminating the need for neural modeling of scene components.

As stated earlier, we recognize the multi-view photometric NCC error between image patches as an indicator for highly reflective regions. If the surface nature is strongly view-dependent, switching to nearby viewpoints will likely cause the corresponding warped patch's texture to change significantly. We measure this variation

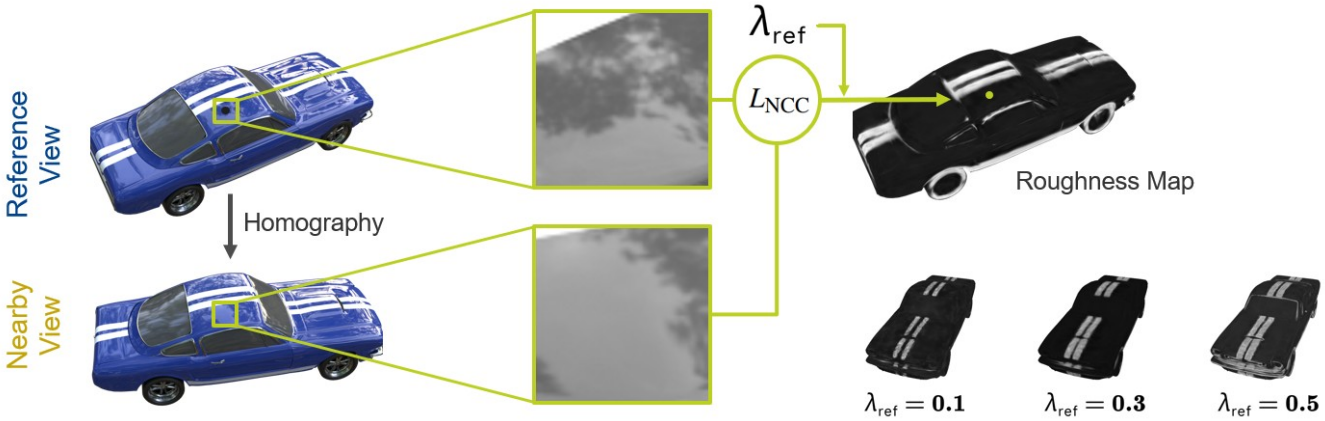

Figure 5: *Roughness supervision based on multi-view photometric variation. The photometric variation is quantified with the NCC error L_{NCC}, and a thresholding value λ_{ref} is used to penalize or reward the corresponding roughness values sampled from the rendered roughness map of the reference view. As λ_{ref} increases (bottom right), more and more regions become diffuse, i.e., their multi-view photometric variation is not regarded as inconsistency caused by reflective surfaces.*

for 3×3 grayscale patches, constructed from sampled pixels of ground-truth images, illustrated in Figure 5. The use of NCC helps emphasize texture-based similarity rather than absolute pixel intensity, thereby making the error resilient to brightness changes or geometric misalignment. Equation 5 describes how L_{NCC} is computed given a sampled image pixel $\mathbf{p}$, where $\mathcal{P}_r$ is the 3×3 image patch enclosing $\mathbf{p}$ in the reference view, and $\hat{\mathcal{P}}_n$ is the warped version of $\mathcal{P}_r$ in a nearby view via homography. We maintain, for each ground-truth image, a heuristic list of nearby viewpoints that are close to the reference viewpoint by jointly thresholding the Euclidean distance between camera centers and the angular difference between viewing rays, and then selecting a fixed number of candidates via stratified sampling over the sorted list.

$$L_{NCC}(\mathbf{p}) = 1 - \frac{\sum_p (\mathcal{P}_r(p) - \mu_{\mathcal{P}_r})(\hat{\mathcal{P}}_n(p) - \mu_{\hat{\mathcal{P}}_n})}{\sqrt{\sum_p (\mathcal{P}_r(p) - \mu_{\mathcal{P}_r})^2} \sqrt{\sum_p (\hat{\mathcal{P}}_n(p) - \mu_{\hat{\mathcal{P}}_n})^2}} \quad (5)$$

Once L_{NCC} is computed for each sampled pixel $\mathbf{p}$, we modulate the corresponding roughness values using a threshold λ_{ref}: roughness values at pixels with $L_{NCC} > \lambda_{ref}$ are added to the roughness loss $\mathcal{L}_{ro}$, and those below the threshold are subtracted. A fixed λref is not suitable for all scenes, as sharp reflections require higher thresholds to preserve non-diffuse regions, whereas blurred reflections benefit from lower thresholds to increase the sensitivity of $\mathcal{L}_{ro}$ to weak multi-view cues. In practice, we recommend choosing λ_{ref} based on the dominant material characteristics of the scene: for scenes containing primarily glossy or mirror-like objects with sharp view-dependent highlights, a higher threshold (e.g., $\lambda_{ref} \in [0.9, 1.1]$) helps avoid over-smoothing specular regions, whereas for scenes dominated by semi-glossy or rough materials with weak or blurred reflections, a lower threshold (e.g., $\lambda_{ref} \in [0.6, 0.9]$) makes the loss more responsive to subtle multi-view photometric inconsistencies. For mixed-material scenes, we found that intermediate values around $\lambda_{ref} \approx 0.9$ provide a robust trade-off without requiring per-object tuning.

Figure 6: *Simply relying on L_{NCC} to identify multi-view photometric variation results in incorrect roughness supervision (middle) at textureless regions. Replacing these regions with gradient-based patches helps L_{NCC} produce more faithful results (right).*

There's a limitation to the described supervision strategy so far, in which textureless regions yield high L_{NCC} despite being non-reflective. When there is no variation, the denominator of Equation 5 approaches zero, making L_{NCC} explode and unstable. To combat this, we filter out pixels in the reference views where the standard deviation $\sigma_r(\mathbf{p}) = \sqrt{\sum_p (\mathcal{P}_r(p) - \mu_{\mathcal{P}_r})^2}$ less than an empirically derived threshold of 0.01, and replace them with L_{NCC} but applied on the gradient versions of $\mathcal{P}_r$ and $\hat{\mathcal{P}}_n$, as illustrated in Figure 6. Once L_{NCC} values are filtered, we implement the roughness loss $\mathcal{L}_{ro}$ using Equation 6. Here, to avoid abrupt constraining of roughness values $\mathcal{R}(\mathbf{p})$, we remap L_{NCC} via the tanh function, and apply a scale factor k_{ro} that controls how rapidly the weights change from promoting to penalizing $\mathcal{R}(\mathbf{p})$. We set $k_{ro} = 8.0$, determined via empirical experiments.

$$\mathcal{L}_{ro} = \frac{1}{|\mathcal{R}|} \sum_{\mathbf{p}} \tanh\left(k_{ro}(L_{NCC}(\mathbf{p}) - \lambda_{ref})\right) \mathcal{R}(\mathbf{p}) \quad (6)$$

CD ↓	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean	Time
VolSDF	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	0.83	0.98	0.56	0.37	1.13	0.59	0.60	1.45	0.95	0.78	0.52	1.43	0.36	0.45	0.45	**0.77**	∼ 8h
RegSDF	0.60	1.41	0.64	0.43	1.34	0.62	0.60	0.90	0.92	1.02	0.60	0.59	0.30	0.41	0.39	**0.72**	∼ 3.5h
NeuS2	0.56	0.76	0.49	0.37	0.92	0.71	0.76	1.22	1.08	0.63	0.59	0.89	0.40	0.48	0.55	**0.70**	∼ 5m
NeuralWarp	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**	> 12h
Neuralangelo	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	**0.61**	∼ 16h
SuGaR	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**	15–45m
GaussianSurfels	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**	6.67m
2DGS	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**	10.9m
GOF	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**	18.4m
MILo	0.43	0.74	0.34	0.37	0.80	0.74	0.70	1.21	1.22	0.66	0.62	0.80	0.37	0.76	0.48	**0.68**	25m
PGSR	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
GausSurf	0.35	0.55	0.34	0.34	0.77	0.58	0.51	1.10	0.69	0.60	0.43	0.49	0.32	0.40	0.37	**0.52**	7.2m
Ours w/o BRDF	0.34	0.57	0.39	0.34	0.75	0.51	0.49	1.03	0.62	0.57	0.46	0.56	0.31	0.38	0.36	**0.51**	22.4m
Ours	0.40	0.59	0.38	0.35	0.75	0.55	0.59	1.06	0.62	0.59	0.47	0.48	0.34	0.36	0.37	**0.53**	51.0m

Table 1: *Quantitative results of mesh reconstruction performance for the DTU dataset [JDV*14]. The Chamfer Distance (CD) ↓ for each scan of the 15 scenes in the dataset is reported. We compare our approach with SoTA neural implicit methods: VolSDF [YGKL21], NeuS [WLL*21], RegSDF [ZYL*22], NeuS2 [WHH*23], NeuralWarp [DBD*22], Neuralangelo [LME*23]; and explicit methods: SuGaR [GL24], GaussianSurfels [DXX*24], 2DGS [HYC*24], GOF [YSG24], MILo [GGM*25], GausSurf [WLW*24], PGSR [CLY*25]. The top-three best performing results are highlighted in color for ease of visualization, with red, orange, and yellow indicating 1st, 2nd, and 3rd, respectively. Note that the reconstruction runtimes are not directly comparable because these methods are trained on different GPUs.*

With $\mathcal{L}_{\text{ro}}$ regulating roughness, we apply the total variance (TV) loss $\mathcal{L}_{\text{tv}}$ to the rendered normal map $\mathcal{N}$, where $\mathcal{R}$ acts as weights detached from the computation graph. Specifically, the roughness values are remapped to penalize varied normals at smooth regions more aggressively, and the combined depth-normal consistency term $\mathcal{L}_{\text{dn}}$ jointly helps refine distorted or non-watertight surfaces. Finally, we incorporate a smoothness term, $\mathcal{L}_{\text{sm}}$, into the total loss to regulate BRDF parameters [LZF*24, ZWY25]. The details of these loss terms are provided in the supplementary.

Training and mesh extraction

Our joint optimization process first trains for 5,000 iterations to bootstrap the initialized 3D Gaussians. During this stage, all loss terms are suppressed, except $\mathcal{L}_{\text{rgb}}$, $\mathcal{L}_{\text{plane}}$, and the binary cross-entropy loss $\mathcal{L}_{\text{alpha}}$ between the rendered alpha map and ground-truth masks if such masks are provided. Following the bootstrap stage, the training jointly optimizes geometric and material parameters, where all loss terms are activated and $\mathcal{L}_{\text{rgb}}$ is replaced with $\mathcal{L}_{\text{pbr}}$, as defined in Equation 7. Note that each loss term has a corresponding weight λ, the details of which are provided in the supplementary. Thanks to $\mathcal{L}_{\text{ro}}$, the learning of BRDF parameters can operate in parallel with other parameters to regulate geometric details. All scenes are trained for at most 30,000 iterations.

$$\mathcal{L} = \mathcal{L}_{\text{plane}} + \mathcal{L}_{\text{alpha}} + \mathcal{L}_{\text{dn}} + \mathcal{L}_{\text{mv}} + \mathcal{L}_{\text{tv}} + \mathcal{L}_{\text{sm}} + \mathcal{L}_{\text{ro}} + \mathcal{L}_{\text{pbr}} \quad (7)$$

After training, we extract triangle meshes via TSDF fusion [NIH*11]. We render RGB-D images from all viewpoints, optionally mask depths using ground-truth alpha, and integrate them into a TSDF volume using Open3D [ZPK18]. The final mesh is obtained by marching cubes, filtered to a single connected component, and assigned per-vertex RGB colors.

4. Experiments

We conduct experiments to validate the effectiveness of our method using the DTU benchmark [JDV*14], two scenes from the TnT dataset [KPZK17], and the Shiny Blender Synthetic dataset [VHM*22]. All experiments are run using a single RTX4090 GPU with 24GB VRAM, with the model split into two variants:

- **Ours** w/o BRDF: training is performed without joint optimization of BRDF parameters. This helps validate the effectiveness of the new occlusion-aware filtering and multi-view normal consistency integrated into $\mathcal{L}_{\text{mv}}$.
- **Ours**: the joint optimization process as described in Section 3.2.

Table 1 compares the performance of our model with SoTA implicit and explicit mesh reconstruction methods using the DTU benchmark. As with previous works, we report the Chamfer distance (CD) for each of the 15 scenes in the dataset, together with the average runtime. It is clear that all explicit methods, including GS-2M, consume significantly less training time and computing resources compared to implicit methods. Our joint training pipeline, however, takes roughly double the reconstruction time of the variant without BRDF optimization. This is due to the deferred rendering step and an extra overhead resulting from the computation of our under-optimized $\mathcal{L}_{\text{ro}}$. Nevertheless, the quantitative results show that both variants perform on par with SoTA explicit surface reconstruction methods and outperform all neural implicit methods, maintaining the reconstruction quality even when jointly optimizing with BRDF parameters and the PBR pipeline. In some specific scenes, we even outperform the current SoTA explicit methods, proving the effectiveness of the modified multi-view constraints.

Table 1 also reveals that our joint optimization framework with BRDF parameters introduces an inevitable cost–benefit trade-off. While the deferred PBR pipeline and roughness regulariza-

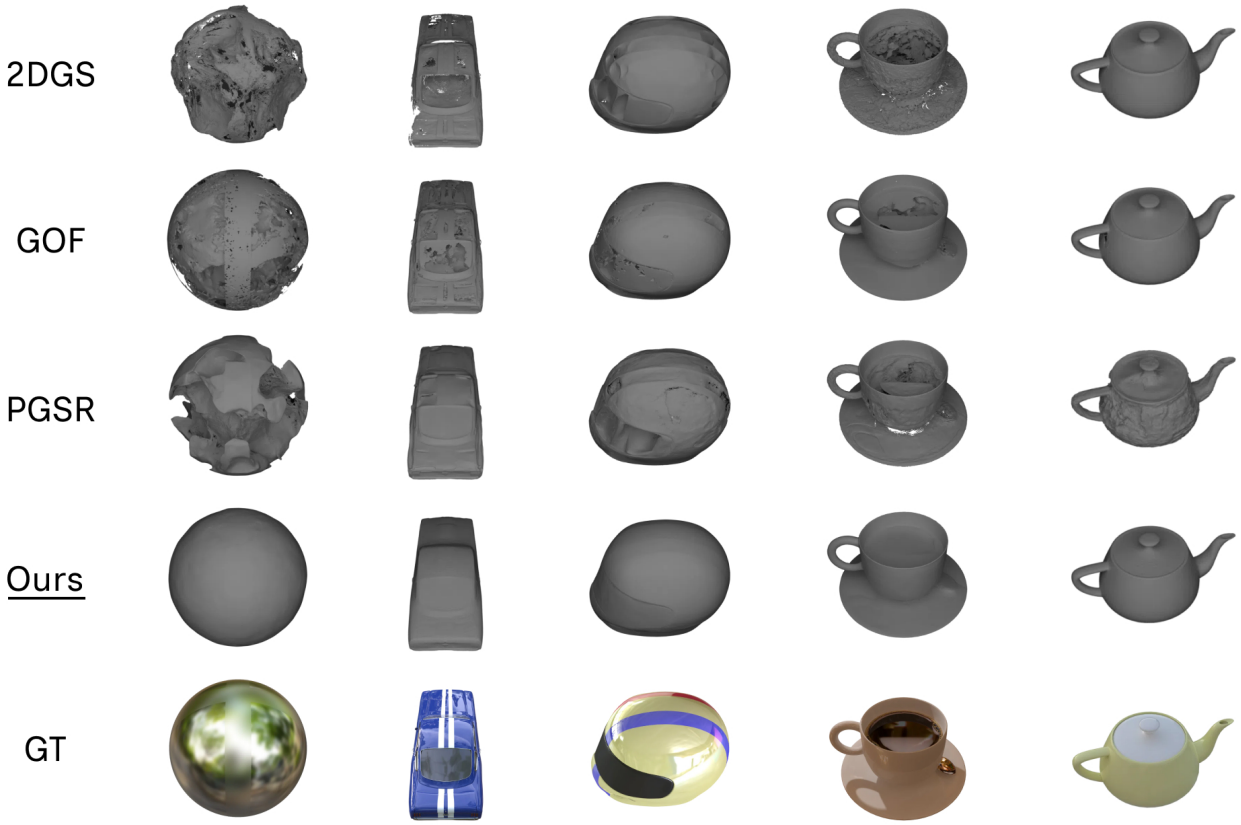

Figure 7: *Qualitative comparisons for the Shiny Blender Synthetic dataset [VHM*22]. We compare the reconstructed meshes of our full model with current SoTA neural explicit methods: 2DGS [HYC*24], GOF [YSG24], PGSR [CLY*25]. Due to the limited appearance modeling, these methods sacrifice geometric details for view-dependent effects caused by specular surfaces, resulting in non-watertight or distorted meshes. In contrast, our method produces more uniform surfaces thanks to the joint constraints of the $\mathcal{L}_{\text{ro}}$, $\mathcal{L}_{\text{tv}}$, and $\mathcal{L}_{\text{dn}}$ loss terms. Moreover, GS-2M still maintains the reconstruction quality on the DTU benchmark, making it a unified solution for both mesh reconstruction and material decomposition. Please refer to the supplementary material and our project page for more experimental results.*

tion roughly double the training time, they do not always improve the Chamfer distance; in several scenes, our model without BRDF marginally outperforms the full model. We attribute this to shape–reflectance ambiguities and the additional degrees of freedom introduced by BRDF parameters, which can slightly hinder geometric convergence in scenes with weak or inconsistent reflective cues. Nevertheless, these differences are small and scene-dependent, and the joint model remains competitive with SoTA explicit methods while additionally recovering physically meaningful material properties, making it more suitable for photorealistic rendering and downstream PBR tasks.

The major advantage of our framework, however, is the ability to reconstruct reflective objects, as shown in Figure 7. We compare the reconstructed mesh of our joint optimization solution with meshes produced by recent SoTA neural explicit methods. These qualitative results are produced using code provided by the authors of the respective works. For a fair comparison, we match the number of training iterations and modify the published code to handle white backgrounds without generating floaters. The qualitative comparisons validate the effectiveness of our joint optimization framework. Specifically, the multi-view self-supervision term $\mathcal{L}_{\text{ro}}$ helps detect diffuse and specular regions, driving the TV loss $\mathcal{L}_{\text{tv}}$ to force smooth normals accordingly via roughness weighting. These smooth normals, in turn, regulate the geometric details thanks to the depth-normal consistency loss $\mathcal{L}_{\text{dn}}$. These terms cooperate in a unified optimization framework to deliver high-quality reconstruction results for both diffuse and specular surfaces.

As with previous neural implicit methods, we report the novel-view synthesis (NVS) performance of our model using the DTU benchmark, measured by the peak signal-to-noise ratio (PSNR) between the rendered and ground-truth images. For a fair comparison, we evaluate the NVS metrics on the training set, aligning with

PSNR ↑	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
RegSDF	24.78	23.06	23.47	22.21	28.57	25.53	21.81	28.89	26.81	27.91	24.71	25.13	26.84	21.67	28.25	**25.31**
NeuS	26.49	26.17	27.66	27.78	30.63	27.42	25.38	30.00	26.40	29.63	25.87	28.82	28.80	27.36	31.19	**28.00**
NeuS2	28.44	27.14	29.70	29.67	31.75	27.83	24.84	31.24	26.86	30.57	26.05	28.93	28.98	27.82	32.48	**28.82**
VolSDF	26.28	25.61	26.55	26.76	31.57	31.50	29.38	33.23	28.03	32.13	33.16	31.49	30.33	34.90	34.75	**30.38**
Neuralangelo	30.64	27.78	32.70	34.18	35.15	35.89	31.47	36.82	30.13	35.92	36.61	32.60	31.20	38.41	38.05	**33.84**
PGSR	32.20	28.07	31.10	33.41	33.99	33.22	32.16	32.83	31.31	33.82	36.03	34.64	32.71	37.33	37.09	**33.33**
Ours w/o BRDF	33.16	30.47	31.54	34.04	35.78	34.41	32.47	33.76	31.58	34.55	36.74	35.54	33.20	38.30	37.73	**34.22**
Ours	31.62	29.30	31.55	33.41	35.05	33.32	31.77	37.44	31.87	34.62	34.48	35.73	32.20	37.70	37.85	**33.86**

Table 2: *Quantitative results of novel-view synthesis (NVS) performance for the DTU dataset [JDV*14]. We report the Peak Signal-to-Noise Ratio (PSNR) ↑ for each scene of the 15 scenes in the dataset. We compare our approach with SoTA surface reconstruction methods: VolSDF [YGKL21], NeuS [WLL*21], RegSDF [ZYL*22], NeuS2 [WHH*23], Neuralangelo [LME*23], and PGSR [CLY*25]. Except for neural implicit methods, all NVS metrics are reproduced using the published code and evaluated on unmasked images. The top-three best performing results are highlighted, with red, orange, and yellow indicating 1st, 2nd, and 3rd, respectively. Incorporating multi-view normal consistency and occlusion-aware filtering enhances NVS quality while preserving the fidelity of the extracted mesh. Our method outperforms all SoTA reconstruction approaches, thanks to these additions. Note that the NVS performance for the DTU dataset is evaluated on the whole training data, aligning with previous works.*

previous works. Table 2 compares the NVS quality of our method with neural implicit works that reported the same metric. Additionally, we reproduce the NVS performance of PGSR [CLY*25] using their publicly available code. The table reveals that enhancing the multi-view loss term with normal consistency and filtering occluded samples helps our solution outperform all SoTA surface reconstruction methods in terms of NVS quality. For the PBR variant of our model, there is a slight drop in NVS performance, which we attribute to the limited constraints for the environment lighting and albedo. Specifically, the optimization process of our method still leaves abundant freedom for these two components, resulting in noise artifacts in the material decomposition results. Nevertheless, our joint optimization with BRDF parameters still keeps the NVS quality on par with current SoTA methods.

Finally, we extend our experiments to unbounded scenes using the TnT dataset [KPZK17]. Unlike DTU and Shiny Blender Synthetic, the TnT dataset features sequences acquired outside controlled labs under realistic conditions, including both challenging outdoor scenes and complex indoor environments. Since our joint framework adopts the PBR pipeline designed for object-centric scenes, we only use the TnT dataset to validate the w/o BRDF variant. This design choice reflects our intentional focus on object-centric reconstruction, where controlled lighting, limited background clutter, and stable material properties make physically based shading both meaningful and reliable. Nevertheless, even outside this intended regime, our framework retains practical advantages, including fast convergence, low memory footprint relative to neural implicit methods, and strong geometric consistency from the modified multi-view constraints. As with previous works, we report the F1 scores, balancing the precision and recall tests between the reconstructed mesh and the ground-truth scan. Table 3 compares the reconstruction performance of our model with SoTA neural implicit and explicit methods for the Barn and Truck scenes. The comparison reveals that the new modification to the multi-view geometric term maintains the quality of the reconstructed mesh even without the exposure compensation module that PGSR employed to benchmark with the TnT dataset. However,

F1-Score ↑	NeuS	Geo-NeuS	Neuralangelo	SuGaR	2DGS	GOF	PGSR	**Ours** w/o BRDF
Barn	0.29	0.33	0.70	0.14	0.36	0.51	0.66	0.57
Truck	0.45	0.45	0.48	0.26	0.26	0.58	0.66	0.67
Mean	0.37	0.39	0.59	0.15	0.31	0.55	0.66	0.62

Table 3: *Quantitative results of mesh reconstruction performance for the Truck and Barn scenes from the TnT dataset [KPZK17], where the F1-score ↑ is reported. We compare our reconstructed meshes with those of NeuS [WLL*21], Geo-NeuS [FXOT22b], Neuralangelo [LME*23], SuGaR [GL24], 2DGS [HYC*24], GOF [YSG24], and PGSR [CLY*25]. The top-three best performing results are highlighted, with red, orange, and yellow indicating 1st, 2nd, and 3rd, respectively.*

we restrict our experiments to these two scenes, as the remaining ones cause out-of-memory errors due to the lack of control over generating new Gaussians, especially for scenes with overwhelming background details. We discuss the limitations of our approach and promising solutions in the next section.

5. Ablation, Limitations, Future Work

For ablation, we study the effectiveness of the enhanced multi-view geometric term and the proposed roughness supervision loss. Since our framework is largely constructed from components introduced in previous works [CLY*25, HSG24, LZF*24, ZWY25], please refer to their respective papers for deeper insights. Figure (8) compares the appearance decomposition results when training with and without the multi-view roughness supervision term ($\mathcal{L}_{\text{ro}}$). The proposed loss reduces noise artifacts in the captured lighting and prevents specular highlights from bleeding into the albedo. Conversely, relying solely on $\mathcal{L}_{\text{pbr}}$ is extremely under-constrained, contaminating learned BRDF parameters and environment lighting.

Table 4 studies the effect of two modifications made to the multi-view geometric constraint L_{g}, using the DTU benchmark. The ablation reveals that the addition of multi-view normal consistency to the L_{g} term substantially enhances both surface reconstruction and NVS performance. As demonstrated in Figure 4, the new L_{g}

DTU benchmark	CD ↓	PSNR ↑
Ours w/o mv normal	0.58	26.73
Ours w/o filtering	0.53	33.76
Ours	0.53	33.86

Table 4: *Ablation study for the modified multi-view geometric constraint, conducted on the DTU benchmark [JDV*14]. Introducing multi-view normal consistency substantially increases reconstruction and NVS performance, while occlusion filtering further enhances NVS quality.*

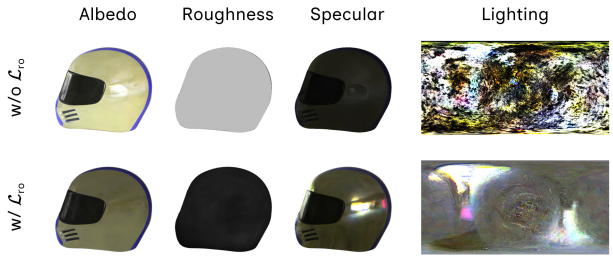

Figure 8: *Training without $\mathcal{L}_{ro}$ regulating roughness (top row) results in noisy lighting and albedo bleeding. In comparison, the use of $\mathcal{L}_{ro}$ (bottom row) enables the optimization process to learn the correct roughness values at shiny regions.*

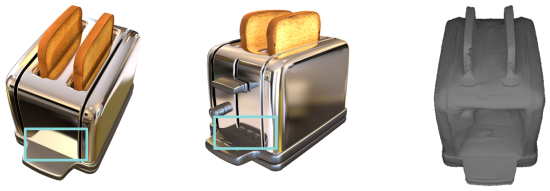

Figure 9: *The reconstructed mesh of the Toaster scene from the Shiny Blender Synthetic dataset [ZCW*25, MLMP*24]. We fail to recover this scene due to the limited capability of the employed shading model.*

Figure 10: *The approximation of metallic from roughness is not coherent with the appearance properties of some objects. Notice how the white stripes of the car in the rendered metallic map are approximated as dielectric, where they should be metal.*

also helps achieve more consistent geometry in regions with high-frequency textures. The integration of occlusion-aware filtering, on the other hand, only causes a slight improvement in NVS quality. Nevertheless, we recognize certain limitations of our method that provide opportunities for future enhancement.

Under-constrained albedo and lighting Despite the effectiveness of the roughness supervision term, the optimization process of our method still lacks constraints to supervise albedo and lighting. Although contemporary works handle this ill-posed problem with encoders–decoders [ZCW*25] or pre-trained priors [LHG*25], we prefer self-supervision strategies similar to $\mathcal{L}_{ro}$. In other words, finding reliable features to guide albedo and lighting without heavily relying on neural components is an interesting direction that we regard as a potential improvement for decomposing appearance.

Self-reflection and self-shadowing The incorporation of BRDF parameters helps cope with reflective surfaces, yet our implementation cannot faithfully decompose the appearances of objects exhibiting self-reflection. Figure 9 illustrates our method's limitation in reconstructing such objects, where self-reflecting regions pose challenges to the optimization process. These appearance properties require ray-based approaches [GGL*23, MLMP*24] to model indirect lighting, and we regard this as a promising enhancement to our joint supervision framework.

Learnable metallic Currently, we approximate the metallic component of the BRDF as $m = 1 - \rho$, which is not ideal in some scenes. Figure 10 illustrates one such case, where metallic values along the white stripes are not coherent with the car surface because they are derived from the roughness map. To this end, making the metallic parameter learnable is a preferred improvement, yet this would require advanced supervision strategies to avoid overfitting, which we leave as a future perspective.

Better densification As discussed in the previous section, we encounter out-of-memory errors when reconstructing unbounded scenes [KPZK17]. This is the direct consequence of using the unmodified adaptive density control (ADC) of 3DGS [KKLD23], where excessive Gaussians are generated due to the vast background details. Moreover, the shading function we employ for our method is designed for object-centric scenes, making the joint optimization pipeline unfavorable for these scenarios. We are interested in adopting recent outstanding works [KRS*24] to help with reconstructing large-scale scenes.

Object-centric scenes Our framework is primarily designed for object-centric reconstruction because the adopted PBR pipeline and deferred shading formulation are not well suited for large-scale or unbounded scenes with complex illumination and extensive background geometry. While our method can still recover reasonable geometry in such cases, the lack of explicit modeling for global illumination and environment lighting limits both stability and scalability. We therefore position our approach as a high-fidelity object-level reconstruction method and regard extensions toward unbounded scenes as an important direction for future work.

6. Conclusion

We have presented and discussed our material-aware, joint-optimization framework for high-fidelity mesh reconstruction 3D Gaussian splatting. We started introducing the emergence of neural rendering as a faithful paradigm to replace manual, laborious work in visual computing domains. While neural implicit methods promise their effectiveness in addressing a large number of vision tasks, we chose to pursue Gaussian splatting approaches due to their explicit and resource-friendly nature. On assessing the capabilities of SoTA neural explicit methods for mesh reconstruction, we discovered that they struggle to reconstruct highly reflective surfaces due to the lack of appearance modeling. We therefore propose a joint optimization solution with BRDF parameters independent of external factors and a novel roughness supervision strategy based solely on multi-view photometric variation. The experiments show that our framework maintains the reconstruction performance with SoTA for diffuse objects while simultaneously handling highly reflective ones, making it suitable for both mesh reconstruction and material decomposition. Despite still carrying several limitations and ideal for object-centric scenes, we believe our solution is a starting point in unifying both tasks, aiming toward high-fidelity surface and appearance reconstruction.

Acknowledgements

This research was carried out within the COSI Erasmus Mundus Master's Program. We acknowledge the support of Carl Zeiss AG for providing computational resources and research infrastructure.

References

[BBJ*21] Boss M., Braun R., Jampani V., Barron J. T., Liu C., Lensch H. P.: Nerd: Neural reflectance decomposition from image collections. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (2021), pp. 12664–12674. doi:10.1109/ICCV48922.2021.01245. 3

[BJB*21] Boss M., Jampani V., Braun R., Liu C., Barron J. T., Lensch H. P. A.: Neural-pil: neural pre-integrated lighting for reflectance decomposition. In *Proceedings of the 35th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2021), NIPS '21, Curran Associates Inc. 3

[BZZ*24] Bi Z., Zeng Y., Zeng C., Pei F., Feng X., Zhou K., Wu H.: Gs3: Efficient relighting with triple gaussian splatting. In *SIGGRAPH Asia 2024 Conference Papers* (New York, NY, USA, 2024), SA '24, Association for Computing Machinery. URL: https://doi.org/10.1145/3680528.3687576, doi:10.1145/3680528.3687576. 3

[CLY*25] Chen D., Li H., Ye W., Wang Y., Xie W., Zhai S., Wang N., Liu H., Bao H., Zhang G.: Pgsr: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. *IEEE Transactions on Visualization and Computer Graphics 31*, 9 (Sept. 2025), 6100–6111. URL: http://dx.doi.org/10.1109/TVCG.2024.3494046, doi:10.1109/tvcg.2024.3494046. 2, 3, 4, 5, 7, 8, 9

[CT82] Cook R. L., Torrance K. E.: A reflectance model for computer graphics. *ACM Trans. Graph. 1*, 1 (Jan. 1982), 7–24. URL: https://doi.org/10.1145/357290.357293, doi:10.1145/357290.357293. 5

[DBD*22] Darmon F., Bascle B., Devaux J., Monasse P., Aubry M.: Improving neural implicit surfaces geometry with patch warping. In *CVPR* (2022). 2, 3, 7

[DXX*24] Dai P., Xu J., Xie W., Liu X., Wang H., Xu W.: High-quality surface reconstruction using gaussian surfels. In *ACM SIGGRAPH 2024 Conference Papers* (2024), Association for Computing Machinery. 2, 3, 4, 7

[FH15] Furukawa Y., Hernández C.: *Multi-View Stereo: A Tutorial*. Now Foundations and Trends, 2015. doi:10.1561/0600000052. 3

[FP10] Furukawa Y., Ponce J.: Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 32*, 8 (2010), 1362–1376. doi:10.1109/TPAMI.2009.161. 3

[FXOT22a] Fu Q., Xu Q., Ong Y.-S., Tao W.: Geo-neus: geometry-consistent neural implicit surfaces learning for multi-view reconstruction. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2022), NIPS '22, Curran Associates Inc. 3

[FXOT22b] Fu Q., Xu Q., Ong Y.-S., Tao W.: Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)* (2022). 9

[GGL*23] Gao J., Gu C., Lin Y., Zhu H., Cao X., Zhang L., Yao Y.: Relightable 3d gaussian: Real-time point cloud relighting with brdf decomposition and ray tracing. *arXiv:2311.16043* (2023). 10

[GGM*25] Guédon A., Gomez D., Maruani N., Gong B., Drettakis G., Ovsjanikov M.: Milo: Mesh-in-the-loop gaussian splatting for detailed and efficient surface reconstruction. *ACM Transactions on Graphics* (2025). URL: https://anttwo.github.io/milo/. 2, 3, 7

[GL24] Guédon A., Lepetit V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. *CVPR* (2024). 3, 7, 9

[GPL*22] Guo H., Peng S., Lin H., Wang Q., Zhang G., Bao H., Zhou X.: Neural 3D Scene Reconstruction with the Manhattan-world Assumption . In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, June 2022), IEEE Computer Society, pp. 5501–5510. URL: https://doi.ieeecomputersociety.org/10.1109/CVPR52688.2022.00543, doi:10.1109/CVPR52688.2022.00543. 3

[HSG24] Huang Z., Shi Y., Gong M.: Visibility-aware pixelwise view selection for multi-view stereo matching. In *Pattern Recognition: 27th International Conference, ICPR 2024, Kolkata, India, December 1–5, 2024, Proceedings, Part XVIII* (Berlin, Heidelberg, 2024), Springer-Verlag, p. 130–144. URL: https://doi.org/10.1007/978-3-031-78456-9_9, doi:10.1007/978-3-031-78456-9_9. 3, 4, 5, 9

[HYC*24] Huang B., Yu Z., Chen A., Geiger A., Gao S.: 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH 2024 Conference Papers* (2024), Association for Computing Machinery. doi:10.1145/3641519.3657428. 2, 3, 4, 7, 8, 9

[JDV*14] Jensen R., Dahl A., Vogiatzis G., Tola E., Aanæs H.: Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (2014), IEEE, pp. 406–413. 2, 7, 9, 10

[JLX*23] Jin H., Liu I., Xu P., Zhang X., Han S., Bi S., Zhou X., Xu Z., Su H.: Tensoir: Tensorial inverse rendering. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2023), pp. 165–174. doi:10.1109/CVPR52729.2023.00024. 3

[JTL*24] Jiang Y., Tu J., Liu Y., Gao X., Long X., Wang W., Ma Y.: Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2024), pp. 5322–5332. doi:10.1109/CVPR52733.2024.00509. 3, 4

[Kaj86] KAJIYA J. T.: The rendering equation. *SIGGRAPH Comput. Graph. 20*, 4 (Aug. 1986), 143–150. URL: https://doi.org/10.1145/15886.15902, doi:10.1145/15886.15902. 5

[KG13] KARIS B., GAMES E.: Real shading in unreal engine 4. In *Proceedings of Physically Based Shading Theory Practice* (2013), vol. 4, p. 1. 3, 5

[KKLD23] KERBL B., KOPANAS G., LEIMKUEHLER T., DRETTAKIS G.: 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph. 42*, 4 (July 2023). URL: https://doi.org/10.1145/3592433, doi:10.1145/3592433. 2, 4, 10

[KPZK17] KNAPITSCH A., PARK J., ZHOU Q.-Y., KOLTUN V.: Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics 36*, 4 (2017). 2, 7, 9, 10

[KRS*24] KHERADMAND S., REBAIN D., SHARMA G., SUN W., TSENG Y.-C., ISACK H., KAR A., TAGLIASACCHI A., YI K. M.: 3d gaussian splatting as markov chain monte carlo. In *Advances in Neural Information Processing Systems* (2024), Globerson A., Mackey L., Belgrave D., Fan A., Paquet U., Tomczak J., Zhang C., (Eds.), vol. 37, Curran Associates, Inc., pp. 80965–80986. URL: https://proceedings.neurips.cc/paper_files/paper/2024/file/93be245fce00a9bb2333c17ceae4b732-Paper-Conference.pdf. 10

[LHG*25] LAI S., HUANG L., GUO J., CHENG K., PAN B., LONG X., LYU J., LV C., GUO Y.: Glossygs: Inverse rendering of glossy objects with 3d gaussian splatting. *IEEE Transactions on Visualization and Computer Graphics* (2025), 1–14. doi:10.1109/TVCG.2025.3547063. 2, 3, 5, 10

[LHK*20a] LAINE S., HELLSTEN J., KARRAS T., SEOL Y., LEHTINEN J., AILA T.: Modular primitives for high-performance differentiable rendering. *ACM Trans. Graph. 39*, 6 (Nov. 2020). URL: https://doi.org/10.1145/3414685.3417861, doi:10.1145/3414685.3417861. 3

[LHK*20b] LAINE S., HELLSTEN J., KARRAS T., SEOL Y., LEHTINEN J., AILA T.: Modular primitives for high-performance differentiable rendering. *ACM Transactions on Graphics 39*, 6 (2020). 5

[LME*23] LI Z., MÜLLER T., EVANS A., TAYLOR R. H., UNBERATH M., LIU M.-Y., LIN C.-H.: Neuralangelo: High-fidelity neural surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2023). 2, 3, 7, 9

[LWL*23] LIU Y., WANG P., LIN C., LONG X., WANG J., LIU L., KOMURA T., WANG W.: Nero: Neural geometry and brdf reconstruction of reflective objects from multiview images. *ACM Trans. Graph. 42*, 4 (July 2023). URL: https://doi.org/10.1145/3592134, doi:10.1145/3592134. 3

[LWZW24] LI J., WANG L., ZHANG L., WANG B.: Tensosdf: Roughness-aware tensorial representation for robust geometry and material reconstruction. *ACM Trans. Graph. 43*, 4 (July 2024). URL: https://doi.org/10.1145/3658211, doi:10.1145/3658211. 3

[LYZ*24] LI H., YANG X., ZHAI H., LIU Y., BAO H., ZHANG G.: Vox-surf: Voxel-based implicit surface representation. *IEEE Transactions on Visualization and Computer Graphics 30*, 3 (Mar. 2024), 1743–1755. URL: https://doi.org/10.1109/TVCG.2022.3225844, doi:10.1109/TVCG.2022.3225844. 3

[LZF*24] LIANG Z., ZHANG Q., FENG Y., SHAN Y., JIA K.: GS-IR: 3D Gaussian Splatting for Inverse Rendering . In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (Los Alamitos, CA, USA, June 2024), IEEE Computer Society, pp. 21644–21653. URL: https://doi.ieeecomputersociety.org/10.1109/CVPR52733.2024.02045, doi:10.1109/CVPR52733.2024.02045. 3, 4, 5, 7, 9

[MESK22] MÜLLER T., EVANS A., SCHIED C., KELLER A.: Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph. 41*, 4 (July 2022), 102:1–102:15. URL: https://doi.org/10.1145/3528223.3530127, doi:10.1145/3528223.3530127. 2, 3

[MLMP*24] MOENNE-LOCCOZ N., MIRZAEI A., PEREL O., DE LUTIO R., ESTURO J. M., STATE G., FIDLER S., SHARP N., GOJCIC Z.: 3d gaussian ray tracing: Fast tracing of particle scenes. *ACM Transactions on Graphics and SIGGRAPH Asia* (2024). 10

[MST*20] MILDENHALL B., SRINIVASAN P. P., TANCIK M., BARRON J. T., RAMAMOORTHI R., NG R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV* (2020). 2

[NIH*11] NEWCOMBE R. A., IZADI S., HILLIGES O., MOLYNEAUX D., KIM D., DAVISON A. J., KOHLI P., SHOTTON J., HODGES S., FITZGIBBON A.: Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality* (USA, 2011), ISMAR '11, IEEE Computer Society, p. 127–136. URL: https://doi.org/10.1109/ISMAR.2011.6092378, doi:10.1109/ISMAR.2011.6092378. 7

[NMOG20] NIEMEYER M., MESCHEDER L., OECHSLE M., GEIGER A.: Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), pp. 3501–3512. doi:10.1109/CVPR42600.2020.00356. 3

[OPG21] OECHSLE M., PENG S., GEIGER A.: UNISURF: Unifying Neural Implicit Surfaces and Radiance Fields for Multi-View Reconstruction . In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)* (Los Alamitos, CA, USA, Oct. 2021), IEEE Computer Society, pp. 5569–5579. URL: https://doi.ieeecomputersociety.org/10.1109/ICCV48922.2021.00554, doi:10.1109/ICCV48922.2021.00554. 3

[SCD*06] SEITZ S., CURLESS B., DIEBEL J., SCHARSTEIN D., SZELISKI R.: A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)* (2006), vol. 1, pp. 519–528. doi:10.1109/CVPR.2006.19. 3

[SCW*22] SUN J., CHEN X., WANG Q., LI Z., AVERBUCH-ELOR H., ZHOU X., SNAVELY N.: Neural 3d reconstruction in the wild. In *ACM SIGGRAPH 2022 Conference Proceedings* (New York, NY, USA, 2022), SIGGRAPH '22, Association for Computing Machinery. URL: https://doi.org/10.1145/3528233.3530718, doi:10.1145/3528233.3530718. 3

[SD99] SEITZ S. M., DYER C. R.: Photorealistic scene reconstruction by voxel coloring. *Int. J. Comput. Vision 35*, 2 (Nov. 1999), 151–173. URL: https://doi.org/10.1023/A:1008176507526, doi:10.1023/A:1008176507526. 3

[SDZ*21] SRINIVASAN P. P., DENG B., ZHANG X., TANCIK M., MILDENHALL B., BARRON J. T.: Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 7491–7500. doi:10.1109/CVPR46437.2021.00741. 3

[SMP07] SINHA S. N., MORDOHAI P., POLLEFEYS M.: Multi-view stereo via graph cuts on the dual of an adaptive tetrahedral mesh. In *2007 IEEE 11th International Conference on Computer Vision* (2007), pp. 1–8. doi:10.1109/ICCV.2007.4408997. 3

[SSS06] SNAVELY N., SEITZ S. M., SZELISKI R.: Photo tourism: exploring photo collections in 3d. *ACM Trans. Graph. 25*, 3 (July 2006), 835–846. URL: https://doi.org/10.1145/1141911.1141964, doi:10.1145/1141911.1141964. 3

[SWW*25] SHI Y., WU Y., WU C., LIU X., ZHAO C., FENG H., ZHANG J., ZHOU B., DING E., WANG J.: Gir: 3d gaussian inverse rendering for relightable scene factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025), 1–12. doi:10.1109/TPAMI.2025.3575937. 3

[SZFP16] SCHÖNBERGER J. L., ZHENG E., FRAHM J.-M., POLLEFEYS M.: Pixelwise view selection for unstructured multi-view stereo. In *Computer Vision – ECCV 2016* (Cham, 2016), Leibe B., Matas J., Sebe

N., Welling M., (Eds.), Springer International Publishing, pp. 501–518. 3

[VHM*22] VERBIN D., HEDMAN P., MILDENHALL B., ZICKLER T., BARRON J. T., SRINIVASAN P. P.: Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2022), pp. 5481–5490. doi:10.1109/CVPR52688.2022.00541. 2, 3, 7, 8

[WHH*23] WANG Y., HAN Q., HABERMANN M., DANIILIDIS K., THEOBALT C., LIU L.: Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)* (2023). 2, 3, 7, 9

[WLL*21] WANG P., LIU L., LIU Y., THEOBALT C., KOMURA T., WANG W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *NeurIPS* (2021). 2, 3, 7, 9

[WLW*24] WANG J., LIU Y., WANG P., LIN C., HOU J., LI X., KOMURA T., WANG W.: Gaussurf: Geometry-guided 3d gaussian splatting for surface reconstruction. *arXiv preprint arXiv:2411.19454* (2024). 2, 3, 4, 7

[WMLT07] WALTER B., MARSCHNER S. R., LI H., TORRANCE K. E.: Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics Conference on Rendering Techniques* (Goslar, DEU, 2007), EGSR'07, Eurographics Association, p. 195–206. 5

[WSL*25] WU T., SUN J.-M., LAI Y.-K., MA Y., KOBBELT L., GAO L.: Deferredgs: Decoupled and relightable gaussian splatting with deferred shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence 47*, 8 (2025), 6307–6319. doi:10.1109/TPAMI.2025.3560933. 3, 5

[WSW22] WANG Y., SKOROKHODOV I., WONKA P.: Hf-neus: improved surface reconstruction using high-frequency details. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2022), NIPS '22, Curran Associates Inc. 3

[WYJT10] WU T.-P., YEUNG S.-K., JIA J., TANG C.-K.: Quasi-dense 3d reconstruction using tensor-based multiview stereo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2010), pp. 1482–1489. doi:10.1109/CVPR.2010.5539796. 3

[XCW18] XU H., CAI Y., WANG R.: Depth estimation in multi-view stereo based on image pyramid. In *Proceedings of the 2018 2nd International Conference on Computer Science and Artificial Intelligence* (New York, NY, USA, 2018), CSAI '18, Association for Computing Machinery, p. 345–349. URL: https://doi.org/10.1145/3297156.3297238, doi:10.1145/3297156.3297238. 3

[YGKL21] YARIV L., GU J., KASTEN Y., LIPMAN Y.: Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems* (2021). 2, 3, 7, 9

[YGL*24] YE K., GAO C., LI G., CHEN W., CHEN B.: Geosplatting: Towards geometry guided gaussian splatting for physically-based inverse rendering. *arXiv preprint arXiv:2410.24204* (2024). 3

[YHR*23] YARIV L., HEDMAN P., REISER C., VERBIN D., SRINIVASAN P. P., SZELISKI R., BARRON J. T., MILDENHALL B.: Bakedsdf: Meshing neural sdfs for real-time view synthesis, 2023. URL: https://arxiv.org/abs/2302.14859, arXiv:2302.14859. 3

[YHZ24] YE K., HOU Q., ZHOU K.: 3d gaussian splatting with deferred reflection. In *ACM SIGGRAPH 2024 Conference Papers* (New York, NY, USA, 2024), SIGGRAPH '24, Association for Computing Machinery. URL: https://doi.org/10.1145/3641519.3657456, doi:10.1145/3641519.3657456. 3, 4, 5

[YKM*20] YARIV L., KASTEN Y., MORAN D., GALUN M., ATZMON M., RONEN B., LIPMAN Y.: Multiview neural surface reconstruction by disentangling geometry and appearance. In *Advances in Neural Information Processing Systems* (2020), Larochelle H., Ranzato M., Hadsell R., Balcan M., Lin H., (Eds.), vol. 33, Curran Associates, Inc., pp. 2492–2502. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/1a77befc3b608d6ed363567685f70e1e-Paper.pdf. 3

[YPN*22] YU Z., PENG S., NIEMEYER M., SATTLER T., GEIGER A.: Monosdf: exploring monocular geometric cues for neural implicit surface reconstruction. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (Red Hook, NY, USA, 2022), NIPS '22, Curran Associates Inc. 3

[YSG24] YU Z., SATTLER T., GEIGER A.: Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics* (2024). 2, 3, 4, 7, 8, 9

[YZL*22] YAO Y., ZHANG J., LIU J., QU Y., FANG T., MCKINNON D., TSIN Y., QUAN L.: Neilf: Neural incident light field for physically-based material estimation. In *Computer Vision – ECCV 2022* (Cham, 2022), Avidan S., Brostow G., Cissé M., Farinella G. M., Hassner T., (Eds.), Springer Nature Switzerland, pp. 700–716. 3

[ZCW*25] ZHANG Y., CHEN A., WAN Y., SONG Z., YU J., LUO Y., YANG W.: Ref-gs: Directional factorization for 2d gaussian splatting. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2025), pp. 26483–26492. doi:10.1109/CVPR52734.2025.02466. 2, 3, 5, 10

[ZLW*21] ZHANG K., LUAN F., WANG Q., BALA K., SNAVELY N.: Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), pp. 5449–5458. doi:10.1109/CVPR46437.2021.00541. 3

[ZPK18] ZHOU Q.-Y., PARK J., KOLTUN V.: Open3D: A modern library for 3D data processing. *arXiv:1801.09847* (2018). 7

[ZPvBG01] ZWICKER M., PFISTER H., VAN BAAR J., GROSS M.: Ewa volume splatting. In *Proceedings Visualization, 2001. VIS '01.* (2001), pp. 29–538. doi:10.1109/VISUAL.2001.964490. 4

[ZWY25] ZHU Z.-L., WANG B., YANG J.: Gs-ror^2: Bidirectional-guided 3dgs and sdf for reflective object relighting and reconstruction, 2025. URL: https://arxiv.org/abs/2406.18544, arXiv:2406.18544. 2, 3, 4, 5, 7, 9

[ZYL*22] ZHANG J., YAO Y., LI S., FANG T., MCKINNON D., TSIN Y., QUAN L.: Critical regularizations for neural surface reconstruction in the wild. In *CVPR* (2022). URL: https://arxiv.org/abs/2206.03087v1. 2, 3, 7, 9

[ZZWZ25] ZHOU Y., ZHANG F.-L., WANG Z., ZHANG L.: Rtr-gs: 3d gaussian splatting for inverse rendering with radiance transfer and reflection, 2025. URL: https://arxiv.org/abs/2507.07733, arXiv:2507.07733. 3