

Car-GS: Addressing Reflective and Transparent Surface Challenges in 3D Car Reconstruction

Congcong Li, Jin Wang, Xiaomeng Wang, Xingchen Zhou, Wei Wu,
Yuzhi Zhang, and Tongyi Cao[✉]
DeepRoute.AI

{congcongli, jinwang03, xiaomengwang, xingchenzhou,
weiwu, yuzhizhang, tongyicao}@deeproute.ai

Abstract

3D car modeling is crucial for applications in autonomous driving systems, virtual and augmented reality, and gaming. However, due to the distinctive properties of cars, such as highly reflective and transparent surface materials, existing methods often struggle to achieve accurate 3D car reconstruction. To address these limitations, we propose Car-GS, a novel approach designed to mitigate the effects of specular highlights and the coupling of RGB and geometry in 3D geometric and shading reconstruction (3DGS). Our method incorporates three key innovations: First, we introduce view-dependent Gaussian primitives to effectively model surface reflections. Second, we identify the limitations of using a shared opacity parameter for both image rendering and geometric attributes when modeling transparent objects. To overcome this, we assign a learnable geometry-specific opacity to each 2D Gaussian primitive, dedicated solely to rendering depth and normals. Third, we observe that reconstruction errors are most prominent when the camera view is nearly orthogonal to glass surfaces. To address this issue, we develop a quality-aware supervision module that adaptively leverages normal priors from a pre-trained large-scale normal model. Experimental results demonstrate that Car-GS achieves precise reconstruction of car surfaces and significantly outperforms prior methods. The project page is available at [Car-GS](#).

1. Introduction

Accurate 3D reconstruction is a critical task in various domains, including autonomous driving simulation and virtual/augmented reality (VR/AR) applications. However, reconstructing 3D models of cars presents significant challenges due to the complex surface properties of vehicles. Car surfaces are often highly reflective, such as glossy paints, and frequently transparent, such as windows and

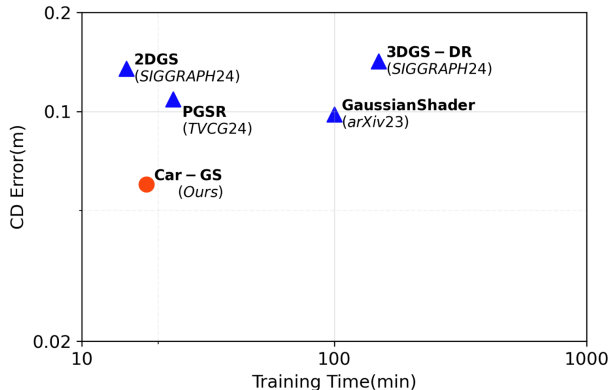


Figure 1. Comparison of various methods based on training time and chamfer distance (CD) error. Our approach not only achieves the highest accuracy but also demonstrates a relatively short training time, highlighting its balance between superior performance and efficiency. This makes it well-suited for real-time applications.

windshields. These properties pose considerable difficulties in accurately recovering both geometry and texture during 3D reconstruction.

Existing methods, such as NeRF [24] and its extensions, including Mip-NeRF [1], Mip-NeRF 360 [2], and NeuS [34], represent 3D scenes as sets of emission radiance points and compute view-dependent colors based on the viewing direction. However, these approaches do not account for the full interaction of light rays as they travel from the source to the camera, neglecting phenomena such as light scattering and reflection. To address these limitations, Ref-NeRF [31] incorporates surface light field rendering [4, 36] and replaces NeRF’s directional parameterization with an integrated reflection encoding. This modification significantly enhances the realism and accuracy of specular reflections.

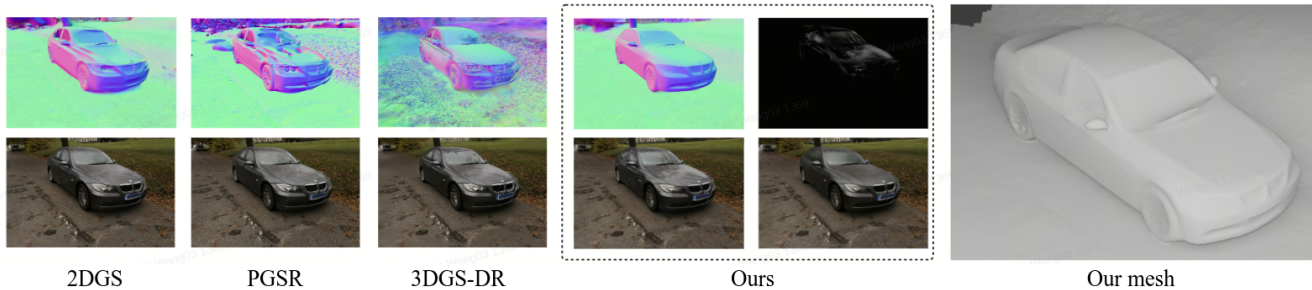


Figure 2. Our Car-GS method accurately separates reflections and recovers surface normals, achieves photo-realistic rendering, and demonstrates superior reconstruction quality of the car geometry, including detailed handling of reflective and transparent regions such as the car’s body and windshield.

Recent advancements, such as feature grid-based encoding for the directional domain [37], have significantly enhanced the efficiency of directional representations. In the context of 3D Gaussian Splatting (3DGS) [17], directly applying view direction reflections as view-dependent color queries within efficient Gaussian splatting frameworks introduces several challenges. Specifically, each primitive in the model independently inherits both orientation and spherical harmonics (SH) color, which can result in misalignment when the viewing direction changes during parameter updates. To address this issue, Spec-Gaussian [40] incorporates smooth regularization and higher-order view-dependent color modeling into the rendering process, thereby achieving notable improvements in rendering reflective surfaces. Nevertheless, despite these advancements in rendering quality, accurately modeling geometry remains a significant challenge.

In this paper, we introduce Car-GS, a novel approach that enhances the reconstruction of vehicles by mitigating the impact of reflective and transparent surfaces. Specifically, we propose using view-dependent Gaussians to model surface reflections. Furthermore, we identify that a crucial limitation of existing methods for modeling transparent surfaces is the shared opacity parameter used for rendering both image and geometric attributes. To overcome this limitation, we assign a learnable, geometry-specific opacity parameter to each 2D Gaussian primitive, which is used exclusively for rendering depth and normals, thereby ensuring a more accurate representation of transparent surfaces. In addition, we incorporate a pre-trained normal model [43] to provide additional geometry supervision during the training process. However, predictions of this model may not be perfect, and we observe that errors are most prominent when the view ray is nearly orthogonal to glass surfaces. Therefore, we develop a quality-aware supervision module that can adaptively refine the rendered normals, leading to improved reconstruction accuracy.

In summary, we make the following contributions:

- We propose the use of view-dependent Gaussians to

model surface reflections, thereby enhancing the accuracy of 3D car reconstruction by effectively addressing the challenges posed by reflective surfaces.

- We introduce a learnable, geometry-specific opacity parameter for each 2D Gaussian primitive, dedicated solely to rendering depth and normals. This approach overcomes the limitations of using a shared opacity parameter for both image and geometric attributes in existing methods, ensuring a more precise representation of transparent surfaces.
- We develop a quality-aware supervision module that leverages normal priors from a pre-trained, large-scale normal model. This module adaptively refines the rendered normals, thereby improving reconstruction accuracy by mitigating errors in normal predictions.
- Our approach achieves state-of-the-art performance in car surface reconstruction, outperforming existing methods in terms of accuracy and robustness.

2. Related Work

2.1. Novel View Synthesis

Recent advances in neural implicit representations, particularly Neural Radiance Fields (NeRF) [3, 6, 13, 20, 24], have revolutionized the field of novel view synthesis. The emergence of NeRF has catalyzed numerous architectural innovations and optimization techniques in this domain. Instant-NGP [25] significantly enhanced computational efficiency through the integration of hash-based encoding and adaptive sampling strategies, demonstrating orders of magnitude improvement in both training and inference latency. TensorIR [10] extended the classical NeRF framework by incorporating tensor decomposition techniques for inverse rendering, while MegaNeRF [29] introduced a scalable block-based architecture capable of handling large-scale scene reconstruction.

The introduction of 3D Gaussian Splatting (3DGS) and its variants [7, 17, 23, 28, 46] marked a paradigm shift from implicit to explicit scene representations. This ap-

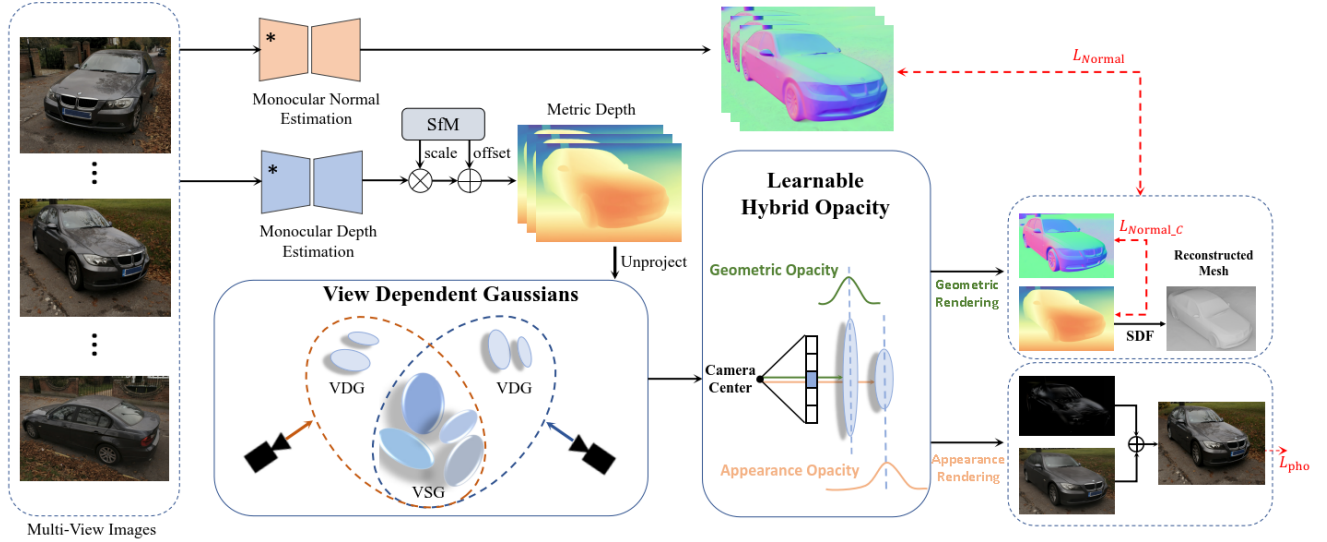


Figure 3. **Overview of Car-GS.** We initialize View-Dependent Gaussians (VDG) and View-Shared Gaussians (VSG) using monocular depth estimates aligned with structure-from-motion (SfM). VDG models view-specific attributes, while VSG captures shared information. During rendering, a learnable hybrid opacity is applied to RGB images and depth/normal maps. Additionally, a quality-aware supervision module leverages pretrained normal priors to regulate rendered normals, especially in reflective regions.

proach models scenes using a set of 3D Gaussian primitives with associated spherical harmonic coefficients, enabling efficient rasterization-based rendering without the computational overhead of volumetric sampling. Although 3DGS achieves superior visual fidelity and temporal coherence compared to NeRF-based methods, it encounters challenges in maintaining geometric consistency across multiple views. The subsequent development of 2D Gaussian Splatting (2DGS) [15] addressed these limitations by constraining the Gaussian primitives to disk-like structures, effectively reducing the degrees of freedom in the geometric representation. However, 2DGS exhibits notable limitations in accurately reconstructing scenes with specular surfaces and complex reflective properties.

2.2. Surface Reconstruction

Surface Reconstruction of General Object. Surface reconstruction of general object has been a long-standing goal in computer vision. Several NeRF-based reconstruct object surface by incorporating intermediate representations such as occupancy [26] or signed distance fields [34, 41]. Although NeRF-based frameworks demonstrate powerful surface reconstruction capabilities, the use of stacked multi-layer perceptron (MLP) layers limits both inference speed and representational capacity. To overcome these limitations, subsequent research has focused on reducing reliance on MLP layers by decomposing scene information into more separable structures, such as points [38] and voxels [18, 19, 21]. Recently, 3DGS-based surface reconstruction methods [5, 14, 48] have seen rapid development. To

solve the problem of difficulty in accurately reconstructing surfaces, some approaches focus on reducing 3D Gaussians to 2D Gaussians by applying a set of regularization terms, ensuring that the Gaussian primitives align with the object surfaces [8, 12, 14]. Additionally, some methods incorporate priors from large-scale datasets [30] or multi-view stereo techniques [35], or employ specialized surface extraction algorithms [42, 45] to recover 3D geometry from 3D Gaussians. However, these methods are difficult to reconstruct reflective and transparent object surfaces.

Surface Reconstruction of Reflective and Transparent Object. Objects with strong reflective properties present a significant challenge for surface reconstruction using methods such as vanilla NeRF [24] and NeuS [34]. This challenge arises from the high-frequency variations in reflections that occur across different viewing angles, resulting in view-dependent color inconsistencies at the same surface point, a phenomenon known as anisotropy. To overcome these limitations, various approaches have been proposed to enhance reflective scene reconstruction within the NeRF framework. Notable examples include methods like Ref-NeRF [31], NeRF-Casting [32], and inverse rendering techniques such as NeRO [22], which specifically address the complexities of anisotropy to improve reconstruction accuracy in reflective environments.

Ref-NeRF [31] modified NeRF’s original formulation by replacing outgoing radiance with reflective radiance, introducing view-dependent scene attributes to decouple specular components. NeRF-Casting [32] employed fundamental ray-tracing principles, sampling along reflection rays

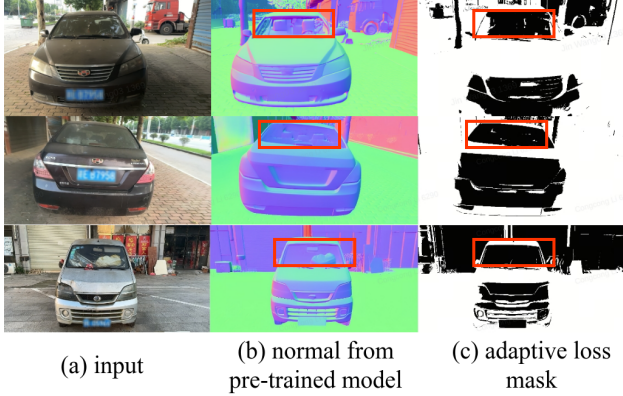


Figure 4. (a), (b), and (c) represent the input image, the normal map from the pretrained model [43], and the loss mask computed by Eq. (10), respectively. In the mask, black indicates a value of 0, while white indicates a value of 1. We observe that errors in the pretrained model’s predictions are most prominent when the camera view is nearly orthogonal to the glass surface. Therefore, we adaptively assign a value of 0 to these orthogonal regions.

to achieve high-quality reconstruction of scenes containing reflective objects. While NeuS-based methods are generally better suited for surface representation, their inherent isotropic mesh representation poses significant challenges when dealing with complex reflective scenes, resulting in notable artifacts. UniSDF [33] introduced a learnable weighting network to combine camera view radiance fields with reflected view radiance fields. Their approach, rather than directly modeling reflective and non-reflective regions separately, demonstrated superior results in view-dependent mesh reconstruction. Ref-NeuS [11] achieved high-quality reflective surface reconstruction by incorporating reflection-aware photometric loss and reflection direction-dependent radiance to reduce ambiguities caused by reflective surfaces. Gaussian Shader [16], built upon 3D Gaussian Splatting, presents a novel approach to modeling both reflective and non-reflective regions. It employs simplified shading functions to decouple the modeling of reflective scenes, offering an effective solution for handling complex reflective environments.

3. Method

Our approach is based on 2DGS [14] and we aim to accurately reconstruct the surfaces of cars. An overview of our method is shown in Fig. 3. Specifically, we first present view-dependent Gaussians to model reflections from each view. Next, we introduce a novel learnable hybrid opacity to reduce ambiguity during the rendering of both appearance and geometry maps. Finally, we introduce a quality-aware supervision module to adaptively refine rendered normals.

3.1. View-Dependent Gaussians

In computer graphics, the color of an object’s surface, I_{obj} , is equal to the sum of ambient light I_{ambient} , diffuse reflection I_{diffuse} , and specular reflection I_{specular} :

$$I_{\text{obj}} = I_{\text{ambient}} + I_{\text{diffuse}} + I_{\text{specular}}. \quad (1)$$

Among these components, specular reflection is view-dependent, which means that the presence of specular reflection may lead to different observations of the same point in 3D space from different views and cause ambiguities and ultimately affect the quality of geometric surface reconstruction. Therefore, we first aim to remove the specular reflection from each viewpoint, leaving only the view-independent components: ambient light and diffuse reflection. For each view, we define a set of Gaussians, termed view-dependent Gaussians (VDG), and train them to fit the specular highlights for the current viewpoint. During the evaluation and mesh extraction process, these VDG are discarded, and only the view-shared Gaussians (VSG) are involved.

Initialization of VDG. Unlike the original 3DGS, VDG are designed to represent reflections. These reflections often occur in texture-less regions, which are typically absent from the initial point cloud generated by COLMAP. To address this limitation, we use a point cloud converted from an estimated monocular depth map as our initial VDG. Specifically, we first utilize an open source instance segmentation model SAM2 [27] to obtain the mask of the car. Within the mask, a set of pixels is randomly sampled. Based on the camera’s intrinsic matrix \mathbf{K} , each sampled pixel $[x_i, y_i]$ (the pixel coordinates in the image plane) is mapped to a corresponding point $\mathbf{p}_{\text{camera},i}$ in the camera coordinate system:

$$\mathbf{p}_{\text{camera},i} = \mathbf{K}^{-1} \begin{bmatrix} x_i \cdot z_i \\ y_i \cdot z_i \\ z_i \end{bmatrix}. \quad (2)$$

During this mapping process, the depth value z_i for each sampled pixel is obtained from monocular depth estimation [39] result $z_{\text{rel},i}$, which represents relative depth. This depth is then aligned with COLMAP’s scale:

$$z_i = z_{\text{rel},i} \cdot s + o, \quad (3)$$

where s and o represent the scale and offset, respectively, obtained following the method in 3DGS [17]. The mapped points are subsequently transformed into the world coordinate system, serving as the initialization points for the VDG:

$$\mathbf{p}_{\text{world},i} = \mathbf{R} \cdot \mathbf{p}_{\text{camera},i} + \mathbf{T}. \quad (4)$$

where R and T represent the rotation matrix and translation vector, respectively, for transforming coordinates from the camera frame to the world frame.

Method	3DRealCar					Mean	Training Time
	Scene 1	Scene 2	Scene 3	Scene 4	Scene 5		
Chamfer Distance ↓							
3DGS-DR	0.156	0.120	0.147	0.169	0.120	0.142	2h30min
Gaussian Shader	0.073	0.077	0.105	0.136	0.099	0.098	1h40min
PGSR	0.115	0.064	0.127	0.142	0.095	0.109	23min
2DGS	0.134	0.104	0.133	0.190	0.115	0.135	15min
Ours	0.067	0.052	0.059	0.060	0.062	0.060	20min
Accuracy ↑							
3DGS-DR	0.314	0.460	0.348	0.343	0.457	0.384	2h30min
Gaussian Shader	0.723	0.707	0.483	0.468	0.566	0.589	1h40min
PGSR	0.473	0.794	0.550	0.550	0.635	0.600	23min
2DGS	0.576	0.675	0.541	0.314	0.537	0.529	15min
Ours	0.745	0.825	0.740	0.636	0.808	0.751	20min
F1 Score ↑							
3DGS-DR	0.443	0.596	0.464	0.464	0.571	0.508	2h30min
Gaussian Shader	0.817	0.783	0.581	0.560	0.661	0.680	1h40min
PGSR	0.567	0.830	0.609	0.644	0.713	0.673	23min
2DGS	0.687	0.768	0.625	0.401	0.629	0.622	15min
Ours	0.776	0.833	0.745	0.664	0.800	0.764	20min

Table 1. Quantitative results of reconstruction quality on the 3DRealCar dataset. ‘Red’, ‘Orange’, and ‘Yellow’ denote the best, second-best, and third-best results, respectively. Our proposed method outperforms existing mesh reconstruction techniques in terms of reconstruction quality, while also achieving relatively shorter training times.

Regularization of VDG. Training these VDG in the same manner as VSG does not yield the desired results, as these VDG may overfit the current view. To mitigate this, we incorporate opacity regularization into the loss function, ensuring that the VDG contribute to the rendered image only when necessary. Specifically, the loss is defined as:

$$\mathcal{L}_{vdg} = w_{vdg} \frac{\sum_{i=1}^{N_{vdg}} o_{vdg,i}}{N_{vdg}}, \quad (5)$$

where N_{vdg} is the total number of VDG, $o_{vdg,i}$ is the opacity of the i -th VDG and w_{vdg} is the loss weight.

3.2. Learnable Hybrid Opacity

Current GS-based methods use the same opacity parameter for rendering both appearance and geometry maps. However, for transparent objects such as glass, depth rays and appearance rays should behave differently. The depth ray should terminate at the object’s surface, whereas the appearance ray should continue to propagate forward.

To address this issue, we propose a learnable hybrid opacity (LHO), which includes two parameters: appearance opacity (sometimes simply referred to as opacity in this paper) and geometric-specific opacity. These parameters are used to separately render RGB images and geometry maps, e.g., normal and depth maps. The rendered map M is computed as follows:

$$M = \sum_{i \in N} attr_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \quad (6)$$

where $attr_i$ represents the color of the i -th Gaussian when M is an RGB map, and α_i is derived by evaluating the 2D covariance matrix Σ [44], scaled by the appearance opacity. Similarly, $attr_i$ represents the geometric attribute of the i -th Gaussian when M is a geometry map, and α_i is computed by evaluating Σ , scaled by the geometry opacity.

Based on our LHO, transparent surfaces can be modeled by learning the geometry opacity as 1 and the RGB opac-

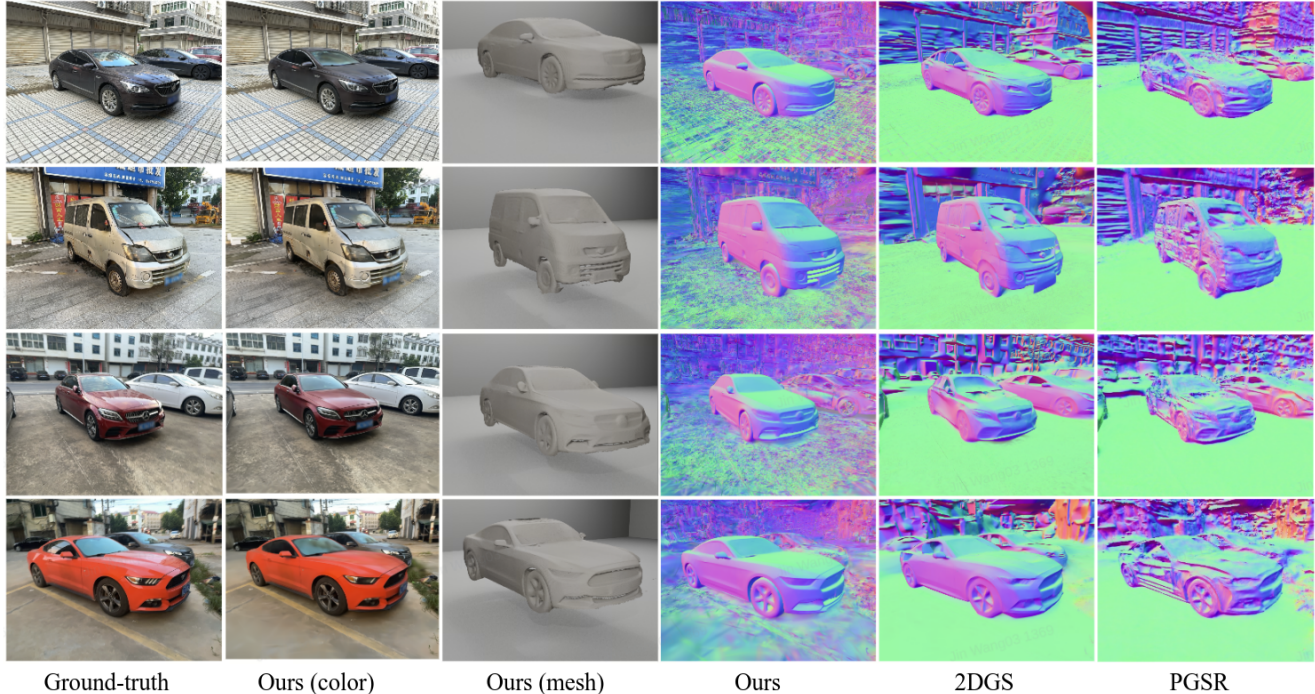


Figure 5. Visual comparisons on test-set views from the 3DRealCar dataset. Note that we focus on the reconstruction of the vehicle body rather than the entire scene. Our method excels at synthesizing geometrically accurate radiance fields and surface reconstructions, outperforming other baseline approaches in capturing sharp edges and intricate details. In contrast, baseline models often fail in areas with transparent glass and reflective surfaces on the vehicle body.

ity as 0, for example. However, such a design provides the model with greater flexibility, which could lead to overfitting. Specifically, We use a loss item similar to 2DGS to align the splats’ normal N_s with the gradients of the depth maps N_d as follows:

$$\mathcal{L}_{ds} = w_{ds}|N_d - N_s|, \quad (7)$$

where w_{ds} is the corresponding loss weight. However, when there is a discrepancy between the geometry opacity and appearance opacity, the model may learn completely transparent points at arbitrary locations to reduce the aforementioned loss terms without affecting other loss components. This would ultimately result in unpredictable reconstruction outcomes. Therefore, we constrain the geometry opacity and appearance opacity to be as consistent as possible. Specifically, we introduce another loss item:

$$\mathcal{L}_{lho} = w_{lho} \frac{\sum_{k=1}^N |o_k - geo_o_k|}{N}, \quad (8)$$

where w_{lho} is the loss weight, N is the total number of view-shared Gaussians, o_k and geo_o_k are the appearance opacity and geometry opacity respectively of the k -th Gaussian.

Note that, since our VDG are used for modeling reflections, we do not assign LHO to these Gaussians. In addition,

in the original 2DGS, the appearance opacity of each 2D primitive is reset every few iterations and may be pruned due to its small opacity. In this work, we simultaneously reset the hybrid opacity, and prune the 2D Gaussian only when both the appearance opacity and the geometry opacity are small.

3.3. Quality-aware Supervised Module

In texture-less regions, adding more views does not provide supplementary geometric information. We argue that reconstruction of these surfaces using only RGB information is an ill-posed problem. To address this, we introduce additional supervision based on surface normals.

Specifically, we employ an open-source, pre-trained normal estimation model [43] to serve as pseudo labels \hat{n} , which are used to constrain the normals of VSG. Therefore, our final normal loss function is defined as follows:

$$\mathcal{L}_{normal} = w_n|N_d - \hat{n}| + w_n|N_s - \hat{n}| + \mathcal{L}_{ds}, \quad (9)$$

where w_n is a weighting factor, which is determined dynamically by our pixel-wise quality-aware supervision module(QSM).

This quality-aware supervision module is designed to address the imperfections inherent in predictions made by pre-trained models. We observe that such errors predominantly

occur when the view ray is nearly orthogonal to the 2D Gaussian, as shown in Fig. 4 (b). Based on this observation, we calculate the angle θ between the normal vector of each 2D Gaussian and the pixel ray. If this angle value falls below a certain threshold τ , we disregard the normal supervision for that pixel, as it likely corresponds to an incorrect pseudo label. Formally, w_n is defined as:

$$w_n = \begin{cases} 1, & \text{if } \theta > \tau, \\ 0, & \text{if } \theta \leq \tau. \end{cases} \quad (10)$$

This approach ensures that the supervision is applied adaptively, enhancing the robustness of the reconstruction process in challenging regions with incorrect normal predictions, as shown in Fig. 4 (c), where it can be observed that after incorporating our proposed quality-aware supervision module, we adaptively set the constraint weight to zero for these incorrect regions, effectively mitigating the impact of erroneous priors.

3.4. Loss Functions

Our total loss function, which integrates multiple components to optimize the reconstruction process, is defined as follows:

$$\mathcal{L}_{total} = \mathcal{L}_c + \mathcal{L}_{normal} + \mathcal{L}_{vdg} + \mathcal{L}_{lho}, \quad (11)$$

where \mathcal{L}_c represents the RGB reconstruction loss. This loss combines the ℓ_1 loss and the D-SSIM term, as proposed in 3DGS [17], ensuring high-fidelity color and structural consistency in the reconstructed images.

The weighting factors for the individual loss components are set as $w_{ds} = w_n = 0.1$ in \mathcal{L}_{normal} , $w_{vdg} = 0.2$ for the VDG-related term \mathcal{L}_{vdg} , and $w_{lho} = 3.0$ for the term of LHO optimization \mathcal{L}_{lho} .

4. Experiments

4.1. Datasets and Evaluation Metrics

We conducted experiments on the 3DRealCar [9] and Ref-NeRF Real [31] datasets. The 3DRealCar dataset represents the first extensive 3D dataset of real cars, comprising over 2,500 vehicles captured under diverse lighting conditions. Each car is represented by an average of 200 dense, high-resolution 360-degree RGB-D views, facilitating high-fidelity 3D reconstruction. Ground truth data for evaluation were derived from meshes obtained using a high-precision 3D scanner. The Ref-NeRF Real dataset includes three large-scale scenes characterized by reflective surfaces, namely "sedan," "garden spheres," and "toycar." Due to the inherent challenges in accurately capturing glass surfaces with a 3D scanner, we carefully selected five scenes with reliable ground truth data to ensure robustness and fairness in the evaluation process. These scenes were labeled

as 20240710062023, 20240707060631, 20240709190817, 20240710052200, and 20240710051501, and are referred to as Scene 1, Scene 2, Scene 3, Scene 4, and Scene 5, respectively, as shown in Tab. 1. To evaluate surface quality, we utilized the Chamfer Distance metric along with Accuracy and F1-score measurements.

Note that our method primarily focuses on the geometric reconstruction of cars, with particular emphasis on accurately capturing challenging surface regions, such as highly reflective areas (e.g., glossy paint) and frequently transparent components. In addition to geometric reconstruction, our method is also capable of delivering high-quality novel view synthesis. To evaluate the quality of appearance reconstruction, we conducted experiments on the 3DRealCar and the Ref-NeRF real datasets. For the assessment of novel view synthesis, we selected three widely-used metrics: peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM), and learned perceptual image patch similarity (LPIPS) [47]. These metrics were employed to validate the fidelity and perceptual quality of the generated novel views.

4.2. Implementation Details

We implement our Car-GS with custom CUDA kernels, building upon the framework of 2DGS. Specifically, We use opacity and geometry opacity to render rgb image and geometry maps(normal/depth map), respectively. We initialize the number of VDGs for each camera perspective to 10,000 and keep it constant during the training process and discard these VDGs during test. We conduct all the experiments on a single L20 GPU.

Mesh Extraction. Following 2DGS, we render depth maps of the training views using the depth value of the splats projected to the pixels and utilize truncated signed distance fusion (TSDF) to fuse the reconstruction depth maps, using Open3D. We set the voxel size to 0.004 and the truncated threshold to 0.02 during TSDF fusion. We also extend the original 3DGS-DR to render depth and employ the same technique for surface reconstruction for a fair comparison.

4.3. Quantitative and Qualitative Results

Results on the 3DRealCar Dataset. Quantitative results on reconstruction quality for the 3DRealCar dataset are presented in Tab. 1. Compared to state-of-the-art (SOTA) 3D reconstruction methods, especially those designed to handle reflective surfaces, our proposed method demonstrates significant improvements in reconstruction quality. Furthermore, it achieves shorter training times in comparison to existing mesh reconstruction techniques. In Fig. 5, we present visual comparisons of the reconstructed normals and meshes generated by both our method and the competing methods. Notably, our approach delivers more de-

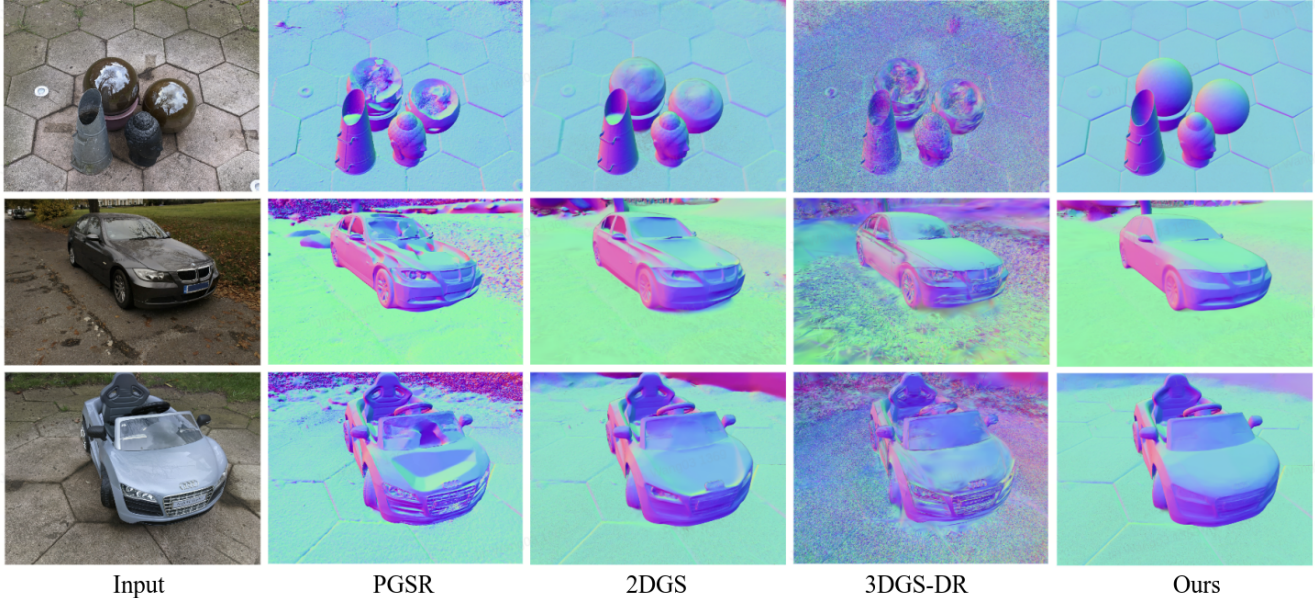


Figure 6. Visual comparisons on test-set views from the Ref-NeRF real dataset. Our method achieves superior normal estimations, particularly in regions with reflective surfaces and glass, where other approaches fail and exhibit missing or noisy surfaces.

Method	3DRealCar			Ref-NeRF			Average		
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
3DGS-DR	23.954	0.767	0.343	22.633	0.585	0.382	23.294	0.676	0.363
Gaussian Shader	23.246	0.767	0.326	23.463	0.647	0.257	23.355	0.707	0.292
PGSR	25.818	0.876	0.122	23.180	0.615	0.319	24.499	0.746	0.220
2DGS	24.608	0.818	0.242	24.307	0.679	0.279	24.458	0.749	0.261
Ours	23.982	0.803	0.258	23.843	0.664	0.287	23.913	0.734	0.273

Table 2. Quantitative comparisons of rendering quality, evaluated using PSNR, SSIM, and LPIPS metrics, on the 3DRealCar and Ref-NeRF real datasets. The results are highlighted with 'Red' indicating the best performance, 'Orange' the second-best, and 'Yellow' the third-best.

tailed mesh reconstructions and more accurate normal estimations, particularly in challenging regions such as reflective surfaces and transparent materials like glass. In contrast, existing methods often introduce substantial noise in these areas, while our approach preserves finer details.

Results on the Ref-NeRF Real Dataset. Since ground truth geometry is unavailable for the Ref-NeRF real dataset, we provide visual comparison results in Fig. 6. As observed, our method achieves robust reconstruction results in reflective and transparent glass regions, where competing methods fail. Remarkably, our approach consistently yields geometrically accurate surface reconstructions. Compared to other methods, it achieves significantly superior results in both visual quality and geometric consistency, as evidenced by higher numerical performance metrics and more realistic visual outputs. These results highlight the advantages of our method in handling challenging real-world scenarios, particularly in datasets where ground truth data may not be

accessible.

Although our primary focus is on geometric reconstruction, we also present a comparison of rendering performance. Quantitative results are provided in Tab. 2. As shown, in addition to outperforming competing methods in reconstruction quality, our approach also demonstrates superior rendering performance, further emphasizing the efficacy of our method.

Model Setting	CD \downarrow	Accuracy \uparrow	F1 Score \uparrow
w/o VDG	0.096	0.513	0.574
w/o LHO	0.072	0.581	0.604
w/o QSM	0.079	0.594	0.619
Full Model	0.060	0.636	0.664

Table 3. Ablation studies on the proposed View-Dependent Gaussians (VDG), Learnable Hybrid Opacity (LHO), and Quality-Aware Supervision (QAS) module on the 3DRealCar dataset.

4.4. Ablation Study

In this section, we conduct ablation studies on our model using the 3D real car dataset. Quantitative results are presented in Tab. 3, and qualitative visualizations are shown in Fig. 7.

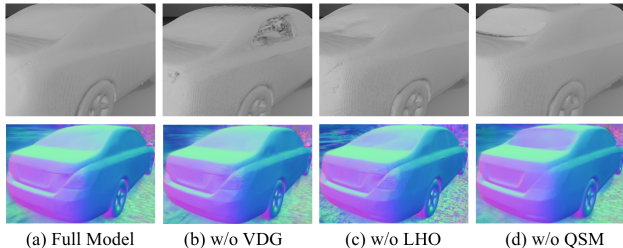


Figure 7. Rendered normal maps and reconstructed meshes on the 3DRealCar dataset, with different components removed. The best performance is achieved when all three components—View-Dependent Gaussians (VDG), Learnable Hybrid Opacity (LHO), and Quality-Aware Supervision (QAS)—are present. Omitting any of these components leads to a progressive degradation in performance.

Effectiveness of View-Dependent Gaussians. Specular reflections frequently appear on car surfaces, particularly in metallic and glass regions. These reflections cause the color of the same 3D point to vary across different viewpoints, introducing ambiguity in geometric reconstruction. By incorporating View-Dependent Gaussians (VDGs) to model specular highlights for each viewpoint, our model effectively mitigates the interference of view-dependent color variations on geometry reconstruction. As demonstrated in Tab. 3, removing VDGs leads to a significant decline in reconstruction accuracy, especially in edge and detail regions. Additionally, the visual results in Fig. 7 illustrate that models without VDGs exhibit noticeable errors and discontinuities in areas with high specular reflections, confirming the crucial role of VDGs in enhancing reconstruction precision.

Impact of Learnable Hybrid Opacity. The introduction of Learnable Hybrid Opacity assigns both geometry opacity and appearance opacity to each Gaussian, enabling more accurate modeling of complex surfaces such as transparent windows. In our ablation experiments, we evaluated the performance of the model without the learnable hybrid opacity mechanism. The results indicate that omitting hybrid opacity leads to inconsistencies between depth and color information in transparent regions, resulting in erroneous surface reconstructions. For instance, depth maps may incorrectly penetrate through glass surfaces, or there may be a mismatch between RGB opacity and geometry opacity, causing unnatural visual artifacts. Quantitative metrics in Tab. 3 show that incorporating the learnable hybrid opacity significantly improves the accuracy of depth and normal maps. Moreover, qualitative assessments in

Fig. 7 reveal that transparent areas are reconstructed with greater realism and finer details when hybrid opacity is utilized.

Effectiveness of Quality-aware Supervised Module.

Reconstructing surfaces with missing textures, such as the body of a car, poses substantial challenges due to the lack of supplementary geometric information from additional views. To address this, we introduced a Quality-aware Supervised Module that adaptively leverages ground truth normals from a pre-trained normal estimation model to provide additional constraints. In the ablation study, removing this module resulted in a significant decrease in geometry reconstruction accuracy, leading to loss of geometric details and increased inconsistencies in surface reconstruction. The Quality-aware Supervised Module selectively applies supervision based on the reliability of normal predictions, thereby preventing erroneous normals from degrading reconstruction quality. As shown in Tab. 3, the inclusion of this module reduces overall normal errors and enriches the geometric details of the reconstructed surfaces. Furthermore, Fig. 7 demonstrates that the module enhances the reconstruction quality in challenging regions with ambiguous or incorrect normal predictions.

5. Limitation and Conclusion

In this work, we presented Car-GS, a novel approach designed to tackle the challenges of reconstructing 3D models of vehicles with reflective and transparent surfaces. By introducing view-dependent Gaussians for modeling surface reflections and addressing the key limitation of shared opacity in rendering images and geometric attributes, we significantly improved the accuracy of 3D car surface reconstruction. Our method decouples geometry opacity from RGB opacity, providing a more precise representation of transparent surfaces and mitigating the ambiguity caused by specular highlights. Additionally, we incorporated pre-trained large models and proposed a pixel-wise ground truth quality-aware module, which adaptively adjusts loss weights to handle potential inaccuracies in model predictions. Through extensive experimentation on a recently released real-world vehicle dataset, we demonstrated that Car-GS not only accurately reconstructs car surfaces but also outperforms existing methods. Our results suggest that the proposed approach effectively addresses the unique challenges posed by reflective and transparent materials, offering a promising solution for applications that require high-fidelity 3D reconstructions. Future work can extend this framework to other complex materials and explore further improvements in handling varying levels of transparency and reflectivity, potentially advancing the state-of-the-art in 3D reconstruction for diverse domains.

References

- [1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5855–5864, 2021. 1
- [2] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022. 1
- [3] Wenjie Chang, Yueyi Zhang, and Zhiwei Xiong. Depth estimation from indoor panoramas with neural scene representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 899–908, 2023. 2
- [4] Anpei Chen, Minye Wu, Yingliang Zhang, Nianyi Li, Jie Lu, Shenghua Gao, and Jingyi Yu. Deep surface light fields. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 1(1):1–17, 2018. 1
- [5] 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, 2024. 3
- [6] Kai Cheng, Xiaoxiao Long, Wei Yin, Jin Wang, Zhiqiang Wu, Yuexin Ma, Kaixuan Wang, Xiaozhi Chen, and Xuejin Chen. Uc-nerf: Neural radiance field for under-calibrated multi-view cameras in autonomous driving. *arXiv preprint arXiv:2311.16945*, 2023. 2
- [7] 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 *Forty-first International Conference on Machine Learning*, 2024. 2
- [8] Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using gaussian surfels. In *ACM SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. 3
- [9] Xiaobiao Du, Haiyang Sun, Shuyun Wang, Zhuojie Wu, Hongwei Sheng, Jiaying Ying, Ming Lu, Tianqing Zhu, Kun Zhan, and Xin Yu. 3drealcar: An in-the-wild rgb-d car dataset with 360-degree views. *arXiv preprint arXiv:2406.04875*, 2024. 7
- [10] Siyuan Feng, Bohan Hou, Hongyi Jin, Wuwei Lin, Junru Shao, Ruihang Lai, Zihao Ye, Lianmin Zheng, Cody Hao Yu, Yong Yu, et al. Tensorir: An abstraction for automatic tensorized program optimization. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2*, pages 804–817, 2023. 2
- [11] Wenhao Ge, Tao Hu, Haoyu Zhao, Shu Liu, and Ying-Cong Chen. Ref-neus: Ambiguity-reduced neural implicit surface learning for multi-view reconstruction with reflection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4251–4260, 2023. 4
- [12] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. *CVPR*, 2024. 3
- [13] Wenbo Hu, Yuling Wang, Lin Ma, Bangbang Yang, Lin Gao, Xiao Liu, and Yuewen Ma. Tri-miprf: Tri-mip representation for efficient anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19774–19783, 2023. 2
- [14] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *SIGGRAPH 2024 Conference Papers*. Association for Computing Machinery, 2024. 3, 4
- [15] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pages 1–11, 2024. 3
- [16] Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. Gaussian-shader: 3d gaussian splatting with shading functions for reflective surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5322–5332, 2024. 4
- [17] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 2, 4, 7
- [18] Hai Li, Xingrui Yang, Hongjia Zhai, Yuqian Liu, Hujun Bao, and Guofeng Zhang. Vox-surf: Voxel-based implicit surface representation. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 3
- [19] 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 *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3
- [20] Junchen Liu, Wenbo Hu, Zhuo Yang, Jianteng Chen, Guoliang Wang, Xiaoxue Chen, Yantong Cai, Huan-ang Gao, and Hao Zhao. Rip-nerf: Anti-aliasing radiance fields with ripmap-encoded platonic solids. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2
- [21] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *NeurIPS*, 2020. 3
- [22] Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. Nero: Neural geometry and brdf reconstruction of reflective objects from multiview images. *ACM Transactions on Graphics (TOG)*, 42(4):1–22, 2023. 3
- [23] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20654–20664, 2024. 2
- [24] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2, 3

- [25] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022. 2
- [26] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision, 2020. 3
- [27] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. 4
- [28] Kerui Ren, Lihan Jiang, Tao Lu, Mulin Yu, Linning Xu, Zhangkai Ni, and Bo Dai. Octree-gs: Towards consistent real-time rendering with lod-structured 3d gaussians. *arXiv preprint arXiv:2403.17898*, 2024. 2
- [29] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12922–12931, 2022. 2
- [30] 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 *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025. 3
- [31] 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 *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022. 1, 3, 7
- [32] Dor Verbin, Pratul P Srinivasan, Peter Hedman, Ben Mildenhall, Benjamin Attal, Richard Szeliski, and Jonathan T Barron. Nerf-casting: Improved view-dependent appearance with consistent reflections. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–10, 2024. 3
- [33] Fangjinhua Wang, Marie-Julie Rakotosaona, Michael Niemeyer, Richard Szeliski, Marc Pollefeys, and Federico Tombari. Unisdf: Unifying neural representations for high-fidelity 3d reconstruction of complex scenes with reflections. *arXiv preprint arXiv:2312.13285*, 2023. 4
- [34] 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. *arXiv preprint arXiv:2106.10689*, 2021. 1, 3
- [35] Yaniv Wolf, Amit Bracha, and Ron Kimmel. GS2Mesh: Surface reconstruction from Gaussian splatting via novel stereo views. In *European Conference on Computer Vision (ECCV)*, 2024. 3
- [36] Daniel N Wood, Daniel I Azuma, Ken Aldinger, Brian Curless, Tom Duchamp, David H Salesin, and Werner Stuetzle. Surface light fields for 3d photography. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 487–496. 2023. 1
- [37] Liwen Wu, Sai Bi, Zexiang Xu, Fujun Luan, Kai Zhang, Iliyan Georgiev, Kalyan Sunkavalli, and Ravi Ramamoorthi. Neural directional encoding for efficient and accurate view-dependent appearance modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21157–21166, 2024. 2
- [38] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. *arXiv preprint arXiv:2201.08845*, 2022. 3
- [39] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 4
- [40] Ziyi Yang, Xinyu Gao, Yangtian Sun, Yihua Huang, Xiaoyang Lyu, Wen Zhou, Shaohui Jiao, Xiaojuan Qi, and Xiaogang Jin. Spec-gaussian: Anisotropic view-dependent appearance for 3d gaussian splatting. *arXiv preprint arXiv:2402.15870*, 2024. 2
- [41] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces, 2021. 3
- [42] Chongjie Ye, Yinyu Nie, Jiahao Chang, Yuantao Chen, Yihao Zhi, and Xiaoguang Han. Gaustudio: A modular framework for 3d gaussian splatting and beyond. *arXiv preprint arXiv:2403.19632*, 2024. 3
- [43] Chongjie Ye, Lingteng Qiu, Xiaodong Gu, Qi Zuo, Yushuang Wu, Zilong Dong, Liefeng Bo, Yuliang Xiu, and Xiaoguang Han. Stablenormal: Reducing diffusion variance for stable and sharp normal. *ACM Transactions on Graphics (TOG)*, 43(6):1–18, 2024. 2, 4, 6
- [44] Wang Yifan, Felice Serena, Shihao Wu, Cengiz Öztireli, and Olga Sorkine-Hornung. Differentiable surface splatting for point-based geometry processing. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH ASIA)*, 38(6), 2019. 5
- [45] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 2024. 3
- [46] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics (TOG)*, 43(6):1–13, 2024. 2
- [47] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 7
- [48] Wenyuan Zhang, Yu-Shen Liu, and Zhizhong Han. Neural signed distance function inference through splatting 3d gaussians pulled on zero-level set. In *Advances in Neural Information Processing Systems*, 2024. 3