TR-Gaussians: High-fidelity Real-time Rendering of Planar Transmission and Reflection with 3D Gaussian Splatting

Yong Liu, Keyang Ye, Tianjia Shao, and Kun Zhou, *Fellow, IEEE*

Abstract—We propose Transmission-Reflection Gaussians (TR-Gaussians), a novel 3D-Gaussian-based representation for high-fidelity rendering of planar transmission and reflection, which are ubiquitous in indoor scenes. Our method combines 3D Gaussians with learnable reflection planes that explicitly model the glass planes with view-dependent reflectance strengths. Real scenes and transmission components are modeled by 3D Gaussians and the reflection components are modeled by the mirrored Gaussians with respect to the reflection plane. The transmission and reflection components are blended according to a Fresnel-based, view-dependent weighting scheme, allowing for faithful synthesis of complex appearance effects under varying viewpoints. To effectively optimize TR-Gaussians, we develop a multi-stage optimization framework incorporating color and geometry constraints and an opacity perturbation mechanism. Experiments on different datasets demonstrate that TR-Gaussians achieve real-time, high-fidelity novel view synthesis in scenes with planar transmission and reflection, and outperform state-of-the-art approaches both quantitatively and qualitatively.

Index Terms—Novel view synthesis, real-time rendering, Gaussian Splatting.

I. INTRODUCTION

TRANSPARENT glass panes (e.g., windows and show-cases) are ubiquitous in indoor scenes in people's daily life, exhibiting complex combinations of transmission and reflection at their surfaces. It is of critical importance to accurately model such optical phenomena in photorealistic novel view synthesis (NVS) of indoor scenes. While there has been a rapid development of NVS in recent years, where 3D Gaussian Splatting (3DGS) [1] has emerged as the state-of-the-art method, demonstrating high rendering quality in many types of scenes, it still struggles to faithfully reconstruct indoor scenes with glass panes. The fundamental reason is, given the training images with both transmission and reflection effects, 3DGS simply overfits them using low-opacity Gaussians, without correctly distinguishing or modeling the two components separately. As a result, while training views may be fitted well, under test views the reflection often exhibits noisy artifacts (e.g., *Bookcase2* in Fig. 3) and can be completely missing in regions unobserved by training views (e.g., *Loft* in Fig. 3).

To the best of our knowledge, there has not been 3DGS-related works studying the high-fidelity novel-view rendering of transparent glass panes in indoor scenes. The most

relevant work is MirrorGaussian [2], which focuses only on mirrors with pure reflection without considering transmission. Recently several NeRF-based methods were proposed aiming to address the mixed phenomenon of transmission and reflection. They decompose the reflection and transmission either by regarding the scene as a single shell without multiple surfaces [3], or by suppressing gradient similarities between primary and reflected colors [4], which only holds on limited views or lacking robustness on textureless regions. Moreover, these methods inevitably suffer from NeRF's inherent computational inefficiency.

In this paper, we propose a novel representation of 3D Gaussians named Transmission-Reflection Gaussians (TR-Gaussians) enabling high-fidelity modeling of light transmission and reflection on transparent glass panes, which can be rendered in real time during novel view synthesis. TR-Gaussians consist of a set of primary Gaussians representing the real scene, a reflection plane representing the glass pane and modeling the view-dependent reflection strengths, and a set of glass Gaussians marking the reflective regions on the glass pane. We apply the Fresnel reflectance model on the reflection plane, where the reflection plane is associated with the learnable properties of position, orientation and base reflectance, and the reflection strength on each point of the plane is computed as the ratio of reflected light intensity relative to the incident light intensity based on these properties. The rendering of TR-Gaussians involves two passes. In the first pass, the primary Gaussians are used to render the real scene and the transmission image, and the primary and glass Gaussians are utilized together to render a reflection mask on the reflection plane marking the arbitrarily shaped reflective regions. We also use the reflection plane to compute the reflection strength map and filter it with the reflection mask. In the second pass, by leveraging the symmetry of planar reflections, we generate the mirrored Gaussians by reflecting the primary Gaussians about the reflection plane, and the mirrored Gaussians are rasterized to obtain the reflection image. The final image is obtained by blending the transmission image and reflection image based on the reflection strength map.

We design effective optimization strategies to optimize TR-Gaussians. First, to restrain the primary Gaussians from wrongly fitting the reflective regions with low-opacity Gaussians, which are commonly floating far from the real surfaces (see Fig. 2 for example), we introduce a depth variance loss to enforce the primary Gaussians to be distributed close to the surface, by minimizing the distances between the primary

Yong Liu, Keyang Ye, Tianjia Shao, Kun Zhou are with the State Key Lab of CAD & CG, Zhejiang University, Hangzhou 310058, China.
E-mail: {yongliu6, yekeyang, tjshao}@zju.edu.cn, kunzhou@acm.org.

Gaussians' center depths and the primary Gaussian rendered depths. By combining the depth variance loss with the image loss [1] and the gradient conflict loss [4], it is supposed that we can correctly decompose the reflection regions and real scenes. However, we find that relying solely on loss functions for decomposition may still get stuck in local optima. To this end, we propose a multi-stage optimization framework. In the first stage, all Gaussians are optimized together as vanilla 3DGS and then the reflection planes and glass Gaussians are initialized. In the second stage we start the joint optimization of the reflection planes along with the primary and glass Gaussians, but ignore the gradients from the reflection images to primary Gaussians so as to avoid the interference from reflection colors, because we find incorporating reflection colors together will yield wrongly placed reflection planes, which will cause the reflection cannot be correctly modeled and is degenerated to vanilla 3DGS. In the final stage, the primary Gaussians, glass Gaussians and reflection planes are jointly optimized to obtain the final image. Furthermore, we propose an opacity perturbation strategy to periodically add noise to the opacity of primary Gaussians during optimization. With this simple operation, we can simultaneously perturb both the depth variance and gradient similarity, thereby escaping from local optima and progressively refining the decomposition.

Extensive experiments on both our captured real dataset and public mirror reflection dataset demonstrate that TR-Gaussians can achieve high-fidelity novel view rendering results of indoor scenes with complex transmission and reflection, and outperforms state-of-the-art methods both quantitatively and qualitatively. In summary, our contributions include:

- We present a novel 3D Gaussian representation to model high-fidelity transmission and reflection from glass panes.
- We design effective optimization strategies to achieve high-quality decomposition of the reflection regions and real scenes.
- We provide a real-world indoor scene dataset containing common transparent glass panes with pronounced transmission-reflection effects.

II. RELATED WORKS

A. Novel View Synthesis

Given a set of calibrated images of a scene, the goal of novel view synthesis is to generate high-quality images from new viewpoints [5], [6]. NeRF [7] has become a significant technique in this field. Leveraging the continuity of MLPs and the volumetric rendering equation, NeRF can achieve end-to-end scene reconstruction and high-quality rendering in novel viewpoints. However, due to the massive MLP evaluation for every ray sample, it suffers from slow training and rendering speed. Subsequent works [8]–[14] have attempted to employ more advanced representations to tackle these problems.

Notably, a representative work 3DGS [1] represents the scene as a collection of explicit 3D Gaussian points, assigning each point opacity and spherical harmonics (SH) coefficients. With the aid of a differentiable rasterizer, 3DGS significantly enhances the training speed and enables high-quality real-time rendering at high resolution. Subsequently, there have been a lot of works focusing on improving the rendering quality of 3DGS [15]–[20]. Mip-Splatting [16] introduces a 3D smoothing filter and a 2D Mip filter to eliminate the aliasing artifacts. 3DGS-MCMC [18] rewrites the densification and pruning strategy as a deterministic state transition of MCMC samples, which effectively eliminates a lot of floaters and improves the rendering quality. These methods focus on improving quality in general scenes but do not specifically address reflection from glass. In this work, our method focuses on reconstructing the transmission and reflection from transparent glass panes, which remains a challenge for previous works.

B. Reflection Reconstruction

Reflection modeling has been extensively studied recently, with existing approaches falling broadly into two categories: physically-based modeling and representational enhancement. These approaches have been explored within both NeRF [3], [4], [21]–[30] and 3DGS [2], [31]–[35] frameworks.

Physically-based methods aim to approximate light transport governed by reflection laws. A common strategy involves using environment maps under the assumption of far-field illumination. Representative works [21]–[23], [31] jointly estimate geometry, material properties, and environment lighting using differentiable rendering techniques. For instance, GaussianShader [31] augments each Gaussian with physically meaningful attributes (e.g., diffuse reflectance, tint) and learns an environment map for efficient reflection lookup. 3DGS-DR [35] employs deferred rendering techniques to achieve accurate surface normal and a detailed environment map, which further improves the quality of reflection. However, these approaches struggle to model near-field reflections, which violate the far-field assumption, especially in indoor scenes.

To address the limitations of far-field assumptions, another line of physically-based methods simulates reflection by explicitly tracing reflected rays within the scene [4], [26]–[29], [33], [34] or introducing auxiliary radiance fields to account for reflected content [3], [30]. These ray-based techniques can handle near-field and spatially-varying reflections with higher fidelity, but often suffer from high computational cost due to the ray queries.

However, in special cases such as planar mirror surfaces, strong geometric priors allow for more efficient solutions. By exploiting the symmetry of planar reflections, Mirror-Gaussian [2] avoids full ray tracing while still preserving high-quality reflections. Specifically, it adopts a dual-rendering scheme that renders both the original scene and its mirror-transformed counterpart, achieving photorealistic mirror effects without expensive ray sampling. Nevertheless, its formulation is limited to ideal specular mirrors and cannot be generalized to complex transmission-reflection mixtures commonly found in transparent glass.

Complementary to physically-based approaches, another line of work focuses on enhancing the expressiveness of the reflection representation itself. Instead of modeling the underlying light transport, these methods directly regress reflection appearance using MLPs [24], [25], [32]. While effective in representing soft highlights and anisotropic effects, the inher-

ent smoothness of MLPs limits their ability to reproduce sharp or spatially discontinuous reflections.

Our method builds on the physically-based reflection model and adopts the planar reflection approximation that supports scenes with both planar mirror and glass-like materials. By unifying reflection and transmission modeling under a single 3DGS framework, our approach enables real-time rendering of complex transparent surfaces while maintaining photorealistic quality.

C. Transmission-Reflection Decomposition

The problem of separating transmission and reflection components has been studied extensively, both in 2D image processing and in 3D scene reconstruction. Existing methods can be broadly categorized into 2D image-based and 3D scene-based decomposition.

a) Image-based Decomposition: Traditional approaches attempt to recover the transmission and reflection layers from one or more images by exploiting various priors, such as gradient sparsity [36], depth cues [37], or gradient consistency [38]. For example, Xue *et al.* [39] utilize parallax differences between the two layers from multi-view inputs to aid the decomposition. More recent methods adopt deep learning models to perform single-image separation [40]–[44], where networks are trained in a supervised or weakly-supervised manner. Zhang *et al.* [42] propose a convolutional network using perceptual losses to remove reflection from a single image. However, how to extend image-based methods to 3D space remains underexplored.

b) 3D Scene-based Decomposition: With the advent of neural scene representations, several works have attempted to jointly model transmission and reflection in 3D space. NeRFReN [3] introduces two separate neural radiance fields for transmitted and reflected components, relying on geometric priors to enable disentanglement. However, their approach is restricted to forward-facing scenes with narrow viewpoint coverage, and fails to generalize to more unconstrained environments. MS-NeRF [30] decomposes the scene into multiple parallel feature subspaces without explicitly modeling physical properties, often leading to results that diverge from human perceptual expectations. Gao *et al.* [4] propose to suppress reflection using edge-aware regularization based on color gradient dissimilarity, but this method tends to fail in textureless regions due to the lack of structural cues.

Our method builds upon this line of work by introducing both geometric and photometric constraints for robust decomposition. Specifically, we incorporate a depth variance loss and a gradient conflict loss, which together enforce consistency in 3D space and separation in 2D image gradients. Furthermore, we introduce an *opacity perturbation* operation to escape local minima during optimization, improving robustness and convergence.

III. METHOD

In this section, we first introduce how to combine 3DGS with the reflection plane to model transmission and reflection on glass panes (Section III-A). We then introduce the regulations for the decomposition (Section III-B). Finally, we outline the details of our training process (Section III-C).

A. Transmission and Reflection Modeling

We use 3D Gaussians and reflection planes as our scene representation. As shown in Fig. 1, to synthesize novel views, the transmission image $\mathbf{C}_t$ rendered by primary Gaussians $\mathcal{G}_i$ and reflection image $\mathbf{C}_r$ rendered by mirrored Gaussians $\hat{\mathcal{G}}_i$ are blended by the the reflection strength map $\mathbf{R}$. Combining these components, we can get the final image $\mathbf{C}$ as follows:

$$\mathbf{C} = (1 - \mathbf{R}) \cdot \mathbf{C}_t + \mathbf{R} \cdot \mathbf{C}_r. \tag{1}$$

We will explain the details of each component.

1) Transmission image: Following the setting of the vanilla 3DGS [1], we model the real scene and render the transmission image with the primary Gaussians $\mathcal{G}_i$. Each Gaussian has position $\mathbf{u}_i$, opacity o_i, SH coefficients for view-dependent color $\mathbf{c}_i(\mathbf{d})$, 3D covariance $\mathbf{\Sigma}_i = \mathbf{R}_i \mathbf{S}_i \mathbf{S}_i^\top \mathbf{R}_i^\top$, where $\mathbf{S}_i$ is the scaling matrix and $\mathbf{R}_i$ is the rotation matrix.

During the rendering process, Gaussians are projected to screen space to evaluate Gaussian values G_i with $u_i' \in \mathbb{R}^2$ and $\sum' \in \mathbb{R}^{2 \times 2}$ via EWA Splatting [45]:

$$G_i(u) = exp(-\frac{(\mathbf{u} - \mathbf{u_i'})^\top \mathbf{\Sigma}_i'^{-1}(\mathbf{u} - \mathbf{u}_i')}{2}), \tag{2}$$

where $\mathbf{u}_i'$ and $\mathbf{\Sigma}_i'$ are the projected position and covariance matrix in screen space.

Then Gaussians are sorted and alpha blended into pixels to generate the transmission image $\mathbf{C}_t$:

$$\mathbf{C}_t(\mathbf{u}) = \sum_{i=1} \mathbf{c}_i(\mathbf{d}) o_i G_i(\mathbf{u}) T_i, \ T_i = \prod_{j=1}^{i-1}(1 - o_j G_j(\mathbf{u})). \tag{3}$$

We can also compute the depth map $\mathbf{D}_t$ as:

$$\mathbf{D}_t(\mathbf{u}) = \sum_{i=1} z_i o_i G_i(\mathbf{u}) T_i / \mathbf{T}_t(u), \ \mathbf{T}_t(\mathbf{u}) = \sum_{i=1} o_i G_i(\mathbf{u}) T_i. \tag{4}$$

where z_i denotes the depth of the Gaussian center.

2) Reflection strength map: To achieve realistic blending between transmission and reflection, we introduce a *reflection strength map* $\mathbf{R}(\mathbf{u})$, which modulates the contribution of the reflection image in the final composition. This map is defined as the element-wise product of two components: a view-dependent *raw reflection strength map* $\mathbf{R}_{\text{raw}}(\mathbf{u})$ computed through the reflection plane, and a *reflection mask* $\mathbf{M}(\mathbf{u})$ used to refine reflective regions:

$$\mathbf{R}(\mathbf{u}) = \mathbf{R}_{\text{raw}}(\mathbf{u}) \cdot \mathbf{M}(\mathbf{u}). \tag{5}$$

To obtain $\mathbf{R}_{\text{raw}}(\mathbf{u})$, we define a learnable rectangle *reflection plane* to model the transparent glass pane. This plane is parameterized as $(\mathbf{u}_p, \mathbf{n}_p, w, h, R_0)$, where $\mathbf{u}_p$ is the center of the plane, $\mathbf{n}_p$ is the normal vector, (w, h) denote the size, and R_0 is the base reflectance coefficient in Schlick's model, which provides an efficient formulation of the Fresnel reflection model. Given the camera viewpoint, we emit primary rays and compute their intersections with the reflection plane,
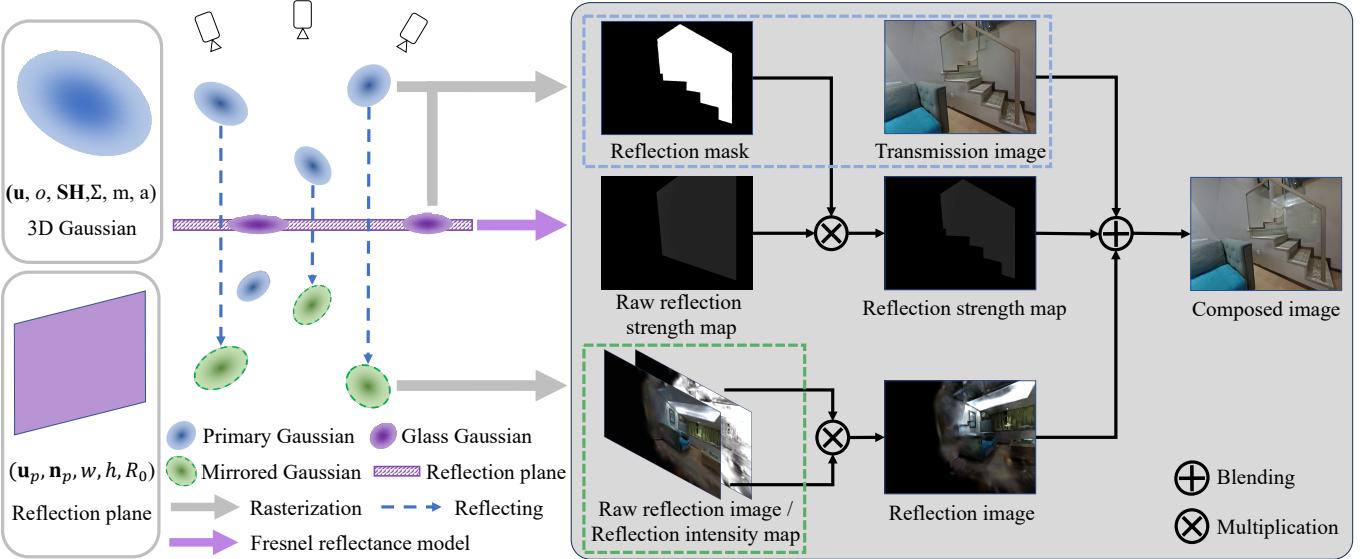
Fig. 1. Our rendering pipeline. The pipeline consists of two rendering passes. In the first pass, we render the transmission image $\mathbf{C}_t$ using primary Gaussians, and generate the reflection mask $\mathbf{M}$ by rendering both primary and glass Gaussians. We then compute the raw reflection strength map $\mathbf{R}_{raw}$ using Fresnel reflectance model, and multiply it with the reflection mask $\mathbf{M}$ to obtain the reflection strength map $\mathbf{R}$. In the second pass, we mirror the primary Gaussians across the reflection plane $\mathbf{P}$ to produce the mirrored Gaussians, which are rasterized to generate the raw reflection image $\mathbf{C}_{raw}$ and the reflection intensity map $\mathbf{A}$. These two are multiplied to produce the reflection image $\mathbf{C}_r$. Finally, we blend the transmission image $\mathbf{C}_t$ and the reflection image $\mathbf{C}_r$ using the reflection strength map $\mathbf{R}$ to produce the final image.

generating a binary hit mask $\mathbf{H}(\mathbf{u})$. For pixels where an intersection exists ($\mathbf{H}(\mathbf{u}) = 1$), the raw reflection strength is computed as:

$$\mathbf{R}_{\text{raw}}(\mathbf{u}) = R_0 + (1 - R_0)(1 - \mathbf{n}_p \cdot \mathbf{d})^5, \qquad (6)$$

where $\mathbf{d}$ denotes the view direction. For pixels with no intersection, $\mathbf{R}_{\text{raw}}$ is set to zero.

However, real-world reflective surfaces such as mirrors or glass panes are rarely perfect rectangle planes and the occlusions of foreground also needs to be considered. To handle this, we add a learnable per-Gaussian property m_i to mark the glass or mirror regions. The reflection mask $\mathbf{M}(\mathbf{u})$ is then rendered in a similar way as the transmission image:

$$\mathbf{M}(\mathbf{u}) = \sum_{i=1} m_i o_i G_i(\mathbf{u}) T_i. \qquad (7)$$

We apply a mask regularization loss $\mathcal{L}_m$ (detailed in Section III-B) to make $\mathbf{M}(\mathbf{u}) = 1$ in mirrors or glass regions and $\mathbf{M}(\mathbf{u}) = 0$ in other regions. We define Gaussians with $m_i > 0$ as *glass Gaussians*, which are only used in the rendering of $\mathbf{M}(\mathbf{u})$ to indicate glass regions and do not participate in the rendering of transmission or reflection images.

While the formulation above assumes a single reflection plane for simplicity, our method naturally generalizes to $N \geq 2$ reflection planes through iterative computation of $\mathbf{R}(\mathbf{u})$ and multi-pass rendering. Details of the multi-plane setup are provided in the supplementary material.

3) Reflection image: To render the reflection image, we need to reflect the primary Gaussians across the reflection plane to generate mirrored Gaussians, by keeping opacity and scale unchanged, and modifying position, rotation and view-dependent color. Note that we only reflect the primary Gaussians that are in front of the reflection plane.

First, the position $\hat{\mathbf{u}}_i$ of mirrored Gaussian $\hat{\mathcal{G}}_i$ is computed as:

$$\hat{\mathbf{u}}_{\mathbf{i}} = \mathbf{u}_i - 2\frac{(\mathbf{u}_i - \mathbf{u}_p)^\top \mathbf{n}_p}{\|\mathbf{n}_p\|_2}\mathbf{n}_p. \qquad (8)$$

Next, we reflect $\mathbf{R}_i$ to get the mirrored rotation matrix $\hat{\mathbf{R}}_i$. $\mathbf{R}_i$ can be represented as $[\mathbf{R}_i^1, \mathbf{R}_i^2, \mathbf{R}_i^3]$, where $\mathbf{R}_i^1$, $\mathbf{R}_i^2$, and $\mathbf{R}_i^3$ are the three principal axes of the 3D Gaussian. Therefore, symmetrizing the rotation matrix is equivalent to symmetrizing the three principal axes. And we invert $\hat{\mathbf{R}}_i^1$ to ensure that the three principal axes after reflection still remain in a right-handed coordinate system:

$$\hat{\mathbf{R}}_i = [-\hat{\mathbf{R}}_i^1, \hat{\mathbf{R}}_i^2, \hat{\mathbf{R}}_i^3]. \qquad (9)$$

$$\hat{\mathbf{R}}_i^j = \mathbf{R}_i^j - 2\frac{\mathbf{R}_i^{j\top}\mathbf{n}_p}{\|\mathbf{n}_p\|_2}\mathbf{n}_p. \qquad (10)$$

Finally, the view-dependent color $\mathbf{c}_i(\mathbf{d})$ should be adjusted accordingly. The most straightforward method is to reflect the SH coefficients across the reflection plane and then compute the color from the current viewpoint. However, mirroring SH coefficients is time-consuming. We reflect the current viewpoint across the reflection plane and compute the color $\hat{\mathbf{c}}_i(\mathbf{d})$ using the reflected view direction:

$$\hat{\mathbf{c}}_i(\mathbf{d}) = \mathbf{c}_i(\hat{\mathbf{d}}), \hat{\mathbf{d}} = \mathbf{d} - 2\frac{\mathbf{d}^\top \mathbf{n}_p}{\|\mathbf{n}_p\|_2}\mathbf{n}_p. \qquad (11)$$

Then the raw reflection image is computed as follows:

$$\mathbf{C}_{raw}(\mathbf{u}) = \sum_{i=1} \hat{\mathbf{c}}_i(\mathbf{d})\hat{o}_i\hat{G}_i(\mathbf{u})\hat{T}_i. \qquad (12)$$

Since the Fresnel-based reflection model requires colors to be in linear space, we follow [4] and introduce a reflection intensity parameter $a_i \in \mathbb{R}$ for each Gaussian to compensate

for the non-linearity caused by HDR tone mapping. Finally, the reflection image $\mathbf{C}_r(\mathbf{u})$ is computed as:

$$\mathbf{C}_r(\mathbf{u}) = \mathbf{C}_{raw}(\mathbf{u})\mathbf{A}(\mathbf{u}), \ \mathbf{A}(\mathbf{u}) = \sum_{i=1} \hat{a}_i \hat{o}_i \hat{G}_i(\mathbf{u})\hat{T}_i. \quad (13)$$

B. Regulation

1) Depth variance loss: Separating the input image into transmission and reflection components is an underconstrained problem. Especially when reconstructing a scene using the 3D Gaussians, due to its high flexibility, it tends to overfit virtual reflections behind mirrors or glass using numerous low-opacity Gaussians, leading to ambiguous geometry that becomes entangled with real geometry located behind the reflective surface, as shown in Fig. 2. Based on this observation, we propose the depth variance loss, encouraging Gaussians to form distinct surface structures. We first render a depth map $\mathbf{D}_t$ using Equ. 4 with primary Gaussians. Then we compute the depth variance loss $\mathcal{L}_d$ as follows:

$$\mathcal{L}_d = \mathbf{H}\sum_{i=1} \|z_i - \mathbf{D}_t(\mathbf{u})\|_2 o_i G_i(\mathbf{u})T_i, \quad (14)$$

where z_i is the depth of the Gaussian center and $\mathbf{H}$ is the hit mask described in Section III-A2. Multiplying by $\mathbf{H}$ avoids degrading the quality of non-reflective regions. By minimizing $\mathcal{L}_d$, the primary Gaussians are encouraged to be close to the predicted surfaces, and the reflection components are decomposed from the transmission components.

Viewpoint in front of the glass
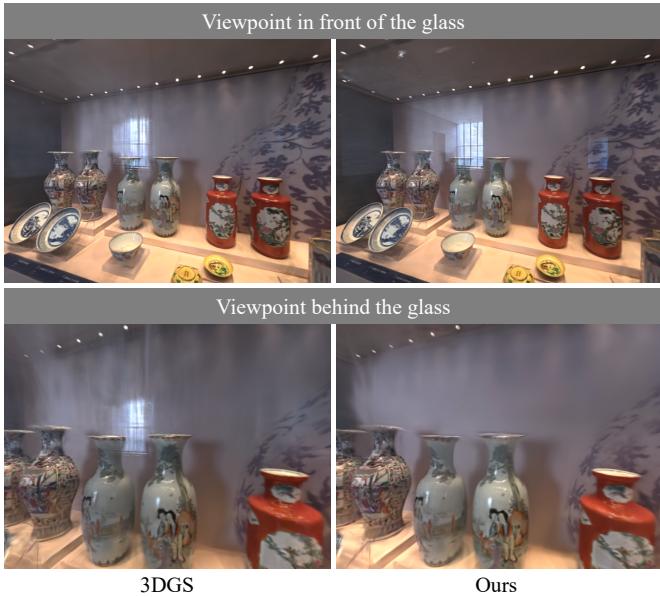
Viewpoint behind the glass

3DGS   Ours

Fig. 2. Rendering comparison of 3DGS (left column) and our method (right column) from two viewpoints — in front of the glass (top row) and behind the glass (bottom row). Our method effectively disentangles reflection and transmission components: clear reflections occur solely on the glass's front side; no reflections are visible from behind. By comparison, 3DGS fails to accurately model reflections on transparent glass, producing many floaters of "reflection ghost" far from the real surface.

2) Gradient conflict loss: Drawing inspiration from prior reflection separation works [4], [36], [39], we apply the gradient conflict regulation $\mathcal{L}_c$. It is based on the observation

that significant color gradients that appear in the reflection image are unlikely to appear in the transmission image:

$$\mathcal{L}_c = \mathbf{H}\|\nabla(\mathbf{C}_t) \cdot sg(\nabla(\mathbf{C}_r))\|_1, \quad (15)$$

where ∇ is the Sobel operator. We compute the color gradients of $\mathbf{C}_t$ and $\mathbf{C}_r$, then minimize their dot product, which selectively removes the reflection components in the transmission image. And $sg(\cdot)$ means that we stop the gradient flow from $\mathcal{L}_c$ to $\nabla(\mathbf{C}_r)$, ensuring that the reflection image is not corrupted.

3) Reflection mask loss: To constrain the reflection mask $\mathbf{M}(\mathbf{u}) = 1$ only at mirror or glass regions, we apply an L1 supervision using manually annotated reflection masks $\mathbf{M}_a$ (detailed in next section):

$$\mathcal{L}_m = \|\mathbf{M}(\mathbf{u}) - \mathbf{M}_a\|_1, \quad (16)$$

which enforces the reflection regions to stay within the annotated areas while still allowing a learning-based refinement.

C. Training

1) Reflection plane initialization: Recall that the reflection image is rendered by mirroring the primary Gaussians across to the reflection plane. A randomly initialized reflection plane leads to meaningless reflection images, making optimization infeasible. Therefore, we propose a method to initialize the reflection plane using a small number of manually annotated reflection masks.

Specifically, we annotate 5-10 reflection masks $\mathbf{M}_a$ with Segment Anything (SAM) [46]. For scenes with multiple glass panes, we annotate them with distinct class labels. Then the parameters of reflection planes can be estimated by masks and their corresponding camera poses (see supplementary materials for details). Due to the difficulty of initializing point clouds on reflective surfaces via SfM, we uniformly sample 1000 points on each reflective plane to ensure that the initial Gaussians cover the entire reflective area. We initialize these Gaussians with $m_i = 1$, while the Gaussians initialized from SfM are set to 0.

2) Loss function: The overall loss function consists of five terms:

$$\begin{aligned} \mathcal{L} = &(1 - \lambda)\mathcal{L}_1 + \lambda\mathcal{L}_{D-SSIM} \\ &+ \lambda_d\mathcal{L}_d + \lambda_c\mathcal{L}_c + \lambda_m\mathcal{L}_m, \end{aligned} \quad (17)$$

where $\mathcal{L}_1$ and $\mathcal{L}_{D-SSIM}$ are the same image losses as in 3DGS [1]. We set $\lambda = 0.2$, $\lambda_d = 0.005$, $\lambda_c = 0.2$ and $\lambda_m = 0.5$ for all scenes.

3) Opacity perturbation: We observe that, even with the aforementioned regulations, the transmission image may still retain some residual reflection components. To avoid getting stuck in such local optima, we choose to periodically add noise to the opacity of the primary Gaussians which are behind the reflection plane.

Firstly, under all training viewpoints, we project the primary Gaussians onto the reflection plane. Gaussians with the projections of their centers falling inside the plane are selected. Then we add a uniformly distributed noise in the range of $[-0.4, 0.4]$ to the opacities of the selected Gaussians and clamp the results every 1000 iterations. To avoid conflict with the

opacity clamping operation in the vanilla 3DGS, we interleave opacity-perturbation periods with opacity-clamping periods.

4) Multi-stage optimization: The optimization is divided into three stages to help transmission and reflection decomposition.

Stage 1: initialization. In the first stage, we directly adopt the transmission image as the final output $\mathbf{C} = \mathbf{C}_t$ and optimize the primary Gaussians via $\mathcal{L}_1$ and $\mathcal{L}_{D-SSIM}$.

$$\mathcal{L} = (1 - \lambda)\mathcal{L}_1 + \lambda\mathcal{L}_{D-SSIM}. \tag{18}$$

This stage follows the same pipeline as vanilla 3DGS and runs for 3,000 iterations. Also, we initialize the reflection plane $\mathbf{P}$ as described in Section III-C1 at this stage.

Stage 2: reflection plane adjustment. In the previous stage, both the reflection and transmission components were fitted by the primary Gaussians. Therefore, in this stage, we leverage the reflection image as a reference to remove "reflection ghost" from the transmission, and guide the refinement of the reflection plane. To achieve this, we enable the full rendering equation (Equ. 1) and loss functions (Equ. 17), but block the gradients from the reflection image back to the Gaussians, avoiding coupling between the optimization of the reflection image and the reflection plane. This stage also lasts for 3,000 iterations.

Stage 3: joint optimization. In the previous stage, we have obtained a reasonably accurate reflection plane. In this stage, we reduce the learning rates of the plane's position and normal (all multiplied by 0.1) to further fine-tune it, and allow full gradient propagation. Opacity perturbation is also enabled at this stage. We allocate 24,000 iterations to stage 3, maintaining the total 30,000 iterations consistent with the vanilla 3DGS.

IV. EXPERIMENTS

A. Implementation Details

Our method is implemented based on the Pytorch framework of vanilla 3DGS. The rendering process consists of two main passes. Given a camera pose, the first pass takes the primary Gaussians, glass Gaussians and the reflection plane as input and renders both C_t and R. The second pass takes the mirrored Gaussians as input to render C_r. The final result is obtained by composing these outputs. During training, we adopt the same densification strategy in 3DGS. To comprehensively validate the rendering quality improvements brought by our representation for scenes with glass or mirrors, we integrate 3DGS-MCMC [18] as an enhanced baseline for comparison. All experiments are conducted on a single NVIDIA RTX 4090 GPU.

B. Datasets

We use a self-captured dataset and a publicly available dataset for comparison.

Real transmission-reflection dataset (RTR). To the best of our knowledge, there is no existing public dataset featuring 360-degree scenes with both transmission and reflection through glass panes. Therefore, we captured six real-world scenes that include common glass windows and showcases. Five of these scenes contain a single glass pane, while one features multiple panes. Each scene consists of 100–200 images captured at a resolution of 960×720.

Mirror-NeRF dataset. We use the real-world dataset introduced by Mirror-NeRF [26] to evaluate performance under mirror reflection scenarios. This dataset includes three scenes with mirrors, each comprising 260–320 images at a resolution of 960×720.

C. Baselines and Metrics

On the RTR dataset, we compare our method and our MCMC-enhanced method with the following baselines: 3DGS [1]: vanilla 3D Gaussian Splatting without any reflection modeling; Spec-Gaussian [32]: a 3DGS-based method which utilizes an ASG appearance field to model specular and anisotropic components; EnvGS [33]: a 3DGS-based method which renders the reflection color with a ray-tracing-based renderer; Ref-NeRF [24]: a reflection-specialized NeRF-based method; NeRFReN [3]: a method employing two separate neural radiance fields for transmission and reflections (originally designed for forward-facing scenes); MS-NeRF [30]: a NeRF-based method that decomposes the scene into parallel feature subspaces; 3DGS-MCMC [18]: a 3DGS-based method that modifies the densification and pruning strategy as a deterministic state transition of MCMC samples to enhance rendering quality. On the mirror reflection dataset, we additionally include Mirror-NeRF [26]: a NeRF-based method that models reflections using Whitted-style ray tracing.

For quantitative evaluations, we select Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS) [47] as metrics. On the RTR dataset, all methods were evaluated at the same resolution of 960×720 for fair comparison. On the mirror reflection dataset, Mirror-NeRF struggles to converge at high resolutions, so we keep its original setting with a resolution of 480×360. Other methods are evaluated at a resolution of 960×720.

D. Comparison

We conducted comparisons in terms of rendering quality, decomposition result, and efficiency.

Rendering quality. As quantitatively demonstrated in Table I and Table II, our method achieves higher quality compared to the baseline approaches. And after integrating 3DGS-MCMC, our method achieves a further improvement compared to our 3DGS-based implementation and the 3DGS-MCMC baseline. The visual comparisons on the RTR dataset are shown in Fig. 3. NeRFReN and MS-NeRF produce blurry reflections due to ineffective transmission–reflection decomposition in general scenes. 3DGS exhibits a noisy mixture of transmission and reflection under novel views as seen in *Ship* and *Bookcase2*. Spec-Gaussian and EnvGS produce almost no reflections as shown in *Gundam* and *Bookcase2* because neither method can accurately model transparent glass panes from SfM initialized point clouds. In contrast, our approach, which integrates 3DGS with the explicitly defined reflection plane and the physically based reflection model, produces high-quality reflections on transparent glass surfaces. Fig. 4 also
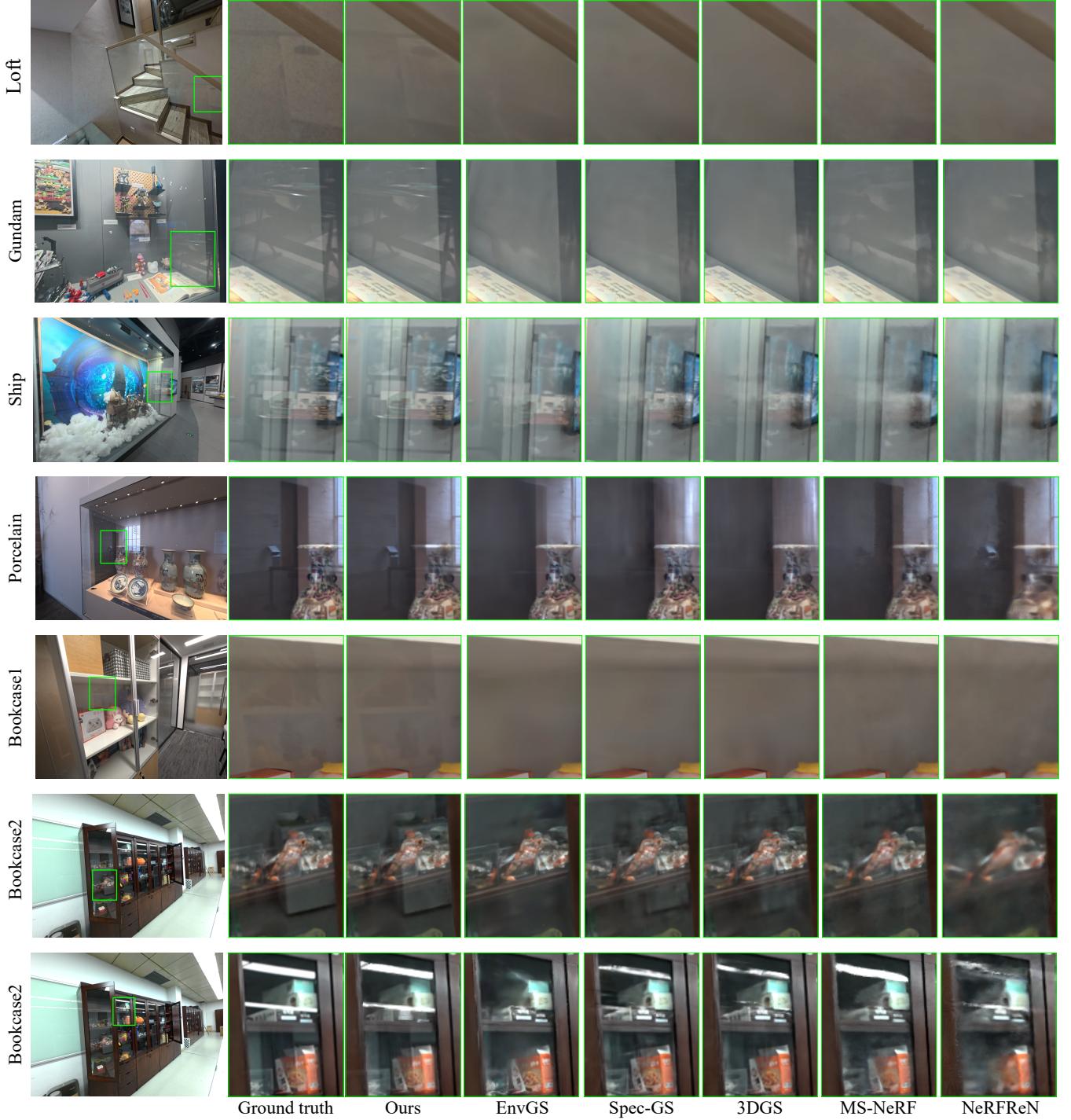
Fig. 3. Qualitative comparison with EnvGS [33], Spec-Gaussian [32], 3DGS [1], MS-NeRF [30] and NeRFReN [3] on the RTR dataset. We show images rendered from novel test viewpoints. Compared to these baselines, our method produces more accurate and detailed reflections, and yields results closest to the ground truth.

TABLE I
PER-SCENE IMAGE QUALITY COMPARISON ON THE RTR DATASET.

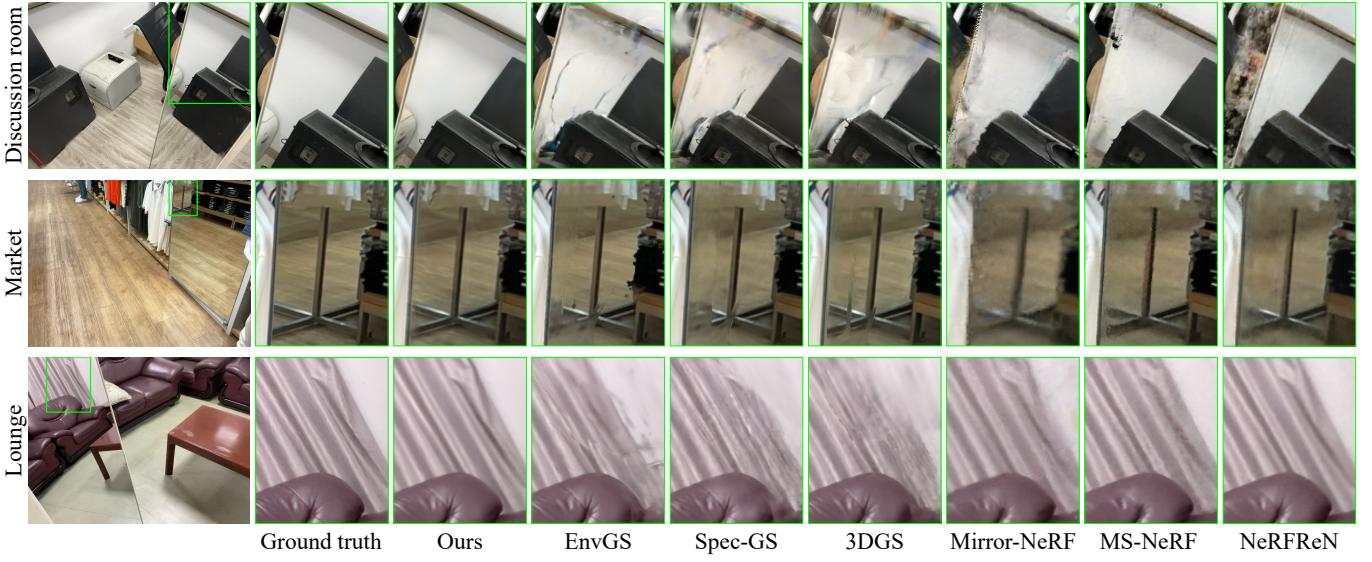| Metric	Method	Loft	Gundam	Porcelain	Ship	Bookcase1	Bookcase2	Average
PSNR ↑	Ref-NeRF	33.39	28.48	31.58	29.68	29.76	26.50	29.90
	MS-NeRF	33.42	29.20	31.68	30.72	29.71	24.94	29.94
	NeRFRen	32.29	28.66	32.19	29.67	26.28	25.97	29.18
	3DGS	34.10	28.84	31.16	31.16	30.24	25.45	30.16
	Spec-Gaussian	34.13	29.30	32.27	31.20	30.73	26.45	30.68
	EnvGS	34.25	29.69	31.99	31.11	29.63	25.77	30.41
	Ours	34.63	29.64	32.52	31.14	30.83	28.51	31.21
	3DGS-MCMC	35.09	30.60	33.68	31.75	31.76	27.48	31.73
	Ours-MCMC	35.15	30.85	33.58	32.12	32.21	28.71	32.10
SSIM ↑	Ref-NeRF	0.937	0.912	0.917	0.916	0.913	0.859	0.909
	MS-NeRF	0.957	0.928	0.940	0.940	0.940	0.894	0.933
	NeRFRen	0.951	0.923	0.934	0.919	0.885	0.847	0.910
	3DGS	0.965	0.935	0.953	0.953	0.949	0.917	0.945
	Spec-Gaussian	0.966	0.938	0.966	0.953	0.948	0.915	0.948
	EnvGS	0.962	0.937	0.959	0.948	0.942	0.910	0.943
	Ours	0.967	0.940	0.967	0.954	0.951	0.928	0.951
	3DGS-MCMC	0.968	0.950	0.967	0.960	0.957	0.927	0.955
	Ours-MCMC	0.967	0.952	0.968	0.962	0.958	0.931	0.956
LPIPS ↓	Ref-NeRF	0.262	0.256	0.254	0.252	0.262	0.288	0.262
	MS-NeRF	0.189	0.212	0.192	0.184	0.176	0.203	0.193
	NeRFRen	0.209	0.229	0.216	0.242	0.287	0.313	0.250
	3DGS	0.165	0.188	0.153	0.150	0.147	0.144	0.158
	Spec-Gaussian	0.158	0.183	0.106	0.142	0.144	0.146	0.147
	EnvGS	0.168	0.191	0.136	0.152	0.186	0.173	0.168
	Ours	0.164	0.178	0.112	0.146	0.146	0.134	0.146
	3DGS-MCMC	0.167	0.175	0.121	0.137	0.137	0.137	0.146
	Ours-MCMC	0.163	0.174	0.115	0.135	0.135	0.136	0.143

Fig. 4. Qualitative comparison on Mirror-NeRF dataset. Regions with noticeable differences are cropped and shown side-by-side for clear comparison.

shows visual comparisons on the Mirror-NeRF dataset, which focuses solely on mirror reflections. The comparison shows that our method generalizes well to pure mirror reflections and achieves competitive results compared to other approaches.

Decomposition visualization. The enhanced rendering quality of our method stems from its effective transmission-reflection decomposition. Fig. 5 illustrates our decomposition results, where the two components are accurately separated and recombined to produce high-quality novel view images. We also compare the decomposition results with NeRFReN [3], MS-NeRF [30], and EnvGS [33]. As shown in Fig. 6, NeRFReN degenerates into a single radiance field
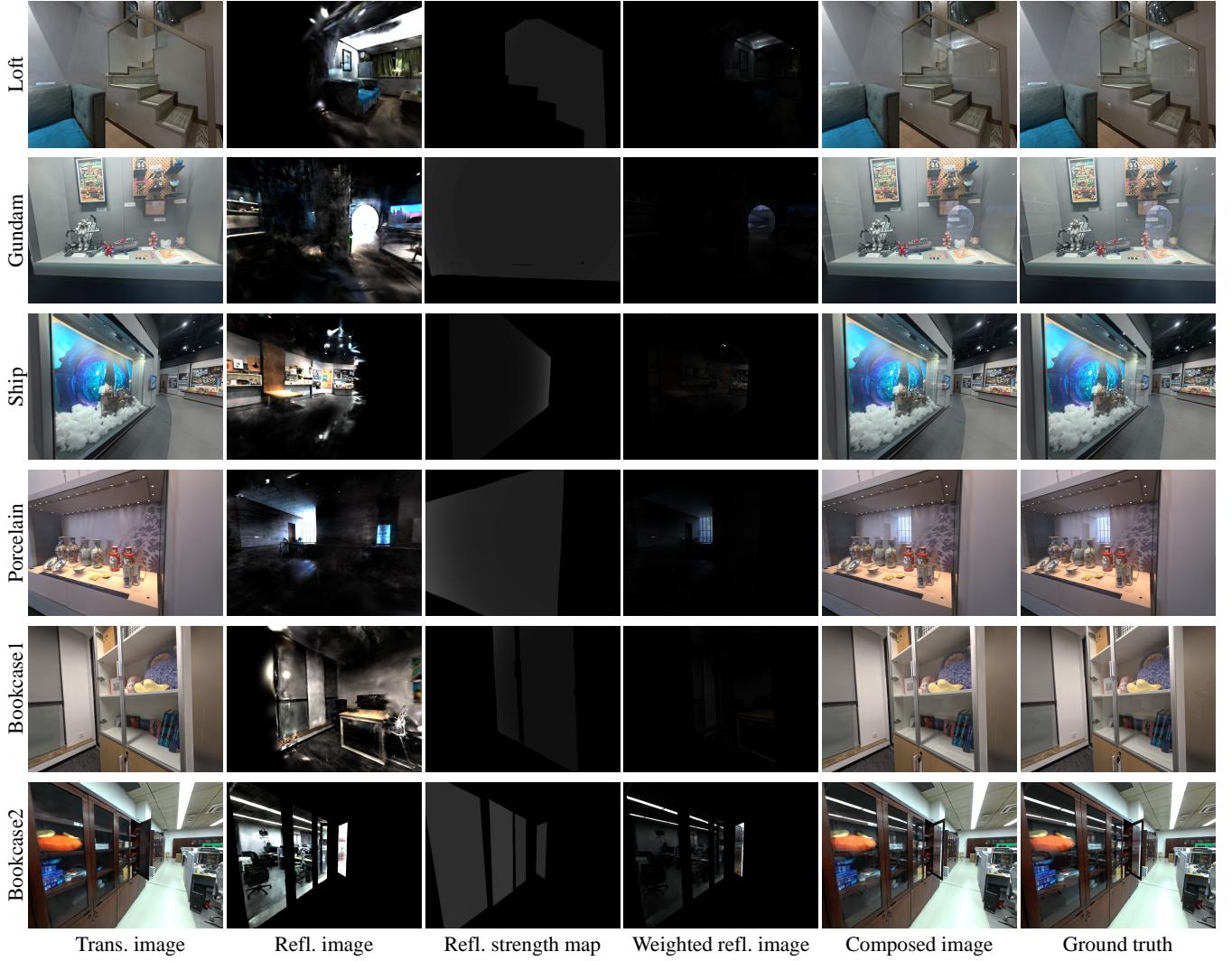
Fig. 5. Decomposition results on RTR dataset. From left to right, there are transmission image $\mathbf{C}_r$, reflection image $\mathbf{C}_r$, reflection strength map $\mathbf{R}$, weighted reflection image $\mathbf{R} \cdot \mathbf{C_r}$ and final image $\mathbf{C}$. Our method can achieve natural transmission-reflection decomposition and produce high-quality novel view synthesis results.

TABLE II
IMAGE QUALITY COMPARISON ON THE MIRROR-NERF DATASET.

	PSNR ↑	SSIM ↑	LPIPS ↓
Ref-NeRF	25.12	0.752	0.376
MS-NeRF	26.58	0.830	0.261
Mirror-NeRF	26.25	0.840	0.187
NeRFReN	25.29	0.77	0.337
3DGS	26.82	0.876	0.196
Spec-Gaussian	26.89	0.867	0.198
EnvGS	26.45	0.866	0.191
Ours	28.01	0.889	0.170
3DGS-MCMC	27.57	0.882	0.188
Ours-MCMC	29.10	0.904	0.155

because its geometric prior assumes that the reflected scene has a simple shell-like geometry, which cannot generalize to wide viewing angles. The transmission-reflection separation in MS-NeRF is not clean, with residual components leaking into each other. This is because it directly fits the image using multiple subspace radiance fields without applying any decomposition constraints. EnvGS suffers from poor reflections as it only supports tracing rays on opaque surfaces and ignores view-dependent Fresnel effects. By comparison, our method accurately decomposes transmission and reflection on glass panes.

Efficiency. Table III lists the training time, rendering speed, and number of Gaussians on RTR dataset. NeRF-based methods require hours of training and also suffer from slow rendering speeds. Among the compared methods, 3DGS serves as a baseline in terms of performance. Spec-Gaussian needs to obtain the view-dependent color of each Gaussian through an MLP, which reduces FPS from 338 to 181. EnvGS performs ray-traced reflections, leading to slow rendering (13 FPS) and long training times (2 hours). Our method has a moderate training time and achieves a rendering speed approximately two-thirds that of 3DGS. In terms of the number of Gaussians, 3DGS models reflections by adding extra Gaussians behind
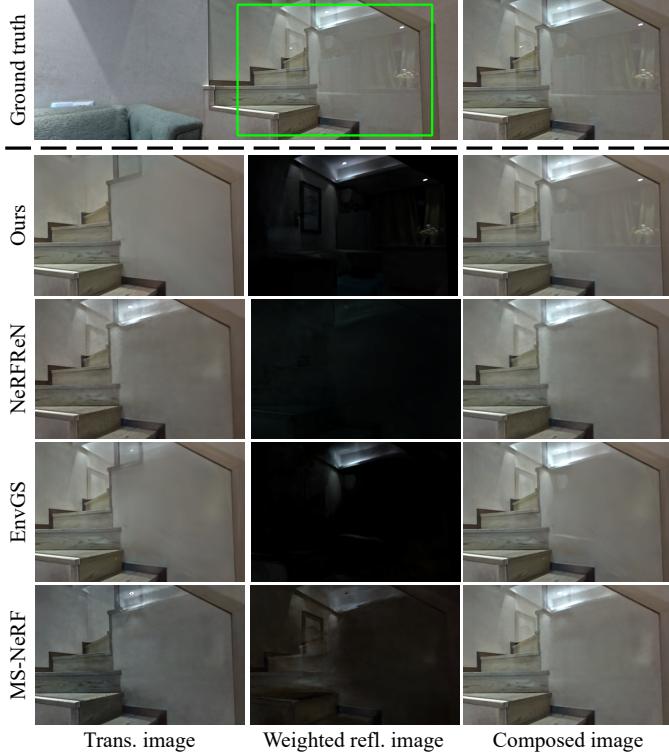
Fig. 6. Decomposition results in *Loft* compared with and NeRFReN [3], MS-NeRF [30], and EnvGS [33]. Our method achieves comparable decomposition results and better novel view synthesis quality.

TABLE III
TRAINING TIME, RENDERING SPEED (FPS), AND NUMBER OF GAUSSIANS ON THE RTR DATASET.

Method	Training Time↓	FPS↑	Gaussian Number↓
Ref-NeRF	6h	0.04	/
MS-NeRF	3h	0.06	/
NeRFReN	50h	0.04	/
3DGS	21m	338	659k
Spec-Gaussian	16m	181	568k
EnvGS	2h	13	1076k
Ours	33m	225	560k

the reflection plane, leading to a larger number of Gaussians compared to our approach. EnvGS introduces an extra set of environment Gaussians to model reflections, significantly increasing the total number of primitives.

E. Ablation Study

We conducted a series of ablation studies to validate the effectiveness of several designs of our method and demonstrate their importance. The experiments are conducted on the *Ship* and *Bookcase2* scenes. The quantitative results of ablation studies are reported in Table IV.

Regulations and opacity perturbation. We conduct ablation studies to evaluate the contributions of the regularization terms and opacity perturbation under various settings: without depth variance loss $\mathcal{L}_d$ and gradient conflict loss $\mathcal{L}_c$, with only one of the regulations, and without opacity perturbation. The

TABLE IV
ABLATION STUDY: QUANTITATIVE IMPACT OF EACH COMPONENT ON IMAGE QUALITY

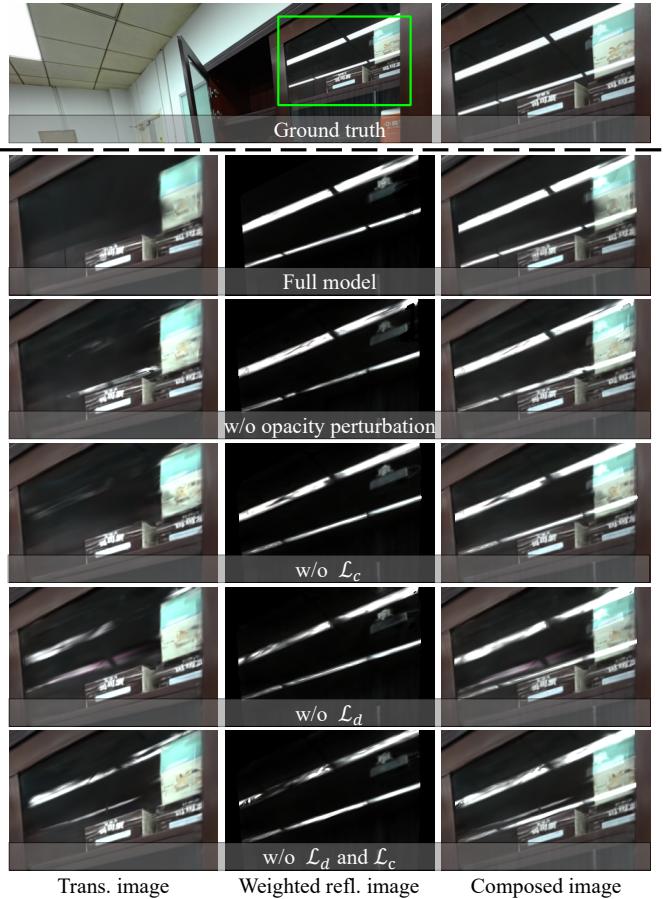| Ablations	PSNR↑	SSIM↑	LPIPS↓
w/o multi-stage optimization	28.51	0.930	0.154
w/o $\mathcal{L}_d$ and $\mathcal{L}_c$	29.33	0.938	0.144
w/o $\mathcal{L}_d$	29.53	0.937	0.145
w/o $\mathcal{L}_c$	29.56	0.938	0.143
w/o opacity perturbation	29.59	0.938	0.144
w/o reflection intensity	29.46	0.936	0.146
w/o Fresnel-based reflection	29.59	0.935	0.143
Full model	29.83	0.941	0.140

Fig. 7. Ablation results of regulations and opacity perturbation in *Bookcase2*. The removal of either component leads to degraded decomposition, highlighting the importance of their joint contribution to realistic rendering.

qualitative ablation results are shown in Fig. 7. When no regulations are applied (row 6), the transmission image contains many "reflection ghosts", leading to a noisy output image. Removing the depth variance loss $\mathcal{L}_d$ (row 5) leads to poorly constrained Gaussian geometry, causing some Gaussians to incorrectly model the reflected LED tubes which are far from the actual surface of the bookcase. Without the gradient conflict loss $\mathcal{L}_c$ (row 4), the transmission image contains needle-like floaters in areas where the gradient resembles that of the reflection image. Without opacity perturbation (row 3), the decomposition remains incomplete, with residual reflections still embedded in the transmission image, which also degrades
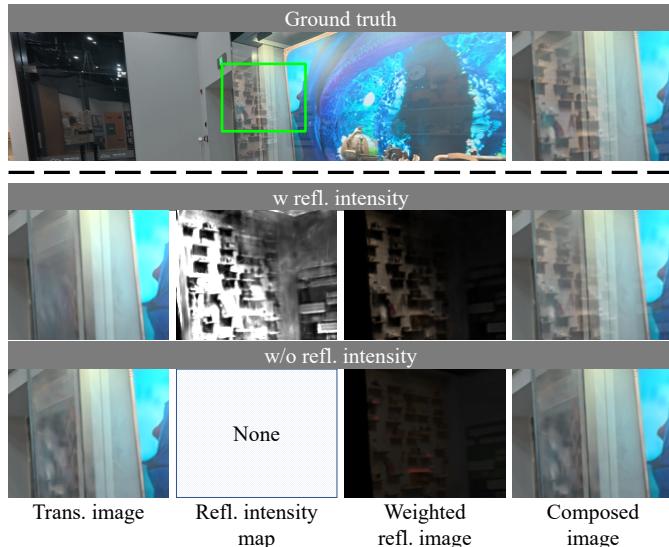
Fig. 8. Without the reflection intensity, the reflection color input to our model no longer adheres to the required linear color space, which hampers accurate reflection modeling and leads to degraded rendeirng quality.
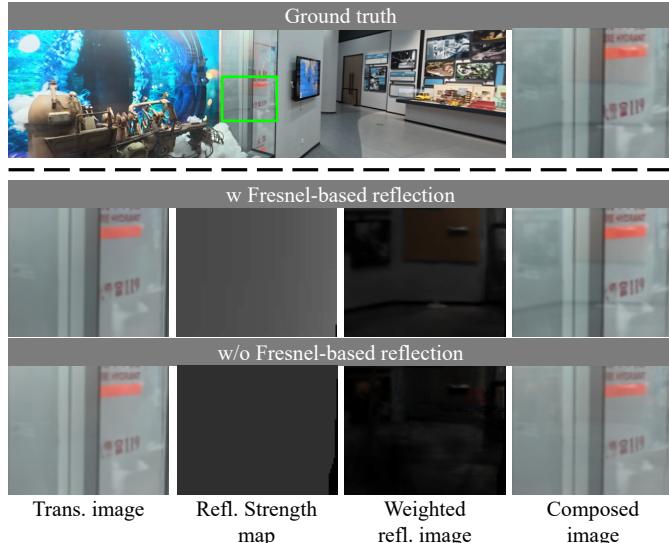

Fig. 9. Ablation results of Fresnel-based reflection in *Ship*. Without it, the model fails to capture view-dependent reflection strength, especially at grazing angles, which results in blurrier outputs.

the quality of novel view synthesis. These results demonstrate that both the proposed regulations and opacity perturbation are crucial for achieving robust decomposition and high-quality image synthesis.

Reflection intensity and Fresnel-based reflection. This ablation study is designed to validate the necessity of reflection intensity maps and Fresnel-based view-dependent reflection effects. For the former, we remove the reflection intensity map by setting $\mathbf{C}_r = \mathbf{C}_{raw}$. For the latter, we replace the Schlick reflection model with a view-independent learnable reflection coefficient $R = R_0$. The visual results are shown in Fig. 8 and Fig. 9. Without reflection intensity (row 3 in Fig 8), the reflection image fed into our model no longer conforms to the required linear color space, making it difficult to model accurate reflections. Without Fresnel-based reflection
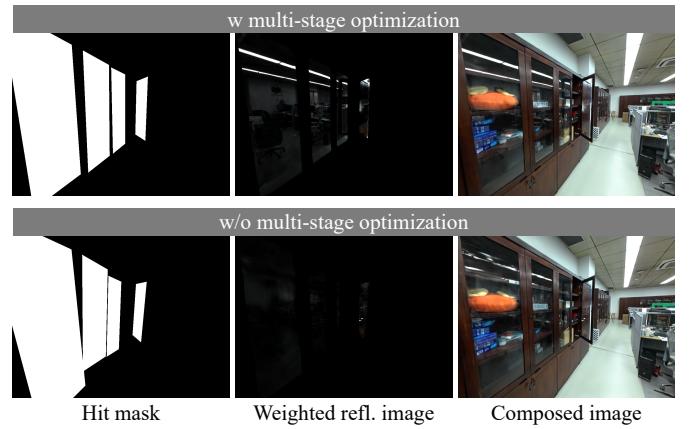

Fig. 10. Disabling multi-stage optimization leads to reflection plane drift and poor separation of reflection and transmission, demonstrating that the it is crucial for accurate reflection modeling and high-quality rendering.

(row 3 in Fig. 9), the model fails to capture the view-dependent variation in reflection strength, especially at grazing angles where reflections become more pronounced, resulting in blurrier and less realistic results. Both lead to a degradation in the final image quality.

Mutli-stage optimization. To evaluate the contribution of our multi-stage optimization, we bypass stage 2 and launch joint optimization immediately after stage 1. As Fig. 10 illustrates, this ablation causes the position and orientation of the reflection plane to drift markedly from the true glass geometry. The misalignment prevents the reflection image from fitting the reflection components in the input images, resulting in poor quality due to the failure to separate reflection from transmission.

F. Limitation and Future Work

Although our method has achieved significant progress in modeling transmission and reflection on glass panes, it has two major limitations: handling curved surfaces and multiple non-parallel planes. Our approach relies on the symmetry of planar reflection, making it inapplicable to curved surfaces. Additionally, scenes containing multiple reflection planes require iterative rendering of reflection colors for each plane, which increases training and rendering times. Future work could focus on optimizing the rendering pipeline and parallelizing rendering and optimization across multiple planes to improve efficiency.

V. CONCLUSION

In this paper, we present TR-Gaussians to enhance 3DGS in rendering scenes with transparent glass panes. By explicitly modeling both transmission and reflection through 3D Gaussians and learnable reflection planes, our method effectively captures the complex appearance of transparent glass panes with high fidelity. A multi-stage optimization framework, combined with tailored constraints and an opacity perturbation strategy, enables accurate decomposition and high-quality rendering. Extensive experiments on different datasets validate the effectiveness and efficiency of our approach.

REFERENCES

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

[2] J. Liu, X. Tang, F. Cheng, R. Yang, Z. Li, J. Liu, Y. Huang, J. Lin, S. Liu, X. Wu *et al.*, "Mirrorgaussian: Reflecting 3d gaussians for reconstructing mirror reflections," in *European Conference on Computer Vision*. Springer, 2024, pp. 377–393.

[3] Y.-C. Guo, D. Kang, L. Bao, Y. He, and S.-H. Zhang, "Nerfren: Neural radiance fields with reflections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 18 409–18 418.

[4] C. Gao, Y. Wang, C. Kim, J.-B. Huang, and J. Kopf, "Planar reflection-aware neural radiance fields," in *SIGGRAPH Asia 2024 Conference Papers*, 2024, pp. 1–10.

[5] S. J. Gortler, R. Grzeszczuk, R. Szeliski, and M. F. Cohen, "The lumigraph," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 453–464.

[6] M. Levoy and P. Hanrahan, "Light field rendering," in *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, 2023, pp. 441–452.

[7] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "Nerf: Representing scenes as neural radiance fields for view synthesis," *Communications of the ACM*, vol. 65, no. 1, pp. 99–106, 2021.

[8] A. Chen, Z. Xu, A. Geiger, J. Yu, and H. Su, "Tensorf: Tensorial radiance fields," in *European conference on computer vision*. Springer, 2022, pp. 333–350.

[9] Z. Chen, Z. Li, L. Song, L. Chen, J. Yu, J. Yuan, and Y. Xu, "Neurbf: A neural fields representation with adaptive radial basis functions," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2023, pp. 4182–4194.

[10] S. Fridovich-Keil, A. Yu, M. Tancik, Q. Chen, B. Recht, and A. Kanazawa, "Plenoxels: Radiance fields without neural networks," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5501–5510.

[11] L. Liu, J. Gu, K. Zaw Lin, T.-S. Chua, and C. Theobalt, "Neural sparse voxel fields," *Advances in Neural Information Processing Systems*, vol. 33, pp. 15 651–15 663, 2020.

[12] T. Müller, A. Evans, C. Schied, and A. Keller, "Instant neural graphics primitives with a multiresolution hash encoding," *ACM transactions on graphics (TOG)*, vol. 41, no. 4, pp. 1–15, 2022.

[13] C. Sun, M. Sun, and H.-T. Chen, "Direct voxel grid optimization: Super-fast convergence for radiance fields reconstruction," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5459–5469.

[14] Q. Xu, Z. Xu, J. Philip, S. Bi, Z. Shu, K. Sunkavalli, and U. Neumann, "Point-nerf: Point-based neural radiance fields," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 5438–5448.

[15] T. Lu, M. Yu, L. Xu, Y. Xiangli, L. Wang, D. Lin, and B. Dai, "Scaffold-gs: Structured 3d gaussians for view-adaptive rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 654–20 664.

[16] Z. Yu, A. Chen, B. Huang, T. Sattler, and A. Geiger, "Mip-splatting: Alias-free 3d gaussian splatting," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2024, pp. 19 447–19 456.

[17] Z. Fan, K. Wang, K. Wen, Z. Zhu, D. Xu, Z. Wang *et al.*, "Lightgaussian: Unbounded 3d gaussian compression with 15x reduction and 200+ fps," *Advances in neural information processing systems*, vol. 37, pp. 140 138–140 158, 2024.

[18] S. Kheradmand, D. Rebain, G. Sharma, W. Sun, Y.-C. Tseng, H. Isack, A. Kar, A. Tagliasacchi, and K. M. Yi, "3d gaussian splatting as markov chain monte carlo," *Advances in Neural Information Processing Systems*, vol. 37, pp. 80 965–80 986, 2024.

[19] Z. Yan, W. F. Low, Y. Chen, and G. H. Lee, "Multi-scale 3d gaussian splatting for anti-aliased rendering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 20 923–20 931.

[20] P. Papantonakis, G. Kopanas, B. Kerbl, A. Lanvin, and G. Drettakis, "Reducing the memory footprint of 3d gaussian splatting," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 7, no. 1, pp. 1–17, 2024.

[21] M. Boss, R. Braun, V. Jampani, J. T. Barron, C. Liu, and H. Lensch, "Nerd: Neural reflectance decomposition from image collections," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 12 684–12 694.

[22] Q. Zhang, S.-H. Baek, S. Rusinkiewicz, and F. Heide, "Differentiable point-based radiance fields for efficient view synthesis," in *SIGGRAPH Asia 2022 Conference Papers*, 2022, pp. 1–12.

[23] X. Zhang, P. P. Srinivasan, B. Deng, P. Debevec, W. T. Freeman, and J. T. Barron, "Nerfactor: Neural factorization of shape and reflectance under an unknown illumination," *ACM Transactions on Graphics (ToG)*, vol. 40, no. 6, pp. 1–18, 2021.

[24] D. Verbin, P. Hedman, B. Mildenhall, T. Zickler, J. T. Barron, and P. P. Srinivasan, "Ref-nerf: Structured view-dependent appearance for neural radiance fields," in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022, pp. 5481–5490.

[25] L. Ma, V. Agrawal, H. Turki, C. Kim, C. Gao, P. Sander, M. Zollhöfer, and C. Richardt, "Specnerf: Gaussian directional encoding for specular reflections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 21 188–21 198.

[26] J. Zeng, C. Bao, R. Chen, Z. Dong, G. Zhang, H. Bao, and Z. Cui, "Mirror-nerf: Learning neural radiance fields for mirrors with whitted-style ray tracing," in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 4606–4615.

[27] D. Verbin, P. P. Srinivasan, P. Hedman, B. Mildenhall, B. Attal, R. Szeliski, and J. T. Barron, "Nerf-casting: Improved view-dependent appearance with consistent reflections," in *SIGGRAPH Asia 2024 Conference Papers*, 2024, pp. 1–10.

[28] L. V. Holland, R. Bliersbach, J. U. Müller, P. Stotko, and R. Klein, "Tram-nerf: Tracing mirror and near-perfect specular reflections through neural radiance fields," in *Computer Graphics Forum*, vol. 43, no. 6. Wiley Online Library, 2024, p. e15163.

[29] J. Qiu, P.-T. Jiang, Y. Zhu, Z.-X. Yin, M.-M. Cheng, and B. Ren, "Looking through the glass: Neural surface reconstruction against high specular reflections," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 20 823–20 833.

[30] Z.-X. Yin, J. Qiu, M.-M. Cheng, and B. Ren, "Multi-space neural radiance fields," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 12 407–12 416.

[31] Y. Jiang, J. Tu, Y. Liu, X. Gao, X. Long, W. Wang, and Y. Ma, "Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024, pp. 5322–5332.

[32] Z. Yang, X. Gao, Y.-T. Sun, Y. Huang, X. Lyu, W. Zhou, S. Jiao, X. Qi, and X. Jin, "Spec-gaussian: Anisotropic view-dependent appearance for 3d gaussian splatting," *Advances in Neural Information Processing Systems*, vol. 37, pp. 61 192–61 216, 2024.

[33] T. Xie, X. Chen, Z. Xu, Y. Xie, Y. Jin, Y. Shen, S. Peng, H. Bao, and X. Zhou, "Envgs: Modeling view-dependent appearance with environment gaussian," *arXiv preprint arXiv:2412.15215*, 2024.

[34] N. Moenne-Loccoz, A. Mirzaei, O. Perel, R. de Lutio, J. Martinez Esturo, G. State, S. Fidler, N. Sharp, and Z. Gojcic, "3d gaussian ray tracing: Fast tracing of particle scenes," *ACM Transactions on Graphics (TOG)*, vol. 43, no. 6, pp. 1–19, 2024.

[35] K. Ye, Q. Hou, and K. Zhou, "3d gaussian splatting with deferred reflection," in *ACM SIGGRAPH 2024 Conference Papers*, 2024, pp. 1–10.

[36] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 9, pp. 1647–1654, 2007.

[37] R. Wan, B. Shi, T. A. Hwee, and A. C. Kot, "Depth of field guided reflection removal," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 21–25.

[38] Y. Li and M. S. Brown, "Exploiting reflection change for automatic reflection removal," in *Proceedings of the IEEE international conference on computer vision*, 2013, pp. 2432–2439.

[39] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman, "A computational approach for obstruction-free photography," *ACM Transactions on Graphics (TOG)*, vol. 34, no. 4, pp. 1–11, 2015.

[40] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "A generic deep architecture for single image reflection removal and image smoothing," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 3238–3247.

[41] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, "Crrn: Concurrent multi-scale guided reflection removal network," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, 2018.

[42] X. Zhang, R. Ng, and Q. Chen, "Single image reflection separation with perceptual losses," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4786–4794.

[43] K. Wei, J. Yang, Y. Fu, D. Wipf, and H. Huang, "Single image reflection removal exploiting misaligned training data and network enhancements," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 8178–8187.

[44] Y.-L. Liu, W.-S. Lai, M.-H. Yang, Y.-Y. Chuang, and J.-B. Huang, "Learning to see through obstructions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 215–14 224.

[45] M. Zwicker, H. Pfister, J. Van Baar, and M. Gross, "Ewa volume splatting," in *Proceedings Visualization, 2001. VIS'01.* IEEE, 2001, pp. 29–538.

[46] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo *et al.*, "Segment anything," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2023, pp. 4015–4026.

[47] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586–595.

APPENDIX A
DETAILS OF ESTIMATING PLANE PARAMETERS

We estimate the reflection plane from the annotated masks as follows:

1) Glass boundary extraction. For each annotated mask $\mathbf{M}_a$, we extract its edge and dilate it by 20 pixels. The resulting expanded region is defined as the glass boundary to filter candidate points.
2) Candidate selection via projection. We project the primary Gaussians onto the views with annotated mask. Only those Gaussians whose centers fall within the glass boundary are selected as candidates for plane fitting.
3) Plane fitting with RANSAC. Using the selected Gaussians, we fit the plane via RANSAC and get the plane normal $\mathbf{n}_p$. The centroid of the inlier positions found by RANSAC are then computed to define the plane center $\mathbf{u}_p$.
4) Extent determination via 2D bounding box. We project the inliers onto the estimated plane and obtain a set of 2D points. We then apply the rotating calipers algorithm to compute the minimum-area oriented bounding box (OBB). The width w and height h of this box define the spatial extent of the reflection plane.
5) Plane initialization. Finally, we initialize the reflection plane as $\mathbf{P} = (\mathbf{u}_p, \mathbf{n}_p, w, h, R_0)$. The initial base reflectance R_0 is set depending on the material: $R_0 = 0.2$ for transparent glass, and $R_0 = 1.0$ for mirrors.

APPENDIX B
RENDERING MULTIPLE REFLECTION PLANES

Our method is capable of extending to scenes with $N_p \geq 2$ reflection planes by iterative rendering. Specifically, we compute the reflection strength image $\mathbf{M}_i$, hit mask $\mathbf{H}_i$ for each plane, and reflect primary Gaussians about each plane to render reflection images $\mathbf{C}_r^i$. Then we aggregate per-plane reflection strength image $\mathbf{R}_i$ and reflection image $\mathbf{C}_i^r$ based on hit mask $\mathbf{H}_i$ to produce the composite outputs:

$$\mathbf{R} = \sum_i^{N_p} \mathbf{H}_i \mathbf{R}_i, \ \mathbf{C}_r = \sum_i^{N_p} \mathbf{H}_i \mathbf{C}_r^i. \tag{19}$$

APPENDIX C
EXTENSION WITH 3DGS-MCMC

We integrate 3DGS-MCMC [18] as an enhanced baseline into our framework to further improve rendering quality. Specifically, we substitute the densification process in vanilla 3DGS [1] with the relocation strategy from 3DGS-MCMC [18] and integrate its opacity and covariance regulation into our loss function:

$$\begin{aligned} \mathcal{L} = &(1 - \lambda)\mathcal{L}_1 + \lambda\mathcal{L}_{D-SSIM} \\ &+ \lambda_d\mathcal{L}_d + \lambda_c\mathcal{L}_c + \lambda_m\mathcal{L}_m \\ &+ \lambda_o\mathcal{L}_o + \lambda_{\sum}\mathcal{L}_{\sum}, \end{aligned} \tag{20}$$

where we set $\lambda_o = 0.01$ and $\lambda_{\sum} = 0.01$, following the original paper. 3DGS-MCMC also requires setting a maximum capacity for the number of Gaussians. For fair comparison, we set it to match the number of Gaussians obtained by our original method.

APPENDIX D
DECOMPOSITION RESULTS ON MIRROR-NERF DATASET

We further present qualitative decomposition results on the Mirror-NeRF [26] dataset, as shown in Fig. 11. Our method accurately separates reflection components in scenes with mirrors. This demonstrates that our approach is not limited to transparent glass panes, but also generalizes well to purely specular reflections. This capability stems from our Fresnel-based reflection model, which explicitly accounts for both transmission and reflection phenomena in a unified framework.

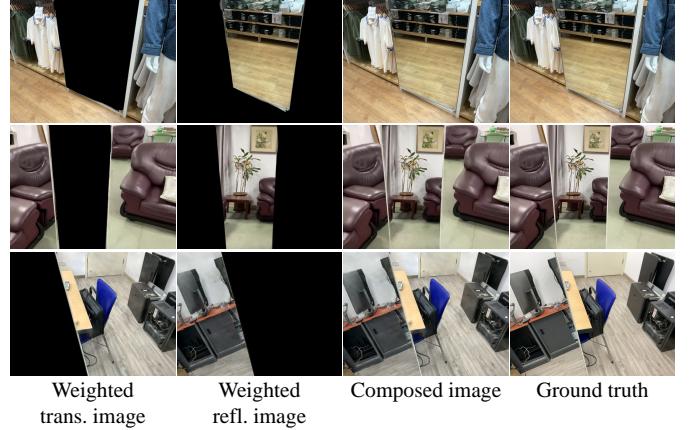| Weighted trans. image | Weighted refl. image | Composed image | Ground truth |

Fig. 11. Decomposition results on Mirror-NeRF dataset. From top to bottom: Market, Lounge, and Discussion room. From left to right: weighted transmission image $\mathbf{C}_t \cdot (1 - \mathbf{R})$, weighted reflection image $\mathbf{C}_r \cdot \mathbf{R}$, composed image $\mathbf{C}$, and ground truth.

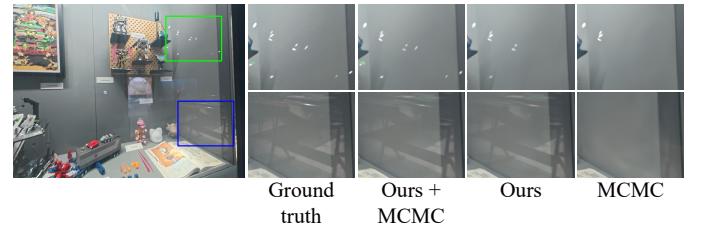## APPENDIX E
ABLATION STUDY ON INTEGRATION WITH 3DGS-MCMC

| Ground truth | Ours + MCMC | Ours | MCMC |

Fig. 12. From left to right: ground truth, our method integrated with 3DGS-MCMC, our method and 3DGS-MCMC.

To validate the generalizability of our representation, we integrate the 3DGS-MCMC [18] baseline into our TR-Gaussians pipeline to further enhance image quality. The qualitative results are shown in Fig. 12. As 3DGS-MCMC does not explicitly model reflections, it exhibits similar limitations to the vanilla 3DGS: reflection components are either missing in novel viewpoints (first row) or appear blurred (second row). In contrast, our method incorporates an explicitly defined reflection plane and a Fresnel-based reflection model into the 3DGS framework, enabling the synthesis of high-fidelity reflections on transparent glass surfaces. Furthermore, by adopting the densification and pruning strategy from 3DGS-MCMC, our method can capture finer reflection details, such as the complete reconstruction of multiple reflected light bulbs (first row), which are only partially recovered by other

baselines. These results demonstrate that our representation can be seamlessly integrated into advanced 3DGS variants like 3DGS-MCMC, leading to enhanced reconstruction quality and more accurate reflection modeling.