Opti-NeuS: Neural Reconstruction for Dual-Layered Transparent and Opaque Objects
Supplementary Material

Yi Yang[1] Gaoyang Zhang[1] Jun Tan[1] Xinguo Liu[1†]
[1]Zhejiang University, China
yiy@zju.edu.cn blurgy@zju.edu.cn tanj08@zju.edu.cn xinguoliu@zju.edu.cn

1. Proof of Equation.

1.1 Proof of Equation (8)

We aim to derive the second derivative of the rendering weight $w(t)$. Recall Eq. (8) in the main paper:

$$w''(t) = \frac{d^2 w(t)}{dt^2} = -s\, f''(t) \cdot \Phi_s'(f(t)) \cdot T(t), \qquad \textbf{(8*)}$$

Step 1: Second derivative of $w(t)$. The rendering weight is $w(t) = T(t) \cdot \rho(t)$, where $T(t) = \exp(-\int_0^t \rho(u)\, du)$. Following NeuS [4], $T(t)\rho(t) = -\frac{dT}{dt}(t)$. Applying the product rule and substituting:

$$\begin{aligned} w'(t) &= T(t)\big[\rho'(t) - \rho^2(t)\big], \\ w''(t) &= T(t)\big[\rho''(t) - 2\rho(t)\rho'(t)\big] \\ &\quad + T'(t)\big[\rho'(t) - \rho^2(t)\big]. \end{aligned} \tag{1-1}$$

At local minima points $w'(t) = 0$, since $T(t) > 0$, we have $\rho'(t) = \rho^2(t)$, so the second term vanishes. To simplify the calculation, we retain the dominant contribution from $\rho''(t)$:

$$w''(t)\big|_{w'=0} \approx T(t)\, \rho''(t). \tag{1-2}$$

In our real implementation, the experiments have shown that this approximation does not have an impact on the reconstruction quality.

Step 2: Second derivative of $\rho(t)$. Following HF-NeuS [5], the volume density along the ray is:

$$\rho(t) = s\,(1 - \Phi_s(f)), \tag{1-3}$$

where $\Phi_s(x) = (1 + e^{-sx})^{-1}$ is the cumulative distribution function of the logistic distribution, f denotes the SDF value at point $p(t)$, and θ is the angle between the ray direction and the surface normal.

Differentiating Eq. (2) along the ray via the chain rule:

$$\rho''(t) = -s\left[\Phi_s''(f(t)) \cdot (f'(t))^2 + \Phi_s'(f(t)) \cdot f''(t)\right] \tag{1-4}$$

At the actual surface, $f(t_{surface}) = 0$ and $f'(t_{surface}) \approx 0$ (the SDF reaches a local minima along the ray), so the first term vanishes, leaving the leading term:

$$\rho''(t)\big|_{\text{surface}} = -s\, \Phi_s'(f(t)) \cdot f''(t). \tag{1-5}$$

Substituting Eq. (1-5) into Eq. (1-2), and noting that $w''(t) = -w''(t)$ is defined as a positive curvature measure (since $w'' < 0$ at a maximum), we obtain Eq. (8).

1.2 Proof of Equation (9)

Recall Eq. (9) in the main paper:

$$\mathcal{S}(f) = \frac{w''(t)}{w_{\max}(t)} = \frac{2s\, e^{-sf}}{(1 + e^{-sf})^2} \cdot f''. \qquad \textbf{(9*)}$$

To obtain a scale-invariant measure, we normalize $w''(t)$ by the peak magnitude $w_{\max}(t)$. Since $T(t)$ varies slowly near the surface, we have $w_{\max} \approx T(t) \cdot \rho_{\max}$. Evaluating $\rho(t) = s(1 - \Phi_s(f))$ at $f = 0$ gives the peak density:

$$\rho_{\max} = s\,(1 - \Phi_s(0)) = \frac{s}{2}, \tag{1-6}$$

so that $w_{\max} \approx \frac{s}{2}\, T(t)$. Substituting Eq. (1-2) and expanding $\Phi_s'(f) = \frac{se^{-sf}}{(1+e^{-sf})^2}$:

$$\mathcal{S}(f) = \frac{s\, f'' \cdot \Phi_s'(f) \cdot T(t)}{\frac{s}{2}\, T(t)} = \frac{2s\, e^{-sf}}{(1 + e^{-sf})^2} \cdot f'', \tag{1-7}$$

which is Eq. (9) in the main paper.

2. Outer errors propagation in Stage II.

The reconstruction in Stage I inevitably contains minor geometric errors, but these have little impact to Stage II. We

†Corresponding author.

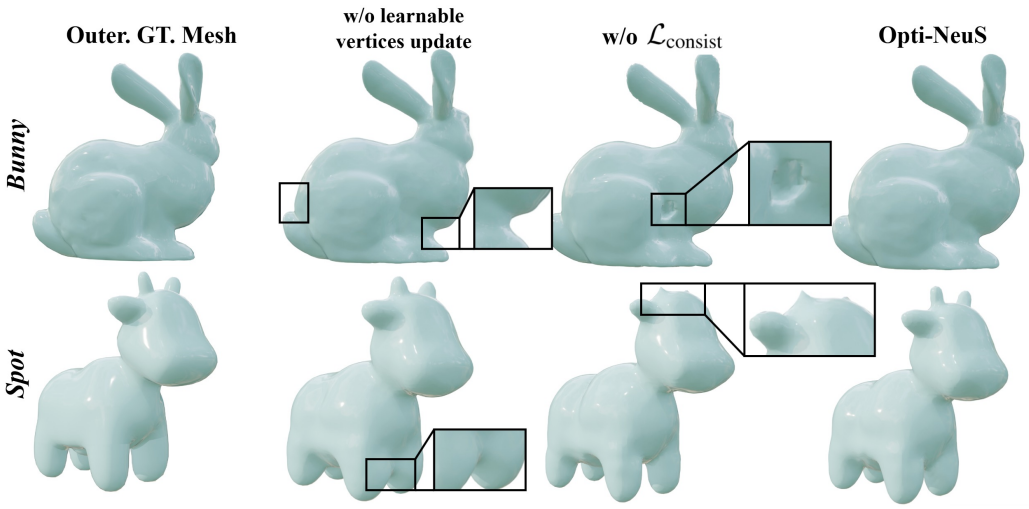

Figure 1. **Ablation of learnable vertices update and L_{consist}.** Qualitative reconstruction comparison.

Algorithm 1 Maximum Bounding Box Outer Surface Extraction

$\mathcal{M} \leftarrow$ Marching Cubes$(f, \tau = 0)$
$\{C_i\}_{i=1}^{n} \leftarrow$ Connected Components$(\mathcal{M})$
$V_{max} = 0$, $C_{outer} = \emptyset$
for $i = 1$ to n **do**
  $V_i = \prod_{j=1}^{3}(b_{max}^{j} - b_{min}^{j})$
  if $V_i > V_{max}$ **then**
    $V_{max} = V_i$, $C_{outer} = C_i$
  end if
end for
$\mathbf{d}_k = \mathbf{f}_k - \mathbf{c}_{mass}$ for face centers $\mathbf{f}_k$
$\alpha = \frac{1}{|F|}\sum_{k=1}^{|F|} \mathbf{n}_k \cdot \mathbf{d}_k$
if $\alpha < 0$ **then**
  $C_{outer} = \arg\max_{C_j \neq C_{outer}} V_j$
end if
return C_{outer}

address this through serving vertices as learnable parameters to update, and using maximum bounding box algorithm to eliminate floaters(described in Algorithm 1.). First, We operate joint optimization in Stage II where outer vertices serve as learnable parameters and are fine-tuned via refractive rendering loss. For convergence stability, we utilize gradient accumulation and adaptive update schedule. Second, we use the maximum bounding box algorithm to extract the most complete surface from Stage I, which eliminates floaters. If above methods are not applied, the geometry will suffer from significant degeneration. We demonstrate the ablation experiment in Figure 1 and supplement it in the revision. We demonstrate the ablation experiment in Fig. 1 and supplement it in the revision.

3. Effectiveness of L_{consist}.

Without L_{consist}, the IoRNetwork could predict spatially arbitrary IoR, leading to ambiguity: the true IoR and multiple incorrect IoR can both achieve the same photometric effect through different refraction paths. As shown in Fig. 1, L_{consist} addresses this ambiguity by prompting spatial coherence. Real-world transparent materials exhibit smooth IoR variations rather than abrupt variations. What's more, L_{consist} enforces continuity in the same materials, not across boundaries. Eq. 15 contains a featrue similarity component $\frac{1+\cos(f_i, f_j)}{2}$, which becomes small when geometric normals vary significantly. At air-transparent surfaces, the normal discontinuity and ∇SDF changes cause low similarity.

4. Intuition and Validation of IoRNentwork.

The intuition follows Snell's law: refraction ray direction is determined by the normal and IoR (Eq. 17). Given outer surface, only correct IoR can trace the ray to the true background color(supervised by the frozen background Network). Wrong IoR lead to significant rendering penalties, providing a strong gradient signal for optimization.

As for 3-dimensional inputs: IoR is only dependent on coordinates x and is optimized by viewing directions ω and normals n. Since we rely on the background Network for supervision instead of GT IoR, every refraction direction (computed by IoR, ω and n) must correspond to the observed background color from its ω, if not it will call the network to optimize current IoR. We demonstrate the prediction of IoRNetwork as Fig. 2: IoRNetwork can learn meaningful and stable IoR. When facing a special scene: *two half-ball* with abrupt IoR variations, IoRNetwork can not learn the correct IoR distribution (the same in NU-NeRF).

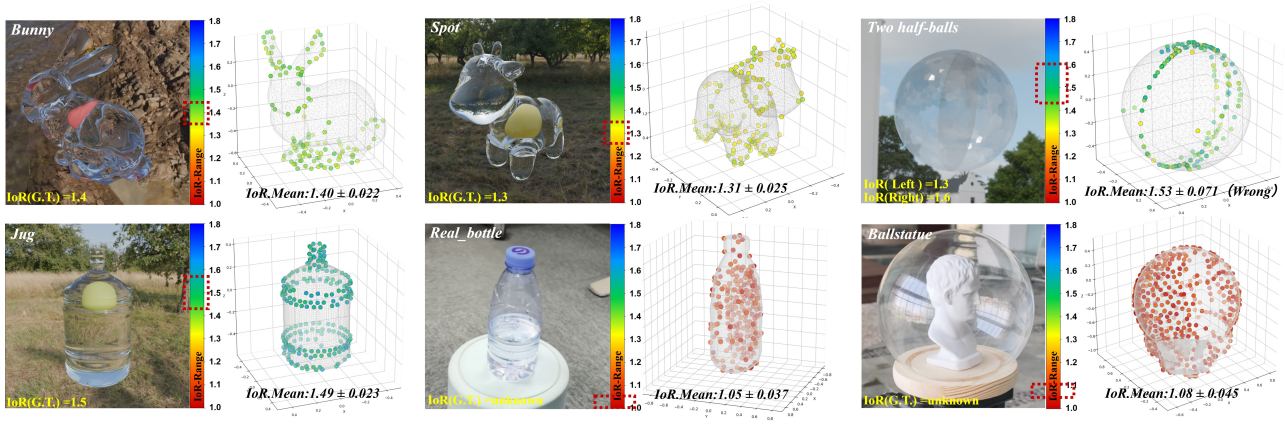

Figure 2. **Visualization and evaluation of IoRNetwork prediction.** IoRNetwork predicts IoR at keypoints of transparent surfaces.

5. Difference from previous works.

1) Opti-NeuS uses adaptive α to reconstruct high-fidelity transparent surfaces. Compared to Alpha-NeuS's fixed iso-threshold for surface extraction, we extract unbiased surfaces through sharpness $\mathcal{S}$. which represents a class of methods for decoupling complex objects.
2) In the two-stage strategy, Opti-NeuS focuses on decoupling optical-spatial scenes, aiming to address optical distortion, while NU-NeRF[2] focuses on decoupling material and appearance, aiming for progressive refinement. The two-stage strategy designed for decoupling complex objects had been proposed in NeRO[1] and ReNeuS[3].
3) Opti-NeuS uses IoRNetwork to enable stable ray tracing, While NU-NeRF directly predicts refractive color without explicit ray tracing. while NU-NeRF skips explicit ray tracing and uses MLPs to predict refractive color. In the main paper's Fig. 6-*Magician-box*, when facing semi-enclosed object, NU-NeRF reconstructs the wrong surface because of incorrect color prediction.

6. More Outer Transparent Surfaces.

Our two-stage strategy is a hierarchical decoupling method that can be recursively applied. We treat each inner region as a new sub-scene, enabling nested decoupling for multi-layered objects. In theory, as long as the transparent shells can be distinguished from the input images, Opti-NeuS can extend the Stage to III because of each independent Stage training. In the experiment, we generate a dataset by blender. Finally we get complete triple-layered objects. But every additional layer accumulates geometric errors inevitably, Opti-NeuS cannot address these scenes like stages beyond III and the shell with complicated shape. We qualitatively demonstrate this on a synthetic scene shown in Fig. 3, where Opti-NeuS is able to reconstruct high-quality surfaces across 3 layers. We also find that Opti-NeuS cannot address scenes such as training stages beyond III and the shells with complicated shape.

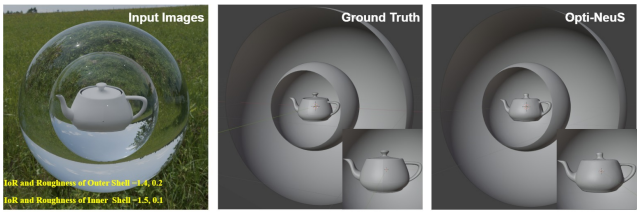

Figure 3. **Visualization of two transparent shells reconstruction.** Opti-NeuS can separately operate refraction ray tracing twice to reconstruct transparent surfaces.

References

[1] 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. In *SIGGRAPH*, 2023. 3

[2] Jia-Mu Sun, Tong Wu, Ling-Qi Yan, and Lin Gao. Nu-nerf: Neural reconstruction of nested transparent objects with uncontrolled capture environment. *ACM Transactions on Graphics (TOG)*, 43(6):1–14, 2024. 3

[3] Jinguang Tong, Sundaram Muthu, Fahira Afzal Maken, Chuong Nguyen, and Hongdong Li. Seeing through the glass: Neural 3d reconstruction of object inside a transparent container. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12555–12564, 2023. 3

[4] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, 2021. 1

[5] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. Hf-neus: improved surface reconstruction using high-frequency details. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2022. 1