Ref-NeuS: Ambiguity-Reduced Neural Implicit Surface Learning for Multi-View Reconstruction with Reflection

Wenhang Ge[1,2]   Tao Hu[4]   Haoyu Zhao[1]   Shu Liu[5]   Ying-Cong Chen[1,2,3,*]

[1]HKUST(GZ)   [2]HKUST(GZ)-SmartMore Joint Lab   [3]HKUST   [4]CUHK   [5]SmartMore

Abstract

Neural implicit surface learning has shown significant progress in multi-view 3D reconstruction, where an object is represented by multilayer perceptrons that provide continuous implicit surface representation and view-dependent radiance. However, current methods often fail to accurately reconstruct reflective surfaces, leading to severe ambiguity. To overcome this issue, we propose Ref-NeuS, which aims to reduce ambiguity by attenuating the effect of reflective surfaces. Specifically, we utilize an anomaly detector to estimate an explicit reflection score with the guidance of multi-view context to localize reflective surfaces. Afterward, we design a reflection-aware photometric loss that adaptively reduces ambiguity by modeling rendered color as a Gaussian distribution, with the reflection score representing the variance. We show that together with a reflection direction-dependent radiance, our model achieves high-quality surface reconstruction on reflective surfaces and outperforms the state-of-the-arts by a large margin. Besides, our model is also comparable on general surfaces.

1. Introduction

3D reconstruction is a crucial task in computer vision that serves as the foundation for multiple fields such as computer-aided design [20, 8], computer animation [33, 23], and virtual reality [40]. Among various 3D reconstruction techniques, image-based 3D reconstruction is particularly challenging, which aims to recover 3D structures from posed 2D images. Traditional multi-view stereo (MVS) approaches [11, 39, 51] generally require a multi-step pipeline with supervision, which can be cumbersome. Recently, neural implicit surface learning [45, 52, 31] has gained increasing attention due to its ability to achieve remarkable reconstruction quality with a neat formulation that supports end-to-end and unsupervised training. However, as illus-
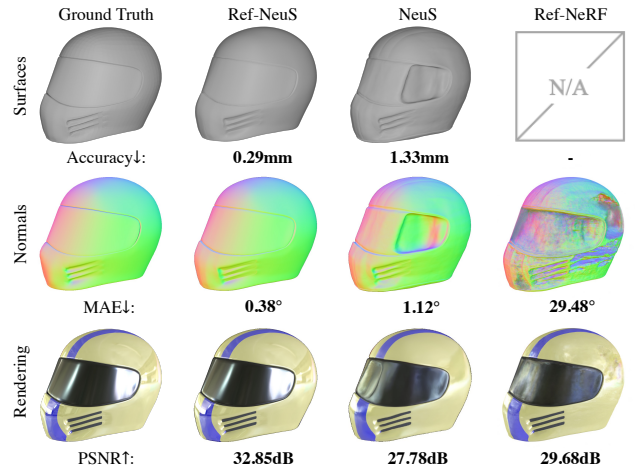

Figure 1. Our ambiguity-reduced framework significantly improves explicit surfaces geometry, surface normals and rendering realism for modeling reflective surfaces compared to NeuS and Ref-NeRF. Reconstruction accuracy (lower is better), surface normal mean angular error, i.e., MAE (lower is better) and PSNR (higher is better) are compared.

trated in Fig. 1, existing methods tend to produce erroneous results in reflective surfaces. Since these approaches infer geometry information with multi-view consistency, which is compromised due to the ambiguous surface prediction on reflective surfaces. As a result, their practicality becomes limited in scenarios where reflection is unavoidable.

Several recent works [3, 42, 55, 57, 43] have investigated reflection modeling in neural radiance fields. These methods typically decompose the appearance of an object into several physical components, allowing reflection to be explicitly represented. This can lead to a better estimation of the 3D geometry by eliminating the influence of the reflection component. Nevertheless, physical decomposition can be highly ill-posed [14], and inaccurate decomposition can significantly constrain performance. For example, as shown in Fig. 1, the predicted normals in Ref-NeRF [43] are not accurate enough, which leads to suboptimal performance.

---

*Corresponding author.

In this paper, we propose a simple yet effective solution that does not rely on challenging physical decomposition. Instead, we suggest reducing ambiguity by introducing a reflection-aware photometric loss that adaptively lowers the weights fitting reflective surfaces based on the reflection score. By doing so, we avoid devastating multi-view consistency. Additionally, inspired by Ref-NeRF [43] and Neural-Warp [7], we show that we can further improve the geometry by substituting radiance dependency with reflection direction to obtain a more accurate radiance fields. As demonstrated in Fig. 1, our model outperforms other competitive approaches in predicting surface geometry (top row) and surface normals (middle row). Besides, by estimating more accurate surface normals that determine the accuracy of reflection direction, we can also achieve promising rendering realism as an added benefit (bottom row).

Although the idea discussed above is simple, designing the reflection-aware photometric loss is non-trivial. One straightforward approach is to follow the NeRF-W [27], in which the vanilla photometric loss is extended to a Bayesian learning framework [19]. It formulates radiance as a Gaussian distribution with the learned uncertainty representing the variance, expecting the uncertainty can localize transient components of an image in the wild and eliminate its influence for static components learning. However, this method is not applicable in reflective scenes, since it learns the implicit uncertainty that only considers the information of a single ray and ignores the multi-view context.

To address this issue, we propose defining an explicit reflection score that leverages multi-view context obtained through pixel colors from multi-view images referring to the same surface point. First, we identify the visibility of all source views given a surface point. Next, we project the point to visible images to obtain pixel colors. Based on this, we use an anomaly detector to estimate the reflection score, which serves as the variance. By minimizing the negative log-likelihood of the Gaussian distribution of the color, a large variance attenuates the importance. We further demonstrate that, by using reflection direction-dependent radiance, our model achieves promising results for multi-view reconstruction with a better radiance fields.

To summarize, our contributions are listed as follows.

- To the best of our knowledge, we present the first neural implicit surface learning framework for reconstructing objects with reflective surfaces.

- We propose a simple yet effective approach that enable neural implicit surface learning to handle reflective surfaces. Our approach can produce high-quality surface geometry and surface normals.

- Extensive experiments on several datasets show that the proposed framework significantly outperforms the state-of-the-art methods on reflective surfaces.

2. Related Works

2.1. Multi-View Stereo for 3D Reconstruction

Multi-View Stereo (MVS) is a technique that aims to reconstruct fine-grained scene geometry from multi-view images. Traditional MVS methods can be classified into four categories based on the output scene representation: volumetric-based methods [6, 41], mesh-based methods [10], point cloud-based methods [11, 24], and depth map-based methods [4, 12, 38, 39, 47]. Among them, depth map-based methods are the most flexible, estimating depth maps for each view by utilizing photometric consistency between reference and neighboring images [11], then fusing all the depth maps into dense point clouds. Surface reconstruction methods [6, 18, 22], such as screened Poisson Surface Reconstruction [18], are then employed to reconstruct surfaces from the point clouds.

Recent learning-based MVS methods use deep learning for improved reconstruction. SurfaceNet [15] and LSM [17] were the first volumetric learning-based MVS pipelines proposed to regress surface voxels. More recently, MVSNet [50] extracts deep image features and warps them into the reference camera frustum to construct a 3D cost volume via differentiable warping.

However, learning-based MVS methods may still produce unsatisfactory results in certain scenarios, such as surfaces with specular reflections, regions with low texture, and non-Lambertian regions. In these cases, photometric consistency across multi-view images is not guaranteed, which can lead to severe artifacts and missing parts in the reconstruction results.

2.2. Neural Implicit Surface for 3D Reconstruction

Recently, learning-based approaches using implicit surface representations have been proposed. In these representations, a neural network maps continuous points in 3D space to an occupancy field [29, 36] or a Signed Distance Function (SDF) [34]. These methods perform multi-view reconstruction with additional supervision corresponding to the occupancy value or SDF for each point. However, supervision for these methods is not always available with only multi-view 2D images, which limits their scalability.

The introduction of volumetric approaches in NeRF [30], which combines classical volume rendering [16] with implicit functions for novel view synthesis, has attracted a lot of attention towards 3D reconstruction using neural implicit surface representations and volume rendering [31, 52, 45]. Unlike NeRF, which aims to render novel view images while keeping the geometry unconstrained, these methods define the surface in a more explicit manner and are therefore better suited for surface extraction. UNISURF [31] uses the occupancy field [29], while IDR [53], VolSDF [52], and NeuS [45] use the SDF field [34] as an implicit surface

representation. Despite their promising performance on 3D reconstruction, these methods fall short of recovering the correct geometry of objects with reflections, leading to ambiguous surface optimization. Our approach builds upon NeuS [45], but we believe it can be adapted to fit any volumetric neural implicit framework.

2.3. Modeling for Object with Reflection

We discuss rendering and reconstruction for objects with reflections. Recent works [3, 42, 55, 57, 43] have investigated rendering view-dependent reflective appearance by decomposing a scene into shape, reflectance, and illumination for novel view synthesis and relighting. However, the recovered meshes are not explicitly validated, and the geometry is often unsatisfactory. Reconstruction, on the other hand, aims to recover explicit geometry, which is still under-explored due to the inherent challenges. For instance, PM-PMVS [5] formulates the reconstruction task as joint energy minimization over surface geometry and reflectance, while nLMVS-Net [49] formulates MVS as an end-to-end learnable network, leveraging surface normals as view-independent surface features for cost volume construction and filtering. However, none of these methods combines neural implicit surfaces with volume rendering for reconstruction.

2.4. Warping-based Consistency Learning

Warping-based consistency learning is widely used in both multi-view stereo [44, 48, 47, 54] and neural implicit surface learning [7, 9] for 3D reconstruction by exploiting inter-image correspondence with differentiable warping operations. Generally, consistency learning in MVS-based pipelines is performed at the CNN feature level. For example, MVSDF [54] warps the predicted surface point to its neighboring views and enforces pixel-wise feature consistency, while ACMM [47] warps the coarse predicted depth to form a multi-view aggregated geometric consistency cost to refine finer scales. Consistency learning in neural implicit surface-based pipelines, on the other hand, is typically performed at the image level. NeuralWarp [7] warps the sampled points along a ray to source images to obtain their RGB values and optimizes them jointly with a radiance network, while Geo-NeuS [9] warps the gray-scale patch centered on the predicted surface point to its neighboring images to guarantee multi-view geometry consistency. However, they ignore view-dependent radiance and are limited when multi-view consistency is not reasonable due to reflection, which can lead to artifacts when minimizing patch similarity regardless of reflection. Furthermore, visibility identification is not well-handled, and both rely on cumbersome preprocessing to determine source images. Alternatively, we leverage inconsistency to reduce ambiguity for high-fidelity reconstruction.

3. Approach

Given N calibrated multi-view images $\mathcal{X} = \{\mathbf{I}_i\}_{i=1}^N$ of an object with reflective surfaces, we aim to reconstruct the surfaces by neural implicit surface learning. Section 3.1 introduces NeuS, our baseline for reconstruction. Section 3.2 introduces a reflection-aware photometric loss. It reduces the influence of reflection by formulating rendering color as a Gaussian distribution with an explicit variance estimation that considers multi-view context. Section 3.3 discusses how we identify the visibility of source views to obtain unbiased reflection score. Section 3.4 shows that together with a reflection direction-dependent radiance, our model achieves better geometry with a better radiance fields. Finally, Section 3.5 presents our full optimization. An overview of our approach is provided in Figure 2.

3.1. Volume Rendering with Implicit Surface

Volume rendering [16] is used in NeRF [30] for novel view synthesis. The idea is to represent the continuous attributes (i.e., density and radiance) of a 3D scene with neural networks. α compositing [28] aggregates these attributes along a ray $\mathbf{r}$ to approximate the pixel RGB values by:

$$\hat{\mathbf{C}}(\mathbf{r}) = \sum_{i=1}^P T_i \alpha_i \mathbf{c}_i, \tag{1}$$

where $T_i = \exp\left(-\sum_{j=1}^{i-1} \alpha_j \delta_j\right)$ and $\alpha_i = 1 - \exp\left(-\sigma_i \delta_i\right)$ denote the transmittance and alpha value of sampled point, respectively. δ_i is the distance between neighboring sampled points. P is the number of sampled points along a ray. σ_i and $\mathbf{c}_i$ are predicted attributes by the neural networks conditioned on position $\mathbf{x} = (x, y, z)$ and view direction $\mathbf{d} = (\theta, \phi)$. The training object $\mathcal{L}$ of NeRF is the mean square error between the ground-truth pixel color $\mathbf{C}(\mathbf{r})$ and the rendering color $\hat{\mathbf{C}}(\mathbf{r})$ formulated as

$$\mathcal{L}_{\text{color}} = \sum_{\mathbf{r} \in \mathcal{R}} \|\mathbf{C}(\mathbf{r}) - \hat{\mathbf{C}}(\mathbf{r})\|_2^2, \tag{2}$$

where $\mathcal{R}$ is the set of all rays shooting from the camera center to image pixels.

However, density-based volume rendering lacks a clear definition of the surface, which makes it difficult to extract precise geometry. Alternatively, Signed Distance Function (SDF) clearly defines the surface as the zero-level set, making SDF-based volume rendering more effective in surface reconstruction. Following NeuS [45], the attributes of a 3D scene include signed distance and radiance parameterized by a geometry network f and a radiance network c by:

$$s = f(\mathbf{x}), \quad \mathbf{c} = c(\mathbf{x}, \mathbf{d}), \tag{3}$$

where the geometry network f maps a spatial position $\mathbf{x}$ to its signed distance $f(\mathbf{x})$ to the object and the radiance
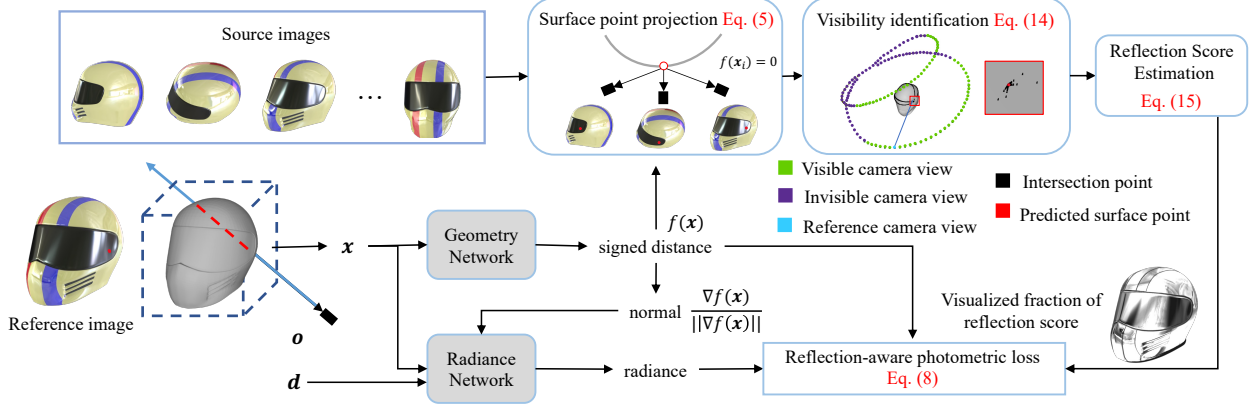
Figure 2. An overview of our framework. We propose a reflection-aware photometric loss to address the ambiguity induced by reflection. This loss attenuates the importance of reflective surfaces by lowering the weights of reflective pixels. Our loss relies on the estimated reflection score to localize these pixels. We use an anomaly detector that leverages a multi-view context to obtain the reflection score. This context is captured by projecting surface points to visible source views with a visibility identification module.

network c predicts the color conditioned on position $\mathbf{x}$ and view direction $\mathbf{d}$ to model the view-dependent radiance. To aggregate the signed distances and colors of sampled points along a ray $\mathbf{r}$ for pixel color approximation, we utilize volume rendering similar to that of NeRF. The key difference is the formulation of α_i, which is calculated from the signed distance $f(\mathbf{x})$ rather than density σ_i as

$$\alpha_i = \max\left(\frac{\Phi_s\left(f(\mathbf{x}_i)\right) - \Phi_s\left(f(\mathbf{x}_{i+1})\right)}{\Phi_s\left(f(\mathbf{x}_i)\right)}, 0\right), \quad (4)$$

where $\Phi_s(x) = (1 + e^{-sx})^{-1}$ and $1/s$ is a trainable parameter which indicates the standard deviation of $\Phi_s(x)$.

3.2. Anomaly Detection for Reflection Score

For multi-view reconstruction, multi-view consistency is the promise for accurate surface reconstruction. However, for reflective pixels, the geometry network often predicts ambiguous surfaces, which devastates the multi-view consistency. To overcome the issue, we propose to reduce the influence of reflective surfaces by a reflection-aware photometric loss, which adaptively lowers the weights assigned to reflective pixels. To achieve this, we first define the reflection score, which allows us to identify reflective pixels.

A naive solution is to treat the uncertainty defined in NeRF-W [27] as the reflection score. This approach models the radiance value of a scene as a Gaussian distribution and considers the predicted uncertainty as the variance. By minimizing the negative log-likelihood of the Gaussian distribution, a large variance reduces the importance of a pixel with high uncertainty. Ideally, the reflective pixels should be assigned a large variance to attenuate its influence for reconstruction. However, the implicit uncertainty learned by the MLP is defined on a single ray without considering the multi-view context. Therefore, it may not accurately localize reflective surfaces without explicit supervision.

Similar to NeRF-W [27], we also formulate rendering the color of a ray as a Gaussian distribution $\hat{\mathbf{C}}(\mathbf{r}) \sim (\overline{\mathbf{C}}(\mathbf{r}), \overline{\beta}^2(\mathbf{r}))$, where $\overline{\mathbf{C}}(\mathbf{r})$ and $\overline{\beta}^2(\mathbf{r})$ are the mean and variance, respectively. We adopt Eq. (1) to query $\overline{\mathbf{C}}(\mathbf{r})$. However, unlike NeRF-W, which only defines the implicit variance based on the information of a single ray, we explicitly define the variance based on the multi-view context. Specifically, we utilize multi-view pixel colors that refer to the same surface point to determine the variance.

To obtain the multi-view pixel colors, we project the surface point $\mathbf{x}$ onto all N images $\{\mathbf{I}_i\}_{i=1}^N$ and use bilinear interpolation to obtain the corresponding pixel colors $\{\mathbf{C}_i(\mathbf{r})\}_{i=1}^N$. For simplicity, omitting the subscript, the pixel color $\mathbf{C}$ is obtained by

$$\begin{aligned} \mathcal{G} &= \mathbf{K} \cdot (\mathbf{R} \cdot \mathbf{x} + \mathbf{T}), \\ \mathbf{C} &= \mathrm{interp}(\mathbf{I}, \mathcal{G}), \end{aligned} \quad (5)$$

where interp indicates bilinear interpolation, $\mathbf{K}$ denotes the internal calibration matrix, $\mathbf{R}$ denotes the rotation matrix, $\mathbf{T}$ denotes the translate matrix and $\cdot$ is matrix multiplication.

Considering the fact that only the local region of partial images contains reflection, we treat reflection localization as an anomaly detection problem, with the expectation that reflective surfaces will be regarded as the anomaly and assigned a high reflection score. To this end, we estimate a view-dependent reflection score $\overline{\beta}_i^2(\mathbf{r})$ by an anomaly detector empirically, which uses the Mahalanobis distance [26] as the reflection score (i.e., variance) as follows:

$$\overline{\beta}_i^2(\mathbf{r}) = \gamma \frac{1}{N} \sum_{j=1}^N \sqrt{(\mathbf{C}_i(\mathbf{r}) - \mathbf{C}_j(\mathbf{r}))^T \boldsymbol{\Sigma}^{-1} (\mathbf{C}_i(\mathbf{r}) - \mathbf{C}_j(\mathbf{r}))}, \quad (6)$$

where γ is the a scale factor to control the scale of the reflection score and $\boldsymbol{\Sigma}$ is the empirical covariance matrix. As reflections do not dominate the majority of training images,

if the currently rendered pixel color $\mathbf{C}_i(\mathbf{r})$ is contaminated by reflection, a larger reflection score is generated as most of the relative divergences become large.

Then, we extend the photometric loss in Eq. (2) to a reflection-aware one by minimizing the negative log-likelihood of distribution of rays $\mathbf{r}$ from a batch $\mathcal{R}$ similar to NeRF-W [27] and ActiveNeRF [32] as follows:

$$\mathcal{L}_{\text{color}} = -\log p(\hat{\mathbf{C}}(\mathbf{r})) = \sum_{\mathbf{r} \in \mathcal{R}} \frac{\|\mathbf{C}(\mathbf{r}) - \overline{\mathbf{C}}(\mathbf{r})\|_2^2}{2\overline{\beta}^2(\mathbf{r})} + \frac{\log \overline{\beta}^2(\mathbf{r})}{2}. \tag{7}$$

Since $\overline{\beta}^2(\mathbf{r})$ is estimated by Eq. (6) explicitly instead of implicitly learning by MLP, it is a constant and can be removed from the objective function. Besides, following previous works [45, 52, 9] for multi-view reconstruction, we use L1 loss instead of L2 loss. Finally, our reflection-aware photometric loss is quite simple formulated as

$$\mathcal{L}_{\text{color}} = \sum_{\mathbf{r} \in \mathcal{R}} \frac{|\mathbf{C}(\mathbf{r}) - \overline{\mathbf{C}}(\mathbf{r})|}{\overline{\beta}^2(\mathbf{r})}. \tag{8}$$

Note that we slightly abuse the term $\overline{\beta}^2(\mathbf{r})$, which should be corresponding to L2 loss.

3.3. Visibility Identification for Reflection Score

The computation of reflection score using all pixel colors $\{\mathbf{C}_j(\mathbf{r})\}_{j=1}^N$ assumes that the point on the surface has a valid projection on all source images. However, in practice, this assumption is not true due to self-occlusion. The projected pixel colors are meaningless if the point is invisible in the source images, and the corresponding pixel colors should not be used in Eq. (6).

To address this issue, we design a visibility identification module, which leverages intermediate reconstructed meshes to identify visibility, as illustrated in Fig. 3. Specifically, given a pixel p_i corresponding to a ray $\mathbf{r}_i$, the implicit surfaces on ray $\mathbf{r}_i$ can be represented according to the signed distance of sampled points $\mathbf{x}$ as follows:

$$\hat{S}_i = \{\mathbf{x} \mid f(\mathbf{x}) = 0\}. \tag{9}$$

Since an infinite number of points exist along a ray, we need to sample discrete points on this ray. Based on the sampled points and their signed distances, we can determine the intervals where surface points exist by

$$T_i = \{\mathbf{x}_j \mid f(\mathbf{x}_j) \cdot f(\mathbf{x}_{j+1}) < 0\}. \tag{10}$$

If the sign of the sampled point $f(\mathbf{x}_j)$ is different from the sign of the next sampled point $f(\mathbf{x}_{j+1})$, the interval $[\mathbf{x}_j, \mathbf{x}_{j+1}]$ intersects with the surface. The intersection point set $\hat{S}_i$ can be obtained by linear interpolation by

$$\hat{S}_i = \left\{\mathbf{x} \mid \mathbf{x} = \frac{f(\mathbf{x}_j)\mathbf{x}_{j+1} - f(\mathbf{x}_{j+1})\mathbf{x}_j}{f(\mathbf{x}_j) - f(\mathbf{x}_{j+1})}, \mathbf{x}_j \in T_i\right\}. \tag{11}$$
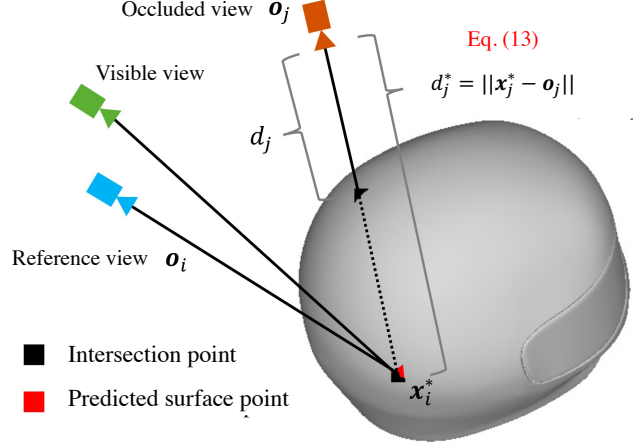

Figure 3. Illustration of visibility identification.

In practice, the ray $\mathbf{r}_i$ may intersect with the object at multiple surfaces. For our reflection score computation, only the first intersection is meaningful, and it is formulated as

$$\mathbf{x}_i^* = \arg\min \mathcal{D}(\mathbf{x}, \mathbf{o}_i), \tag{12}$$

where $\mathbf{x} \in \hat{S}_i$ and $\mathcal{D}(\cdot, \mathbf{o}_i)$ indicate the distance between point $\mathbf{x}$ and the origin of the ray $\mathbf{r}_i$, respectively.

After capturing the predicted surface point $\mathbf{x}_i^*$, we can calculate the distances $\{d_j^*\}_{j=1}^N$ between this point and all camera locations $\{\mathbf{o}_j\}_{j=1}^N$ as follows:

$$d_j^* = \|\mathbf{x}_i^* - \mathbf{o}_j\|. \tag{13}$$

Meanwhile, we compute the distances $\{d_j\}_{j=1}^N$ from all camera locations $\{\mathbf{o}_j\}_{j=1}^N$ to the first intersection of the intermediate reconstructed meshes by ray casting [37]. Based on these two distances, the visibility of images $\mathbf{I}_j$ can be estimated by

$$v_j = \mathbb{I}(d_j^* \leq d_j) \tag{14}$$

where $\mathbb{I}(\cdot)$ is an indicator function. Based on the approximation of visibility, we eliminate the invisible pixel colors $\{\mathbf{C}_j(\mathbf{r}) \mid v_j = 0\}$ that are used in the calculation of the reflection score in Eq. (6). We then refine the reflection score as follows:

$$\overline{\beta}_i^2(\mathbf{r}) = \gamma \frac{1}{\sum_{j=1}^N v_j} \sum_{j=1}^N v_j \text{Mdis},$$

$$\text{Mdis} = \sqrt{(\mathbf{C}_i(\mathbf{r}) - \mathbf{C}_j(\mathbf{r}))^T \mathbf{\Sigma}^{-1} (\mathbf{C}_i(\mathbf{r}) - \mathbf{C}_j(\mathbf{r}))}. \tag{15}$$

We provide some examples in Fig. 4 to illustrate the estimated reflection score.

Figure 4. Visualization of the fraction of the reflection score, which can localize reflective surfaces. Black indicates a high score.

3.4. Reflection Direction Dependent Radiance

As suggested in Ref-NeRF [43], conditioning the radiance on reflection direction in reflective scenes can lead to a more accurate radiance fields, which has been shown to be beneficial for reconstruction in NeuralWarp [7]. Inspired by this, we reparameterize the radiance network as a function of the reflection direction about the surface normal, and the formulation in Eq. (3) becomes

$$\mathbf{c} = c(\mathbf{x}, \boldsymbol{\omega}_r), \tag{16}$$

where $\hat{\mathbf{d}}$ is the reflection direction calculated by

$$\boldsymbol{\omega}_r = 2(-\mathbf{d} \cdot \hat{\mathbf{n}})\hat{\mathbf{n}} + \mathbf{d}, \tag{17}$$

where $\mathbf{n}$ is the surface normal formulated by

$$\hat{\mathbf{n}} = \frac{\nabla f(\mathbf{x})}{||\nabla f(\mathbf{x})||}. \tag{18}$$

Compared to Ref-NeRF [43], the reflection direction is more precise in our framework due to the well-estimated surface normal, which leads to a more accurate radiance fields. Compared to NeuralWarp [7] that ignores reflection in their framework, we take the view-dependent radiance into consideration and estimate a more accurate radiance fields. As a result, our method is more reliable and promising for multi-view reconstruction of objects with reflection.

3.5. Optimization

During reconstructing a scene, our total loss function is

$$\mathcal{L} = \mathcal{L}_{\text{color}} + \alpha \mathcal{L}_{\text{eik}}. \tag{19}$$

$\mathcal{L}_{\text{color}}$ is the reflection-aware photometric loss in Eq. (8) with reflection score estimated by Eq. (15). The rendered color is computed by Eq. (1) with radiance parameterized in Eq. (16). $\mathcal{L}_{\text{eik}}$ is an eikonal term [13] to regularize the gradients of geometry network formualated as

$$\mathcal{L}_{\text{eik}} = \frac{1}{P} \sum_{i=1}^{P} \left(|\nabla f(\boldsymbol{x}_i)| - 1 \right)^2. \tag{20}$$

In our experiments, we choose α as 0.1.

4. Experimetns

4.1. Datasets and Evaluation Protocol

To evaluate the effectiveness of our model, we conducted experiments on objects from several datasets, including Shiny Blender [43], Blender [30], SLF [46], and Bag of Chips [35]. The selected objects in these datasets are glossy.

Shiny Blender. The Shiny Blender dataset is a synthetic dataset introduced in [43]. The dataset includes six different glossy objects that were rendered in Blender with more challenging material properties. The original dataset was created for novel view synthesis evaluation. We selected four objects (i.e., helmet, coffee, toaster, and car) for reconstruction, as the geometry of the ball is too simple (i.e., a sphere of radius 1), and the teapot is less reflective. For all reconstruction tasks, we used the original 200 testing images for training, as we found that there are more reflective surfaces, which makes it more challenging for reconstruction. For quantitative evaluation, we provide ground-truth dense point clouds for each scene by upsampling points from the ground-truth meshes exported from Blender. Since rendering quality was also compared on ShinyBlender, we emphasize that for fair comparison with existing methods, we used the original 100 training images for training.

Blender. We used the glossy drums from Blender dataset [30]. This dataset was rendered in Blender as Shiny Blender dataset with 100 images for training.

SLF. We used the glossy fish from SLF [46]. This dataset was captured in a lab-controlled environment. The cameras are distributed on a hemisphere around the center object.

Bag of Chips. We used the glossy cans and corncho1 from [35], which provides meshes scanned by RGBD sensors for evaluation. Since there are more than 2000 images, we only sampled half of them for training due to limited memory. [1]

Evaluation Protocol. The evaluation metric is the Chamfer Distance, provided by the DTU evaluation metrics [1]. The metric can be divided into two parts: *accuracy* and *completeness* (see supplement for details). For the Shiny Blender dataset, the ground-truth meshes are double-layered, which results in many redundant points in the ground truth. Therefore, we only reported the accuracy in Table 1, as it is more meaningful than completeness. Additionally, for the Shiny Blender dataset, mean angular error (MAE) and PSNR were used for evaluating the estimated normals and rendering quality, respectively.

4.2. Implementation Details

We implemented our model based on NeuS [45]. The structure of the geometry network and radiance network

---

[1]The number of training images does not necessarily affect the reconstruction results when there are hundreds of uniformly-distributed views.

Methods	helmet		toaster		coffee		car		Mean	
	Acc↓	MAE↓	Acc↓	MAE↓	Acc↓	MAE↓	Acc↓	MAE↓	Acc↓	MAE↓
PhySG [55]	-	2.32	-	9.75	-	22.51	-	8.84	-	10.86
Ref-NeRF [43]	-	29.48	-	42.87	-	12.24	-	14.93	-	24.88
IDR [53]	-	-	-	-	3.79	3.68	<u>0.55</u>	1.25	-	-
UNISURF [31]	1.69	1.78	3.75	5.51	2.88	3.15	2.34	1.98	2.67	3.11
VolSDF [52]	1.55	1.37	<u>2.02</u>	<u>2.53</u>	2.23	2.28	0.58	<u>1.04</u>	<u>1.60</u>	1.81
NeuS [45]	1.33	<u>1.12</u>	3.26	2.87	<u>1.42</u>	<u>1.99</u>	0.73	1.10	1.69	<u>1.77</u>
NeuralWarp [7]	2.25	1.94	5.90	3.51	1.54	2.04	0.65	1.07	2.59	2.14
Geo-NeuS [9]	<u>0.74</u>	2.36	6.35	3.76	3.85	6.36	2.88	5.67	3.46	4.54
Ours	**0.29**	**0.38**	**0.42**	**1.47**	**0.77**	**0.99**	**0.37**	**0.80**	**0.46**	**0.91**

Table 1. Comparison with state-of-the-art methods on Shiny Blender Dataset. Except for Ref-NeRF and PhySG, whose results are taken from the original paper of Ref-NeRF, we implemented the released code on Shiny Blender dataset for other methods. IDR failed to recover meaningful geometry for helmet and toaster, so the results remain empty. Note that NeRF-W failed to produce meshes since it focuses on novel view synthesis in the wild, and COLMAP generated meshes with severe artifacts and missing parts. The quantitative results were not provided. **Bold** results have the best score and <u>underlined</u> the second best. Our method outperforms these methods by a large margin.

Method	drums	fish	cans	corncho1
IDR* [53]	1.91	**0.77**	1.45	0.87
NeuS [45]	2.29	1.14	1.89	0.92
Geo-NeuS [9]	-	0.87	2.80	1.19
Ours	**1.35**	0.81	**1.21**	**0.82**

Table 2. Comparison with state-of-the-art methods on drums, fish, cans and corncho1. IDR* requires images masks for training. Geo-Neus failed to recover drums. The average of accuracy and completeness is reported.

Method	RS	Vis	Ref	Acc↓	MAE↓
NeuS				1.69	1.77
NeuS w/ RS	✓			1.17	1.43
NeuS w/ Ref			✓	1.36	1.50
Ref-NeuS w/o Ref	✓	✓		0.80	1.21
Ours (full)	✓	✓	✓	**0.46**	**0.91**

Table 4. Ablation study on Shiny Blender dataset. "RS" indicates reflection score. "Vis" indicate visibility identification. "Ref" indicates reflection direction-dependent radiance.

Method	helmet	toaster	coffee	car	mean
NeuS [45]	27.78	23.51	28.82	26.34	26.61
Ref-NeRF [43]	29.68	25.70	**34.21**	**30.82**	30.10
Ours	**32.85**	**26.97**	31.05	29.92	**30.20**

Table 3. Rendering quality comparison on Shiny Blender dataset. PSNR is adopted as evluation metric.

was the same as that of NeuS. Please refer to our supplement for more details. To estimate visibility in an on-the-fly manner, the intermediate reconstruction result was updated every 500 iterations with a resolution of 128 to decrease the computation cost. γ was set to 5. We trained our model for 200k iterations, which took approximately 7 hours on a single NVIDIA RTX 3090 Ti GPU for the reconstruction task. After convergence, a mesh can be extracted from the signed distance functions (SDFs) in a predefined bounding box using Marching Cubes [25] with a resolution of 512.

4.3. Comparison with State-of-the-Art Methods

We compared the reconstruction quality of our method with several other methods, including IDR [53], three baseline methods for multi-view reconstruction (UNISURF [31], VolSDF [52], and NeuS [45]), two warp-based consistency learning methods (NeuralWarp [7] and Geo-NeuS [9]), and two methods specifically designed for modeling objects with reflection (Ref-NeRF [43] and PhySG [55]). The quantitative results are shown in Tables 1 and

2, where *accuracy* is indicated with the term "Acc" [1]. As COLMAP and NeRF-W fail to recover reflective surfaces, we did not compare with them. Instead, we presented the reconstructed meshes with severe artifacts of COLMAP in the supplement. For NeRF-W, it focuses on novel view synthesis in the wild and failed to produce meshes.

For PhySG and Ref-NeRF, we obtained the MAE from the Ref-NeRF paper, as these methods focus on novel view synthesis, the reconstruction accuracy is not reported in Ref-NeRF paper. For other methods, we implemented the released code on the corresponding datasets. IDR [53] was unable to generate meaningful reconstructions for the helmet and toaster objects, so we only reported the results for the car and coffee objects. Our method significantly outperformed all other compared methods by a large margin. As can be seen qualitatively in Figures 1 and 5, our method yields promising improvements in both geometry accuracy and surface normals. Ref-NeRF adopts Integrated Positional Encoding similar to mip-NeRF [2], which makes its geometry parameters dependent on view direction, making the meshes inaccessible. Furthermore, our results achieve higher rendering quality (PSNR) compared to Ref-NeRF except for a failure case as reported in Table 3. This indicates that more accurate surface normals can lead to improved novel view synthesis quality since the conditioned reflection direction is calculated more accurately compared to Ref-NeRF. Note that, due to memory limitations, we only sampled 1024 rays instead of 4096×4 as in Ref-NeRF.

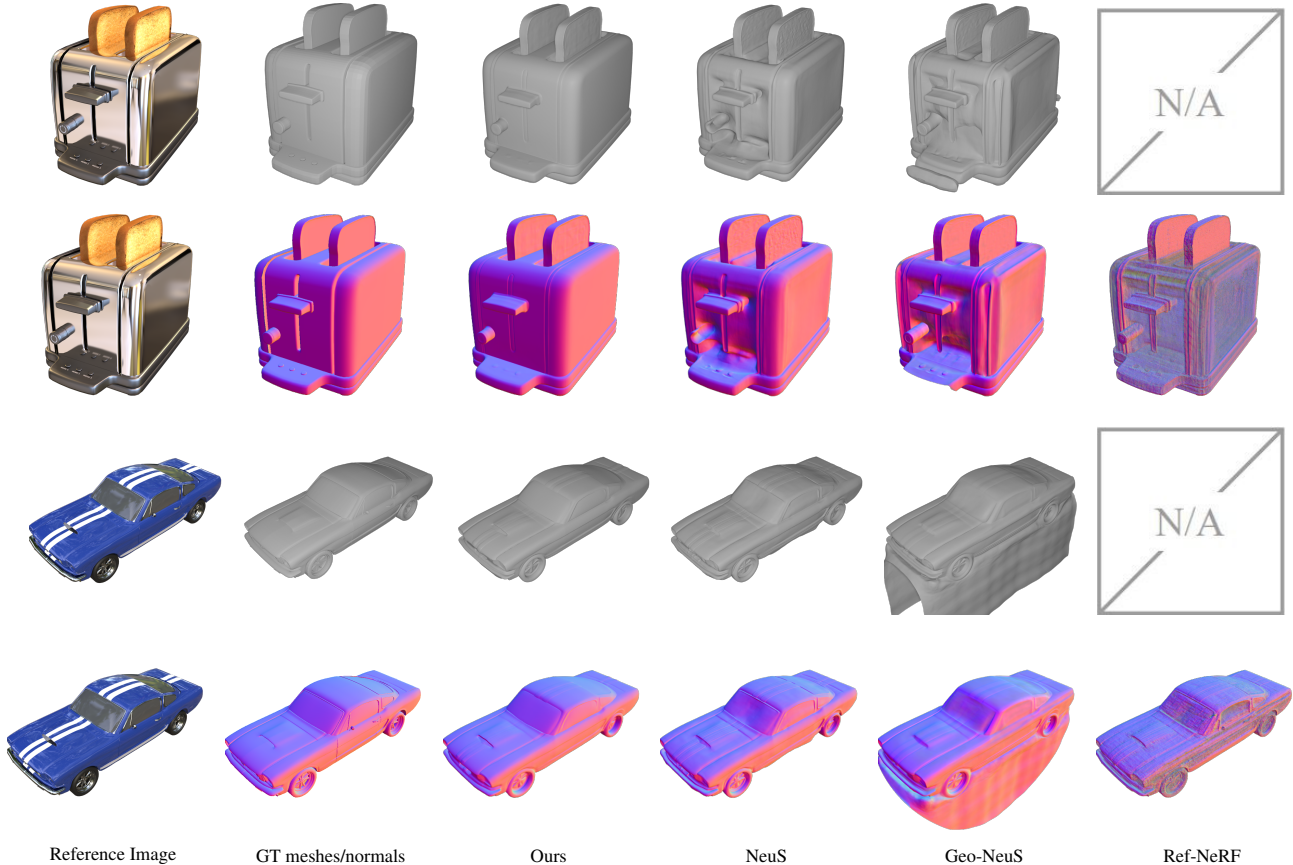| Reference Image | GT meshes/normals | Ours | NeuS | Geo-NeuS | Ref-NeRF |

Figure 5. The geometry of reconstructed meshes and estimated surface normals on Shiny Blender dataset [43]. We ran NeuS, Geo-Neus and Ref-NeRF official implementations. Our method obviously produces better geometry and surface normals than other methods.

4.4. Ablation Study

We conducted an ablation study on the Shiny Blender dataset to evaluate the effectiveness of each component in our model. The experimental results are reported in Table 4. "NeuS w/ RS" indicates that we computed the reflection-aware photometric loss with the reflection score estimated by Eq. (6), without accounting for visibility. This leads to a smaller improvement in performance, suggesting that using the reflection score as a source of variance can still be beneficial in improving geometry. "NeuS w/ Ref" only replaces the radiance dependency with the reflection direction without considering the reflection score. The performance improved over the baseline, except for one failure case, which we discussed and provided per-scene results for in the supplementary material. "Ref-NeuS w/o Ref" indicates that we further identified visibility for reflection score estimation using Eq. (15). The performance significantly improved, highlighting the importance of an unbiased reflection score in accurate geometry recovery. Finally, we used both reflection-aware photometric loss and reflection direction-dependent radiance in our model. The performance significantly improved compared to using only one of these methods, indicating that both methods are comple-

mentary for achieving accurate surface reconstruction.

5. Limitation and Conclusion

Limitation. Although our method shows promising results in multi-view reconstruction with reflection, several limitations remain. Firstly, estimating the reflection score inevitably increases the computational cost. Secondly, simply replacing the dependency of the radiance network with the reflection direction, regardless of the object material, can lead to artifacts in certain circumstances. We present an example of such an artifact in the supplementary material.

Conclusion. In this paper, we investigate the issue of multi-view reconstruction for objects with reflective surfaces, which is an important but under-explored problem. Reflection-induced ambiguity can significantly disrupt multi-view consistency, but our proposed Ref-NeuS method addresses this issue by introducing a reflection-aware photometric loss, where the importance of reflective pixels is attenuated using a Gaussian distribution model. Additionally, our method employs reflection direction-dependent radiance, which further improves the geometry with a better radiance fields, including geometry and surface normals.

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision (IJCV)*, 2016.

[2] 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 (ICCV)*, 2021.

[3] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. Nerd: Neural reflectance decomposition from image collections. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)*, 2021.

[4] Neill DF Campbell, George Vogiatzis, Carlos Hernández, and Roberto Cipolla. Using multiple hypotheses to improve depth-maps for multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2008.

[5] Ziang Cheng, Hongdong Li, Yuta Asano, Yinqiang Zheng, and Imari Sato. Multi-view 3d reconstruction of a textureless smooth surface of unknown generic reflectance. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[6] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996.

[7] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[8] Laura Falivene, Zhen Cao, Andrea Petta, Luigi Serra, Albert Poater, Romina Oliva, Vittorio Scarano, and Luigi Cavallo. Towards the online computer-aided design of catalytic pockets. *Nature Chemistry*, 2019.

[9] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *arXiv preprint arXiv:2205.15848*, 2022.

[10] Pascal Fua and Yvan G Leclerc. Object-centered surface reconstruction: Combining multi-image stereo and shading. *International Journal of Computer Vision (IJCV)*, 1995.

[11] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2009.

[12] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015.

[13] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020.

[14] Xiaojie Guo, Xiaochun Cao, and Yi Ma. Robust separation of reflection from multiple images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[15] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2017.

[16] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 1984.

[17] Abhishek Kar, Christian Häne, and Jitendra Malik. Learning a multi-view stereo machine. *Advances in neural information processing systems (NeurIPS)*, 2017.

[18] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 2013.

[19] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? *Advances in neural information processing systems (NeurIPS)*, 2017.

[20] Ronald W Kennard and Larry A Stone. Computer aided design of experiments. *Technometrics*, 1969.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[22] Patrick Labatut, Jean-Philippe Pons, and Renaud Keriven. Efficient multi-view reconstruction of large-scale scenes using interest points, delaunay triangulation and graph cuts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2007.

[23] John Lasseter. Principles of traditional animation applied to 3d computer animation. In *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987.

[24] Maxime Lhuillier and Long Quan. A quasi-dense approach to surface reconstruction from uncalibrated images. *IEEE transactions on pattern analysis and machine intelligence (TPAMI)*, 2005.

[25] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 1987.

[26] Prasanta Chandra Mahalanobis. On the generalised distance in statistics. In *Proceedings of the national Institute of Science of India*, 1936.

[27] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[28] Nelson Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics (VCG)*, 1995.

[29] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2019.

[30] 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*, 2021.

[31] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[32] Xuran Pan, Zihang Lai, Shiji Song, and Gao Huang. Activenerf: Learning where to see with uncertainty estimation. In *European conference on computer vision (ECCV)*, 2022.

[33] Rick Parent. *Computer animation: algorithms and techniques*. Newnes, 2012.

[34] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2019.

[35] Jeong Joon Park, Aleksander Holynski, and Steven M Seitz. Seeing the world in a bag of chips. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[36] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision (ECCV)*, 2020.

[37] Scott D Roth. Ray casting for modeling solids. *Computer graphics and image processing*, 1982.

[38] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, 2016.

[39] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision (ECCV)*, 2016.

[40] Martijn J Schuemie, Peter Van Der Straaten, Merel Krijn, and Charles APG Van Der Mast. Research on presence in virtual reality: A survey. *Cyberpsychology & behavior*, 2001.

[41] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *International Journal of Computer Vision (IJCV)*, 1999.

[42] Pratul P Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan T Barron. Nerv: Neural reflectance and visibility fields for relighting and view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[43] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[44] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[45] 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.

[46] 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 *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000.

[47] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[48] Zhenyu Xu, Yiguang Liu, Xuelei Shi, Ying Wang, and Yunan Zheng. Marmvs: Matching ambiguity reduced multiple view stereo for efficient large scale scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[49] Kohei Yamashita, Yuto Enyo, Shohei Nobuhara, and Ko Nishino. nlmvs-net: Deep non-lambertian multi-view stereo. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023.

[50] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, 2018.

[51] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[52] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[53] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[54] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[55] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[56] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.

[57] Xiuming Zhang, Pratul P Srinivasan, Boyang Deng, Paul Debevec, William T Freeman, and Jonathan T Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 2021.

Supplementary

A. Optimization and Additional Model Details

Optimization Details. We used Adam [21] as our optimizer. For the first 5,000 iterations, the learning rate was linearly increased from 0 to 5×10^{-4} using a warm-up strategy. After that, we controlled it using the cosine decay schedule to the minimum learning rate of 2.5×10^{-5}. We trained each model for 200,000 iterations, which took a total of 7 hours on a single NVIDIA RTX3090Ti GPU. For the novel view synthesis task, we trained each model for 1,000,000 iterations over 80 hours using a smaller batch size with fewer sampled rays on a single NVIDIA RTX3090Ti GPU. To ensure consistency with the reconstruction baselines, we used single-image batching with 512 sampled rays for all reconstruction tasks. For novel view synthesis, we used single-image batching with 1024 sampled rays, limited to the GPU memory, instead of 4096×4 as used in Ref-NeRF. On each ray, we sampled 64 coarse points, 64 fine points, and 32 points to model the background, as in NeRF++ [56].

Network architecture. Our network architecture is similar to NeuS [45], comprising of a geometry network and a radiance network to encode SDF and view-dependent radiance, respectively. The geometry network parametrizes the signed distance function and consists of 8 hidden layers with a hidden size of 256. Instead of ReLU, we used Softplus with $\beta = 100$ for all hidden layers. We used a skip connection [30] to connect the input with the output of the fourth layer. The geometry network takes the spatial position $\mathbf{x}$ of points as input and outputs the signed distances to the object. In addition, the geometry network produces a geometry feature with dimension 256, which is further used as input to the radiance network to acquire view-dependent radiance. The radiance network comprises 4 hidden layers of size 256, which parametrize view-dependent radiance. It takes as input the spatial position $\mathbf{x}$, the normal vector $\hat{\mathbf{n}}$, the reflection direction $\boldsymbol{\omega}_r$, and the 256-dimensional geometry feature vector. We applied positional encoding with 6 frequencies to the spatial location $\mathbf{x}$ and 4 frequencies to the view direction $\boldsymbol{\omega}_r$.

B. Evaluation Details

Meshes. For the ShinyBlender [43] and Blender [30] datasets, the ground truth meshes were exported from Blender files. Due to the original models' small scales with a radius around 1, we exported them with a scale factor of 150. For the fish from SLF [46] and the cans/corncho1 from Bag of Chips [35], we increased the meshes' sizes by 100 and 1000 times, respectively, resulting in similar scales for all ground truth meshes. During training, we normalized the object to a unit sphere. During inference, we transferred
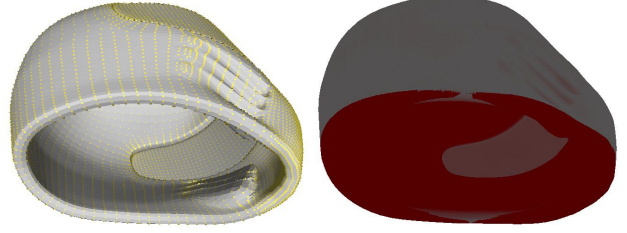

Figure 6. In the ground truth meshes (left) of ShinyBlender, there are two layers. However, on multi-view images, the inner layer is invisible. This means the completeness is biased, as most of the points in the inner layer contribute to meaningless error (right). Points with large errors are marked in red.
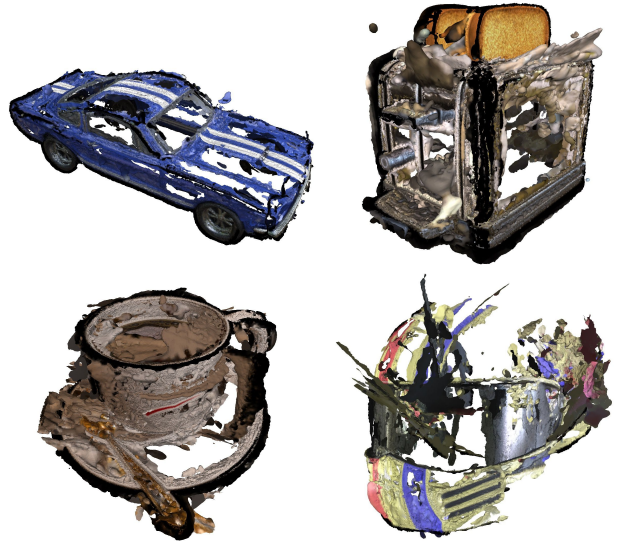

Figure 7. Visualization of the reconstruction results by COLMAP on the ShinyBlender dataset. COLMAP failed to reconstruct objects with reflective surfaces, as the multi-view consistency was not plausible.

the meshes to the original space to compute the Chamfer Distance.

Since the original meshes contain too few points, we upsampled the points in each triangle to obtain dense point clouds for evaluation. Finally, the Chamfer Distance was computed by

$$d(S_1, S_2) = \frac{1}{S_1} \sum_{x \in S_1} \min_{y \in S_2} ||x - y||_2^2 + \frac{1}{S_2} \sum_{y \in S_2} \min_{x \in S_1} ||y - x||_2^2, \tag{21}$$

where the first term is used to test *accuracy*, and the second term validates *completeness* [1]. S_1 and S_2 are the recovered point clouds upsampled from meshes and ground truth dense point clouds, respectively. For the ShinyBlender dataset, shown in Fig. 6, the ground truth dense point clouds include two layers. However, the inner layer is invisible on

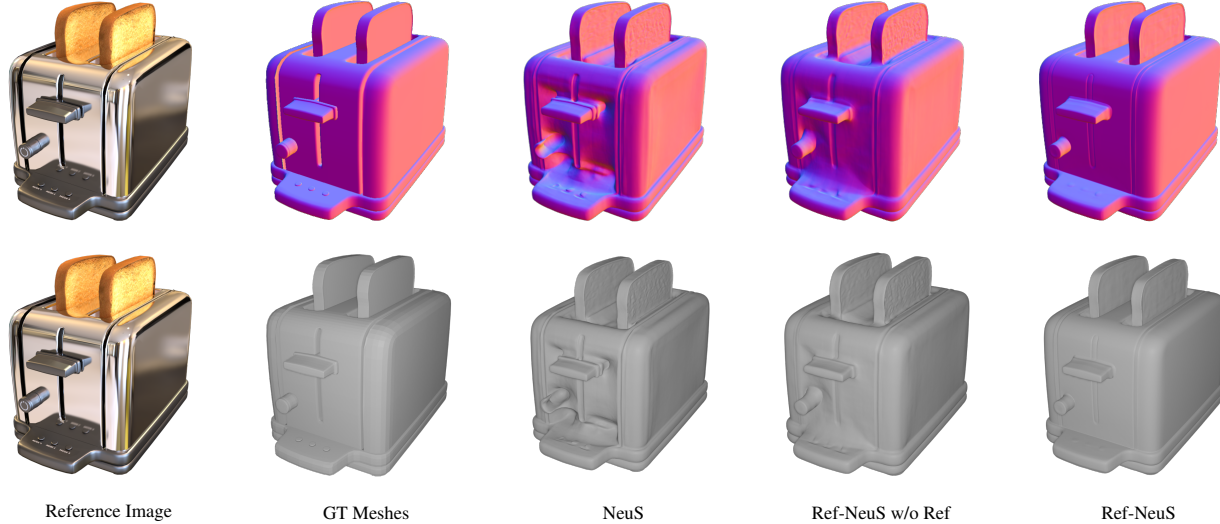| Reference Image | GT Meshes | NeuS | Ref-NeuS w/o Ref | Ref-NeuS |

Figure 8. Surface geometry and surface normals of ablation models on toaster of ShinyBlender.

multi-view images and cannot be reconstructed, resulting in a biased completeness, so only accuracy is reported.

Surface Normals. To compute the surface normal for a pixel p, we compute the normals of sampled points along the ray $\mathbf{r}$ derived from the SDF as follows:

$$\hat{\mathbf{n}}_{\mathbf{i}} = \frac{\nabla f(\mathbf{x_i})}{||\nabla f(\mathbf{x_i})||}. \quad (22)$$

Then, the volume rendering procedure is performed to aggregate these normals, forming a single surface normal:

$$\hat{\mathbf{n}}(\mathbf{r}) = \sum_{i=1}^{N} T_i \alpha_i \hat{\mathbf{n}}_i. \quad (23)$$

We used the normalized normals $\bar{\mathbf{n}}(\mathbf{r}) = \frac{\hat{\mathbf{n}}(\mathbf{r})}{||\hat{\mathbf{n}}(\mathbf{r})||}$ for evaluating MAE for all pixels.

C. Results of using training views

In the main text, we used original 200 test views for reconstruction training, here we show the results of using 100 training views for reconstruction training in Table 6. The conclusion is similar to that obtained using the test views.

D. Detailed Results of Ablation Study

We reported the quantitative metrics (accuracy and MAE) for each scene of ShinyBlender in Table 5.

E. Additional Results on diffuse materials

We also carried out experiments on non-reflective objects to show that reflection score will not cause performance degradation of non-reflective objects, where DTU scenes (i.e., scan55, scan83, scan105, scan106, scan114, scan118) were used. The results are reported in Table 7.

F. Additional Results on scenes with both diffuse and shiny materials

Previous reconstructed scenes are either shiny or diffuse materials. To show the robustness of our method, we further diverse the scenes with both diffuse and shiny materials. Given that such scenes are uncommon in existing datasets, besides materials for Blender dataset, we further employed Blender to render multi-view images that combine helmet for Shinyblender and hotdog from Blender. The comparison is presented in Fig. 9. Our method can reconstruct the shiny objects better, while do not lead to performance drop on diffuse materials.

G. Additional Visualizations

Visualizations of COLMAP. We visualized the reconstruction results of COLMAP [39], an MVS-based method on the Shiny Blender dataset, as shown in Fig. 7. COLMAP fails to recover reflective surfaces, indicating that the multi-view consistency is not reasonable in reflective scenes, leading to severe missing parts and artifacts.

Visualizations of ablation study. We visualized how the reflection-aware photometric loss and reflection direction-dependent radiance improve surface quality in Fig. 8. Due to the reflective surfaces, the toaster is extremely challenging even for human perception. It is challenging to distinguish where the real surface lies. NeuS reconstructs the toaster with severe missing parts due to reflection. "Ref-Neus w/o Ref" reconstructs the surface with fewer missing parts, indicating that reflection-aware photometric loss can

Method	RS	Vis	Ref	helmet		toaster		coffee		car		mean	
				Acc	MAE	Acc	MAE	Acc	MAE	Acc	MAE	Acc	MAE
NeuS				1.33	1.12	3.26	2.87	1.42	1.99	0.73	1.10	1.69	1.77
NeuS w/ RS	✓			0.75	0.85	2.14	2.23	1.11	1.58	0.66	1.05	1.17	1.43
NeuS w/ Ref			✓	0.41	0.69	0.59	1.59	3.87	2.74	0.55	0.97	1.36	1.50
Ref-NeuS w/o Ref	✓	✓		0.43	0.71	1.43	2.12	0.77	0.99	0.58	1.00	0.80	1.21
Ours (full)	✓	✓	✓	**0.29**	**0.38**	**0.42**	**1.47**	**0.77**	**0.99**	**0.37**	**0.80**	**0.46**	**0.91**

Table 5. Detailed quantitative metrics of ablation study on Shiny Blender dataset.

Methods	helmet		toaster		car	
	Acc↓	MAE↓	Acc↓	MAE↓	Acc↓	MAE↓
Neus	0.92	0.88	3.34	2.73	0.72	1.08
Ref-NeuS	0.33	0.39	0.45	1.56	0.36	0.77

Table 6. Results of using training views for reconstruction training.

scene	55	83	105	106	114	118
NeuS	0.37	1.45	0.78	0.52	0.36	0.45
NeuS w/ RS	0.36	1.27	0.72	0.51	0.36	0.46

Table 7. The results on non-reflective objects.

setting	Running time [h]
NeuS	7
Ref-NeuS	7.5

Table 8. Comparison of running time.

localize the reflective surfaces and alleviate the ambiguity. Our full model, Ref-NeuS, achieves better reconstruction results without missing parts.

Visualizations of Ref-NeuS. We present additional visualization results of different objects in Fig. 10, Fig. 11, Fig. 12, and Fig. 13 to demonstrate the effectiveness of our Ref-NeuS. Our Ref-NeuS achieves better reconstruction quality compared to NeuS.

Visualizations of results on Hulk. The real-world objects used in our experiments were captured under strict conditions in a lab-controlled environment. Differently, we captured the Hulk with glossy surfaces using an iPad in a natural environment, capturing both the object and its surroundings with lighting illumination and ambient light. As we captured the object with the iPad moving around it, the light source may have been occluded, resulting in shadows on the surfaces. We show the results in Fig. 14, we can still achieve better performance than NeuS.

H. Running Time

We demonstrate that estimating the reflection score will not significantly increase the running time. There are two main steps that contribute to the increase in running time. The first step involves the intermediate meshes. Since we extract intermediate meshes with a resolution of 128, it only takes approximately 0.35 seconds for each mesh extraction. The second step involves projection and distance computation for visibility identification. To obtain pixel colors, we project the predicted surface point onto visible source images. This step does not incur notable extra computational cost, with only 0.012 seconds per step. In Table 8, we present the total running time.

I. Failure Case

Figure 15 displays the reconstruction results of the coffee object using ShinyBlender, which is a failure case of our method. This object contains water surfaces that possess different reflection coefficients compared to solid objects. Merely substituting the dependency of the radiance network with reflection direction, without considering the object material, can result in artifacts. This motivates future work on how to better model view-dependent radiance while taking the material into consideration. However, incorporating reflection-aware photometric loss can still improve the reconstruction quality over NeuS. We present the results of "Ref-NeuS w/o Ref" for Ref-NeuS on the coffee object of ShinyBlender.
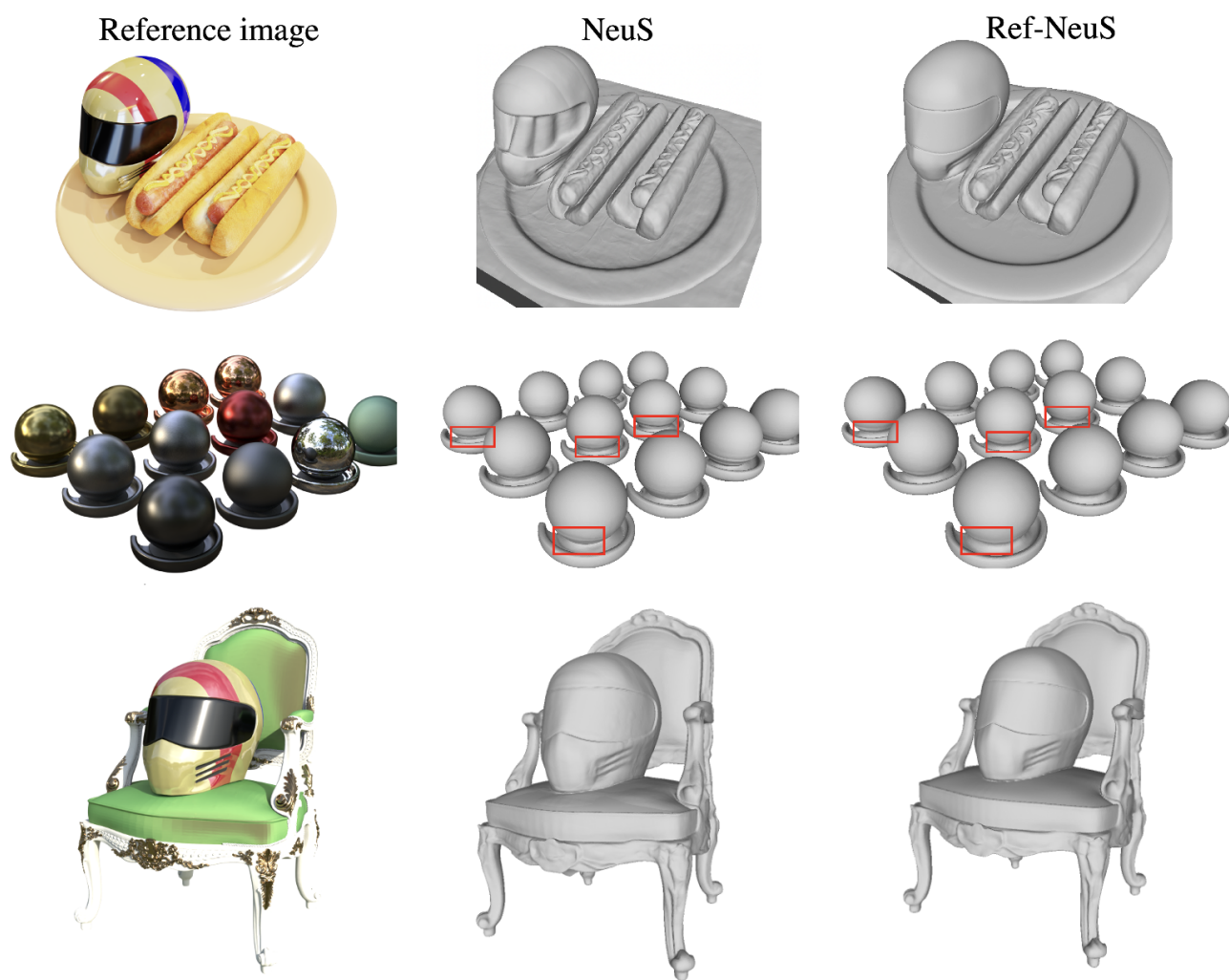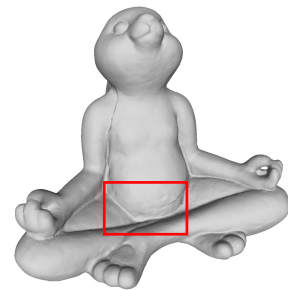
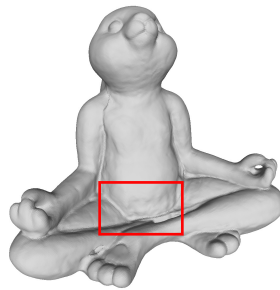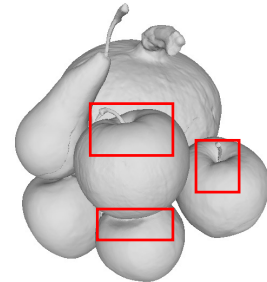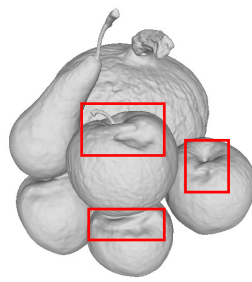| Reference image	NeuS	Ref-NeuS

Figure 9. Comparison on scenes with both diffuse and shiny materials.

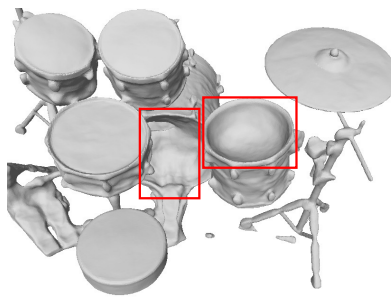Reference image   NeuS   Ref-NeuS
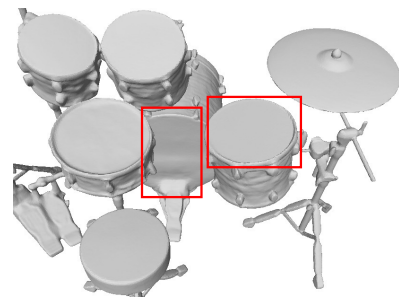
Figure 10. Qualitative comparison with NeuS on scan63 and scan 110 of DTU dataset.
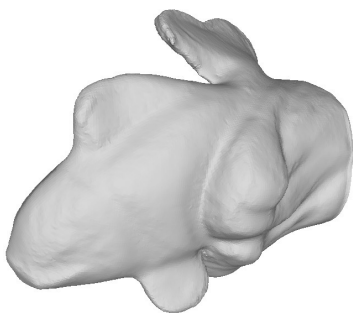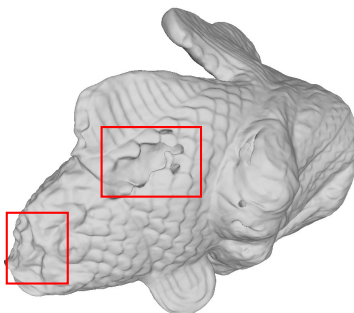

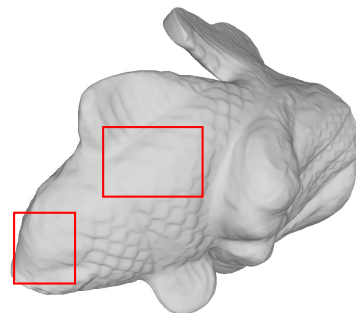
GT   NeuS   Ref-NeuS

Figure 11. Qualitative comparison with NeuS on drums of Blender dataset.

GT   NeuS   Ref-NeuS

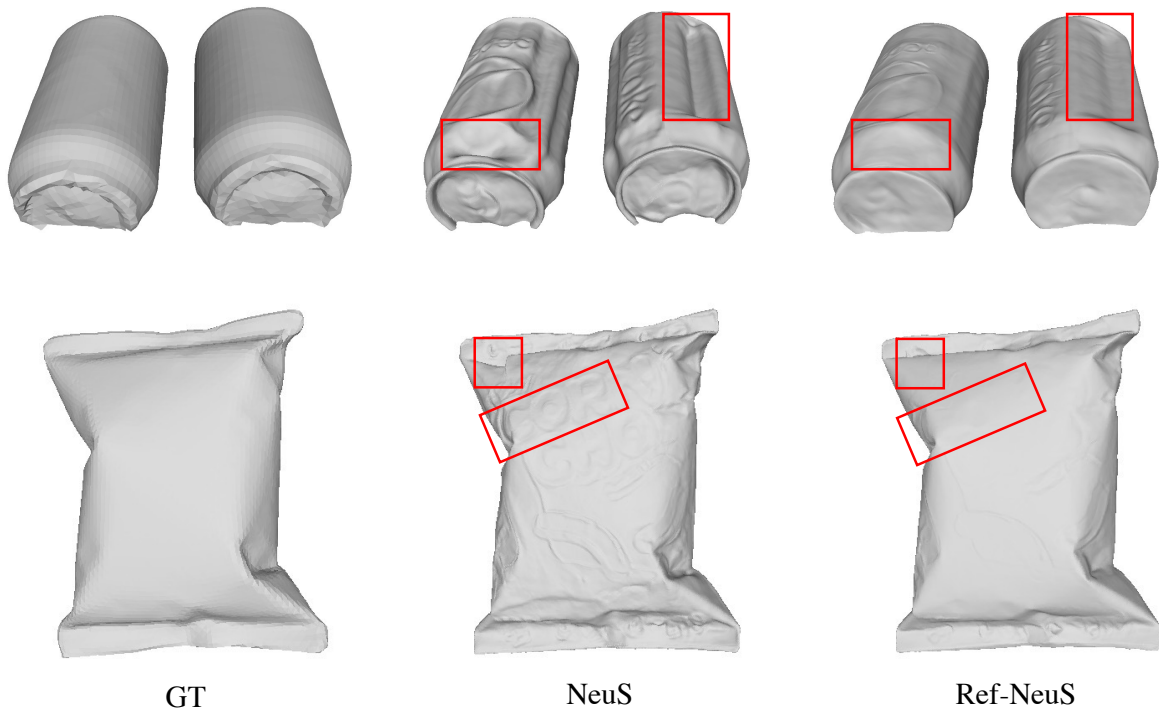Figure 12. Qualitative comparison with NeuS on fish of SLF dataset.

GT　　　　　　　　　　NeuS　　　　　　　　　　Ref-NeuS

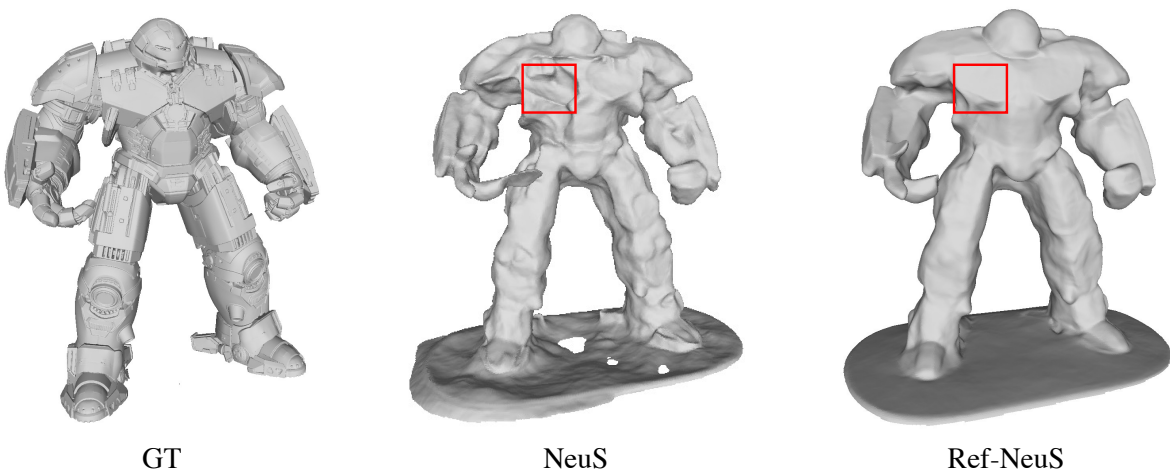Figure 13. Qualitative comparison with NeuS on cans and corncho1 of Bag of Chips dataset.

GT　　　　　　　　　　NeuS　　　　　　　　　　Ref-NeuS

Figure 14. Qualitative comparison with NeuS on Hulk.
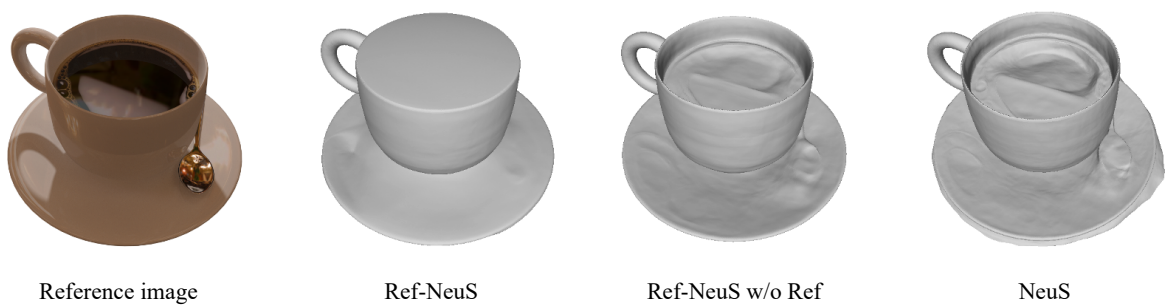

Reference image　　　　Ref-NeuS　　　　Ref-NeuS w/o Ref　　　　NeuS

Figure 15. A failure case on coffee of ShinyBlender.