

# NePF: Neural Photon Field for Single-Stage Inverse Rendering

Tuen-Yue Tsui Qin Zou

School of Computer Science  
Wuhan University

tsui\_tuencyue@whu.edu.cn qzou@whu.edu.cn

## Abstract

We present a novel single-stage framework, *Neural Photon Field* (NePF), to address the ill-posed inverse rendering from multi-view images. Contrary to previous methods that recover the geometry, material, and illumination in multiple stages and extract the properties from various multi-layer perceptrons across different neural fields, we question such complexities and introduce our method - a single-stage framework that uniformly recovers all properties. NePF achieves this unification by fully utilizing the physical implication behind the weight function of neural implicit surfaces and the view-dependent radiance. Moreover, we introduce an innovative coordinate-based illumination model for rapid volume physically-based rendering. To regularize this illumination, we implement the subsurface scattering model for diffuse estimation. We evaluate our method on both real and synthetic datasets. The results demonstrate the superiority of our approach in recovering high-fidelity geometry and visual-plausible material attributes.

## 1. Introduction

Recovering geometry, material, and illumination properties of an object from multi-view images—referred to as inverse rendering [11, 18, 19]—is a fundamental and persistent problem in both computer vision and computer graphics. The challenge arises from the immense solution set inherent to this problem. Early methods try to leverage additional information [1, 16, 23] as prior. In order to tackle these ill-posed problems without prior knowledge, recent techniques adopt a multi-stage strategy or leveraged pre-trained networks [4, 5, 34, 36], recovering distinct properties at different stages. While this multi-stage strategy enables the derivation of plausible results, little attention has been paid to its inherent complexities and limitations.

First, it poses the risk of **accumulated errors** from different stages as shown in Fig. 2, potentially leading to sub-optimal outcomes. Moreover, most previous methods [5, 35, 36] primarily use NeRF or neural implicit repre-

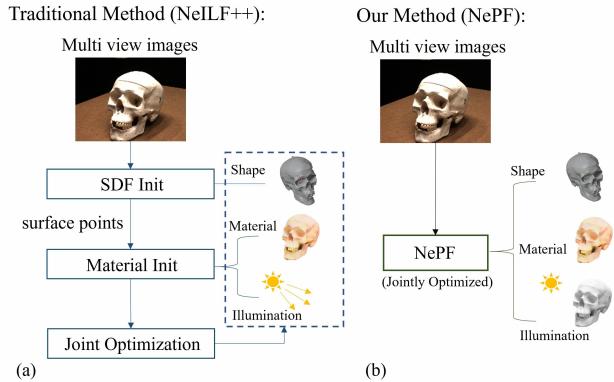


Figure 1. Comparison of the technique pipelines of the proposed single-stage NePF (b) with the SOTA three-stage NeILF++ [34] (a).

sentations [13, 25, 31] only for geometry recovery, overlooking the rich information encapsulated in the outgoing radiance field, resulting in significant **computational waste**. These have highlighted the demand for a unified, single-stage inverse rendering framework.

Appealing to this demand, we question the necessity of a muti-stage solution and introduce **Neural Photon Field**, **NePF**, a single-stage framework for inverse rendering. To address the aforementioned challenges within our method, it becomes imperative to recover the desired properties on the sampled rays together with the geometry density. In this paper, we propose to synchronize the inverse rendering with the outgoing radiance on the rays to ensure the properties are valid. We highlight the physical implications of the NeuS [25] weight functions and view-dependent neural rendering as instrumental in merging the inverse rendering fields (where the surface property is pivotal) with the outgoing radiance field. Our experiments show that it is feasible to recover all the necessary properties by training multiple MLPs based on the same density field.

To bolster the one-stage approach and facilitate rapid



Figure 2. A typical case of accumulated error from NeRFactor [36] when training on DTU scan37 [8]. The pre-trained geometry network extracts the wrong surface and undermines the inverse rendering.

volume physically-based rendering (PBR), we introduce an implicit coordinate-based illumination model for reflectance estimation that outputs both light directions and intensities along with material properties instead of the traditional environment map. To further enhance the accuracy of light directions and intensities predicted, we incorporate subsurface scattering for the diffuse component—which offers directionality in contrast to the conventional Lambertian model. While numerous established methods faltered in extracting the desired properties from intricate real-world scenes such as the DTU dataset [8], our technique proved resilient, generating high fidelity geometry and visual-plausible materials, all within one pass.

The main contributions of this work can be summarized as follows.

- To reduce the accumulated error and computational waste in traditional neural inverse rendering, a single-stage framework is proposed which is capable of jointly optimizing all properties. To the best of our knowledge, this is the first work to address inversing rendering in a single stage manner.
- To accelerate the physically-based rendering in a volumetric setting, a coordinate-based illumination model is developed to directly estimates the lighting directions and intensities.
- To regularize the predicted lighting directions and intensities, subsurface scattering is introduced for material estimation.

## 2. Related Work

### 2.1. Neural Implicit Surfaces for Shape Recovery

Recovering geometry information is a critical step in inverse rendering because it deeply impacts the later material and illumination estimation. Although Neural Radiance Field (NeRF) [13] has shown impressive results in novel view synthesis, it doesn't reconstruct geometry accurately. Some neural rendering methods enhance geometry reconstruction by adding regularizations to the NeRF density field [24, 36], by implementing occupancy field [12, 15, 17, 28], or by training a SDF function via volume rendering [7, 10, 25, 26, 31, 32]. Notably, NeuS (NeuS) [25] has been particu-

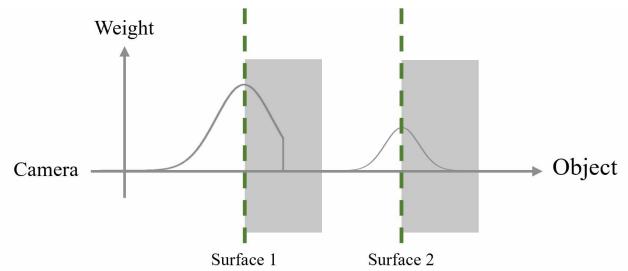


Figure 3. A simple demonstration of NeuS weight function. The weight reaches its maximum at the surface near the camera.

larly effective in reconstructing scene geometry using volume rendering of an SDF density field.

One key innovation in NeuS [25] is its unique weight function, which is unbiased and occlusion-aware. For surfaces closer to the camera, it hits its highest value, making it more significant in the final rendering as presented in Fig. 3. This feature enables NeuS [25] to create realistic 3D geometry from multi-view images. The physical implication behind this weight function led us to integrate geometry recovery into our inverse rendering in a single-stage manner. We can locate the surface property according to the density field by simply querying the max weight. Hence, we choose NeuS [25] as our baseline for geometry recovery. The shape is represented as:

$$\mathcal{S} : \{x \in \mathbb{R}^3 | f(x) = 0\}. \quad (1)$$

### 2.2. Illumination and Material Estimation

Another aspect of inverse rendering is to model the illumination effect to estimate the surface materials. Several previous methods [1, 5, 23] need prior knowledge or pre-trained model of illumination for material estimation. Others assume the illumination to be point/colocated flash lights [2, 3, 21, 27]. Recently, several methods have proved that it is effective to model the illumination by using environmental maps [4, 14, 35, 36]. These recent methods, due to their inherent computational complexity, restrain their application only for surface rendering, making it essential to make surface locating a separate stage. Also, the ill-posed nature of the inverse rendering makes these methods hard to distinguish the environment color and the original color, which leads to intuitively incorrect results.

Recently, Neural Incident Light Field (NeILF) [29] presents a coordinate-based illumination model and achieves satisfying results. However, the neural light field is still separate from the outgoing radiance field, which brings complexity and incoherence. In this work, we come up with a new illumination model that shares the same density field with the outgoing radiance field. This method, by directly regressing the light directions and intensities, enables fast forward rendering in a volumetric setting.

Provided the illumination information in the scene, the material can be estimated by comparing the ground truth rgb with the color of physically-based rendering (PBR) by Bidirectional Reflectance Distribution Function (BRDF) or Bidirectional Scattering Distribution Function (BSDF). Traditional inverse rendering models use BRDF for simplicity. In our work, to better regularize the light directions, we introduce Disney Principled BSDF, which replaces the Lambertian model in BRDF with subsurface scattering, on account of it provides directionality in diffuse part.

### 2.3. Neural Inverse Rendering

Recovering the geometry, material, and illumination from images is an ill-posed task. Recent approaches [5, 34–36] show that it’s possible to recover all these factors using only the true colors in an image as a guide using neural implicit representations.

However, these methods often involve a multi-stage strategy. They first estimate the geometry, then locate the surface coordinates, and finally use these coordinates in further models. PhySG [35] and NeRFactor [36] need an initialization stage for shape reconstruction to extract the surface point, and then perform the rest of the inverse rendering. The best of the SOTA methods: NeILF++ [34], successfully achieves the joint optimization of shape, material and illumination via the inter reflections between surfaces, but it still requires initializations of geometry and material before the final optimization. In our work, we aim to unify these steps. By focusing on the geometric implication of the weight function produced by NeuS [25], we propose to further optimize the outgoing radiance field together with the inverse rendering module all in one pass.

## 3. Methods

Our objective is to extract geometry, material, and illumination for an object in a single-stage manner using a collection of multi-view images taken under unknown lighting conditions. We operate under the assumption that each coordinate in space is predominantly influenced by one major light source, which complies with most indoor real-world scenarios.

### 3.1. NeuS Weight Function

The efficacy of NeuS [25] in geometry recovery is rooted in its distinctive weight function, which is unbiased and occlusion-aware. This weight function reaches its highest value near the camera surface and local peaks at subsequent surfaces, as shown in Fig. 3.

As a result, it retains rich geometric information from the scene. By querying the max weight along the ray, we can easily get the surface PBR result. The weight function

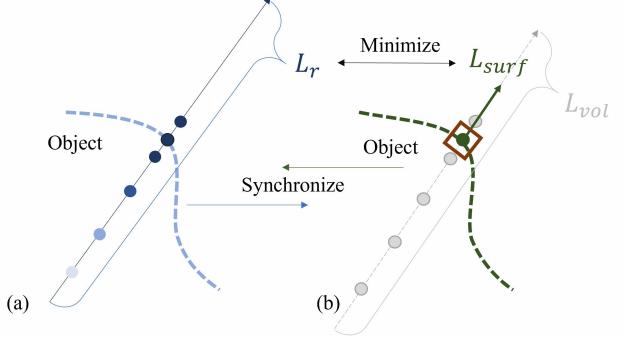


Figure 4. Comparison of the volume rendering and the physical-based rendering. (a) shows that, for the outgoing radiance field, the volumetric radiance is taken into consideration for the final rendering result, while for physical-based rendering (b), only the surface points are considered. The synchronization of this contradicted focus is achieved through supervising the physical-based result (b) under the rendering outcome of (a).

is defined as follows:

$$W(t) = \prod_{j < t} (1 - \alpha(j)) \cdot \alpha(t), \quad (2)$$

$$\alpha(t) = 1 - \exp\left(-\int_{t_i}^{t_{i+1}} p(t) dt\right), \quad (3)$$

where  $p$  refers to the opaque density:

$$p(t) = \max\left(\frac{-\frac{d\phi_s}{dt}(f(x(t)))}{\phi_s(f(x(t)))}, 0\right), \quad (4)$$

where  $\phi_s$  is the logistic density function,  $f(x)$  represents the signed distance function.

### 3.2. Single-Stage Inverse Rendering

While inverse rendering, commonly based on the hard surface assumption, only accounted for surfaces, neural radiance rendering emphasizes volumetric field information. Therefore, the contradicted focal points of the two tasks have kept researchers away from further utilizing the neural radiance field (Fig. 4).

As a result, traditional methods [5, 34–36] usually recover the geometry, or initialize the shape first, then locate the surface points, and finally bridge geometry accuracy with subsequent material and illumination estimated on these surface points (Fig. 1). To unify these steps in one pass, we will have to focus on the volumetric space and recover all properties within the same density field. We will introduce our framework from the direct solution of a single-stage inverse rendering .

**Direct solution:** A direct solution is to train a volumetric

inverse rendering field using ground truth color as supervision, and extract the surface property by querying the max weight along the ray according to Fig. 3. However, it produces inaccurate geometries because of the intricacies of volume rendering and the broad solution space of the inverse rendering, as shown in Fig. 8.

**NePF solution:** To make sure the geometry property is valid, it is essential to train the density field of which the view-dependent outgoing radiance is supervised by the ground truth color. And if we can regress the inverse rendering properties based on the same density field together, then the problems we encounter in direct baseline will be solved. However, several challenges remain.

- How do we regularize the surface properties of inverse rendering in the volume space?
- How do we ensure the inverse rendering field interprets the scene volumetrically in alignment with the outgoing radiance field?

Addressing these challenges requires a connection to relate the two distinct fields, enabling synchronized interpretations on the object. This connection can be found in the physical implications of the rendering outcome by the neural radiance field. The view-dependent characteristic of the outgoing radiance field physically arises from spatially-varying reflectance on surfaces. As such, the outgoing radiance field can serve as a **pseudo ground truth** for the surface property in the inverse rendering field during training. By matching the color  $L_{surf}$  of the surface points  $x_{maxweight}$  with the rendering results  $L_r$  of the outgoing radiance, we ensure the outcome of inverse rendering on the surface is as expected. Also, the selections of these surface points, determined by max weights based on the density field, impose an additional focus on the accurate surfaces that penalize unwanted shape distortions, which is common in the direct baseline. Moreover, by regularizing the surface color, the inverse rendering field is also adjusted, given that the MLPs representing various features are continuous in 3D space. Hence, we synchronize the two distinct neural fields by minimizing the difference between the rendering of view-dependent radiance  $L_r$  and the surface physically-based rendering  $L_{surf}$  as shown in Fig. 5, and we can now train the inverse rendering field concurrently with the outgoing radiance field.

**Shape Refinement:** The geometry extracted solely under the supervision of outgoing radiance is not accurate enough. Since the view-dependent outgoing radiance does not consider specular effects, it could lead to potential errors in shape recovery. Moreover, we find that current neural implicit surfaces methods [25, 31] tend to overfit the textures as shape. Therefore, we propose to further regularize the density field by training the inverse rendering field  $L_{vol}$  under the supervision of ground truth color  $L_{gt}$ . Our experiments in Sec. 4.3 prove that this can help improve the

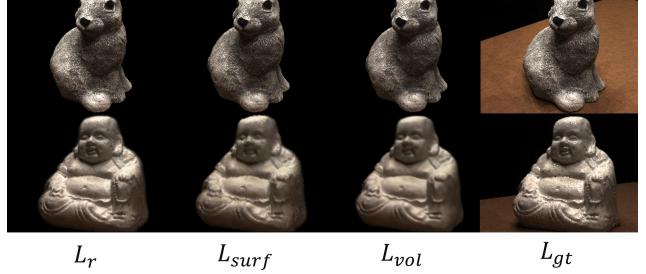


Figure 5. Demonstration of different fields’ synchronizations. The  $L_r$ ,  $L_{surf}$ ,  $L_{vol}$  represent the rendering of the outgoing radiance field, surface by querying max weight and the inverse rendering field. The figure shows that the synchronization of two fields is successfully built, while the surface property is valid.

quality of geometry recovery.

**Loss function:** We calculate three 11 losses for each part, where  $\mathcal{L}_r = \|L_r - L_{gt}\|$ ,  $\mathcal{L}_{surf} = \|L_{surf} - L_r\|$ ,  $\mathcal{L}_{vol} = \|L_{vol} - L_{gt}\|$ . Our overall loss function for color can be defined as follows:

$$\mathcal{L} = \mathcal{L}_r + \lambda_1 * \mathcal{L}_{surf} + \lambda_2 * \mathcal{L}_{vol}. \quad (5)$$

We empirically set  $\lambda_1 = 0.0003$  and  $\lambda_2 = 0.0001$  in our experiments.

**Surface Approaching:** It’s important to note that in actual applications, the structure might not always sample the exact surface points, especially at the early stages of training. However, we’ve observed that this doesn’t compromise the overall performance, with the network progressively focusing on points near the surface due to the importance-sampling mechanism. Still, to help the network converge faster to the surface points, we multiply an additional  $(1 - weight_{max})$  to the  $L_{surf}$ .

### 3.3. Coordinate-based Illumination

We use a straightforward method for illumination. It mirrors the IDR [30] input, where normals and feature vectors are concatenated with coordinates. The network produces a unit-length vector  $l$  calculated by two angles  $\rho$ ,  $\phi$ , representing light directions and a corresponding radiance value  $I$ . The network for illumination is outlined as follows:

$$\mathcal{P} : \{x, n, feature\} \rightarrow \{\rho, \phi, I\}. \quad (6)$$

For clarity, this MLP’s output isn’t related to photon mapping, even though our inspiration came from it. We adopt this name to provide readers with an intuitive grasp. As this illumination is coordinate-driven, its information can be directly applied for efficient volume physically-based rendering.

### 3.4. BSDF Modeling and Material Estimation:

To ensure a valid estimation of lighting direction, we incorporate subsurface scattering in place of the Lambertian

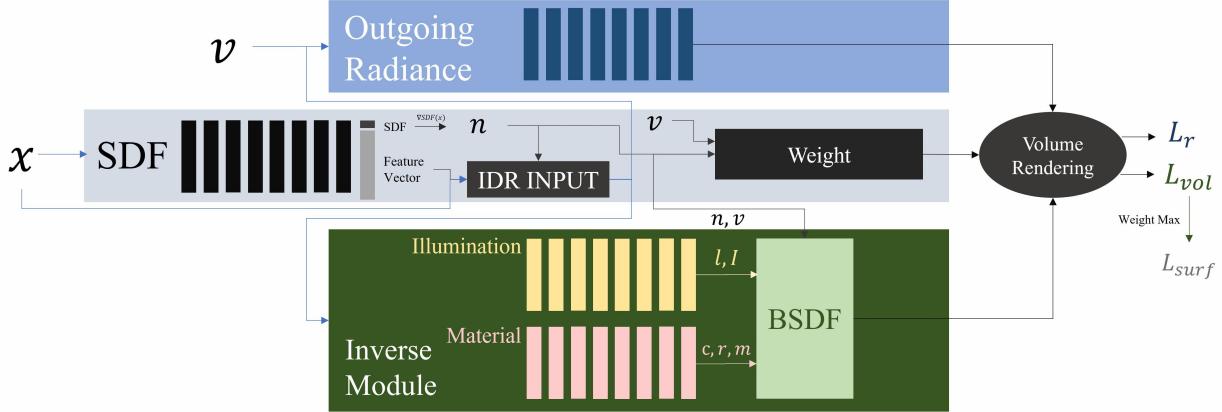


Figure 6. The overall structure of NePF. The SDF network first takes the coordinates  $x$  as input and then outputs the SDF value along with a 256-dimension feature vector. We calculate the gradient of SDF to get the surface normals. Then, we concatenate the coordinates with normals and feature vectors as the input for NePF. The illumination MLP  $\mathcal{P}$  and the Material MLP  $\mathcal{M}$  return the information needed for BSDF estimation. We perform two volume renderings based on the weight function provided by SDF along the ray. We get the PBR outcome on the surface by querying the max weight. The outgoing radiance field remains the same structure as in NeuS [25] with sine activations as in SIREN [22]. More details can be found in supplementary materials.

model. The former offers directional regularization. We utilize the Disney Principled BSDF within our method. The diffuse component is defined as follows:

$$f_d = \frac{c}{\pi} \left(1 - \frac{f_l}{2}\right) \cdot \left(1 - \frac{f_v}{2}\right) + f_{retro}. \quad (7)$$

The  $f_{retro}$  is retro reflection term:

$$f_{retro} = \frac{c}{\pi} \cdot R \cdot [f_l + f_v + f_l \cdot f_v \cdot (R - 1)], \quad (8)$$

where  $f_l, f_v$  are defined by

$$f_l = (1 - n \cdot l)^5, \quad f_v = (1 - n \cdot v)^5, \quad (9)$$

and  $R$  is calculated by

$$R = 2 \cdot r \cdot (h \cdot v)^2. \quad (10)$$

The specular term is computed as follows:

$$f_s = \frac{F(v, h; c, m) \cdot G(l, v; h, r) \cdot D(h, r)}{4 \cdot (n \cdot v) \cdot (n \cdot l)}, \quad (11)$$

where  $F, G, D$  denote the Fresnel term, geometry term, and normal distribution term, respectively. More implementation details can be found in supplementary materials.

In the above equations, we use  $c$  to represent albedo,  $r, m$  to represent roughness and metallic as material attributes,  $l, v, n$  represent light direction, view direction and normals respectively.  $h$  is the half vector of  $l$  and  $v$ .

The same input structure as the illumination part is followed by the material part, producing a 5D vector encompassing roughness, metallic values that both range from [0,

1] and albedo value that ranges from [0, 1]<sup>3</sup>. The material network is defined as follows:

$$\mathcal{M} : \{x, n, feature\} \rightarrow \{c, r, m\}. \quad (12)$$

Feeding these parameters into the BSDF model enables us to conduct physically-based renderings.

### 3.5. Rendering

In our framework, we concurrently execute two volume renderings [9]. The first originates from outgoing radiance field, while the other, rooted in our Neural Photon Field baseline, considers the BSDF for subsurface scattering and specular effects:

$$L_r = \int_{\Omega} W(t) \cdot L(x(t), v) dt, \quad (13)$$

$$L_{vol} = \int_{\Omega} W(t) \cdot f_{BSDF}(l, v, h, n, c, m, r) \cdot I(x(t)) dt. \quad (14)$$

The weight function  $W(t)$  is defined in Sec. 3.1. Our entire approach is illustrated in Fig. 6. In conclusion, our single stage inverse rendering framework NePF can be briefly described as:

- Render the results of outgoing radiance field  $L_r$  and inverse rendering field  $L_{vol}$ , both share the same density field in space.
- Locate the surface PBR result  $L_{surf}$  by querying the max weight.
- Minimize the difference between the  $L_{surf}$  with  $L_r$  to synchronize two fields.(Fig. 4)
- Supervise  $L_r$  and  $L_{vol}$  under the ground truth color to adjust the density field.

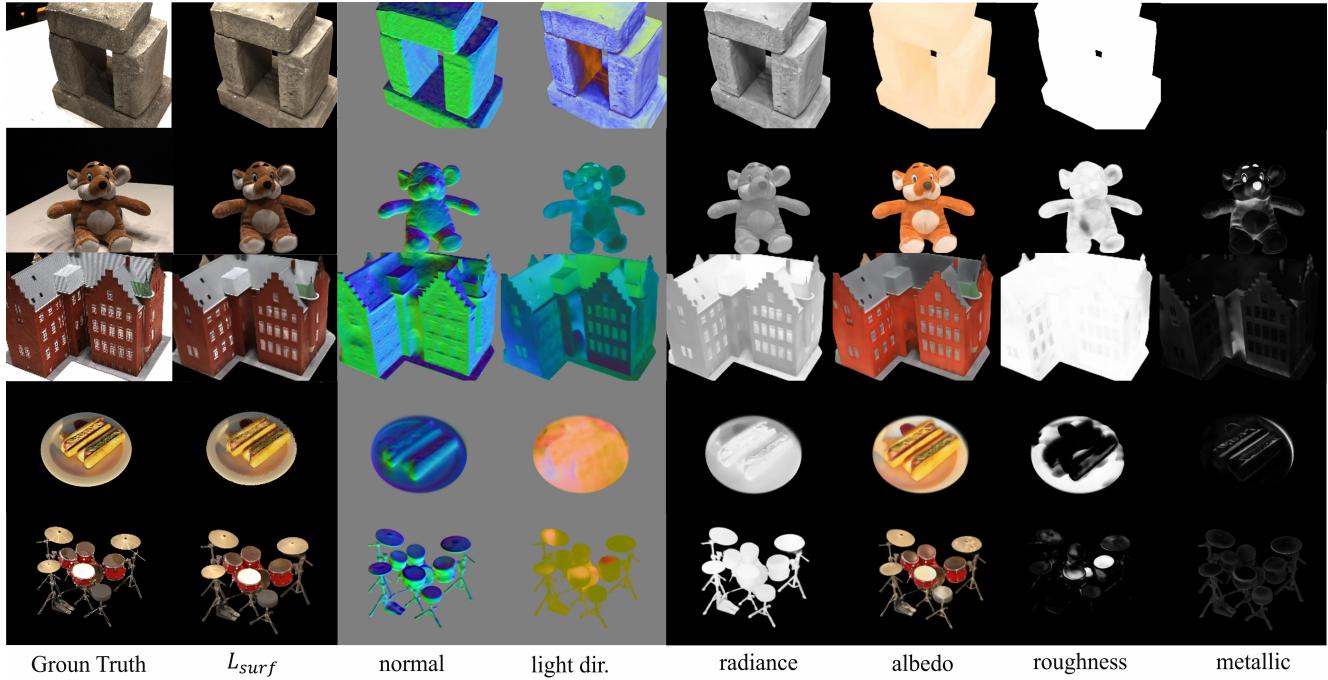


Figure 7. Qualitative results on DTU [8] and NeRF synthetic datasets [13]. The results show that the proposed method successfully recovered the high-fidelity geometry and accurate novel view synthesis.

## 4. Experiments

We test our method on DTU datasets [8] and NeRF synthetic datasets [13]. We further evaluate the significance of some of our key innovations in ablation studies. Since most of the previous methods promise a satisfying result over the synthetic datasets, to signify our improvements, we use the challenging DTU [8] datasets mainly for quantitative and qualitative comparison.

### 4.1. Experimental setting

**Baseline:** We compare our model with several current methods on neural inverse rendering, PhySG, Neural PIL, NeRFactor, and NeILF++ [5, 34–36]. We further test the geometry reconstruction with NeILF++ [34] and NeRF [13] since several SOTA methods [5, 36] use NeRF for geometry recovery.

**Implementation details:** We sample 512 rays per epoch. We train our network with 250000 iterations for around 20 hours on a single Nvidia V100 GPU. More details of the training scheme are in supplementary materials.

### 4.2. PBR Novel View and Shape Reconstruction

We extract the meshes at where  $SDF=0$  and generate the novel view of PBR on the surface by locating the global maxima of weights along the ray.

Since inverse rendering is an ill-posed problem, we compared our model with other SOTA methods in terms

of PBR novel view synthesis for quantitative comparison. We further evaluate the geometry accuracy based on the Chamfer Distance. We also provide the material estimation as well as the illumination estimation in Fig. 7. As Tab. 1 shows, in novel view synthesis, our methods accomplish approximately the same, in some cases even better results comparing to NeILF++ [34], the best SOTA method which implements three stages for inverse rendering, in a single stage manner. Also, in most of the test cases, our method is able to recover high-fidelity geometry. These data prove the efficacy of our single-stage method.

### 4.3. Ablation Studies

**Outgoing Radiance Field Supervision:** As the essence of our method, we explore the necessity of outgoing radiance field supervision in Fig. 8. As the figure shows, in the absence of outgoing radiance supervision, the network wrongly predicts the geometry in the square due to the immense solution set of inverse rendering.

**The Color Loss:** The key to synchronizing the inverse rendering field with the outgoing radiance field is to supervise the surface PBR results  $L_{surf}$  with the volume rendering of outgoing radiance  $L_r$ . To explain why we have to supervise the surface PBR results with the  $L_r$  during training instead of the ground truth RGB  $L_{gb}$ , we test the model with the surface PBR  $L_{surf}$  supervised under different pseudo ground

Table 1. Quantitative results obtained by the proposed method and the comparison methods on DTU scans [8]. The best one is marked as red, while the second and third are marked as orange and yellow respectively, in novel view synthesis chart. Our framework successfully ties the best of SOTA methods, NeILF++ [34] in terms of PBR novel view synthesis. Although our method cannot recover high quality geometry compare to NeILF++ [34] on average, in certain cases, we show potential to outperform the best SOTA method. The results of other methods are from NeILF++ [34].

Task	Methods	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
Novel View (PSNR↑)	PhySG [35]	15.11	17.31	17.38	20.65	18.71	18.89	18.47	18.08	21.98	20.67	18.75	17.55	21.20	18.78	23.16	19.11
	Neural PIL [5]	19.51	19.88	20.67	19.12	21.01	23.70	18.94	17.05	20.54	19.67	18.20	17.75	21.38	21.69	-	19.94
	NeRFactor [36]	21.91	20.45	23.24	23.33	26.86	22.70	24.71	27.59	22.56	25.08	26.30	25.14	21.35	26.44	26.53	24.28
	NeILF++ [34]	24.17	24.60	26.40	27.24	29.85	28.16	27.39	29.82	25.50	28.19	31.84	30.20	27.71	30.87	33.62	28.37
	Ours	25.66	21.88	28.34	27.14	29.29	27.92	28.94	32.00	26.35	28.27	30.33	27.64	26.49	28.82	31.08	28.01
Geometry (Chamfer Distance↓)	NeRF [13]	1.920	1.730	1.920	0.800	3.410	1.390	1.510	5.440	2.040	1.100	1.010	2.880	0.910	1.000	0.790	1.857
	NeILF++ [34]	1.303	1.911	0.954	0.595	1.271	0.881	0.860	1.529	1.242	1.125	0.723	2.096	0.488	0.838	0.654	1.098
	Ours	1.716	3.474	<b>0.724</b>	<b>0.413</b>	2.222	1.064	1.801	1.807	2.078	1.132	<b>0.600</b>	2.432	0.760	1.099	0.740	1.472

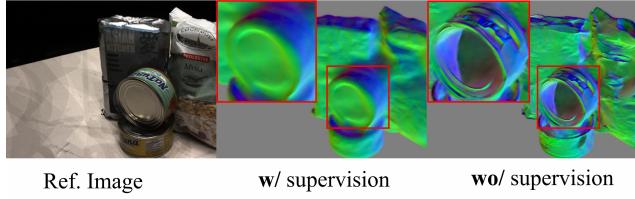


Figure 8. Qualitative study on outgoing radiance supervision. The geometry recovered without the outgoing radiance supervision is unreliable.

Table 2. Quantitative study on two ways of supervision on  $L_{surf}$  on DTU scan83 [8]. In terms of PBR novel view synthesis and geometry recovery, supervised under the ground truth color  $L_{gt}$  leads to worse results.

sup. methods	Novel View (PSNR)↑	Geometry (CD)↓
$L_{gt}$	27.9577	1.8913
$L_r$	<b>31.9960</b>	<b>1.8074</b>

truth in Fig. 9 and Tab. 2. As presented in both Fig. 9 and Tab. 2, supervising the surface loss  $L_{surf}$  with the radiance field loss  $L_r$  yields improved outcome. This enhancement can be attributed to the fact that the ground truth color encompasses a broader spectrum of high-frequency details when compares to the radiance field’s volume rendering, especially during the early phases of training. Given that the density field’s regression is predominantly dependent on the radiance field’s volume rendering results, supervising the  $L_{surf}$  under the  $L_{gt}$  leads to a larger overall loss that will affect the shape recovery, as well as the later PBR, then supervising under the  $L_r$ .

Also, to better understand the role of each component in our color loss, we test the scenarios where the surface loss  $L_{surf}$  and the volume loss  $L_{vol}$  are not used in Fig. 10,

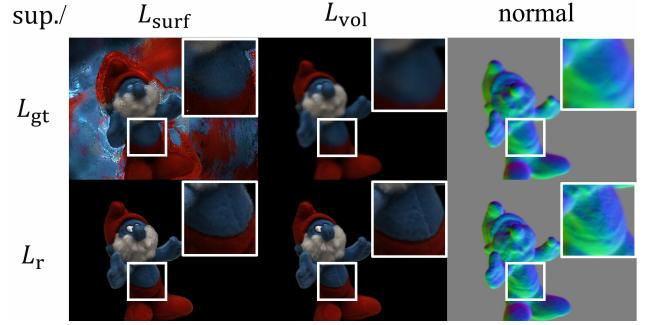


Figure 9. Qualitative study on two ways of supervision on  $L_{surf}$  on DTU scan83 [8]. Despite the apparent overfitting in the background, when supervised under the ground truth, the overall novel view PBR quality is significantly lower, as well as the geometry recovery.



Figure 10. Qualitative study on surface loss  $L_{surf}$ . As shown in the figure, without the surface loss, the  $L_{surf}$  remains noisy while  $L_{vol}$  already converges.

Tab. 3 and Fig. 11.

The results in the Fig. 10 suggest that without the surface loss, the surface PBR, due to the nature of volume rendering, won’t provide us with the desired results. While the Tab. 3 proves that supervising the volume rendering of inverse rendering field under the ground truth can help regularize the density field by considering the specular effects and textures.

Table 3. Quantitative study on the effect of volume loss  $\mathcal{L}_{vol}$  on the DTU datasets [8]. As the table demonstrates, without the volume loss, the chamfer distance of the geometry recovery is significantly higher.

Methods	scan40	scan63	scan97
w/ $\mathcal{L}_{vol}$	<b>0.724</b>	<b>2.222</b>	<b>2.078</b>
wo/ $\mathcal{L}_{vol}$	0.875	2.909	2.577

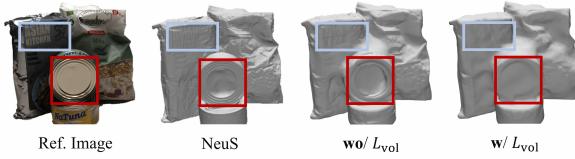


Figure 11. Qualitative study on the effect of volume loss  $\mathcal{L}_{vol}$ . Our method is able to distinguish the textures and specular effects from shape.

**Subsurface scattering:** We also test the role subsurface scattering plays in our framework. Although we do not have a ground truth of illumination for comparison, we choose DTU scan40 [8] for experiments because the framework predicts no specular effect on this scan so that the light directions are solely affected by the diffuse part. Fig. 12 shows that with subsurface scattering, the framework can demonstrate more complex lighting environments. Intuitively, the white square and the yellow square in the images, though share the same normals, should have different lighting directions due to the occlusion and shades. Fig. 12 proves that with subsurface scattering, the framework can tell such difference while for the traditional Lambertian model, the lighting directions are only dependent on the surface normals due to our MLP design.

**Accumulated errors:** To visualize the improvements of our method on alleviating the accumulated errors, we compared NePF with NeILF++ [34] and NeRFactor [36] in Fig. 13.

## 5. Limitations

While NePF ties the state-of-the-art methods in a single-stage manner, it relies on specific assumptions that limit complex lighting modeling. Future research might explore enhancing the robustness of our current coordinate-based lighting model. Additionally, there is still room for further advancements in geometry recovery, as our method finds difficulties in cases where the geometry variation is significant.

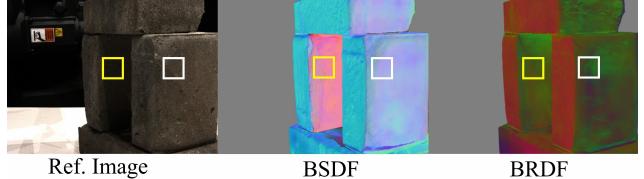


Figure 12. Qualitative study on subsurface scattering. The images show the light directions under the BSDF and BRDF models. The white square and the yellow square are under different illumination contexts due to the occlusion. While the BSDF model can tell such differences, the BRDF fails to do so.

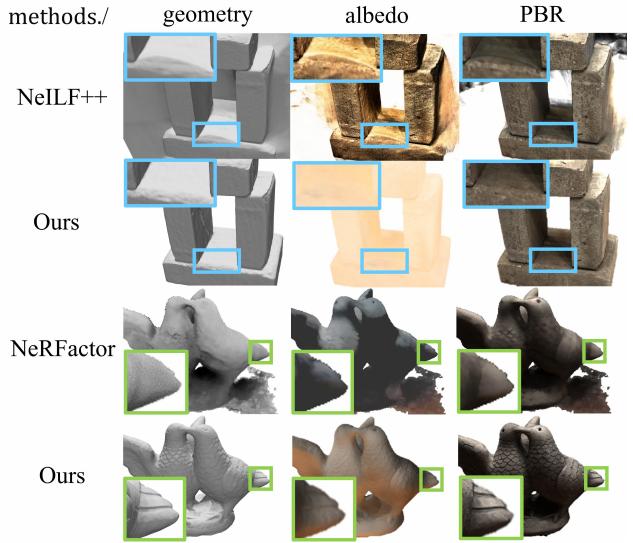


Figure 13. Qualitative study on accumulated errors by comparing our method with the SOTA methods. Our method not only addresses the accumulated errors that occur in the previous methods, but also yields a more plausible albedo.

## 6. Conclusions

In this paper, we've proposed Neural Photon Field (NePF), a novel single-stage framework for the traditional multi-pass inverse rendering. By fully employing the physical meaning of the weight function of neural implicit surfaces and the outgoing radiance, we are able to recover all the desired properties in a single pass. Furthermore, we develop a novel coordinate-based illumination model that enhances the proposed one-pass technique. Our method is evaluated on both real-world and synthetic datasets. The results demonstrate that the proposed method obtains high-fidelity geometry, high-quality novel view synthesis, and alleviates the problem of accumulated errors and high computational waste in the traditional methods. We consider this work as a significant advancement in the field of inverse rendering.

## References

- [1] Sai Bi, Zexiang Xu, Pratul P. Srinivasan, Ben Mildenhall, Kalyan Sunkavalli, Milos Hasan, Yannick Hold-Geoffroy, David J. Kriegman, and Ravi Ramamoorthi. Neural reflectance fields for appearance acquisition. *Corr*, abs/2008.03824, 2020. [1](#), [2](#)
- [2] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, Milos Hasan, Yannick Hold-Geoffroy, David J. Kriegman, and Ravi Ramamoorthi. Deep reflectance volumes: Relightable reconstructions from multi-view photometric images. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part III*, pages 294–311. Springer, 2020. [2](#)
- [3] Sai Bi, Zexiang Xu, Kalyan Sunkavalli, David J. Kriegman, and Ravi Ramamoorthi. Deep 3d capture: Geometry and reflectance from sparse multi-view images. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 5959–5968. Computer Vision Foundation / IEEE, 2020. [2](#)
- [4] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T. Barron, Ce Liu, and Hendrik P. A. Lensch. Nerd: Neural reflectance decomposition from image collections. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 12664–12674. IEEE, 2021. [1](#), [2](#)
- [5] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan T. Barron, and Hendrik P. A. Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 10691–10704, 2021. [1](#), [2](#), [3](#), [6](#), [7](#)
- [6] Blender Online Community. Blender – a 3d modelling and rendering package. Blender Foundation, Stichting Blender Foundation, Amsterdam. Software available at <https://www.blender.org/>, 2023. Version 3.6. [2](#)
- [7] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *CVPR*, 2022. [2](#)
- [8] Rasmus Ramsbøl Jensen, Anders Lindbjerg Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 406–413. IEEE Computer Society, 2014. [2](#), [6](#), [7](#), [8](#)
- [9] James T. Kajiya. The rendering equation. In *13th annual conference on Computer graphics and interactive techniques*, 1986. [5](#)
- [10] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *CVPR*, 2023. [2](#)
- [11] Steve Marschner. *Inverse Rendering for Computer Graphics*. PhD thesis, Cornell University, USA, 1998. [1](#)
- [12] Lars M. Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*, pages 4460–4470. Computer Vision Foundation / IEEE, 2019. [2](#)
- [13] 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. In *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, pages 405–421. Springer, 2020. [1](#), [2](#), [6](#), [7](#)
- [14] Jacob Munkberg, Wenzheng Chen, Jon Hasselgren, Alex Evans, Tianchang Shen, Thomas Müller, Jun Gao, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 8270–8280. IEEE, 2022. [2](#)
- [15] Michael Niemeyer, Lars M. Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3501–3512. Computer Vision Foundation / IEEE, 2020. [2](#)
- [16] Merlin Nimier-David, Zhao Dong, Wenzel Jakob, and Anton Kaplanyan. Material and lighting reconstruction for complex indoor scenes with texture-space differentiable rendering. In *32nd Eurographics Symposium on Rendering, EGSR 2021 - Digital Library Only Track, Saarbrücken, Germany, June 29 - July 2, 2021*, pages 73–84. Eurographics Association, 2021. [1](#)
- [17] Michael Oechsle, Songyou Peng, and Andreas Geiger. UNISURF: unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 5569–5579. IEEE, 2021. [2](#)
- [18] Ravi Ramamoorthi and Pat Hanrahan. A signal-processing framework for inverse rendering. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 2001, Los Angeles, California, USA, August 12-17, 2001*, pages 117–128. ACM, 2001. [1](#)
- [19] Yoichi Sato, Mark D. Wheeler, and Katsushi Ikeuchi. Object shape and reflectance modeling from observation. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997, Los Angeles, CA, USA, August 3-8, 1997*, pages 379–387. ACM, 1997. [1](#)
- [20] Christophe Schlick. An inexpensive BRDF model for physically-based rendering. *Comput. Graph. Forum*, 13(3): 233–246, 1994. [1](#)
- [21] Carolin Schmitt, Simon Donné, Gernot Riegler, Vladlen Koltun, and Andreas Geiger. On joint estimation of pose, geometry and svbrdf from a handheld scanner. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19*

- 2020, pages 3490–3500. Computer Vision Foundation / IEEE, 2020. 2
- [22] Vincent Sitzmann, Julien N. P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 5, 1
- [23] 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 *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 7495–7504. Computer Vision Foundation / IEEE, 2021. 1, 2
- [24] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd E. Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 5481–5490. IEEE, 2022. 2
- [25] 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 *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 27171–27183, 2021. 1, 2, 3, 4, 5
- [26] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. Pet-neus: Positional encoding tri-planes for neural surfaces. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17-24, 2023*, pages 12598–12607. IEEE, 2023. 2
- [27] Rui Xia, Yue Dong, Pieter Peers, and Xin Tong. Recovering shape and spatially-varying surface reflectance under unknown illumination. *ACM Trans. Graph.*, 35(6):187:1–187:12, 2016. 2
- [28] Wenqi Yang, Guanying Chen, Chaofeng Chen, Zhenfang Chen, and Kwan-Yee K. Wong.  $S^3$ -nerf: Neural reflectance field from shading and shadow under a single viewpoint. In *NeurIPS*, 2022. 2
- [29] Yao Yao, Jingyang Zhang, Jingbo Liu, Yihang Qu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Neilf: Neural incident light field for physically-based material estimation. In *Computer Vision - ECCV 2022 - 17th European Conference, Tel Aviv, Israel, October 23-27, 2022, Proceedings, Part XXXI*, pages 700–716. Springer, 2022. 2
- [30] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Ronen Basri, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 4
- [31] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 4805–4815, 2021. 1, 2, 4
- [32] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 6505–6514. IEEE, 2021. 2
- [33] Jingyang Zhang, Yao Yao, Shiwei Li, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Critical regularizations for neural surface reconstruction in the wild. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 6260–6269. IEEE, 2022. 1
- [34] Jingyang Zhang, Yao Yao, Shiwei Li, Jingbo Liu, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Neilf++: Inter-reflectable light fields for geometry and material estimation. *ICCV*, 2023. 1, 3, 6, 7, 8
- [35] 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 *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 5453–5462. Computer Vision Foundation / IEEE, 2021. 1, 2, 3, 7
- [36] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul E. Debevec, William T. Freeman, and Jonathan T. Barron. Nerfactor: neural factorization of shape and reflectance under an unknown illumination. *ACM Trans. Graph.*, 40(6):237:1–237:18, 2021. 1, 2, 3, 6, 7, 8

# NePF: Neural Photon Field for Single-Stage Inverse Rendering

## Supplementary Material

### 7. Network Structure

We use 8 layers SIREN [22] MLP with 256 width for outgoing radiance network and material network, while the photon network and the SDF network are relu activated with a depth of 8 and a width of 256. Detailed architecture is presented in Fig. 14

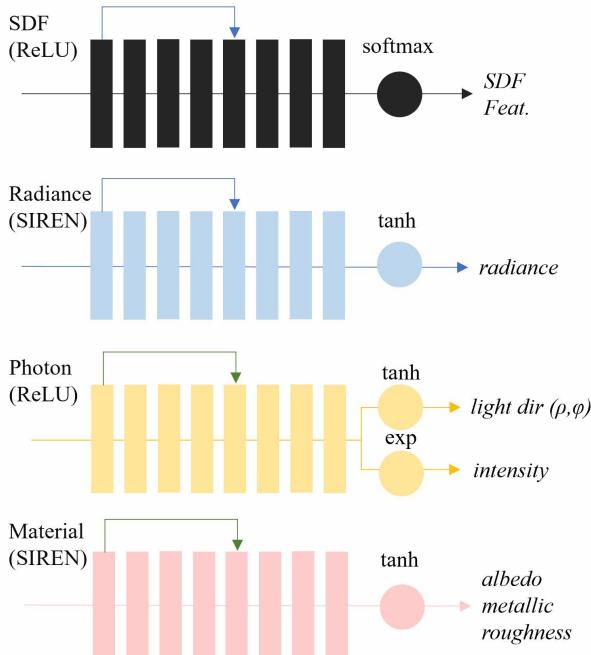


Figure 14. Detailed network structure.

## 8. Trainning scheme

### 8.1. Loss functions

We use several loss functions during the trainning.

#### 8.1.1 Color loss

The color loss has three parts that have been well described in Sec. 3.

$$\mathcal{L} = \mathcal{L}_r + \lambda_1 * \mathcal{L}_{surf} + \lambda_2 * \mathcal{L}_{vol}. \quad (15)$$

#### 8.1.2 Geo loss

The geometry loss has two parts: The eikonal loss and the hessian loss.

**Eikonal loss** This is a typical loss function for SDF training that encourages the expectation of gradient to be 1.

$$L_{eikonal} = |1 - \|\nabla f(x)\||. \quad (16)$$

**Hessian loss** This loss function encourages a smoother reconstructed surface by minimizing the Hessian Matrix [33], which is the second derivative of the SDF.

$$L_{hessian} = |\nabla^2 f(x)|. \quad (17)$$

### 8.1.3 Light variance loss

For a smoother light direction prediction, we implement a light variance loss that impose a penalty on highly spatially-various illumination prediction.

$$L_{light} = \lambda_3 * |var(x) - var(I)| + \quad (18)$$

$$\lambda_4 * |var(n) - var(l)| + \quad (19)$$

$$\lambda_5 * |var(x) - var(l)|. \quad (20)$$

We use  $x$  for coordinates,  $n$  for surface normals,  $I$  for light intensities and  $l$  for light directions. The aim of this loss function is to assume that the illumination should encompass a local consistency. We assume that the variance of coordinates affect the variance of intensities, while both the variance of coordinates and the surface normals will affect the variance of light directions, though the former contribute more. We set  $\lambda_3$  and  $\lambda_4$  as 1.0 while  $\lambda_5$  as 0.1. The weight for the overall loss is tiny during the trainning. We use 0.0001 to prevent it affecting the overall trainning quality.

### 8.1.4 Mask loss

We run our method under the mask supervision. We use the typical BCE loss for mask supervision:

$$L_{mask} = BCE(weight_{sum}, mask). \quad (21)$$

## 8.2. Learning Rate

We implement the same anneal strategy of learning rate in NeuS [25]. However, we set our base learning rate as 0.0003 for stability.

## 8.3. Specular Terms

We implement Schlick PBR [20] to model the specular part in BSDF estimation. For the Fresnel term:

$$F_0 = 0.04 \cdot (1 - m) + c \cdot m, \quad (22)$$

$$F(v, h) = F_0 + (1 - F_0)(1 - h \cdot v)^5. \quad (23)$$

For the geometry term:

$$G1(v) = \frac{n \cdot v}{2 \cdot [\alpha + (1 - \alpha)(n \cdot v)]}, \quad (24)$$

$$G2(l) = \frac{n \cdot l}{2 \cdot [\alpha + (1 - \alpha)(n \cdot l)]}, \quad (25)$$

$$G(l, v) = G1(v) \cdot G2(l). \quad (26)$$

And for the normal distribution term:

$$D(h) = \frac{\alpha^2}{\pi \cdot ((n \cdot h)^2(\alpha^2 - 1) + 1)^2}. \quad (27)$$

The  $\alpha$  in the above equations are defined as the square of roughness.

## 9. Synchronization

To visualize the synchronization during the training process, we validate the PSNR of rendering results of outgoing radiance field, inverse rendering field and surface PBR every 2500 epoch in Fig. 15

## 10. Relighting and Material Editing

We extract the meshes as well as its BSDF properties on the vertex and import them in Blender [6] to demonstrate relighting and material editing Fig. 16.

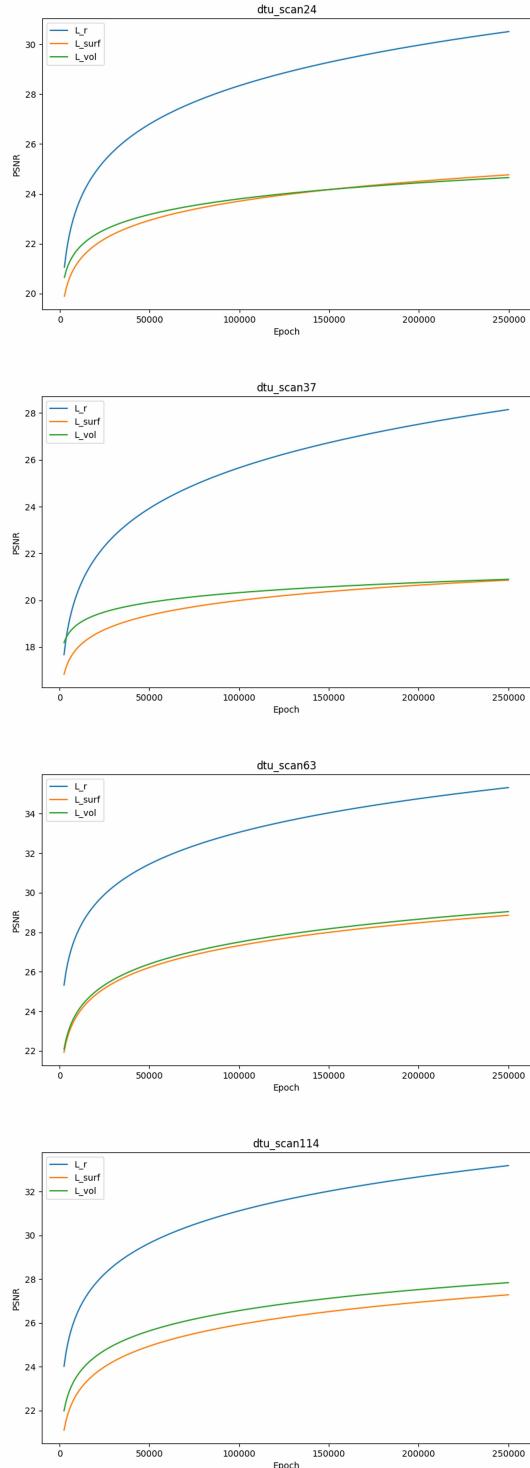


Figure 15. Demonstrations of synchronization during the training process. The horizontal axis represents the epoch, while the vertical axis denotes the PSNR. The blue, orange, and green lines correspond to  $L_r$ ,  $L_{surf}$ ,  $L_{vol}$  respectively. Four cases, scan24, scan37, scan63, and scan114 from DTU datasets [8] are illustrated from top to bottom.



Figure 16. Demonstrations of relighting and material editing effects. The left column is relighting, while the right column is material editing.