Factorized Inverse Path Tracing for Efficient and Accurate Material-Lighting Estimation

Liwen Wu[1*] Rui Zhu[1*] Mustafa B. Yaldiz[1] Yinhao Zhu [2] Hong Cai [2] Janarbek Matai [2]
Fatih Porikli [2] Tzu-Mao Li [1] Manmohan Chandraker [1] Ravi Ramamoorthi [1]
[1]UC San Diego [2]Qualcomm AI Research
{liw026,rzhu,myaldiz,tzli,mkchandraker,ravir}@ucsd.edu
{yinhaoz,hongcai,jmatai,fporikli}@qti.qualcomm.com

Abstract

Inverse path tracing has recently been applied to joint material and lighting estimation, given geometry and multi-view HDR observations of an indoor scene. However, it has two major limitations: path tracing is expensive to compute, and ambiguities exist between reflection and emission. Our Factorized Inverse Path Tracing (FIPT) addresses these challenges by using a factored light transport formulation and finds emitters driven by rendering errors. Our algorithm enables accurate material and lighting optimization faster than previous work, and is more effective at resolving ambiguities. The exhaustive experiments on synthetic scenes show that our method (1) outperforms state-of-the-art indoor inverse rendering and relighting methods particularly in the presence of complex illumination effects; (2) speeds up inverse path tracing optimization to less than an hour. We further demonstrate robustness to noisy inputs through material and lighting estimates that allow plausible relighting in a real scene. The source code is available at: https://github.com/lwwu2/fipt

1. Introduction

We address the task of estimating the materials and lighting of an indoor scene based on image observations (Fig. 1). Recent work has shown that optimizing per-scene material and emission profiles through photometric loss and a differentiable renderer, with geometry reconstructed with the existing 3D reconstruction algorithms [35, 26, 47], can lead to promising results [1, 29, 45]. However, key challenges remain unsolved in these methods: (1) they require expensive Monte Carlo estimation for both the loss and derivative evaluations; (2) inherent ambiguity exists between material and lighting, and this ill-posed inverse problem hinders the optimization. We present an alternative inverse rendering algorithm that outperforms the state-of-the-art in terms of both efficiency and accuracy.

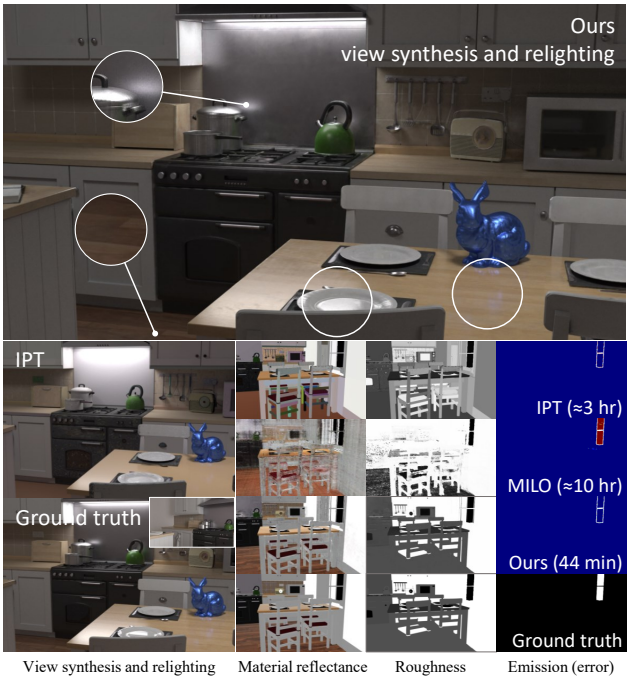


Figure 1: **Ours vs standard IPT.** IPT [1] takes a piecewise constant parameterization of material to reduce Monte Carlo variance and ambiguity for inverse rendering, losing fine spatial details as a result. Directly extending it to complex material representation (*e.g.* MILO [45]) shows very slow convergence. In contrast, we propose Factorized Inverse Path Tracing (FIPT) to get rid of variance and reduce ambiguities, yielding efficient and high quality BRDF and emission (4th row), appealing relighting (1st row), and object insertion (the bunny on the table). The presented scene is synthetic with the inset showing the input (lower-left sub-figure). We further showcase results on real scenes in Fig. 10 and 11.

Optimizing scene parameters with Monte Carlo differentiable rendering can suffer from high variance and lead to slow convergence. Inspired by classical irradiance caching literature [43], our key idea to address this challenge is to factorize the material term out of the rendering integral and

*Equal contribution

bake the incoming radiance to significantly speed up inverse rendering. Unlike prior work which also applies a similar factorization (*e.g.* [32, 22]) but does not consider view-dependent reflections, our method extends to general specular materials and both local and global illumination.

To address the ill-posed nature of joint optimization of material and lighting, we observe that by taking out the emission term in the rendering equation for the first bounce, only emissive surfaces will have high rendering loss. This observation allows us to design an effective way to detect emitters. We incorporate our emitter detection method into a full inverse rendering pipeline and independently estimate the emission after emitter detection.

Overall, our method achieves fast convergence over the material-lighting estimation task thanks to our factorized light transport formulation and emitter extraction strategy (Fig. 1). To demonstrate accurate BRDF-emission comparison, we perform exhaustive experiments on synthetic scenes (Sec. 5.1, 5.2) while also validating on noisy data of captured real scenes (Sec. 5.3). The results show our method is able to obtain high-quality reconstruction for complicated indoor scenes that can easily fail for the state-of-the-art (Tab. 2), yet the training speed is 4-10 times faster (Tab. 3).

2. Related Work

Inverse rendering. Inverse rendering aims to estimate the intrinsic properties of an observed scene, via decoupling material, geometry and lighting which jointly contribute to image appearance. Given the inherent ambiguity between the aforementioned high-dimensional factors, classical methods seek to regularize the solution with a surface rendering objective. Approaches include a low-dimensional surface reflectance representation [46], sparsity priors for intrinsic images [4], and spherical-harmonics-based lighting representation [25]. These methods rely on simplified representation of material or lighting, and their regression-based nature calls for heuristic-based priors which may not be appropriate for a wide variety of scenes.

Earlier work can already photorealistically render synthetic objects in a photograph by estimating lighting and geometry [11, 16, 17]. These methods do not retrieve the materials of the scene, and thus cannot show the reflection of the object on a specular surface in the scene.

Learning-based methods. Learning-based approaches leverage priors learned from datasets. These methods typically take a single image [36, 20, 51, 42, 21] or a pair of stereo images [39], and apply deep learning models to predict spatially-varying materials and lighting. Although learned priors help to regularize individual components, these methods do not explicitly model the physics of global light transport and have to rely on approximated inference [24].

Philip *et al*. [32] take multiple images and aggregate multiview irradiance and albedo information to a pre-trained network to synthesize the relit image. The network takes physically rendered shading using light sources that are semi-automatically estimated as inputs, and outputs an image after relighting. We show in the results that in our synthetic scenes, their method's reliance on the network to render the final image can lead to undesired artifacts, while our use of a physically-based renderer delivers more realistic images.

$(\cdot)_+$	dot product clamped to positive value
$\boldsymbol{\omega}_i$	incident (light) direction
$\boldsymbol{\omega}_o$	outgoing (viewing) direction
$\mathbf{h}$	half vector: $(\boldsymbol{\omega}_i + \boldsymbol{\omega}_o)/\|\boldsymbol{\omega}_i + \boldsymbol{\omega}_o\|_2$
$\mathbf{n}$	surface normal
$\mathbf{a}(\mathbf{x})$	surface base color
$m(\mathbf{x})$	surface metallic
$\sigma(\mathbf{x})$	surface roughness
$\mathbf{k}_d(\mathbf{x})$	diffuse reflectance: $\mathbf{a}(\mathbf{x})(1 - m(\mathbf{x}))$
$\mathbf{k}_s(\mathbf{x})$	specular reflectance: $\mathbf{a}(\mathbf{x})m(\mathbf{x}) + 0.04(1 - m(\mathbf{x}))$
$D(\cdot)$	GGX normal distribution [41]
$F(\cdot)$	Schlick's approximation of Fresnel coefficient [34]
$G(\cdot)$	Geometry (Shadow-Masking) term [41]

Table 1: **Notations**

Local or distant lighting. Many recent methods aim to model a specific form of light transport. Some methods focus on a single object or distant illumination (environment map) [12, 49, 28, 7, 8, 50]. Srinivasan *et al*. [38] model two-bounce volumetric lighting with known light sources, and Yao *et al*. [44] represent incident radiance as a 5D network. However, optimization of spatially-varying lighting without physically-based constraints is extremely ill-posed especially without abundant observation of light sources. Moreover, object-centric methods do not trivially generalize to indoor settings, where complex lighting effects including occlusion, inter-reflections, and directional highlights call for modeling of long-range interactions of lighting and scene properties.

Global light transport. Most related to our work, to model general global light transport, recent methods [1, 29, 45] build on a per-scene optimization pipeline using a differentiable path tracer [19, 3, 30, 48]. These methods jointly optimize material and lighting along extensively sampled light paths, and thus are subject to incorrect and slow convergence and high variance due to expensive path queries, gradient propagation, and Monte Carlo sampling, as well as the inherent ambiguity between materials and lighting. We propose an inverse rendering pipeline that models the global light transport, but converges significantly faster and more accurately than existing methods. Our variance reduction technique using light baking is inspired by classical rendering methods [43, 18, 37], and we tightly integrate the technique in an inverse rendering pipeline. A concurrent work TexIR [23] adopts similar ideas to ours by using a pre-baked irradiance as HDR texture map to recover scene materials. However, they do not model view-dependent light transport and do not estimate emission.

3. Background

Given posed HDR image captures of an indoor scene, our method builds upon input mesh or existing 3D reconstruction algorithms (*e.g.* MonoSDF [47]) to further estimate the

material and lighting of the scene. To ensure the problem is well-constrained, we make similar assumptions about scene acquisition as in previous works [1, 32, 45] that the dominant light sources and most of the scene geometry are observed in input images.

The material is described as a spatially varying BRDF [15] (including the cosine term) with notations specified in Tab. 1:

$$f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = \frac{\mathbf{k}_d(\mathbf{x})}{\pi} (\mathbf{n} \cdot \boldsymbol{\omega}_i)_+ + \frac{F(\boldsymbol{\omega}_i, \mathbf{h}, \mathbf{k}_s(\mathbf{x})) D(\mathbf{h}, \mathbf{n}, \sigma(\mathbf{x})) G(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o, \mathbf{n}, \sigma(\mathbf{x}))}{4(\mathbf{n} \cdot \boldsymbol{\omega}_o)}, \quad (1)$$

where $\mathbf{k}_d = \mathbf{a}(1-m)$ and $\mathbf{k}_s = 0.04(1-m) + \mathbf{a}m$ are the diffuse and specular reflectance with base color $\mathbf{a}$ and metallic m controlling the two coefficients. The emitted light is assumed to be view-independent across the surface: $\mathbf{L}_e(\mathbf{x}, \boldsymbol{\omega}_o) = \mathbf{L}_e(\mathbf{x})$, which generalizes well to the emission profile of the indoor scene (extension to more complex emitters is possible; see Sec. 5.4).

With the parameterization above, our goal is to find $\mathbf{a}, \sigma, m, \mathbf{L}_e$ that minimize the difference of renderings with respect to the ground truth over the training images:

$$\min_{\mathbf{a},\sigma,m,\mathbf{L}_e} \sum_{\mathbf{x},\boldsymbol{\omega}_o} \|\mathbf{L}_o(\mathbf{x}, \boldsymbol{\omega}_o) - \mathbf{L}_{gt}(\mathbf{x}, \boldsymbol{\omega}_o)\|_2^2 \quad (2)$$

$$\mathbf{L}_o(\mathbf{x}, \boldsymbol{\omega}_o) = \mathbf{L}_e(\mathbf{x}, \boldsymbol{\omega}_o) + \mathbf{L}_r(\mathbf{x}, \boldsymbol{\omega}_o) \quad (3)$$

$$\mathbf{L}_r(\mathbf{x}, \boldsymbol{\omega}_o) = \int_{\Omega^+} \mathbf{L}_i(\mathbf{x}, \boldsymbol{\omega}_i) f(\mathbf{x}, \boldsymbol{\omega}_i, \boldsymbol{\omega}_o) d\boldsymbol{\omega}_i. \quad (4)$$

$\mathbf{L}_{gt}$ is a ground truth RGB pixel obtained from camera ray $(\mathbf{x}, \boldsymbol{\omega}_o)$. $\mathbf{L}_o$ denotes the synthesized rendering following the rendering equation [14], where $\mathbf{L}_e$ is the surface emitted radiance and $\mathbf{L}_r$ is the reflected radiance, given by integrating incident radiance $\mathbf{L}_i$ times the BRDF response (Eq. 4). Note, $\mathbf{L}_i$ is defined in a recursive manner with multi-bounce illumination naturally taken into account of.

4. Factorized Inverse Path Tracing

To optimize re-rendering error (Eq. 2), previous works [1, 29] apply differentiable path tracing to solve Eq. 3 and update BRDF and emission jointly with gradient descent. This approach can be unstable and inefficient: (1) gradient descent optimization is computationally intensive, which limits the number of path tracing samples and therefore increases the estimation variance; (2) fundamental ambiguities exist between BRDF and emission, making emission optimization difficult to regularize or converge. To reduce variance in optimization, we propose a factorized light transport representation (Sec. 4.1) which utilizes pre-baking of diffuse and specular shading maps (Eq. 6) to separate the BRDF coefficients out of the rendering integral.

Our full pipeline is demonstrated in Fig. 2, which optimizes dense BRDF and emission from posed images and scene geometry. The pipeline consists of 3 stages: (1) first,

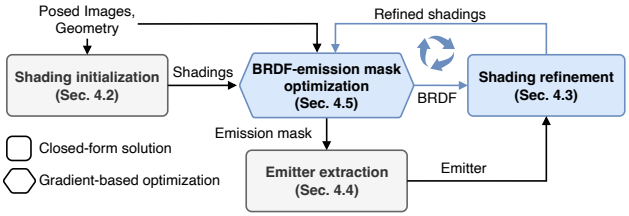


Figure 2: **Our inverse rendering pipeline** approximates diffuse and specular shadings from input images and geometry, which are used for efficient renderings during the BRDF-emitter optimization. The optimized BRDF and emitters are then passed into a path tracer to refine shadings. The BRDF and the shadings are then updated alternatively.

the factorized diffuse and specular shadings are initialized (baked) as described in Sec. 4.2; (2) given baked shadings, BRDF and emission mask are then optimized (Sec. 4.5), followed by emitter extraction (Sec. 4.4); (3) given current BRDF-emission estimation, the shadings are refined (Sec. 4.3), and the algorithm alternates between (2) and (3) until convergence.

4.1. Factorized light transport

A common way to speed up path tracing in the rendering literature [43, 18, 37] is to factor the BRDF from the rendering integral, and then to pre-bake and reuse the integral parts. Employing a similar idea, we rewrite the reflection equation (Eq. 4) as:

$$\mathbf{L}_r(\mathbf{x}, \boldsymbol{\omega}_o) = \mathbf{k}_d \mathbf{L}_d(\mathbf{x}) + \mathbf{k}_s \mathbf{L}_s^0(\mathbf{x}, \boldsymbol{\omega}_o, \sigma) + \mathbf{L}_s^1(\mathbf{x}, \boldsymbol{\omega}_o, \sigma) \quad (5)$$

$$\mathbf{L}_d(\mathbf{x}) = \int_{\Omega^+} \mathbf{L}_i(\mathbf{x}, \boldsymbol{\omega}_i) \frac{(\mathbf{n} \cdot \boldsymbol{\omega}_i)_+}{\pi} d\boldsymbol{\omega}_i$$

$$\mathbf{L}_s^0(\mathbf{x}, \boldsymbol{\omega}_o, \sigma) = \int_{\Omega^+} \mathbf{L}_i(\mathbf{x}, \boldsymbol{\omega}_i) \frac{F_0 DG}{4(\mathbf{n} \cdot \boldsymbol{\omega}_o)} d\boldsymbol{\omega}_i \quad (6)$$

$$\mathbf{L}_s^1(\mathbf{x}, \boldsymbol{\omega}_o, \sigma) = \int_{\Omega^+} \mathbf{L}_i(\mathbf{x}, \boldsymbol{\omega}_i) \frac{F_1 DG}{4(\mathbf{n} \cdot \boldsymbol{\omega}_o)} d\boldsymbol{\omega}_i,$$

where $\mathbf{L}_d$ is the diffuse shading; $\mathbf{L}_s^0, \mathbf{L}_s^1$ are the two specular shadings associated with two Fresnel components [34]:

$$F(\mathbf{h}, \boldsymbol{\omega}_i, \mathbf{k}_s(\mathbf{x})) = \mathbf{k}_s(\mathbf{x}) F_0 + F_1$$
$$F_0 = \left(1 - (1 - \mathbf{h} \cdot \boldsymbol{\omega}_i)^5\right), F_1 = (1 - \mathbf{h} \cdot \boldsymbol{\omega}_i)^5. \quad (7)$$

The specular shadings are further approximated by linear interpolation of 6 pre-defined roughness levels:

$$\mathbf{L}_s^*(\cdot, \sigma) \approx \text{lerp}(\{\mathbf{L}_s^*(\cdot, \sigma_k) | \sigma_k \in \text{linspace}(0,1,6)\}, \sigma), \quad (8)$$

such that $\mathbf{k}_d, \mathbf{k}_s, \sigma$ are all separated out of the integral.

With this factorization, we can bake the shadings $\mathbf{L}_d, \mathbf{L}_s^0, \mathbf{L}_s^1$ offline for each input view into image buffers, then query the shading pixels at training time to speed up rendering. Baking shadings offline allows us to use a large sampling rate for variance reduction. Owing to its linear

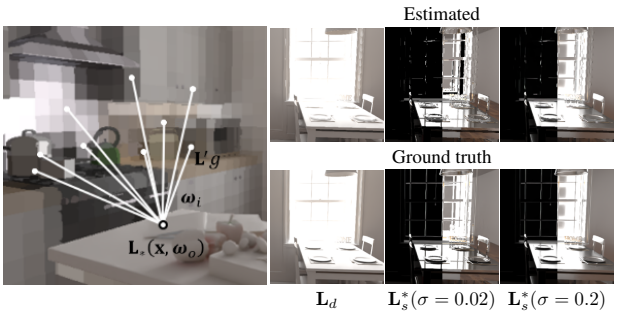


Figure 3: **Diffuse and specular shadings are initialized** by tracing a voxel representation of the surface light field $\mathbf{L}'$ (left), which gives approximations (top row on right) close to the ground truth (bottom row on the right; obtained by path tracing).

formulation, we also empirically found this factorized rendering handles mirror-like objects much better (Fig. 9), while standard Monte Carlo integration can easily obtain unstable gradients caused by the large BRDF value.

4.2. Image-based shading initialization

As we pre-bake the shadings $\mathbf{L}_d, \mathbf{L}_s^0, \mathbf{L}_s^1$, they are fixed during BRDF-lighting estimation. They need to be properly initialized; otherwise, the optimization will not converge. For simplicity, we abstract the integrands in Eq. 6 to the form: $\mathbf{L}_* = \mathbf{L}_i g$, where g denotes the factorized BRDF term. In path tracing notation, the shading integral can be initialized by querying a surface light field approximation:

$$\begin{aligned} \mathbf{L}_*(\mathbf{x}) &= \mathbf{L}(\mathbf{x}_1 \to \mathbf{x}) g(\mathbf{x}_1 \to \mathbf{x}) \\ &\approx \mathbf{L}'(\mathbf{x}_1) g(\mathbf{x}_1 \to \mathbf{x}), \end{aligned} \tag{9}$$

where $\mathbf{L}$ is the exact surface light field at sampled location $\mathbf{x}_1$ towards $\mathbf{x}$, and $\mathbf{L}'$ is its approximation obtained by average pooling all the input pixels onto a voxel grid spanned on the scene geometry (Fig. 3 left). Since objects in an indoor scene are often near-diffuse, and the renderings are essentially low pass filtering the incident light field [33] that blurs the detail, we find that using a 256^3 voxel grid with nearest neighbor radiance query gives good shading approximations (Fig. 3 right).

4.3. Path-traced shading refinement

Eq. 9 gives incorrect shading if the surface light field is sampled at locations that are mainly specular ($\mathbf{L}$ is view dependent), which subsequently leads to incorrect BRDF estimation. Given BRDF-emitter estimations optimized from Eq. 2 under current shading estimations, we re-estimate the light transport on specular surfaces by growing the path in Eq. 9 until the ray either hits an emitter or intersects with a

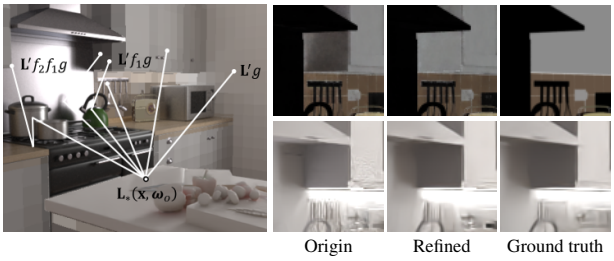


Figure 4: **Shading refinement:** The cabinet's diffuse reflectance estimation is initially darker than ground truth, owing to the excessive incident light received from the range hood that reflects non-diffuse light (2nd column). The artifacts are reduced by growing the path for the specular surface according to the optimized BRDF (1st column), which gives more accurate shadings that can be used to further refine the BRDF (3rd column).

Figure 5: **Diffuse radiance cache** from $\mathbf{L}'$ helps reduce variance and error for shading estimation (2nd image). Without it, sampling the tiny emitters below the cabinet will be difficult (1st image), which leads to incorrect shading and albedo (4th image).

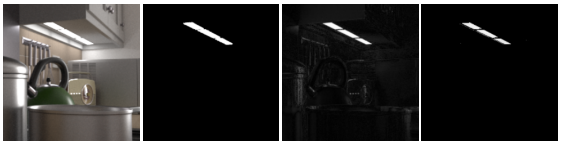
near diffuse surface (identified by $\sigma > 0.6$; Fig. 4):

$$\begin{aligned} \mathbf{L}_*(\mathbf{x}) = \mathbf{R}(\mathbf{x}_n) \prod_{i=1}^{n-1} f(\mathbf{x}_{i+1} \to \mathbf{x}_i) g(\mathbf{x}_1 \to \mathbf{x}), \\ \text{s.t. } \sigma(\mathbf{x}_i) \le 0.6, \forall i < n \end{aligned} \tag{10}$$

$$\mathbf{R}(\mathbf{x}_n) = \begin{cases} \mathbf{L}'(\mathbf{x}_n) & \sigma(\mathbf{x}_n) > 0.6 \\ \mathbf{L}_e(\mathbf{x}_n) & \mathbf{L}_e(\mathbf{x}_n) > 0 \end{cases}, \tag{11}$$

where n is the length of a certain path before it terminates. The above equation essentially estimates the shadings by multi-bounced path tracing with $\mathbf{L}'$ being a diffuse radiance cache, which helps speed up the evaluation and also reduce error: initial estimations of fs may retain large error but $\mathbf{L}'$ is very close to a diffuse surface light field (as it is also view-independent). Most of the path hits a diffuse surface within one to two bounces, such that the errors from the BRDF will not be magnified (Fig. 5).

Substituting shadings in factorized rendering by their refinements makes Eq. 5 more closely match the ground truth light transport, such that BRDF can be re-estimated with fewer artifacts (Fig. 4: 'Origin' VS 'Refined'). The re-estimated BRDF in turn is applied to further improve the shadings, and this BRDF and shading refinement is performed alternatively until convergence.



RGB GT emission Rendering error Emission mask

Figure 6: **Rendering images without emission terms** produces distinctive error near emissive surfaces (3rd image). By jointly optimizing an emission mask (4th image) to cancel this error, the emitter can be found by checking the mask's response, which is robust even for tiny emitters (2nd image for ground truth).

4.4. Error-driven emitter estimation

If we replace rendering equation Eq. 3 by Eq. 4 that excludes the emission, the objective Eq. 2 still converges for non-emissive surfaces (as their $\mathbf{L}_e = 0$), but regions with emission will present large errors, which is a good indicator of emitters (Fig. 6). With this intuition, we introduce an emission mask (encouraged to be small) $\alpha \in [0, 1]$ to the rendering loss:

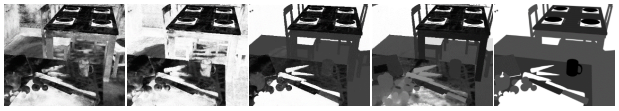
$$\min_{\mathbf{a},\sigma,m,\alpha} \sum_{\mathbf{x},\boldsymbol{\omega}_o} \|(1-\alpha)\mathbf{L}_r + \alpha\mathbf{L}_{gt} - \mathbf{L}_{gt}\|_2^2 \quad \text{s.t. } \alpha \to 0. \tag{12}$$

When a surface is non-emissive, α will stay small owing to the regularization and the loss is minimized by adjusting $\mathbf{L}_r$ towards $\mathbf{L}_{gt}$; but $\mathbf{L}_r$ cannot model the emission, so α for an emissive surface has to become large to accommodate the error. In practice we apply a L1 sparsity loss to α. By changing the optimization objective to Eq. 12, we first jointly estimate the BRDF-emission mask, and then threshold the mask to find the emitter ($\alpha > 0.01$). Afterward, each emitter's emission $\mathbf{L}_e$ is estimated independently from BRDF:

$$\mathbf{L}_e = \begin{cases} \arg\min_{\mathbf{L}_e} \sum_{\mathbf{x},\boldsymbol{\omega}_o} \|\mathbf{L}_e + \mathbf{L}_r - \mathbf{L}_{gt}\|_1 & \alpha > 0.01 \\ 0 & \text{otherwise} \end{cases}. \tag{13}$$

Our formulation is found to be more stable than joint optimization (demonstrated in ablations in Sec. 5.4), because the emission mask value is in the same range as BRDF coefficients, such that the gradient update is balanced between the BRDF and the emission mask. In contrast, surface emission can be much larger than BRDF coefficients, making it more difficult to directly fit or regularize.

Emitter extraction. We assume emission is constant for each mesh triangle. After α is optimized, we uniformly sample 100 locations on each triangle and find their corresponding α value. A triangle is then classified as an emitter if the mean of its αs is above 0.01. Eq. 13 in general is ill-posed (*e.g.* $\mathbf{k}_d$ can be increased by decreasing $\mathbf{L}_e$), so we make the assumption that an emitter reflects zero light ($f = 0$). In such a situation, $\mathbf{L}_e$ for a triangle has the closed-form solution as the median of RGBs from all input pixels it intersects, which does not require any gradient descent optimization, so it can be estimated efficiently and accurately.



w/o l_d, l_p w/o l_p w/ part w/ semantic GT

Figure 7: **Roughness optimization** can be ambiguous without any regularization (1st image). By encouraging a surface to be diffuse, specular surfaces still get an incorrect roughness value if no highlights are observed (2nd image). The roughness can be more reasonably estimated with part segmentation for guidance (3rd image). Semantic segmentation (4th image) shows similar results except the roughness for small objects get blurred.

4.5. Optimization

Given either initial or refined shadings, the BRDF and emission mask are optimized using the objective in Eq. 12. We encode BRDF and the emission mask with two MLPs:

$$\begin{aligned} (\mathbf{a}, m, \sigma) &= \text{Sigmoid}\left(\text{MLP}_{\text{brdf}}(\mathbf{x})\right) \\ \alpha &= 1 - \exp\left(-\text{ReLU}(\text{MLP}_{\text{emit}}(\mathbf{x}))\right), \end{aligned} \tag{14}$$

where MLP_{brdf} uses hash encoding [27] and MLP_{emit} is a positional encoded MLP [26]. The objective Eq. 12 is converted to a gradient descent loss function as a tone-mapped L2 loss l plus a L1 regularization term l_e:

$$l = \sum_{\mathbf{x},\mathbf{o}} \|\Gamma((1-\alpha)\mathbf{L}_r + \alpha\mathbf{L}_{gt}) - \Gamma(\mathbf{L}_{gt})\|_2^2 \tag{15}$$

$$l_e = \lambda_e \sum_{\mathbf{x}} \|\text{MLP}_{\text{emit}}(\mathbf{x})\|_1, \lambda_e = 1, \tag{16}$$

where Γ is the tone-mapping function by Munkberg *et al.* [28] to help suppress noise from high dynamic range values. We prefer neural networks rather than a textured mesh (as in [1, 29]) as scenes with complex geometries can create degenerate UVs, which reduces the BRDF quality.

Roughness-metallic regularization. Surface roughness and metallic can take arbitrary values if there are no highlights ($\mathbf{L}_s^0, \mathbf{L}_s^1 \approx 0$), which leads to ambiguity. We prevent this by encouraging surfaces to be diffuse:

$$l_d = \lambda_d \sum_{\mathbf{x}} \left(\|1 - \sigma(\mathbf{x})\|_1 + \|m(\mathbf{x})\|_1\right), \lambda_d = \text{5e-4}, \tag{17}$$

such that a diffuse surface will not be misinterpreted as a specular surface with weak reflection. To get valid roughness-metallic for input pixels that do not observe highlights, we further assume they stay constant inside each material part, and utilize image-level part segmentation to group input pixels. The roughness-metallic from pixels with highlights are propagated to their corresponding group by

another regularization loss:

$$l_p = \lambda_p \sum_{\mathbf{x}} \left\| \begin{bmatrix} \sigma(\mathbf{x}) \\ m(\mathbf{x}) \end{bmatrix} - \begin{bmatrix} \sigma'(\mathbf{x}) \\ m'(\mathbf{x}) \end{bmatrix} \right\|_1, \lambda_p = \text{5e-3} \quad (18)$$

$$\begin{bmatrix} \sigma'(\mathbf{x}) \\ m'(\mathbf{x}) \end{bmatrix} = \sum_{\text{Seg}(\mathbf{x}')=\text{Seg}(\mathbf{x})} \frac{w(\mathbf{x}')}{\sum_{\mathbf{x}'} w(\mathbf{x}')} \begin{bmatrix} \sigma(\mathbf{x}') \\ m(\mathbf{x}') \end{bmatrix} \quad (19)$$

$$w(\mathbf{x}') = \text{sg}\left(\|\mathbf{k_s}\mathbf{L}_s^0 + \mathbf{L}_s^1\|_1\right),$$

where $\text{Seg}(\mathbf{x})$ gives the segmentation ID for $\mathbf{x}$, $\text{sg}(\cdot)$ denotes stop the gradient, and $w(\mathbf{x})$ is a propagation kernel that weights the pixel by the amount of highlights. While part segmentation in practice can be hard to obtain, semantic segmentation is readily available from pre-trained model *e.g.* Mask2Former [10], where multiple material parts may stay inside the same semantic label. To account for such detail loss, we consider two pixels belong to the same material part only if: (1) they share the same semantic ID; (2) have similar albedo value; (3) and are close to each other, which suggests an alternative propagation kernel $w(\mathbf{x}, \mathbf{x}')$:

$$w(\mathbf{x}', \mathbf{x}) = \text{sg}\left(e^{-\frac{\|\mathbf{a}(\mathbf{x})-\mathbf{a}(\mathbf{x}')\|_2^2}{2\sigma_a^2}} e^{-\frac{\|\mathbf{x}-\mathbf{x}'\|_2^2}{2\sigma_x^2}} \right) \quad (20)$$

$$\sigma_a = \text{1.6e-2}, \sigma_x = \text{1e-2}.$$

By replacing $\mathbf{w}(\mathbf{x})$ with $\mathbf{w}(\mathbf{x}, \mathbf{x}')$ and changing the regularization weight to $\lambda_p = \text{1e-3}$, we can still have reasonable roughness-metallic estimation even with semantic segmentation (Fig. 7).

5. Experiments

We evaluate our method on 4 synthetic and 2 real indoor scenes, where the synthetic scenes are obtained from Bitterli's rendering resources [5] with large glass objects being removed (as we do not model transmission), and the real scenes are captured by us. Each synthetic scene contains around 200 posed HDR images generated by Mitsuba3 [13], per-camera view BRDF-emission maps generated by Blender [6], and ground truth geometry. For synthetic scenes, we show our method with both part segmentation (FIPT) and semantic segmentation mask (FIPT-sem).

The real scenes (Conference room and Classroom) are captured by a Sony A7M3 camera with around 200 HDR images reconstructed by 5-stop exposure bracketing. The camera poses are estimated from COLMAP [35] and the geometry is reconstructed using MonoSDF [47]. Please refer to the supplementary material for details on real scene capturing and additional results.

5.1. Synthetic: BRDF-emission estimation

While synthetic scenes allow us to directly compare with the ground truth BRDF-emission without noise from geometry or image captures, BRDF parameterizations can vary across different baselines. For fair comparison, we empirically found diffuse reflectance $\mathbf{k}_d$ for diffuse surfaces, roughness σ, and the material reflectance defined by $\mathbf{a}' = \int_{\Omega^+} f d\boldsymbol{\omega}_i$ are very close across different BRDF models [9, 15]. We therefore measure the PSNR for these metrics in image space for BRDF comparison. The $\mathbf{k}_d$ is compared only for diffuse surfaces, and $\mathbf{a}'$ is estimated using Monte Carlo integration of 128 samples per pixel. For emission, we estimate the IoU of emission mask and $\log$ L2 error of emission map.

	Method	$\mathbf{k}_d$	$\mathbf{a}'$	σ	$\mathbf{L}_e$	
		PSNR↑			IoU↑	logL2↓
Bathroom	Li22 [21]	19.92	15.78	13.77	0.45	1.35
	NeILF [44]	10.12	9.01	14.82	-	-
	IPT [1]	22.43	18.59	14.69	0.33	1.09e-1
	MILO [45]	11.83	9.80	5.56	0.05	5.60e-1
	FIPT	**30.13**	**25.28**	**28.79**	**0.63**	**3.18e-2**
	FIPT-sem	27.81	24.00	21.84	**0.63**	**3.18e-2**
Bedroom	Li22 [21]	21.87	17.18	12.12	0.34	2.78
	NeILF [44]	14.88	12.42	11.30	-	-
	IPT [1]	29.39	22.46	13.33	0.92	4.01e-3
	MILO [45]	23.65	15.16	15.42	0.08	1.59e-2
	FIPT	**31.10**	**29.41**	23.19	**0.96**	4.95e-4
	FIPT-sem	31.00	28.45	**25.23**	**0.96**	**4.93e-4**
Livingroom	Li22 [21]	17.25	15.32	12.72	0.17	3.61
	NeILF [44]	12.34	10.97	13.45	-	-
	IPT [1]	21.24	19.01	11.77	0.90	6.08e-3
	MILO [45]	22.88	18.39	13.98	0.06	1.39e-2
	FIPT	28.86	**28.70**	**32.48**	**0.95**	**8.06e-4**
	FIPT-sem	**29.09**	28.62	25.15	**0.95**	8.09e-4
Kitchen	Li22 [21]	18.14	14.54	10.82	0.43	1.41
	NeILF [44]	12.63	9.96	10.64	-	-
	IPT [1]	25.68	21.61	11.84	0.83	1.08e-2
	MILO [45]	18.25	13.86	12.56	0.10	8.28e-2
	FIPT	33.07	**27.53**	**29.24**	**0.91**	**1.54e-3**
	FIPT-sem	**33.25**	27.38	21.70	**0.91**	**1.54e-3**

Table 2: **BRDF-emission comparison on synthetic scenes** shows that our method gives the overall best reconstruction. The results are similar even if only semantic segmentation is provided (FIPT-sem). NeILF does not estimate emitters. The best method is marked in bold.

Baselines. We compare with the original inverse path tracing (IPT) [1] and its extension MILO [45] that also parameterizes spatially varying BRDF with neural networks. IPT assumes BRDF parameters to be constant inside each mesh triangle, and MILO takes manual input of number of emitters. Both IPT and MILO are evaluated by their original authors due to non-public code, and the MILO training is stopped after 10 hours. Meanwhile, we also compare NeILF [44] that models illumination in an unconstrained way and the learning based approach [21] (Li22) for single-view inverse rendering.

Results. As is shown in Tab. 2, our method gives the best BRDF and emission estimation with the fastest training speed (Tab. 3) even when only semantic level segmentation is provided (FIPT-sem). The learning-based approach (Li22) fails to generate reasonable reconstruction as it does

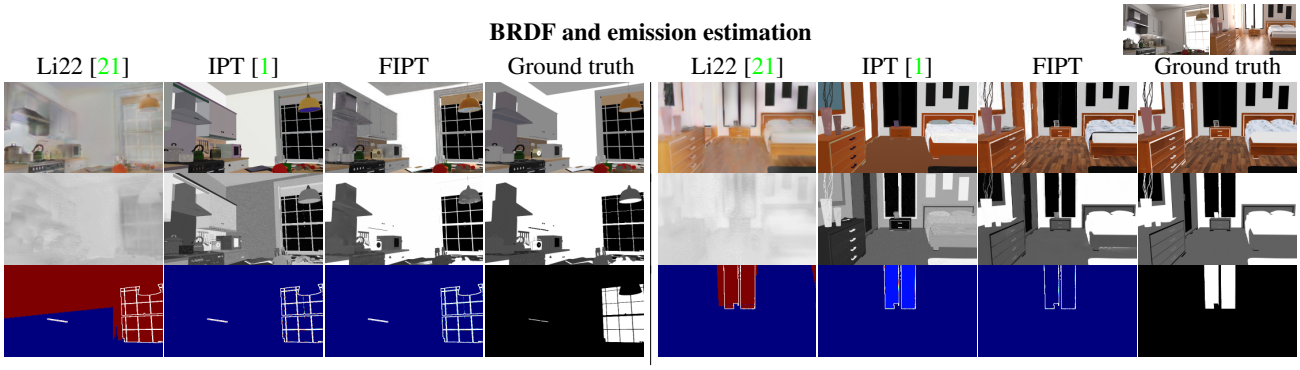


Figure 8: **BRDF and emission estimation results on 2 synthetic scenes** shows our method successfully reconstructs material reflectance (1st row), roughness (2nd row), and emission (3rd row) with high frequency details and less ambiguity. Emission estimation is shown as error heatmaps (warmer colors indicate higher emission error; GT emitter boundary is marked in white lines). Input views are shown in upper-right corner.

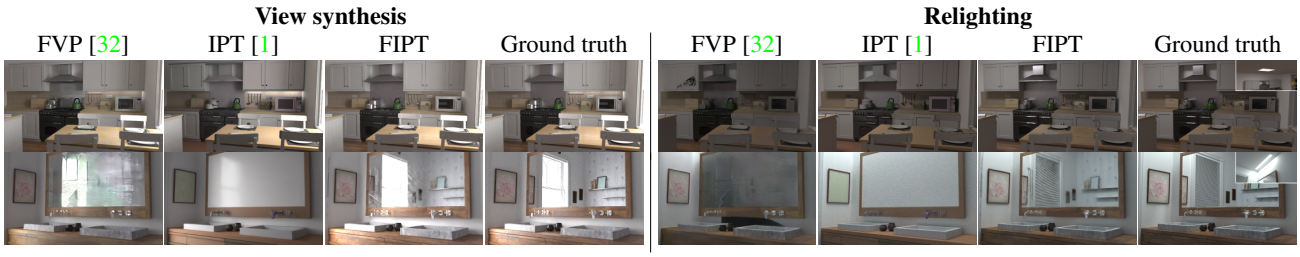


Figure 9: **Qualitative results of view synthesis (left) and relighting (right) on 2 synthetic scenes** demonstrate accurate light transport can be simulated with our estimated BRDF and emission even for very specular surfaces. The reflection of the chair from the microwave oven can be seen in kitchen scene on top, and mirrors are correctly rendered for bathroom (bottom).

Our per-stage profiling			
	Stage 1	Stage 2	Stage 3
Memory	3.2GB	2.8GB	3.4GB
Time	6min	2min	16min

Method	training time↓
NeILF [44]	1h38min
IPT [1]	≈3hr
MILO [45]	≈10hr
FIPT	**44min**

Table 3: **Averaged training speed comparison** suggests our method is very efficient (right table). The per-stage profiling is shown on the left with Stage 2 and 3 being repeated twice. The comparison is made on a 3090Ti GPU.

not utilize multi-view cues, while unconstrained optimization (NeILF) suffers from the ambiguity between material and lighting. While IPT converges, its accuracy is limited by a piece-wise constant constraint to reduce the variance. MILO also fails to reconstruct high frequency details because of the Monte Carlo noise from path tracing, and it requires manual specification of the number of emitters to constrain the emission optimization. In contrast, our method requires no human input during optimization, which allows more stable and faster convergence with results that match the ground truth well (Fig. 8).

5.2. Synthetic: view synthesis and relighting

To demonstrate the applications of inverse rendering outputs, we compare the rendered scenes under novel views and novel lighting using estimated BRDF and emission. For quantitative comparison, we tone-map the rendered images with $\gamma = 1/2.2$ then calculate their PSNR with respect to the ground truth.

Baselines. Besides IPT, MILO, and Li22, we also consider FVP [32] that performs view synthesis and relighting in a learning-based way. FVP assumes emissions come from saturated regions on the images, which may wrongly classify surfaces with strong reflection as emitters. So we offer ground truth emission to FVP instead as oracle. The renderings for IPT, MILO, and our method are obtained by path tracing with 1024 samples per pixel, which is further denoised by the Optix denoiser [31].

Results. As is shown in Tab. 4, our (FIPT and FIPT-sem) estimated BRDF-emission gives the most accurate view synthesis and relighting results. While results from FVP are seemingly visually appealing, the method is not guaranteed to be physically plausible and fails to match the ground truth. As shown in Fig. 9, our method handles specular reflections and even mirror reflection well, which is difficult to be modeled by standard inverse path tracing owing to its high variance.

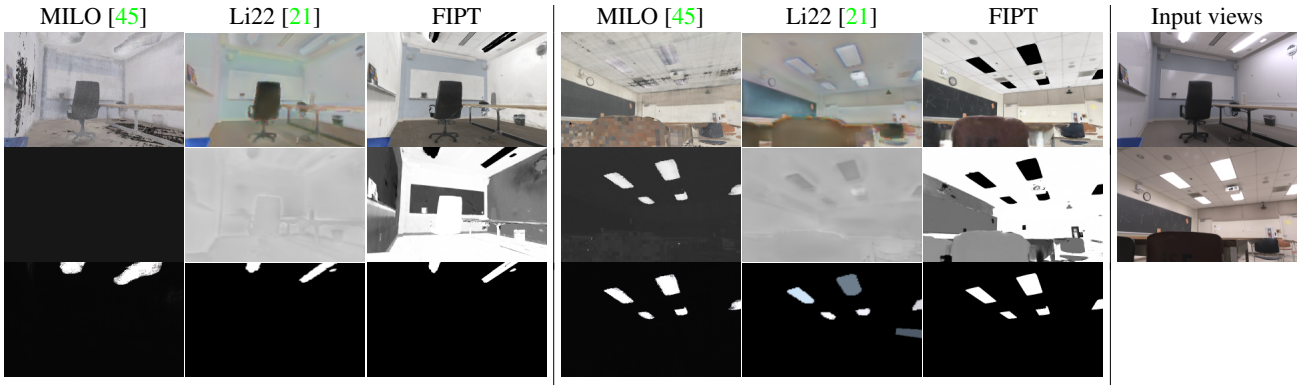


Figure 10: **BRDF and emission estimation results on 2 real world scenes** demonstrate our method gives reasonable estimation of BRDF and emission. The albedo and roughness preserve details without noticeable artifacts (row 1-2), and the emitters are correctly identified (3rd row).

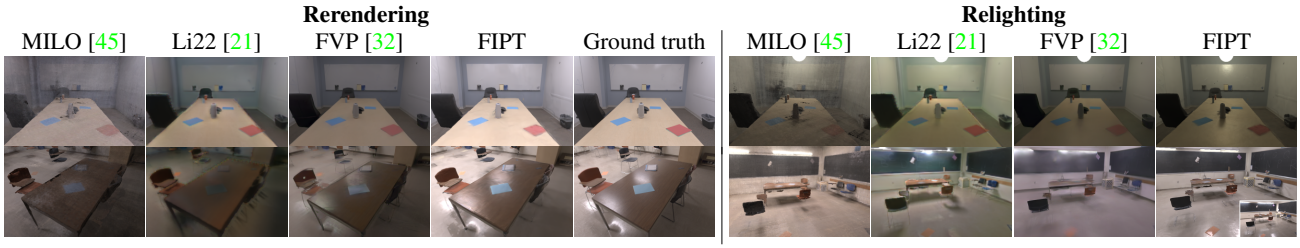


Figure 11: **Rerendering and relighting on 2 real world scenes** show our estimation fits the ground truth well (column 1-5) and gives good rendering under novel light (column 6-9). The inset in lower-right sub-figure shows the reference relighting for the Classroom scene.

	Method	Bathroom	Bedroom	Livingroom	Kitchen
View synthesis	FVP [32]	23.38	20.49	24.63	20.77
	IPT [1]	14.76	21.85	23.87	19.94
	MILO [45]	20.62	20.25	24.47	18.09
	FIPT	25.42	29.84	**30.86**	**25.38**
	FIPT-sem	**25.76**	**29.89**	30.84	25.27
Relight	Li22 [21]	22.86	23.20	19.83	21.76
	FVP [32]	23.72	24.11	19.51	23.31
	IPT [1]	20.61	28.16	27.26	27.28
	MILO [45]	14.97	23.39	22.10	19.62
	FIPT	**31.28**	36.64	**31.56**	**29.13**
	FIPT-sem	31.03	**36.69**	30.82	28.79

Table 4: **Quantitative results (PSNR) of view synthesis and relighting on synthetic scenes** show our estimation yields very consistent rendering under novel views and lighting. View synthesis is unavailable for Li22.

5.3. Real-world scenes

Considering that it is not possible to obtain ground truth BRDF and emission from just RGB captures, we only showcase qualitative comparison with MILO, FVP, and Li22 in terms of material-lighting estimation, rerendering, and relighting. Only semantic segmentation is used for the real scenes as ground truth part segmentation is unavailable.

Results. As is shown in Fig. 10, our method visually produces more reasonable material reflectance and roughness with emission masks closer to the actual emitters. The rerendering and relighting in Fig. 11 further serve as indirect measurements of the reconstruction quality, where our method is capable of reconstructing the reflection of the emitters on the Conference room wall and the Classroom table, and our relighting is visually closer to reference photos (obtained by switching lights; see supplementary) than the baselines with good reproduction of specular highlights and shadows.

5.4. Ablation study

Training strategy. Tab. 5 shows the effect of different training strategies on the kitchen scene. If we jointly optimize the emission and BRDF with the regularization term in IPT [1], the BRDF optimization can still converge, but the emission estimation does not converge given the same amount of time (2 epochs). Since the majority of the scene receives incident light from nearby diffuse surfaces, the reconstruction result is still reasonable without shading refinement (stage 3), but further refining the BRDF estimation helps to correct light transport for specular surfaces. If we simply path-trace the BRDF-emission without using the radiance cache from stage 1, the refined shadings will accumulate too much estimation error causing the subsequent BRDF estimation to deviate from ground truth.

Training strategy	$\mathbf{k}_d$	$\mathbf{a}'$	σ	$\mathbf{L}_e$	
	PSNR↑			IoU↑	logL2↓
Joint $\mathbf{L}_e$ opt.	32.31	25.50	22.77	0.10	6.62e-1
w/o stage 3	32.20	25.18	23.37	**0.91**	**1.54e-3**
w/o rad. cache	19.35	16.82	25.65	**0.91**	**1.54e-3**
Full model	**33.07**	**27.53**	**29.24**	**0.91**	**1.54e-3**

Table 5: **Ablation study on training strategy** shows joint BRDF-emission optimization can lead to slow convergence of the emission, and shading refinement helps further improve the reconstruction quality.

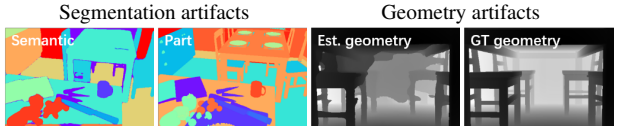


Figure 12: **Example of input noise** introduced by semantic segmentation and estimated geometry.

Training input	$\mathbf{k}_d$	$\mathbf{a}'$	σ	$\mathbf{L}_e$		Relight
	PSNR↑			IoU↑	logL2↓	PSNR↓
Fewer views	32.64	26.7	28.80	**0.91**	1.57e-3	28.90
Est. geometry	27.01	22.57	21.33	0.78	0.1142	27.90
Semantic seg.	**33.25**	27.38	21.70	**0.91**	**1.54e-3**	28.79
Part seg.	33.07	**27.53**	**29.24**	**0.91**	**1.54e-3**	**29.13**

Table 6: **Ablation study on training input** shows our method gives similar performance with fewer training images. If the input geometry or the segmentation is not perfect (semantic segmentation), the reconstruction quality downgrades but can still give reasonable relighting.

Sensitive analysis on training inputs. We run our algorithm on the kitchen scene with different setups: (1) using 60 training views instead of 200; (2) using geometry from MonoSDF [47] instead of the ground truth; (3) and replacing part segmentation by semantic segmentation. As is shown in Tab. 6, training with fewer views shows almost equal quality as long as they cover most of the scene. However, it still requires around 200 images in capturing to get good geometry reconstruction. Both estimated geometry and semantic segmentation introduce inaccuracy for detailed objects (Fig. 12), which disrupts roughness estimation for regions that see weak highlights. However, given the lighting for those regions is mostly ambiguous, it does not affect the overall reconstruction quality or relighting too much.

Complex emitters. Our method also works with lamp-like emitters with complex geometry (Fig. 13). For windows with hollow geometry, we can model the window light as a directional light source and let rays that fail to hit any indoor geometry to query an outdoor environment map. Each pixel of the environment map is estimated similarly as the emission of an emitter triangle. While environment lighting may not be fully observed and consequently causes artifacts near windows (see Sec. 6), a majority of the surfaces can still be reconstructed well owing to the diffuse radiance cache.

Figure 13: **Reconstruction with complex emitters.** Our method can also correctly identify complex lamps (first 2 columns) or environment lighting (last 2 columns) with reasonable BRDF reconstruction. Ground truth is shown in the insets.

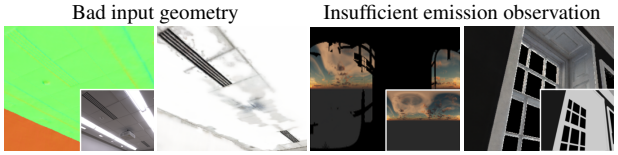


Figure 14: **Limitations.** Bad geometry estimation of an emitter (1st image) will lead to incorrect BRDF reconstruction of its nearby regions (2nd image). Incomplete observation of the environment light (3rd image) can cause artifacts in BRDF estimation (4th image).

6. Limitations and Future Work

Our method shares certain limitations with standard inverse path tracing. The framework does not optimize geometry, so that the BRDF and emission estimations can be inaccurate if the input geometry (especially for emitters) is extremely bad (Fig 14, left). Combining differentiable geometry optimization [2, 40] may help improve the robustness. Meanwhile, the BRDF estimation fails if the dominant light source (*e.g.* the sun) is not directly observed, which can happen very often with environment emitters whose observations are blocked by the windows (Fig. 14, right). Incorporating learning based methods may help. Lastly, the optimization relies on photometric observations, which means it cannot remove ambient occlusion effects out of the BRDF maps (as radiance there is near zero) and our model does not model transparent objects.

Acknowledgements. This work was supported in part by NSF grants 1751365, 2100237, 2105806, 2110409, 2120019, 2127544, ONR grant N000142012529, N000142312526, a Sony research Award, gifts from Qualcomm, Adobe and Google, the Ronald L. Graham Chair and the UC San Diego Center for Visual Computing.

Additionally, we would like to thank Bohan Yu, Dejan Azinovic, Julien Philip, and Zhengqin Li for generous assistance in evaluation of their methods, David Forsyth, Shenlong Wang, and Merlin Nimier-David for insightful discussions, as well as Jiaer Zhang for assistance in implementation.

References

[1] Dejan Azinovic, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. Inverse path tracing for joint material and lighting estimation. In *CVPR*, 2019. 1, 2, 3, 5, 6, 7, 8

[2] Sai Praveen Bangaru, Michaël Gharbi, Fujun Luan, Tzu-Mao Li, Kalyan Sunkavalli, Milos Hasan, Sai Bi, Zexiang Xu, Gilbert Louis Bernstein, and Frédo Durand. Differentiable rendering of neural sdfs through reparameterization. In *SIGGRAPH Asia*, 2022. 9

[3] Sai Praveen Bangaru, Tzu-Mao Li, and Frédo Durand. Unbiased warped-area sampling for differentiable rendering. In *ACM Transactions on Graphics (TOG)*, volume 39, pages 1–18, 2020. 2

[4] Jonathan T Barron and Jitendra Malik. Intrinsic scene properties from a single rgb-d image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 17–24, 2013. 2

[5] Benedikt Bitterli. Rendering resources, 2016. https://benedikt-bitterli.me/resources/. 6

[6] Blender Online Community. Blender - a 3d modelling and rendering package, 2022. 6

[7] Mark Boss, Raphael Braun, Varun Jampani, Jonathan T Barron, Ce Liu, and Hendrik Lensch. Nerd: Neural reflectance decomposition from image collections. In *ICCV*, pages 12684–12694, 2021. 2

[8] Mark Boss, Varun Jampani, Raphael Braun, Ce Liu, Jonathan Barron, and Hendrik Lensch. Neural-pil: Neural pre-integrated lighting for reflectance decomposition. In *NeurIPS*, volume 34, pages 10691–10704, 2021. 2

[9] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *SIGGRAPH*, volume 2012, pages 1–7. vol. 2012, 2012. 6

[10] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *CVPR*, 2022. 6

[11] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *SIGGRAPH*, page 189–198, 1998. 2

[12] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics (TOG)*, 36(6):1–14, 2017. 2

[13] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org. 6

[14] James T Kajiya. The rendering equation. In *SIGGRAPH*, pages 143–150, 1986. 3

[15] Brian Karis. Real shading in unreal engine 4. In *SIGGRAPH 2013 Course: Physically Based Shading in Theory and Practice*, 2013. 3, 6

[16] Kevin Karsch, Varsha Hedau, David Forsyth, and Derek Hoiem. Rendering synthetic objects into legacy photographs. *ACM Transactions on Graphics (TOG)*, 30(6):1–12, 2011. 2

[17] Kevin Karsch, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, Hailin Jin, Rafael Fonte, Michael Sittig, and David Forsyth. Automatic scene inference for 3d object compositing. *ACM Transactions on Graphics (TOG)*, 33(3):1–15, 2014. 2

[18] Jaroslav Křivánek, Pascal Gautron, Greg Ward, Okan Arikan, and Henrik Wann Jensen. Practical global illumination with irradiance caching. In *ACM SIGGRAPH 2007 courses*, 2007. 2, 3

[19] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. In *ACM Transactions on Graphics (TOG)*, volume 37, pages 1–11, 2018. 2

[20] Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In *CVPR*, 2020. 2

[21] Zhengqin Li, Jia Shi, Sai Bi, Rui Zhu, Kalyan Sunkavalli, Miloš Hašan, Zexiang Xu, Ravi Ramamoorthi, and Manmohan Chandraker. Physically-based editing of indoor scene lighting from a single image. In *ECCV*, 2022. 2, 6, 7, 8

[22] Zhen Li, Lingli Wang, Mofang Cheng, Cihui Pan, and Jiaqi Yang. Multi-view inverse rendering for large-scale real-world indoor scenes. *arXiv preprint arXiv:2211.10206*, 2022. 2

[23] Zhen Li, Lingli Wang, Mofang Cheng, Cihui Pan, and Jiaqi Yang. Multi-view inverse rendering for large-scale real-world indoor scenes. *arXiv preprint arXiv:2211.10206*, 2022. 2

[24] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single image. In *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, volume 37, 2018. 2

[25] Robert Maier, Kihwan Kim, Daniel Cremers, Jan Kautz, and Matthias Nießner. Intrinsic3d: High-quality 3d reconstruction by joint appearance and geometry optimization with spatially-varying lighting. In *ICCV*, pages 3114–3122, 2017. 2

[26] 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 *ECCV*, 2020. 1, 5

[27] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. In *ACM Transactions on Graphics (ToG)*, volume 41, pages 1–15, 2022. 5

[28] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *CVPR*, pages 8280–8290, 2022. 2, 5

[29] Merlin Nimier-David, Zhao Dong, Wenzel Jakob, and Anton Kaplanyan. Material and Lighting Reconstruction for Complex Indoor Scenes with Texture-space Differentiable Rendering. In Adrien Bousseau and Morgan McGuire, editors, *Eurographics Symposium on Rendering - DL-only Track*, 2021. 1, 2, 3, 5

[30] Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. Mitsuba 2: A retargetable forward and inverse renderer. In *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, volume 38, 2019. 2

[31] Steven G Parker, James Bigler, Andreas Dietrich, Heiko Friedrich, Jared Hoberock, David Luebke, David McAllister, Morgan McGuire, Keith Morley, Austin Robison, et al. Optix: a general purpose ray tracing engine. In *Acm transactions on graphics (tog)*, volume 29, pages 1–13, 2010. 7

[32] Julien Philip, Sébastien Morgenthaler, Michaël Gharbi, and George Drettakis. Free-viewpoint indoor neural relighting from multi-view stereo. In *ACM Transactions on Graphics (TOG)*, volume 40, pages 1–18, 2021. 2, 3, 7, 8

[33] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *SIGGRAPH*, pages 497–500, 2001. 4

[34] Christophe Schlick. An inexpensive brdf model for physically-based rendering. In *Computer graphics forum*, volume 13, pages 233–246. Wiley Online Library, 1994. 2, 3

[35] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 1, 6

[36] Soumyadip Sengupta, Jinwei Gu, Kihwan Kim, Guilin Liu, David W Jacobs, and Jan Kautz. Neural inverse rendering of an indoor scene from a single image. In *ICCV*, pages 8598–8607, 2019. 2

[37] Dario Seyb, Peter-Pike Sloan, Ari Silvennoinen, Michał Iwanicki, and Wojciech Jarosz. The design and evolution of the UberBake light baking system. In *ACM Trans. Graph. (Proc. SIGGRAPH)*, volume 39, 2020. 2, 3

[38] 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 *CVPR*, pages 7495–7504, 2021. 2

[39] Pratul P Srinivasan, Ben Mildenhall, Matthew Tancik, Jonathan T Barron, Richard Tucker, and Noah Snavely. Lighthouse: Predicting lighting volumes for spatially-coherent illumination. In *CVPR*, pages 8080–8089, 2020. 2

[40] Delio Vicini, Sébastien Speierer, and Wenzel Jakob. Differentiable signed distance function rendering. In *ACM Transactions on Graphics (TOG)*, volume 41, pages 1–18, 2022. 9

[41] Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. Microfacet models for refraction through rough surfaces. In *Eurographics conference on Rendering Techniques*, pages 195–206, 2007. 2

[42] Zian Wang, Jonah Philion, Sanja Fidler, and Jan Kautz. Learning indoor inverse rendering with 3d spatially-varying lighting. In *ICCV*, pages 12538–12547, 2021. 2

[43] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A ray tracing solution for diffuse interreflection. In *SIGGRAPH*, pages 85–92, 1988. 1, 2, 3

[44] 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 *ECCV*, pages 700–716, 2022. 2, 6, 7

[45] Bohan Yu, Siqi Yang, Xuanning Cui, Siyan Dong, Baoquan Chen, and Boxin Shi. Milo: Multi-bounce inverse rendering for indoor scene with light-emitting objects. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023. 1, 2, 3, 6, 7, 8

[46] Yizhou Yu, Paul Debevec, Jitendra Malik, and Tim Hawkins. Inverse global illumination: Recovering reflectance models of real scenes from photographs. In *SIGGRAPH*, pages 215–224, 1999. 2

[47] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. In *NeurIPS*, 2022. 1, 2, 6, 9

[48] Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. Monte Carlo estimators for differential light transport. In *ACM Trans. Graph. (Proc. SIGGRAPH)*, volume 40, 2021. 2

[49] 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 *CVPR*, pages 5453–5462, 2021. 2

[50] 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. In *ACM Transactions on Graphics (TOG)*, volume 40, pages 1–18, 2021. 2

[51] Rui Zhu, Zhengqin Li, Janarbek Matai, Fatih Porikli, and Manmohan Chandraker. Irisformer: Dense vision transformers for single-image inverse rendering in indoor scenes. In *CVPR*, pages 2822–2831, 2022. 2