Learning-based Inverse Rendering of Complex Indoor Scenes with Differentiable Monte Carlo Raytracing

Jingsen Zhu
zhujingsen@zju.edu.cn
State Key Lab of CAD&CG, Zhejiang University
China

Fujun Luan
fluan@adobe.com
Adobe Research
USA

Yuchi Huo*
huo.yuchi.sc@gmail.com
State Key Lab of CAD&CG, Zhejiang University
Zhejiang Lab
China

Zihao Lin
zihaolin@zju.edu.cn
State Key Lab of CAD&CG, Zhejiang University
China

Zhihua Zhong
zhongzhihua@zju.edu.cn
State Key Lab of CAD&CG, Zhejiang University
China

Dianbing Xi
db.xi@zju.edu.cn
State Key Lab of CAD&CG, Zhejiang University
China

Jiaxiang Zheng
xuanfeng@qunhemail.com
KooLab, Manycore
China

Rui Tang
ati@qunhemail.com
KooLab, Manycore
China

Hujun Bao
bao@cad.zju.edu.cn
State Key Lab of CAD&CG, Zhejiang University
China

Rui Wang*
rwang@cad.zju.edu.cn
State Key Lab of CAD&CG, Zhejiang University
China

Figure 1: We present a learning-based approach for inverse rendering of complex indoor scenes with differentiable Monte Carlo raytracing. Our method takes a single indoor scene RGB image as input and automatically infers its underlying surface reflectance (represented by microfacet GGX), geometry, and spatially-varying illumination (first column). Consequently, this enables us to perform photorealistic editing of the scene, such as inserting multiple complex virtual objects (second column, note that the inserted models are highly glossy) and editing surface materials faithfully with global illumination (last two columns, note that the wall is modified to a mirror that correctly presents specular reflections of the kitchen, and the glossy cooktop is modified to Lambertian appearance).

ABSTRACT

Indoor scenes typically exhibit complex, spatially-varying appearance from global illumination, making inverse rendering a challenging ill-posed problem. This work presents an end-to-end, learning-based inverse rendering framework incorporating differentiable Monte Carlo raytracing with importance sampling. The framework takes a single image as input to jointly recover the underlying geometry, spatially-varying lighting, and photorealistic materials. Specifically, we introduce a physically-based differentiable rendering layer with screen-space ray tracing, resulting in more realistic specular reflections that match the input photo. In addition, we create a large-scale, photorealistic indoor scene dataset with significantly richer details like complex furniture and dedicated decorations. Further, we design a novel out-of-view lighting network with uncertainty-aware refinement leveraging hypernetwork-based neural radiance fields to predict lighting outside the view of the input photo. Through extensive evaluations on common benchmark datasets, we demonstrate superior inverse rendering quality of our method compared to state-of-the-art baselines, enabling various applications such as complex object insertion and material editing with high fidelity. Code and data will be made available at https://jingsenzhu.github.io/invrend.

*Denotes corresponding author.

Conference'17, July 2017, Washington, DC, USA
2022. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

CCS CONCEPTS

• **Computing methodologies** → **Rendering**; **Ray tracing**.

KEYWORDS

ray tracing, lighting estimation, inverse rendering

ACM Reference Format:

Jingsen Zhu, Fujun Luan, Yuchi Huo, Zihao Lin, Zhihua Zhong, Dianbing Xi, Jiaxiang Zheng, Rui Tang, Hujun Bao, and Rui Wang. 2022. Learning-based Inverse Rendering of Complex Indoor Scenes with Differentiable Monte Carlo Raytracing. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Inverse rendering of complex indoor scenes has been a long-standing challenge in computer graphics and vision. Given a single real-world image, global illumination effects such as shadows, specular highlights, and glossy interreflections are baked into the observed pixel values, imposing a particularly difficult task of simultaneously recovering the underlying scene geometry, spatially-varying surface reflectance, and arbitrary unknown illumination. Traditional optimization-based approaches rely on dedicatedly designed regularization and hand-crafted priors to tackle this problem. Unfortunately, such methods often fail in real-world scenarios due to overly simplified assumptions, leading to noticeable artifacts in both the decomposition and re-rendered results.

On the other hand, recent advances in inverse rendering [Li et al. 2020; Srinivasan et al. 2020; Wang et al. 2021b] leveraging deep learning methods have demonstrated impressive results on such scene inference tasks, where the underlying physical priors are supposed to be learnt automatically through an offline, supervised training process, typically on a large-scale, synthetic or labeled real training dataset of complex indoor scenes. Note that, it is extremely difficult, if not impossible, to generate ground truth labels of spatially-varying illumination and materials of an arbitrary real-world scene, and hence one crucial keypoint to the success of such methods is the fidelity and photorealism of the synthetic training data. Another essential factor strongly influencing the inference accuracy is the network structure design. Intuitively speaking, since inverse rendering is inverting the physical light transport, a physically-based differentiable rendering layer regularizing the parameter space can act as a meaningful prior, improving the robustness and generalization capability of the neural network regarding material and lighting decomposition, and thus also the geometry estimation in return. Consequently, the performance of these learning-based inverse rendering methods heavily depends on: 1) the quality of the training datasets, and 2) the design of the neural network architecture.

To address the aforementioned challenges, we propose a novel Monte Carlo differentiable rendering layer with importance sampling to faithfully simulate the physical light transport of an indoor scene. Experiments show that this is especially helpful in restoring the specular reflections of a given scene, and our method produces much more realistic re-rendered results comparing previous baselines. Unlike previous work that directly uses the local feature at a ray-surface intersection point, our approach importance samples the local incident radiance field of it via screen space ray tracing (SSRT) and uncertainty-aware, hypernetwork-based out-of-view lighting estimation. To facilitate training, we introduce a large-scale (~4000) complex indoor scene dataset, InteriorVerse. As far as we know, our dataset contains the highest quality with rich details compared to existing indoor scene datasets (e.g., OpenRooms [Li et al. 2021] or SUNCG [Song et al. 2017]), including complex furniture and dedicated decorations procedurally designed by professional digital artists, rendered with physically-based GGX model [Walter et al. 2007] using a modern GPU-based path tracing engine.

Concretely, our contributions include:

- A learning-based monocular inverse rendering framework of complex indoor scenes that recovers albedo, surface normal, depth, metallic, and roughness from a single indoor scene image.
- A novel Monte Carlo differentiable rendering layer with importance sampling, which correctly estimates the local incident radiance field using screen space ray tracing.
- An uncertainty-aware out-of-view light network leveraging hypernetwork-based neural radiance fields for robust out-of-view lighting estimation.
- A high-quality, large-scale complex indoor scene dataset, InteriorVerse, that contains rich details with high fidelity.

2 RELATED WORK

Inverse Rendering of Indoor Scenes. Inverse rendering attempts to reconstruct geometry and spatially-varying material and lighting information from monocular (which is our case) or multiple RGB images. Most previous methods only recognize one or part of the above attributes. Geometry reconstructions, including depth estimation and surface normal reconstruction, has been widely studied [Eigen and Fergus 2015; Liu et al. 2019]. Most material reconstruction methods are only able to either estimate diffuse albedo [Barron and Malik 2013; Karsch et al. 2014; Li and Snavely 2018] or classify material categories [Bell et al. 2015]. For lighting estimation, recent deep learning methods have made progress in estimating global [Gardner et al. 2019, 2017] and even spatially-varying [Garon et al. 2019; Li et al. 2020; Song and Funkhouser 2019] lighting conditions. Recent works attempt to predict multiple intrinsics jointly by a holistic inverse rendering framework. Li et al. [2020] proposed a method to reconstruct disentangled geometry, spatially-varying reflectance and lighting from a single RGB indoor scene image.

Lighting Estimation and Relighting. Light estimation is one of the sub-tasks of inverse rendering. Most previous works ignore spatially-varying effects and predict a single environment map for the whole scene [Gardner et al. 2017; Munkberg et al. 2022; Sengupta et al. 2019]. Indoor scenes suffer from spatial variations, thus recent work explores spatially-varying lighting estimation for indoor scenes. The representation of spatially-varying illumination includes environment maps, per-pixel spherical lobes [Li et al. 2020] (spherical Harmonics/Gaussians), or 3D voxel grids [Wang et al. 2021b]. Relighting is also a widely-studied relevant task. Griffiths et al. [2022] leverages screen-space method to detect occlusion and cast shadows to relight an outdoor image. Li et al. [2022] proposed a novel pipeline to modify the light conditions within an indoor scene.

Neural Scene Representations. Neural representations are a rapidly growing area of research. Recent advances include voxels [Sun et al. 2021; Yu et al. 2021a], hashgrids [Müller et al. 2022], point

clouds [Aliev et al. 2020], and neural implicit functions [Mildenhall et al. 2020; Wang et al. 2021a; Yariv et al. 2021, 2020]. Neural radiance fields (NeRFs) [Mildenhall et al. 2020] represents scenes as neural implicit functions, encoding a scene as a continuous volumetric radiance field of color and density. With volume rendering, a NeRF can synthesize novel view images with promising results. Our proposed method uses a NeRF as the representation of the out-of-view area of the scene (Sec. 4.3).

Differentiable Rendering. A number of recent inverse rendering works utilize differentiable rendering to recover complex light transport effects. Some recent works have proposed general-purpose physically-based differentiable renderers [Li et al. 2018a; Nimier-David et al. 2019]. Zhang et al. [2020] and Zeltner et al. [2021] discussed a rigurous theory of differentiable light transport and Monte-Carlo combinations. These physically-based methods achieve high-quality global illumination effects at the cost of substantial performance overhead. Some differentiable rendering techniques are customized for specific purpose such as texture [Nimier-David et al. 2021], split-sum lighting and mesh extraction [Munkberg et al. 2022]. Our method designs a Monte-Carlo based in-network differentiable rendering layer to recover the appearance of indoor scenes (Sec. 4.4).

Indoor Scene Datasets. Supervised learning requires a large database of indoor scene images and their corresponding ground truth geometry, material, and lighting for network training. Datasets include 3D shape models [Chang et al. 2015], real-world scans [Chang et al. 2017; Dai et al. 2017], and scene datasets [Li et al. 2018b, 2021; Savva et al. 2017; Song et al. 2017], which can be classified as either real or synthetic data. Real datasets provide real-world images and geometry, while synthetic datasets provide arbitrary scene annotations for inverse rendering, some of which, such as materials and illumination, are difficult to acquire from real world. To the best of our knowledge, InteriorNet [Li et al. 2018b] and OpenRooms [Li et al. 2021] are so far the highest-quality public indoor datasets with spatially-varying photorealistic material and illumination annotations. Unfortunately, InteriorNet provides only LDR results, while OpenRooms provides only lighting information on the scene surface (instead of at any 3D location), and lacks the complexity of material and furniture variations. We present a new indoor scene HDR dataset to tackle their shortcomings.

3 INTERIORVERSE: A LARGE-SCALE, PHOTOREALISTIC INDOOR SCENE DATASET

A high-quality dataset is crucial for learning-based inverse rendering. It's extremely difficult to acquire spatially-varying material and lighting ground truth in real world complex indoor scenes. Therefore, we render a synthetic dataset to supervise training. The SUNCG dataset [Song et al. 2017] is a manually-created large-scale dataset for indoor scenes, but they use non-physical Phong BRDF and render with OpenGL, which severely limits its photorealism. PBRS [Zhang et al. 2017] and CG-PBR [Sengupta et al. 2019] datasets are rendered with physically-based renderers, but both still use Phong BRDF and do not provide spatially-varying lighting ground truth for an arbitrary 3D location. InteriorNet [Li et al. 2018b] is a large-scale photorealistic indoor scene dataset providing multiple camera views and panoramas, but the images they provide are LDR, limiting the dynamic range of illumination. OpenRooms [Li et al. 2021] is by far the only HDR dataset with spatially-varying lighting rendered using physically-based microfacet BRDF. However, as shown in Fig. 2, it presents overly simplified furniture models and layouts, insufficient material and lighting variations, leading to less faithful appearance comparing to real world data consequently.

In this work, we create a new high-quality indoor scene dataset called InteriorVerse, which has the following advantages in data quality over existing alternatives: (1) the scene layouts of our dataset have richer details, including complex furniture and decorations. (2) Our dataset is rendered with GGX BRDF model [Walter et al. 2007], which has stronger material modeling capability than any BRDF models that existing indoor scene datasets use. (3) Our dataset provides not only pixel-wise surface environment maps, but also contains environment maps located anywhere in the 3D scene space. Fig. 2 compares some example scenes in our dataset and OpenRooms, showing our dataset's higher scene quality.

4 NETWORK DESIGN

Our inverse rendering framework takes a single image of indoor scene as input and jointly predicts the spatially-varying material, geometry and lighting of the scene, and can further re-render the appearance of the input image. Fig. 3 overviews the architecture of our framework, which consists of three parts: material-geometry network (§4.1), lighting network (§4.2 and §4.3), and a differentiable Monte-Carlo rendering layer (§4.4). The material-geometry network is an end-to-end convolutional network which directly predict the reconstruction results. The lighting network **LightNet** is comprised of three sub-parts: A Resnet34 encoder to produce local feature map from the input image (like pixelNeRF [Yu et al. 2021b]), a screen-space ray tracer to trace the source of the light, and a final MLP decoder to predict the lighting radiance result. The rendering layer takes G-Buffers and lighting as input, and uses Monte Carlo raytracing to reproduce realistic rendering results.

Figure 2: Example dataset images from OpenRooms [Li et al. 2021] (left) and our InteriorVerse dataset (right). Note that our dataset contains more diversified geometry, material (especially glossy and specular BRDFs) and complex lighting conditions comparing OpenRooms. Zoom in for details.

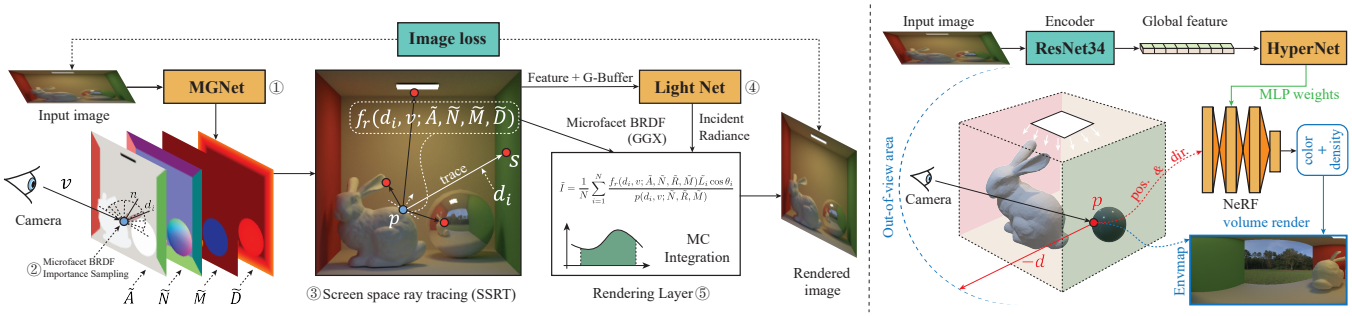

Figure 3: Overview of the pipeline. On the left, we show the workflow throughout our inverse rendering framework: (i) The spatially-varying material and geometry maps are predicted by MGNet (ii) According to the predicted material, geometry and view direction associated with each pixel point p, a BRDF importance sampling is performed to generate per-pixel incident directions $\mathbf{d}_i$ (iii) We use screen-space raytracing to trace the source point s of the query light. The corresponding local feature vector is extracted from feature map F via projection of point s. (iv) The feature is passed to LightNet along with light direction and auxiliary G-Buffer information to predict the incident radiance $\tilde{L}_i$ (v) Monte-Carlo integration (Eq. 7) is used to calculate the final re-rendered result. On the right, we show our out-of-view light estimation. We use a hypernetwork to predict the weights of the NeRF MLP and volume render the background lighting.

4.1 Material and Geometry Network

The input to our material and geometry prediction network **MGNet** is a single high dynamic range image, which can be directly obtained from our synthetic dataset. For real-world photos, we preprocess them with an inverse gamma correction. We use a single DenseNet121 [Huang et al. 2017] encoder to extract deep features of the material and shape parameters of the scene with different depth, as well as four separate decoders to obtain the final predicted albedo (A), material (M), normal (N), and depth (D), where M consists of two parts: roughness R and metallic M_t. While decoding neural features of different depths, upsampling and skip links are used to preserve multi-level details. Please refer to supplementary material for the detailed architecture of **MGNet**.

4.2 Lighting Network

We now describe our approach to predict any incident light intensity $L_i(\mathbf{p}, \mathbf{d})$ at point $\mathbf{p}$ with direction $\mathbf{d}$ from a single image. We fix our coordinate system as the *view space* of the input image and specify position $\mathbf{p}$ and light direction $\mathbf{d}$ in this coordinate system.

Given an input image $\mathbf{I}$ of a scene, we first extract a feature map $\mathbf{F} = E(\mathbf{I})$, where E is an encoder with ResNet34 [He et al. 2016] architecture. For any location $\mathbf{x}$ in the scene, we can retrieve the corresponding image feature by projecting $\mathbf{x}$ onto the image coordinates $\pi(\mathbf{x})$ using camera intrinsics and extract the local feature vector $\mathbf{F}[\pi(\mathbf{x})]$. Instead of directly using the local feature at incident point $\mathbf{p}$, we trace the ray from $\mathbf{p}$ with direction $-\mathbf{d}$ to point $\mathbf{s}$ in the scene, which can be treated as a virtual point light of $L_i(\mathbf{p}, \mathbf{d})$. We extract the local feature vector $\mathbf{F}[\pi(\mathbf{s})]$. The local feature is then passed into the final MLP decoder f, along with view direction $\mathbf{d}$ and some local G-Buffers (diffuse albedo K_d, specular albedo K_s, normal N and roughness R) at $\pi(\mathbf{s})$ $\mathbf{G}[\pi(\mathbf{s})]$, as

$$\mathbf{s} = \text{trace}(\mathbf{p}, -\mathbf{d}), \tag{1}$$

$$L_i(\mathbf{p}, \mathbf{d}) = f(\gamma(\mathbf{d}), \mathbf{F}[\pi(\mathbf{s})], \mathbf{G}[\pi(\mathbf{s})]), \tag{2}$$

where $\gamma(\cdot)$ is positional encoding function which is common used in NeRF [Mildenhall et al. 2020] to capture the high-frequency details within the data. The trace operation is implemented by **screen space ray tracing** (SSRT). We show our pipeline schematically in Fig. 3.

Our **screen space ray tracer** works on the depth map of the scene. It takes depth map $\mathbf{D}$, starting point $\mathbf{p}$, and the tracing direction $\mathbf{d}$ as inputs. The screen space ray tracer performs ray marching through pixels from the start point. At each step, the current depth of the ray is updated and compared with the surface depth of the pixel. If the ray depth is larger, it indicates that the ray has passed through the pixel surface, i.e. an intersection has occurred. Otherwise, it continues ray marching to an adjacent pixel until hitting the edge of the image.

4.3 Uncertainty-Aware Out-of-View Lighting Network

A limitation of screen space ray tracing is that the traced ray does not necessarily intersect within the field of view of the image. Therefore, an additional network (named "out-of-view lighting network") is designed to handle lights from the out-of-view area of the scene. The design of our out-of-view lighting network is inspired by Neural Radiance Fields (NeRF) [Mildenhall et al. 2020], which uses an MLP to represent the scene and uses volume rendering to predict the radiance of a ray. In the original version, the weights of the MLP are trained scene-specifically. Instead, we leverage hypernetwork [Ha et al. 2016] to reconstruct out-of-view lighting by predicting the scene-specific weights of the NeRF MLP, and then query the radiance by the same volume rendering and alpha compositing techniques.

The out-of-view lighting network architecture is shown in Fig. 3 (right-hand side). Given the input image $\mathbf{I}$, we first extract a global feature $\mathbf{F}_g = G(\mathbf{I})$, where G is an encoder with ResNet34 architecture (separate from the encoder in Section 4.2). Then, $\mathbf{F}_g$ is taken by hypernetwork H and the MLP's weights Φ are returned. To

query an incident light intensity $L_i(\mathbf{p}, \mathbf{d})$, we sample N 3D points $\{\mathbf{x}_i = \mathbf{p} - t_i\mathbf{d}\}$ on ray $(\mathbf{p}, -\mathbf{d})$. With positional encoding γ and NeRF MLP f, density σ and RGB color $\mathbf{c}$ are returned. The complete process can be formulated as:

$$\Phi = H(G(\mathbf{I})), \tag{3}$$

$$\{\sigma_i, \mathbf{c}_i = f(\gamma(\mathbf{x}_i); \Phi)\}_{i=1}^{N}. \tag{4}$$

Then light intensity L can be composited by

$$\hat{L} = \sum_{i=1}^{N} T_i(1 - \exp(-\sigma_i\delta_i))\mathbf{c}_i, \text{where } T_i = \exp\left(-\sum_{j=1}^{i-1}\sigma_j\delta_j\right), \tag{5}$$

where $\delta_i = t_{i+1} - t_i$ is the distance between adjacent samples. Due to performance consideration, our NeRF MLP is a small-scale network and does not take ray direction $\mathbf{d}$ as the MLP input like the original paper [Mildenhall et al. 2020] does.

The out-of-view lighting network is capable of predicting lighting anywhere in the scene. We now describe how we use it to refine the light predictions within field-of-view. Screen space ray tracing has a limitation that it may report some false positive of intersections. For real intersections, the difference between surface depth and ray depth is small, while the depth difference will increase when the intersection is a false positive. We model it as the "uncertainty" of SSRT, which is activated by hyperbolic tangent function: $u = \tanh(10\Delta d)$ where $\Delta d \in [0, \infty)$ is the depth difference. The refined light prediction is then formulated as

$$\hat{L}_{\text{refined}} = (1 - u) \times \hat{L}_{\text{SSRT}} + u \times \hat{L}_{\text{out-of-view}}, \tag{6}$$

where $\hat{L}_{\text{SSRT}}$ is the light prediction by our SSRT-based lighting network and $\hat{L}_{\text{out-of-view}}$ is the light prediction by our out-of-view lighting network. When uncertainty value u is large, screen space ray tracing becomes untrusted and the final prediction is dominated by out-of-view lighting prediction. We ablate between using only out-of-view network predictions and using full model predictions combined with Eq. 6 in our supplementary material.

4.4 Rendering Layer

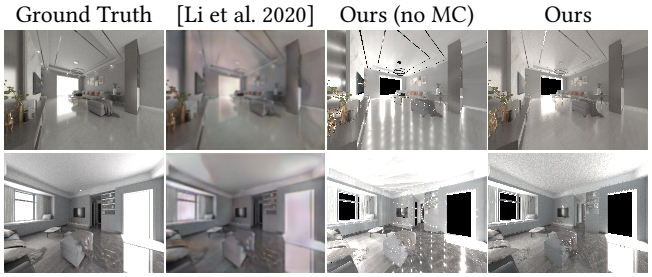

Figure 4: Qualitative comparison on re-rendered image. "Ours (no MC)" means that we re-render the image using our lighting prediction results but Li et al. [2020]'s rendering layer (instead of our MC rendering layer). Note that Li et al. [2020]'s render layer causes significant artifacts on glossy surfaces.

Unlike Li et al. [2020] which discretes the incident hemisphere to approximate the integration, we leverage differentiable Monte Carlo raytracing to produce photorealistic re-rendering results. Given sample count N, we use BRDF importance sampling to sample N ray directions $\{d_i\} = \{\phi_i, \theta_i\}$ according to view direction, surface normal and material parameters (roughness and metallic) at pixel point $\mathbf{p}$. We then perform screen-space raytracing according to d_i to trace the source point and predict the radiance of the corresponding direction $\{\tilde{L}_i\}$ from **LightNet**. The rendering layer computes the unbiased re-rendered image by

$$\tilde{I} = \frac{1}{N}\sum_{i=1}^{N}\frac{f_r(v, d_i; \tilde{A}, \tilde{N}, \tilde{R}, \tilde{M}_t)\tilde{L}_i \cos\theta_i}{p(v, d_i; \tilde{N}, \tilde{R}, \tilde{M}_t)}, \tag{7}$$

where $f_r(\omega_i, \omega_o)$ is the BRDF evaluation value and $p(\omega_i, \omega_o)$ is the probability distribution function (PDF) value of BRDF importance sampling, and v is the view direction. Our importance sampling rendering layer can produce much more realistic re-rendered images compared to [Li et al. 2020], especially in specular reflections and highlights. As shown in Fig. 4, our rendering layer is capable of recovering specular reflections on the glossy floor, while the rendering layer used by [Li et al. 2020] produces significant artifacts. The artifacts of [Li et al. 2020]'s discretization rendering algorithm are caused by the deterministic discrete direction sampling at each pixel, which is likely to miss important directions in the specular BRDF term. The missing of important reflection directions results in interleaved patterns in the re-rendered result. In contrast, our importance sampling strategy can faithfully recover high-frequency reflections on glossy surfaces.

5 TRAINING

We train our network models with the supervision of ground truth $\{I, A, N, D, R, M, L\}$ from our synthetic InteriorVerse dataset, where A, N, D, R, M denote albedo, normal, depth, roughness, and metallic, respectively, and L denotes spatially-varying lighting ground truth.

For geometry and material reconstruction, we use direct supervision to calculate the error between ground truth and network prediction. For lighting estimation, inspired by prior work [Li et al. 2020; Wang et al. 2021b], to encourage photorealistic scene appearance reconstruction, we additionally use a differentiable in-network rendering layer to re-render the image according to the predicted material, geometry, and lighting, and try to recover the original input image through an image loss. Note that, unlike prior work, our render layer incorporates physically-based Monte Carlo sampling via screen space ray tracing, which explicitly regularizes the physical parameter space with GGX importance sampling. As we will demonstrate later, this makes our method significantly more robust to handle specular reflections in the interior scene.

5.1 Material-Geometry Network

We train **MGNet** with the weighted combination of material losses (albedo loss $\mathcal{L}_{\text{albedo}}$, roughness-metallic loss $\mathcal{L}_{\text{material}}$) and geometry losses (normal loss $\mathcal{L}_{\text{normal}}$ and depth loss $\mathcal{L}_{\text{depth}}$):

$$\mathcal{L}_{\textbf{MGNet}} = \lambda_a\mathcal{L}_{\text{albedo}} + \lambda_n\mathcal{L}_{\text{normal}} + \lambda_m\mathcal{L}_{\text{material}} + \lambda_d\mathcal{L}_{\text{depth}}. \tag{8}$$

We add perceptual loss [Johnson et al. 2016] in the albedo, normal and material term, which helps to recognize the semantic boundaries in the image. The detailed definitions of separate losses and weights are presented in the supplemental material.

5.2 Lighting Network

We train **LightNet** with the weighted combination of direct light supervision loss $\mathcal{L}_{\text{light}}$ and re-rendering loss $\mathcal{L}_{\text{re-render}}$:

$$\mathcal{L}_{\textbf{LightNet}} = \mathcal{L}_{\text{light}} + \lambda_r \mathcal{L}_{\text{re-render}} \tag{9}$$

where $\mathcal{L}_{\text{light}}$ is the HDR supervision loss function proposed by [Mildenhall et al. 2021], while $\mathcal{L}_{\text{re-render}}$ is an L_2 loss between the re-rendered image and the original image. Please refer to supplementary material for the detailed definition of $\mathcal{L}_{\text{light}}$ and $\mathcal{L}_{\text{re-render}}$.

We find that re-rendering loss can significantly improve the lighting prediction, especially on specular surfaces. This benefit comes from enforcing the network to learn correct pixel brightness in $\hat{I}$, thus producing accurate lighting supervision in the scene and preventing blurry or spot artifacts in the re-rendered image. Ablation studies on the usage of re-rendering loss are presented in the supplementary material.

5.3 Training Scheme

We use a progressive training scheme to train our model in the order of data dependencies between different components of our framework. We first train material-geometry module to ensure correct predictions of albedo, normal, roughness, metallic and depth. This is because our lighting network depends on these properties (e.g., SSRT depends on depth, and MLP decoder depends on G-Buffers). Then we train lighting module with re-rendering loss.

6 EXPERIMENTS

6.1 Experiment Settings

Training data. We train our network on our new photorealistic indoor scene dataset, introduced in Sec. 3. When evaluating on real world data, we also fine-tune our model on IIW dataset [Bell et al. 2014] for albedo and NYUv2 [Silberman et al. 2012] for depth and normal. Please refer to our supplementary material for more details on training and evaluation data.

Baselines. We compare our method with Li et al. [2020], which is the state-of-the-art holistic inverse rendering frameworks for indoor scenes. To ensure a fair comparison, *we fine-tune [Li et al. 2020] on our new dataset*, which significantly improves its performance (Fig. 5). For lighting prediction, we compare with [Li et al. 2020] as well as another state-of-the-art lighting estimation method Lighthouse [Srinivasan et al. 2020], which requires a stereo image pair as input instead of a single image.

6.2 Evaluation of Material and Geometry

We evaluate material (albedo, roughness, and metallic) and geometry (normal and depth) prediction on InteriorVerse synthetic indoor dataset, as well as real-world dataset (NYUv2 dataset [Silberman et al. 2012] for geometry and IIW dataset [Bell et al. 2014] for albedo).

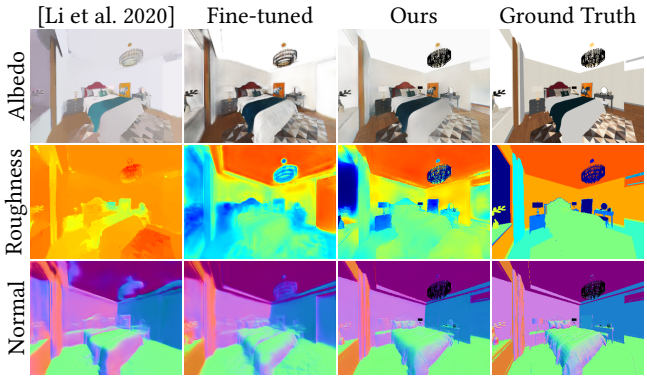

Figure 5: Qualitative results of geometry and BRDF estimation on synthetic dataset between Li et al. [2020] and our method. In second row, we show the improved prediction of Li et al. [2020] by fine-tuning it on our InteriorVerse dataset. We omit metallic comparison since Li et al. [2020] does not support it. See supplementary for more results.

Evaluation on synthetic dataset. We compare our method with the baseline methods on our InteriorVerse dataset. As shown in Fig. 5, our method outperforms [Li et al. 2020], For albedo prediction, while [Li et al. 2020] tends to over-smooth the result, our method faithfully preserves the texture details (e.g., the wooden textures of the floor). For normal prediction, our method is capable of preserving sharp edges between walls and floors. This attributes to the usage of perceptual loss, which helps the model recognize semantic borders in the image. Please refer to the supplementary material for an ablation study on the usage of perceptual loss.

Evaluation on real-world datasets. We evaluate albedo prediction on IIW dataset [Bell et al. 2014] with sparse pairwise human albedo annotations. We use the official metric suggested by [Bell et al. 2014], Weighted Human Disagreement Rate (WHDR), which measures the error when albedo predictions disagree with human annotations. We also evaluate geometry prediction on NYUv2 dataset [Silberman et al. 2012]. As shown in Table 1, we observe a lower error compared to prior works [Li et al. 2020; Wang et al. 2021b], indicating the advantage of our photo-realistic training datasets and our network design. Qualitative results of geometric and material predictions on real-world data are presented in the supplementary material.

Table 1: Evaluation of normals and depth on NYUv2 dataset (2nd and 3rd columns), and albedo on IIW dataset (last column).

Method	Normal Angular Error	Depth si-MSE	WHDR
[Li et al. 2020]	24.12°	0.160	15.9
[Wang et al. 2021b]	22.95°	0.181	18.2
Ours	**21.86°**	**0.155**	**15.5**

6.3 Evaluation of Lighting

Evaluation on virtual object insertion. We evaluate our lighting estimation method on a crucial augmented reality application: virtual object insertion. With the help of screen space ray tracing and

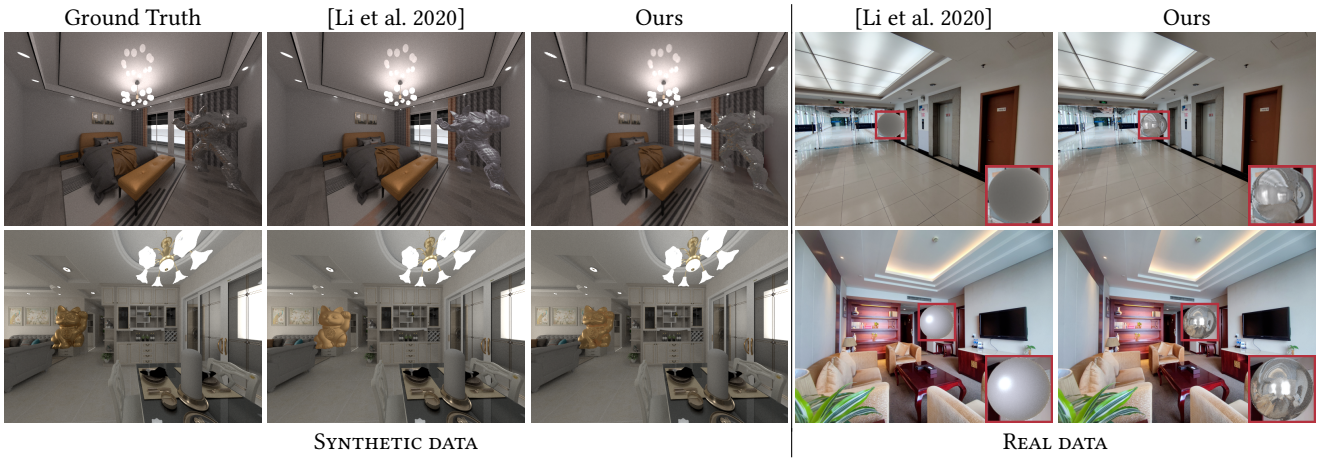

Figure 6: Qualitative comparison of object insertion results on synthetic dataset and real-world images. The ground truth object insertion results of synthetic scenes are provided. Li et al.'s results use the same highly-specular GGX BRDF as our results. However, because of their low-frequency lighting prediction, the inserted objects contain no sharp reflections and therefore resemble Lambertian appearance.

the Monte Carlo rendering layer, we can achieve promising results in specular reflection effects. Fig. 6 shows results of our method compared to baselines, consisting of both synthetic data and real world images. In order to emphasize the ability to recover high-frequency lighting details, the materials of the inserted objects are *highly specular*. For synthetic data, we insert complex objects and ground truths are provided. Li et al. [2020]'s lighting estimation is 2D spatially-varying, which cannot handle 3D points far from 2D surfaces. Moreover, their Spherical Gaussian lighting representation is incapable of capturing high-frequency angular details. Therefore, the appearance of inserted highly specular objects does not contain sharp reflections. In contrast, our method produces photorealistic shading and specular highlights on the inserted object. For real world data, we choose to insert highly specular spheres. The reflection on the sphere is supposed to be consistent with the surrounding environments. Li et al. [2020] also fails in this task, due to its low-frequency lighting predictions, while our method manages to faithfully recover angular details of the surrounding environment on the inserted sphere.

Figure 7: Qualitative comparison of object insertion results between Lighthouse [2020] and our method on Lighthouse's test set.

We also compare our method with another state-of-the-art lighting estimation method Lighthouse [Srinivasan et al. 2020], which requires a stereo pair of images as input. To show our method's cross-domain ability, we evaluate on *Lighthouse's test set* from InteriorNet [Li et al. 2018b] *without fine-tuning our network*. As shown in Figure 7, our method outperforms Lighthouse, even with a lower number of input images and a potential domain gap. We can observe that Lighthouse's lighting prediction has significantly less variation in lighting intensity. This may be because Lighthouse is trained from LDR panoramas, and cannot handle HDR lighting.

We also explore more applications of our lighting estimation method, including re-rendering and scene material edit. Please refer to our supplementary material for these additional results.

7 CONCLUSION AND LIMITATIONS

We present a learning-based method for inverse rendering of complex indoor scenes. Our approach handles spatially-varying illumination and faithfully recovers specular reflections thanks to the differentiable Monte Carlo rendering layer, enabling photorealistic editing such as complex object insertion and material change. Lastly, we introduce a large-scale indoor dataset, InteriorVerse, which contains much richer details than existing alternatives.

There are some limitations of our method. Our out-of-view lighting network is not capable of predicting high-frequency details due to its limited network capacity. Monte Carlo sampling would also lead to noisy re-render results, and raising the required sample budget can be computationally expensive. Further, emission of light sources is not supported currently, which we leave as future work.

REFERENCES

Kara-Ali Aliev, Artem Sevastopolsky, Maria Kolos, Dmitry Ulyanov, and Victor Lempitsky. 2020. Neural point-based graphics. In *European Conference on Computer Vision*. Springer, 696–712.

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

Sean Bell, Kavita Bala, and Noah Snavely. 2014. Intrinsic images in the wild. *ACM Transactions on Graphics (TOG)* 33, 4 (2014), 1–12.

Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. 2015. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3479–3487.

Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niebner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. 2017. Matterport3D: Learning from RGB-D Data in Indoor Environments. In *2017 International Conference on 3D Vision (3DV)*. IEEE Computer Society, 667–676.

Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. 2015. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012* (2015).

Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5828–5839.

David Eigen and Rob Fergus. 2015. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*. 2650–2658.

Marc-André Gardner, Yannick Hold-Geoffroy, Kalyan Sunkavalli, Christian Gagné, and Jean-François Lalonde. 2019. Deep parametric indoor lighting estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7175–7183.

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

Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. 2019. Fast spatially-varying indoor lighting estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6908–6917.

David Griffiths, Tobias Ritschel, and Julien Philip. 2022. OutCast: Outdoor Single-image Relighting with Cast Shadows. In *Computer Graphics Forum*, Vol. 41. Wiley Online Library, 179–193.

David Ha, Andrew Dai, and Quoc V Le. 2016. Hypernetworks. *arXiv preprint arXiv:1609.09106* (2016).

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4700–4708.

Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*. Springer, 694–711.

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

Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. 2018a. Differentiable Monte Carlo Ray Tracing through Edge Sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 37, 6 (2018), 222:1–222:11.

Wenbin Li, Sajad Saeedi, John McCormac, Ronald Clark, Dimos Tzoumanikas, Qing Ye, Yuzhong Huang, Rui Tang, and Stefan Leutenegger. 2018b. Interiornet: Mega-scale multi-sensor photo-realistic indoor scenes dataset. *arXiv preprint arXiv:1809.00716* (2018).

Zhengqin Li, Mohammad Shafiei, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. 2020. Inverse rendering for complex indoor scenes: Shape, spatially-varying lighting and svbrdf from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2475–2484.

Zhengqin Li, Jia Shi, Sai Bi, Rui Zhu, Kalyan Sunkavalli, Miloš Hašan, Zexiang Xu, Ravi Ramamoorthi, and Manmohan Chandraker. 2022. Physically-Based Editing of Indoor Scene Lighting from a Single Image. *arXiv preprint arXiv:2205.09343* (2022).

Zhengqi Li and Noah Snavely. 2018. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 371–387.

Zhengqin Li, Ting-Wei Yu, Shen Sang, Sarah Wang, Meng Song, Yuhan Liu, Yu-Ying Yeh, Rui Zhu, Nitesh Gundavarapu, Jia Shi, et al. 2021. OpenRooms: An Open Framework for Photorealistic Indoor Scene Datasets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7190–7199.

Chen Liu, Kihwan Kim, Jinwei Gu, Yasutaka Furukawa, and Jan Kautz. 2019. Planercnn: 3d plane detection and reconstruction from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4450–4459.

Ben Mildenhall, Peter Hedman, Ricardo Martin-Brualla, Pratul Srinivasan, and Jonathan T Barron. 2021. NeRF in the Dark: High Dynamic Range View Synthesis from Noisy Raw Images. *arXiv preprint arXiv:2111.13679* (2021).

Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2020. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*. Springer, 405–421.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. 2022. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *arXiv preprint arXiv:2201.05989* (2022).

Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. 2022. Extracting Triangular 3D Models, Materials, and Lighting From Images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8280–8290.

Merlin Nimier-David, Zhao Dong, Wenzel Jakob, and Anton Kaplanyan. 2021. Material and Lighting Reconstruction for Complex Indoor Scenes with Texture-space Differentiable Rendering. In *Eurographics Symposium on Rendering - DL-only Track*, Adrien Bousseau and Morgan McGuire (Eds.). The Eurographics Association. https://doi.org/10.2312/sr.20211292

Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. 2019. Mitsuba 2: A Retargetable Forward and Inverse Renderer. *Transactions on Graphics (Proceedings of SIGGRAPH Asia)* 38, 6 (Dec. 2019). https://doi.org/10.1145/3355089.3356498

Manolis Savva, Angel X Chang, Alexey Dosovitskiy, Thomas Funkhouser, and Vladlen Koltun. 2017. MINOS: Multimodal indoor simulator for navigation in complex environments. *arXiv preprint arXiv:1712.03931* (2017).

Soumyadip Sengupta, Jinwei Gu, Kihwan Kim, Guilin Liu, David W Jacobs, and Jan Kautz. 2019. Neural inverse rendering of an indoor scene from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 8598–8607.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. 2012. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*. Springer, 746–760.

Shuran Song and Thomas Funkhouser. 2019. Neural illumination: Lighting prediction for indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6918–6926.

Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. 2017. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1746–1754.

Pratul P Srinivasan, Ben Mildenhall, Matthew Tancik, Jonathan T Barron, Richard Tucker, and Noah Snavely. 2020. Lighthouse: Predicting lighting volumes for spatially-coherent illumination. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8080–8089.

Cheng Sun, Min Sun, and Hwann-Tzong Chen. 2021. Direct Voxel Grid Optimization: Super-fast Convergence for Radiance Fields Reconstruction. *arXiv preprint arXiv:2111.11215* (2021).

Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. 2007. Microfacet Models for Refraction through Rough Surfaces. *Rendering techniques* 2007 (2007), 18th.

Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021a. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689* (2021).

Zian Wang, Jonah Philion, Sanja Fidler, and Jan Kautz. 2021b. Learning indoor inverse rendering with 3d spatially-varying lighting. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12538–12547.

Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. 2021. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems* 34 (2021).

Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. 2020. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems* 33 (2020), 2492–2502.

Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. 2021a. Plenoxels: Radiance Fields without Neural Networks. *arXiv preprint arXiv:2112.05131* (2021).

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. 2021b. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4578–4587.

Tizian Zeltner, Sébastien Speierer, Iliyan Georgiev, and Wenzel Jakob. 2021. Monte Carlo Estimators for Differential Light Transport. *Transactions on Graphics (Proceedings of SIGGRAPH)* 40, 4 (Aug. 2021). https://doi.org/10.1145/3450626.3459807

Cheng Zhang, Bailey Miller, Kai Yan, Ioannis Gkioulekas, and Shuang Zhao. 2020. Path-Space Differentiable Rendering. *ACM Trans. Graph.* 39, 4 (2020), 143:1–143:19.

Yinda Zhang, Shuran Song, Ersin Yumer, Manolis Savva, Joon-Young Lee, Hailin Jin, and Thomas Funkhouser. 2017. Physically-based rendering for indoor scene understanding using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5287–5295.