Inverse Rendering for Complex Indoor Scenes: Shape, Spatially-Varying Lighting and SVBRDF from a Single Image

Zhengqin Li* zhl378@eng.ucsd.edu
Mohammad Shafiei* moshafie@eng.ucsd.edu
Ravi Ramamoorthi* ravir@cs.ucsd.edu
Kalyan Sunkavalli† sunkaval@adobe.com
Manmohan Chandraker* mkchandraker@eng.ucsd.edu
*University of California, San Diego †Adobe Research, San Jose

Abstract

We propose a deep inverse rendering framework for indoor scenes. From a single RGB image of an arbitrary indoor scene, we obtain a complete scene reconstruction, estimating shape, spatially-varying lighting, and spatially-varying, non-Lambertian surface reflectance. Our novel inverse rendering network incorporates physical insights – including a spatially-varying spherical Gaussian lighting representation, a differentiable rendering layer to model scene appearance, a cascade structure to iteratively refine the predictions and a bilateral solver for refinement – allowing us to jointly reason about shape, lighting, and reflectance. Since no existing dataset provides ground truth high quality spatially-varying material and spatially-varying lighting, we propose novel methods to map complex materials to existing indoor scene datasets and a new physically-based GPU renderer to create a large-scale, photorealistic indoor dataset. Experiments show that our framework outperforms previous methods and enables various novel applications like photorealistic object insertion and material editing.

1. Introduction

We address a long-standing challenge in inverse rendering to *reconstruct geometry, spatially-varying complex reflectance and spatially-varying lighting* from a single RGB image of an arbitrary indoor scene captured under uncontrolled conditions. This is a challenging setting – indoor scenes display the entire range of real-world appearance, including arbitrary geometry and layouts, localized light sources that lead to complex spatially-varying lighting effects and complex, non-Lambertian surface reflectance. In this work we take a step towards an automatic, robust and holistic solution to this problem, thereby enabling a range of scene understanding and editing tasks. For example, in Figure 1(h), we use our reconstruction to enable photorealistic virtual object insertion in a real image. Note how the inserted glossy spheres have realistic shading, shadowing due to scene occlusions and even reflections from the scene.

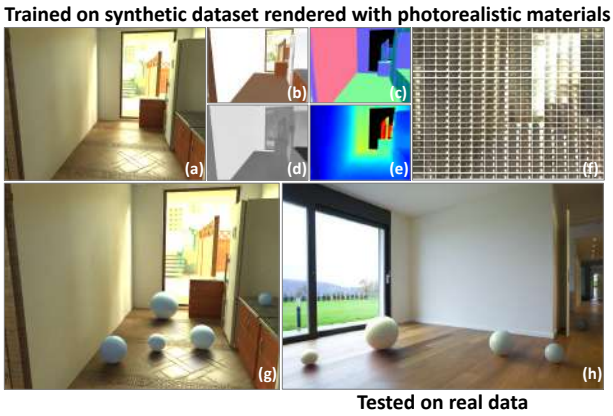

Figure 1. Given a single image of an indoor scene (a), we recover its diffuse albedo (b), normals (c), specular roughness (d), depth (e) and spatially-varying lighting (f). We build a large-scale high-quality synthetic training dataset rendered with photorealistic SVBRDF. By incorporating physical insights into our network, our high-quality predictions support applications like object insertion, even for specular objects (g) and in *real images (h)*. Note the completely shadowed sphere on the extreme right in (h).

Driven by the success of deep learning methods on similar scene inference tasks (geometric reconstruction [16], lighting estimation [17], material recognition [9]), we propose training a deep convolutional neural network to regress these scene parameters from an input image. Ideally, the trained network should learn meaningful priors on these scene factors, and jointly model the interactions between them. In this work, we present two major contributions to address this.

Training deep neural networks requires large-scale, labeled training data. While datasets of real-world geometry exist [14, 10], capturing real-world lighting and reflectance at scale is non-trivial. Thus, we use synthetic indoor datasets like [49] that contain scenes with complex geometry. However, their materials are not realistic [55], so we replace them with photorealistic SVBRDFs from a high-quality 3D material dataset [50]. We automatically map our SVBRDFs using deep features from a material estimation network, thus preserving scene semantics. We render the new scenes

Figure 2. Comparison of single-image object insertion on real images. Barron et al. [4] predict spatially varying log shading, but their lighting representation does not preserve high frequency signal and cannot be used to render shadows and inter-reflections. Gardner et al. [17] predict a single lighting for the whole scene and thus, cannot model spatial variations. Garon et al. [18] also predict spatially-varying lighting, but use spherical harmonics as their representation. Thus, it cannot model high frequency lighting well. In contrast, our method solves the indoor scene inverse rendering problem in a holistic way, which results in photorealistic object insertion. The quality of our output may be visualized in a video, without any temporal constraints, in **supplementary material**.

Figure 3. A material editing example on a real image, where we replace a material (on the kitchen counter-top) with a different one. Note the specular highlights on the surface, which cannot be handled by conventional intrinsic decomposition methods since they do not recover the lighting direction. In contrast, we recover spatially-varying lighting and material properties.

	Karsch 2014	Barron 2013	Eigen 2015	Gardner 2017	Li 2018	LeGendre 2019	Azionvić 2019	Garon 2019	Song 2019	Sengupta 2019	Ours
Geometry	✓	✓	✓	✗	✗	✗	✗	✗	✓	✓	✓
Reflectance	Diffuse	Diffuse	✗	✗	Diffuse	✗	Microfacet	✗	✗	Phong	Microfacet
Lighting	Local	Local	✗	Global	✗	Global	Local	Local	Local	Global	Local

Figure 4. A summary of scene-level inverse rendering. Karsch14's parametric lights cannot handle effects like shadowing [28]. Gardner17 [17] and Sengupta19 [44] predict a single lighting for the scene, thus, cannot handle spatial variations. Li18's shading entangles geometry and lighting [31]. Barron13 uses RGBD input and non-physical image formation [4]. Azinović19 [1] needs multiple images with 3D reconstruction as input. Our spherical Gaussians representation for local lighting is demonstrably better than spherical harmonics in Barron13 [4], Sengupta19 [44] and Garon19 [18]. Song19 [48] and several others do not handle complex SVBRDF.

using a GPU-based global illumination renderer, to create high-quality input images. We also render the new scene reflectance and lighting and use them to supervise our inverse rendering network. As far as we know, this is the first demonstration of mapping high-quality non-Lambertian, photorealistic materials to indoor scene datasets.

An inverse rendering network would have to learn a model of image formation. The forward image formation model is well understood, and has been used in simple settings like planar scenes and single objects [15, 33, 32, 35]. Indoor scenes are more complicated and exhibit challenging light transport effects like occlusions and inter-reflections. We address this by using a local lighting model—spatially-varying spherical gaussians (SVSGs). This bakes light transport effects directly into the lighting and makes rendering a purely local computation. We leverage this to design a fast, differentiable, *in-network* rendering layer that takes our geometry, SVBRDFs and SVSGs and computes radiance values. During training, we render our predictions and backpropagate the error through the rendering layer; this fixes the forward model, allowing the network to focus on the inverse task.

To the best of our knowledge, our work is the first demonstration of scene-level inverse rendering that truly accounts for complex geometry, materials and lighting, with effects like inter-reflections and shadows. Previous methods either solve a subset of the problem or rely on simplifying assumptions (Figure 4). Despite tackling a much harder problem, we obtain strong results on the individual tasks. Most important, by truly decomposing a scene into physically-based scene factors, we enable novel capabilities like photorealistic 3D object insertion and scene editing in images acquired in-the-wild. Figure 2 shows object insertion examples on real indoor images, where our method achieves superior performance compared to [4, 17, 18]. Figure 3 shows a material editing example, where we replace the material of a surface in a real image, while preserving spatially-varying specular highlights. Such visual effects cannot be handled by previous intrinsic decomposition methods. Extensive additional results are included in **supplementary material**.

2. Related Work

The problem of reconstructing shape, reflectance, and illumination from images has a long history in vision. It has been studied under different forms, such as intrinsic images (reflectance and shading from an image) [6] and shape-from-shading (shape, and sometimes reflectance, from an image) [22]. Here, we focus on *single* image methods.

Single objects. Many inverse rendering methods focus on reconstructing single objects. Even this problem is ill-posed and many methods assume some knowledge of the object in terms of known lighting [40, 23] or geometry [36, 43]. Recent methods have leveraged deep networks to reconstruct complex SVBRDFs from single images of planar scenes [15, 32], objects of a specific class [35] or homogeneous BRDFs [37]. Other methods address illumination estimation [19]. We tackle the much harder case of large-scale scene modeling and do not assume scene information. Barron and Malik [3] propose an optimization-based approach with hand-crafted priors to reconstruct shape, Lambertian reflectance, and distant illumination from a single image. Li et al. [33] tackle the same problem with a deep network and an object-specific rendering layer. Extending these methods to scenes is non-trivial because the light transport is significantly more complex.

Indoor scenes. Previous work recognizes materials in indoor scenes [9] and decomposes indoor images into reflectance and shading layers [8, 31]. Techniques have also been proposed for single image geometric reconstruction [16] and lighting estimation [21, 17]. Those methods estimate only one scene factor. Barron and Malik [4] reconstruct Lambertian reflectance and spatially-varying lighting but require RGBD input. Karsch et al. [27] estimate geometry, Lambertian reflectance and 3D lighting, but rely on extensive user input to annotate geometry and initialize lighting. An automatic, rendering-based optimization is proposed in [28] to estimate all these scene factors, but using strong heuristics that are often violated in practice. Recent deep networks also do not account for either spatially-varying lighting [44] or complex SVBRDF [56]. Several works are compared in Figure 4. In contrast to all those methods, our network learns to predict geometry, complex SVBRDFs and spatially-varying lighting in an end-to-end fashion.

Datasets. The success of deep networks has led to an interest in datasets for supervised training. This includes real world scans [14, 10], synthetic shape [11] and scene [49, 31, 44] datasets. All these datasets have unrealistic material (Lambertian or Phong) and lighting specifications. We build on the dataset of [49] to improve its quality in this regard, but our method is applicable to other datasets too.

Differentiable rendering. A number of recent deep inverse rendering methods have incorporated in-network, differentiable rendering layers that are customized for simple settings: faces [46, 52, 45], planar surfaces [15, 32], single objects [35, 33]. Some recent work has proposed differentiable general-purpose global illumination renderers [30, 12]; unlike our more specialized, fast rendering layer, these are too expensive to use for neural network training.

3. Indoor Dataset with Photorealistic Materials

It is extremely difficult, if at all possible, to acquire large-scale ground truth with spatially-varying material, lighting and global illumination. Thus, we render a synthetic dataset, but must overcome significant challenges to ensure utility for handling real indoor scenes at test time. Existing datasets for indoor scenes are rendered with simpler assumptions on material and lighting. In this section, we describe our approach to photorealistically map our microfacet materials to geometries of [49], while preserving semantics. Further, rendering images with SVBRDF and global illumination, as well as ground truth for spatially-varying lighting, is computationally intensive, for which we design a custom GPU-accelerated renderer that outperforms Mitsuba on a modern 16-core CPU by an order of magnitude (see supplementary material). Using the proposed method, we render 78794 HDR images at 480×640 resolution, with 72220 for training and 6574 for testing. We also render per pixel ground-truth lighting for 26719 training images and all test images, at a spatial resolution of 120×160. Our renderer will also be made publicly available.

3.1. Mapping photorealistic materials

Our goal is to map our materials to geometries such as [49] in a semantically meaningful way. Previous datasets are either rendered with Lambertian material [31] or use Phong BRDF [41] for their specular component [44], which is not suitable for complex materials [39]. Our materials, on the other hand, are represented by a physically motivated microfacet BRDF model [25][1]. This mapping is non-trivial: (i) Phong specular lobes are not realistic [39, 51], (ii) an optimization-based fitting collapses due to local minima leading to over-fitting when used for learning and (iii) we must replace materials with similar semantic types while being consistent with geometry, for example, replace material on walls with other paints and on sofas with other fabrics. Thus, we devise a three-step method (Figure 5).

Step 1: Tileable texture synthesis Directly replacing original textures with our non-tileable ones will create artifacts near boundaries. Most frameworks for tileable texture synthesis [34, 38] use randomized patch-based methods [2], which do not preserve structures such as sharp straight edges that are common for indoor scene materials. Instead, we first search for an optimal crop from our SVBRDF texture by minimizing gradients for diffuse albedo, normals and roughness perpendicular to the patch boundaries. We next

[1] Our dataset consists of 1332 materials with high resolution 4096×4096 SVBRDF textures. Please refer to the supplementary material for details.

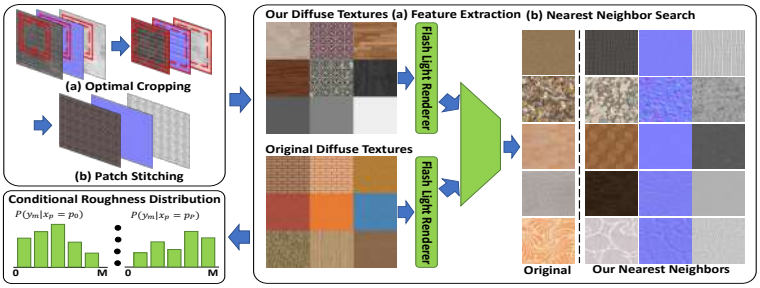

Figure 5. The pipeline of material mapping from original dataset with Phong BRDF to our microfacet BRDF. It has three steps. (Top left) Tileable texture synthesis to turn our SVBRDF textures into tileable ones. (Right) Spatially varying material mapping from original dataset with diffuse texture to our materials. (Bottom left) Homogeneous material mapping to convert specular parameters of homogeneous materials from Phong BRDF to our microfacet BRDF.

Figure 6. The first column is rendered with materials from our dataset. The second and third columns are images rendered with the original materials using Lambertian and Phong models. The image rendered with our materials has realistic specular highlights.

find the best seam for tiling by encouraging similar gradients at seams [29]. Please see supplementary material for details.

Step 2: Mapping SVBRDFs We may now replace original materials in a semantically meaningful way. Since the original specular reflectance is not realistic, we do this only for diffuse textures and directly use specularity from our dataset to render images. We manually divide textures from the two datasets into 10 categories based on appearance and semantic labels, such as fabric, stone or wood. We render both sets of diffuse textures on a planar surface under a flash light and use an encoder similar to [32] to extract features, then use nearest neighbors to map the materials. We randomly choose from 10 nearest neighbors for our dataset.

Step 3: Mapping homogeneous BRDFs For homogeneous materials, we keep the diffuse albedo unchanged and map specular Phong parameters to our microfacet model. Since the two lobes are very different, a direct fitting does not work. Instead, we compute a distribution of microfacet parameters conditioned on Phong parameters based on the mapping of diffuse textures, then randomly sample from that distribution. Specifically, let $\mathbf{x}_P \in \mathcal{P}$ be Phong specular parameters and $\mathbf{y}_M \in \mathcal{M}$ be those of our microfacet BRDF. If a material in the original dataset has specular parameters $\mathbf{x}_P = \mathbf{p}_b$, we count the number of pixels in its 10 nearest neighbors from our dataset whose specular parameters are $\mathbf{y}_M = \mathbf{m}_a$. We sum up the number across the whole dataset as $N(\mathbf{m}_a, \mathbf{p}_b)$. The probability of material with specularity $\mathbf{y}_M$ given the original material has specularity $\mathbf{x}_P$ is:

$$P(\mathbf{y}_M = \mathbf{m}_a | \mathbf{x}_P = \mathbf{p}_b) = \frac{N(\mathbf{p}_b, \mathbf{m}_a)}{\sum_{\mathbf{m}_c \in \mathcal{M}} N(\mathbf{p}_b, \mathbf{m}_c)}.$$

Comparative results Figure 6 compares rendering with Lambertian, Phong and our BRDF models. The Lambertian image does not have any specularity, Phong has strong but flat specularity, while ours has realistic highlights. All materials in our rendering are tiled well and assigned to correct objects, which shows the effectiveness of our mapping.

3.2. Spatially Varying Lighting

To enable tasks such as object insertion or material editing, we must estimate lighting at every spatial location that encodes complex global interactions. We obtain ground truth by rendering a 16×32 environment map at the corresponding 3D point on object surfaces at every pixel. In Figure 8, we show that an image obtained by integrating the product of this lighting and BRDF over the hemisphere is very close to the original, with high frequency specular highlights correctly rendered. Note that global illumination and occlusion have already been baked into per-pixel lighting, which makes it possible for a model trained on our lighting dataset to reason about those complex effects.

4. Network Design

Estimating spatially-varying lighting, complex SVBRDF and geometry from a single indoor image is an extremely ill-posed problem, which we solve using priors learned by our physically-motivated deep network (architecture shown in Figure 7). Our network consists of cascaded stages of a SVBRDF and geometry predictor, a spatially-varying lighting predictor and a differentiable rendering layer, followed by a bilateral solver for refinement.

Material and geometry prediction The input to our network is a single gamma-corrected low dynamic range image I, stacked with a predicted three-channel segmentation mask $\{\tilde{M}_o, \tilde{M}_a, \tilde{M}_e\}$ that separates pixels of object, area lights and environment map. The mask is obtained through a pre-trained network and useful since some predictions are not defined everywhere (for example, BRDF is not defined on light sources). Inspired by [32, 33], we use a single encoder to capture correlations between material and shape parameters, obtained using four decoders for diffuse albedo (A), roughness (R), normal (N) and depth (D). Skip links are used for preserving details. Then the initial estimates of

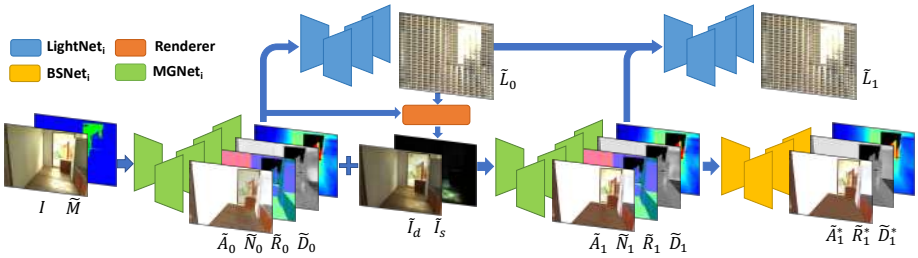

Figure 7. Our network design consists of a cascade, with one encoder-decoder for material and geometry prediction and another one for spatially-varying lighting, along with a physically-based differentiable rendering layer and a bilateral solver for refinement.

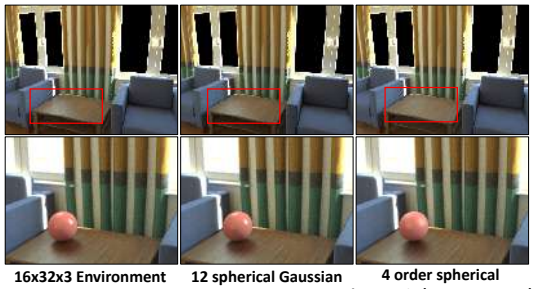

Figure 8. Comparisons of images rendered with lighting approximations. The first row: images rendered by our rendering layer using ground-truth normals and materials but with different lighting representations. The second row: inserting a sphere into the scene. In both examples, we can clearly see that spherical Gaussians can recover high frequency lighting much better with fewer parameters.

material and geometry are given by

$$\tilde{A}, \tilde{N}, \tilde{R}, \tilde{D} = \mathbf{MGNet}_0(I, M). \quad (1)$$

Spatially Varying Lighting Prediction Inverse rendering for indoor scenes requires predicting spatially varying lighting for every pixel in the image. Using an environment map as the lighting representation leads to a very high dimensional output space, that causes memory issues and unstable training due to small batch sizes. Spherical harmonics are a compact lighting representation that have been used in recent works [24, 33], but do not efficiently recover high frequency lighting necessary to handle specular effects [42, 7]. Instead, we follow pre-computed radiance transfer methods [53, 20, 54] and use isotropic spherical Gaussians that approximate all-frequency lighting with a smaller number of parameters. We model the lighting as a spherical function $L(\eta)$ approximated by the sum of spherical Gaussian lobes:

$$L(\eta) = \sum_{k=1}^{K} F_k G(\eta; \xi_k, \lambda_k), \quad G(\eta; \xi, \lambda) = e^{-\lambda(1-\eta\cdot\xi)}, \quad (2)$$

where η and ξ are vectors on the unit sphere $\mathcal{S}^2$, F_k controls RGB color intensity and λ controls the bandwidth.

Each spherical Gaussian lobe is represented by 6 dimensional parameters $\{\xi_k, \lambda_k, F_k\}$. Figure 8 compares the images rendered with a 12-spherical Gaussian lobes approximation (72 parameters) and a fourth-order spherical harmonics approximation (75 parameters). Quantitative comparisons of lighting approximation and rendering errors are in supplementary material. It is evident that even using fewer parameters, the spherical Gaussian lighting performs better, especially close to specular regions.

Our novel lighting prediction network, $\mathbf{LightNet}_0(\cdot)$, accepts predicted material and geometry as input, along with the image. It uses a shared encoder and separate decoders to predict $\{\tilde{\xi}_k\}, \{\tilde{\lambda}_k\}, \{\tilde{F}_k\}$. Please refer to supplementary material on how to predict spherical Gaussian parameters.

$$\{\tilde{\xi}_k\}, \{\tilde{\lambda}_k\}, \{\tilde{F}_k\} = \mathbf{LightNet}_0(I, \tilde{M}, \tilde{A}, \tilde{N}, \tilde{R}, \tilde{D}). \quad (3)$$

Our predicted lighting is HDR, which is important for applications like relighting and material editing.

Differentiable rendering layer Our dataset in Section 3 provides ground truth for all scene components. But to model realistic indoor scene appearance, we additionally use a differentiable in-network rendering layer to mimic the image formation process, thereby weighting those components in a physically meaningful way. We implement this layer by numerically integrating the product of SVBRDF f and spatially-varying lighting L over the hemisphere. Let $l_{ij} = l(\phi_i, \theta_j)$ be a set of light directions sampled over the upper hemisphere, with v the view direction. The rendering layer computes diffuse $\tilde{I}_d$ and specular images $\tilde{I}_s$ as:

$$\tilde{I}_d = \sum_{i,j} f_d(v, l_{ij}; \tilde{A}, \tilde{N}) L\left(l_{ij}; \{\xi_k, \lambda_k, F_k\}\right) \cos\theta_j \mathrm{d}\omega, \quad (4)$$

$$\tilde{I}_s = \sum_{i,j} f_s(v, l_{ij}; \tilde{R}, \tilde{N}) L\left(l_{ij}; \{\xi_k, \lambda_k, F_k\}\right) \cos\theta_j \mathrm{d}\omega, \quad (5)$$

where $\mathrm{d}\omega$ is the differential solid angle. We sample 16×8 lighting directions. While this is relatively low resolution, we empirically find, as shown in Figure 8, that it is sufficient to recover most high frequency lighting effects.

Loss Functions Our loss functions incorporate physical insights. We first observe that two ambiguities are difficult to resolve: the ambiguity between color and light intensity, as well as the scale ambiguity of single image depth estimation. Thus, we allow the related loss functions to be scale invariant. For material and geometry, we use the scale invariant L_2 loss for diffuse albedo ($\mathcal{L}_A$), L_2 loss for normal ($\mathcal{L}_N$) and roughness ($\mathcal{L}_R$) and a scale invariant log-encoded loss for depth ($\mathcal{L}(D)$) due to its high dynamic range:

$$\mathcal{L}_D = \|(\log(D+1) - \log(c_d\tilde{D}+1)) \odot (M_a + M_o)\|_2^2, \quad (6)$$

where c_d is a scale factor computed by least squares regression. For lighting estimation, we find supervising both the environment maps and spherical Gaussian parameters is important for preserving high frequency details. Thus, we compute ground-truth spherical Gaussian lobe parameters by approximating the ground-truth lighting using the LBFGS method, as detailed in supplementary material. We use the same scale invariant log-encoded loss as (8) for weights ($\{\mathcal{L}_{F_k}\}$), bandwidth ($\{\mathcal{L}_{\lambda_k}\}$) and lighting ($\{\mathcal{L}_L\}$), with an L_2 loss for direction (L_{ξ_k}). We also add a a scale invariant L_2 rendering loss:

$$\mathcal{L}_{ren} = ||(I - c_{diff}\tilde{I}_d - c_{spec}I_s) \odot M_o||_2^2 \quad (7)$$

where $\tilde{I}_d$ and $\tilde{I}_s$ are rendered using (4) and (5), respectively, while c_{diff} and c_{spec} are positive scale factors computed using least square regression. The final loss function is a weighted summation of the proposed losses:

$$\mathcal{L} = \alpha_A\mathcal{L}_A + \alpha_N\mathcal{L}_N + \alpha_R\mathcal{L}_R + \alpha_D\mathcal{L}_D + \alpha_L\mathcal{L}_L \\ \alpha_{ren}\mathcal{L}_{ren} + \sum_{k=1}^{K} \alpha_\lambda\mathcal{L}_{\lambda_k} + \alpha_\xi\mathcal{L}_{\xi_k} + \alpha_F\mathcal{L}_{F_k}. \quad (8)$$

Refinement using bilateral solver We use an end-to-end trainable bilateral solver to impose a smoothness prior [5, 31]. The inputs include the prediction, the estimated diffuse albedo $\tilde{A}$ as a guidance image and confidence map C. We train a shallow network with three sixteen-channel layers for confidence map predictions. Let $\mathbf{BS}(\cdot)$ be the bilateral solver and $\mathbf{BSNet}_X(\cdot)$ be the network for confidence map predictions where $X \in \{A, R, D\}$. We do not find refinement to have much effect on normals. The refinement process is:

$$\tilde{C}_X = \mathbf{BSNet}(\tilde{X}, I, \tilde{M}), \quad X \in \{A, R, D\} \quad (9)$$
$$\tilde{X}^* = \mathbf{BS}(\tilde{X}; C_X, \tilde{A}) \quad (10)$$

where we use $(^*)$ for predictions after refinement.

Cascade Network Akin to recent works on high resolution image synthesis [26, 13] and inverse rendering [33], we introduce a cascaded network that progressively increases resolution and iteratively refines the predictions through global reasoning. We achieve this by sending both the predictions and the rendering layer applied on the predictions to the next cascade stages, $\mathbf{MGNet}_1(\cdot)$ for material and geometry and $\mathbf{LightNet}_1(\cdot)$ for lighting, so that the network can reason about their differences. Cascade stages have similar architectures as their initial network counterparts.

5. Experiments

We now conduct studies on the roles of various components in our pipeline, compare to prior works and illustrate applications such as high quality object insertion and material editing in real images that can only be enabled by our holistic solution to inverse rendering.

	Cascade 0		Cascade 1		
	Ind.	Joint	Ind.	Joint	BS
$A(10^{-2})$	1.28	1.28	1.18	1.18	**1.16**
$N(10^{-2})$	4.91	4.91	4.51	4.51	**4.51**
$R(10^{-1})$	1.72	1.72	1.72	1.72	**1.70**
$D(10^{-2})$	8.06	8.00	7.29	7.26	**7.20**

Table 1. Quantitative comparisons of shape and material reconstructions on our test set. We use scale invariant L2 error for diffuse albedo (A), scale invariant $\log^2$ error for depth (D) and L2 error for normal (N) and roughness (R).

	Cascade 0				Cascade 1	
	No MG	No SG	Ind.	Joint	Ind.	Joint
L	2.87	2.85	2.54	2.50	2.49	2.43
$I(10^{-2})$	4.91	1.55	1.56	1.06	1.92	1.11

Table 2. Quantitative comparison of lighting predictions on test set. We use scale invariant L2 error for rendered image (I) and scale invariant $\log^2$ error for lighting (L).

5.1. Analysis of Network and Training Choices

We study the effect of the cascade structure, joint training and refinement. Quantitative results for material and geometry predictions on the proposed dataset are summarized in Table 1, while those for lighting are shown in Table 2.

Cascade The cascade structure leads to clear gains for shape, BRDF and lighting estimation by iteratively improving and upsampling our predictions in Tables 1 and 2. This holds for real data too, as shown in Figure 10. We observe that the cascade structure can effectively remove noise and preserve high frequency details for both materials and lighting. The errors in our shape, material and lighting estimates are low enough to photorealistically edit the scene to insert new objects, while preserving global illumination effects.

Joint training for inverse rendering Next we study whether BRDF, shape and lighting predictions can help improve each other. We compare jointly training the whole pipeline ("Joint") using the loss in (8) and compare to independently training ("Ind") each component $\mathbf{MGNet}_i$ and $\mathbf{LightNet}_i$. Quantitative errors in Tables 1 and 2 show that while shape and BRDF errors remain similar, those for rendering and lighting decrease. Next, we test lighting predictions without predicted BRDF as input for the first level of cascade ("No MG"). Both quantitative results in Table 2 and qualitative comparison in supplementary material demonstrate that the predicted BRDF and shape are important to recover spatially varying lighting. This justifies our choice of jointly reasoning about shape, material and lighting. We also test lighting predictions with and without ground-truth SVSG parameters as supervision ("No SG"), finding that direct supervision leads to a sharper lighting prediction.

Refinement Finally, we study the impact of the bilateral solver. Quantitative improvements over the second cascade stage in Table 1 are modest, which indicates that the network

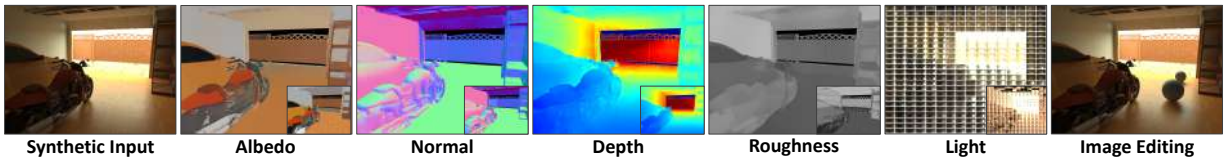

Figure 9. Results on a synthetic image. Given a single input image, our estimated albedo, normals, depth, roughness and lighting are close to ground truth shown as insets. These are used for object insertion (right).

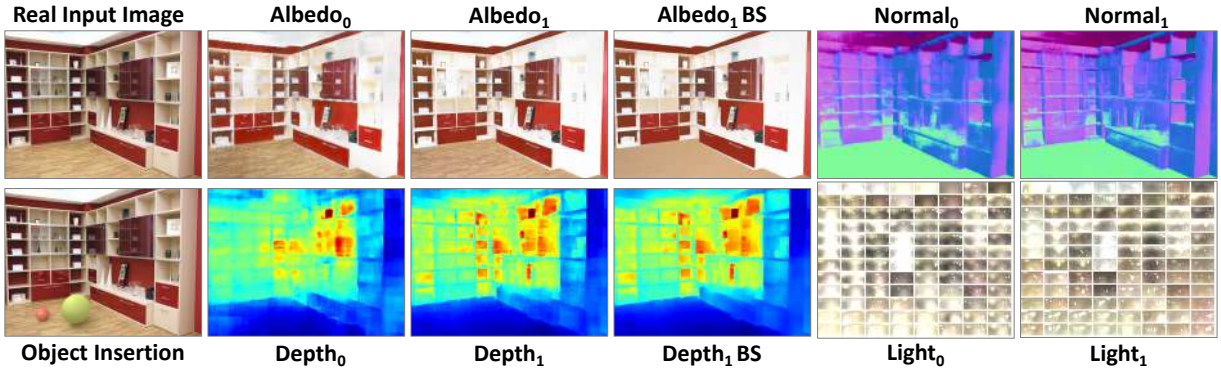

Figure 10. Results on a real image, for single-image depth, normals, spatially-varying material and lighting. Improvements are observed due to the cascade structure and bilateral solver. The estimates are accurate enough to insert a novel object with realistic global illumination.

Method	Training Set	WHDR
Ours (cascade 0)	Ours	23.29
Ours (cascade 1)	Ours	21.99
Ours (cascade 0)	Ours + IIW	16.83
Ours (cascade 1)	Ours + IIW	**15.93**
Li. et al[31]	CGI + IIW	17.5

Table 3. Intrinsic decomposition on the IIW dataset. Lower is better for the WHDR metric.

already learns good smoothness priors by that stage. But we find the qualitative impact of the bilateral solver to be noticeable on real images (for example, diffuse albedo in Figure 10), thus, we use it in all our real experiments.

Qualitative examples In Figure 9, we use a single input image from our synthetic test set to demonstrate depth, normal, SVBRDF and spatially-varying lighting estimation. The effectiveness is illustrated by low errors with respect to ground truth. Accurate shading and global illumination effects on an inserted object, as well as photorealistic editing of scene materials, show the utility of our decomposition.

5.2. Comparisons with Previous Works

We address the problem of holistic inverse rendering with spatially-varying material and lighting which has not been tackled earlier. Yet, it is instructive to compare our approach to prior ones that focus on specific sub-problems.

Intrinsic decomposition We compare two versions of our method on the IIW dataset [8] for intrinsic decomposition evaluation: our network trained on our data alone and our network fine-tuned on the IIW dataset. The results are tabulated in Table 3. We observe that the cascade structure is beneficial. We also observe a lower error compared to the prior work of [31], which indicates the benefit of our dataset that is rendered with a higher photorealism, as well as a network design that closely reflects physical image formation.

Method	Mean(°)	Median(°)	Depth(Inv.)
Ours (cascade 0)	25.09	18.00	0.184
Ours (cascade 1)	24.12	17.27	0.176

Table 4. Normal and depth estimation on NYU dataset [47].

Lighting estimation We compare with [4] on our test set. Our scale-invariant shading errors on {R, G, B} channels are $\{0.87, 0.86, 0.83\}$, compared to their $\{2.33, 2.10, 1.90\}$. Our physically-motivated network trained on a photorealistic dataset leads to this improvement. Next, we compare with the work of Gardner et al. [17]. Quantitative results on our test set show that their mean $\log L_2$ error across the whole image is 3.34 while ours is 2.43. Qualitative results are shown in Figure 2 and supplementary material. Since only one environment lighting for the whole scene is predicted by [17], no spatially-varying lighting effects can be observed.

Depth and normal estimation We fine-tune our network, trained on our synthetic dataset, on NYU dataset [47]. Please refer to supplementary material for more training details. The test error on NYU dataset is summarized in Table 4. For both depth and normal prediction, the cascade structure consistently helps improve performance. Zhang et al. [55] achieve state-of-the-art performance for normal estimation using a more complex fine-tuning strategy and with more than six times as much training data. Eigen et al. [16] achieve better results by using 120K frames of raw video data, while we pre-train on synthetic images with larger domain gap, and only use 795 images from NYU dataset for fine-tuning. Although we do not achieve state-of-the-art performance on this task, it's not our main focus. Rather, we aim to show the wide utility of our proposed dataset and demonstrate

Figure 11. Comparisons of object insertion on real images of Garon et al. [18]. Our overall appearances look more realistic. For example, note the bunny under bright light (top right) in the top row and in the shadow (bottom middle) in bottom row. Also see Table 5.

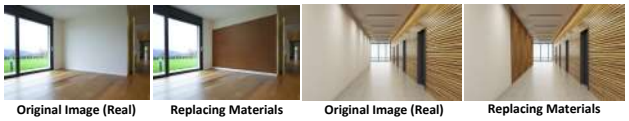

Figure 12. Material editing on real images. Left is the original image and right is the rendered one with the material replaced in a part of the scene. We observe that the edited material looks photo-realistic and even high frequency details from specular highlights and spatially-varying lighting are rendered well.

Method	Barron15	Gardner17	Garon19	Ours
Single objects	12.6%	27.0%	32.6%	**33.9%**
Multi objects	12.9%	26.1%	30.0%	**33.6%**

Table 5. Object insertion user study on the dataset of [18].

estimation of factors of image formation good enough to support photo-realistic augmented reality applications.

Object insertion Given a single real image, we insert a novel object with photorealistic shading, specularity and global light transport effects. This is a crucial ability for high quality augmented reality applications. To simplify the demonstration, we estimate the shape, material and lighting using our cascade network, then select a planar region of the scene to insert an object. We relight the object using the estimated lighting. It may be observed on qualitative examples in Figures 1(h), 2, 10 and 11 (all containing real images) that even complex visual effects such as shadows and reflections from other parts of the scene are faithfully rendered on the inserted object. Further, [18] provides a dataset of 20 real indoor images with ground truth spatially-varying lighting. For each image, we render a virtual bunny into the scene lit by ground-truth or predicted lighting (Figure 11). We also performed an AMT user study on these images. Following the protocol in [18], users are shown image pairs rendered with ground truth and estimated lighting, and asked to pick which is more realistic (50% is ideal performance). As shown in Tab. 5, we outperform prior methods, both when objects are inserted at a single or multiple locations.

Material Editing Editing material properties of a scene using a single photograph has applications for interior design and visualization. Our disentangled shape, material and lighting estimation allows rendering new appearances by replacing materials and rendering using the estimated lighting. In Figures 3 and 12 (all real images), we replace the material of a planar region with another kind of material and render the image using the predicted geometry and lighting, whose spatial variations are clearly observable. In the first example in Figure 3, we can see the specular highlight in the original image is preserved after changing the material. This is not possible for intrinsic decomposition methods, which cannot determine incoming lighting direction.

Supplementary material contains details for: (i) tileable texture synthesis (ii) renderer (iii) optimization for SVSG ground truth (iv) SG parameter prediction (v) SVSG comparison with SH (vi) SVBRDF dataset (vii) training strategy. It includes several additional examples for estimating scene factors on real images, object insertion and material editing.

6. Conclusions

We have presented the first holistic inverse rendering framework that estimates disentangled shape, SVBRDF and spatially-varying lighting, from a single image of an indoor scene. Insights from computer vision, graphics and deep convolutional networks are utilized to solve this challenging ill-posed problem. A GPU-accelerated renderer is used to synthesize a large-scale, realistic dataset with complex materials and global illumination. Our per-pixel SVSG lighting representation captures high frequency effects. Our network imbibes intuitions such as a differentiable rendering layer, which are crucial for generalization to real images. Design choices such as a cascade structure and a bilateral solver lead to further benefits. Despite solving the joint problem, we obtain strong results on various sub-problems, which highlights the impact of our dataset, representations and network. We demonstrate object insertion and material editing on real images that capture global illumination effects, motivating applications in augmented reality and interior design.

Acknowledgements: Z. Li and M. Chandraker are supported by NSF CAREER 1751365 and a Google Research Award, M. Shafiei and R. Ramamoorthi by ONR grant N000141712687.

References

[1] Dejan Azinović, Tzu-Mao Li, Anton Kaplanyan, and Matthias Nießner. Inverse path tracing for joint material and lighting estimation. *arXiv preprint arXiv:1903.07145*, 2019. 2

[2] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. PatchMatch: A randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 28(3), Aug. 2009. 3

[3] Jonathan Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *PAMI*, 37(8):1670–1687, 2013. 3

[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, 3, 7

[5] Jonathan T Barron and Ben Poole. The fast bilateral solver. In *European Conference on Computer Vision*, pages 617–632. Springer, 2016. 6

[6] Harry G. Barrow and J. Martin Tenenbaum. Recovering intrinsic scene characteristics from images. *Computer Vision Systems*, pages 3–26, 1978. 3

[7] Ronen Basri and David W. Jacobs. Lambertian reflectance and linear subspaces. *PAMI*, 25(2), 2003. 5

[8] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics (TOG)*, 33(4):159, 2014. 3, 7

[9] Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. *Computer Vision and Pattern Recognition (CVPR)*, 2015. 1, 3

[10] Angel Chang, Angela Dai, Thomas Funkhouser, Maciej Halber, Matthias Niessner, Manolis Savva, Shuran Song, Andy Zeng, and Yinda Zhang. Matterport3D: Learning from RGB-D data in indoor environments. *International Conference on 3D Vision (3DV)*, 2017. 1, 3

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

[12] Chengqian Che, Fujun Luan, Shuang Zhao, Kavita Bala, and Ioannis Gkioulekas. Inverse transport networks. *arXiv preprint arXiv:1809.10820*, 2018. 3

[13] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1511–1520, 2017. 6

[14] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proc. Computer Vision and Pattern Recognition (CVPR), IEEE*, 2017. 1, 3

[15] Valentin Deschaintre, Miika Aittala, Fredo Durand, George Drettakis, and Adrien Bousseau. Single-image svbrdf capture with a rendering-aware deep network. *ACM Transactions on Graphics (TOG)*, 37(4):128, 2018. 2, 3

[16] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 1, 3, 7

[17] 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 Trans. Graphics*, 9(4), 2017. 1, 2, 3, 7

[18] Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. Fast spatially-varying indoor lighting estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6908–6917, 2019. 2, 8

[19] Stamatios Georgoulis, Konstantinos Rematas, Tobias Ritschel, Mario Fritz, Tinne Tuytelaars, and Luc Van Gool. What is around the camera? In *ICCV*, 2017. 3

[20] Paul Green, Jan Kautz, and Frédo Durand. Efficient reflectance and visibility approximations for environment map rendering. In *Computer Graphics Forum*, volume 26, pages 495–502. Wiley Online Library, 2007. 5

[21] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-François Lalonde. Deep outdoor illumination estimation. In *CVPR*, 2017. 3

[22] Berthold K. P. Horn and Michael J. Brooks, editors. *Shape from Shading*. MIT Press, Cambridge, MA, USA, 1989. 3

[23] M. K. Johnson and E. H. Adelson. Shape estimation in natural illumination. In *CVPR*, 2011. 3

[24] Yoshihiro Kanamori and Yuki Endo. Relighting humans: occlusion-aware inverse rendering for fullbody human images. *SIGGRAPH Asia*, 37(270):1–270, 2018. 5

[25] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 4, 2013. 3

[26] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 6

[27] Kevin Karsch, Varsha Hedau, David Forsyth, and Derek Hoiem. Rendering synthetic objects into legacy photographs. *ACM Transactions on Graphics*, 30(6):1, 2011. 3

[28] 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*, (3):32:1–32:15, 2014. 2, 3

[29] Vivek Kwatra, Arno Schödl, Irfan Essa, Greg Turk, and Aaron Bobick. Graphcut textures: image and video synthesis using graph cuts. *TOG*, 22(3):277–286, 2003. 4

[30] Tzu-Mao Li, Miika Aittala, Frédo Durand, and Jaakko Lehtinen. Differentiable monte carlo ray tracing through edge sampling. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)*, 37(6):222:1–222:11, 2018. 3

[31] Zhengqi Li and Noah Snavely. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In *ECCV*, pages 371–387, 2018. 2, 3, 6, 7

[32] Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. Materials for masses: Svbrdf acquisition with a single mobile phone image. In *ECCV*, pages 72–87, 2018. 2, 3, 4

[33] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single

image. In *SIGGRAPH Asia*, page 269. ACM, 2018. 2, 3, 4, 5, 6

[34] Lin Liang, Ce Liu, Ying-Qing Xu, Baining Guo, and Heung-Yeung Shum. Real-time texture synthesis by patch-based sampling. *ACM Transactions on Graphics (ToG)*, 20(3):127–150, 2001. 3

[35] Guilin Liu, Duygu Ceylan, Ersin Yumer, Jimei Yang, and Jyh-Ming Lien. Material editing using a physically based rendering network. In *ICCV*, 2017. 2, 3

[36] Stephen Lombardi and Ko Nishino. Reflectance and natural illumination from a single image. In *ECCV*, 2012. 3

[37] Abhimitra Meka, Maxim Maximov, Michael Zollhoefer, Avishek Chatterjee, Hans-Peter Seidel, Christian Richardt, and Christian Theobalt. Lime: Live intrinsic material estimation. In *CVPR*, 2018. 3

[38] Joep Moritz, Stuart James, Tom SF Haines, Tobias Ritschel, and Tim Weyrich. Texture stationarization: Turning photos into tileable textures. In *Computer Graphics Forum*, volume 36, pages 177–188. Wiley Online Library, 2017. 3

[39] Addy Ngan, Frédo Durand, and Wojciech Matusik. Experimental analysis of brdf models. *Rendering Techniques*, 2005(16th):2, 2005. 3

[40] Geoffrey Oxholm and Ko Nishino. Shape and reflectance from natural illumination. In *ECCV*, 2012. 3

[41] Bui Tuong Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6):311–317, 1975. 3

[42] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *SIGGRAPH*, 2001. 5

[43] Fabiano Romeiro and Todd Zickler. Blind reflectometry. In *ECCV*, 2010. 3

[44] 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. *arXiv preprint arXiv:1901.02453*, 2019. 2, 3

[45] Soumyadip Sengupta, Angjoo Kanazawa, Carlos D. Castillo, and David W. Jacobs. Sfsnet: Learning shape, refectance and illuminance of faces in the wild. In *CVPR*, 2018. 3

[46] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. In *CVPR*, 2017. 3

[47] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from RGBD images. In *ECCV*, 2012. 7

[48] Shuran Song and Thomas Funkhouser. Neural illumination: Lighting prediction for indoor environments. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6918–6926, June 2019. 2

[49] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. *Proceedings of 30th IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 1, 3

[50] Adobe Stock. Royalty-free 3d assets to enhance your projects, 2017. 1

[51] Tiancheng Sun, Henrik Wann Jensen, and Ravi Ramamoorthi. Connecting measured brdfs to analytic brdfs by data-driven diffuse-specular separation. *ACM Transactions on Graphics (TOG)*, 37(6):273, 2018. 3

[52] A. Tewari, M. Zollhofer, H. Kim, P. Garrido, F. Bernard, P. Perez, and C. Theobalt. Mofa: Model-based deep convolutional face autoencoder for unsupervised monocular reconstruction. In *ICCV*, 2018. 3

[53] Yu-Ting Tsai and Zen-Chung Shih. All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. In *TOG*, volume 25, pages 967–976. ACM, 2006. 5

[54] Kun Xu, Wei-Lun Sun, Zhao Dong, Dan-Yong Zhao, Run-Dong Wu, and Shi-Min Hu. Anisotropic spherical gaussians. *ACM Transactions on Graphics (TOG)*, 32(6):209, 2013. 5

[55] Yinda Zhang, Shuran Song, Ersin Yumer, Manolis Savva, Joon-Young Lee, Hailin Jin, and Thomas Funkhouser. Physically-based rendering for indoor scene understanding using convolutional neural networks. *CVPR*, 2017. 1, 7

[56] Hao Zhou, Xiang Yu, and David W. Jacobs. Glosh: Global-local spherical harmonics for intrinsic image decomposition. In *ICCV*, pages 7820–7829, October 2019. 3