Spatially-Varying Outdoor Lighting Estimation from Intrinsics

Yongjie Zhu[1] † Yinda Zhang[2] Si Li[1] * Boxin Shi[3, 4] *
[1]School of Artificial Intelligence, Beijing University of Posts and Telecommunications
[2]Google [3]NELVT, Department of Computer Science and Technology, Peking University
[4]Institute for Artificial Intelligence, Peking University

Abstract

We present SOLID-Net, a neural network for spatially-varying outdoor lighting estimation from a single outdoor image for any 2D pixel location. Previous work has used a unified sky environment map to represent outdoor lighting. Instead, we generate spatially-varying local lighting environment maps by combining global sky environment map with warped image information according to geometric information estimated from intrinsics. As no outdoor dataset with image and local lighting ground truth is readily available, we introduce the SOLID-Img dataset with physically-based rendered images and their corresponding intrinsic and lighting information. We train a deep neural network to regress intrinsic cues with physically-based constraints and use them to conduct global and local lightings estimation. Experiments on both synthetic and real datasets show that SOLID-Net significantly outperforms previous methods.

1. Introduction

Estimating outdoor lighting from a single image is one of the fundamental problems in computer vision. By providing outdoor scene properties from the physical aspect, it has huge impact on many applications, *e.g.*, face/body relighting, scene understanding, augmented reality (AR), and so on. This task is rather challenging since images are formed by conflating lighting with complex surface reflectance distribution and object geometry. In the outdoor scenario, existing solutions usually employ low-dimensional parametric models such as the Hošek-Wilkie (HW) sky model [11] with four parameters to fit the sky illumination. The capacity of parametric models is not sufficient to represent the complex real-world illumination, and a recent non-parametric approach using an autoencoder to learn the sky illumination model from a large-scale sky panorama dataset and encoding the lighting information from a single limited Field-of-View (FOV) image shows more promising results [10].

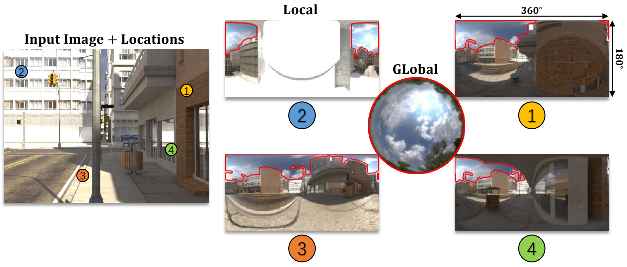


Figure 1: Given a single low-dynamic-range (LDR) image with limited FOV and a location in pixel coordinate (marked by numbers), SOLID-Net, for the first time, infers a panoramic HDR illumination map representing the light arriving from all directions at the location. Note that the global environment map (could be estimated using existing method [10]) is only able to cover a small part of the local lighting (red contours).

However, as far as we know, all existing outdoor lighting estimation methods [11, 10, 24] only consider the outdoor illumination as a single global map without any spatially-varying consideration, *i.e.*, the light probe is surrounded by an environment map that casts rays from infinitely far away. A spatially-varying lighting estimation has proved to be successful in indoor scenarios, which is achieved by modeling local indoor lighting using low-frequency parametric lighting represented by spherical harmonics (SH) [9, 5] or panoramic environment map [18].

Extending spatially-varying lighting estimation from indoor to outdoor is non-trivial in three aspects: 1) The extremely high-dynamic-range (HDR) sunlight and the complicated sky light under different weather conditions make it more difficult to parameterize outdoor than indoor lighting [9, 5], while the existing non-parametric sky model [10] treats it as a pure deep learning task without considering physics image formation constraint. 2) Non-parametric

* Corresponding authors. † Part of this work was finished as a visiting student at Peking University. This work was supported by National National Natural Science Foundation of China under Grant No. 61872012, 62088102, National Key R&D Program of China (2019YFF0302902), and Beijing Academy of Artificial Intelligence (BAAI).

spatially-varying local lighting estimation is highly ill-posed, since different 3D locations should have different lighting observations and the majority of local observation is missing [18]. 3) HDR and panoramic images capturing local lighting and geometry information in outdoor are not yet available, despite there are many datasets for such a purpose in the indoor scenario by synthetically generating scenes from SUNCG [19] and Matterport3D [4].

In this paper, we propose **SOLID-Net**, a neural network for **S**patially-varying **O**utdoor **L**ighting estimation using cues from **I**ntrinsic image **D**ecomposition, as shown in Figure 1. We tackle the three major challenges mentioned above by proposing a two-stage framework: 1) We train a single-in-multi-output CNN to decompose an input image into intrinsic parts: albedo (material-related), normal and plane distance (geometry-related), and shadow (lighting-related). These intrinsics provide a physically-based shading constraint by fitting SH-represented global lighting with low-frequency information, which is then combined with extracted sky features from the input image to generate a non-parametric sky model like [10]. 2) With the estimated geometry from decomposed intrinsics, we further warp the input image and estimated shadow map with limited FOV to a spherical projection centered at the target location, which provides panoramic observation to reduce the ill-posed issue. This is then combined with global sky lighting from the previous step as input to train a multi-input-single-output CNN for complementing high-frequency local lighting estimation. 3) We use the Blender SceneCity [2] to create city models that contain a large set of outdoor scenes and render a synthetic outdoor lighting estimation dataset with labeled location information and corresponding lighting effects using a physically-based path-tracer to facilitate the training of our network. SOLID-Net demonstrates significant improvements over other methods by making contributions in

- integrating shading constraint from intrinsic decomposition into the global sky lighting estimation;
- producing high-frequency local lighting estimation via panoramic warping and shadow map reference; and
- building the first spatially-varying outdoor lighting estimation dataset with ground truth labels.

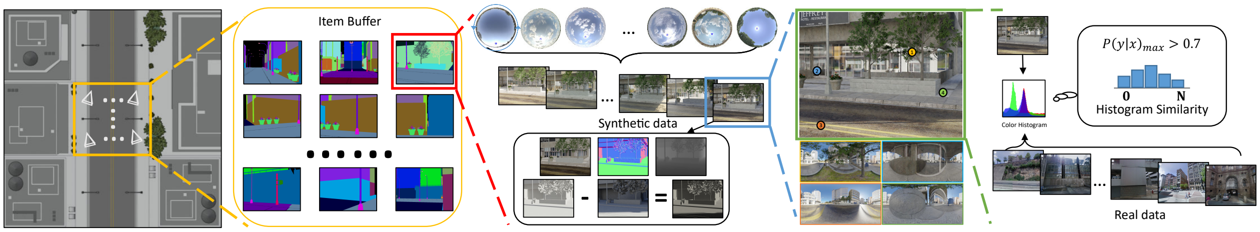
2. Related Work

Outdoor lighting estimation. Stumpfel *et al.* [21] proposed to explicitly capture the HDR outdoor lighting environments that include the sun and sky with multiple exposures. Lalonde *et al.* [12] first proposed lighting estimation from a single, generic outdoor scene. Their approach relied on multiple cues (such as shadows, shading, and sky appearance variation) extracted from the image. There are solutions using parametric models to represent outdoor lighting: Cheng *et al.* [6] estimated lighting from the front and back camera of a mobile phone. However, they represented lighting using low-frequency SH, which does not appropriately model outdoor lighting. Hold-Geoffroy *et al.* [11] learned to estimate Hošek-Wilkie (HW) sky model parameters from a single image, which is further extended by Zhang *et al.* [24] with a more flexible parametric Lalonde-Matthews (LM) sky model. To include more information of the sky, Hold-Geoffroy *et al.* [10] designed an autoencoder to learn a non-parametric sky model from a large sky panorama dataset [13] and trained a network to learn the sky lighting from limited-FOV images. LeGendre *et al.* [14] used a mobile phone camera with three different reflective spheres to capture lighting ground truth and used these data to train their deep model effectively, but these spheres are still global lighting probes.

Local lighting estimation. A direct way of obtaining the local lighting of an environment is to capture the lighting intensity at a target location using a probe of known shape. Debevec *et al.* [7] showed that HDR environment maps can be captured with a reflective metallic sphere captuerd with the scene. Barron an Malik [5] decomposed the scene into intrinsic components including spatially-varying SH-based lighting, but it required an RGBD image as input and relied on hand-crafted priors. To learn local lighting representation, Garon *et al.* [9] predicted fifth-order SH coefficients from an input image and local patches with synthetic data. In more recent progress, Li *et al.* [15] proposed a dense spherical Gaussian lighting representation with differentiable rendering to conduct scene editing. But all the methods mentioned above only considered indoor parametric lighting and are hard to be extended to outdoor lighting. Song *et al.* [18] proposed a cascaded model (denoted as NeurIllum for brevity) to recover high-frequency local lighting with warped color image according to recovered geometry, which showed promising texture details but the lighting positions are sometimes less accurate due to the loss of massive information in the panorama.

3. Dataset

A large dataset containing HDR images and their corresponding illumination measured at different locations in a scene is required to learn to estimate outdoor intrinsics and local lightings. Existing outdoor panorama datasets, such as [10, 25] only provide a single global illumination map assuming distant lighting, which cannot be used to learn local lighting estimation. To provide training data for solving “SOLID” problem, we introduce the **SOLID-Img**, a dataset for **S**patially-varying **O**utdoor **L**ighting estimation with ground truth **I**ntrinsic **D**ecomposition labels and a large amount of rendered **Im**ages as shown in Figure 2.



A Piece of City Model (top view) (a) Camera Selection (b) Scene Rendering (c) Local Lighting Collection (d) Data Filtering

Figure 2: Pipeline of data generation and filtering for creating SOLID-Img dataset with physically based rendering.

3.1. Data Generation

We adopt 3D city models from the Blender SceneCity [2] to create synthetic scenes. In Blender SceneCity, there are 450 unique objects in 80 material categories. The object models provide surface materials, including diffuse albedo, roughness, and transparency, which are used to obtain photo-realistic renderings.

Camera setting. For each road block, we select a set of cameras with diverse views seeing most objects in the context, to provide comprehensive information for lighting estimation, as shown in Figure 2(a). Our process starts by selecting the "best" camera [26] for each of the six horizontal view direction sectors in every road block. For each sector, we select the view with the highest percentage pixel coverage according to item buffer, as long as it has more than three object categories[1].

Scene rendering. We collect 70 HDR environment maps from HDRI Haven [3] which cover different solar zenith angles from sunrise to sunset. To simulate different sunlight directions, we rotate each HDR environment map along the latitude direction with a random angle sampled uniformly in $[0°, 60°]$. Then we render images using the camera settings above and these HDR environment maps with the physically-based Blender Cycles rendering engine [1], to generate photo-realistic renderings. The resolution is set as 320×240 with a physically-based path tracer of 512 samples. We record the material buffer (diffuse albedo buffer, normal buffer, depth buffer) as intermediate ground truth. We represent 3D geometry using the surface normal and plane distance, and render both as suggested in [20]. To render shadows, we set the whole scene as a single Lambertian material and render it twice with shadow turned on and off respectively, from which shadow maps are calculated by checking the difference, as shown in Figure 2(b).

Local lighting collections. To obtain the ground truth of global lighting, we save the rotated environment maps with 256×128 solution. To collect local lighting, we randomly sample 4 locations in the scene to render 4 local light probes. The image is split into 4 quadrants, and a random 2D coordinate is sampled uniformly in each quadrant (excluding the sky part and 5% pixels near the image boundary). The 3D centers of local cameras are calculated by casting a ray from the camera recording the scene to the surface of the scene and getting the first intersection point. From that point, we move the local camera center 10cm away along the plane surface normal to prevent large invalid pixels and render a local light probe at this position with 256×128 resolution. All local light probes are rendered in the equirectangular representation, as shown in Figure 2(c).

3.2. Data Filtering

Inspired by [26], we remove low-quality renderings that are with different color distribution with natural images, *e.g.*, with overly low or high intensities. To obtain a prior color distribution on real images, we compute normalized color histogram for 1100 selected real images from the Google Street View Dataset [23]. For each rendered image, we calculate the histogram similarity with those from Google Street View as the sum of minimal value of each bin; and then we assign it with a score calculated as the largest histogram similarity by comparing it to all real images; finally, we select all images with color similarity score larger than 0.7, as shown in Figure 2(d). This process selects 38000 images from the initially rendered image set, composing the SOILD-Img dataset. Then care is taken to split the dataset into the train/test set according to different lighting conditions.

4. Method

This section introduces the design methodology of SOLID-Net whose pipeline is shown in Figure 3. It is a two-stage framework that learns to reconstruct locally HDR outdoor environment maps, trained with the SOLID-Img dataset introduced in Section 3.

4.1. Problem Formulation

We formulate illumination estimation as a regression problem. Given an LDR image $\mathbf{I}$ with limited FOV and a selected pixel location "$\mathbf{l}$" in homogeneous coordinate $(x_0, y_0, 1)^\top$, our model outputs an HDR illumination $\mathbf{H}_\mathbf{l}$ centered around the 3D location of the pixel "$\mathbf{l}$" and a global

[1] More details are in the supplementary material.

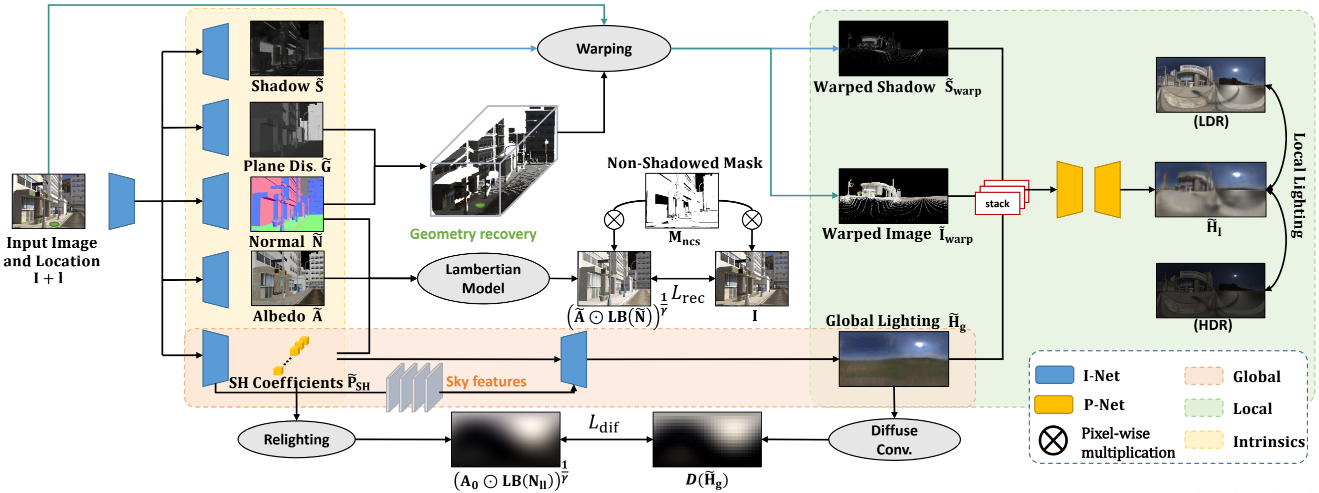


Figure 3: Pipeline of SOLID-Net. Stage 1: I-Net takes input image $\mathbf{I}$ as input and estimate the intrinsic parts: $\tilde{\mathbf{A}}$, $\tilde{\mathbf{N}}$, $\tilde{\mathbf{G}}$, $\tilde{\mathbf{S}}$ and intermediate $\tilde{\mathbf{P}}_{\mathrm{SH}}$, then $\tilde{\mathbf{P}}_{\mathrm{SH}}$ is decoded with sky features to generate $\tilde{\mathbf{H}}_{\mathrm{g}}$. The recovered geometry from $\tilde{\mathbf{N}}$ and $\tilde{\mathbf{G}}$ is used to warp $\mathbf{I}$ and $\tilde{\mathbf{S}}$ into panoramic images $\tilde{\mathbf{I}}_{\mathrm{warp}}$ and $\tilde{\mathbf{S}}_{\mathrm{warp}}$ around an input location $\mathbf{l}$. Stage 2: P-Net takes in warped images and $\tilde{\mathbf{H}}_{\mathrm{g}}$ to predict a high-frequency HDR spatially-varying lighting. The whole network is trained in an end-to-end manner.

sky environment HDR illumination $\mathbf{H}_{\mathrm{g}}$, where both $\mathbf{H}_{\mathrm{l}}$ and $\mathbf{H}_{\mathrm{g}}$ are represented as a panoramic image with full FOV.

4.2. Network Architecture

A straightforward approach to estimating the outdoor lighting from the scene would be to simply take the single limited FOV image as input, encode it into a feature map using a CNN, and feed the feature map into a lighting-regression sub-network [8, 10]. Unsurprisingly, we find it results in outdoor lighting estimation with higher error (see Figure 6), presumably because it is difficult for the network to understand how to extract full FOV lighting from a limited FOV image. One way to improve it is to bring regularization from the Lambertian rendering equation [22], which however is challenging for outdoor spatially-varying lighting estimation because: 1) Outdoor scenes have large areas of shadow occlusion which cannot be directly fitted by the Lambertian model. 2) SH lighting has a limited dynamic range and it is too smooth to represent sharp sky lighting and detailed texture. Therefore, we propose a two-stage framework to jointly solve these problems by: 1) proposing an intrinsic image decomposition network (denoted as **I-Net**) that takes a limited FOV image as input and estimates its intrinsic components as well as a global sky environment map and 2) designing a panoramic completion module (denoted as **P-Net**) that estimates local lighting from outputs in the previous stage and the input location.

I-Net. As shown by blue blocks in Fig 3, I-Net takes a single limited-FOV LDR image $\mathbf{I}$ as input and various outputs including diffuse albedo $\mathbf{A}$, surface normal $\mathbf{N}$, plane distant map $\mathbf{G}$ [20], shadow map $\mathbf{S}$, and second-order SH coefficients $\mathbf{P}_{\mathrm{SH}}$, and the sky environment map $\mathbf{H}_{\mathrm{g}}$ generated from SH coefficients and sky features. We use a single encoder to capture global features of intrinsic information, and then use five decoders for $\mathbf{A}$, $\mathbf{N}$, $\mathbf{G}$, $\mathbf{S}$, and $\mathbf{P}_{\mathrm{SH}}$ followed by a decoder lighting branch for sky environment map regression. Skip links are used for preserving details. In particular the sky map regression, we use a fully-connected (FC) layer to process the output feature maps of lighting branch encoder to generate a latent vector of size 27 (second-order SH in RGB). For the decoder, we reshape this vector and upsample it 8 times, and then combine it by flipping padded sky features to generate a 256×128 HDR sky environment map. The lighting information encoded in $\mathbf{P}_{\mathrm{SH}}$ can be considered as the low-frequency form of $\mathbf{H}_{\mathrm{g}}$, and it is used to guide the recovery of the high-frequency sky environment maps with sky features extracted from the input images. In summary, I-Net predicts intrinsic components (examples are provided in Figure 4) and sky environment map:

$$\{\tilde{\mathbf{A}}, \tilde{\mathbf{N}}, \tilde{\mathbf{G}}, \tilde{\mathbf{S}}, \tilde{\mathbf{H}}_{\mathrm{g}}(\tilde{\mathbf{P}}_{\mathrm{SH}})\} = \text{I-Net}(\mathbf{I}). \tag{1}$$

P-Net. The 3D location is calculated by the predicted normal vector $\mathbf{n}$ in $\tilde{\mathbf{N}}$ and plane distance p in $\tilde{\mathbf{G}}$ by I-Net for each pixel. If the camera intrinsic matrix is fixed as $\mathbf{K} = [f_x, 0, c_x; 0, f_y, c_y; 0, 0, 1]$ and 2D pixel locations $(x_i, y_i, 1)^\top$ of the whole image are provided, we can reproject them as a 3D scene $\mathbf{P} = (x, y, z)^\top$ by $\mathbf{P} = -\frac{p}{\mathbf{v}\cdot\mathbf{n}}\mathbf{v}$, where $\mathbf{v} = (\frac{x_i - c_x}{f_x}, \frac{y_i - c_y}{f_y}, 1)^\top$.

By using $\mathbf{P}$ we warp the input image $\mathbf{I}$ and estimated shadow map $\tilde{\mathbf{S}}$ and spatially align them with the output local lighting to provide panoramic observation. First, we

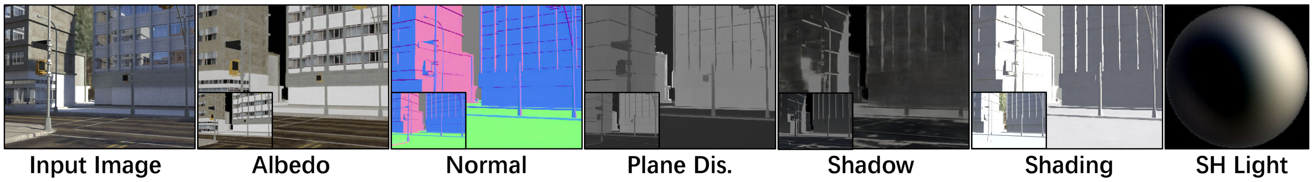


Figure 4: An example of intrinsic decomposition results using our SOLID-Img test dataset. Given an input image, our estimated albedo, normal, plane distance, shadow, and shading show close appearance to the ground truth (shown as insets).

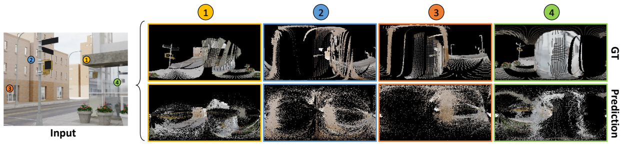


Figure 5: An example of panoramic warping. By using the estimated geometry-related intrinsics, we warp the observed image into panorama coordinates according to the input pixel location. (Please zoom-in for details.)

compute the camera location according to the input point position $\mathbf{l}$ and apply 10cm translation (defined in Section 3) along the normal direction of the supporting plane to align with training data. Second, we perform a panoramic warping through a forward projection using the estimated geometry and camera location to map pixels in $\mathbf{I}$ and $\tilde{\mathbf{S}}$ as panoramic images (an example is provided in Figure 5). The Z-buffer is computed to discard invisible points and the points without projected positions are set to 0.

Since the local lightings share the same camera rotation, the sky parts in local lighting should be consistent, this motivates us to take the sky as an input to P-Net. As shown by orange blocks in Fig 3, P-Net concatenates the two incomplete panoramic image $\tilde{\mathbf{I}}_{\text{warp}}$ and $\tilde{\mathbf{S}}_{\text{warp}}$ and the global lighting estimated by I-Net $\tilde{\mathbf{H}}_{\text{g}}$ as inputs and outputs a dense pixel-wise prediction of local lighting panorama with full FOV and high-frequency details, as

$$\tilde{\mathbf{H}}_{\text{l}} = \text{P-Net}(\tilde{\mathbf{I}}_{\text{warp}}(\mathbf{l}), \tilde{\mathbf{S}}_{\text{warp}}(\mathbf{l}), \tilde{\mathbf{H}}_{\text{g}}). \quad (2)$$

P-Net is implemented as a fully convolutional U-Net [16].

4.3. Loss Functions

Direct supervision loss. Direct supervision $\mathcal{L}_{\text{sup1}}$ for I-Net is provided to 1) diffuse albedo predictions via L_2 loss, 2) shadow predictions via L_2 loss, 3) surface normal predictions via cosine loss, 4) plane distance map predictions via L_1 loss, and 5) sky environment map predictions via L_1 loss. Then direct supervision $\mathcal{L}_{\text{sup2}}$ for P-Net is provided to local lighting predictions via L_1 loss.

$$\begin{aligned} \mathcal{L}_{\text{sup1}} &= \|\tilde{\mathbf{A}} - \mathbf{A}\|_2 + \|1 - \tilde{\mathbf{N}} \cdot \mathbf{N}\|_2 + \|\tilde{\mathbf{G}} - \mathbf{G}\|_1 \\ &\quad + \|\tilde{\mathbf{S}} - \mathbf{S}\|_2 + \|\tilde{\mathbf{H}}_{\text{g}} - \mathbf{H}_{\text{g}}\|_1, \quad (3) \\ \mathcal{L}_{\text{sup2}} &= \|\tilde{\mathbf{H}}_{\text{l}} - \mathbf{H}_{\text{l}}\|_1. \quad (4) \end{aligned}$$

where the $\tilde{*}$ means the estimations of I-Net and $\cdot$ is the dot product for each vector in a matrix.

Diffuse convolution loss. In order to guide the sky environment map estimated by I-Net to extract low-frequency lighting information from the encoded SH coefficients, we add a diffuse convolution loss $\mathcal{L}_{\text{dif}}$ to force $\mathbf{H}_{\text{g}}$ applied with the diffuse convolution to have a close appearance with a relighted pure Lambertian surface from $\mathbf{P}_{\text{SH}}$:

$$\mathcal{L}_{\text{dif}} = \frac{1}{N}\sum_{i=1}^{N}[D(\mathbf{H}_{\text{g}}, i) - \text{diag}(\alpha_{\mathbf{o}})\mathbf{L}\mathbf{b}(\mathbf{N}_{\mathbf{ll}}(i))]^2, \quad (5)$$

where $\alpha_{\mathbf{o}} = [1, 1, 1]^\top$ is the global diffuse albedo, $\mathbf{L} \in \mathbb{R}^{3\times 9}$ is the SH coefficients by reshaping $\mathbf{P}_{\text{SH}} \in \mathbb{R}^{1\times 27}$, $\mathbf{N}_{\mathbf{ll}}$ is the normal map of a sphere in panorama coordination and the second order SH basis is given by: $\mathbf{b}(\mathbf{n}) = [1, n_x, n_y, n_z, 3n_z^2 - 1, n_x n_y, n_x n_z, n_y n_z, n_x^2 - n_y^2]^\top$. D is the diffuse convolution function defined as

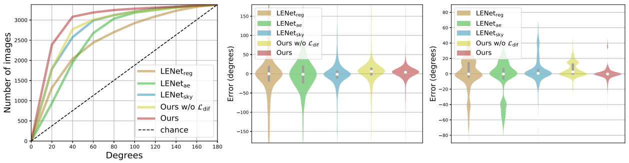
$$D(\mathbf{H}, i) = \frac{1}{K_i}\sum_{\omega\in\Omega_i}\mathbf{H}(\omega)s(\omega)(\omega \cdot \mathbf{n}), \quad (6)$$

where Ω_i is the hemisphere centered at pixel i on the global lighting environment map, $\mathbf{n}$ is the normal vector at pixel i, and K_i is the sum of solid angles on Ω_i. ω is a unit vector of direction and $s(\omega)$ is the solid angle for a pixel in the panorama map of direction ω with different scale factors (because pixels in the panorama map at different latitudes correspond to projections on the unit sphere with different area sizes).

Inverse rendering reconstruction loss. To make the network learn constraints from physically-based image formation model, we put SH coefficients as an intermediate variable and provide indirect supervision to $\mathbf{P}_{\text{SH}}$ via an inverse rendering reconstruction loss $\mathcal{L}_{\text{rec}}$ on the directly illuminated part, by multiplying a non-shadowed mask to disregard the effect of shadows:

$$\mathcal{L}_{\text{rec}} = ||\mathbf{M}_{\mathbf{ncs}} \odot (\mathbf{I}_{\text{im}} - (\tilde{\mathbf{A}} \odot \mathbf{L}\mathbf{B}(\tilde{\mathbf{N}}))^{1/\gamma})||_2, \quad (7)$$

where $\odot$ represents the element-wise product. We use a fixed $\gamma = 2.2$ to compress the dynamic range. $\mathbf{M}_{\mathbf{ncs}}$ is the non-shadowed mask computed using shadow maps from intrinsics; a binary Otsu segmentation on histogram of shadow maps is further used to eliminate weak interreflections; $\mathbf{I}_{\text{im}} \in \mathbb{R}^{3\times K}$ is the RGB image matrix; $\tilde{\mathbf{A}} \in \mathbb{R}^{3\times K}$



(a) Sun angular error (b) Azimuth error (c) Elevation error

Figure 6: Quantitative evaluation of sun position estimation. (a) The cumulative sun angular error comparison between baseline methods and ours. The estimation error of sun azimuth (b) and elevation angles (c) is displayed as a "violin plot" where the envelope of each bin represents the percentile, the gray line represents the percentile of 25% to 75%, and the median is shown as a white point.

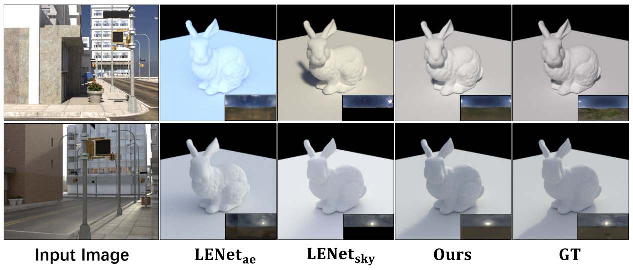


Figure 7: Relighting results with global lighting (shown as insets) on our SOLID-Img dataset.

is the estimated diffuse albedo matrix; $\mathbf{B}(\tilde{\mathbf{N}}) \in \mathbb{R}^{9\times K}$ is a matrix stacked by $\mathbf{b}(\mathbf{n})$ which applied SH basis on the normal map.

Tonemapped SSIM loss. A structural similarity index measure (SSIM) loss between dynamic range compressed images with a fixed gamma parameter (2.2 in our experiment) is used to recover structure similarity between the estimation and ground truth:

$$\mathcal{L}_{\text{tom}} = \|\Im((2^e \cdot \tilde{\mathbf{H}}_{\text{l}})^{1/\gamma}) - \Im((2^e \cdot \mathbf{H}_{\text{l}})^{1/\gamma})\|_{\text{SSIM}},$$

$$\Im(\mathbf{H}) = \begin{cases} 1 & \mathbf{H} > 1 \\ 0 & \mathbf{H} < 0 \\ \mathbf{H} & 0 \leq \mathbf{H} \leq 1 \end{cases} \tag{8}$$

where e is the exposure intensity fixed as -0.3 in our experiments.

I-Net is trained by summing up direct supervision loss, diffuse convolution loss, and inverse rendering reconstruction loss as: $\mathcal{L}_{\text{I}} = \mathcal{L}_{\text{sup1}} + \mathcal{L}_{\text{diff}} + \mathcal{L}_{\text{rec}}$, and then P-Net is trained by summing up direct supervision loss and tonemapped SSIM loss as: $\mathcal{L}_{\text{P}} = \mathcal{L}_{\text{sup2}} + \mathcal{L}_{\text{tom}}$.

5. Experiments

We perform detailed network analysis and present qualitative and quantitative results on our SOLID-Img test set. We also capture a small set of real LDR outdoor local environment maps to analyse the generalization of our method. Finally, we show relighted bunny results to validate our methods qualitatively[2]. To measure the accuracy of our predicted global sky environment map $\mathbf{H}_{\text{g}}$ and local illumination maps $\mathbf{H}_{\text{l}}$, we use mean absolute error (MAE) on the HDR sky environment map, angular error on the sun position and sun azimuth/elevation angles, and SSIM on the detailed local lighting as error metrics.

5.1. Analysis using Synthetic Dataset

Effectiveness of I-Net. To validate the design of intrinsic decomposition, we compare our global lighting estimation branch with three baseline models for the accuracy of estimated sun positions: 1) $\mathbf{LENet_{reg}}$ is a regression-based model that directly regresses the global sky from the input image. 2) $\mathbf{LENet_{ae}}$ is a two-stream convolution network used to regress sun azimuth angle and normalized HDR panorama from an LDR panorama [25]; we modify the input as a single limited-FOV image to adapt our task. 3) $\mathbf{LENet_{sky}}$ learns to estimate both the sun azimuth angle and a non-parametric sky [10]. In particular, $\mathbf{LENet_{ae}}$ learns azimuth estimation as a regression task, while $\mathbf{LENet_{sky}}$ treats it as a classification problem. All baseline models are retrained using SOLID-Img training dataset with the same setting[3]. Since our global lighting is represented by a non-parametric sky environment map, we compute the sun position by finding the largest connected component of the sky above a threshold (98%) and computing its centroid. And then we rotate estimated sky environment maps around their azimuth angles to make sure the sun is in the center of the image so that we can compare it with their baseline models.

From Figure 6, we can see that our method shows significant improvement than $\mathbf{LENet_{reg}}$ and $\mathbf{LENet_{ae}}$ and comparable improvement than $\mathbf{LENet_{sky}}$, thanks to the intrinsic cues. Qualitative results on the test dataset are shown in Figure 7[4]. Our relighting results and estimated lightings show a closer appearance to the ground truth (shown as insets) than other methods.

To help understanding how SH coefficients decode the global lighting information, we perform the Grad-CAM [17] on our global lighting encoder. We use the maximum response value of SH coefficients as the target backward label to find which regions of input are important for global lighting prediction. From Figure 8, the feature heatmaps validate that I-Net mostly captures directly illuminated information to estimate global lighting.

Effectiveness of P-Net. We train our P-Net with combinations of different inputs: warped incomplete LDR image

[2] More results are in the supplementary material.

[3] Detailed model structures are in the supplementary material.

[4] Numerical results and MAE errors on estimated sky environment map are provided in the supplementary material.

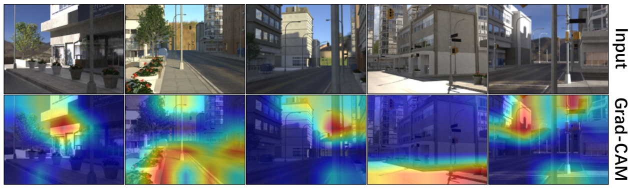


Figure 8: Visualization of Grad-CAM [17] on our SH lighting prediction using SOLID-Img test set.

Table 1: Ablation study about our multi-input module.

Inputs	SSIM	MAE
W_{col}	0.689	0.848
W_{col} + W_{sha}	0.736	0.730
W_{col} + W_{SH}	0.787	0.618
W_{col} + W_{sky}	0.793	0.531
W_{col} + W_{sha}+ W_{sky}	**0.803**	**0.523**

panorama W_{col}, relighted Lambertian surface W_{SH}, estimated sky environment map W_{sky}, and incomplete shadow panorama image W_{sha}. During training, we only process direct supervision on local lighting. We evaluate the SSIM and MAE errors between the estimated local lighting and ground truth. From Table 1, we can tell that directly providing W_{sky} rather than W_{SH} improves our algorithm marginally, while also providing W_{sha} improves it a bit more. We conjecture it is because shadows provide occlusion information which is helpful for lighting estimation. In Figure 9, we show results without global lighting, with W_{SH} as global lighting, and with W_{sky} as global lighting, respectively. We find that P-Net is incapable to learn the correct sun position only with the warped color image but can recover it accurately by adding W_{SH} as shown in the first column and third column. Although the sun position is well recovered with W_{SH}, the sun intensity still has a large gap from the real condition.

For an off-the-shelf renderer (*e.g.*, Blender), we can achieve multi-object rendering by setting it to render only the object in the selected lighting position, and then blending this result with rendering results from other positions through the alpha channel. In Figure 10, we show the visual quality of synthetic object insertion to better illustrate the usefulness of spatially-varying outdoor lighting estimation. As can be observed, our method can render correct lighting effects (specular highlights and shadows) on rabbits under different materials.

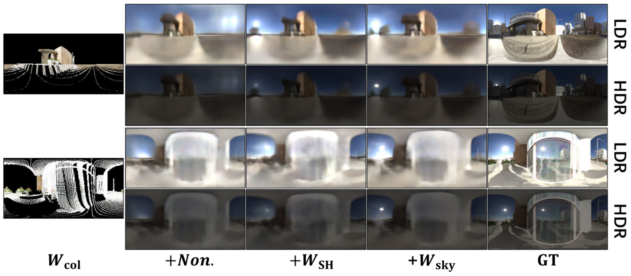


Figure 9: Estimated local lighting with different inputs.

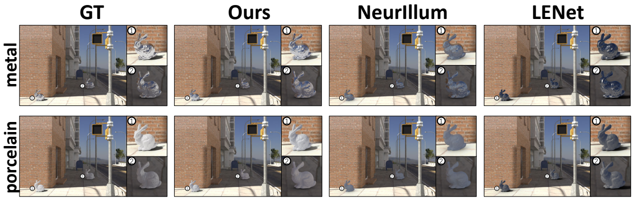


Figure 10: Synthetic examples of inserting virtual objects of different materials, compared with NeurIllum [18] and LENet [10].

Effects of different losses. To verify the necessity of each loss function, we evaluate the performance of I-Net and P-Net using different combinations of loss functions. In Figure 6, we can observe that our models have comparable improvement to $\mathbf{LENet_{sky}}$ even without $\mathcal{L}_{\text{dif}}$ due to constraint from intrinsics provided by I-Net. By further adding $\mathcal{L}_{\text{dif}}$, I-Net can learn the global sky environment map more effectively with the guidance of SH encoded lighting and then produce a more accurate sky estimation than $\mathbf{LENet_{sky}}$. If $\mathcal{L}_{\text{tom}}$ is ablated, the performance on test dataset becomes 0.760 / 0.564 (SSIM / MAE); by adding this loss, the numbers are **0.798 / 0.552** (SSIM / MAE), which shows that P-Net can predict local lighting more accurately especially on structure similarity.

5.2. Evaluation on Real Dataset

Real data capture. To validate that SOLID-Net is able to perform outdoor local lighting estimation, we capture real outdoor city street view scenes and the corresponding spatially-varying local environment maps (see Figure 11). The images are captured by a Ricoh Theta SC2 camera with dual fisheye lens. For local lighting environment maps, the scenes are captured 1/2500s shutter speed with $f2.0$ aperture by placing the panoramic camera as a light probe at different locations. Due to the limited dynamic range of our panoramic camera, the local environment maps are not able to faithfully record the intensity of sunlight. To obtain the accurate sun position for evaluation, we further capture a low-exposure panorama with 1/25000s shutter speed and label the sun position manually. The captured LDR local lighting is aligned to its view vector with respect to the camera facing direction. In total, our real test dataset includes 29 outdoor scenes and 67 LDR local lighting environment maps for evaluating our method quantitatively.

Comparison with previous work. We first compare the accuracy of global lighting estimation with $\mathbf{LENet_{sky}}$ model [10] using sun position errors. The azimuth/elevation angular errors of $\mathbf{LENet_{sky}}$ are $37.5°/9.5°$. In contrast,

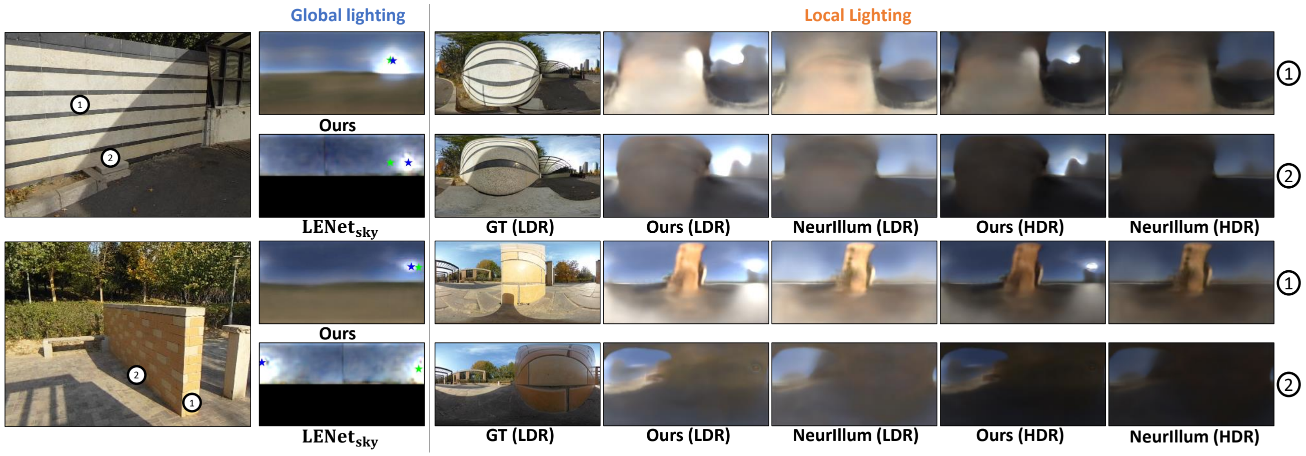


Figure 11: Comparison of estimated global and local lighting on our real test dataset. Column 1 shows the input image and selected pixel locations. Column 2 shows the estimated global lighting of DeepSky model [10] and our method (blue star marks the estimated sun position by computing the centroid of largest connected component, while the green star marks the ground truth sun position labeled from a low-exposure environment map by us manually). Column 3 shows the LDR local lighting environment map. Columns 4−7 show the estimated local lighting by NeurIllum [18] and our methods in both LDR and HDR formats.

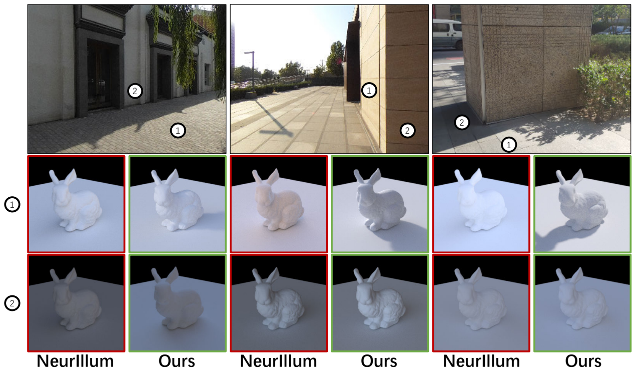


Figure 12: Qualitative comparison of relighting results using our real dataset.

our method maintains a high accuracy with $\mathbf{28.9^\circ/6.8^\circ}$. From Column 2 of Figure 11, we can see that our method can generate a clearer environment map with different sky conditions and our estimated sun positions are closer to the ground truth than $\mathbf{LENet_{sky}}$. To evaluate the estimated local lighting, we compare our method with NeurIllum [18] retrained on our synthetic dataset on estimated spatially-varying lighting quantitatively and qualitatively. Overall, our method achieves a better SSIM / MAE (the higher is better / the lower is better) performance of **0.235 / 0.203**, compared to 0.228 / 0.233 for NeurIllum. Compared estimations of our method (Column 4-5) and NeurIllum (Column 6-7) with ground truth (Column 3) in Figure 11, we note that their method does not capture the accurate sun position and intensity, caused by missing panoramic information which our method handles well. We also show relighted bunny results to further compare estimated spatially-varying lighting effects of our method and NeurIllum (see Figure 12). These

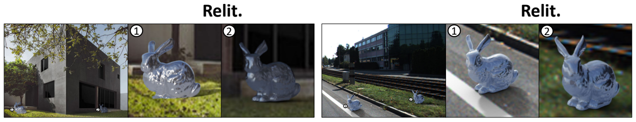


Figure 13: Real examples of virtual object insertion.

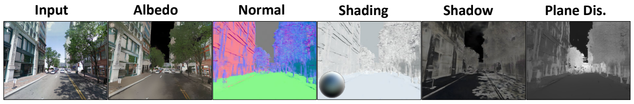


Figure 14: Real examples of intrinsic decomposition.

show that our approach adapts to strongly spatially-varying local lighting effects in real scenes.

6. Discussion

We present the first end-to-end outdoor spatially-varying lighting estimation framework and demonstrate it significantly outperforms previous works via extensive evaluations on both synthetic and real datasets. Our method is able to generalize on real scenes with a slightly different appearance from our synthetic scene. An example is shown in Figure 13, in which the virtual object is reasonably relit in a scene of rarely seen structures (with railway and glass) in the synthetic training data.

Limitations and future work. Due to the material diversity gap between synthetic and real data, the intrinsic decomposition results on real data may not be as accurate as those on synthetic data (Figure 14 compared with Figure 4). Although SOLID-Net estimates HDR lighting environment map to support realistic relighting effects, our lighting model is not suitable for generating animations that are sensitive at harsh lighting boundaries, which will be an interesting direction for future work.

References

[1] Blender. https://www.blender.org. 3

[2] Blender SceneCity. https://www.cgchan.com/store/scenecity. 2, 3

[3] HDRI HAVEN. https://hdrihaven.com. 3

[4] Chang Angel, 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. 2

[5] Jonathan T. Barron and Jitendra Malik. Intrinsic Scene Properties from a Single RGB-D Image. In *Proc. of Computer Vision and Pattern Recognition*, 2013. 1, 2

[6] Dachuan Cheng, Xiaoming Deng Jian Shi, Yanyun Chen, and Xiaopeng Zhang. Learning Scene Illumination by Pairwise Photos from Rear and Front Mobile Cameras. *Computer Graphics Forum*, 37:213–221, 2018. 2

[7] Paul Debevec. Rendering Synthetic Objects into Real Scenes: Bridging Traditional and Image-based Graphics with Global Illumination and High Dynamic Range Photography. In *Proc. of ACM SIGGRAPH*, 1998. 2

[8] 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. In *Proc. of ACM SIGGRAPH Asia*, 2017. 4

[9] Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. Fast Spatially-Varying Indoor Lighting Estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 1, 2

[10] Yannick Hold-Geoffroy, Akshaya Athawale, and Jean-François Lalonde. Deep Sky Modeling for Single Image Outdoor Lighting Estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 1, 2, 4, 6, 7, 8

[11] Yannick Hold-Geoffroy, Emiliano Gambaretto Kalyan Sunkavalli, Sunil Hadap, and Jean-François Lalonde. Deep Outdoor Illumination Estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2017. 1, 2

[12] Alexei A. Efros Jean-François Lalonde and Srinivasa G. Narasimhan. Estimating the Natural Illumination Conditions from a Single Outdoor Image. *International Journal of Computer Vision*, 98(2):123–145, 2012. 2

[13] JF Lalonde, LP Asselin, J Becirovski, Y Hold-Geoffroy, M Garon, MA Gardner, and J Zhang. The Laval HDR sky database, 2016. http://sky.hdrdb.com. 2

[14] Chloe LeGendre, Graham Fyffe Wan-Chun Ma, John Flynn Laurent Charbonnel, Jay Busch, and Paul Debevec. DeepLight: Learning Illumination for Unconstrained Mobile Mixed Reality. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 2

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

[16] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation. In *Medical Image Computing and Computer-Assisted Intervention*, 2015. 5

[17] Ramprasaath R. Selvaraju, Abhishek Das Michael Cogswell, Devi Parikh Ramakrishna Vedantam, and Dhruv Batra. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. In *Proc. of International Conference on Computer Vision*, 2017. 6, 7

[18] Shuran Song and Thomas Funkhouser. Neural Illumination: Lighting Prediction for Indoor Environments. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 1, 2, 7, 8

[19] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic Scene Completion from a Single Depth Image. In *Proc. of Computer Vision and Pattern Recognition*, 2017. 2

[20] Shuran Song, Andy Zeng, Angel X Chang, Manolis Savva, Silvio Savarese, and Thomas Funkhouser. Im2Pano3D: Extrapolating 360 Structure and Semantics Beyond the Field of View. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 3, 4

[21] Jessi Stumpfel, Andreas Wenger Andrew Jones, Tim Hawkins Chris Tchou, and Paul Debevec. Direct HDR Capture of the Sun and Sky. In *Proc. of ACM SIGGRAPH*, 2004. 2

[22] Ye Yu and William AP Smith. InverseRenderNet: Learning single image inverse rendering. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 4

[23] Amir Roshan Zamir and Mubarak Shah. Image Geo-localization Based on Multiple Nearest Neighbor Feature Matching using Generalized Graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PP(99):1–1, 2014. 3

[24] Jinsong Zhang, Yannick Hold-Geoffroy Kalyan Sunkavalli, Jonathan Eisenmann Sunil Hadap, and Jean-François Lalonde. All-Weather Deep Outdoor Lighting Estimation. In *Proc. of Computer Vision and Pattern Recognition*, 2019. 1, 2

[25] Jinsong Zhang and Jean-François Lalonde. Learning High Dynamic Range from Outdoor Panoramas. In *Proc. of International Conference on Computer Vision*, 2017. 2, 6

[26] 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. In *Proc. of Computer Vision and Pattern Recognition*, 2017. 3