NIID-Net: Adapting Surface Normal Knowledge for Intrinsic Image Decomposition in Indoor Scenes

Jundan Luo*, Zhaoyang Huang*, Yijin Li,
Xiaowei Zhou, *Member, IEEE*, Guofeng Zhang, *Member, IEEE*, and Hujun Bao, *Member, IEEE*

(a) Original image (b) Copy & paste (c) Ours

(d) Target reflectance (e) Li and Snavely [34] (f) GLoSH [56]

Fig. 1: Surface re-coloring comparison. (b): the composition result loses lighting effects when the new reflectance is copied and pasted to the original image naively. (c)(e)(f): the surfaces re-rendered with estimated shading are more realistic but still contain some artifacts. For example, the result based on GLoSH [56] is contaminated by residual textures in the predicted shading (circled in blue), and it lacks distinct shadows (pointed by red arrows). The result based on Li and Snavely (CGI+IIW+SAW) [34] has few lighting effects. By contrast, our result achieves the best realism with fewer residual textures and more faithful lighting effects.

Abstract—Intrinsic image decomposition, *i.e.*, decomposing a natural image into a reflectance image and a shading image, is used in many augmented reality applications for achieving better visual coherence between virtual contents and real scenes. The main challenge is that the decomposition is ill-posed, especially in indoor scenes where lighting conditions are complicated, while real training data is inadequate. To solve this challenge, we propose NIID-Net, a novel learning-based framework that adapts surface normal knowledge for improving the decomposition. The knowledge learned from relatively more abundant data for surface normal estimation is integrated into intrinsic image decomposition in two novel ways. First, normal feature adapters are proposed to incorporate scene geometry features when decomposing the image. Secondly, a map of integrated lighting is proposed for propagating object contour and planarity information during shading rendering. Furthermore, this map is capable of representing spatially-varying lighting conditions indoors. Experiments show that NIID-Net achieves competitive performance in reflectance estimation and outperforms all previous methods in shading estimation quantitatively and qualitatively. The source code of our implementation is released at `https://github.com/zju3dv/NIID-Net`.

Index Terms—Intrinsic image decomposition, image processing, augmented reality

1 Introduction

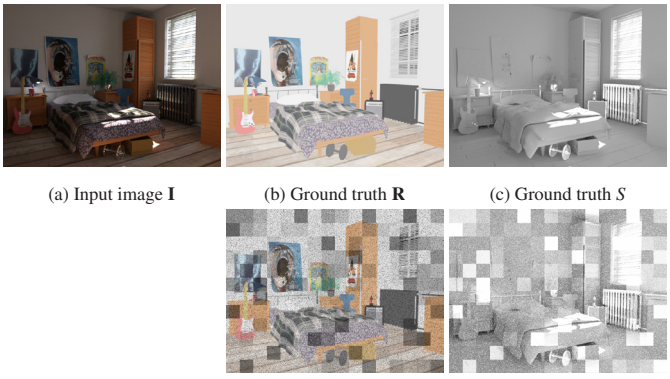
Intrinsic image decomposition, which was first introduced by Barrow and Tenenbaum [3], aims to decode a photo of a scene into multiple underlying physical characteristics, such as surface orientation, materials, and environment illumination. Many works, including ours, have focused on Lambertian scenes, which simplifies the decomposition of an input photo $\mathbf{I}$ into estimating a reflectance image $\mathbf{R}$ and a shading image $\mathbf{S}$:

$$\mathbf{I} = \mathbf{R} \times \mathbf{S},$$

where $\times$ is channel-wise multiplication. Reflectance, also known as albedo, describes the ability of a material to reflect incident lights. For Lambertian materials, reflectance is invariant to viewpoint, lighting conditions and scene geometry. Shading, the result of interaction between surface orientation and lighting conditions, consists of varied

- **J. Luo and Z. Huang assert equal contributions.*
- *J. Luo, Z. Huang, Y. Li, X. Zhou, G. Zhang and H. Bao are with the State Key Lab of CAD&CG, Zhejiang University, and also affiliated with ZJU-SenseTime Joint Lab of 3D Vision. E-mails: {luojundan, drinkingcoder, eugenelee, xwzhou}@zju.edu.cn, {zhangguofeng, bao}@cad.zju.edu.cn.*
- *H. Bao is the corresponding author.*

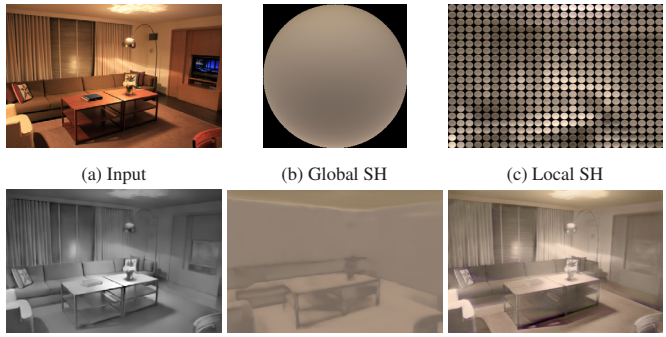
Manuscript received 18 May 2020; revised 26 July 2020; accepted 17 Aug. 2020. Date of publication 17 Sept. 2020; date of current version 3 Nov. 2020. Digital Object Identifier no. 10.1109/TVCG.2020.3023565



(a) Input image **I** (b) Ground truth **R** (c) Ground truth S

(d) Constructed **R** (e) Constructed S

Fig. 2: Two solutions to the decomposition of the same input image. (b) and (c) are ground-truth reflectance and shading taken from the CGI [34] dataset. (d) and (e) are another pair that is manually constructed by us but does not exist in reality.



(a) Input (b) Global SH (c) Local SH

(d) S (ours) (e) **S** (global SH) (f) **S** (local SH)

Fig. 3: Visual comparison among shading images rendered by a global lighting model or multiple local lighting models. (b) and (c) are global lighting and pixel-wise local lighting predicted by GLoSH [56]. (e) and (f) are shading images rendered by (b) and (c), respectively. Compared with (f), (e) lacks common local lighting effects in indoor environments, such as shadows and highlights. (d) is the shading image predicted by our proposed NIID-Net which also considers spatially-varying lighting conditions.

lighting effects, such as shadows, highlights and inter-reflections.

Intrinsic images are used in the augmented reality systems proposed by Meka *et al.* [36, 37] for photorealistic scene editing. As explained in Fig. 1, by re-rendering with estimated shading, the virtual surfaces visually have more coherent illumination with the nearby real objects. Seamlessly blending virtual components within the real world is an effective way to enhance immersion in an augmented reality environment. Besides, synthesizing an image from a pair of intrinsic images is simply a three-channel multiplication, which benefits real-time interaction with virtual components. In addition to image editing [8, 11, 12], intrinsic images are also used for illumination-invariant image transformation [45], shape from shading [26, 51, 53, 59] and depth refinement [52].

Intrinsic image decomposition is an ill-posed problem. There are infinite pairs of reflectance and shading that can reconstruct the same input. A sample input and two solutions to the decomposition equation are illustrated in Fig. 2. In order to distinguish between reflectance and shading, physical constraints or statistical priors are adopted. Several methods [16, 26, 53] incorporate additional depth/surface normal input data, because surface normals, which provide scene geometry information, are an essential component of shading formulation. Inspired by these methods, we propose to exploit learned surface normal knowledge for intrinsic image decomposition. The normal estimation module is singled out during its training process, and we propose two novel ways to integrate the pre-trained module back into the whole framework. To the best of our knowledge, there are no methods or devices to annotate dense intrinsic image labels in real-world indoor environments, while synthetic training data is limited and inevitably biased compared with real-world photos. By contrast, dense depth/surface normal labels are much easier to collect in real general scenes, owing to the development of high-quality commercial depth sensors (*e.g.*, Microsoft Kinect, Matterport, and FARO Focus S350). For this reason, intrinsic image decomposition can be improved by making use of available surface normal datasets.

We propose NIID-Net, a deep learning framework that first estimates surface normals and then decomposes the single input image into intrinsic images with adapted surface normal knowledge. NIID-Net consists of a normal estimation module (NEM) and an intrinsic image decomposition subnet (IID-Net). The two framework components utilize different available datasets that provide different labels. The NEM is pre-trained on the real-world NYUv2 [48] and DIODE [50] datasets with only annotations of surface normals. And the IID-Net is trained on the synthetic CGI [34] dataset with only ground-truth intrinsic images. We propose normal feature adapters (NFAs) to propagate scene geometry features from the NEM encoder into the IID-Net. These geometry priors support the IID-Net to learn complicated indoor lighting conditions in a normal-dependent way.

Besides, we propose to predict integrated lighting, which is designed for propagating geometry contour and planarity information during shading reconstruction. In terms of the Lambert's law [31], an integrated lighting vector is represented by a three-dimensional vector. A map of pixel-wise integrated lighting vectors is able to encode spatially-varying lighting conditions. In real scenes, indoor lighting conditions are usually so complicated that the spatial variations can not be ignored. As illustrated in Fig. 3, a single global spherical harmonics (SH) lighting model can not model local lighting effects (*e.g.*, shadows and highlights), while the GLoSH [56] lighting model achieves much better by adopting pixel-wise local SH models. Our motivation for predicting a map of integrated lighting is similar to GLoSH, and we also model changing lighting conditions by pixel-wise representation. However, the complexity of GLoSH is higher as each SH model has nine coefficients for each color channel. And during training, mass sampling of lighting on spheres is required for enforcing non-negative lighting intensity. By contrast, each integrated lighting vector has only three coefficients, and the vector is naturally valid in both positive and negative ranges. Therefore, integrated lighting prediction is more straightforward to constrain and train.

Summarily, our main contributions are as follows:

- We propose NIID-Net, a unified framework that jointly estimates surface normals, reflectance and shading. In this framework, scene geometry features learned from available normal datasets are integrated by proposed normal feature adapters. The IID-Net understands complicated indoor lighting conditions better with the assistance of these features.
- We propose to reconstruct shading images from predicted surface normals and predicted integrated lighting. The proposed map of integrated lighting is light-weight and capable of representing spatially-varying lighting conditions. Besides, the map is exploited to convey scene geometry information during rendering while mitigating artifacts in estimated surface normal maps.
- In terms of both visual effects and numerical accuracy, our NIID-Net achieves state-of-the-art performance in reflectance estimation and significantly outperforms all previous methods in shading estimation.

2 Related Works

Classical intrinsic image decomposition. The methods based on optimization incorporate empirical assumptions and priors into energy functions, such as the monochromatic (gray-scale) shading assumption [9, 10, 19, 20, 55], Retinex [19, 20, 55] (*i.e.*, sharp and smooth

intensity variations are respectively caused by reflectance and shading), and the reflectance sparsity assumption [9, 10, 19]. However, these hand-crafted constraints are not sufficient to reconstruct the complex real world and are sometimes violated in specific cases, such as colorful indoor lighting, hard shadows, and occlusions.

Deep intrinsic image decomposition. Data-driven approaches [5, 17, 18, 34, 39, 46], which have drawn lots of attention recently, learn priors automatically from training data via deep learning. Annotating shading or reflectance in the real world is too hard, which results in a lack of training data for learning-based methods (especially scene-level decomposition algorithms). IIW [9] is the most widely used real-world dataset, providing crowdsourced sparse annotations of relative comparisons of reflectance intensity. Given this type of labels, early neural networks are not trained to predict images of pixel-wise reflectance. Instead, they predict special priors, *e.g.*, relative reflectance ordering [58, 60] or lightness differences between pixels [40], both of which can be incorporated into later optimization steps to estimate dense reflectance. Nestmeyer and Gehler [41] supervise their network by a hinge loss, and their model is the first to directly predict reflectance images. Since then, several works [18, 34, 56] have also successfully trained deep neural networks on the IIW dataset. Even so, learning-based methods are still hindered by the sparse reflectance annotations and no shading annotations.

Synthetic datasets [15, 32, 34] are proposed to provide dense ground-truth reflectance and shading. However, scenes from the MPI Sintel dataset [15], which is not rendered photo-realistically, do not generalize well to the real world. Even for photorealistic rendering, photometric consistency between a synthetic image and its equivalent real-world image needs verification [7]. At present, the CGI dataset proposed by Li and Snavely [34] achieves relatively best generalization, but it is still biased to real scenes. Therefore, the network proposed by Li and Snavely [34] requires training on the real-world SAW [29] and IIW datasets for better performance. Another problem for synthetic datasets is that the rendered scenes are simple and too clean.

Moreover, some methods seek additional information to constrain the decomposition, such as multiple time-lapse images [35], semantic segmentation [4], depth [27], and edge detection [18].

Geometry-based intrinsic image decomposition. Several methods incorporate geometry information (*i.e.*, depth and surface normals) to distinguish between shading and reflectance. We classify these methods into two categories: geometry-similarity-based methods and lighting-model-based methods. Geometry-similarity-based methods [16] adopt the shading smoothness assumption and measure shading similarity among pixels according to their depth/surface normal similarity. Lighting-model-based methods establish relationships between surface normals and shading via a specific lighting representation. For object- or face-level intrinsic image decomposition tasks, a single global representation of lighting is commonly adopted, such as the global SH lighting model [2, 44, 51] and the single light source representation [24], because lighting conditions in their environments are usually simple and almost globally consistent. The scene-level decomposition algorithms [26,53,54] that use a single global SH model can not do well in indoor scenes, especially when there are shadows, highlights, occlusions, and inter-reflections. To simulate typical spatially-varying lighting indoors, Jeon *et al.* [25] assume that the direction of the incident light at each pixel is different, but they assume that lighting intensity is the same everywhere, which results in low-contrast shading images. Also, pixel-wise SH lighting models [1, 56] are proposed, but the total number of coefficients for this lighting representation is relatively large.

Image editing applications. The images that are edited on intrinsic components can be more realistic, as real shading effects are preserved. Several works [10–12, 36, 37] implement the application of recoloring textures, which is commonly adopted as a visual judgment for shading estimation. Beigpour *et al.* [8] apply band-sifting [14] operations on intrinsic images to manipulate novel material effects perceptually. Garces *et al.* [19] relight an image by editing its shading layer, and Bousseau *et al.* [13] convert a photo from day to night. Intrinsic images are also used for face editing. For example, Li *et al.* [33] propose makeup simulation and transfer. Shu *et al.* [47] use intrinsic images as a latent face representation for manipulating faces.

3 Proposed Method

3.1 Image Formation

Our method allows colorful lighting, so the chromaticity of the reflectance $\mathbf{R}_i$, shading $\mathbf{S}_i$, and input image $\mathbf{I}_i$ at pixel i are different. We assume that each pixel of a shading image $\mathbf{S}$ has the same chromaticity $\mathbf{c}$. Under this assumption, an input image is reconstructed by:

$$\mathbf{I}_i = \mathbf{S}_i \times \mathbf{R}_i = \mathbf{c} \times S_i \times \mathbf{R}_i, \tag{1}$$

where $\mathbf{I}_i$, $\mathbf{R}_i$ and $\mathbf{c}$ are represented in the three-channel RGB space, while S_i is a single-channel gray-scale intensity of the shading. $\times$ is channel-wise multiplication. Equation 1 only holds in the linear RGB space. In this paper, images in the equations are described in the linear RGB space, but we simulate a nonlinear gamma transformation to convert intrinsic images into the sRGB space when visualizing them:

$$\mathrm{sRGB}(\mathbf{R}_i) = \mathbf{R}_i^{\frac{1}{\gamma}},\ \gamma = 2.2.$$

3.2 Integrated Lighting

In order to make use of geometry cues, we do not directly predict shading images, but render them by the predicted surface normals and integrated lighting. As with many previous works, we only handle Lambertian surfaces. In terms of the Lambert's law [31], the resulting shading of a visible incident light is rendered by:

$$s = \langle \mathbf{n}, \mathbf{l}_{in} \rangle,$$

where $\mathbf{n}$ denotes the normal to a surface. $\mathbf{l}_{in}$ is a three-dimensional vector, denoting a gray-scale light. The direction and magnitude of $\mathbf{l}_{in}$ respectively represent the direction and intensity of the light. $\langle \cdot, \cdot \rangle$ is dot product.

In indoor environments, a surface is illuminated by incident lights of multiple directions. As indicated by Basri and Jacobs [6], the final shading of an illuminated point p is the summation of the contribution of each light:

$$s = \int_0^{2\pi} \int_0^{\frac{\pi}{2}} \langle \mathbf{n}, \mathbf{l}(\theta,\phi) \rangle \sin\theta d\theta d\phi, \tag{2}$$

where the function $\mathbf{l}(\theta,\phi)$ expresses incident lighting passing through the point (θ,ϕ) on the upper hemisphere of the point p. We rewrite Equation 2 as follows:

$$\begin{aligned} s =& n^x \int_0^{2\pi} \int_0^{\frac{\pi}{2}} l(\theta,\phi)^x \sin\theta d\theta d\phi + \\ & n^y \int_0^{2\pi} \int_0^{\frac{\pi}{2}} l(\theta,\phi)^y \sin\theta d\theta d\phi + \\ & n^z \int_0^{2\pi} \int_0^{\frac{\pi}{2}} l(\theta,\phi)^z \sin\theta d\theta d\phi \\ =& \langle \mathbf{n}, \mathbf{a} \rangle, \end{aligned}$$

where x, y, z represent the three components of vectors. An integrated lighting vector $\mathbf{a}$ is defined as:

$$\begin{cases} \mathbf{a} = [a^x, a^y, a^z], \\ a^c = \int_0^{2\pi} \int_0^{\frac{\pi}{2}} l(\theta,\phi)^c \sin\theta d\theta d\phi,\ c \in \{x,y,z\}, \end{cases} \tag{3}$$

The integrated lighting $\mathbf{a}$ adapts to the specific normal $\mathbf{n}$, as the upper hemisphere and $\mathbf{l}(\theta,\phi)$ will change along with the scene geometry. Therefore, $\mathbf{a}$ can only reconstruct shading correctly without changing the direction of $\mathbf{n}$. To model spatially-varying lighting conditions, we propose to estimate a map of pixel-wise integrated lighting vectors, *i.e.*, $\mathbf{A}$. In our experiments, this map also adapts imperfect predicted surface normals for generating detailed shading.

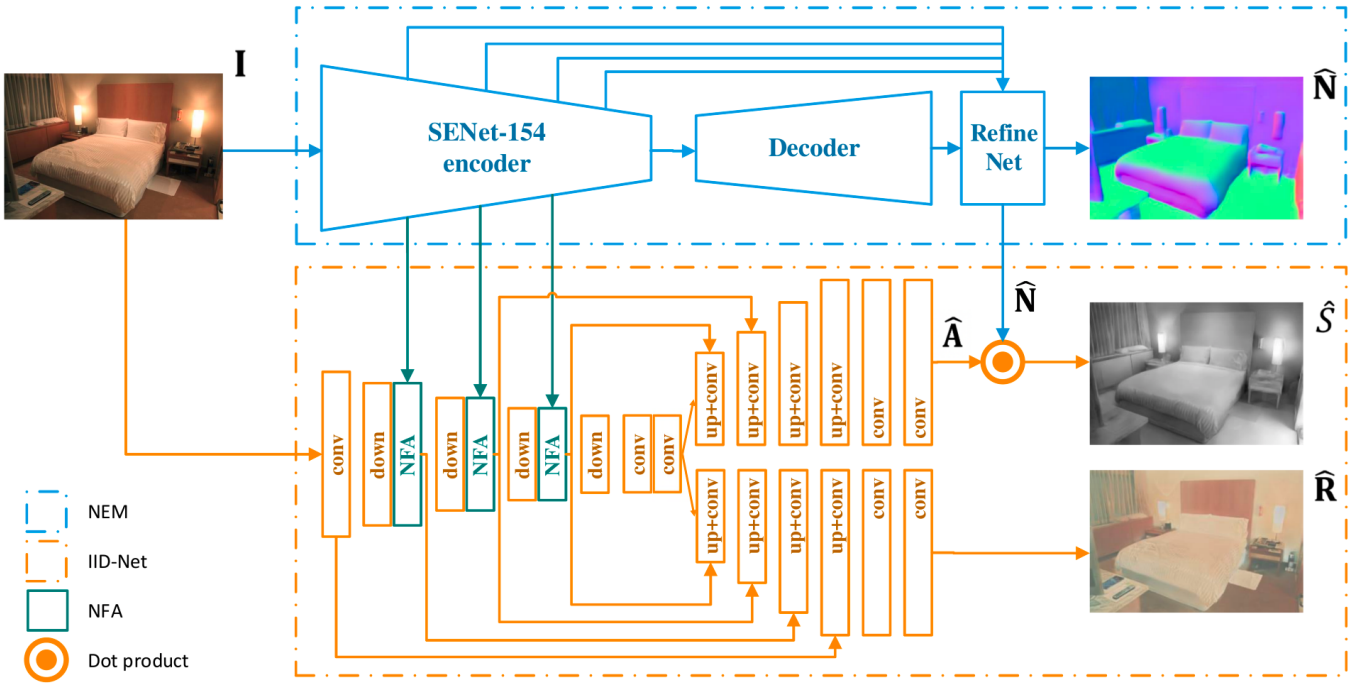
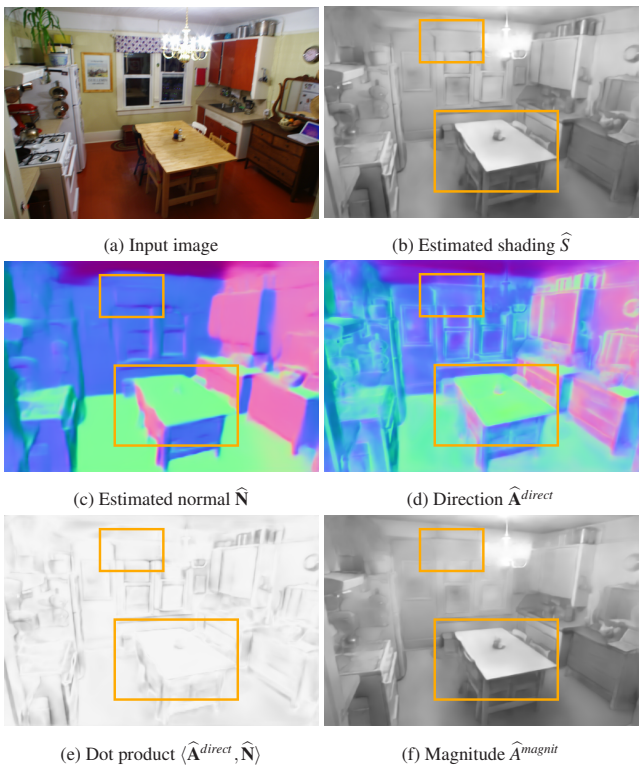


Fig. 4: Framework overview. Given a single sRGB input image, the proposed NIID-Net predicts a colorful reflectance image and a gray-scale shading intensity image. The NIID-Net contains a NEM (blue rectangle) and an IID-Net (orange rectangle). The IID-Net integrates surface normal knowledge via the NFAs and shading rendering.



(a) Input image (b) Estimated shading $\widehat{S}$

(c) Estimated normal $\widehat{\mathbf{N}}$ (d) Direction $\widehat{\mathbf{A}}^{direct}$

(e) Dot product $\langle \widehat{\mathbf{A}}^{direct}, \widehat{\mathbf{N}} \rangle$ (f) Magnitude $\widehat{A}^{magnit}$

Fig. 5: Visualization of integrated lighting. (c): the estimated surface normal map describes large planes in the scene, but it lacks geometry details. (d) and (e): the direction of integrated lighting vectors adapts surface normals for detailed and sharp geometry contours via dot product. (f): the magnitude of integrated lighting vectors captures variations of lighting intensity. Orange rectangles: smooth surface normals and smooth integrated lighting naturally generate smooth shading.

According to Equation 3, our image formation (Equation 1) is finally rewritten as:

$$\mathbf{I}_i = \mathbf{c} \times \langle \mathbf{N}_i, \mathbf{A}_i \rangle \times \mathbf{R}_i,$$

where $\mathbf{N}$, $\mathbf{A}$, and $\mathbf{R}$ denote a normal map, a map of integrated lighting, and a reflectance image, respectively. These three maps are predicted by the NIID-Net. The shading color $\mathbf{c}$ is estimated by energy optimization only before an application of image sequence editing, which is presented in the supplementary materials.

A predicted map of integrated lighting $\widehat{\mathbf{A}}$ is visualized in Fig. 5. The map is estimated based on priors of geometry knowledge, and it conveys features provided by surface normals. In particular, edge information matched with the direction of integrated lighting vectors is utilized to generate sharp object contours. In addition, the planarity information is exploited to reduce misinterpreted textures in estimated shading. Besides, as integrated lighting vectors are spatially-varying, local lighting effects can be produced.

3.3 Proposed Framework

Our proposed NIID-Net predicts a normal map $\widehat{\mathbf{N}}$, a shading intensity image $\widehat{S}$, and a reflectance image $\widehat{\mathbf{R}}$ from a single sRGB input image. Particularly, $\widehat{S}$ is the dot product of $\widehat{\mathbf{N}}$ and a predicted map of integrated lighting $\widehat{\mathbf{A}}$. As shown in Fig. 4, the deep neural network is made up of a normal estimation module (NEM) and an intrinsic image decomposition subnetwork (IID-Net). The NEM propagates geometry knowledge to the IID-Net via normal feature adapters (NFAs) and the dot product.

3.3.1 Normal Estimation Module

The NEM adopts the coarse-refinement framework proposed by Hu *et al.* [21] to predict normal maps $\widehat{\mathbf{N}}$. The NEM comprises an SENet-154 [22] encoder, a decoder containing several up-projection [30] layers, and a refinement module fusing multi-scale encoder features with outputs of the decoder. We modify the last layer of the refinement module to predict a three-channel normal map, which is then upsampled to match the spatial resolution of the map of integrated lighting from the IID-Net.

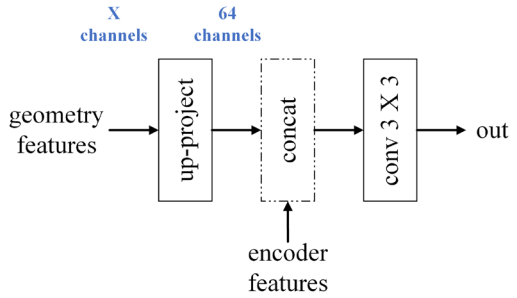


Fig. 6: A normal feature adapter. Geometry features and encoder features from the shading-reflectance shared encoder are fused.

3.3.2 Intrinsic Image Decomposition Subnetwork

We propose IID-Net, which is based on the U-Net [43] structure, to predict reflectance and shading. This subnetwork consists of three components: a shading-reflectance shared encoder, a reflectance decoder, and a shading decoder. Compared with the classical U-Net, we remove two low-level skip connections from the shared encoder to the shading decoder, as shading estimation does not require low-level features that are rich in image details. We assume that the high-frequency variations of input images are caused by reflectance rather than shading.

The shading-reflectance shared encoder extracts features for intrinsic images with the assistance of additional geometry features. It consists of four down blocks (4×4 convolutional layers with $stride = 2$), three NFAs and other three 3×3 convolutional layers ($stride = 1$). The decoders in the IID-Net use up-projection layers to upsample features. Each convolutional layer is followed with a batch normalization (BN) [23] and a ReLU [38] except for the last convolutional layers of the two decoders.

Normal feature adapter. As depicted in Fig. 6, each NFA $\times 2$ upsamples geometry features and encodes the features into 64 channels by an up-projection layer. Then it combines all the features by a 3×3 convolutional layer.

3.4 Supervision

3.4.1 Training Strategy

Ground-truth surface normals can be captured in real scenes, but dense intrinsic image annotations only come from synthetic data. For this situation, our framework offers an alternative way to train normal estimation and intrinsic image decomposition parts on different datasets. We first train the NEM on the real-world NYUv2 [48] and DIODE [50] datasets with dense ground-truth surface normals. After that, the IID-Net is trained on the synthetic CGI [34] dataset with the NEM fixed.

Our NIID-Net is implemented in PyTorch [42] and optimized by ADAM [28]. The NEM is initialized with parameters pre-trained by Hu *et al.* [21] for monocular depth estimation, and then it is refined with an initial learning rate of 0.00005, a batch size of 16, and weight decay of 0.0001. Training images from the NYUv2 dataset are randomly cropped to the size of 512×384 in depth-valid regions. Then input images from both the NYUv2 dataset and the DIODE dataset are resized to 320×240. We perform data augmentation via color jitter (brightness=0.4, contrast=0.4, saturation=0.4).

The IID-Net is optimized with an initial learning rate of 0.0001, and a batch size of 8. Training images are resized to 320×240. A random horizontal flip is implemented for data augmentation.

3.4.2 Surface Normal Loss

A predicted map of surface normals $\widehat{\mathbf{N}}$ is supervised by:

$$\mathscr{L}_{normal} = \left\| \mathbf{N} - \widehat{\mathbf{N}} \right\|_1 + \sum_m \left\| \nabla \mathbf{N}_m - \nabla \widehat{\mathbf{N}}_m \right\|_1,$$

where $\mathbf{N}$ is the ground-truth surface normal map, and ∇ denotes image gradient. m indicates different image scales.

3.4.3 Shading Loss

The complete shading loss is defined as:

$$\begin{cases} \mathscr{L}_{shading} = w_{s1} \left\| S - \widehat{S} \right\|_1 + w_{s2} \left\| \nabla S - \nabla \widehat{S} \right\|_1 \\ \qquad\qquad + w_{s3} \mathscr{L}_{smooth\_A}, \\ \widehat{S}_i = \langle \widehat{\mathbf{N}}_i, \widehat{\mathbf{A}}_i \rangle, \end{cases}$$

where w denotes a loss weight. Before rendering shading, surface normals $\widehat{\mathbf{N}}$ are normalized to unit vectors. To constrain predictions of integrated lighting and reduce misinterpreted textures in final shading images, we assume that integrated lighting vectors are locally consistent in the direction $\widehat{\mathbf{A}}_i^{direct}$ and magnitude $\widehat{A}_i^{magnit}$ both. This assumption is implemented by the local smoothness term $\mathscr{L}_{smooth\_A}$:

$$\begin{cases} \mathscr{L}_{smooth\_A} = \sum_{i \in \{i | \left\| (\nabla \widehat{\mathbf{N}})_i \right\|_1 \le \delta_N\}} \left\| (\nabla \widehat{A}^{magnit})_i \right\|_1 \\ \qquad\qquad + w_{a1} \sum_i \sum_{j \in \mathscr{N}_i} -\frac{1}{|\mathscr{N}_i|} \langle \widehat{\mathbf{A}}_i^{direct}, \widehat{\mathbf{A}}_j^{direct} \rangle, \\ \widehat{A}_i^{magnit} = \left\| \widehat{\mathbf{A}}_i \right\|_2, \\ \widehat{\mathbf{A}}_i^{direct} = \widehat{\mathbf{A}}_i / \widehat{A}_i^{magnit}, \end{cases}$$

where $\mathscr{N}_i$ denotes the 4-neighbors of a pixel i. The magnitude is smoothed only in the flat neighborhood, and we detect flat regions by $\{i | \left\| (\nabla \widehat{\mathbf{N}})_i \right\|_1 \le \delta_N\}$.

3.4.4 Reflectance Loss

In addition to supervision for pixel-wise reflectance values, we adopt the multi-scale gradient term [34] and introduce a reflectance consistency term $\mathscr{L}_{consist\_R}$ into the loss function for reflectance estimation:

$$\mathscr{L}_{reflect} = w_{r1} \left\| \mathbf{R} - \widehat{\mathbf{R}} \right\|_1 + w_{r2} \sum_m \left\| \nabla \mathbf{R}_m - \nabla \widehat{\mathbf{R}}_m \right\|_1 + w_{r3} \mathscr{L}_{consist\_R},$$

where m indicates different image scales.

Reflectance Consistency Term. The reflectance consistency term $\mathscr{L}_{consist\_R}$ is proposed to focus on pixels with similar ground-truth reflectance. For each iteration, we randomly select one key pixel $\mathbf{R}_{key}$, and then find a set of pixels $\mathscr{A}_{key}$ that have similar reflectance to $\mathbf{R}_{key}$:

$$\mathscr{A}_{key} = \{i | \left\| \mathbf{R}_i - \mathbf{R}_{key} \right\|_1 < \delta_R\},$$

where δ_R is a difference threshold. The reflectance consistency term is defined as:

$$\mathscr{L}_{consist\_R} = \sum_{i \in \mathscr{A}_{key}} \left\| (\mathbf{R}_i - \mathbf{R}_{key}) - (\widehat{\mathbf{R}}_i - \widehat{\mathbf{R}}_{key}) \right\|_1.$$

4 Experiments

4.1 Datasets

4.1.1 Training Datasets

NYUv2 dataset [48] is a real-world indoor dataset with raw depth labels annotated by the Microsoft Kinect. It consists of 249 video sequences for training and 215 video sequences for test. We compute ground-truth surface normals by Open3D [57]. The training subset we use has about 50K images, which was sampled by Hu *et al.* [21] from the official train split.

DIODE dataset [50] is a high-quality indoor and outdoor dataset, providing depth data collected by the FARO Focus S350. Dense ground-truth surface normals computed from depth are also provided. The input

Table 1: Ablation study. The metrics based on P(c) are particularly improved, which means that our proposed NFAs and MIL improve shading estimation significantly in challenging areas. After adding the reflectance decoder, the performance of the final model degrades slightly.

Method	P(u)(↑)		P(c)(↑)	
	AP(%)	mF1	AP(%)	mF1
Base + low-level skip.	98.82	0.949	96.69	0.902
Base	98.73	0.947	97.21	0.915
Base + NFAs	**99.18**	**0.960**	98.16	0.932
Base + NFAs + MIL	99.06	0.956	**98.47**	**0.942**
Final model	99.12	0.957	98.40	0.940

images and depth/normal maps have a resolution of 1024×768. We use the official indoor train split, comprising approximately $8,500$ images.

CGI dataset [34], which provides pixel-wise ground-truth reflectance and shading, is a scene-centric and photo-realistically rendered dataset. It re-renders over 20K images from SUNCG [49] and incorporates 152 images rendered by Bonneel *et al.* [11].

4.1.2 Test Datasets and Metrics

IIW dataset [9] comprises $5,230$ real-world photos (mostly of indoor scenes) and total $875,833$ pairs of human-annotated relative reflectance judgements. For each pair of pixels (R_i, R_j), the dataset annotates the reflectance intensity of the pixel i is lighter, equal, or darker to that of the pixel j. We use the test split provided by Narihira *et al.* [40].

For reflectance estimation evaluation, the IIW dataset introduces a human-perceptual error metric: weighted human disagreement rate (WHDR). WHDR is the average rate of how often predictions and labels are inconsistent. And the average is weighted by human confidence weights.

SAW dataset [29] contains $6,677$ images of real-world indoor scenes: $5,228$ images from the IIW dataset and $1,449$ images from the NYUv2 dataset. The official test set of SAW is consistent with the test sets of IIW and NYUv2, so the training data for surface normal estimation does not appear in the test set.

SAW provides smooth or non-smooth shading labels for some pixels, and thereby shading estimation is assessed as a binary classification problem. For scale invariance, Li and Snavely [35] propose to make judgements in the log domain and classify a pixel i as having smooth shading if it satisfies $\left\|(\nabla \log \widehat{S})_i\right\|_2 < \delta_{smooth}$. In addition to the unweighted precision (P(u)) [35], Li and Snavely propose the challenge precision (P(c)) [34]. When evaluating shading using P(c), a pixel is weighted less if it is located in an easy region where the input image intensity and the shading intensity are both smooth. Furtherly, precision-recall pairs are computed for different shading smoothness thresholds δ_{smooth}. The average unweighted precision (AP(u)) and the average challenge precision (AP(c)) are then calculated.

In order to take both precision and recall into account, we introduce the maximum F1 score:

$$\text{mF1} = \max_t \left(2 \times \frac{\text{precision}_t \times \text{recall}_t}{\text{precision}_t + \text{recall}_t}\right),$$

where t indicates the tth shading smoothness threshold.

Summarily, we quantitatively compare shading estimation by three metrics: precision-recall curves, AP, and mF1. Each metric is computed based on both P(u) and P(c).

In augmented reality applications, visual perception is of great concern. As a higher numerical accuracy does not necessarily indicate better visual effects, we also conduct qualitative comparisons, including a comparison of visualized intrinsic images and a comparison of image editing results.

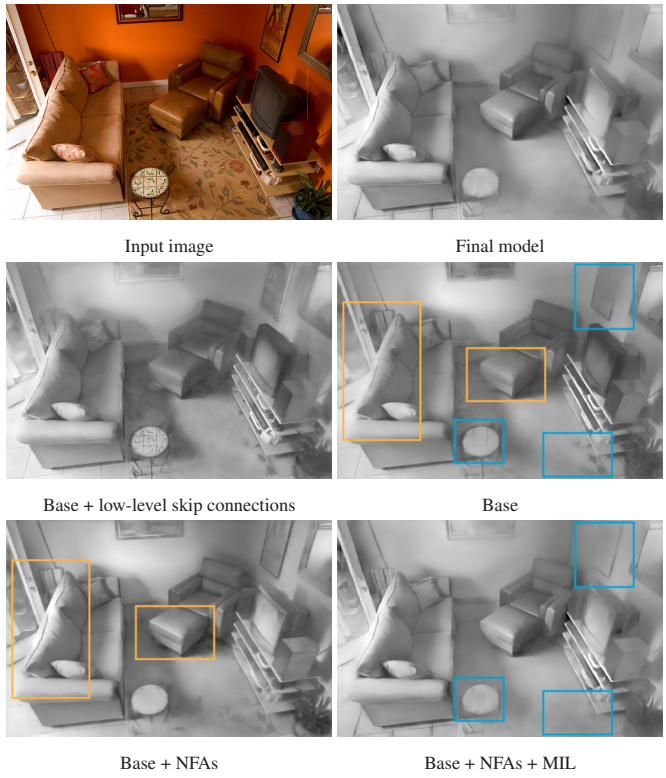


Fig. 7: Visualization of the ablation study results. Orange rectangles: the IID-Net recovers more distinct shading variations (*e.g.*, sharper highlights on the sofas and sharper contours) with geometry features encoded by the NFAs. Blue rectangles: texture residuals are reduced a lot due to the map of integrated lighting (MIL) which is locally smooth. The erroneous textures, which are slightly added into the predicted shading image by the NFAs, are finally removed when using the MIL.

4.2 Ablation Study

To avoid interference from the reflectance branch and better explain our contributions to shading estimation, the reflectance decoder is temporarily removed in the ablation study, and it is finally put back to obtain the final model.

The base network we use directly predicts single-channel shading intensity images via the IID-Net without any surface normal knowledge. In order to compare more fairly, the convolutional layers from the NFAs are preserved in the base network, but the NFAs do not incorporate geometry features from the NEM. We first evaluate the base network, and then measure the performance of the models that use NFAs and a map of integrated lighting (MIL) in order.

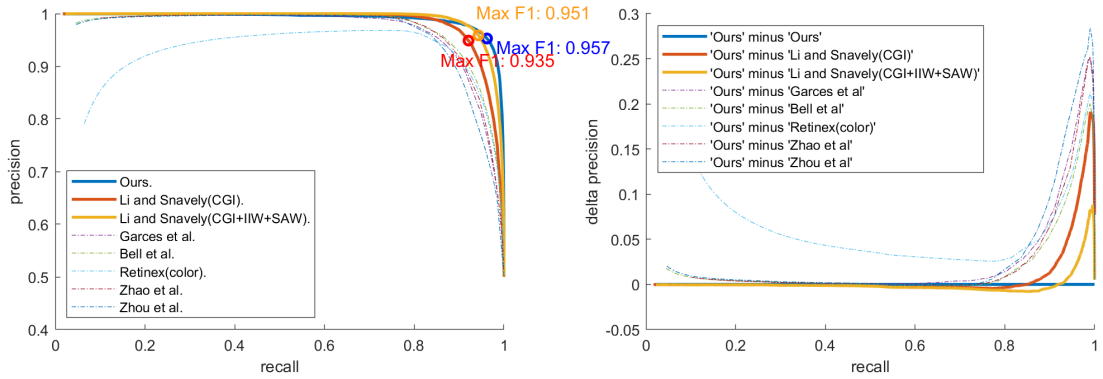
As shown in Table 1, our framework benefits from surface normal knowledge learned from the available datasets, especially in challenging areas that are focused on by the P(c) metric. The ablation study results are visualized in Fig. 7. As geometry priors support the IID-Net to understand complex lighting conditions, the IID-Net deals with shading variations better when it has NFAs. On flat regions, the smooth integrated lighting and smooth surface normals produce fewer artifacts in the final shading. Object contours are not harmed by the local smoothness assumption for integrated lighting, because the surface normal map provides reliable geometry cues.

4.3 Comparisons

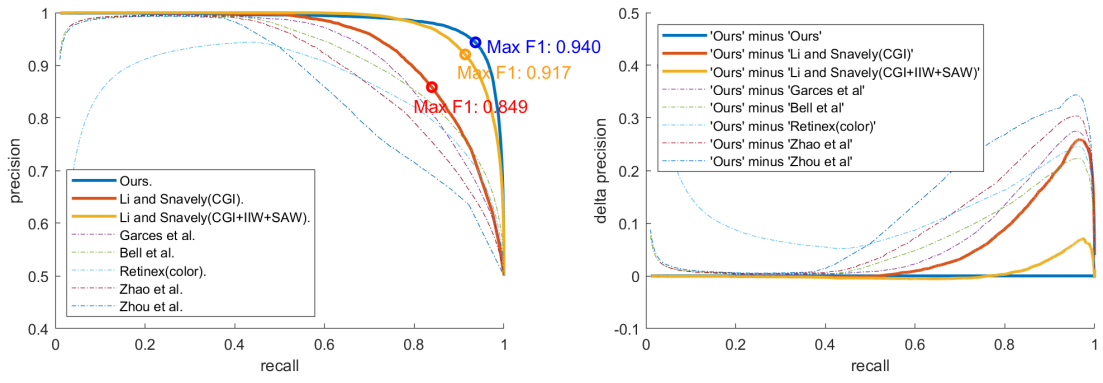
We test our method on a desktop PC with an NVIDIA GTX 1070 graphics card. For 320×240 images, the mean inference time per image (batch size = 1) is 70ms, that is, NIID-Net reaches 14 FPS.

Table 2: Numerical comparisons on the SAW and IIW test sets. Methods with "-" in the "Training set" column are based on optimization. "*" indicates predictions of the network have been post-processed. GLoSH [56] also exploits surface normal information. Our method outperforms all the previous methods in shading estimation. Our method achieves the fourth-best performance in reflectance estimation, and performs the best if compared with the neural networks only trained on the synthetic datasets (*i.e.*, CGI and SUNCG).

Method	Training set	WHDR(%)(↓)	P(u)		P(c)	
			AP(%)(↑)	mF1(↑)	AP(%)(↑)	mF1(↑)
Retinex(color) [20]	-	26.9	91.93	0.907	85.26	0.819
Garces *et al.* [19]	-	24.8	96.89	0.906	92.39	0.823
Zhao *et al.* [55]	-	23.8	97.11	0.909	89.72	0.795
Bi *et al.* [10]	-	17.7	-	-	-	-
Bell *et al.* [9]	-	20.6	97.37	0.913	92.18	0.834
Zhou *et al.* [58]	IIW	19.9	96.24	0.898	86.34	0.759
Nestmeyer and Gehler [41]	IIW	19.5	97.26	-	89.94	-
Nestmeyer and Gehler [41]*	IIW	17.7	96.85	-	88.64	-
Fan *et al.* [18]	IIW	**14.5**	-	-	-	-
Li and Snavely [34]	CGI	17.8	98.39	0.935	94.05	0.849
Li and Snavely [34]	CGI + IIW + SAW	15.5	99.11	0.951	97.93	0.917
GLoSH [56]	SUNCG	26.8	-	-	92.40	-
GLoSH [56]	SUNCG + IIW + SAW	15.2	-	-	95.01	-
Ours	CGI	16.6	**99.12**	**0.957**	**98.40**	**0.940**



(a) Normal (P(u))



(b) Challenge (P(c))

Fig. 8: Precision-recall curves for shading predictions on the SAW test set. The metric in the normal graph is P(u), while that in the challenge graph is P(c). In the right images, delta precision is defined as the precision of our model minus that of another method. Delta precision is positive if our method is better than the other. Precision in the challenge graph is much more distinguishable than that in the normal graph, which means that shading estimation for areas with rich textures is still challenging. When maintaining a high recall value (recall>0.9), our method achieves significantly higher precision than the others.

4.3.1 Evaluation on the SAW and IIW Datasets

As shown in Table 2, shading predictions are evaluated using P(u) and P(c), and reflectance predictions are evaluated using WHDR. Compared with the models only trained on the synthetic datasets (*i.e.*, CGI and SUNCG), our model achieves the best in both shading and reflectance estimation. Compared with the models trained more on the real-world datasets, our model also achieves the best in shading estimation even though the IID-Net part has never observed real-world data during training. Precision-recall curves are drawn in Fig. 8. In terms of the

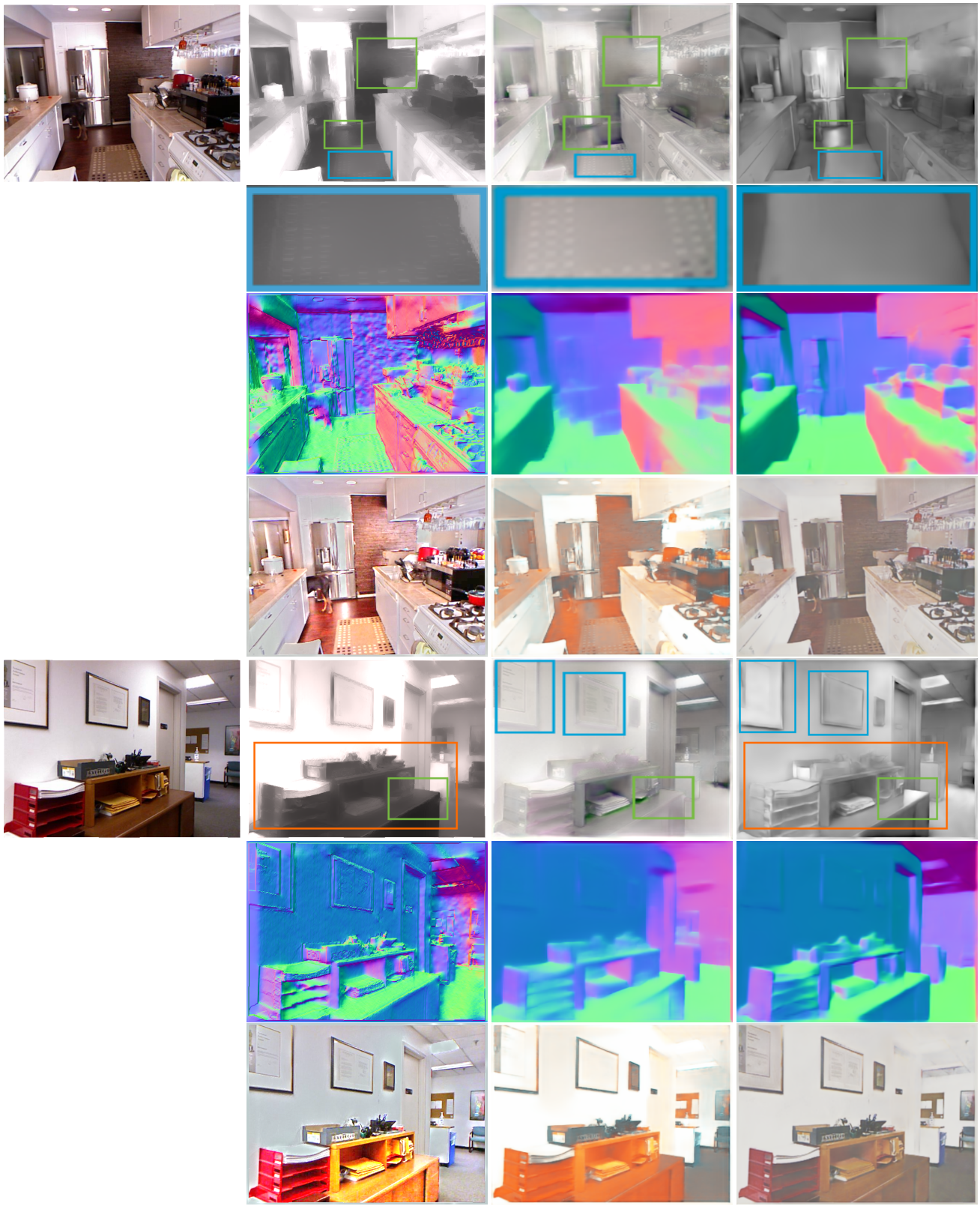


Fig. 9: Visual comparisons on the NYUv2 test set. The first, second and third rows are estimated shading images, predicted or ground-truth normal maps, and estimated reflectance images. We and GLoSH (SUNCG+IIW+SAW) [56] predict surface normals by the deep neural networks, while Chen and Koltun [16] compute surface normals from ground-truth depth. Geometry contours in our predicted shading images are the sharpest. Blue rectangles: we remove the most textures from the predicted shading. Green rectangles: we recover the highlights best. Orange rectangles: the intensity of predicted shading from Chen and Koltun [16] is strongly affected by that of the input image, while the intensity of our predictions is more coherent in the neighborhood. Our reflectance images are also better than those of Chen and Koltun [16], as many shading variations are shifted into their reflectance.

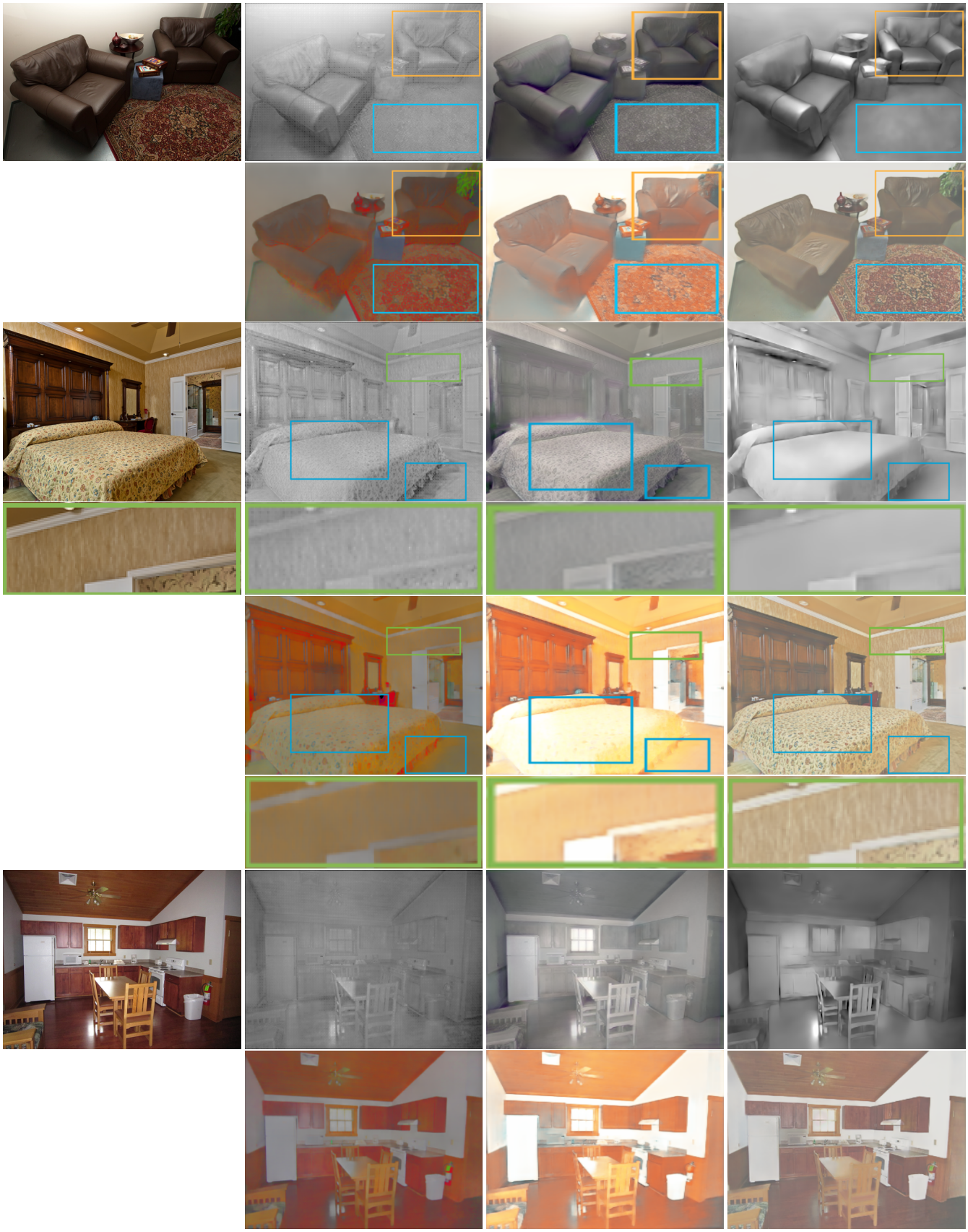


Fig. 10: Visual comparisons on the IIW/SAW test sets. We compare our results with Li and Snavely's (CGI+IIW+SAW) [34], and GLoSH (SUNCG+IIW+SAW) [56]. For each sample, the first row shows predicted shading images, and the second row shows predicted reflectance images. Blue rectangles: our shading results have the least texture residuals. Orange rectangles: our method best captures the shading effects. Green rectangles: our method predicts the most detailed reflectance as well as the most smooth shading. More results are presented in the supplementary material.

Captured photo Edited photo

Fig. 11: Image composition of a real scene and virtual posters. The input photo is captured by a Samsung Galaxy Note 8.

mF1 scores, our model achieves the best balance between precision and recall.

Fig. 9 shows visual comparisons among the methods that utilize surface normals. Our method predicts shading images with fewer misinterpreted textures and sharper variations (*e.g.*, geometry contours and highlights) than GLoSH [56], as our method makes use of geometry information more effectively. Compared with the work of Chen and Koltun [16] which takes ground-truth depth as input, our shading is as smooth as theirs, but their method can not deal with local shading effects well due to their imperfect priors. To remove residual textures from shading, Chen and Koltun strictly constrain shading to be as smooth as surface normals, causing many shading variations to be shifted into estimated reflectance through an image reconstruction energy term (similar to $\mathbf{I} = \widehat{\mathbf{S}} \times \widehat{\mathbf{R}}$). And the reflectance assumptions they conduct can not correct these erroneous shading residuals. Differently, shading effects are allowed to be more spatially-varying in our work, as priors for the pixel-wise integrated lighting are automatically learned by deep learning with the assistance of geometry features.

More visual comparisons are shown in Fig. 10. To the best of our knowledge, the model proposed by Li and Snavely [34] achieves the second best in shading estimation quantitatively at present (the first is our NIID-Net). However, their shading images are of low contrast, because scene geometry features are weakened when shading is smoothed without considering surface normals. Besides, in order to train the network on the real datasets with sparse reflectance annotations or only shading annotations, Li and Snavely adopt the assumption that reflectance is piece-wise constant, which rejects details in reflectance images. As intrinsic image decomposition part of our framework does not require training on real-world data, we do not adopt this commonly violated assumption. Note that as annotations of reflectance are sparse, the WHDR metric can not evaluate high-frequency reflectance. Among the methods taking a single RGB image as input, our framework achieves the best in recovering shading variations and avoiding texture shifting.

4.3.2 Image Editing Comparison

To better explain our improvements in visual effects, we compare our method with two previous methods [34, 56] by a surface re-coloring application in Fig. 1. In order to compare more fairly, for each method, the target reflectance is scaled globally to have the same average intensity as the source (estimated) reflectance in the target area. Then each composition result is produced by multiplying the scaled target reflectance with the estimated shading. For the "copy & paste" test, the target reflectance is scaled to match the original image, and then pasted to the original image without any illumination information. Among the four editing results, ours preserves the most lighting effects and has the least artifacts, which looks the most real.

4.4 Applications

We implement an image composition application that can insert virtual posters into real scenes by editing reflectance layers. Fig. 11 shows an example of photo editing. Furthermore, a demonstration of photorealistic editing of illumination-varying image sequences is provided in the supplementary materials. This application is suitable for augmented reality systems such as advertising and scene refurnishing.

5 Limitations

Although our NIID-Net achieves state-of-the-art performance in shading estimation, there are still some limitations that can be improved in the future. First, generating shading from the predicted surface normals is a double-edged sword. While benefiting from geometry cues provided by surface normals, shading images sometimes suffer from low-quality normal predictions. Although the estimated integrated lighting can alleviate this negative impact to some extent, it is difficult to correct serious errors in normals. Second, there is no clear definition of intrinsic images of non-Lambertian surfaces (*e.g.*, mirrors and glass). For this reason, the CGI dataset does not provide non-Lambertian ground truth, and we do not constrain the behavior of our model towards those surfaces, which leads to failure in some non-Lambertian surfaces. Third, we believe that if we could know where the highlights and shadows are in a scene, and incorporate this information into the local smoothness term of integrated lighting in the future, shading variations could be better recovered.

6 Conclusions

In this paper, we introduce NIID-Net, a novel learning-based framework that estimates intrinsic images from a single input image. NIID-Net effectively makes use of available normal datasets via the proposed normal feature adapters and the map of pixel-wise integrated lighting. The framework significantly improves shading in reducing texture residuals and recovering more variations. Besides, through image editing applications, we show that shading can be utilized to improve visual coherence between virtual contents and the real world. We believe that these improvements will promote practical applications of intrinsic images in augmented reality in the future.

Acknowledgments

The authors would like to thank the reviewers for their constructive comments on improving this paper. This work was partially supported by NSF of China (Nos. 61672457 and 61932003), and the Fundamental Research Funds for the Central Universities (No. 2019XZZX004-09).

References

[1] J. T. Barron and J. 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] J. T. Barron and J. Malik. Shape, illumination, and reflectance from shading. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(8):1670–1687, 2015.

[3] H. G. Barrow and J. M. Tenenbaum. Recovering intrinsic scene characteristics from images. *Computer Vision Systems*, 1978.

[4] A. S. Baslamisli, T. T. Groenestege, P. Das, H.-A. Le, S. Karaoglu, and T. Gevers. Joint learning of intrinsic images and semantic segmentation. In *Proceedings of the European Conference on Computer Vision*, pages 286–302, 2018.

[5] A. S. Baslamisli, H.-A. Le, and T. Gevers. Cnn based learning using reflection and retinex models for intrinsic image decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6674–6683, 2018.

[6] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(2):218–233, 2003.

[7] S. Beigpour, A. Kolb, and S. Kunz. A comprehensive multi-illuminant dataset for benchmarking of the intrinsic image algorithms. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 172–180, 2015.

[8] S. Beigpour, S. Shekhar, M. Mansouryar, K. Myszkowski, and H.-P. Seidel. Light-field appearance editing based on intrinsic decomposition. *Journal of Perceptual Imaging*, 1(1):10502–1, 2018.

[9] S. Bell, K. Bala, and N. Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics*, 33(4):159, 2014.

[10] S. Bi, X. Han, and Y. Yu. An L_1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition. *ACM Transactions on Graphics*, 34(4):78, 2015.

[11] N. Bonneel, B. Kovacs, S. Paris, and K. Bala. Intrinsic decompositions for image editing. *Computer Graphics Forum*, 36(2):593–609, 2017.

[12] N. Bonneel, K. Sunkavalli, J. Tompkin, D. Sun, S. Paris, and H. Pfister. Interactive intrinsic video editing. *ACM Transactions on Graphics*, 33(6):197, 2014.

[13] A. Bousseau, S. Paris, and F. Durand. User-assisted intrinsic images. *ACM Transactions on Graphics*, 28(5):130, 2009.

[14] I. Boyadzhiev, K. Bala, S. Paris, and E. Adelson. Band-sifting decomposition for image-based material editing. *ACM Transactions on Graphics*, 34(5):163, 2015.

[15] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision*, pages 611–625, 2012.

[16] Q. Chen and V. Koltun. A simple model for intrinsic image decomposition with depth cues. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 241–248, 2013.

[17] L. Cheng, C. Zhang, and Z. Liao. Intrinsic image transformation via scale space decomposition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 656–665, 2018.

[18] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf. Revisiting deep intrinsic image decompositions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8944–8952, 2018.

[19] E. Garces, A. Muñoz, J. Lopez-Moreno, and D. Gutierrez. Intrinsic images by clustering. *Computer Graphics Forum*, 31(4):1415–1424, 2012.

[20] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *IEEE International Conference on Computer Vision*, pages 2335–2342. IEEE, 2009.

[21] J. Hu, M. Ozay, Y. Zhang, and T. Okatani. Revisiting single image depth estimation: Toward higher resolution maps with accurate object boundaries. In *IEEE Winter Conference on Applications of Computer Vision*, pages 1043–1051, 2019.

[22] J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[23] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

[24] M. Janner, J. Wu, T. D. Kulkarni, I. Yildirim, and J. Tenenbaum. Self-supervised intrinsic image decomposition. In *Advances in Neural Information Processing Systems*, pages 5936–5946, 2017.

[25] J. Jeon, S. Cho, X. Tong, and S. Lee. Intrinsic image decomposition using structure-texture separation and surface normals. In *European Conference on Computer Vision*, pages 218–233. Springer, 2014.

[26] K. Kim, A. Torii, and M. Okutomi. Joint estimation of depth, reflectance and illumination for depth refinement. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1–9, 2015.

[27] S. Kim, K. Park, K. Sohn, and S. Lin. Unified depth prediction and intrinsic image decomposition from a single image via joint convolutional neural fields. In *European Conference on Computer Vision*, pages 143–159, 2016.

[28] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, 2015.

[29] B. Kovacs, S. Bell, N. Snavely, and K. Bala. Shading annotations in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6998–7007, 2017.

[30] I. Laina, C. Rupprecht, V. Belagiannis, F. Tombari, and N. Navab. Deeper depth prediction with fully convolutional residual networks. In *Fourth International Conference on 3D Vision*, pages 239–248, 2016.

[31] J. H. Lambert. *Photometria Sive de Mensura et Gradibus Luminis, Colorum et Umbrae*. Eberhard Klett, 1760.

[32] L. Lettry, K. Vanhoey, and L. Van Gool. Unsupervised deep single-image intrinsic decomposition using illumination-varying image sequences. *Computer Graphics Forum*, 37(7):409–419, 2018.

[33] C. Li, K. Zhou, and S. Lin. Simulating makeup through physics-based manipulation of intrinsic image layers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4621–4629, 2015.

[34] Z. Li and N. Snavely. CGIntrinsics: Better intrinsic image decomposition through physically-based rendering. In *Proceedings of the European Conference on Computer Vision*, pages 381–399, 2018.

[35] Z. Li and N. Snavely. Learning intrinsic image decomposition from watching the world. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9039–9048, 2018.

[36] A. Meka, G. Fox, M. Zollhöfer, C. Richardt, and C. Theobalt. Live user-guided intrinsic video for static scenes. *IEEE Transactions on Visualization and Computer Graphics*, 23(11):2447–2454, 2017.

[37] A. Meka, M. Zollhöfer, C. Richardt, and C. Theobalt. Live intrinsic video. *ACM Transactions on Graphics*, 35(4):109, 2016.

[38] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning*, pages 807–814, 2010.

[39] T. Narihira, M. Maire, and S. X. Yu. Direct intrinsics: Learning albedo-shading decomposition by convolutional regression. In *Proceedings of the IEEE international conference on computer vision*, pages 2992–2992, 2015.

[40] T. Narihira, M. Maire, and S. X. Yu. Learning lightness from human judgement on relative reflectance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2965–2973, 2015.

[41] T. Nestmeyer and P. V. Gehler. Reflectance adaptive filtering improves intrinsic image estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6789–6798, 2017.

[42] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

[43] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[44] S. Sengupta, A. Kanazawa, C. D. Castillo, and D. W. Jacobs. Sfsnet: Learning shape, reflectance and illuminance of facesin the wild'. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6296–6305, 2018.

[45] M. Shakeri and H. Zhang. Illumination invariant representation of natural images for visual place recognition. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 466–472, 2016.

[46] J. Shi, Y. Dong, H. Su, and S. X. Yu. Learning non-lambertian object intrinsics across shapenet categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1685–1694, 2017.

[47] Z. Shu, E. Yumer, S. Hadap, K. Sunkavalli, E. Shechtman, and D. Samaras. Neural face editing with intrinsic image disentangling. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5541–5550, 2017.

[48] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from RGBD images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012.

[49] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1746–1754, 2017.

[50] I. Vasiljevic, N. Kolkin, S. Zhang, R. Luo, H. Wang, F. Z. Dai, A. F. Daniele, M. Mostajabi, S. Basart, M. R. Walter, and G. Shakhnarovich. DIODE: A Dense Indoor and Outdoor DEpth Dataset. *CoRR*, abs/1908.00463, 2019.

[51] C. Wu, M. Zollhöfer, M. Nießner, M. Stamminger, S. Izadi, and C. Theobalt. Real-time shading-based refinement for consumer depth cameras. *ACM Transactions on Graphics*, 33(6):200, 2014.

[52] S. Yan, C. Wu, L. Wang, F. Xu, L. An, K. Guo, and Y. Liu. Ddrnet: Depth map denoising and refinement for consumer depth cameras using cascaded cnns. In *Proceedings of the European Conference on Computer Vision*, pages 155–171, 2018.

[53] L.-F. Yu, S.-K. Yeung, Y.-W. Tai, and S. Lin. Shading-based shape refinement of RGB-D images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1415–1422, 2013.

[54] Y. Yu and W. A. Smith. Inverserendernet: Learning single image inverse rendering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3155–3164, 2019.

[55] Q. Zhao, P. Tan, Q. Dai, L. Shen, E. Wu, and S. Lin. A closed-form solution to retinex with nonlocal texture constraints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7):1437–1444, 2012.

[56] H. Zhou, X. Yu, and D. W. Jacobs. Glosh: Global-local spherical harmonics for intrinsic image decomposition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 7820–7829, 2019.

[57] Q.-Y. Zhou, J. Park, and V. Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.

[58] T. Zhou, P. Krahenbuhl, and A. A. Efros. Learning data-driven reflectance priors for intrinsic image decomposition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3469–3477, 2015.

[59] M. Zollhöfer, A. Dai, M. Innmann, C. Wu, M. Stamminger, C. Theobalt, and M. Nießner. Shading-based refinement on volumetric signed distance functions. *ACM Transactions on Graphics*, 34(4):96, 2015.

[60] D. Zoran, P. Isola, D. Krishnan, and W. T. Freeman. Learning ordinal relationships for mid-level vision. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 388–396, 2015.