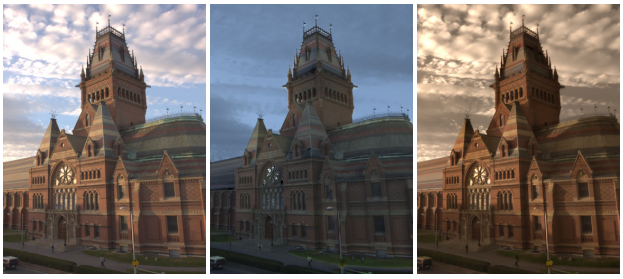
Learning to Separate Multiple Illuminants in a Single Image

Zhuo Hui,[1] Ayan Chakrabarti,[2] Kalyan Sunkavalli,[3] and Aswin C. Sankaranarayanan[1]
[1]Carnegie Mellon University [2]Washington University in St. Louis [3]Adobe Research

Abstract

We present a method to separate a single image captured under two illuminants, with different spectra, into the two images corresponding to the appearance of the scene under each individual illuminant. We do this by training a deep neural network to predict the per-pixel reflectance chromaticity of the scene, which we use in a physics-based image separation framework to produce the desired two output images. We design our reflectance chromaticity network and loss functions by incorporating intuitions from the physics of image formation. We show that this leads to significantly better performance than other single image techniques and even approaches the quality of the prior work that require additional images.



(a) Input image (b) Output separated images

Figure 1. Our method separates a single image (a) captured under two illuminants with different spectra (sun and sky illumination here) into two images corresponding to the appearance of the scene under the individual lights. Note that we are able to accurately preserve the shading and shadows for each light.

1. Introduction

Natural environments are often lit by multiple light sources with different illuminant spectra. Depending on scene geometry and material properties, each of these lights causes different light transport effects like color casts, shading, shadows, specularities, etc. An image of the scene combines the effects from the different lights present, and is a superposition of the images that would have been captured under each individual light. We seek to invert this superposition, i.e., separate a single image observed under two light sources, with different spectra, into two images, each corresponding to the appearance of the scene under one light source alone. Such a decomposition can give users the ability to edit and relight photographs, as well as provide information useful for photometric analysis.

However, the appearance of a surface depends not only on the properties of the light sources, but also on its geometry and material properties. When all of these quantities are unknown, disentangling them is a significantly ill-posed problem. Thus, past efforts to achieve such separation have relied heavily on extensive manual annotation [8, 7, 9] or access to calibrated scene and lighting information [12, 11]. More recently, Hui et al. [24, 25] demonstrate that the lighting separation problem can be reliably solved if one additionally knows the reflectance chromaticity of all surface points — which they recover by capturing a second image of the same scene under flash lighting. Given that the flash image is used in their processing pipeline only for estimating the reflectance chromaticity, could we computationally estimate the reflectance chromaticity from a *single image*, thereby avoiding the need to capture a flash photograph all together? This would greatly enhance the applicability of the method especially for scenarios where it is challenging to sufficiently illuminate every pixel with the flash; for example, when the flash is not strong enough, the scene is large, or the ambient light sources are too strong.

Our work is also motivated by the success of deep convolutional neural networks for solving closely related problems like intrinsic decompositions [32, 43], and reflectance estimation [41, 36, 34]; hence, we propose training a deep convolution neural network to perform this separation. However, we find that standard architectures, trained only with the respect to the quality of the final separated images, are unable to learn to effectively perform the separation. Therefore, we guide the design of our network using a physics-based analysis of the task [25] to match the expected sequence of inference steps and intermediate outputs — reflectance chromaticities, shading chromaticities, separated shading maps, and final separated images. In addition to ensuring that our architecture has the ability to express these required computations, this decomposition also allows us to provide supervision to intermediate layers in

our network, which proves crucial to successful training. Once trained, we find that our approach is able to successfully solve this ill-posed problem and produce high-quality lighting decompositions that, as can be seen in Figure 1, capture complex shading and shadows. In fact, our network is able to match, and in specific instances outperform, the quality of results from Hui et al.'s two-image method [25], despite needing only a single image as input.

Contributions. We make the following contributions.

1. We introduce a learning-based approach for separating a single image into two images, each illuminated by light source of a distinct spectra.
2. We incorporate physical-based constraints in the network design and training, a strategy that is critical to our methods success.
3. We demonstrate the practical utility of the method for a wide range of applications, including white balancing, sun/sky separation for the outdoor scenes.

2. Related Work

Estimating illumination and scene geometry from a single image is a highly ill-posed problem. Previous work has focused on specific subsets of this problem; we discuss previous works on illumination analysis as well as prior intrinsic image decomposition methods that aim to jointly estimate illumination and surface reflectance.

Illumination estimation. Estimating the ambient illumination from a single photograph has been a long-standing goal in computer vision and computer graphics. A number of past techniques have studied color constancy [18] — the problem of removing the color casts of ambient illumination. One popular solution is to model the scene with single dominant light source [15, 14, 17]. To deal with mixtures of illuminants in a scene, previous works [13, 19, 37] typically characterize each local region with a different but single light source. However, these approaches do not generalize well to scenes where multiple light sources mix smoothly. To address this, Boyadzhiev et al. [10] utilize user scribbles to indicate scene attributes such as white surfaces and constant lighting regions. Hsu et al. [23] propose a method to address mixtures of two light sources in the scene; however, they require precise knowledge of the color of each illuminant. Prinet et al. [35] resolve the color chromaticity of two light sources from a sequence of images by leveraging the consistency of the reflectance of the scene. Sunkavalli et al. [40] demonstrate this (and image separation) for time-lapse sequences of outdoor scenes.

In parallel, many techniques have been developed to explicitly model the illumination of the scene, rather than removing the color of the illuminants. Lalonde et al. [31] propose the parametric model to characterize the sky and sun for the outdoor photographs. Hold-Geoffroy et al. [22] extend the idea to model the outdoor illumination by incorporating a deep neutral network. Gardner et al. [16] similarly train a deep neural network to recover indoor illumination from a single LDR photograph. In contrast, our method does not explicitly model the illuminants, but directly regresses the single-illuminant images.

Intrinsic image decomposition. Intrinsic image decomposition methods seek to separate a single image into a product of reflectance and illumination layers. This problem is commonly solved by assuming that the reflectance of the scene is piece-wise constant while the illumination varies smoothly [4]. Several approaches build on this by further imposing priors on non-local reflectance [42, 38, 5], or on the consistency of reflectance for image sequences captured with varying illumination [30, 21, 26]. A common assumption in intrinsic image methods is that the scene is lit by a single dominant illuminant. This does not generalize to real world scenes that are often illuminated with multiple light sources with different spectra. Recent methods have proposed using deep neural networks, trained with large amounts of data, to address this problem [43, 32, 33]. While effective, these techniques also focus on scenes illuminated with a single light source. Barron and Malik [2, 3] resolve this by incorporating a global lighting model with hand-crafted priors. While this lighting model works well for single objects, it is unable to capture high-frequency spatial information, like shadows that are often present in real scenes. In comparison, our technique generalizes well to complex scenes lit with mixtures of multiple light sources. In addition, as opposed to predicting the reflectance of the scene, our method only requires predicting its *chromaticity*, which is an easier problem to solve.

3. Problem Statement

Our objective is to take as input, a single photograph of a scene lit by a mixture of two illuminants, and estimate the images lit by each single light source. In this section, we set up the image formation model and describe the physical priors we impose to supervise the intermediate results produced by the network.

We adopt the image formation model from Hui et al. [25] by assuming that the scene is Lambertian and is imaged by a three-channel color camera. However, instead of modeling infinite-dimensional spectra using subspaces, we assume that the camera color response is narrow-band, allowing us to characterize both the light source and albedo in RGB. That is, the intensity observed at a pixel $\mathbf{p}$ in a single

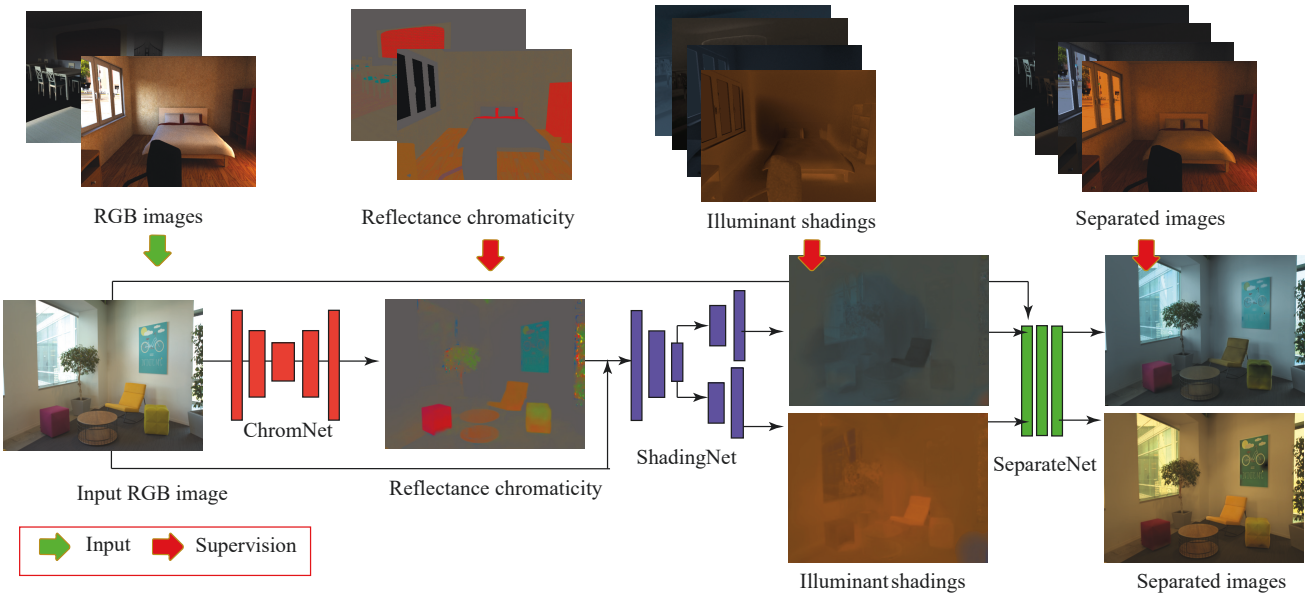


Figure 2. Given a single image lit by a mixture of two lights, our method automatically produces the images lit by each of these illuminants. We train a cascade of three sub-networks with three specific tasks. First, we estimate the reflectance color chromaticity of the scene via ChromNet. Given this estimation, we concatenate it with the input RGB image and feed them into ShadingNet to predict the illuminant shadings. We append these to the input image and pass it to SeparateNet to produce the output. During training, we supervise the reflectance chromaticity, illuminant shadings and the separated images.

photograph I is given by:

$$I^c(\mathbf{p}) = R^c(\mathbf{p}) \sum_{i=1}^{N} \lambda_i(\mathbf{p})\, \ell_i^c, \quad \text{for } c \in \{r, g, b\}, \qquad (1)$$

where $R(\mathbf{p}) = [R^r(\mathbf{p}), R^g(\mathbf{p}), R^b(\mathbf{p})]$ is the three-color albedo. In our work, we focus on the scenes that are lit by $N = 2$ light sources and we denote the light chromaticities as $\{\boldsymbol{\ell}_1, \boldsymbol{\ell}_2\}$. Note that $\boldsymbol{\ell}_i = [\ell_i^r, \ell_i^g, \ell_i^b] \in \mathbb{R}^3$ with $\sum_c \ell_i^c = 1$. Similar to Hui et al. [25], we assume that the light source chromaticities are unique, i.e., $\boldsymbol{\ell}_1 \neq \boldsymbol{\ell}_2$. The term $\lambda_i(\mathbf{p})$ is the shading observed at pixel $\mathbf{p}$ due to the i-th light source multiplied by the light-source brightness. Given the fact that two sources with the same color are clustered together, the shading term $\lambda_i(\mathbf{p})$ has a complex dependency on the lighting geometry and does not have a simple analytical form. Our goal is to compute the separated images corresponding to the each light source k as:

$$\widehat{I}^c_{\text{sep},k}(\mathbf{p}) = R^c(\mathbf{p})\, \lambda_k(\mathbf{p})\, \ell_k^c. \qquad (2)$$

To solve this, Hui et al. [25] capture an additional image under flash illumination to directly compute reflectance color chromaticities for each pixel. They use these to disentangle reflectance from illumination shading, and solve for the color of each light source as well as the per-pixel contribution of each illuminant. We provide a quick summary of the key steps of their computational pipeline below, and refer readers to their paper for more details.

Step 1 — Flash to reflectance chromaticity. Given the flash color, the pure flash photograph enables us to estimate the reflectance chromaticity, $\alpha^c(\mathbf{p})$, that is defined as:

$$\alpha^c(\mathbf{p}) = \frac{R^c(\mathbf{p})}{\sum_{\tilde{c}} R^{\tilde{c}}(\mathbf{p})}. \qquad (3)$$

Step 2 — Estimate shading chromaticity. The reflectance chromaticity $\alpha^c(\mathbf{p})$ can be used to remove the contribution from albedo as follows. We define $\beta^c(\mathbf{p}) = I^c(\mathbf{p})/\alpha^c(\mathbf{p})$ and normalize it to obtain shading chromaticities as:

$$\gamma^c(\mathbf{p}) = \frac{\beta^c(\mathbf{p})}{\sum_{\tilde{c}} \beta^{\tilde{c}}(\mathbf{p})} = \frac{\sum_i \lambda_i(\mathbf{p})\, \ell_i^c}{\sum_i \lambda_i(\mathbf{p})} = \sum_i z_i(\mathbf{p})\, \ell_i^c, \qquad (4)$$

where $z_i(\mathbf{p}) = \lambda_i(\mathbf{p}) / \left(\sum_{\tilde{i}} \lambda_{\tilde{i}}(\mathbf{p})\right)$ is relative shading.

Step 3 — Estimate relative shading. The histogram of shading chromaticities across the image can be fit to a multi-illuminant model (see [25] for details) to estimate the illuminant shadings S_i^c for each light source, defined as:

$$S_i^c(\mathbf{p}) = z_i(\mathbf{p}) \ell_i^c. \qquad (5)$$

The separated images can then be recovered as:

$$\widehat{I}^c_{\text{sep},k}(\mathbf{p}) = I^c(\mathbf{p}) \frac{S_k^c(\mathbf{p})}{\sum_{i=1}^{N} S_i^c(\mathbf{p})}. \qquad (6)$$

In this paper, we design our network by mimicking the steps in the derivation above, but each processing element is

replaced with deep networks as shown in Figure 2. In particular, we utilize three sub-networks — ChromNet, ShadingNet and SeparateNet — to estimate the reflectance chromaticity, illuminant shadings and separated images, respectively. ChromNet predicts the values of reflectance chromaticity α, defined in (3), with its input being the RGB image that we seek to separate. ShadingNet takes in as the output of ChromeNet concatenated with the input RGB image to regress the illuminant shadings in (5). Finally, SeparateNet gathers the estimated illuminant shadings as well as the input RGB image to estimate the separated images.

4. Learning Illuminant Separation

We now dicuss our proposed method for decomposing an input photo into images lit by individual illuminants, including how we generate training data with ground truth scene annotations (for reflectance and shading), and how we design our proposed deep neural network for this problem.

4.1. Generating the training dataset

We utilize the databases of CGIntrinsics [32] and Flash/No-Flash [1] to produce images with (approximate) ground truth reflectance chromaticity, illuminant shadings and separated images. Figure 3 shows training data examples from each dataset.

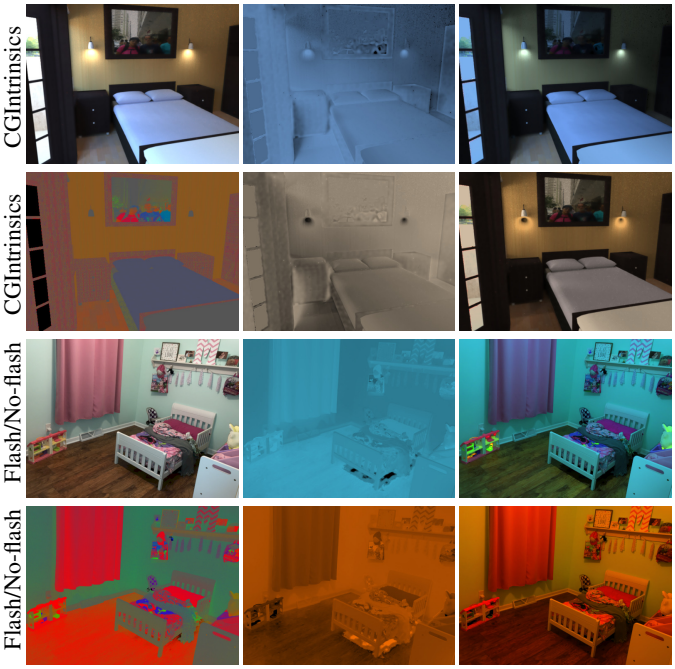
The CGIntrinsics dataset consists of 20160 rendered scenes from SUNCG [39]. It provides the ground truth reflectance, and hence, the reflectance chromaticity. The shadings chromaticity is then estimated via (4).

The Flash/No-flash dataset consists of 2775 image pairs. We estimate the reflectance chromaticity as the color chromaticity of the pure flash image, which is the difference between the flash and the no-flash photograph. We anecdotally observed that the majority of the scenes in this dataset are only illuminated by a single light source — which, as such, makes it uninteresting for our application. To resolve this, we add the flash image back to no-flash image and create photographs illuminated by two light sources. By changing the color of the flash photograph, we can enhance the amount of training data; this allows us to generate 29060 input-output pairs, where the input is a photo, and the output is the reflectance chromaticity, a pair of its corresponding illuminant shadings as well as the separated images.

4.2. Network architecture

As shown in Figure 2, we use a deep neural network to match the computation of the separation algorithm in Section 3. Specifically, our network consists of three subnetworks that produce the reflectance chromaticity, illuminant shadings, and the separated images respectively.

ChromNet. We design the first sub-network to explicitly estimate the reflectance chromaticity (3) from the input color image. This essentially requires the network to solve the ill-posed problem of estimating and removing the illumination color cast given only a single photograph. We adopt an architecture similar to that of Johnson et al. [28] to map the input image to a three channel reflectance chromaticity map.[1]



(a) Input (Top) / Chromaticity (Bottom) (b) Illuminant shadings (c) Separated images

Figure 3. We showcase each sample of train pairs from CGIntrinsics (top) and Flash/No-flash database (bottom). Given the input photograph (a), we use reflectance chromaticity together with illuminant shadings (b) and separated images (c) to supervise the output of the network.

ShadingNet. The second sub-network in our framework takes reflectance chromaticity estimates as inputs, and solves for the two illuminant shadings in (5). From Section 3, we expect the first part of this computation to involve deriving γ from the chromaticities and original input, on a purely per-pixel basis as per (4). However, we found computing the γ values explicitly to lead to instability in training, likely since this involves a division. Instead, we produce a general feature map intended to encode the γ information (note that we do not require it to exactly correspond to γ values) by concatenating the input image with the estimated chromaticities. Given this feature map, our second sub-network produces the two separated illuminant shading maps. Since this requires global reasoning, we use an architecture similar to the pixel-to-pixel network of Isola

[1] A detailed description of the construction of each subnetwork is provided in the supplemental material.

et al. [27] to incorporate a large receptive field.

SeparateNet. Given the illuminant shadings and previously estimated reflectance chromaticity, the last computation step is to produce the separated image. Here, we use a series of pixel-wise layers to express the computation in (6). Our third sub-network concatenates the two predicted shading maps and the input RGB photograph into a nine-channel input, and uses three 1×1 convolution layers to produce a six-channel output corresponding to the two final separated RGB images.

Note that the output of our first sub-network—reflectance chromaticity—is sufficient to perform separation using the method of Hui et al. [25]. However, training this sub-network based directly on the quality of reflectance chromaticity estimates proves insufficient, because the final separated image quality can degrade differently with different kind of errors in chromaticity estimates. Thus, our goal is to instead train the reflectance chromaticity estimation sub-network to be optimal towards final separation quality. Unfortunately, the separation algorithm in [25] has non-differentiable processing steps, as well as other computation that produces unstable gradients. Hence, we use two additional sub-networks to approximate the processing in Hui et al.'s algorithm [25]. However, once trained, we find it is optimal to directly use the reflectance chromaticity estimates with the exact algorithm in [25], over the output of these sub-networks.

4.3. Loss functions

ChromNet loss. For the reflectance chromaticity estimation task, we use a scale-invariant loss. We also incorporate ℓ_1 loss in gradient domain, to enforce that the estimated reflectance chromaticity is piece-wise constant. In particular, we define our loss function as

$$\mathcal{L}_\alpha = \frac{1}{M}\sum_{i=1}^{M}\|\alpha_i^* - c_\alpha\alpha_i\|_1 + \sum_{t=1}^{L}\frac{1}{M_t}\sum_{i=1}^{M_t}\|\nabla\alpha_{t,i}^* - c_\alpha\nabla\alpha_{t,i}\|_1, \tag{7}$$

where α^* denotes the predicted chromaticity, α is the ground truth provided, and c_α is a term to compensate for the global scale difference, which can be estimated via least squares. We also use a mask to disregard the loss at pixels where we do not have reliable ground truth (e.g. pixels that are close to black or pixels corresponding to the outdoor environment map in the SUNCG dataset). M indicates the total number of valid pixels in an image. Similar to the approach of Li et al. [32], we include a multi-scale matching term, where L is the total number of layers specified (3 in the paper) and M_t denotes the corresponding number of pixels not masked as invalid pixels.

ShadingNet loss. We impose an ℓ_2 loss on both the absolute value and the gradients of the relative shadings. This encourages spatially smooth shading solutions (as is commonly done in prior intrinsic images work). However, the network outputs two potential relative shadings and swapping these two predictions should not induce any loss. To address this, we define our loss function as

$$\mathcal{L}_S = \min\{\mathcal{L}_{S_{11}} + \mathcal{L}_{S_{22}}, \mathcal{L}_{S_{12}} + \mathcal{L}_{S_{21}}\}$$

where $\mathcal{L}_{S_{ij}}$ denote the loss between the i-th output with j-th illuminant shadings defined in (5). Specifically, $\mathcal{L}_{S_{ij}}$ is defined as $\mathcal{L}_{S_{ij}} = \mathcal{L}_{\text{data}(i,j)} + \mathcal{L}_{\text{grad}(i,j)}$, where

$$\mathcal{L}_{\text{data}(i,j)} = \frac{1}{M}\sum_{u=1}^{M}\|S_{i,u}^* - c_S S_{j,u}\|_2, \tag{8}$$

$$\mathcal{L}_{\text{grad}(i,j)} = \sum_{t=1}^{L}\frac{1}{M_t}\sum_{u=1}^{M_t}\|\nabla S_{i,t,u}^* - c_S\nabla S_{i,t,j}\|_2, \tag{9}$$

Here, S_i^* denotes the i-th illuminant shading prediction while S_j is the ground truth, and c_S is the global scale to compensate for the illuminant brightness.

SeparateNet loss. Our loss for the two separated images is similar to our ShadingNet loss:

$$\mathcal{L}_I = \min\{\mathcal{L}_{I_{11}} + \mathcal{L}_{I_{22}}, \mathcal{L}_{I_{12}} + \mathcal{L}_{I_{21}}\},$$

where $\mathcal{L}_{I_{ij}}$ is the ℓ_1 loss. Specifically, $\mathcal{L}_{I_{ij}}$ is defined as

$$\mathcal{L}_{I_{ij}} = \frac{1}{M}\sum_{u=1}^{M}\|I_{i,u}^* - c_I I_{j,u}\|_1, \tag{10}$$

where, I_i^* denote the i-th separated image predication while I_j is the ground truth for the j-th light source, and c_I is scale factor for the global intensity difference.

Training details. We resize our training images to 384×512. We use Adam optimizer [29] to train our network with $\beta_1 = 0.5$. The initial learning rate is set to be 5×10^{-4} for all sub-networks. We cut down the learning rate by $1/10$ after 35 epochs. We then train for 5 epochs with the reduced learning rate. We ensure that all our networks have converged with this scheme.

5. Evaluation

We now present an extensive quantitative and qualitative evaluation of our proposed method. Please refer to our supplementary material for more details and results.

5.1. Test dataset

Synthetic benchmark dataset. To quantitatively evaluate our method, we utilize the high quality synthetic dataset of [6]. This dataset has approximately 52 scenes, each rendered under several different single illuminants. We first

Name	Network architecture	Supervision
Chrom-Only	ChromNet	Chromaticity
Final-Only	ChromNet + ShadingNet + SeparateNet	Sep. images
Full-Direct	ChromNet + ShadingNet + SeparateNet	Chromaticity + Shadings + Sep. images
SingleNet	Single Unet	Sep. images

Table 1. Variant versions of proposed network architectures with different supervisions.

white balance each image of the same scene, and then modulate the white-balanced images with pre-selected light colors; these represent the ground truth separated images. The input images are then created by adding pairs of these separated images, each corresponding to one of the lights in the scene. We produce 400 test samples in the dataset, and evaluate our method using both the ground truth of reflectance chromaticity and separated results.

Real dataset. We also evaluate the performance of our proposed technique on real images captured for both indoor and outdoor scenes. Specifically, we utilize the dataset of the indoor scenes collected by Hui et al. [25] as well as time-lapse videos for outdoor scenes. Hui et al. [25] capture a pair of flash/no-flash for the same scene. We take the no-flash images in the dataset as the input to the network. For the time-lapse videos, each frame serves as a test input as shown in Figure 1 (a).

Error metric. We characterize the performance of our approach on both reflectance chromaticity and the separated images. We adopt the ℓ_1 error to quantitative measure the performance of the reflectance chromaticity. To evaluate the performance of the separated results, we compute the error for the separated result against the ground truth as:

$$\text{Loss} = \min\{E_{I_{1,1}} + E_{I_{2,2}}, E_{I_{1,2}} + E_{I_{2,1}}\} \qquad (11)$$

where E denote the ℓ_1 error between two images. We use a global scale-invariant loss because we are most interested in capturing relative variations between the two images.

5.2. Quantitative results on synthetic benchmark

In Table 2, we report the performance of our approach, and compare it to several baselines (summarized in Table 1). We begin by quantifying the importance of supervision. We train different models for our network: with full supervision, with supervision only on the quality of the final separated images (**Final-Only**), and training only the first sub-network, i.e., ChromNet, with supervision only on reflectance chromaticities (**Chrom-Only**). Moreover, for our fully supervised model (**Full**), we consider using the separated images directly predicted by our full network (**Full-Direct**), as well as taking only the reflectance chromaticity estimates and using Hui et al.'s algorithm [25]—which includes more complex processing—to perform the separation (**Full+[25]**). For the model with only chromaticity supervision, we also use [25] perform separation, and for the final-only supervised model (where intermediate chromaticities are not meaningful), we only consider the final output.

Methods	Chromaticity	Separated Images
Proposed		
Chrom-Only	**0.0308**	0.0398
Final-Only	—	0.0351
Full-Direct	0.0537	0.0288
Full+[25]	0.0537	**0.0207**
SingleNet	—	0.0679
Shen et al. [38]	0.0821	0.0791
Bell et al. [5]	0.0785	0.0763
Li et al. [32]	0.0833	0.0821
†Hsu et al. [23]	—	0.0678
†Hui et al. [25]	—	0.0101

† Use additional information as input.

Table 2. We measure performance of versions of our network—trained with different kinds of supervision, and with different approaches to perform separation—as well as other baselines. Reported here are ℓ_1 error values for both estimated reflectance chromaticity (when available), as well as the final separated images.

We find that our model trained with full supervision has the best performance in terms of the quality of final separated images. Interestingly, the **Chrom-Only** model is better at predicting chromaticity, but as expected, this does not translate to higher quality image outputs. The **Final-Only** model also yields worse separation results despite being trained with respect to their quality, highlighting the importance of intermediate supervision. Finally, we find that using our **Full** model in combination with [25] yields comparatively better results than taking the direct final output of the network. Thus, our final sub-networks (ShadingNet and SeparateNet) are able to only approximate [25]'s algorithm. Thus, their main benefit in our framework is in allowing back-propagation to provide supervision for chromaticity estimation, in a manner that is optimal for separation.

We also include comparisons to a network with a more traditional architecture (rather than three sub-networks) to

(a) Input (b) SingleNet (c) Final-Only

(d) Chrom-Only (e) Full-direct (f) Full+[25]

Figure 4. Qualitative comparison of image separation results of different versions of our network, as well as of the single encoder-decoder architecture network (SingleNet). We see that both SingleNet (b) and our Final-only (c) model both fail to separate the effects of illuminant shading from the input. Our Chrom-Only model (d) yields a better result, but has severe artifacts in certain regions—highlighting that better chromaticity estimates do not lead to better separation. The results from our model with Full supervision yields the best results—with better separation of shadow and shading effects when we use its chromaticity outputs in conjunction with [25].

do direct separation (**SingleNet**). We use the same architecture as the encoder-decoder portion of our ShadingNet, and train this again with supervision only on the final separated outputs. We find that this performs significantly worse (than even **Final-Only**), illustrating the utility of our physically-motivated architecture. Finally, we also include the comparisons with baselines where different intrinsic image decomposition methods [38, 5, 32] are used to estimate reflectance chromaticity from a single image, and these are used for separation with [25]. We find these methods yield lower accuracy in both reflectance chromaticity estimation and lighting separation—likely because they, like most intrinsic image methods, assume a single light source.

Finally, we evaluate on two methods that require additional information beyond a single image: ground truth light colors for Hsu et al. [23], and a flash/no-flash pair which provides direct access to reflectance chromaticity, for Hui et al. [25]. We produce better results than [23], but as expected, [25] yields the most accurate separation, since it has direct access to chromaticity information—but requires capturing an additional flash image.

5.3. Qualitative evaluation on real data

Figure 4 shows results on a real image for the different versions of our network (as well as of **SingleNet**), while Figure 5 compares our results comparison to Hui et al.'s method [25] when using a flash/no-flash pair. These results confirm our conclusions from Table 2—we find that the version of our network trained with full supervision performs best, especially when used in combination with [25] to carry out the separation from predicted chromaticities. Moreover, despite requiring only a single image input, it comes close to matching Hui et al.'s [25] performance with a flash/no-flash pair. We show an example in Figure 6 where our method affords a distinct advantage even when an image with flash is available, but when several regions in the scene are too far from the flash. This leads to artifacts in those regions for [25], while our approach is able to perform a higher quality separation.

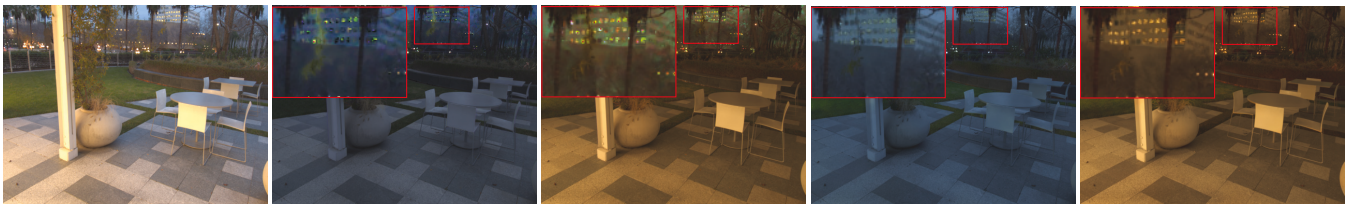
6. Conclusions

We describe a learning-based approach to separate the lighting effect of two illuminants in an image. Our method relies on the use of a deep-neural network based estimator, whose architecture is motivated by a physics-based analysis of the problem and associated intermediate supervision. Our ablation experiments demonstrate the importance of this supervision. Crucially, we show that we are able to produce high-quality outputs that match the performance of previous methods that required a flash/no-flash pair, while being more practical in requiring only a single image.

(a) Input photographs

(b) Hui et al. [25] (c) Our results

Figure 5. We evaluate our technique against the flash photography technique by Hui et al [25]. While the proposed method may lead to small artifacts in the resulting image, we can achieve nearly the same visual quality as Hui et al. [25], which captures two photographs for the same scene. In comparison, by using a single photograph, our proposed technique yields a more practical solution to the problem.



(a) Input (b) Hui et al. [25] (c) Our result

Figure 6. In this outdoor scene (a), the camera flash was not strong enough to illuminate the distant scene points, resulting in the artifacts in results using the flash-based method of Hui et al. [25] (b). In contrast, our method takes a single photograph and does not rely on flash illumination. As can be seen (c), the artifacts can be eliminated and visual quality has been significantly improved.

Acknowledgments

This research was supported, in part, by the National Geospatial-Intelligence Agency's Academic Research Program (Award No. HM0476-17-1-2000), the NSF CAREER grant CCF-1652569, and a gift from Adobe Research. Chakrabarti was supported by the NSF Grant IIS-1820693.

References

[1] Yağız Aksoy, Changil Kim, Petr Kellnhofer, Sylvain Paris, Mohamed Elgharib, Marc Pollefeys, and Wojciech Matusik. A dataset of flash and ambient illumination pairs from the crowd. In *ECCV*, 2018. 4

[2] Jonathan T Barron and Jitendra Malik. Color constancy, intrinsic images, and shape estimation. In *ECCV*. 2012. 2

[3] Jonathan T Barron and Jitendra Malik. Shape, illumination, and reflectance from shading. *PAMI*, 37(8):1670–1687, 2015. 2

[4] H. Barrow and J. Tenenbaum. Recovering intrinsic scene characteristics from images. *Computer Vision Systems*, 1978. 2

[5] Sean Bell, Kavita Bala, and Noah Snavely. Intrinsic images in the wild. *TOG*, 33(4):159, 2014. 2, 6, 7

[6] Nicolas Bonneel, Balazs Kovacs, Sylvain Paris, and Kavita Bala. Intrinsic decompositions for image editing. In *Computer Graphics Forum*, 2017. 5

[7] Nicolas Bonneel, Kalyan Sunkavalli, James Tompkin, Deqing Sun, Sylvain Paris, and Hanspeter Pfister. Interactive intrinsic video editing. *TOG*, 33(6):197, 2014. 1

[8] Adrien Bousseau, Sylvain Paris, and Frédo Durand. User-assisted intrinsic images. In *TOG*, volume 28, page 130, 2009. 1

[9] Ivaylo Boyadzhiev, Kavita Bala, Sylvain Paris, and Frédo Durand. User-guided white balance for mixed lighting conditions. *TOG*, 31(6):200, 2012. 1

[10] Ivaylo Boyadzhiev, Sylvain Paris, and Kavita Bala. User-assisted image compositing for photographic lighting. *TOG*, 32(4):36–1, 2013. 2

[11] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *SIGGRAPH 2008 classes*, page 32, 2008. 1

[12] Paul Debevec. The Light Stages and Their Applications to Photoreal Digital Actors. In *SIGGRAPH Asia*, 2012. 1

[13] Marc Ebner. Color constancy using local color shifts. In *ECCV*, 2004. 2

[14] Graham Finlayson, MS Drew, and BV Funt. Enhancing von kries adaptation via sensor transformations. 1993. 2

[15] Graham D Finlayson, Mark S Drew, and Brian V Funt. Diagonal transforms suffice for color constancy. In *ICCV*, 1993. 2

[16] Marc-Andre Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagn, and Jean-Franois Lalonde. Learning to predict indoor illumination from a single image. *TOG*, 9(4), 2017. 2

[17] Peter Vincent Gehler, Carsten Rother, Andrew Blake, Tom Minka, and Toby Sharp. Bayesian color constancy revisited. In *CVPR*, 2008. 2

[18] A. Gijsenij, T. Gevers, and J. van de Weijer. Computational color constancy: Survey and experiments. *TIP*, 20(9):2475–2489, 2011. 2

[19] Arjan Gijsenij, Rui Lu, and Theo Gevers. Color constancy for multiple light sources. *TIP*, 21(2):697–707, 2012. 2

[20] Roger Grosse, Micah K Johnson, Edward H Adelson, and William T Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *CVPR*, 2009.

[21] Daniel Hauagge, Scott Wehrwein, Kavita Bala, and Noah Snavely. Photometric ambient occlusion. In *CVPR*, 2013. 2

[22] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-François Lalonde. Deep outdoor illumination estimation. In *CVPR*, 2017. 2

[23] Eugene Hsu, Tom Mertens, Sylvain Paris, Shai Avidan, and Fredo Durand. Light mixture estimation for spatially varying white balance. In *TOG*, volume 27, page 70, 2008. 2, 6, 7

[24] Zhuo Hui, Aswin C. Sankaranarayanan, Kalyan Sunkavalli, and Sunil Hadap. White balance under mixed illumination using flash photography. In *ICCP*, 2016. 1

[25] Zhuo Hui, Kalyan Sunkavalli, Sunil Hadap, and Aswin C. Sankaranarayanan. Illuminant spectra-based source separation using flash photography. In *CVPR*, 2018. 1, 2, 3, 5, 6, 7, 8

[26] Zhuo Hui, Kalyan Sunkavalli, Joon-Young Lee, Sunil Hadap, Jian Wang, and Aswin C. Sankaranarayanan. Reflectance capture using univariate sampling of brdfs. In *ICCV*, 2017. 2

[27] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. *CVPR*, 2017. 5

[28] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 4

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[30] Pierre-Yves Laffont and Jean-Charles Bazin. Intrinsic decomposition of image sequences from local temporal variations. In *ICCV*, 2015. 2

[31] Jean-François Lalonde, Srinivasa G Narasimhan, and Alexei A Efros. What do the sun and the sky tell us about the camera? *IJCV*, 88(1):24–51, 2010. 2

[32] Zhengqi Li and Noah Snavely. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In *ECCV*, 2018. 1, 2, 4, 5, 6, 7

[33] Zhengqi Li and Noah Snavely. Learning intrinsic image decomposition from watching the world. In *CVPR*, 2018. 2

[34] Zhengqin Li, Kalyan Sunkavalli, and Manmohan Chandraker. Materials for masses: Svbrdf acquisition with a single mobile phone image. In *ECCV*, 2018. 1

[35] Veronique Prinet, Dani Lischinski, and Michael Werman. Illuminant chromaticity from image sequences. In *ICCV*, 2013. 2

[36] Konstantinos Rematas, Tobias Ritschel, Mario Fritz, Efstratios Gavves, and Tinne Tuytelaars. Deep reflectance maps. In *CVPR*, 2016. 1

[37] Christian Riess, Eva Eibenberger, and Elli Angelopoulou. Illuminant color estimation for real-world mixed-illuminant scenes. In *ICCVW*, 2011. 2

[38] Jianbing Shen, Xiaoshan Yang, Xuelong Li, and Yunde Jia. Intrinsic image decomposition using optimization and user

scribbles. *IEEE Transactions on Cybernetics*, 43(2):425–436, 2013. 2, 6, 7

[39] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. *CVPR*, 2017. 4

[40] Kalyan Sunkavalli, Fabiano Romeiro, Wojciech Matusik, Todd Zickler, and Hanspeter Pfister. What do color changes reveal about an outdoor scene? In *CVPR*, 2008. 2

[41] Yichuan Tang, Ruslan Salakhutdinov, and Geoffrey Hinton. Deep lambertian networks. *arXiv preprint arXiv:1206.6445*, 2012. 1

[42] Qi Zhao, Ping Tan, Qiang Dai, Li Shen, Enhua Wu, and Stephen Lin. A closed-form solution to retinex with nonlocal texture constraints. *PAMI*, 34(7):1437–1444, 2012. 2

[43] Tinghui Zhou, Philipp Krahenbuhl, and Alexei A Efros. Learning data-driven reflectance priors for intrinsic image decomposition. In *ICCV*, 2015. 1, 2