Single Image Intrinsic Decomposition with Discriminative Feature Encoding

Zongji Wang[1,2] and Feng Lu[1,2,*]
[1]State Key Laboratory of VR Technology and Systems, SCSE, Beihang University, Beijing, China
[2]Peng Cheng Laboratory, Shenzhen, China
{wzjgintoki, lufeng}@buaa.edu.cn

Abstract

Intrinsic image decomposition is an important and long-standing computer vision problem. Given a single input image, recovering the physical scene properties is ill-posed. In this work, we take the advantage of deep learning, which is proven to be highly efficient in solving the challenging computer vision problems including intrinsic image decomposition. Our focus lies in the feature encoding phase to extract discriminative features for different intrinsic layers from a single input image. To achieve this goal, we explore the distinctive characteristics between different intrinsic components in the high dimensional feature embedding space. We propose a feature divergence loss to force their high-dimensional embedding feature vectors to be separated efficiently. The feature distributions are also constrained to fit the real ones. In addition, we provide an approach to remove the data inconsistency in the MPI Sintel dataset, making it more proper for intrinsic image decomposition. Experimental results indicate that the proposed network structure is able to outperform the state-of-the-art methods.

1. Introduction

In terms of intrinsic image decomposition, the albedo image A indicates the surface material's reflectivity which is invariable under different illumination conditions, while the shading image S accounts for the illumination effects due to object geometry and camera viewpoint [3]. It is an ill-posed problem to reconstruct the two intrinsic images from a single color image I, with the formation model:

$$I = A \cdot S. \tag{1}$$

To solve this challenging inverse image formation problem, many researchers tried to apply physically-motivated priors as constraints to disambiguate decompositions [18, 26, 30, 29, 37, 2, 7, 28]. These methods usually adopt the priors in form of energy terms and solve the decomposition problem through graph-based inference algorithms. With the surge of ground truth intrinsic decompositions [14, 8, 4], data-driven deep learning methods [24, 33, 31, 3, 12, 21] have achieved promising decomposition results and have been drawing more and more research interest. However, fully-supervised methods require high-quality and densely-labelled decompositions, which are expensive to acquire. To overcome this problem, methods training across different datasets [21], training on synthetic datasets [31, 21], adding additional constraints [12] and reusing physically-motivated priors [3] have been proposed.

When developing their specific deep learning techniques, previous methods usually extract features via a shared encoder, and then use different decoders to disentangle information for specific intrinsic layers. Observing the different distributions between albedo and shading in gradient domain [18], it is natural to assume that features representing different intrinsic layers can be separated in the embedding space. With the features separated in the encoding phase, decoders can be released from distilling clues for specific targets and focus on the reconstruction procedure. This idea motivates our research in this paper.

We propose a novel two-stream encoder-decoder network for intrinsic image decomposition. In particular, the feature divergence loss is designed to encourage the two encoders to extract distinctive features for different intrinsic layers. The feature distribution constraint is used to encourage the features of a reconstructed intrinsic layer to have similar distribution pattern with the ground truth decomposition. Moreover, we provide an approach to deal with the illumination inconsistency between the ground truth shading and input images in the MPI Sintel dataset, making it more suitable for intrinsic image decomposition.

Our contributions can be summarized as follows:

1) A discriminative feature encoding approach consisting of the feature divergence loss and the feature distribution constraint is proposed.

2) A novel two-stream encoder-decoder network for intrinsic image decomposition is proposed.

3) A data refinement algorithm is provided for the

*Corresponding author: Feng Lu.

MPI Sintel dataset to produce a more physically consistent dataset that better fits the intrinsic decomposition task.

4) Experimental results on various datasets demonstrate the effectiveness of our proposed method. An ablation study is also conducted to validate our design for discriminative feature encoding.

2. Related work

Intrinsic image decomposition is a long standing computer vision problem. However, it is seriously ill-posed to recover an albedo layer and a shading layer from a single color image [31]. In the recent decades, much effort has been devoted to this challenging problem. These approaches can be coarsely classified into optimization-based methods using physically-motivated priors, and deep learning based data-driven methods [6, 31]. There are also approaches using multiple images as inputs [36, 23, 17, 22], treating the reflectance as a constant factor under variant illuminations. Depth cues are also taken into account in some works [1, 10, 19, 16]. In this section, we focus on the works using a single RGB image as input.

Physically-motivated Priors. To solve this ill-posed intrinsic decomposition problem, researchers have derived several physics-inspired priors to constrain the solution space [31]. Land et al. [18] proposed the Retinex algorithm, exploring the different properties of intrinsic components in gradient domain (large derivatives are perceived as changes in reflectance properties, while smoother variations are seen as changes in illumination). Based on this assumption, many priors for intrinsic image decomposition have been explored. Derived from the piece-wise constant property, reflectance sparsity [26, 30] and low-rank reflectances [7] are used as constraints. There are other constrains such as the distribution difference in gradient domain [5, 20], non-local texture [29, 37], shape and illumination [2], and user strokes [7, 28]. These hand-crafted priors are not likely to hold on complex datasets [6].

Deep learning Methods. Thanks to the publicly available intrinsic image datasets including the MIT intrinsic [14], the MPI Sintel [8] and the IIW [4], there has been a surge of applying deep learning to intrinsic decomposition [34, 24, 38, 39, 25]. Direct Intrinsics [33] is the first entirely deep learning model that directly outputs the albedo and shading layers given a color image. Results yielded by this method are blurry due to down-samplings in encoding phase and deconvolutions in decoding phase. Facing the fact that high-quality and densely-labelled intrinsic images are expensive to acquire, many methods have been developed to train models with additional constraints [12], reusing physically-motivated priors [3], expanding the dataset with synthetic images [31, 21] and training across datasets [21]. Fan et al. [12] provided a network structure using domain filter between the edges in guidance map to encourage the reflectance piece-wise constancy. Baslamisli et al. [3] presented a two-stage framework to firstly split the image gradients into albedo and shading components, which are then fed into decoders to predict pixel-wise intrinsic values. Shi et al. [31] trained a model to learn albedo, shading and specular images on a large-scale object-level synthetic dataset by rendering ShapeNet [9]. Li et al. [21] presented an end-to-end learning approach that learns better intrinsic image decompositions by leveraging datasets with different types of labels. In contrast to these works, we try to exploit the difference between intrinsic components in feature space.

3. Method

3.1. Network structure

Our full network architecture is visualized in Figure 1. The framework consists of two streams of encoder-decoder sub-networks. One is for albedo image prediction, and the other is for shading image. Taking the albedo stream for example, the input image is passed through the convolutional encoder to extract multi-level features, which are then aggregated by $(upsample, concatenate, convolution)$ sequences. In the decoding phase, the fused multi-scale features are fed into the sequence of three residual dilated blocks to reconstruct the albedo intrinsic image. The structure of the shading stream is the same as the albedo one. In practice, we adopt VGG-19 [32] pretrained on ImageNet [11] as the initialized encoder.

Previous works usually use a shared encoder to extract features containing both albedo and shading information. Then different decoders are applied to distill clues from the comprehensive features for specific intrinsic image prediction. The 'Y' shaped framework can be formulated as:

$$\begin{aligned} A &= g(f(I;\ \Theta);\ \Omega_a) = g_a \circ f(I), \\ S &= g(f(I;\ \Theta);\ \Omega_s) = g_s \circ f(I), \end{aligned} \tag{2}$$

where $f(\cdot;\ \Theta)$ and $g(\cdot;\ \Omega)$ denote the feature encoder and decoder respectively. Θ and Ω represent the corresponding trainable parameters.

Different from the above methods, our designed network is composed of two encoders for albedo and shading images respectively. In this paper, we denote this structure as 'X' shaped framework:

$$\begin{aligned} A &= g(f(I;\ \Theta_a);\ \Omega_a) = g_a \circ f_a(I), \\ S &= g(f(I;\ \Theta_s);\ \Omega_s) = g_s \circ f_s(I). \end{aligned} \tag{3}$$

Through this framework, the encoders ($f_a(\cdot)$, $f_s(\cdot)$) are able to extract features more pertinent to their reconstruction targets (albedo, shading). In Figure 2, we visualize the feature distributions of different network structures, which explains our idea in a more vivid way.

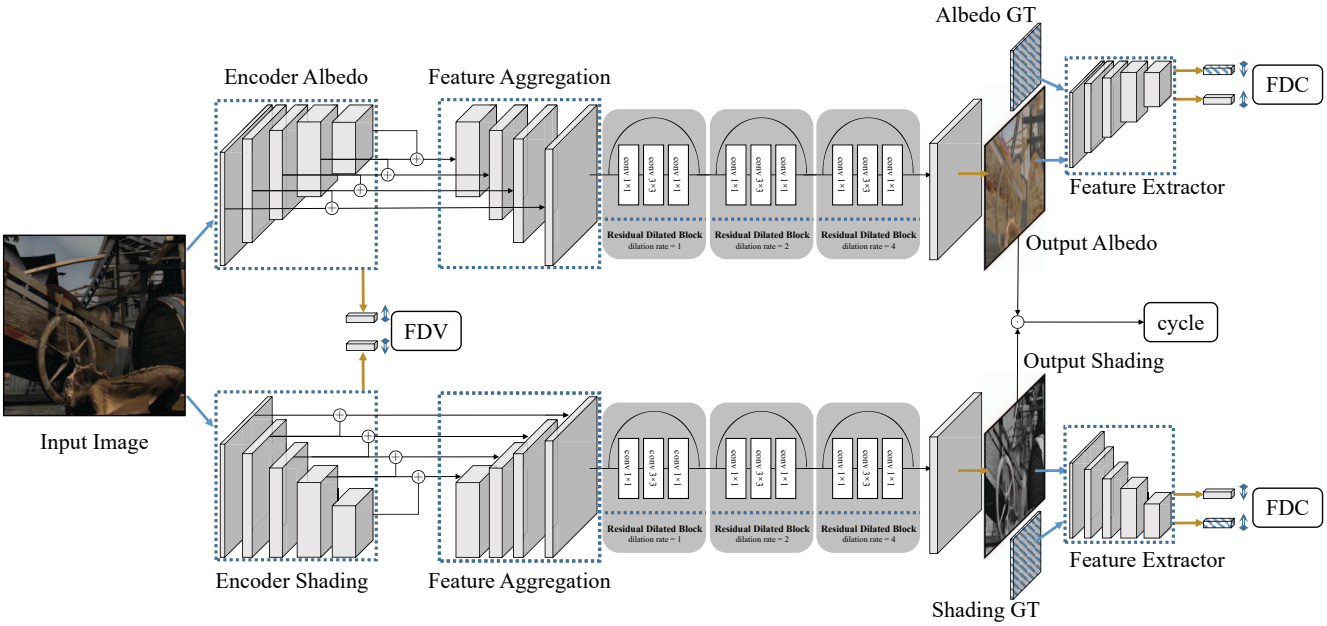
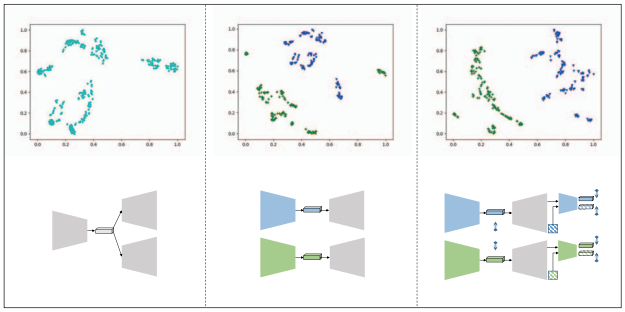


Figure 1. Framework of our two-stream intrinsic image decomposition network. The input image is passed through two streams of sub-network for albedo and shading image reconstructions respectively. We use the extractor in VGG-19 as the encoder structure, which is used to extract multi-scale feature maps. These feature maps are then aggregated by $(upsampling, concatenation, convolution)$ sequences. Finally, three residual dilated blocks are used as decoder to reconstruct intrinsic images from the fused feature maps. $\oplus$ represents the feature aggregation operation described above. $\odot$ denotes element-wise multiplication. The rounded boxes represent loss computations, in which 'cycle' means the cycle loss, 'FDC' means the feature distribution constraint and 'FDV' means the feature divergence loss.



(a) Shared encoder (b) Two encoders (c) Our structure

Figure 2. Feature distributions of different network structures. In each column, the top is the features visualized by t-SNE, and the bottom is the simplified network structure.

In the rest of this section, the core idea and design details of the discriminative feature encoding are introduced. Then, important constraints for our intrinsic decomposition network are explained.

3.2. Discriminative feature encoding

Our work is inspired by Land et al. [18]. Based on the Retinex assumption, albedo and shading layers possess different properties in gradient domain. By utilizing such discriminative properties, the intrinsic decomposition performance can be improved.

In this work, we try to study and exploit the discriminative properties in the more general convolutional feature space. In the following, we describe our proposed discriminative feature encoding detailedly.

Feature divergence loss. As shown in Figure 1, the encoding phase consists of multiple $(convolutions, relu, maxpooling)$ blocks, through which the input signal is encoded into several different abstraction levels. The multi-scale features are denoted as $\{\boldsymbol{f}^{\boldsymbol{E_1}}, \boldsymbol{f}^{\boldsymbol{E_2}}, \ldots, \boldsymbol{f}^{\boldsymbol{E_i}}, \ldots, \boldsymbol{f}^{\boldsymbol{E_n}}\}$, in which $\boldsymbol{f}^{\boldsymbol{E_i}}$ represents the output feature of the i_{th} block. We define the feature distance function as $d : \mathbb{R}^{m\times n\times c} \times \mathbb{R}^{m\times n\times c} \mapsto \mathbb{R}$, where c denotes the feature channel number and $m \times n$ is the spatial size of the input signal:

$$
\begin{aligned}
&d_{cos}(\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}, \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}) = \frac{1}{N_i}\sum_{\forall(x,y)}\left(\frac{<\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}(x,y), \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}(x,y)>}{||\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}(x,y)||_2 \cdot ||\boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}(x,y)||_2}\right)^2, \\
&d_{L_1}(\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}, \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}) = h(||\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}} - \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}||_1), \\
&d(\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}, \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}) = \alpha \cdot d_{cos}(\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}, \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}) + \beta \cdot d_{L_1}(\boldsymbol{f}_{\boldsymbol{a}}^{\boldsymbol{E_i}}, \boldsymbol{f}_{\boldsymbol{s}}^{\boldsymbol{E_i}}).
\end{aligned}
\tag{4}
$$

We design the feature distance measurement based on the $cosine$ and L_1 $norm$ between two vectors. In Eqn.4, $\boldsymbol{f_a}$

and $\boldsymbol{f_s}$ represent features from the albedo encoder and the shading encoder respectively. $< \cdot, \cdot >$ is the inner product in Euclidean space; $N_i = m_i \times n_i$; and (x, y) represents a spatial location in feature maps. $h(\cdot)$ is a distance rescale function modified from Sigmoid function to make $d_{L_1} \in (0, 1)$. We use $h(d) = 1 - \frac{1}{1+exp(-(d-1.2 \cdot exp(1.2))/1.2^2)}$.

Then, the feature divergence loss $\mathcal{L}_{fdv}$ is formulated as:

$$\mathcal{L}_{fdv} = \sum_{i}^{n} \omega_i \cdot d(\boldsymbol{f_a^{E_i}}, \boldsymbol{f_s^{E_i}}), \tag{5}$$

where $\omega_i \geq 0$ is the weight for the feature distance from abstraction level i. Empirically, we extract five different levels of abstraction in experiments ($n = 5$). We set $\omega_{[1,2,3,4,5]} = [0.01, 0.1, 0.5, 0.7, 1.0]$ and $\alpha = 0.3,\ \beta = 0.1$.

Feature distribution constraint. Feature divergence loss is to increase the distance between the feature vectors embedded by different encoders. However, this is not sufficient for discriminative feature encoding. Note that the core idea of Fisher's linear discriminant is to maximize the distance between classes and minimize the distance within classes simultaneously. As an analogue of that, along with the feature divergence loss described above, we use the feature perceptual loss [15] between the predicted and ground truth intrinsic images to constrain the encoding process, encouraging the embedded features to fit the real distribution.

We use the same distance measurement as in the feature divergence loss. $d(\boldsymbol{f_{pred}^{E_i}}, \boldsymbol{f_{real}^{E_i}})$ denotes the feature distance in the i_{th} abstraction level.

The feature distribution constraint $\mathcal{L}_{fdc}$ is formulated as:

$$\begin{aligned}\mathcal{L}_{fdc} = \sum_{i}^{n} \gamma_i ((1 - d(\boldsymbol{f_{pred,a}^{E_i}}, \boldsymbol{f_{real,a}^{E_i}})) + \\ (1 - d(\boldsymbol{f_{pred,s}^{E_i}}, \boldsymbol{f_{real,s}^{E_i}}))),\end{aligned} \tag{6}$$

where $\gamma_i \geq 0$ is the weight factor. Note that $(1 - d(\boldsymbol{f_{pred}}, \boldsymbol{f_{real}})) \in (0, 1)$ represents the feature similarity between $\boldsymbol{f_{pred}}$ and $\boldsymbol{f_{real}}$. Minimizing Eqn.6 encourages the predicted and ground truth intrinsic images to have similar perceptual features. In practice, the encoders are reused to extract features from the predicted and target results in our framework, by which the embedded feature distribution can be optimized directly during training. Empirically, we set $\gamma_{[1,2,3,4,5]} = [1.0,\ 1.0,\ 1.0,\ 1.0,\ 1.0]$ and $\alpha = 0.1,\ \beta = 0.9$.

3.3. Basic supervised constraints

Besides the above constraints for discriminative feature encoding, several basic supervised losses are adopted to train the intrinsic image decomposition network.

As described in Eqn.3, given an image I, the albedo image A and the shading image S are predicted through trained $g_a \circ f_a$ and $g_s \circ f_s$. With the densely-labelled intrinsic images $\widehat{A}$ and $\widehat{S}$ as the ground truth data, we constrain the pixel-wise predictions using the reconstruction loss $\mathcal{L}_{rec}$ and the gradient loss $\mathcal{L}_{grad}$.

Reconstruction loss. We use the L_1 loss $\mathcal{L}_{L_1}$ combined with the SSIM (the structural similarity index [35]) loss $\mathcal{L}_{SSIM}$ as the reconstruction loss:

$$\begin{aligned}\mathcal{L}_{rec} &= \lambda_{L_1}\mathcal{L}_{L_1} + \lambda_{SSIM}\mathcal{L}_{SSIM}, \\ \mathcal{L}_{L_1} &= ||A - \widehat{A}||_1 + ||S - \widehat{S}||_1 + ||A \cdot S - I||_1, \\ \mathcal{L}_{SSIM} &= (1 - SSIM(A, \widehat{A})) + (1 - SSIM(S, \widehat{S})) \\ &\quad + (1 - SSIM(A \cdot S, I)),\end{aligned} \tag{7}$$

in which $SSIM(\boldsymbol{x}, \boldsymbol{y})$ measures the structural similarity between image $\boldsymbol{x}$ and $\boldsymbol{y}$. Thus we define the SSIM loss as $(1 - SSIM(\boldsymbol{x}, \boldsymbol{y}))$, indicating the structural dissimilarity. Empirically, we set $\lambda_{L_1} = 30.0$ and $\lambda_{SSIM} = 0.5$. Note that the cycle loss is used to encourage the product of predicted A and S to be similar with the input image I.

Gradient loss. We also use the image gradients as an supervision to help preserve the details of intrinsic images:

$$\begin{aligned}\mathcal{L}_{grad} = ||\nabla_x A - \nabla_x \widehat{A}||_2^2 + ||\nabla_y A - \nabla_y \widehat{A}||_2^2 + \\ ||\nabla_x S - \nabla_x \widehat{S}||_2^2 + ||\nabla_y S - \nabla_y \widehat{S}||_2^2,\end{aligned} \tag{8}$$

in which $\nabla_{x\ or\ y}$ is the image gradient along x or y axis.

In datasets with ground truth decomposition results like the MIT intrinsic and the MPI Sintel, the total loss is constructed as:

$$\mathcal{L}_{total} = \lambda_1\mathcal{L}_{rec} + \lambda_2\mathcal{L}_{grad} + \lambda_3\mathcal{L}_{fdv} + \lambda_4\mathcal{L}_{fdc}. \tag{9}$$

Empirically, we set $\lambda_{[1,2,3,4]} = [1.0,\ 1.5,\ 0.1,\ 1.0]$.

Different from the densely-labelled datasets, the IIW dataset [4] only provides sparse annotations. Therefore, we use the ordinal loss to measure the difference between the predicted and target intrinsic images.

Ordinal loss. Since dense ground truth labels are not available, [4] introduced the weighted human disagreement rate (WHDR) as the error metric. Similar to [21], we use the ordinal loss based on WHDR as sparse supervision term. For each pair of annotated pixels (i, j) in the predicted albedo image A, we have the error function:

$$e_{i,j}(A) = \begin{cases} \omega_{i,j}(log\ A_i - log\ A_j)^2, & r_{i,j} = 0 \\ \omega_{i,j}(max(0, m - log\ A_i + log\ A_j))^2, & r_{i,j} = +1 \\ \omega_{i,j}(max(0, m - log\ A_j + log\ A_i))^2, & r_{i,j} = -1 \end{cases} \tag{10}$$

in which $r_{i,j}$ is the relative reflectance (albedo) judgements from the IIW. The label $r_{i,j} = [0, +1, -1]$ means that pixel i has $[the\ same, higher, lower]$ brightness level as/than pixel j.

Then, the ordinal loss $\mathcal{L}_{ord}$ is obtained by accumulating all the annotated pairs in the albedo image:

$$\mathcal{L}_{ord} = \sum_{(i,j)} e_{i,j}(A) \tag{11}$$

Besides the sparse supervision using the ordinal loss, we also adopt the same smoothness constraints as [21].

4. Intrinsic data refinement

The MPI Sintel [8] is a publicly-available densely-labelled dataset containing complex indoor and outdoor scenes. It is firstly designed for optical flow evaluation. For the research purpose of intrinsic image decomposition, the ground truth shading images have been rendered with a constant gray albedo considering illumination effects. However, due to the creation process, the original input frames can not be reconstructed from the ground truth albedo and shading layers through Eqn.1.

As shown in the first row of Figure 3, the specular component in the shading image can not be observed in the original image, which means they do not share the same illumination condition. Although the simplified image formation model Eqn.1 need not to be strictly respected, it is not physically correct to extract a shading layer depicting different illumination effects from the original image. To overcome this inconsistency, previous works [33] directly resynthesize original images I from the ground truth albedo A and shading S via Eqn.1. However, this approach does not deal with the specular component in the shading layer, which is considered not modeled well by Eqn.1 [31].

In this paper, we propose an approach to refine the dataset in order to shift it into a domain more representative of real images. The refined MPI Sintel dataset (MPI_RD) is subject to the image formation model Eqn.1, and the shading layers contain no color information (gray shading). In addition, the shading layers in the MPI_RD maintain the consistency with the original images. This can be shown in two aspects. On one hand, the specular component is removed from the shading layers. On the other hand, the shape details observed in the original images are preserved in the shading layers. We describe our data refinement algorithm in Alg.1. In summary, we shift the distribution of the albedo layer to a higher mean value, and then reconstruct the shading layer from the original image and the shifted albedo (step 2 to 6). After that, invalid pixels in the reconstructed shading layer are computed using Local Linear Embedding (LLE) [27] with the input I as the guided image, which is adopted to construct the embedding weights (step 7 to 8). Finally, the input image is resynthesized from the processed albedo and shading images (step 9).

Algorithm 1 Framework of data refinement for MPI Sintel.

Input: The original MPI Sintel dataset consisting of input images I, albedo images A and shading images S; $MPI = \{I, A, S\}$

Output: The refined MPI Sintel dataset $MPI_{refined} = \{I^*, A^*, S^* | I^* = A^* \cdot S^*\}$ under the constraint of intrinsic decomposition model Eqn.1;

1: **for** each $i \in [1, N]$ **do**
2: convert the RGB images into $L^*a^*b^*$ space, and extract the L channel as $\{I_i, A_i, S_i\}$;
3: reconstruct the albedo and shading: $\hat{A}_i = I_i/S_i$, $\hat{S}_i = I_i/A_i$;
4: compute the valid mask for $\hat{A}_i$: $M_i = (0 < \hat{S}_i < 1)$ & $(0 < \hat{A}_i < 1)$;
5: compute the statistics of valid pixels indexed by M_i from $\hat{A}_i$: $(\hat{\mu}, \hat{\sigma})$;
6: shift the distribution of A_i μ, σ to the reconstructed valid statistics: $\tilde{A}_i = \frac{A_i - \mu}{\sigma} \cdot \hat{\sigma} + \hat{\mu}$;
7: reconstruct shading from $\tilde{A}_i$: $\tilde{S}_i = I_i/\tilde{A}_i$;
8: reconstruct the invalid pixels in $\tilde{S}_i$ with the help of I_i: S_i^*;
9: convert $\tilde{A}_i$ into RGB color space: A_i^*, and reconstruct $I_i^* = A_i^* \cdot S_i^*$;
10: **end for**

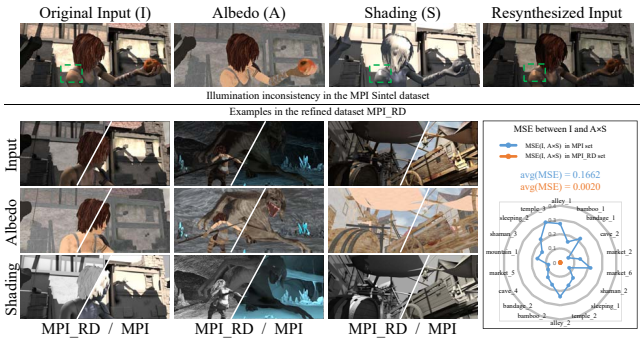


Figure 3. Comparison between the refined MPI Sintel dataset (MPI_RD) and the original MPI Sintel dataset (MPI). Top is a sample for illumination inconsistency in the MPI. Bottom is the illustration for the MPI_RD. In the bottom, each image is split into two parts. The left shows the refined data, and the right is the original data. The shading images in the MPI_RD preserve more geometric details and exclude specular components.

5. Experimental results

5.1. Datasets

5.1.1 MPI Sintel Dataset and our refined version

Sintel is an open source 3D animated short film, which has been published in many formats for various research purposes. For intrinsic image decomposition, the "clean pass" images and the corresponding albedo and shading layers have been published as the "MPI Sintel dataset", containing 18 sequences for a total of 890 frames. As discussed in Section 4, there is severe illumination inconsistency between the input frames and the shading layers in this dataset. Therefore, we provide the refined MPI Sintel dataset as a more proper dataset for intrinsic image decomposition.

In the bottom of Figure 3, we demonstrate the comparison between our refined MPI dataset (MPI_RD) and the original MPI dataset (MPI). In the shading layer of the first column, we can see that in our refined shading image, the specular on the shoulder of the girl is removed, making the shading illumination consistent with the original input im-

age. In the second and third columns, the shading layers from the MPI_RD dataset contain more geometric details than those from the MPI dataset. For instance, the wooden cart's coarse surface is depicted in the refined shading in the third column, while the original shading from the MPI dataset only has smooth surface. These examples demonstrate that our refined MPI_RD ensures the consistency between the intrinsic decompositions and the input image. In the rightmost column in the bottom part of Figure 3, the mean squared error (MSE) between the input image I and the resynthesized image $A \times S$ is computed. The MSE value in the MPI_RD dataset is significantly smaller than that in the MPI dataset, showing that the intrinsic decomposition model Eqn.1 is well respected in the refined dataset.

Training details. For data augmentation, we randomly resize the input image by a scale factor in $[0.8,\ 1.3]$, and randomly crop a 288×288 patch from the resized image per iteration. We also use horizontal flipping in the training phase. To compare with the state-of-the-art methods, similar to [12], we evaluate our results on both a scene split and an image split. For a scene split, half of the scenes are used for training and the other half for testing. For an image split, all 890 images are randomly separated into two sets. Evaluation on a scene split is considered more challenging as it requires more generalization capacity.

5.1.2 IIW Dataset

Intrinsic Images in the Wild (IIW) [IIW-TOG 2014] is a large scale, public dataset for intrinsic image decomposition of real-world scenes. This dataset contains 5,230 real images of mostly indoor scenes, combined with a total of 872,161 crowd-sourced annotations of reflectance comparisons between pairs of points sparsely selected throughout the images (on average 100 judgements per image). Following many prior works [24, 25, 38, 12], we split the IIW dataset by placing the first of every five consecutive images sorted by the image ID into the test set, and the others into the training set. The WHDR from [4] is employed to measure the quality of the reconstructed albedo images.

Training details. As for the IIW dataset, our proposed network structure cannot be directly used due to the lack of dense labelling of albedo and shading layers. Actually, only sparse and relative reflectance annotations are provided. In order to take advantage of the proposed feature divergence loss and feature distribution constraint, we have to slightly modify the network. In detail, the predicted dense albedo is collected into an image pool to describe the distribution of albedo. The reconstructed shading using the original image and predicted albedo is used as the dense supervision for the shading prediction, and is also collected in an image pool to describe the shading distribution. We set the weights in Eqn.6 to be $\gamma_{[1,2,3,4,5]} = [0,\ 0,\ 0,\ 1.0,\ 1.0]$.

5.2. In comparison to state-of-the-art methods

5.2.1 On the MPI Sintel and the refined dataset

Table 1. Numerical results on the MPI Sintel dataset.

	Methods	MSE albedo	MSE shading	MSE avg	LMSE albedo	LMSE shading	LMSE avg	DSSIM albedo	DSSIM shading	DSSIM avg
image split	Retinex [14]	0.0606	0.0727	0.0667	0.0366	0.0419	0.0393	0.2270	0.2400	0.2335
	Barron et al. [2]	0.0420	0.0436	0.0428	0.0298	0.0264	0.0281	0.2100	0.2060	0.2080
	Chen et al. [10]	0.0307	0.0277	0.0292	0.0185	0.0190	0.0188	0.1960	0.1650	0.1805
	MSCR [33]	0.0100	0.0092	0.0096	0.0083	0.0085	0.0084	0.2014	0.1505	0.1760
	Revisiting [12]	0.0069	0.0059	0.0064	0.0044	0.0042	0.0043	0.1194	0.0822	0.1008
	Ours	**0.0047**	**0.0046**	**0.0047**	**0.0037**	**0.0038**	**0.0038**	**0.0950**	**0.0774**	**0.0862**
scene split	MSCR [33]	0.0190	0.0213	0.0201	0.0129	0.0141	0.0135	0.2056	0.1596	0.1826
	Revisiting [12]	0.0189	**0.0171**	**0.0180**	0.0122	**0.0117**	**0.0119**	0.1645	0.1450	0.1547
	Ours	**0.0173**	0.0195	0.0184	**0.0118**	0.0147	0.0133	**0.1587**	**0.1405**	**0.1496**

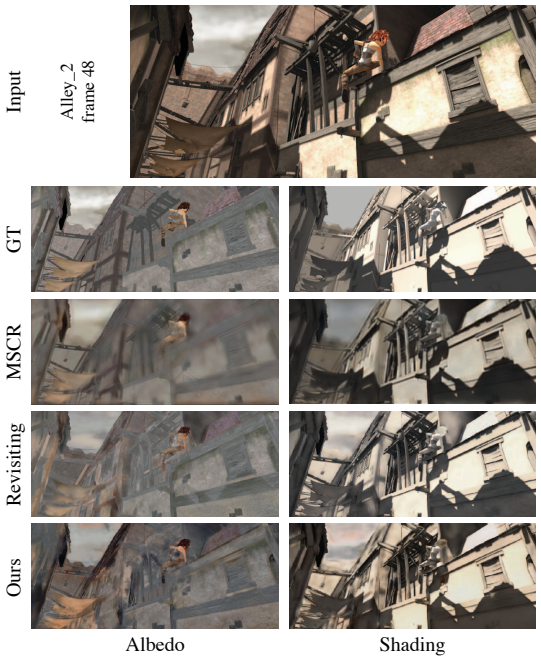


Figure 4. Qualitative comparison on the MPI Sintel dataset. The visual results are evaluated on the more challenging scene split.

As shown in Table 1, our method achieves the best result on the MPI Sintel dataset using the image split. On the more challenging scene split, our method is competitive with the state of the art, and achieves the best results for 5 out of 9 columns in the table. We show a group of qualitative results evaluated on the scene split in Figure 4. While the MSCR [33] results are relatively blurry due to the large kernel convolutions and down-sampling, our method provides sharper results comparable to Revisiting [12]. Moreover, our shading layer depicts better shadow area than [12].

As described in Section 4, the MPI Sintel dataset has issues of data consistency between the original input images and the corresponding shading images. Because of the proposed feature divergence loss, feature distribution constraint

and the use of the cycle loss, our method is sensitive to such data inconsistency. Therefore, we compare our method with the state-of-the-art methods on the more challenging scene split of the refined MPI Sintel dataset. As shown in Table 2, our method achieves the best result, which demonstrates the effectiveness of our method and data refinement process. To further validate the effectiveness of our proposed architecture, we also conduct an ablation study illustrated in the bottom of Table 2. We can observe that using only the feature divergence loss or the feature distribution constraint does not improve the performance much, while using both of them results in considerable performance gain.

Table 2. Numerical results on the Refined MPI Sintel dataset.

	MSE			LMSE			DSSIM		
Methods	albedo	shading	avg	albedo	shading	avg	albedo	shading	avg
MSCR [33]	0.0222	0.0175	0.0199	0.0151	0.0122	0.0136	0.1803	0.1619	0.1711
Revisiting [12]	0.0196	0.0137	0.0167	0.0146	0.0094	0.0120	0.1651	0.1082	0.1366
Ours plain	0.0172	0.0147	0.0159	0.0116	0.0097	0.0106	0.1528	0.1085	0.1307
Ours w/o FDV	0.0166	0.0134	0.0150	0.0112	0.0090	0.0101	0.1474	0.1048	0.1261
Ours w/o FDC	0.0170	0.0130	0.0150	0.0113	0.0089	0.0101	0.1530	0.1070	0.1300
Ours	**0.0157**	**0.0126**	**0.0142**	**0.0105**	**0.0087**	**0.0096**	**0.1419**	**0.1015**	**0.1217**

('Ours plain' is the basic two-stream network without FDV or FDC.)

In Figure 5, a side-by-side comparison with two other methods on the refined dataset MPI_RD is displayed. As shown, our method performs better at separating shading from albedo information. For example, in the bamboo scene, our method outputs consistent shadow on the bamboo under the girl's feet while other methods do not. Similar observations can be made around the girl's neck in the bandage scene, the monster's wing in the cave scene, and the girl's leg in the market scene.

Table 3. Numerical results on the MIT intrinsic dataset.

	MSE			LMSE
Methods	albedo	shading	avg	total
Barron et al. [2]	0.0064	0.0098	0.0081	0.0125
Zhou et al. [38]	0.0252	0.0229	0.0240	0.0319
Shi et al. [31]	0.0216	0.0135	0.0175	0.0271
MSCR [33]	0.0207	0.0124	0.0165	0.0239
Revisiting [12]	0.0134	**0.0089**	0.0111	0.0203
Ours	**0.0120**	0.0095	**0.0108**	**0.0170**

(Note that Barron et al.'s method [2] relies on specialized priors and masked objects particular to this dataset.)

We also experiment on the MIT intrinsic dataset [14], which consists of object-level real images. As in [12], we use the 220 images in the dataset. To compare with previous methods, the split from [2] is used. Our refined MPI Sintel dataset has gray scale shading images, thus we firstly pre-train the model on the MPI_RD and then fine-tune it on the MIT training set. The numerical results are shown in Table 3. Our method achieves the best results for most of the columns in the table. Qualitative results are illustrated in Figure 6. We can observe that our method predicts sharp and accurate intrinsic layers.

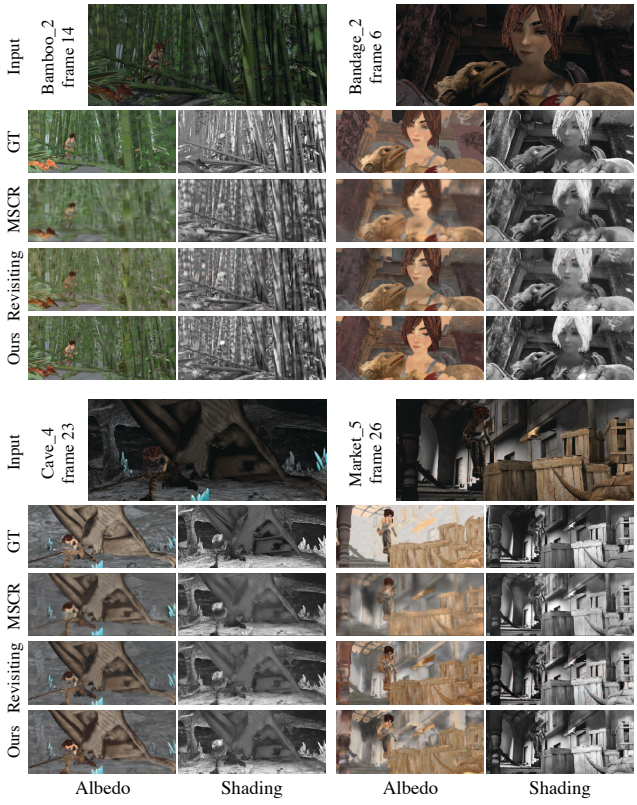


Figure 5. Qualitative comparison on our refined MPI Sintel dataset. The visual results are evaluated on the scene split. Our method is better at separating albedo and shading components.

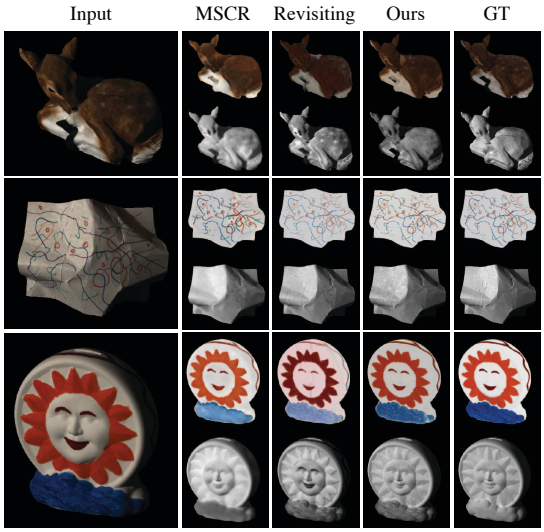


Figure 6. Qualitative comparison on the MIT intrinsic dataset.

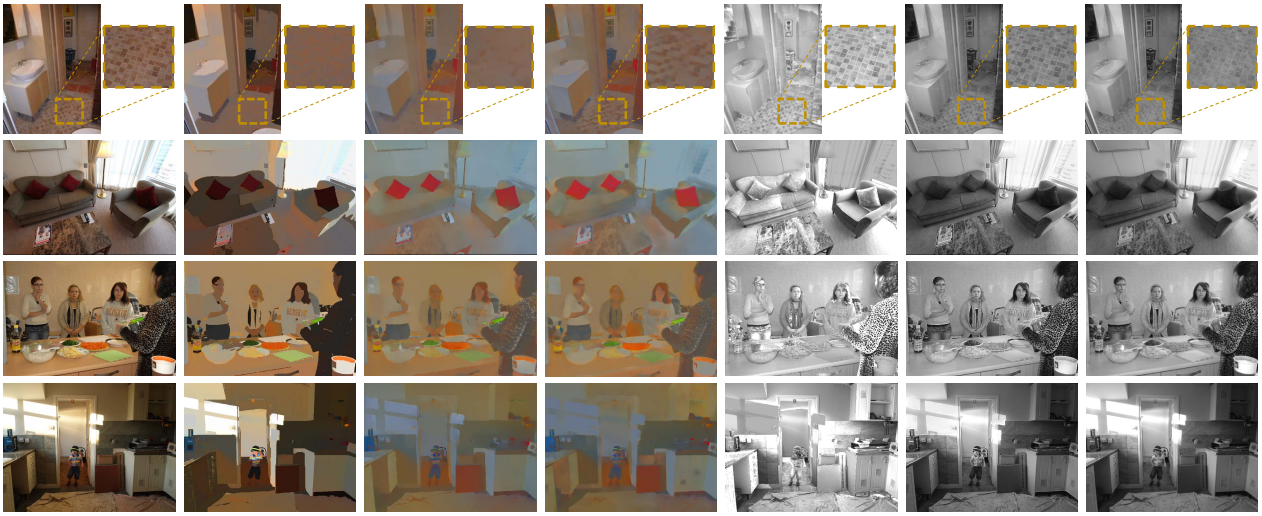


Figure 7. Qualitative comparison on the IIW dataset. The second to fourth columns are the albedo images, and the fifth to seventh columns are the shading layers.

5.2.2 On the IIW dataset

Table 4. Numerical results on the IIW test set.

Methods	WHDR(mean)
Baseline(const shading)	51.37
Baseline(const reflectance)	36.54
Shen et al. 2011 [30]	36.90
Retinex(color) [14]	26.89
Retinex(gray) [14]	26.84
Garces et al. 2012 [13]	25.46
Zhao et al. 2012 [37]	23.20
L_1 flattening [5]	20.94
Bell et al. 2014 [4]	20.64
Zhou et al. 2015 [38]	19.95
Nestmeyer et al. 2017(CNN) [25]	19.49
Zoran et al. 2015* [39]	17.85
Nestmeyer et al. 2017 [25]	17.69
Bi et al. 2015 [5]	17.67
CGIntrinsic [21]	14.80
Revisiting [12]	14.45
Ours	**13.90**

In Table 4, we report the numerical results evaluated on the test set of the IIW dataset. Our proposed method achieves the best performance with a mean WHDR value of 13.90%, which is a considerable improvement compared to the second best one [12] with a mean WHDR value of 14.45%. To better illustrate the performance of our method, we display groups of qualitative comparisons with the state-of-the-art methods in Figure 7. In the first row, the detailed intrinsic decomposition results are shown in the zoom-in windows. It can be observed that our method successfully preserves the texture of the floor tiles in the albedo layer, while the other approaches treat such texture as shading. In the second row, in a zoomed-in view, it can be noted that our albedo layer contains clearer contours for the magazine cover on the table. In the third row, the white block near the left edge of the image is decomposed properly with albedo consistency by our method. In the fourth row, the table in the left corner of the image is well separated from the box under the table in our albedo layer. These examples show that our method can extract better albedo and shading layers from original images, and preserve more detailed information in intrinsic decomposition.

6. Conclusion

In this paper, we present a novel two-stream encoder-decoder network for intrinsic image decomposition. Our method is able to exploit the discriminative properties of the features for different intrinsic images. Specifically, the feature divergence loss is designed to increase the distance between features corresponding to different intrinsic images, and the feature perceptual loss is applied to constrain the feature distribution. These two modules work together to encode discriminative features for intrinsic image decomposition. We provide an algorithm to refine the MPI Sintel dataset to make it more suitable for intrinsic image decomposition. The visual results in the MPI_RD and the more challenging IIW dataset demonstrate that our proposed method can achieve superior results with better albedo/shading separation.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (NSFC) under Grant 61972012 and Grant 61732016.

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

[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, 2014. 1, 2, 6, 7

[3] 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. 1, 2

[4] S. Bell, K. Bala, and N. Snavely. Intrinsic images in the wild. *ACM Transactions on Graphics (TOG)*, 33(4):159, 2014. 1, 2, 4, 6, 8

[5] S. Bi, X. Han, and Y. Yu. An l_1 image transform for edge-preserving smoothing and scene-level intrinsic decomposition. *ACM Trans. Graph. (Proc. SIGGRAPH)*, 34(4), 2015. 2, 8

[6] N. Bonneel, B. Kovacs, S. Paris, and K. Bala. Intrinsic decompositions for image editing. In *Computer Graphics Forum*, volume 36, pages 593–609. Wiley Online Library, 2017. 2

[7] A. Bousseau, S. Paris, and F. Durand. User-assisted intrinsic images. In *ACM Transactions on Graphics (TOG)*, volume 28, page 130. ACM, 2009. 1, 2

[8] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012. 1, 2, 5

[9] A. X. Chang, T. Funkhouser, L. Guibas, P. Hanrahan, Q. Huang, Z. Li, S. Savarese, M. Savva, S. Song, H. Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[10] 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. 2, 6

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[12] 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. 1, 2, 6, 7, 8

[13] E. Garces, A. Munoz, J. Lopez-Moreno, and D. Gutierrez. Intrinsic images by clustering. In *Computer graphics forum*, volume 31, pages 1415–1424. Wiley Online Library, 2012. 8

[14] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman. Ground truth dataset and baseline evaluations for intrinsic image algorithms. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2335–2342. IEEE, 2009. 1, 2, 6, 7, 8

[15] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 4

[16] 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. Springer, 2016. 2

[17] P.-Y. Laffont and J.-C. Bazin. Intrinsic decomposition of image sequences from local temporal variations. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 433–441, 2015. 2

[18] E. H. Land and J. J. McCann. Lightness and retinex theory. *Josa*, 61(1):1–11, 1971. 1, 2, 3

[19] K. J. Lee, Q. Zhao, X. Tong, M. Gong, S. Izadi, S. U. Lee, P. Tan, and S. Lin. Estimation of intrinsic image sequences from image+ depth video. In *European Conference on Computer Vision*, pages 327–340. Springer, 2012. 2

[20] Y. Li and M. S. Brown. Single image layer separation using relative smoothness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2752–2759, 2014. 2

[21] Z. Li and N. Snavely. Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 371–387, 2018. 1, 2, 4, 8

[22] 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. 2

[23] Y. Matsushita, S. Lin, S. B. Kang, and H.-Y. Shum. Estimating intrinsic images from image sequences with biased illumination. In *European Conference on Computer Vision*, pages 274–286. Springer, 2004. 2

[24] 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. 1, 2, 6

[25] 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. 2, 6, 8

[26] C. Rother, M. Kiefel, L. Zhang, B. Schölkopf, and P. V. Gehler. Recovering intrinsic images with a global sparsity prior on reflectance. In *Advances in neural information processing systems*, pages 765–773, 2011. 1, 2

[27] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000. 5

[28] J. Shen, X. Yang, X. Li, and Y. Jia. Intrinsic image decomposition using optimization and user scribbles. *IEEE transactions on cybernetics*, 43(2):425–436, 2013. 1, 2

[29] L. Shen, P. Tan, and S. Lin. Intrinsic image decomposition with non-local texture cues. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE, 2008. 1, 2

[30] L. Shen and C. Yeo. Intrinsic images decomposition using a local and global sparse representation of reflectance. In *CVPR 2011*, pages 697–704. IEEE, 2011. 1, 2, 8

[31] 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. 1, 2, 5, 7

[32] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 2

[33] M. M. Takuya Narihira and S. X. Yu. Direct intrinsics: Learning albedo-shading decomposition by convolutional regression. In *International Conference on Computer Vision (ICCV)*, 2015. 1, 2, 5, 6, 7

[34] Y. Tang, R. Salakhutdinov, and G. Hinton. Deep lambertian networks. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pages 1419–1426, USA, 2012. Omnipress. 2

[35] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 4

[36] Y. Weiss. Deriving intrinsic images from image sequences. In *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001*, volume 2, pages 68–75. IEEE, 2001. 2

[37] 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. 1, 2, 8

[38] 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. 2, 6, 7, 8

[39] 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. 2, 8