FLOWIID: SINGLE-STEP INTRINSIC IMAGE DECOMPOSITION VIA LATENT FLOW MATCHING

Mithlesh Singla, Seema Kumari, and Shanmuganathan Raman

Indian Institute of Technology Gandhinagar, India
{mithlesh.singla, seema.kumari, shanmuga}@iitgn.ac.in

ABSTRACT

Intrinsic Image Decomposition (IID) separates an image into albedo and shading components. It is a core step in many real-world applications, such as relighting and material editing. Existing IID models achieve good results, but often use a large number of parameters. This makes them costly to combine with other models in real-world settings. To address this problem, we propose a flow matching-based solution. For this, we design a novel architecture, FlowIID, based on latent flow matching. FlowIID combines a VAE-guided latent space with a flow matching module, enabling a stable decomposition of albedo and shading. FlowIID is not only parameter-efficient, but also produces results in a single inference step. Despite its compact design, FlowIID delivers competitive and superior results compared to existing models across various benchmarks. This makes it well-suited for deployment in resource-constrained and real-time vision applications.

Index Terms— Intrinsic image decomposition, albedo, shading, latent flow matching, and single-step generation.

1. INTRODUCTION

Intrinsic Image Decomposition (IID) aims to separate an input image I into its reflectance (albedo, A) and shading (S) under the Lambertian assumption [1]: $I = A \cdot S$. Albedo encodes material and color, while shading captures shape and illumination. IID supports tasks including relighting, material editing, object detection, and recognition, thus requiring both accuracy and efficiency in practice.

Existing IID methods adopt two paradigms: (i) separate estimation of A and S [2, 3], which often results in reconstruction inconsistencies; and (ii) direct shading prediction with albedo computed as $A = I/S$ [4, 5], which aligns the reconstruction and is popular in recent approaches.

Conventional IID models rely on deep convolutional networks that are computationally expensive and difficult to deploy. Diffusion-based methods [6, 7] achieve strong results but require slow multi-step inference and large parameter counts. To overcome these limitations, we propose a latent flow matching (LFM) framework that learns continuous transport in a compact latent space. Our architecture integrates a UNet with an encoder–decoder design, achieving efficient decomposition with substantially fewer parameters. This approach offers (i) deterministic training compared to stochastic diffusion, (ii) single-step, fast inference, and (iii) competitive accuracy with a compact model size. Our main contributions are as follows:

(1) We present the first application of LFM to intrinsic image decomposition, enabling efficient single-step prediction.

(2) We introduce a VAE-guided latent representation that stabilizes the decomposition process.

(3) We achieve superior performance on standard benchmarks while ensuring practicality for real-time and embedded deployment.

2. RELATED WORK

Before the deep learning era, intrinsic image decomposition (IID) methods often relied on additional cues such as depth information [8]. However, recent works no longer incorporate such auxiliary information or strong prior assumptions about the scene. With the introduction of large-scale datasets such as CGIntrinsics [3] and Hypersim [9], neural network-based approaches have become the dominant paradigm [10, 4].

Current state-of-the-art models typically rely on architectures with a large number of parameters. For instance, Careaga and Aksoy [4] employ a three encoder–decoder design, while Kocsis et al. [6] and $RGB \leftrightarrow X$ [7] fine-tune the pre-trained text-conditional Stable Diffusion V2 model [11] by treating intrinsic layers as multichannel images. Similarly, Luo et al. [12] leverage a diffusion model combined with ControlNet [13] to generate intrinsic modalities.

Although these models achieve state-of-the-art performance on benchmarks such as ARAP [14] and MIT Intrinsic [15], they require hundreds of millions of parameters. Since IID often serves as a fundamental preprocessing step for downstream applications such as relighting, material estimation, and LDR-to-HDR conversion, their computational overhead makes integration into broader vision pipelines challenging. To address this limitation, we propose a latent flow matching-based approach with a lightweight architecture that integrates a UNet with an encoder–decoder design.

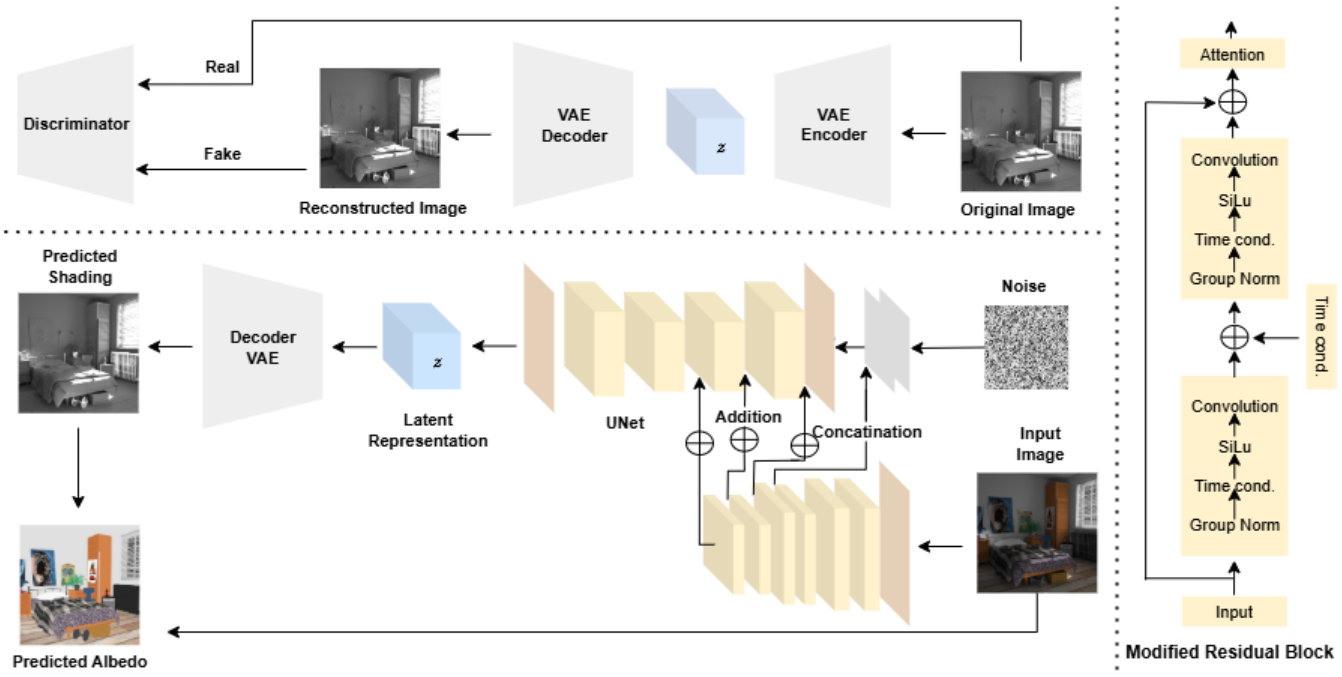

Fig. 1. Architecture of FlowIID. The upper part illustrates that VAE and a discriminator are trained in a VAEGAN setup, and the lower part illustrates model inference. The model follows an encoder–decoder design, where the input image and noise are mapped to a latent representation via the encoder and a UNet backbone with modified residual blocks (shown on the right). The VAE decoder generates the shading component from the latent vector, and the albedo is obtained by dividing the input image by its shading. The VAE encoder and discriminator are used only during training and are not required at inference time.

2.1. Background of Flow Matching

Recently, Flow Matching [16] has emerged as a powerful technique in generative modeling. It uses vector fields to convert samples from a source distribution $p_0(x_0)$ (usually Gaussian) at time $t = 0$ to a complex target distribution $p_1(x_1)$ at time $t = 1$ in a high-dimensional space. To achieve this, we define intermediate distributions $p_t(x_t)$ for $t \in [0, 1]$, which evolve according to the ordinary differential equation (ODE):

$$dx_t = v_t(x_t)\, dt \tag{1}$$

Let θ be our model parameters. At any time step $t \in [0, 1]$, model outputs $u_\theta(x_t, t)$ and will try to predict the v_t. Training is performed by minimizing the mean squared error:

$$\theta = \arg\min_\theta E_{t,x_t} \left\| u_\theta(x_t, t) - v_t \right\|^2 \tag{2}$$

Let $x_0 \sim p_0$ be a sample from the source distribution and $x_1 \sim p_1$ be a sample from the target distribution. Let $\sigma_{\min}$ denote a minimum scaling factor. During training, we randomly select a time step $t \in [0, 1]$ and using optimal transport (OT) equation [16] we get conditional path

$$x_t = \big(1 - (1 - \sigma_{\min})t\big)x_0 + tx_1 \tag{3}$$

and velocity

$$v_t = x_1 - (1 - \sigma_{\min})x_0 \tag{4}$$

Our model will predict $u_\theta(x_t, t)$ and calculate loss using equation (2). During sampling, we start at the time step $(t = 0)$ and go to $(t = 1)$ iteratively. At each time step (t) our model output velocity $u_\theta(x_t, t)$ and we solve it for probability path using equation (1)

3. PROPOSED APPROACH

Given an input image I, our model decomposes it into shading S and albedo A components according to the relation $I = A \cdot S$. To address the complexity of this task while ensuring resource efficiency, we propose FlowIID, a single-step latent flow matching-based architecture (Figure 1). FlowIID operates in two stages: First, the input image and noise are passed through the encoder and UNet to obtain a compact latent shading representation. Second, this latent representation is passed to the VAE decoder to generate the predicted shading. The albedo is then recovered by dividing the input image by the estimated shading. This design integrates generative modeling with intrinsic decomposition, enabling accurate and stable estimation of scene reflectance and illumination. We now describe the architecture of the proposed model and loss function in detail below.

Model Architecture: Our proposed model, FlowIID, consists of four main components: a VAE, a discriminator, a UNet, and an encoder. The VAE and discriminator are

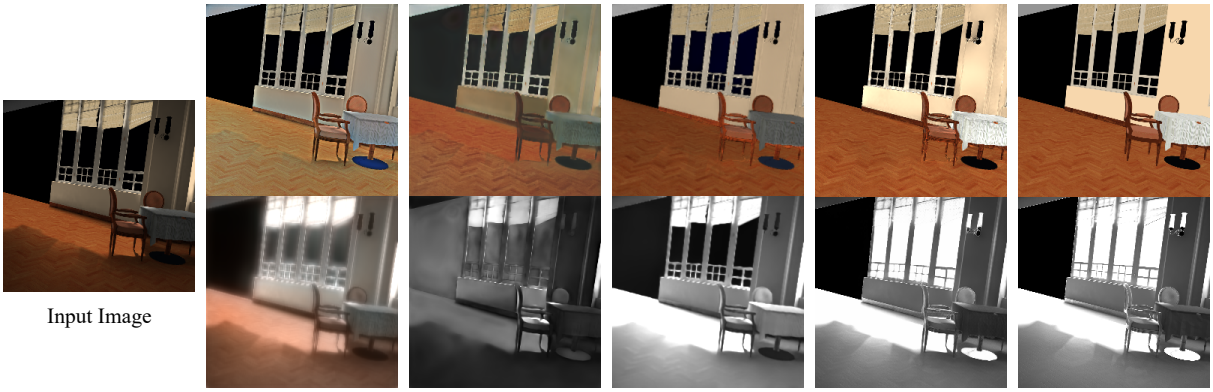

Fig. 2. Qualitative comparison of proposed model with existing work, from left column - (i) input image, (ii) Lettry et al. [17], (iii) Niid-net [2], (iv) Careaga and Askoy [4], (v) Ours, (vi) Ground Truth. The figure shows the albedo and shading components predicted by our model alongside the ground truth and prior methods. Our model produces consistent shading while preserving the color fidelity of the albedo image.

implemented using lightweight convolutional layers. Given an input image of shape $H \times W$, the VAE produces a latent representation of shape $8 \times H/8 \times W/8$. Inspired by the residual block design in [18], we incorporate a modified residual block (MRB) into both the UNet and the encoder, as illustrated in Fig. 1 (right side). The UNet contains two downsampling blocks and two upsampling blocks with skip connections, along with two convolutional projection layers at the ends. Attention layers are included only in the second and third blocks to balance accuracy and efficiency. The UNet takes as input a noise tensor of shape $8 \times H/8 \times W/8$ and predicts a velocity vector of the same dimension.

The encoder contains six downsampling blocks (without attention) and an initial convolutional layer. It processes an input image of shape $3 \times H \times W$. The output from the third block has shape $256 \times H/8 \times W/8$, which is concatenated with the noise tensor of shape $8 \times H/8 \times W/8$. This produces an effective input of shape $264 \times H/8 \times W/8$ for the first convolutional layer of the UNet. Furthermore, the outputs from the last three encoder blocks are pointwise added to the outputs of the UNet's input convolution layer and its two downsampling blocks, respectively.

Training and Loss Functions: We train FlowIID in two stages. In the first stage, we train the VAE and a discriminator in a VAEGAN-type setup. Given a shading image s_0 of shape $H \times W$, the VAE encoder E produces a latent representation $E(s_0)$ of shape $8 \times H/8 \times W/8$. Then we pass the latent representation $E(s_0)$ through the VAE decoder D to get the reconstructed image $\hat{s}_0$. So, our reconstruction loss $\mathcal{L}_{\text{rec}}$ is

$$\mathcal{L}_{\text{rec}} = \|\hat{s}_0 - s_0\|^2 . \tag{5}$$

We further employ a perceptual loss $\mathcal{L}_{\text{perc}}$ and a KL-divergence loss $\mathcal{L}_{\text{KL}}$ for the latent distribution.

Training for VAE proceeds in two steps. For the first 90 epochs, the objective is

$$\mathcal{L}(E, D) = \mathcal{L}_{\text{rec}} + 0.005\,\mathcal{L}_{\text{KL}} + \mathcal{L}_{\text{perc}}. \tag{6}$$

In the subsequent 200 epochs, we introduce the adversarial loss $\mathcal{L}_{\text{A}}$ with weight 0.1, giving

$$\mathcal{L}(E, D) = \mathcal{L}_{\text{rec}} + 0.005\,\mathcal{L}_{\text{KL}} + \mathcal{L}_{\text{perc}} + 0.1\,\mathcal{L}_{\text{A}}. \tag{7}$$

In the second stage, we train our flow matching model. We use loss function given in (2), for 250 epochs. Both the VAE and Flow Matching networks are trained with a batch size of 32 and a learning rate of 1×10^{-4}.

4. EXPERIMENTATION

Datasets and Preprocessing To train our model, we use three standard datasets: Hypersim [9], InteriorVerse [19], and the Multi-Illumination Dataset (MID) [20]. Hypersim and InteriorVerse provide HDR images with albedo, while MID provides only HDR images, for which we use albedo from Careaga and Askoy [4]. Shading images are obtained by dividing HDR by albedo, then tonemapped to $[0, 1]$ without gamma compression [9]. Multiplying LDR shading with albedo yields white-balanced LDR ground truth images in $[0, 1]$. All images are resized to 256×256 via aspect-ratio preserving scaling and cropping.

Table 1. Quantitative comparison of reflectance and shading prediction on MIT Intrinsic dataset

Method	Albedo			Shading		
	MSE↓	LMSE↓	DSSIM↓	MSE↓	LMSE↓	DSSIM↓
CasQNet [21]	0.0091	0.0212	0.0730	0.0081	0.0192	0.0659
PAIDNet. [22]	0.0038	0.0239	0.0368	**0.0032**	0.0267	**0.0475**
USI3D [23]	0.0156	0.0640	0.1158	0.0102	0.0474	0.1310
Cgintrinsics [3]	0.0167	0.0319	0.1287	0.0127	0.0211	0.1376
PIENet [24]	**0.0028**	0.0126	**0.0340**	0.0035	0.0203	0.0485
Ours	0.0040	**0.0043**	0.0435	0.0109	**0.0119**	0.0823

For testing, we evaluate FlowIID on two standard benchmarks: ARAP [14] and the MIT Intrinsic dataset [15]. For ARAP, we follow the protocol of [25], which removes duplicate scenes and adds three scenes from MIST [26]. Since

Table 2. Quantitative comparison of Albedo on ARAP dataset. * implies model is finetuned on ARAP dataset

Method	LMSE↓	RMSE↓	SSIM↑
Niid-net* [2]	0.023	0.129	**0.788**
Lettry et al. [17]	0.042	0.163	0.670
Kocsis et al. [6]	0.030	0.160	0.738
Zhu et al. [19]	0.029	0.184	0.729
Intrinsicanything [28]	0.038	0.171	0.692
Careaga and Aksoy [4]	0.025	0.140	0.671
PIENet [24]	0.031	0.139	0.718
Careaga and Aksoy [25]	0.023	0.145	0.700
Ours	**0.021**	**0.108**	0.760

this protocol introduces some ambiguity in duplicate scene removal and dataset extension, we test baseline models from their publicly available checkpoints to ensure fairness. For MIT Intrinsic, we adopt a similar train and test split of Barron and Malik [27] and fine-tune FlowIID on the training set. All evaluation results are obtained using Euler's method with a single time step.

Table 3. Quantitative comparison of Shading on ARAP Dataset. * implies model is finetuned on ARAP dataset

Method	LMSE↓	RMSE↓	SSIM↑
Niid-net* [2]	0.022	0.206	**0.781**
Lettry et al. [17]	0.042	0.193	0.610
Careaga and Aksoy [4]	0.026	0.168	0.680
PIENet [24]	0.037	0.170	0.718
Ours	**0.022**	**0.132**	0.744

Performance Evaluation on MIT Intrinsic: Table 1 reports quantitative results for albedo and shading prediction. FlowIID achieves the lowest LMSE for albedo (0.0043) and shading (0.0119), indicating strong consistency in structural reconstruction. While PIENet [24] attains the best albedo MSE and DSSIM, and PAIDNet [22] excels in shading MSE and DSSIM, our method provides a balanced trade-off across metrics. Overall, these results show that FlowIID outperforms compared to existing methods.

Performance Evaluation on ARAP: Tables 2 and 3 report our results on albedo and shading prediction. For albedo, our method achieves the lowest LMSE (0.021) and RMSE (0.108), outperforming all prior works, including ARAP-specific finetuned models. For shading, our model obtains the best RMSE (0.132) and competitive LMSE (0.022), while maintaining a strong SSIM (0.744). Although Niid-net [2] reports a slightly higher SSIM, the model is finetuned on ARAP. It demonstrates that our model is based on strong generalization without dataset-specific tuning.

Comparison of Model Parameters: Unlike recent diffusion-based IID approaches [7] and convolutional neural network (CNN)-based methods [4, 24, 25], which typically require tens of iterative steps and rely on hundreds of millions of parameters, our model achieves comparable results with only 52M parameters in a single inference step, as shown in Table 4. This compact and streamlined design makes FlowIID substantially more efficient and practical for deployment in real-world vision pipelines.

Table 4. Comparison of Number of Parameters during inference. * implies parameters while training.

Method	**Parameters**
RGB ↔ X [7]	1.28 B
Niid-net [2]	273.1 M
Careaga and Aksoy [4]	252.05 M
Careaga and Aksoy [25]	548.18 M
PIENet [24]	204.09 M
Ours	**51.71/58.36* M**

Ablation Study: We conduct ablation studies on the ARAP dataset to evaluate the impact of key architectural choices in the UNet (Tables 5 and 6). Removing the concatenation layer results in a noticeable performance drop, highlighting its importance. Increasing the UNet depth from four to five modified residual blocks raises the parameter count by 7.6M but does not provide consistent gains. Our complete model, which employs four blocks with concatenation, achieves the best overall performance across all metrics for both albedo and shading. These findings validate that a compact architecture with selective components is sufficient to achieve strong performance.

Table 5. Ablation Studies on ARAP for Albedo component.

Method	LMSE↓	RMSE↓	SSIM↑
Without Concatenation	0.0242	0.121	0.744
With five blocks	0.0223	0.112	0.755
Ours	**0.0205**	**0.108**	**0.760**

Table 6. Ablation Studies on ARAP for Shading component.

Method	LMSE↓	RMSE↓	SSIM↑
Without Concatenation	0.0242	0.139	0.721
With five blocks	0.0245	0.134	0.714
Ours	**0.0224**	**0.132**	**0.744**

5. CONCLUSION

We introduced FlowIID, a single-step latent flow–matching framework for IID. In contrast to diffusion-based methods that require multiple sampling steps, FlowIID achieves competitive performance in a single forward pass with less number of parameters, making it both memory and computation-efficient. Experiments on standard benchmarks confirm its effectiveness, and its lightweight nature highlights its potential for real-time vision applications. Future work will explore reducing the higher MSE observed near $t = 0$ during training, and extending FlowIID to downstream tasks.

6. REFERENCES

[1] H. Barrow, J. Tenenbaum, A. Hanson, and E. Riseman, "Recovering intrinsic scene characteristics," *Comput. vis. syst*, 1978.

[2] J. Luo, Z. Huang, Y. Li, X. Zhou, G. Zhang, and H. Bao, "Niid-net: Adapting surface normal knowledge for intrinsic image decomposition in indoor scenes," *IEEE Trans. Vis. Comput. Graph.*, 2020.

[3] Z. Li and N. Snavely, "Cgintrinsics: Better intrinsic image decomposition through physically-based rendering," in *ECCV*, 2018, pp. 371–387.

[4] C. Careaga and Y. Aksoy, "Intrinsic image decomposition via ordinal shading," *ACM Trans. Graph.*, 2023.

[5] Q. Fan, J. Yang, G. Hua, B. Chen, and D. Wipf, "Revisiting deep intrinsic image decompositions," in *IEEE CVPR*, 2018.

[6] P. Kocsis, V. Sitzmann, and M. Nießner, "Intrinsic image diffusion for indoor single-view material estimation," in *IEEE CVPR*, 2024.

[7] Z. Zeng et al., "Rgbx: Image decomposition and synthesis using material- and lighting-aware diffusion models," in *ACM SIGGRAPH Conf. Papers*, 2024.

[8] Q. Chen and V. Koltun, "A simple model for intrinsic image decomposition with depth cues," in *IEEE ICCV*, 2013.

[9] M. Roberts et al., "Hypersim: A photorealistic synthetic dataset for holistic indoor scene understanding," in *IEEE ICCV*, 2021.

[10] 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 *ECCV*, 2018, pp. 286–302.

[11] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *IEEE CVPR*, 2022, pp. 10684–10695.

[12] J. Luo et al., "Intrinsicdiffusion: Joint intrinsic layers from latent diffusion models," in *ACM SIGGRAPH*, 2024, pp. 1–11.

[13] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *IEEE ICCV*, 2023.

[14] N. Bonneel, B. Kovacs, S. Paris, and K. Bala, "Intrinsic decompositions for image editing," in *Computer graphics forum*, 2017.

[15] R. Grosse, M. K. Johnson, E. H. Adelson, and W. T. Freeman, "Ground truth dataset and baseline evaluations for intrinsic image algorithms," in *IEEE ICCV*, 2009.

[16] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," *arXiv preprint arXiv:2210.02747*, 2022.

[17] L. Lettry, K. Vanhoey, and L. Van Gool, "Unsupervised deep single-image intrinsic decomposition using illumination-varying image sequences," in *Computer graphics forum*, 2018.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE CVPR*, 2016, pp. 770–778.

[19] J. Zhu et al., "Learning-based inverse rendering of complex indoor scenes with differentiable monte carlo raytracing," in *SIGGRAPH Asia*, 2022.

[20] L. Murmann, M. Gharbi, M. Aittala, and F. Durand, "A multi-illumination dataset of indoor object appearance," in *IEEE ICCV*, 2019.

[21] Y. Ma, X. Jiang, Z. Xia, M. Gabbouj, and X. Feng, "Casqnet: Intrinsic image decomposition based on cascaded quotient network," *IEEE Trans. Circuits Syst. Video Technol.*, 2020.

[22] Y. Huang, K. Liu, T. Chen, Y. Xu, and H. Ji, "Deep intrinsic image decomposition via physics-aware neural networks," *Pattern Recognition*, 2025.

[23] Y. Liu, Y. Li, S. You, and F. Lu, "Unsupervised learning for intrinsic image decomposition from a single image," in *IEEE CVPR*, 2020.

[24] P. Das, S. Karaoglu, and T. Gevers, "Pie-net: Photometric invariant edge guided network for intrinsic image decomposition," in *Proc. IEEE/CVF CVPR*, 2022.

[25] C. Careaga and Y. Aksoy, "Colorful diffuse intrinsic image decomposition in the wild," *ACM Trans. Graph.*, 2024.

[26] X. Hao and B. Funt, "A multi-illuminant synthetic image test set," *Color Res. Appl.*, 2020.

[27] J. T. Barron and J. Malik, "Shape, illumination, and reflectance from shading," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2014.

[28] X. Chen et al., "Intrinsicanything: Learning diffusion priors for inverse rendering under unknown illumination," in *ECCV*, 2024.