

GCC: Generative Color Constancy via Diffusing a Color Checker

Chen-Wei Chang¹ Cheng-De Fan¹ Chia-Che Chang² Yi-Chen Lo²
Yu-Chee Tseng¹ Jiun-Long Huang¹ Yu-Lun Liu¹

¹National Yang Ming Chiao Tung University ²MediaTek Inc.

<https://chenwei891213.github.io/GCC/>

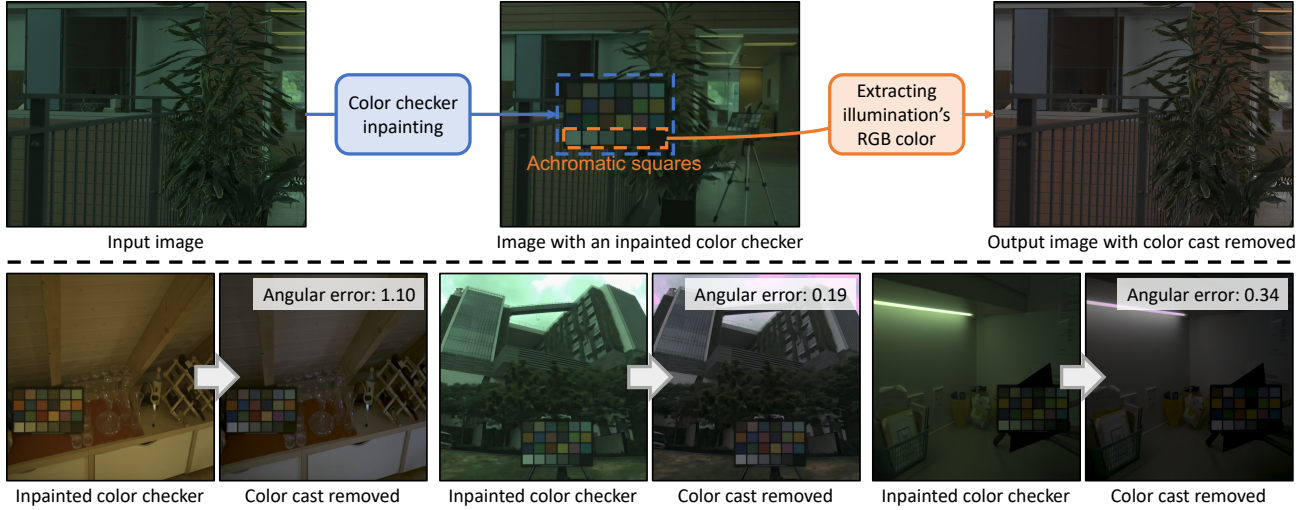


Figure 1. **Our method performs color constancy through diffusion-based color checker inpainting.** (top left) Given an input image, we first inpaint a color checker with Stable Diffusion, aligning the achromatic (gray) squares to accurately reflect the scene illumination (top middle). The RGB color extracted from the achromatic squares is then used to remove the color cast from the input image (top right). (Bottom) Our approach leverages the strong priors of pre-trained diffusion models to accurately estimate scene illumination without requiring physical color checkers during capture, enabling accurate white balance correction across diverse scenes.

Abstract

Color constancy methods often struggle to generalize across different camera sensors due to varying spectral sensitivities. We present GCC, which leverages diffusion models to inpaint color checkers into images for illumination estimation. Our key innovations include (1) a single-step deterministic inference approach that inpaints color checkers reflecting scene illumination, (2) a Laplacian decomposition technique that preserves checker structure while allowing illumination-dependent color adaptation, and (3) a mask-based data augmentation strategy for handling imprecise color checker annotations. By harnessing rich priors from pre-trained diffusion models, GCC demonstrates strong robustness in challenging cross-camera scenarios. These results highlight our method's effective generalization capability across different camera characteristics without requiring sensor-specific training, making it a versatile and practical solution for real-world applications.

1. Introduction

Color constancy is a crucial aspect of computer vision, focused on determining the illumination of a scene to ensure that colors are accurately represented under varying lighting conditions. This process is essential for maintaining a consistent color appearance and for applications ranging from photography to autonomous driving. Traditional statistics-based methodologies [8, 14, 24, 25, 40, 47, 58, 66] rely on various statistical assumptions about scene color distributions. While these methods are computationally efficient, they often struggle in challenging scenes when their underlying assumptions are violated, especially in environments with multiple illuminants or complex lighting conditions.

In contrast, deep learning-based methods [12, 39, 53] have significantly advanced the field of color constancy through their ability to learn complex illumination patterns from training data. These approaches typically employ convolutional neural networks with various architectures to

achieve state-of-the-art performance, particularly in challenging illumination scenarios.

However, a challenge in learning-based color constancy is that models are often constrained to specific camera sensors due to variations in spectral sensitivities. Recent cross-camera approaches [1, 2, 11, 51, 71, 76] have made strides in addressing this limitation through techniques including metric learning, quasi-unsupervised learning, and device-independent representations. Building upon these advances, we explore an approach that leverages foundation models to enhance cross-camera performance.

Inspired by the recent success of DiffusionLight [56], which leverages pre-trained diffusion models for lighting estimation by inpainting a chrome ball, we propose Generative Color Constancy (GCC), a novel approach that harnesses the rich priors of foundation models to overcome the camera-specific limitations of traditional methods. Unlike DiffusionLight [56], which focuses on HDR lighting estimation, our method adapts the concept to color constancy by inpainting a color checker into the input image. Color checkers are widely used calibration tools in color science, and our diffusion model generates one with colors that accurately represent the scene’s illumination. By analyzing the generated color checker’s patches, we can effectively estimate the scene’s illuminant. However, diffusion models typically generate outputs stochastically, which is undesirable for color constancy applications requiring consistency. Drawing insights from recent work on deterministic fine-tuning of image-conditional diffusion models [30], we design a deterministic pipeline that produces consistent illumination estimates while preserving the powerful generalization capabilities of the underlying foundation model. Our approach eliminates the need for camera-specific training data, achieving robust performance across different camera sensors and scene types.

In summary, we make the following contributions:

- We propose a novel color constancy method that leverages diffusion models to inpaint a color checker, which serves as a virtual reference for illumination estimation.
- We introduce a Laplacian decomposition technique that enhances the model to generate color checkers that maintain structure while adapting to scene illumination, improving color extraction accuracy.
- We design a deterministic single step inference pipeline that avoids introducing noise during training and inference, resulting in consistent results and improved computational efficiency compared to traditional diffusion processes.

2. Related Work

Color Constancy and White Balance. Color constancy research spans statistical-based and learning-based approaches. Statistical methods like Gray World [14], Gray Edge [65], Shades-of-Gray [24], Bright Pixels [40], and Gray Index

[58] make assumptions about scene color statistics but struggle with challenging scenes. Learning-based methods have proven more effective, evolving from gamut mapping [7, 17] and regression models [26] to more advanced techniques. Notable developments include CCC [9] and FFCC [10], which use convolutional processing and frequency-domain optimization. Deep learning approaches like FC4 [39], DS-Net [63], RCC-Net [57], and C4 [76] further improve performance with various neural network architectures.

A key challenge is camera-specific spectral sensitivity [1, 29], requiring retraining or calibration for new sensors [49]. Recent solutions include IGTN’s [71] metric learning, quasi-unsupervised learning [11], and cross-dataset approaches [45]. SIIE [1] proposes sensor-independent illumination estimation, while C5 [2] uses unlabeled target camera images during inference, and CLCC [51] employs contrastive learning to improve feature representations. Our work leverages pre-trained diffusion models for color checker inpainting, utilizing their rich knowledge priors to offer a novel approach to illumination estimation with enhanced generalization capability across different camera sensors.

Image-conditional Diffusion Models. Denoising Diffusion Probabilistic Models (DDPMs) [64] achieve state-of-the-art generation by reversing a noising process with UNet architectures [61], demonstrating excellence in density estimation and sample quality [23, 44]. Latent Diffusion Models (LDMs) [60] improved efficiency by operating in compressed latent space and introduced cross-attention conditioning. This enabled powerful inpainting capabilities, demonstrated by Blended Diffusion [5, 6], Paint-by-Example [72], ControlNet [77], and IP-Adapter [74]. Recent work identified that perceived limitations were often due to DDIM scheduler implementation issues [50] rather than fundamental constraints. Our work leverages these insights to effectively adapt diffusion models for color checker inpainting in illumination estimation.

Learning-based Lighting Estimation. Lighting estimation methods traditionally use physical probes like mirror balls [22], 3D objects [52, 69], eyes [55], or faces [15, 75]. Early probe-free approaches used limited models like directional lights [41], sky models [36, 37], or spherical harmonics [32]. Modern methods focus on HDR environment maps, pioneered by Gardner et al. [31]. DeepLight [48] and EverLight [21] handle both indoor and outdoor scenes, while StyleLight [68] uses GANs for joint LDR-HDR prediction. Some works explore panorama outpainting [4, 20] but struggle with HDR [21]. Recently, DiffusionLight [56] introduced virtual chrome ball synthesis using diffusion models. Our work follows a similar direction but focuses on color checker inpainting for illumination estimation.

Fine-tuning Strategies for Diffusion Models. For person-

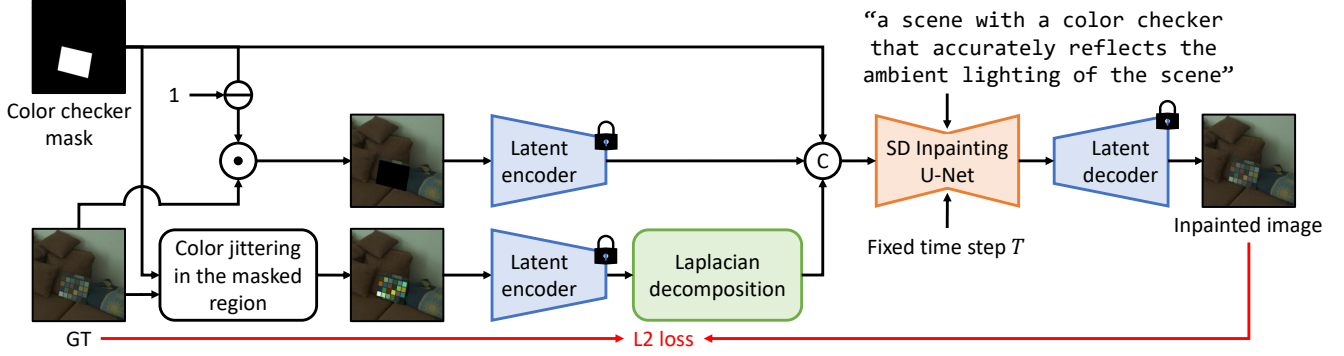


Figure 2. **Overview of our training pipeline.** Starting from stable-diffusion-2-inpainting [60], we enable color checker generation through end-to-end fine-tuning. Given a ground truth color checker image and its mask, we apply color jittering in the masked region. The input image latent passes through Laplacian decomposition before being concatenated with the masked image latent and the resized mask for the SD Inpainting U-Net. The model is trained with an L2 loss between the inpainted output and ground truth image at a fixed timestep T .

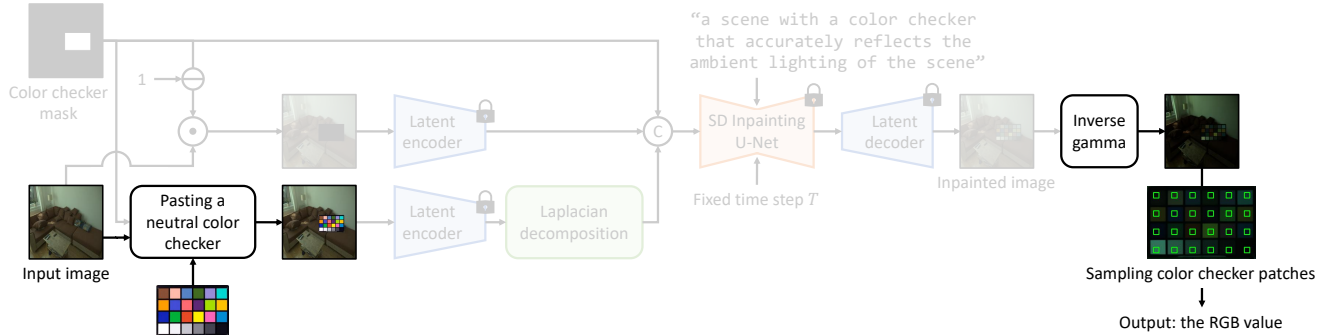


Figure 3. **Overview of our inference pipeline for illumination estimation.** A neutral color checker is pasted onto the input image, which is then encoded into the latent space. The input latent is processed through Laplacian composition before being concatenated with the masked image latent and the resized mask. The modified U-Net generates an inpainted result at fixed timestep T . After inverse gamma correction, we sample the color checker patches to obtain the final RGB illumination value. We highlight the steps and components that are different from the training pipeline.

alization, DreamBooth [62] pioneered special token fine-tuning, while Gal et al. [27] and Voynov et al. [67] proposed learned word embeddings approaches. Similar to DreamBooth, our method fine-tunes pre-trained diffusion models to bind specific visual characteristics to our target domain, enabling a consistent generation of color checkers that reflect scene illumination. For efficiency and fine-tuning strategies, LoRA [38] introduced low-rank weight changes, while SVD-iff [35] and orthogonal fine-tuning [59] proposed alternative parameterizations. For geometry estimation, Marigold [42] demonstrated successful fine-tuning using synthetic data. Inspired by Garcia et al. [30], who showed that simple fine-tuning approaches can be highly effective for deterministic tasks involving low-frequency image components, we adopt their full fine-tuning strategy for our color checker inpainting task. This approach aligns well with our color constancy problem, which primarily focuses on modifying the low-frequency characteristics of color checkers.

3. Method

Instead of directly predicting environmental RGB light, we propose to leverage diffusion models’ rich priors to inpaint a color checker into the scene and extract illumination colors from it. As shown in Figs. 2 and 3, our pipeline consists of (1) During training, we fine-tune a diffusion-based inpainting model at timestep $t=T$ with images containing color checkers, optimizing for deterministic single-step inference (Sec. 3.1-3.2). (2) We introduce Laplacian decomposition to maintain the checker’s high-frequency structure while allowing illumination-aware color adaptation (Sec. 3.3). (3) At inference time, we composite a neutral color checker into a given scene and use our fine-tuned model to inpaint it according to the scene illumination, from which we extract the scene’s light color information (Sec. 3.4).

3.1. Network Architecture

We base our model on stable-diffusion-2-inpainting [60] for its specialized local editing capability. The model consists of a VAE encoder-decoder pair (\mathcal{E}, \mathcal{D}) and a U-Net denoising

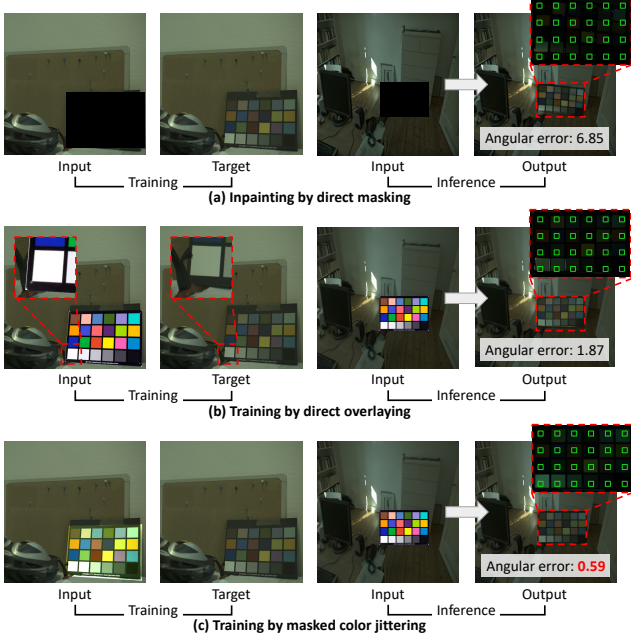


Figure 4. **Analysis of color checker alignment strategies.** (a) Direct inpainting on masked regions leads to poor color checker structure. This is because we do not provide guidance on the desired color checker structure, causing the model to generate contours that do not meet our expectations. (b) Using a homography transform to overlay a template suffers from pixel-level misalignment due to imprecise bounding box annotations. (c) Our mask color jittering approach overcomes corner annotation limitations by allowing the model to generate geometrically consistent color checker structures while accurately reflecting scene illumination.

backbone. Given an RGB image $I \in \mathbb{R}^{H \times W \times 3}$ and a binary mask $M \in \{0, 1\}^{H \times W}$ indicating the color checker region, we first encode both the masked image and the original image into the latent space as $z_{\text{masked}} = \mathcal{E}(I \odot (1 - M))$ and $z = \mathcal{E}(I)$, where \odot denotes element-wise multiplication. The mask M is downsampled by a factor of 8 to match the latent resolution as $M' \in \mathbb{R}^{h \times w}$, where $h = H/8, w = W/8$. During training, the U-Net denoiser ϵ_θ takes as input the concatenation of the noised latent z_t , the downsampled mask M' , and the masked image latent z_{masked} along the channel dimension as $z_{\text{combined}} = [z_t, M', z_{\text{masked}}] \in \mathbb{R}^{h \times w \times (2d+1)}$, where d is the latent dimension. Together with the timestep t and text embedding c , the denoiser is trained to predict the noise as $\epsilon_\theta(z_{\text{combined}}, t, c) \rightarrow \mathbb{R}^{h \times w \times d}$. At inference time, we obtain the final inpainted result by decoding the denoised latent $\hat{I} = \mathcal{D}(z_0)$, where only the color checker region is modified while leaving the rest unmodified, making this architecture particularly suitable for the color constancy task.

3.2. End-to-End Fine-Tuning

Training. Although pre-trained diffusion models like SD and SD inpainting [60] have been exposed to diverse image

collections, additional fine-tuning is crucial for generating precise color checkers that accurately reflect environmental illumination. As shown in experiments Fig. 7, fine-tuning significantly impacts the model’s ability to generate color checkers that faithfully represent scene illumination.

Although SDEdit [54] could be applied to our task, it faces a fundamental trade-off in noise level selection. On one hand, insufficient noise fails to effectively suppress the original chromatic information from the input image, making it difficult to adapt to the target scene illumination. On the other hand, excessive noise, while better at removing unwanted color information, can disrupt the structural consistency between the generated result and the input reference. Furthermore, for color constancy tasks, maintaining a one-to-one correspondence between input and output is essential. While traditional diffusion models’ stochastic nature allows for ensemble improvements through multiple inferences, this comes at an increased computational cost.

Following [30], we adopt an end-to-end fine-tuning approach that enables single-step deterministic inference while maintaining high-quality color checker generation. Specifically, we fine-tune the inpainting U-Net at a fixed timestep $t = T$ as shown in Fig. 2. Given an input image I and its corresponding mask M , we first obtain the augmented image I_{aug} by applying color jittering to the masked region. We then obtain its latent representation through the VAE encoder, $z^* = \mathcal{E}(I_{\text{aug}})$. The latent representation is processed through Laplacian decomposition to extract high-frequency components, $z_h = \mathcal{H}(z^*)$. For single-step prediction, we directly set the noise term $\epsilon = 0$ in the forward process: $z_T = \sqrt{\alpha_T} z_h + \sqrt{1 - \alpha_T} \epsilon$. The denoised latent is then obtained through $\hat{z}_0 = \sqrt{\alpha_T} z_T - \sqrt{1 - \alpha_T} \epsilon_\theta(z_{\text{combined}}, T, c)$, where $z_{\text{combined}} = [z_T, M', z_{\text{masked}}] \in \mathbb{R}^{h \times w \times (2d+1)}$ represents the concatenated input features along the channel dimension, and c denotes the text condition. Finally, we decode the latent to obtain the inpainted image: $\hat{I} = \mathcal{D}(\hat{z}_0)$. The model is optimized using a mean squared error loss:

$$\mathcal{L} = \frac{1}{HW} \sum_{i,j} (I_{i,j}^* - \hat{I}_{i,j})^2, \quad (1)$$

where (i, j) denotes the pixel coordinates, and H and W are the height and width of the image, respectively.

Color checker misalignment issue. Existing color constancy datasets [18, 33] only provide rough bounding boxes for color checkers instead of precise corner point locations. This hinders our ability to accurately align the standard sRGB color checker with the one in the original image, affecting the model’s learning of the transformation from standard to harmonized colors. To overcome this limitation, we designed a mask region-based data augmentation method.

We first analyze two intuitive solutions: directly masking and allowing the model to perform inpainting. This approach results in generated color checkers with contours that do

not meet our expectations, making accurate color extraction from the patches difficult (Fig. 4 (a)). The second solution involved overlaying the color checker template directly onto the original image (Fig. 4 (b)). However, due to the absence of precise corner point locations, alignment with the raw checker remains imperfect at a per-pixel level even when using homography transform.

Masked color jittering. Therefore, we further explored a third approach: directly applying strong color jittering to the mask region (Fig. 4 (c)). This seemingly counterintuitive method aims to destroy clues that may leak sensor-specific information, forcing the model to rely on information outside the mask region to reconstruct the original color checker that aligns with the ground truth.

Random color jittering on masked checkers helps our model learn robust mappings between neutral color references and scene-specific lighting appearances. The augmented image I_{aug} is obtained by:

$$I_{\text{aug}} = (1 - M) \odot I + M \odot \mathcal{T}(I), \quad (2)$$

where I is the input image, M is the binary mask, \odot denotes element-wise multiplication, and $\mathcal{T}(\cdot)$ represents the color jittering function that randomly applies brightness, contrast, and saturation adjustments to the masked region. By randomly perturbing the color checker region, we force the model to rely on contextual illumination cues rather than local color checker patterns. This approach overcomes the limitations of imprecise annotations in existing datasets and enhances the model’s ability to learn accurate illumination estimation from scene context.

3.3. Laplacian Decomposition

Although mask color jittering addresses the imprecise corner annotation issue, the randomness in jittering may occasionally allow low-frequency information leakage from the masked region. This could cause the model to simply *reconstruct* the masked area rather than *harmonize* it with the scene illumination. To address this issue, we introduce the Laplacian decomposition technique.

By extracting only the high-frequency components of the input image through Laplacian decomposition, our approach serves two purposes: First, it preserves the structural details needed to generate a color checker that faithfully maintains the patch layout of our pre-pasted reference. Second, it minimizes the influence of low-frequency color information, encouraging the model to focus on harmonizing the generated color checker with the scene illumination rather than reconstructing the original colors. The key benefit of Laplacian decomposition, as shown in Fig. 7, allows the model to generate color checkers that maintain structural consistency while correctly reflecting scene illumination, enabling accurate illumination estimation.

3.4. Inference

The complete inference pipeline of our method is illustrated in Fig. 3, which consists of the following steps:

Color checker generation. We first composite a fixed-size neutral color checker centered at the mask region. The input image is then gamma-corrected with $\gamma = 2.2$ to transform it to the sRGB domain. This preprocessed image is processed through our model in a single forward pass with fixed timestep $t = T$. The output is then inverse gamma-corrected to obtain the raw domain result.

Illumination estimation. Since we have precise control over initial checker placement and Laplacian decomposition ensures structural preservation, we can reliably extract color information from each patch. Specifically, we directly map the generated checker to a standardized grid, followed by applying fixed masks to sample colors from each patch. The scene illumination is then estimated from the achromatic patches of the color checker.

4. Experiments

4.1. Experimental Setup

Dataset. We use two publicly available color constancy benchmark datasets in our experiments: the NUS-8 dataset [19] and the re-processed Color Checker dataset [33] (referred to as the Gehler dataset). The Gehler dataset [33] contains 568 original images captured by two different cameras, while the NUS-8 dataset [19] contains 1736 original images captured by eight different cameras. Each image in both datasets includes a Macbeth Color Checker chart, which serves as a reference for the ground-truth illuminant color.

Evaluation metrics. To evaluate the performance of color constancy methods, we use the standard angular error metric, which measures the angular difference between the estimated illuminant and the ground-truth illuminant. Specifically, the angular error θ between an estimated illuminant vector $\hat{\mathbf{y}}$ and the ground-truth illuminant vector \mathbf{y} is defined as:

$$\theta = \arccos \left(\frac{\hat{\mathbf{y}} \cdot \mathbf{y}}{|\hat{\mathbf{y}}| |\mathbf{y}|} \right) \quad (3)$$

The angular error is measured in degrees, with smaller values indicating better estimation accuracy. Following previous works, we report the following statistics of the angular error.

4.2. Implementation Details

Our implementation is based on the stable-diffusion-2-inpainting model [60] using PyTorch. All input images are resized to 512×512 resolution for both training and inference. Since the pre-trained VAE was trained on sRGB images, we apply a gamma correction of $\gamma = 1/2.2$ on linear RGB images before encoding to minimize the domain gap. Conversely, after VAE decoding, we apply inverse gamma

Table 1. **Camera-agnostic evaluation.** All results are in units of degrees.

Training set → Testing set	NUS-8 dataset [19] → Gehler dataset [33]					Gehler dataset [33] → NUS-8 dataset [19]				
Method	Mean	Median	Tri-mean	Best 25%	Worst 25%	Mean	Median	Tri-mean	Best 25%	Worst 25%
Statistical Methods										
White-Path [13]	7.55	5.68	6.35	1.45	16.12	9.91	7.44	8.78	1.44	21.27
Gray-World [14]	6.36	6.28	6.28	2.33	10.58	4.59	3.46	3.81	1.16	9.85
1st-order Gray-Edge [66]	5.33	4.52	4.73	1.86	10.43	3.35	2.58	2.76	0.79	7.18
2nd-order Gray-Edge [66]	5.13	4.44	4.62	2.11	9.26	3.36	2.70	2.80	0.89	7.14
Shades-of-Gray [24]	4.93	4.01	4.23	1.14	10.20	3.67	2.94	3.03	0.99	7.75
General Gray-World [8]	4.66	3.48	3.81	1.00	10.09	3.20	2.56	2.68	0.85	6.68
Grey Pixel (edge) [73]	4.60	3.10	-	-	-	3.15	2.20	-	-	-
Cheng et al. [19]	3.52	2.14	2.47	0.50	8.74	2.92	2.04	2.24	0.62	6.61
LSRS [28]	3.31	2.80	2.87	1.14	6.39	3.45	2.51	2.70	0.98	7.32
GI [58]	3.07	1.87	2.16	0.43	7.62	2.91	1.97	2.13	0.56	6.67
Learning-based Methods										
Bayesian [34]	4.75	3.11	3.50	1.04	11.28	3.65	3.08	3.16	1.03	7.33
Chakrabarti [16]	3.52	2.71	2.80	0.86	7.72	3.89	3.10	3.26	1.17	7.95
FFCC [10]	3.91	3.15	3.34	1.22	7.94	3.19	2.33	2.52	0.84	7.01
SqueezeNet-FC ⁴ [39]	3.02	2.36	2.50	0.81	6.36	2.40	2.03	2.10	0.70	4.80
C ⁴ _{SqueezeNet-FC4} [76]	2.73	2.20	2.28	0.72	5.69	2.28	1.90	1.97	0.67	4.60
SIIE [1]	3.72	2.46	2.79	1.02	8.51	4.24	3.88	3.93	1.45	7.66
CLCC [51]	3.05	2.44	2.51	0.89	6.30	3.42	2.95	3.06	0.94	6.70
C ⁵ [2]	3.34	2.57	2.68	0.78	7.39	2.65	1.98	2.14	0.66	5.72
Ours	2.35	2.02	2.06	0.78	4.57	2.38	2.01	2.10	0.80	4.58

correction to convert the output back to the linear domain for metric evaluation.

Following parameter settings from [30], we train our models using the Adam optimizer with an initial learning rate of 5×10^{-5} and apply an exponential learning rate decay after a 150-step warm-up period. For cross-dataset evaluation, when training on the Gehler dataset and testing on NUS-8, we use a batch size of 8 with no gradient accumulation for 20k iterations. When training on NUS-8 and testing on the Gehler dataset, we use a batch size of 8 with gradient accumulation over 2 steps (effective batch size of 16) for 20k iterations.

For data augmentation, we follow FC4 [39] to rescale images by random RGB values in [0.6, 1.4] in the raw domain, noting that we only rescale input images since our training does not require ground truth illumination. We also apply mask color jittering to handle imprecise color checker annotations. For Laplacian decomposition, we use a two-level pyramid ($L = 2$) to balance the preservation of high-frequency structural details and the suppression of low-frequency color information. All experiments were conducted on an NVIDIA RTX 4090 GPU. Additional implementation details are provided in the supplementary material.

4.3. Results and Comparisons

Evaluation protocols. We conduct experiments under three different protocols to comprehensively assess our method’s performance and generalization capabilities. First, following the camera-agnostic evaluation protocol from C4 [76], we evaluate robustness against camera sensitivity variations by training on one dataset and testing on another. Specifically, we train on the NUS-8 dataset and test on the Gehler dataset and vice versa. As shown in Table 1, our method achieves

Table 2. **Leave-one-out evaluation on the NUS-8 Dataset [19].**

NUS-8 Dataset [19]	Mean	Med.	Tri.	Best 25%	Worst 25%
Gray-world [14]	4.59	3.46	3.81	1.16	9.85
Shades-of-Gray [24]	3.67	2.94	3.03	0.98	7.75
Local Surface Reflectance [28]	3.45	2.51	2.70	0.98	7.32
PCA-based B/W Colors [18]	2.93	2.33	2.42	0.78	6.13
Grayness Index [58]	2.91	1.97	2.13	0.56	6.67
Cross-dataset CC [46]	3.08	2.24	-	-	-
Quasi-Unsupervised CC [11]	3.00	2.25	-	-	-
SIIE [1]	2.05	1.50	-	0.52	4.48
FFCC [10]	2.87	2.14	2.30	0.71	6.23
C5 [2]	2.54	1.90	2.02	0.61	5.61
Ours	2.03	1.78	1.83	0.77	3.69

Table 3. **Leave-one-out evaluation on the Gehler Dataset [33].**

Gehler dataset [33]	Mean	Med.	Tri.	Best 25%	Worst 25%
Shades-of-Gray [24]	4.93	4.01	4.23	1.14	10.20
PCA-based B/W Colors [18]	3.52	2.14	2.47	0.50	8.74
ASM [3]	3.80	2.40	2.70	-	-
Woo et al. [70]	4.30	2.86	3.31	0.71	10.14
Grayness Index [58]	3.07	1.87	2.16	0.43	7.62
Cross-dataset CC [46]	2.87	2.21	-	-	-
Quasi-Unsupervised CC [11]	3.46	2.23	-	-	-
SIIE [1]	2.77	1.93	-	0.55	6.53
FFCC [10]	2.95	2.19	2.35	0.57	6.75
C5 [2]	2.50	1.99	2.03	0.47	5.46
Ours	2.80	2.50	2.58	1.10	5.00

competitive performance compared to state-of-the-art approaches. Second, we adopt the leave-one-out protocol from SIIE [1] to assess performance on unseen camera sensors by excluding images from one camera during training and testing on them. This process is repeated for all cameras, with results in Tab. 2 and Tab. 3 demonstrating our method’s effectiveness. Both protocols highlight that our approach leverages diffusion priors to learn sensor-independent illumination features without requiring camera-specific retrain-

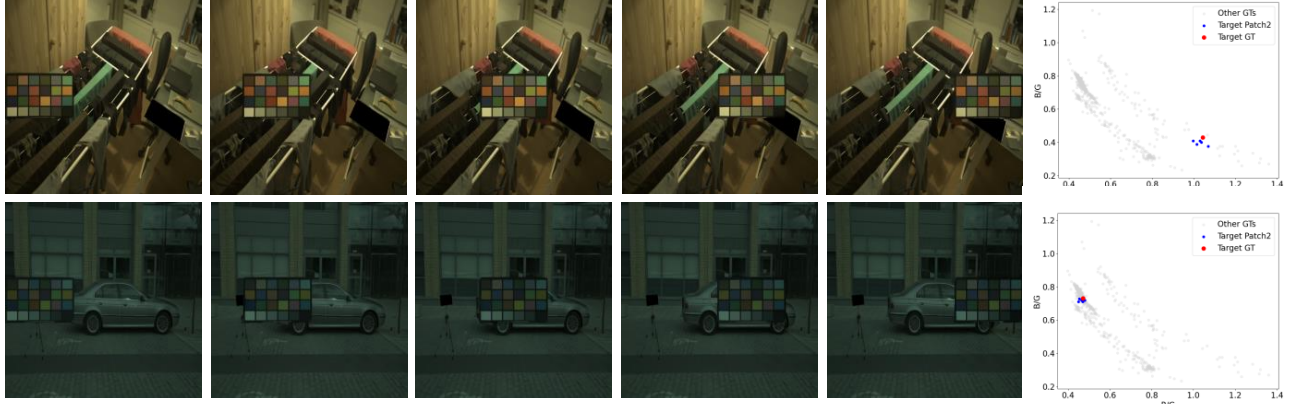


Figure 5. **Sensitivity to color checker placement.** This figure demonstrates the robustness of our method across various color checker positions under a single light source scenario. The left part displays different placements of color checkers and their corresponding processed results, showing that our method remains effective under challenging warm color temperatures (regions with lower data distribution). The scatter plots on the right quantitatively validate this observation, where the estimated illumination values consistently cluster near the ground truth target, confirming the precision and consistency of our approach.

ing. Additionally, we conducted standard three-fold cross-validation on both the NUS-8 [19] and Gehler datasets [33]. As shown in Tab. 4 and Tab. 5, our method achieves performance comparable to other approaches, particularly in worst-case scenarios.

Position-aware Sampling and Consistency. Fig. 5 demonstrates our method’s robustness in single-illumination scenes. Unlike prior approaches, our ability to sample at different positions and generate result ensembles enables the quantification of model consistency, showcasing our approach’s precision and reliability.

Spatially Varying Illumination in Multi-source Scenes. Traditional color constancy methods typically assume a single global illuminant, limiting their applicability in complex lighting scenarios. Our method naturally extends to spatially varying illumination conditions. We evaluated this capability on the LSMI dataset [43], which features challenging multi-illuminant scenes. By dividing each image into a 4×4 grid, inpainting color checkers in each cell, and interpolating these local estimates, our method effectively models different lighting regions. Results in Tab. 6 demonstrate that our approach can handle complex lighting environments without requiring specific fine-tuning for multi-illuminant data. Fig. 6 visually confirms our method’s ability to adapt to lighting transitions in real-world environments.

Computational Efficiency. Our method maintains efficient inference times due to its single-step design. Using an NVIDIA RTX 4090 GPU, it processes a 512×512 image in 180ms, significantly faster than traditional diffusion methods requiring multiple denoising steps as shown in Tab. 7, while preserving accuracy benefits from diffusion priors.

Table 4. **Three-fold cross-validation on NUS-8 dataset [19].**

NUS-8 dataset [19]	Mean	Med.	Tri.	Best 25%	Worst 25%
CCC [9]	2.38	1.48	1.69	0.45	5.85
AlexNet-FC4 [39]	2.12	1.53	1.67	0.48	4.78
FFCC [10]	1.99	1.31	1.43	0.35	4.75
C ⁴ _{SqueezeNet-FC4} [76]	1.96	1.42	1.53	0.48	4.40
CLCC [51]	1.84	1.31	1.42	0.41	4.20
Ours	2.10	1.52	1.69	0.56	4.38

Table 5. **Three-fold cross-validation on Gehler dataset [33].**

Gehler dataset [33]	Mean	Med.	Tri.	Best 25%	Worst 25%
CCC [9]	1.95	1.22	1.38	0.35	4.76
SqueezeNet-FC4 [39]	1.65	1.18	1.27	0.38	3.78
FFCC [10]	1.61	0.86	1.02	0.23	4.27
C ⁴ _{SqueezeNet-FC4} [76]	1.35	0.88	0.99	0.28	3.21
CLCC [51]	1.44	0.92	1.04	0.27	3.48
Ours	1.91	1.80	1.84	0.60	3.46

Table 6. **Zero-shot evaluation on the LSMI Dataset [43].** Mean angular error (MAE) for the spatially varying illumination map.

Method	Galaxy			Nikon		
	Single	Multi	Mixed	Single	Multi	Mixed
LSMI-H [43]	2.85	3.13	3.06	2.76	3.2	2.99
LSMI-U [43]	2.95	2.35	2.63	1.51	2.36	1.95
Ours	2.05	3.44	2.82	2.10	3.58	2.88

4.4. Ablation Studies

We conducted a series of ablation experiments to validate the importance of key design choices, including using Laplacian decomposition, noise prediction-based LoRA fine-tuning, and mask-based data augmentation in Tab. 8.

Without Laplacian decomposition. Without Laplacian decomposition, we use only the VAE encoder’s latent representation as input. As shown in Fig. 7, the generated checker

Table 7. **Comparison between fine-tuned SDXL inpainting and our one-step model.** All metrics are reported in degrees, and inference time is measured on a single 512×512 image using an NVIDIA RTX 4090 GPU. All models are trained on the NUS-8 dataset [19] and evaluated on the Gehler dataset [33].

Method	Steps	Ensemble	Inference time (s)	Mean	Median	Best-25%	Worst-25%
SDXL Inpainting (SDEdit)	25	10	17.98	4.47	3.25	1.07	10.01
Full Model	1	1	0.18	2.35	2.02	0.78	4.57

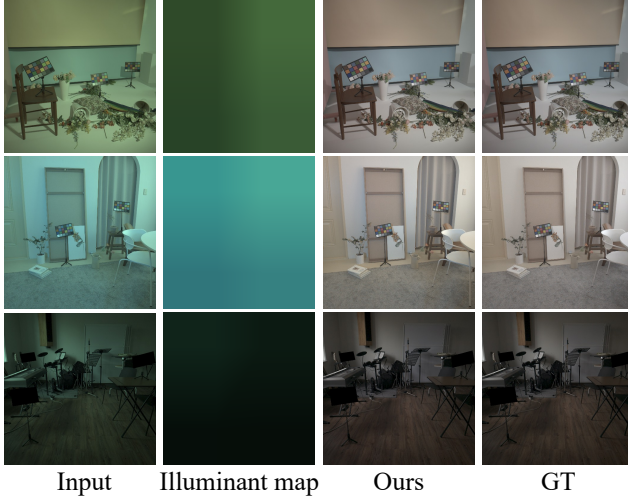


Figure 6. **Spatially varying illumination in multi-source scenes.** From left to right: input image with mixed illumination, illuminant coefficient map showing per-pixel light distribution, our white balanced result, and ground truth white balanced image.

Table 8. **Ablation study on key components of our method.** We evaluate the impact of components: Laplacian decomposition (Lap.), color checker inpainting vs. RGB prediction, and masked color jittering (Mask DA). All models are trained on the NUS-8 dataset [19] and evaluated on the Gehler dataset [33]. The results show that our color checker inpainting approach outperforms direct RGB prediction, and the combination with other components (Laplacian decomposition and masked color jittering) yields the best performance. All error metrics are reported in degrees, with lower values indicating better performance.

Noise	Lap.	Inpaint	Mask DA	Mean	Median	Best-25%	Worst-25%
Zeros	-	✓	✓	3.71	2.86	1.31	7.68
Zeros	✓	✓	-	3.52	2.76	1.25	6.78
Zeros	-	-	-	2.98	2.53	1.26	6.14
Zeros	✓	✓	✓	2.35	2.02	0.78	4.57

is contaminated by low-frequency information from the initial neutral reference, producing disharmonious colors that prevent accurate environmental color estimation.

With noise. In this experiment, we used LoRA [38] to fine-tune the SDXL inpainting model [60] and obtained the final output through ensemble averaging across multiple samples. As shown in Tab. 7, this approach underperforms our final method due to the fundamental trade-off between preserving the color checker’s geometry and suppressing low-frequency information from the neutral reference checker.

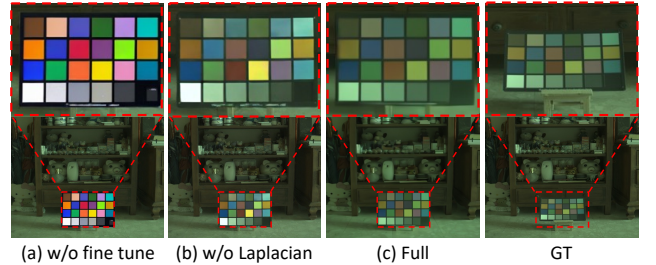


Figure 7. **Effect of fine-tuning and Laplacian decomposition.** (a) Results without fine-tuning show poor color checker quality due to the domain gap between the pre-trained diffusion model’s training data (sRGB images) and our gamma-corrected raw images, leading to disharmonious inpainting results. (b) Results without Laplacian decomposition are biased by low-frequency information from the neutral color checker, leading to inharmonic generation. (c) Our full method with both components produces well-harmonized color checkers that accurately reflect scene illumination.

Without mask data augmentation. Initially, we aligned color checkers using homography based on dataset corner locations, but imprecisions led to alignment errors at the pixel level. Our mask-based data augmentation approach eliminates reliance on specific corner positions, producing more accurate scene-harmonized color checkers that better represent the overall scene lighting.

Without inpainting color checker. In this experiment, we directly used the diffusion model to predict scene illumination RGB, not by inpainting a checker. This direct approach proves less effective than our inpainting method, highlighting the importance of color checker references for accurate illumination estimation.

5. Conclusion

In this work, we introduce a color constancy method that leverages image-conditional diffusion models to inpaint color checkers directly into images. Our approach harnesses the rich priors of foundation models to overcome generalization challenges across varying camera sensors. By employing Laplacian decomposition, our method maintains the checker’s high-frequency structure while adapting to scene illumination, enabling accurate light color estimation without camera-specific training. Experiments demonstrate robust performance in cross-camera scenarios, particularly for challenging cases, making our approach a versatile solution for real-world color constancy applications.

Acknowledgements. This research was funded by the National Science and Technology Council, Taiwan, under Grants NSTC 112-2222-E-A49-004-MY2 and 113-2628-E-A49-023-. The authors are grateful to Google, NVIDIA, and MediaTek Inc. for their generous donations. Yu-Lun Liu acknowledges the Yushan Young Fellow Program by the MOE in Taiwan.

References

- [1] Mahmoud Afifi and Michael S Brown. Sensor-independent illumination estimation for dnn models. In *BMVC*, 2019. 2, 6, 12
- [2] Mahmoud Afifi, Jonathan T Barron, Chloe LeGendre, Yun-Ta Tsai, and Francois Bleibel. Cross-camera convolutional color constancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2, 6
- [3] Arash Akbarinia and C. Alejandro Parraga. Colour constancy beyond the classical receptive field. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(9):2081–2094, 2018. 6
- [4] Naofumi Akimoto, Seito Kasai, Masaki Hayashi, and Yoshimitsu Aoki. 360-degree image completion by two-stage conditional gans. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 4704–4708. IEEE, 2019. 2
- [5] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18208–18218, 2022. 2
- [6] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM Trans. Graph.*, 42(4), 2023. 2
- [7] Kobus Barnard. Improvements to gamut mapping colour constancy algorithms. In *European conference on computer vision*, 2000. 2
- [8] Kobus Barnard, Vlad Cardei, and Brian Funt. A comparison of computational color constancy algorithms. i: Methodology and experiments with synthesized data. *IEEE transactions on Image Processing*, 11(9):972–984, 2002. 1, 6
- [9] Jonathan T Barron. Convolutional color constancy. In *International Conference on Computer Vision*, 2015. 2, 7
- [10] Jonathan T Barron and Yun-Ta Tsai. Fast fourier color constancy. In *Computer Vision and Pattern Recognition*, 2017. 2, 6, 7
- [11] Simone Bianco and Claudio Cusano. Quasi-unsupervised color constancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2, 6
- [12] Simone Bianco, Claudio Cusano, and Raimondo Schettini. Color constancy using cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 81–89, 2015. 1
- [13] David H Brainard and Brian A Wandell. Analysis of the retinex theory of color vision. *JOSA A*, 3(10):1651–1661, 1986. 6
- [14] Gershon Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 1980. 1, 2, 6
- [15] Dan A Calian, Jean-François Lalonde, Paulo Gotardo, Tomas Simon, Iain Matthews, and Kenny Mitchell. From faces to outdoor light probes. In *Computer Graphics Forum*, pages 51–61. Wiley Online Library, 2018. 2
- [16] Ayan Chakrabarti. Color constancy by learning to predict chromaticity from luminance. *Advances in Neural Information Processing Systems*, 28, 2015. 6
- [17] Ayan Chakrabarti, Keigo Hirakawa, and Todd Zickler. Color constancy with spatio-spectral statistics. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011. 2
- [18] Dongliang Cheng, Dilip K. Prasad, and Michael S. Brown. Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution. *J. Opt. Soc. Am. A*, 31(5):1049–1058, 2014. 4, 6
- [19] Dongliang Cheng, Dilip K Prasad, and Michael S Brown. Illuminant estimation for color constancy: why spatial-domain methods work and the role of the color distribution. *JOSA A*, 31(5):1049–1058, 2014. 5, 6, 7, 8, 12, 13, 14, 15
- [20] Mohammad Reza Karimi Dastjerdi, Yannick Hold-Geoffroy, Jonathan Eisenmann, Siavash Khodadadeh, and Jean-François Lalonde. Guided co-modulated gan for 360° field of view extrapolation. In *2022 International Conference on 3D Vision (3DV)*, pages 475–485. IEEE, 2022. 2
- [21] Mohammad Reza Karimi Dastjerdi, Jonathan Eisenmann, Yannick Hold-Geoffroy, and Jean-François Lalonde. Everlight: Indoor-outdoor editable hdr lighting estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7420–7429, 2023. 2
- [22] Paul Debevec. Rendering synthetic objects into real scenes: Bridging traditional and image-based graphics with global illumination and high dynamic range photography. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, page 189–198, New York, NY, USA, 1998. Association for Computing Machinery. 2
- [23] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 2021. 2
- [24] Graham D Finlayson and Elisabetta Trezzi. Shades of gray and colour constancy. In *Color and Imaging Conference*, 2004. 1, 2, 6
- [25] David A Forsyth. A novel algorithm for color constancy. *International Journal of Computer Vision*, 5(1):5–35, 1990. 1
- [26] Brian Funt and Weihua Xiong. Estimating illumination chromaticity via support vector regression. In *Color and Imaging Conference*, 2004. 2
- [27] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 3
- [28] Shaobing Gao, Wangwang Han, Kaifu Yang, Chaoyi Li, and Yongjie Li. Efficient color constancy with local surface reflectance statistics. In *Computer Vision—ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6–12, 2014, Proceedings, Part II 13*, pages 158–173. Springer, 2014. 6

- [29] Shao-Bing Gao, Ming Zhang, Chao-Yi Li, and Yong-Jie Li. Improving color constancy by discounting the variation of camera spectral sensitivity. *JOSA A*, 2017. 2
- [30] Gonzalo Martin Garcia, Karim Abou Zeid, Christian Schmidt, Daan de Geus, Alexander Hermans, and Bastian Leibe. Fine-tuning image-conditional diffusion models is easier than you think. *arXiv preprint arXiv:2409.11355*, 2024. 2, 3, 4, 6, 13
- [31] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Trans. Graph.*, 36(6), 2017. 2
- [32] Mathieu Garon, Kalyan Sunkavalli, Sunil Hadap, Nathan Carr, and Jean-François Lalonde. Fast spatially-varying indoor lighting estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6908–6917, 2019. 2
- [33] Peter Vincent Gehler, Carsten Rother, Andrew Blake, Tom Minka, and Toby Sharp. Bayesian color constancy revisited. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008. 4, 5, 6, 7, 8, 12, 13, 14, 16
- [34] Peter Vincent Gehler, Carsten Rother, Andrew Blake, Tom Minka, and Toby Sharp. Bayesian color constancy revisited. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 6
- [35] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. *arXiv preprint arXiv:2303.11305*, 2023. 3
- [36] Yannick Hold-Geoffroy, Kalyan Sunkavalli, Sunil Hadap, Emiliano Gambaretto, and Jean-François Lalonde. Deep outdoor illumination estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7312–7321, 2017. 2
- [37] Lukas Hosek and Alexander Wilkie. An analytic model for full spectral sky-dome radiance. *ACM Transactions on Graphics (TOG)*, 31(4):1–9, 2012. 2
- [38] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 3, 8, 13
- [39] Yuanming Hu, Baoyuan Wang, and Stephen Lin. Fc4: Fully convolutional color constancy with confidence-weighted pooling. In *Computer Vision and Pattern Recognition*, 2017. 1, 2, 6, 7, 12
- [40] Hamid Reza Vaezi Joze, Mark S Drew, Graham D Finlayson, and Perla Aurora Troncoso Rey. The role of bright pixels in illumination estimation. In *Color and Imaging Conference*, 2012. 1, 2
- [41] Kevin Karsch, Varsha Hedau, David Forsyth, and Derek Hoiem. Rendering synthetic objects into legacy photographs. *ACM Transactions on graphics (TOG)*, 30(6):1–12, 2011. 2
- [42] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9492–9502, 2024. 3
- [43] Dongyoung Kim, Jinwoo Kim, Seonghyeon Nam, Dongwoo Lee, Yeonkyung Lee, Nahyup Kang, Hyong-Euk Lee, ByungIn Yoo, Jae-Joon Han, and Seon Joo Kim. Large scale multi-illuminant (lsmi) dataset for developing white balance algorithm under mixed illumination. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2410–2419, 2021. 7
- [44] Diederik Kingma, Tim Salimans, Ben Poole, and Jonathan Ho. Variational diffusion models. *Advances in neural information processing systems*, 2021. 2
- [45] Samu Koskinen, Dan Yang, and Joni-Kristian Kämäräinen. Cross-dataset color constancy revisited using sensor-to-sensor transfer. In *British Machine Vision Conference*, 2020. 2
- [46] Samu Koskinen¹², Dan Yang, and Joni-Kristian Kämäräinen. Cross-dataset color constancy revisited using sensor-to-sensor transfer. *BMVC*, 2020. 6
- [47] Edwin H Land. The retinex theory of color vision. *Scientific American*, 237(6):108–129, 1977. 1
- [48] Chloe LeGendre, Wan-Chun Ma, Graham Fyffe, John Flynn, Laurent Charbonnel, Jay Busch, and Paul Debevec. Deeplight: Learning illumination for unconstrained mobile mixed reality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019. 2
- [49] Orly Liba, Kiran Murthy, Yun-Ta Tsai, Tim Brooks, Tianfan Xue, Nikhil Karnad, Qiurui He, Jonathan T Barron, Dillon Sharlet, Ryan Geiss, et al. Handheld mobile photography in very low light. *ACM Trans. Graph.*, 2019. 2
- [50] Shanchuan Lin, Bingchen Liu, Jiashi Li, and Xiao Yang. Common diffusion noise schedules and sample steps are flawed. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2024. 2
- [51] Yi-Chen Lo, Chia-Che Chang, Hsuan-Chao Chiu, Yu-Hao Huang, Chia-Ping Chen, Yu-Lin Chang, and Kevin Jou. Clcc: Contrastive learning for color constancy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 2, 6, 7
- [52] Stephen Lombardi and Ko Nishino. Reflectance and illumination recovery in the wild. *IEEE transactions on pattern analysis and machine intelligence*, 38(1):129–141, 2015. 2
- [53] Zitong Lou, Theo Gevers, Nico K. Hu, and Marcel P. Lucassen. Color constancy by deep learning. In *British Machine Vision Conference*, 2015. 1
- [54] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations, 2022. 4
- [55] Ko Nishino and Shree K Nayar. Eyes for relighting. *ACM Transactions on Graphics (TOG)*, 23(3):704–711, 2004. 2
- [56] Pakkapon Phongthawee, Worameth Chinchuthakun, Nontaphat Sinsunthithet, Varun Jampani, Amit Raj, Pramook Khungurn, and Supasorn Suwajanakorn. Diffusionlight: Light probes for free by painting a chrome ball. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 98–108, 2024. 2
- [57] Yanlin Qian, Ke Chen, Jarno Nikkanen, Joni-Kristian Kämäräinen, and Jiri Matas. Recurrent color constancy. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5459–5467, 2017. 2

- [58] Yanlin Qian, Joni-Kristian Kamarainen, Jarno Nikkanen, and Jiri Matas. On finding gray pixels. In *Computer Vision and Pattern Recognition*, 2019. 1, 2, 6
- [59] Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. *arXiv preprint arXiv:2306.07280*, 2023. 3
- [60] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 2, 3, 4, 5, 8, 12, 13
- [61] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 2015. 2
- [62] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 3, 14
- [63] Wu Shi, Chen Change Loy, and Xiaoou Tang. Deep specialized network for illuminant estimation. In *European Conference on Computer Vision*, pages 371–387. Springer, 2016. 2
- [64] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2015. 2
- [65] Joost van de Weijer, Theo Gevers, and Arjan Gijsenij. Edge-based color constancy. *IEEE Trans. Image Process.*, pages 2207–2214, 2007. 2
- [66] Joost Van De Weijer, Theo Gevers, and Arjan Gijsenij. Edge-based color constancy. *IEEE Transactions on image processing*, 16(9):2207–2214, 2007. 1, 6
- [67] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. P+: Extended textual conditioning in text-to-image generation. 2023. 3
- [68] Guangcong Wang, Yinuo Yang, Chen Change Loy, and Ziwei Liu. Stylelight: Hdr panorama generation for lighting estimation and editing. In *European Conference on Computer Vision*, pages 477–492. Springer, 2022. 2
- [69] Henrique Weber, Donald Prévost, and Jean-François Lalonde. Learning to estimate indoor lighting from 3d objects. In *2018 International Conference on 3D Vision (3DV)*, pages 199–207. IEEE, 2018. 2
- [70] Sung-Min Woo, Sang-Ho Lee, Jun-Sang Yoo, and Jong-Ok Kim. Improving color constancy in an ambient light environment using the phong reflection model. *IEEE Transactions on Image Processing*, 27(4):1862–1877, 2018. 6
- [71] Bolei Xu, Jingxin Liu, Xianxu Hou, Bozhi Liu, and Guoping Qiu. End-to-end illuminant estimation based on deep metric learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020. 2
- [72] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18381–18391, 2023. 2
- [73] Kai-Fu Yang, Shao-Bing Gao, and Yong-Jie Li. Efficient illuminant estimation for color constancy using grey pixels. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2254–2263, 2015. 6
- [74] Hu Ye, Jun Zhang, Sibio Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. 2023. 2
- [75] Renjiao Yi, Chenyang Zhu, Ping Tan, and Stephen Lin. Faces as lighting probes via unsupervised deep highlight extraction. In *Proceedings of the European Conference on computer vision (ECCV)*, pages 317–333, 2018. 2
- [76] Huanglin Yu, Ke Chen, Kaiqi Wang, Yanlin Qian, Zhaoxiang Zhang, and Kui Jia. Cascading convolutional color constancy. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 12725–12732, 2020. 2, 6, 7
- [77] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 2

Overview

This supplementary material presents additional details and results to complement the main manuscript. In Section A, we provide comprehensive implementation details, including dataset preprocessing protocols and training configurations. Section B presents an empirical analysis of the impact of different pyramid levels in our Laplacian decomposition technique and provides implementation details of the algorithm. Section C showcases qualitative results demonstrating our method’s effectiveness across various datasets and real-world scenarios. We will release our complete training and inference code along with pre-trained weights to facilitate future research in this area.

A. Implementation Details

A.1. Datasets and Preprocessing

We use two publicly available color constancy benchmark datasets in our experiments: the NUS-8 dataset [19] and the Gehler dataset [33]. The Gehler dataset [33] contains 568 original images captured by two different cameras, while the NUS-8 dataset [19] contains 1736 original images captured by eight different cameras. Each image in both datasets includes a Macbeth Color Checker (MCC) chart, which serves as a reference for the ground-truth illuminant color.

Following the evaluation protocol in [1], several standard metrics are reported in terms of angular error in degrees: mean, median, tri-mean of all the errors, the mean of the lowest 25% of errors, and the mean of the highest 25% of errors.

A.2. Training Details

For all experiments, we process the raw image data before applying gamma correction for sRGB space conversion following the preprocessing protocol from [39]. Since the pre-trained VAE was trained on sRGB images, we apply a gamma correction of $\gamma = 1/2.2$ on linear RGB images before encoding to minimize the domain gap. Conversely, after VAE decoding, we apply inverse gamma correction to convert the output back to the linear domain for metric evaluation.

All experiments are trained for 20000 iterations on an NVIDIA A6000 GPU using the Adam optimizer with an initial learning rate of 5×10^{-5} and apply exponential learning rate decay after a 150-step warm-up period. For data augmentation, we follow FC4 [39] to rescale images by random RGB values in $[0.6, 1.4]$, noting that we only rescale the input images since our training does not require ground truth illumination. The rescaling is performed in the raw domain, followed by gamma correction. This is implemented through a 3×3 color transformation matrix, where diagonal

elements control the intensity of individual RGB channels (color strength), and off-diagonal elements determine the degree of color mixing between channels (color offdiag). For Laplacian decomposition, we use a two-level pyramid ($L = 2$) to balance the preservation of high-frequency structural details and the suppression of low-frequency color information. Additionally, we apply local transformations to masked regions only, including brightness adjustment ($[0.8, 2.0]$), saturation adjustment ($[0.8, 1.4]$), and contrast adjustment ($[0.8, 1.4]$).

Three-fold Cross-validation For cross-validation experiments on both the NUS-8 dataset [19] and the Gehler dataset [33], we use a batch size of 8. During training, we apply random crop with a probability of $p_{crop} = 0.7$, where the crop size ranges from 70% to 100% of the original dimensions. Color augmentation is applied with a probability of $p_{color} = 0.3$.

Leave-one-out Evaluation For the leave-one-out experiments on the NUS-8 dataset [19], we use a batch size of 8 with gradient accumulation over 2 steps (effective batch size of 16). We apply random crop with a probability of $p_{crop} = 0.75$, where the crop size ranges from 70% to 100% of the original image dimensions, and color augmentation with a probability of $p_{color} = 0.65$.

For the Gehler dataset [33], when training on Canon5D and evaluating on Canon1D, we use a batch size of 8, apply random crop with a probability of $p_{crop} = 0.75$ (crop size from 70% to 100%), and color augmentation with a probability of $p_{color} = 0.85$. Similarly, when training on Canon1D and evaluating on Canon5D, we maintain the same batch size of 8, with random crop probability of $p_{crop} = 0.7$ and crop size ranging from 50% to 100%, while keeping the color augmentation probability at $p_{color} = 0.85$.

Cross-dataset Evaluation When training on NUS-8 [19] and testing on the Gehler dataset [33], we use a batch size of 8 with gradient accumulation over 2 steps (effective batch size of 16). We apply random crop with a probability of $p_{crop} = 0.75$, where the crop size ranges from 70% to 100% of the original dimensions, and color augmentation with a probability of $p_{color} = 0.6$. Conversely, when training on the Gehler dataset [33] and testing on NUS-8 [19], we use a batch size of 8 without gradient accumulation. We apply random crop with the same probability of $p_{crop} = 0.75$ and size range of 70% to 100%, while color augmentation is applied with a probability of $p_{color} = 1.0$.

SDXL Inpainting (SDEdit) For the SDXL inpainting model [60] with LoRA fine-tuning experiments, we use a learning rate of 5×10^{-5} and a LoRA rank of 4. In the

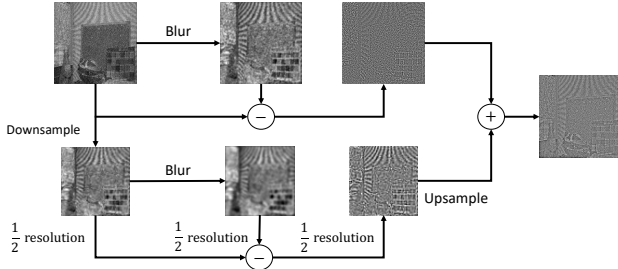


Figure 8. **Flow diagram of Laplacian decomposition.** Frequency component fusion through two-level ($1/2$ resolution) blur, down-sample, and composition operations.

cross-dataset experiment from the NUS-8 dataset [19] to the Gehler dataset [33], we train for 20,000 iterations with batch size 4.

A.3. Inference Settings

Full Model Following Garcia et al. [30], we employ DDIM scheduler with a fixed timestep $t = T$ and **trailing** strategy during inference for deterministic single-step generation. Our implementation is based on the stable-diffusion-2-inpainting model [60].

SDXL Inpainting (SDEdit) For comparison, we also implement a version using SDXL inpainting model [60] with LoRA [38] fine-tuning. During inference, we use the DDIM scheduler with 25 denoising steps and SDEdit with a noise strength of 0.6, a guidance scale of 7.5, and a LoRA scale of 1. The final illumination estimation is obtained by computing the median from an ensemble of 10 generated samples.

B. Laplacian Decomposition

B.1. Laplacian Decomposition Visualization

Figure 8 visualizes the algorithm flow of our Laplacian decomposition technique. Algorithm 1 outlines the detailed steps of this process, which preserves high-frequency structural details while allowing illumination-dependent color adaptation, enabling accurate scene illumination estimation.

B.2. Analysis of Pyramid Level Selection

We conduct experiments with different numbers of pyramid levels ($L = 1, 2, 3$) to analyze the effectiveness of our Laplacian decomposition. As shown in Tab. 9, using two-level decomposition ($L = 2$) achieves the best performance across all metrics. Adding more levels not only increases computational complexity but also leads to performance degradation,

Algorithm 1: High-frequency Extraction via Laplacian Pyramid

Input: Input latent $z \in \mathbb{R}^{B \times C \times H \times W}$, pyramid levels L

Output: High-frequency components z_h

Initialize $z_h = 0$

$k \leftarrow 3 \times 3$ Gaussian kernel

for each channel c in C **do**

$z_{\text{curr}} \leftarrow z[c]$ // Current level features

for $l = 0$ to $L - 1$ **do**

$z_{\text{blur}} \leftarrow k * z_{\text{curr}}$ // Gaussian blur

$z_{\text{high}} \leftarrow z_{\text{curr}} - z_{\text{blur}}$ // High-freq details

if $l = 0$ **then**

$z_h[c] \leftarrow z_{\text{high}}$

else

$z_h[c] \leftarrow z_h[c] + \text{Upsample}(z_{\text{high}})$

end

$z_{\text{curr}} \leftarrow \text{AvgPool}(z_{\text{blur}})$ // Downsample

end

end
return z_h

as the additional levels introduce more low-frequency information that can adversely affect the harmonious generation of color checkers.

C. Additional Qualitative Results

C.1. Benchmark Datasets

On the NUS-8 dataset [19] and Gehler dataset [33], we utilize the original mask locations to place fixed-size neutral color checkers in our experiments. The results Fig. 10 and Fig. 11 demonstrate our method’s ability to generate structurally coherent color checkers that naturally blend with the scene while accurately reflecting local illumination conditions, enabling effective color cast removal across diverse lighting scenarios.

C.2. In-the-wild Images

For in-the-wild scenes, we adopt a center-aligned placement strategy to address camera vignetting effects, which can impact color accuracy near image edges. This consistent central positioning not only mitigates lens shading issues but also demonstrates our method’s flexibility in color checker placement. The results Fig. 12 validate our approach’s robustness in practical photography applications, showing consistent performance in white balance correction despite the fixed central placement strategy.

C.3. Interactive Visualization

We provide an interactive HTML interface that visualizes results with color checkers placed at different locations within scenes. The visualization demonstrates that our method produces accurate outputs with minimal variation across different placement positions. The results show that the estimated



Figure 9. **Failure cases.** Our approach struggles when there is a significant mismatch between the illumination of the original color checker and the ambient lighting in the scene.

Table 9. Analysis of different pyramid levels in Laplacian composition. Results are trained on the NUS-8 dataset [19] and tested on Gehler dataset [33].

Level	Mean	Median	Best-25%	Worst-25%
L = 1	3.53	3.27	1.48	6.03
L = 2	2.35	2.02	0.78	4.57
L = 3	3.16	2.83	1.25	5.62

illumination values consistently cluster near the ground truth target regardless of the checker’s position, confirming our method’s reliability and position-independence in illumination estimation.

D. Limitations

As shown in Fig. 9, our method struggles when there is a significant mismatch between the inpainted color checker and the scene’s ambient lighting. This typically occurs in challenging scenarios with multiple strong light sources of different colors or complex spatially-varying illumination. While diffusion models provide strong image priors, they sometimes prioritize visual plausibility over physical accuracy, especially in extreme lighting conditions.

Our approach also shows sensitivity to dataset size, similar to personalization effects observed in DreamBooth [62]. For datasets with limited samples, we need to crop smaller mask regions to ensure the model can effectively learn the color checker’s appearance and structure. In our experiments, we found that when the training dataset is extremely small, the model generates color checkers with unexpected appearances and distorted structures, preventing accurate color extraction for illumination estimation. This limitation suggests potential future directions for improving our method through more efficient learning strategies or additional data augmentation techniques to better handle scenarios with limited training data.



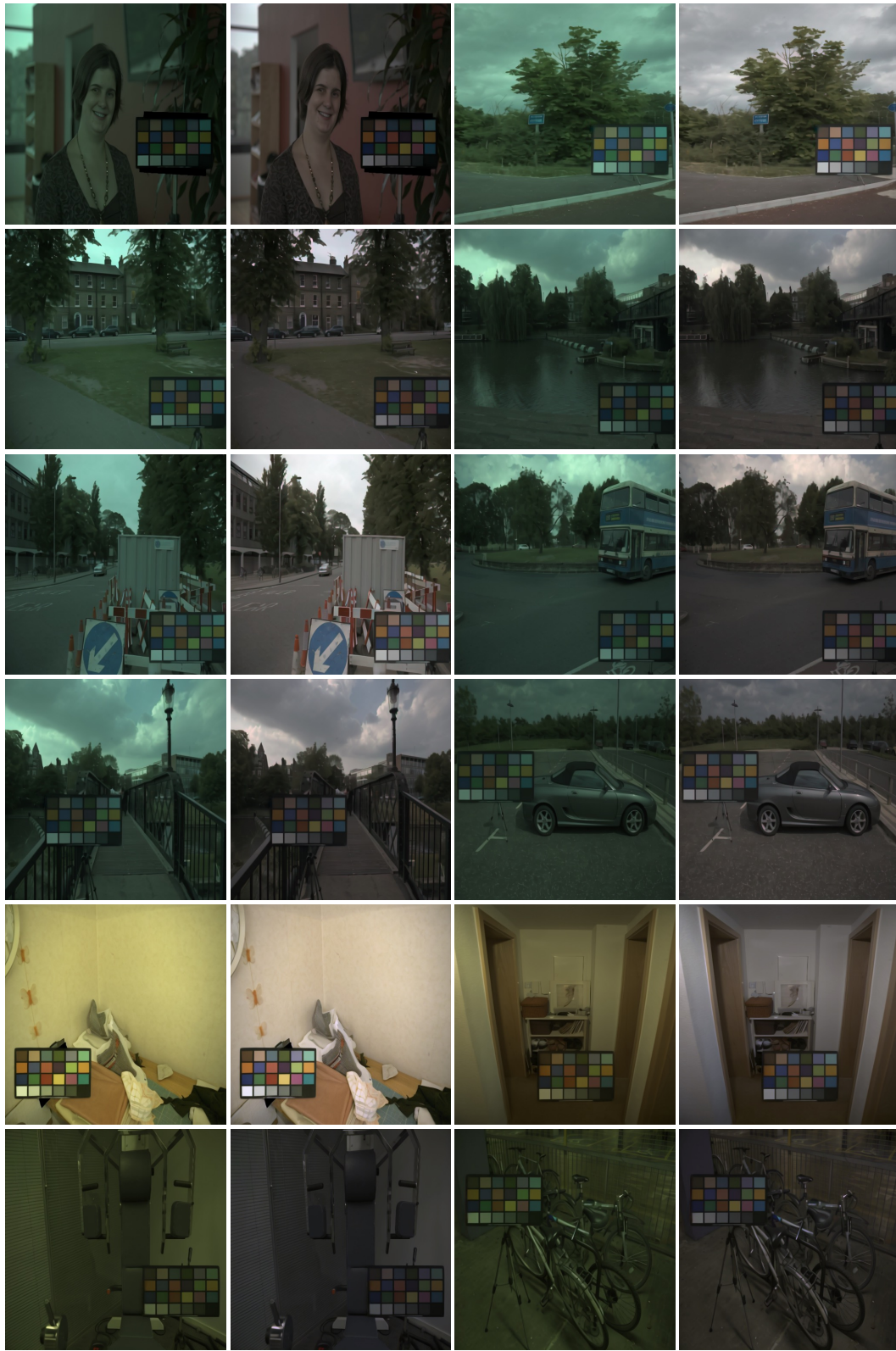
Inpainted color checker

Color cast removed

Inpainted color checker

Color cast removed

Figure 10. Qualitative results for the NUS-8 dataset [19].



Inpainted color checker

Color cast removed

Inpainted color checker

Color cast removed

Figure 11. Qualitative results for the Gehler dataset [33].



Inpainted color checker

Color cast removed

Inpainted color checker

Color cast removed

Figure 12. Qualitative results for in-the-wild images with center-placed color checkers.