DISTINGUISHING INTRINSIC SCENE COLORS FROM COLOR CASTS FOR SRGB WHITE BALANCE CORRECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

White balance (WB) correction for sRGB images requires removing color casts introduced by inaccurate WB settings while preserving intrinsic scene colors. Although existing methods achieve competitive WB correction performance, nonlinear image signal (ISP) processing entangles intrinsic scene colors with color casts, making it difficult to determine which chromatic variations should be corrected and which should be preserved, especially when large regions of intrinsic scene color dominate the image and resemble color casts, which can lead to inaccurate correction. To alleviate this correction ambiguity, we introduce the *color cast state* (CCS), an image-level representation that explicitly characterizes the tendency and severity of color casts, and propose a CCS-guided framework for sRGB white balance correction. Specifically, we learn a structured CCS representation from paired incorrectly white-balanced and ground-truth images using chromatic tendency and severity supervision together with soft similarity learning, thereby preserving continuous relationships among different color cast states. The learned CCS is then used as an explicit condition for the correction network, enabling correction that better preserves intrinsic scene colors while removing color casts. Extensive experiments on public benchmark datasets demonstrate that our method achieves state-of-the-art performance and consistently performs favorably under different WB settings, delivering more visually pleasing correction results.

1 INTRODUCTION

White balance (WB) correction is a fundamental component of camera ISP pipelines, aiming to recover natural scene colors under varying lighting conditions. While WB correction in the RAW domain can often be modeled using approximately linear transformations, correcting rendered sRGB images is substantially more challenging because nonlinear ISP operations alter the original color relationships (Afifi et al., 2019; Zhou et al., 2025). Residual color bias not only degrades visual appearance, but may also affect downstream classification, object detection, and semantic segmentation (Funt et al., 1998; Afifi & Brown, 2019; Fan et al., 2024).

Existing studies have extensively explored sRGB-domain white balance correction. Unlike RAW-domain correction, where white balance can often be approximated under relatively linear sensor responses, sRGB-WB correction operates on images that have undergone nonlinear ISP rendering. A central challenge is therefore determining *which observed color variations should be corrected and which should be preserved.* This ambiguity arises because similar chromatic appearances can originate from either intrinsic scene colors or color casts, despite requiring fundamentally different correction. Such cases become especially severe when dominant intrinsic scene colors resemble the color casts introduced by incorrect white balance. As illustrated in Fig. 1, both examples exhibit blue-dominant color statistics, yet the blue sky in the first image is intrinsic scene color that should be preserved, whereas the blue appearance of the building facade in the second image mainly arises from incorrect white balance and should be corrected. However, most existing sRGB white balance methods (Afifi & Brown, 2020; Li et al., 2023a;b; Cheng et al., 2026) learn a unified end-to-end mapping that jointly infers color casts and restores scene colors. Some methods (Peng & Chen, 2025; Chiu et al., 2025) further incorporate global color statistics as auxiliary cues, but such descriptors

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

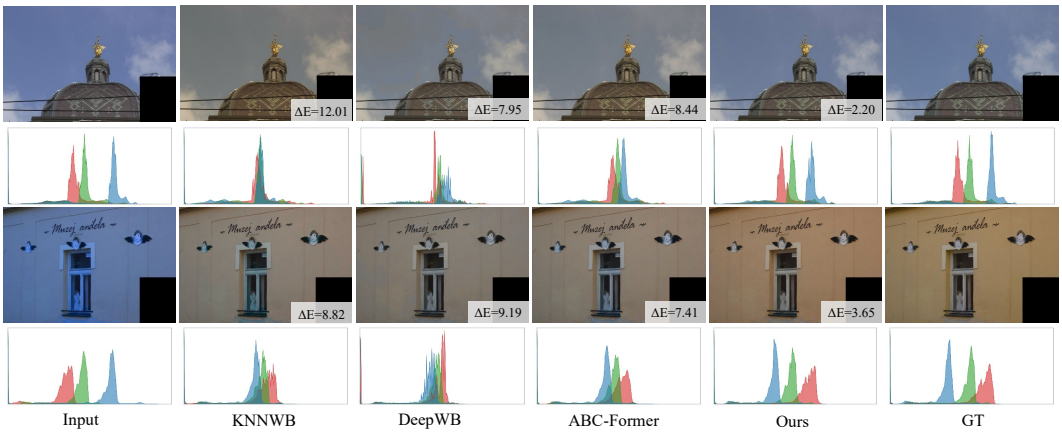


Figure 1: **Distinguishing intrinsic scene colors from color casts is especially challenging when dominant intrinsic scene colors resemble color casts.** RGB histograms are plotted below each image. Both inputs are blue-dominant. The blue sky is intrinsic scene color and should be preserved, whereas the blue cast on the building facade caused by incorrect WB needs to be corrected. Therefore, global color statistics alone may be insufficient to determine whether a dominant chromatic pattern originates from scene content or incorrect white balance.

characterize the color distribution rather than the source of the chromatic deviation, and therefore are biased when dominant intrinsic scene colors resemble color casts.

To tackle these issues, we introduce the *color cast state* (CCS), an image-level representation designed to explicitly characterize color cast. Unlike statistics computed directly from the image distribution, CCS is learned from paired WB renderings to represent the correction-relevant deviation from the correctly white-balanced image. Its representation space is therefore supervised by color casts rather than by the color distribution alone. We further propose CCSG-WB, a CCS-guided framework that first learns a structured CCS representation and then uses it as an explicit condition for white balance correction. Specifically, in the color cast state representation learning (CCSRL) stage, The paired input and ground-truth images provide direct supervision for color casts. The learned representation is encouraged to encode color cast tendency and severity while preserving continuous relationships among different color cast states. The resulting CCS representation is then injected into the correction network through color cast state-guided (CCSG) blocks, where it generates feature-wise modulation parameters for intermediate features. In this way, a shared correction backbone can adapt its feature responses according to the color cast state of each input, allowing different color casts to induce different correction behaviors while preserving intrinsic scene color. Extensive experiments also demonstrate the effectiveness of explicitly modeling color cast state.

Our main contributions are summarized as follows:

- We introduce *color cast state* (CCS) as an explicit, correction-relevant representation and formulate sRGB-WB correction as a CCS-guided correction process.
- We propose a structured CCS representation learning scheme capturing color cast tendency and severity, while preserving continuous relationships across diverse color cast states.
- We develop CCSG-WB, which conditions a shared correction backbone on the learned CCS through feature-wise modulation, enabling state-adaptive white balance correction across different color cast conditions.
- Extensive experiments on multiple benchmark datasets demonstrate state-of-the-art correction performance, together with consistent effectiveness across diverse WB settings.

2 Related Work

RAW-based Methods. Historically, RAW-based WB correction has mainly relied on illuminant estimation under approximately linear sensor responses. Traditional approaches use statistical or

physical priors of natural scenes to estimate the scene illuminant, after which channel-wise gains are applied for color correction (Buchsbaum, 1980; Forsyth, 1990; Finlayson & Trezzi, 2004; Van De Weijer et al., 2007; Qian et al., 2019). With the development of learning-based color constancy, data-driven models have been introduced to achieve more robust illuminant estimation. These approaches have explored discriminative estimation in log-chrominance space, frequency-domain inference, confidence-weighted aggregation, contrastive representation learning, and cross-camera or challenging-domain generalization, among other directions (Barron, 2015; Barron & Tsai, 2017; Hu et al., 2017; Lo et al., 2021; Afifi et al., 2021; Li & Tan, 2024; Liang et al., 2025; Li et al., 2026). Despite their different formulations, these methods generally rely on RAW or approximately linear image formation and ultimately estimate the scene illuminant or corresponding correction gains. Their applicability to sRGB images is therefore limited, since RAW data are often unavailable and nonlinear ISP operations, such as color correction, tone mapping, and gamma encoding, alter the original linear color relationships, making RAW-domain correction difficult to directly transfer to rendered sRGB images, thereby motivating methods that operate directly in the sRGB domain.

sRGB-based Methods. To address the limitations of RAW-based WB methods, recent research has explored sRGB-WB solutions that conduct color correction following ISP rendering. Early work retrieves similarly colored exemplars and estimates nonlinear polynomial mappings between incorrect and target WB renderings (Afifi et al., 2019). Deep White-Balance Editing subsequently learns direct mappings among camera WB settings with encoder–decoder networks (Afifi & Brown, 2020). For mixed illumination, MixedWB predicts pixel-wise weights to combine several preset WB renderings, while StyleWB models scene lighting as a style factor to improve this fusion process (Afifi et al., 2022; Kınlı et al., 2023). Later approaches improve correction through alternative representations and architectures. WBFlow uses reversible neural flows to map sRGB images into a pseudo-RAW space and supports camera adaptation from few examples, whereas SWBNet uses frequency-domain features and contrastive learning to reduce sensitivity to color temperature (Li et al., 2023a;b). HVDualformer couples histogram and visual features, and ABC-Former introduces auxiliary RGB and CIELab histogram branches with cross-domain interaction (Peng & Chen, 2025; Chiu et al., 2025). DCLAN learns a perception-inspired color space to coordinate chromatic and luminance processing (Cheng et al., 2026). Unlike existing sRGB-WB methods that mainly improve the correction mapping or feature processing itself, our method explicitly learns an image-level color cast state as a correction condition, providing additional guidance for determining which chromatic variations should be corrected and which should be preserved.

3 METHOD

Figure 2 presents an overview of our framework, which consists of Color Cast State Representation Learning (CCSRL) and CCS-guided white balance correction. Training proceeds in two stages: the CCS encoder first learns structured color cast representations and is then frozen to guide the training of the correction network. The CCS characterizes the color casts that need to be corrected, while the correction network preserves the intrinsic scene colors.

3.1 STAGE I: COLOR CAST STATE REPRESENTATION LEARNING

A reliable CCS representation $\boldsymbol{z}_{\mathrm{CCS}}$ should capture color casts while reducing ambiguity caused by intrinsic scene colors, thereby providing explicit guidance for the correction. We thus learn the representation using the input image and its ground truth. Since the two images originate from the same raw frame rendered under distinct WB settings, they share identical image content, and their chromatic discrepancies deliver supervision for color casts caused by incorrect white balance.

We instantiate the CCS encoder with a pretrained ViT-B/32 backbone (Dosovitskiy et al., 2021) and adapt it using LoRA (Hu et al., 2022). LoRA adapters are inserted into the query and value projections of its self-attention layers. The proposed CCS representation learning scheme enables the features output by the encoder to better capture fine-grained variations of input color cast states.

Color Cast Tendency Supervision. To encourage the CCS representation to focus on color cast tendencies induced by incorrect white balance rather than image content, we construct a color cast supervision target from the paired input image $\boldsymbol{I}$ and the ground-truth image $\boldsymbol{I}_{\mathrm{gt}}$. This paired

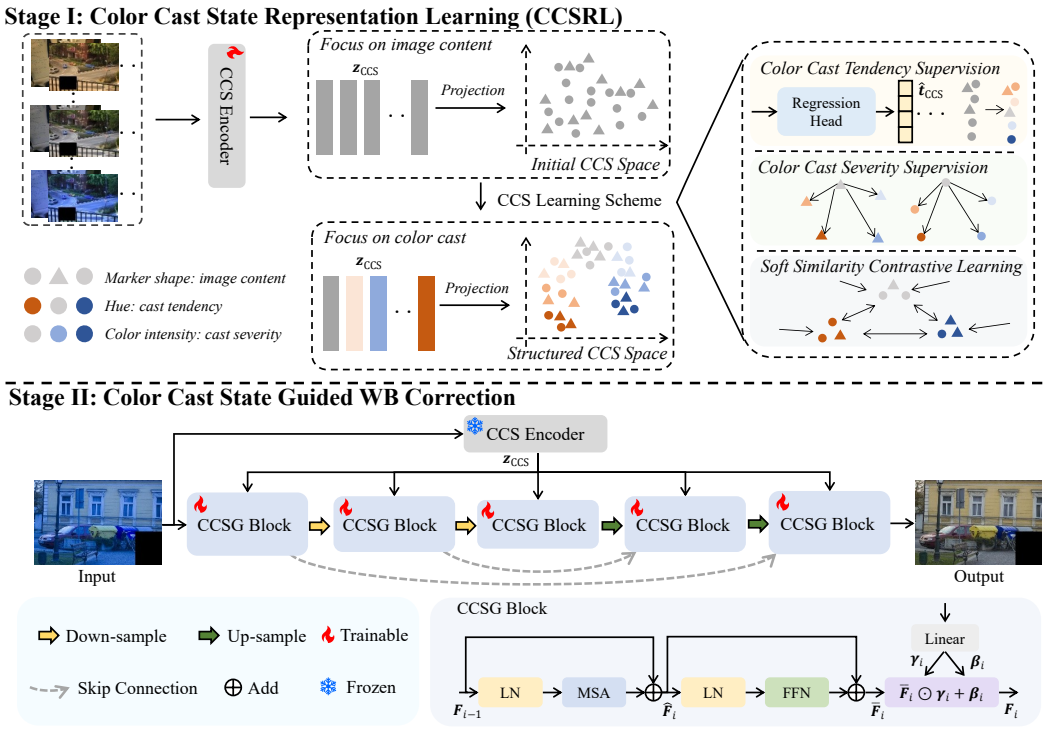


Figure 2: Overview of the proposed framework. Stage I: CCSRL constructs a structured color cast state space via the CCS learning scheme. **Different marker shapes represent distinct image contents, marker colors indicate color cast trends, and color intensity denotes color cast severity.** In Stage II, the CCS encoder provides the CCS representation of the input to the correction network, where CCSG blocks employ FiLM modulation to adapt intermediate features to different color casts.

construction reduces the influence of shared scene-dependent information and captures correction-relevant chromatic variations more directly, thereby guiding the CCS representation toward color cast information instead of scene semantics.

We characterize this discrepancy in both the Lab and YUV color spaces. Specifically, the a^* and b^* channels of Lab capture perceptually relevant chromatic offsets, while the chrominance channels u and v of YUV describe color variations from an image-signal-processing perspective. Together, these complementary spaces provide a comprehensive description of the color cast tendency. Before constructing the CCS target, the chromatic offsets are normalized according to the nominal ranges of the corresponding Lab and YUV channels to ensure comparable numerical scales. For $q \in \{a^*, b^*, u, v\}$, the average chromatic offset and the resulting color cast target are jointly defined as

$$\Delta q = \frac{1}{HW} \sum_{x,y} \left[q\big(\boldsymbol{I}(x,y)\big) - q\big(\boldsymbol{I}_{\mathrm{gt}}(x,y)\big) \right], \qquad \boldsymbol{t}_{\mathrm{CCS}} = [\Delta a^*, \Delta b^*, \Delta u, \Delta v]. \tag{1}$$

During CCSRL, we extract the CCS representation. An auxiliary regression head R_t is adopted to predict the color cast target, driving the representation to capture color cast tendencies.

$$\hat{\boldsymbol{t}}_{\mathrm{CCS}} = R_t(\boldsymbol{z}_{\mathrm{CCS}}), \qquad \mathcal{L}_{\mathrm{t}} = \mathrm{SmoothL1}(\hat{\boldsymbol{t}}_{\mathrm{CCS}}, \boldsymbol{t}_{\mathrm{CCS}}). \tag{2}$$

The tendency loss encourages the CCS representation to preserve the chromatic variations induced by incorrect white balance, thereby enabling the representation to reflect the color cast tendency.

Color Cast Severity Supervision. Besides the color cast tendency, the magnitude of the color cast is also important for adaptive correction. We measure the cast severity using the average color difference between the input and its ground truth in normalized Lab space:

$$s_{\mathrm{CCS}} = \frac{1}{HW} \sum_{x=1}^{H} \sum_{y=1}^{W} \left\| \mathrm{Lab}(\boldsymbol{I}(x,y)) - \mathrm{Lab}(\boldsymbol{I}_{\mathrm{gt}}(x,y)) \right\|_2 . \tag{3}$$

To further encode the severity information into the CCS representation space, we constrain the representation displacement between the input and its reference:

$$\mathcal{L}_{\mathrm{s}} = \mathrm{SmoothL1}\left(\left\| \boldsymbol{z} - \boldsymbol{z}_{\mathrm{gt}} \right\|_2 , s_{\mathrm{CCS}} \right), \tag{4}$$

where $\boldsymbol{z}$ and $\boldsymbol{z}_{\mathrm{gt}}$ denote the CCS representations of $\boldsymbol{I}$ and $\boldsymbol{I}_{\mathrm{gt}}$. We use their representation displacement to encode the severity of the color cast, encouraging mildly shifted inputs to remain closer to their references, while stronger color casts induce larger representation displacements.

Soft Similarity Contrastive Learning. The tendency and severity losses characterize individual samples, but effective conditioning also benefits from a representation space whose neighborhood structure reflects relationships among different color cast states. In particular, inputs with similar color cast states should provide similar correction guidance.

We therefore introduce a soft contrastive learning loss (Liang et al., 2024). Color cast states vary continuously and cannot be appropriately modeled using hard positive–negative assignments (Khosla et al., 2020). Following Similarity Contrastive Estimation (SCE) (Denize et al., 2023), we construct soft targets based on the similarity between color cast states, encouraging the representation similarity to reflect their continuous chromatic relationships.

For each sample pair, we compute its distance in the CCS space and normalize the distances into a soft similarity distribution:

$$d_{ij}^{t} = \left\| \boldsymbol{t}_{\mathrm{CCS},i} - \boldsymbol{t}_{\mathrm{CCS},j} \right\|_2^2 , \qquad p_{ij}^{t} = \frac{\exp(-d_{ij}^{t})}{\sum_{k \neq i} \exp(-d_{ik}^{t})}, \quad j \neq i, \tag{5}$$

We perform ℓ_2 normalization on the raw CCS representations to obtain normalized features $\bar{\boldsymbol{z}}_i$. The corresponding representation-space distribution is defined as

$$p_{ij}^{z} = \frac{\exp(\bar{\boldsymbol{z}}_i^{\top} \bar{\boldsymbol{z}}_j / \tau)}{\sum_{k \neq i} \exp(\bar{\boldsymbol{z}}_i^{\top} \bar{\boldsymbol{z}}_k / \tau)}, \tag{6}$$

where τ is the contrastive temperature. Following the SCE (Denize et al., 2023), we align the learned similarity distribution with the target distribution using

$$\mathcal{L}_{\mathrm{c}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j \neq i} p_{ij}^{t} \log p_{ij}^{z}. \tag{7}$$

This soft contrastive learning loss encourages the neighborhood structure of the representation space to follow that of the color cast target space. Samples with similar color casts yield similar representations, while increasingly different color cast states are progressively separated. Unlike SWBNet (Li et al., 2023b), which learns color temperature invariant features by aligning different renderings of the same scene, our method learns color cast sensitive representations to explicitly guide white balance correction, preserving rather than discarding color cast information.

The overall loss for CCS representation learning is

$$\mathcal{L}_{\mathrm{CCS}} = \lambda_{\mathrm{t}} \mathcal{L}_{\mathrm{t}} + \lambda_{\mathrm{s}} \mathcal{L}_{\mathrm{s}} + \lambda_{\mathrm{c}} \mathcal{L}_{\mathrm{c}}. \tag{8}$$

After this stage, the regression head is discarded, and the CCS encoder is frozen to extract the CCS representation of input images, which serves as an explicit condition to guide the subsequent white balance correction network.

3.2 STAGE II: CCS-GUIDED WHITE BALANCE CORRECTION

The correction network adopts a U-shaped Transformer architecture (Wang et al., 2022; Zamir et al., 2022) composed of multiple color cast state guided (CCSG) blocks. The encoder extracts multi-scale image features, while the decoder progressively reconstructs the correctly white-balanced image through upsampling and skip connections, with CCS injected at each block to guide the correction.

During training of the white balance correction network, we adopt the frozen CCS encoder. After feeding images into the CCS encoder, we acquire the CCS representation $\boldsymbol{z}_{\mathrm{CCS}} \in \mathbb{R}^{128}$ for each input image. The $\boldsymbol{z}_{\mathrm{CCS}}$ is introduced at multiple feature levels through the CCSG blocks. Each CCSG block consists of layer normalization (LN), multi-head self-attention (MSA), a feed-forward network (FFN) (Vaswani et al., 2017), and FiLM modulation (Perez et al., 2018). Given the input feature $\boldsymbol{F}_{i-1}$ of the i-th CCSG block, its feature transformation is formulated as

$$\widehat{\boldsymbol{F}}_i = \mathrm{MSA}\big(\mathrm{LN}(\boldsymbol{F}_{i-1})\big) + \boldsymbol{F}_{i-1}, \qquad \overline{\boldsymbol{F}}_i = \mathrm{FFN}\big(\mathrm{LN}(\widehat{\boldsymbol{F}}_i)\big) + \widehat{\boldsymbol{F}}_i. \tag{9}$$

Specifically, we adopt transposed self-attention for the MSA, which performs attention calculations along feature channels rather than spatial dimensions and improves computational efficiency.

The CCS representation is then mapped to feature-wise scaling and shifting parameters through a linear layer:

$$[\boldsymbol{\gamma}_i, \boldsymbol{\beta}_i] = \mathrm{Linear}_i(\boldsymbol{z}_{\mathrm{CCS}}), \tag{10}$$

where Linear_i outputs $2C_i$ parameters, which are split into $\boldsymbol{\gamma}_i, \boldsymbol{\beta}_i \in \mathbb{R}^{C_i}$, and C_i denotes the channel dimension of $\overline{\boldsymbol{F}}_i$. The block output is obtained through FiLM modulation:

$$\boldsymbol{F}_i = \overline{\boldsymbol{F}}_i \odot \boldsymbol{\gamma}_i + \boldsymbol{\beta}_i, \tag{11}$$

where $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$ are broadcast along the spatial dimensions. Different CCS representations generate different modulation parameters, allowing the shared correction backbone to adapt its intermediate feature responses to different input color casts. Finally, the corrected result is predicted as

$$\hat{\boldsymbol{I}} = G_{\boldsymbol{\theta}}(\boldsymbol{I}, \boldsymbol{z}_{\mathrm{CCS}}). \tag{12}$$

We optimize the white balance correction network via an ℓ_1 reconstruction loss.

$$\mathcal{L}_{\mathrm{rec}} = \left\| \hat{\boldsymbol{I}} - \boldsymbol{I}_{\mathrm{gt}} \right\|_1. \tag{13}$$

In this way, the CCS provides image-level information about the required WB correction, while the U-shaped backbone performs content-adaptive white balance correction.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETTINGS

Datasets. Following Afifi et al. (2019), we conduct experiments on the Rendered WB dataset, which provides sRGB images rendered from RAW data under various WB settings. For training, we randomly select 12,000 images from the fold2 and fold3 of Set 1. For evaluation, we use three benchmark sets: Set 1-Test, Set 2, and the Rendered Cube+ dataset (Afifi et al., 2019).

Evaluation Metrics. We evaluate the differences between corrected images and ground truth images using three widely used metrics following Afifi et al. (2019): mean squared error (MSE), mean angular error (MAE), and ΔE_{2000} (Sharma et al., 2005). For each evaluation metric, we report the mean, lower quartile (Q_1), median (Q_2), and the upper quartile (Q_3) of the error. After resizing all input images to a maximum dimension of 656 pixels, we evaluate the average single-image inference time on an NVIDIA GeForce RTX 4090 GPU for efficiency comparison.

Implementation Details. Our method is trained in two stages. The CCS encoder adopts a pre-trained ViT-B/32 backbone with LoRA adapters inserted into the query and value projections. It is trained for 20 epochs with a batch size of 64 using AdamW (Loshchilov & Hutter, 2019), a learning rate of 1×10^{-4}, and a weight decay of 1×10^{-4}. The CCS encoder is then frozen to train the white balance correction network for 250 epochs. AdamW is used with a learning rate decayed from 2×10^{-4} to 1×10^{-5} by cosine annealing (Loshchilov & Hutter, 2017). Random cropping and flipping are applied for augmentation (Afifi & Brown, 2020).

Table 1: WB results on the Rendered WB Dataset (Afifi et al., 2019) and the Rendered version of the Cube+ Dataset (Afifi et al., 2019; Banić et al., 2017). Inference time is reported in milliseconds and rounded to the nearest integer. Best and second-best results are shown in bold and underlined.

Method	MSE				MAE				ΔE_{2000}				Time
	Mean	Q1	Q2	Q3	Mean	Q1	Q2	Q3	Mean	Q1	Q2	Q3	(ms)
Rendered WB Dataset Set 1-Test (21,046 images)													
KNNWB (CVPR 2019)	77.49	13.74	39.62	94.01	3.06°	1.74°	2.54°	3.76°	3.58	2.07	3.09	4.55	159
DeepWB (CVPR 2020)	82.55	13.19	42.77	102.09	3.12°	1.88°	2.70°	3.84°	3.77	2.16	3.30	4.86	223
MixedWB (WACV 2022)	142.25	26.81	67.17	164.66	4.07°	2.64°	3.68°	5.16°	4.55	3.00	4.15	5.63	823
WBFlow (IJCAI 2023)	78.89	12.99	35.09	79.35	2.67°	1.73°	2.39°	3.24°	3.13	1.92	2.79	3.94	361
SWBNet (AAAI 2023)	111.62	20.61	60.68	137.91	4.11°	2.56°	3.75°	5.22°	4.54	2.73	4.16	5.86	288
HVDualformer (AAAI 2025)	26.01	6.69	15.29	30.49	2.30°	1.43°	1.99°	2.75°	2.47	1.63	2.19	2.97	87
ABC-Former (CVPR 2025)	20.47	4.65	10.02	21.05	1.99°	1.25°	1.73°	2.33°	2.18	1.38	1.86	2.59	126
DCLAN (WACV 2026)	54.79	10.52	26.60	59.04	2.53°	1.59°	2.26°	3.13°	2.79	1.74	2.43	3.42	296
Ours	**17.73**	**2.80**	**6.84**	**16.40**	**1.50°**	**0.91°**	**1.28°**	**1.81°**	**1.67**	**0.99**	**1.37**	**2.03**	**73**
Rendered WB Dataset Set 2 (2,881 images)													
KNNWB (CVPR 2019)	171.09	37.04	87.04	190.88	4.48°	2.26°	3.64°	5.95°	5.60	3.43	4.90	7.06	303
DeepWB (CVPR 2020)	124.07	30.13	76.32	154.44	3.75°	2.02°	3.08°	4.72°	4.90	3.13	4.35	6.08	398
MixedWB (WACV 2022)	188.76	48.64	112.32	219.91	4.92°	2.69°	4.10°	6.37°	6.05	3.45	4.92	7.20	2013
WBFlow (IJCAI 2023)	117.60	31.25	61.68	143.90	3.51°	1.93°	2.92°	4.47°	4.64	3.16	4.07	5.56	488
SWBNet (AAAI 2023)	219.02	55.45	113.98	236.25	5.46°	3.45°	4.78°	6.63°	6.51	4.39	5.84	8.08	454
HVDualformer (AAAI 2025)	119.90	29.30	63.36	148.27	3.54°	1.93°	2.82°	4.40°	4.58	3.06	4.08	5.54	246
ABC-Former (CVPR 2025)	104.31	25.55	58.61	132.90	3.39°	1.87°	2.73°	4.30°	4.56	2.97	4.13	5.63	307
DCLAN (WACV 2026)	115.53	29.97	64.63	143.34	3.56°	1.95°	2.81°	4.44°	4.55	3.05	4.14	5.55	498
Ours	**94.97**	**24.67**	**54.57**	**124.54**	**2.67°**	**1.62°**	**2.29°**	**3.31°**	**4.02**	**2.66**	**3.84**	**5.10**	**189**
Rendered Cube+ Dataset with Different WB Settings (10,242 images)													
KNNWB (CVPR 2019)	194.98	27.43	57.08	118.21	4.12°	1.96°	3.17°	5.04°	5.68	3.22	4.61	6.70	203
DeepWB (CVPR 2020)	80.46	15.43	33.88	74.42	3.45°	1.87°	2.82°	4.26°	4.59	2.68	3.81	5.53	265
MixedWB (WACV 2022)	161.80	16.96	**19.33**	90.81	4.05°	1.40°	2.12°	4.88°	4.89	**2.16**	**3.10**	6.78	959
WBFlow (IJCAI 2023)	75.39	14.22	30.90	72.91	3.34°	1.87°	2.82°	4.11°	4.28	2.68	3.77	5.21	407
SWBNet (AAAI 2023)	74.35	20.46	40.04	86.95	3.15°	**1.33°**	**2.09°**	4.12°	4.28	2.40	3.56	5.09	377
HVDualformer (AAAI 2025)	71.73	12.71	31.56	70.16	3.35°	1.68°	2.68°	4.23°	4.25	2.41	3.57	5.25	112
ABC-Former (CVPR 2025)	60.60	12.15	26.92	57.20	2.99°	1.63°	2.45°	3.69°	3.95	2.35	3.40	4.86	160
DCLAN (WACV 2026)	78.19	23.73	44.67	87.27	3.70°	2.14°	3.11°	4.51°	4.41	2.81	3.88	5.41	362
Ours	**51.70**	**11.53**	23.51	**48.78**	**2.70°**	1.53°	2.22°	**3.26°**	**3.68**	2.24	3.16	**4.45**	**78**

Table 2: Mean ΔE_{2000} comparison across different white balance settings on the Rendered Cube+ dataset. Best and second-best results are shown in bold and underlined. Images rendered via As Shot (AS) and Auto AWB (AU) are excluded due to unfixed WB settings.

Method	Mean	2850 K	3800 K	5500 K	7500 K
KNNWB (CVPR 2019)	5.88	6.44	5.64	5.60	5.86
DeepWB (CVPR 2020)	5.82	6.35	5.74	5.52	5.68
MixedWB (WACV 2022)	5.53	6.24	5.45	5.16	5.28
WBFlow (IJCAI 2023)	4.46	5.30	4.62	3.89	4.03
SWBNet (AAAI 2023)	4.88	5.21	4.70	4.75	4.89
ABC-Former (CVPR 2025)	4.09	4.75	4.28	3.55	3.77
DCLAN (WACV 2026)	4.65	5.62	4.85	3.98	4.16
Ours	**3.91**	**4.46**	**4.14**	**3.32**	**3.73**

4.2 Experimental Results

We compare the proposed method with several SOTA sRGB-WB methods, including KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), MixedWB (Afifi et al., 2022), WBFlow (Li et al., 2023a), SWBNet (Li et al., 2023b), HVDualformer (Peng & Chen, 2025), ABC-Former (Chiu et al., 2025), and DCLAN (Cheng et al., 2026).

Quantitative Analysis. As presented in Table 1, the quantitative results on Rendered WB Set 1-Test, Set 2, and Rendered Cube+ demonstrate that our method outperforms the SOTA sRGB-WB methods. Specifically, our method achieves the best results across all metrics and statistics on Set 1-Test and Set 2, including the latter captured by unseen cameras, indicating strong correction performance and cross-camera generalization. On Rendered Cube+, our method obtains the best mean results for all three metrics, while remaining competitive across the corresponding quartile

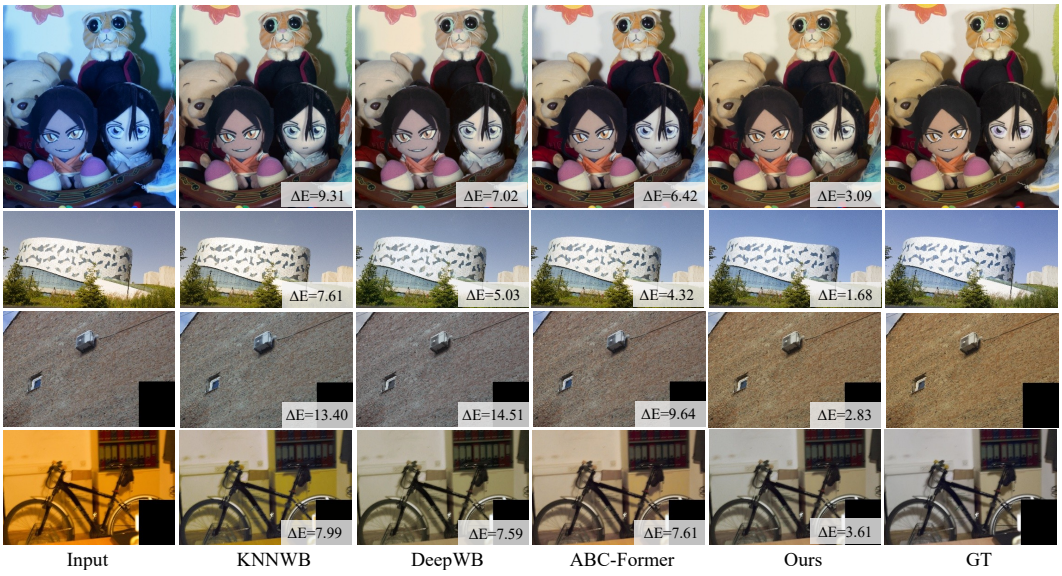


Figure 3: Qualitative comparisons on Set 2 of the Rendered WB Dataset (Afifi et al., 2019) and the rendered Cube+ Dataset (Afifi et al., 2019; Banić et al., 2017). From left to right: input, results of KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), ours, and ground truth. The ΔE_{2000} value is reported at the bottom right of each result.

statistics. Our method also achieves the fastest inference time. For CCS extraction, inputs to the CCS encoder are resized to 224×224 to effectively restrict extra computational cost. Meanwhile, transposed attention (Zamir et al., 2022) is deployed in the correction backbone. Benefiting from the above, our framework achieves minimal inference latency while introducing explicit CCS guidance.

As further shown in Table 2, our method achieves the best ΔE_{2000} under all evaluated WB settings, ranging from 2850 K to 7500 K. Notably, the improvement is most pronounced under the 5500 K setting, where the error is reduced from 3.55 to 3.32 compared with the second-best method, and our method consistently outperforms existing methods across different color temperatures. The consistent performance under different color casts indicates that the proposed method does not favor a specific WB setting. Instead, the CCS representation provides effective guidance, enabling the correction network to adapt to diverse color cast states.

Qualitative Analysis. Figure 3 presents qualitative comparisons on challenging scenes where intrinsic scene colors coexist with substantial color casts. Such cases require the correction model to accurately determine which chromatic variations originate from incorrect white balance and should be removed, while preserving colors that are intrinsic to the scene. As shown in the figure, competing methods often leave noticeable residual color casts after correction, such as bluish residuals on plush toy surfaces and yellowish biases in background regions and furniture. Residual color casts are also visible over large scene surfaces, indicating incomplete removal of the color cast. In contrast, our method effectively removes these color casts while preserving the intrinsic colors of scene objects, producing results that are visually closer to the ground truth. Moreover, our method achieves the lowest ΔE_{2000} in the images. These results further demonstrate that the learned CCS provides correction-relevant guidance for distinguishing intrinsic scene colors from color casts, enabling accurate correction of what should be removed while preserving what should remain.

Ablation Analysis. We conduct ablation studies on the Rendered Cube+ dataset to evaluate the CCS learning scheme and conditioning strategy. All variants use the same correction backbone and training settings. Following the main experiments, we report the mean of MSE, MAE, and ΔE_{2000}.

Table 3 compares different losses for CCS representation learning. The tendency and severity losses encourage the representation to encode the color cast and its severity, while the soft contrastive learning loss organizes the representation space according to color cast similarity. The soft con-

Table 3: Ablation study of different losses for CCS representation learning on the Rendered Cube+ dataset.

Loss Function	MSE↓	MAE↓	ΔE_{2000} ↓
$\mathcal{L}_t + \mathcal{L}_s$	61.80	3.02°	3.96
$\mathcal{L}_c$	54.15	2.73°	3.73
$\mathcal{L}_t + \mathcal{L}_s + \mathcal{L}_c$	**51.70**	**2.70°**	**3.68**

Table 4: Ablation study of different conditioning strategies on the Rendered Cube+ dataset.

Conditioning Strategy	MSE↓	MAE↓	ΔE_{2000} ↓
No condition	69.26	3.20°	4.21
Lab/YUV statistics	63.10	3.08°	3.99
CCS concatenation	56.02	2.74°	3.74
Ours	**51.70**	**2.70°**	**3.68**

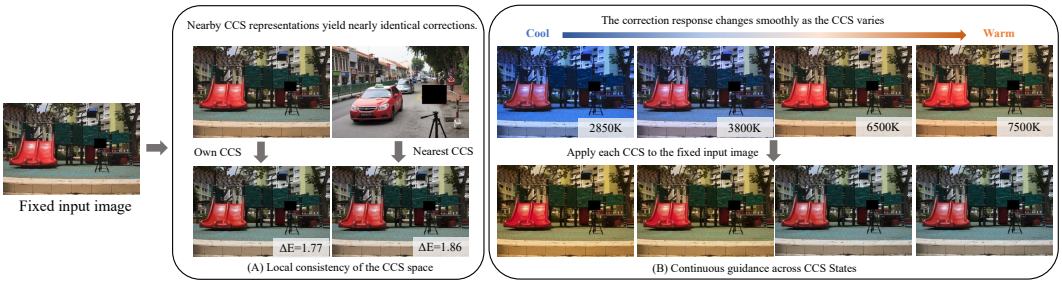


Figure 4: Analysis of the learned CCS space. The first row shows the images used to extract CCS representations, and the second row shows the correction results on the fixed input. Nearby CCS states produce similar corrections, while the correction response varies smoothly from cool to warm.

trastive learning loss achieves better performance, and combining all losses yields the best results, demonstrating that they are complementary.

Table 4 compares different conditioning strategies. As in (Peng & Chen, 2025; Chiu et al., 2025), we use the channel-wise means and variances of Lab and YUV as a statistics-based condition. Its improvement over the model without conditioning confirms the usefulness of global chromatic guidance. Replacing these statistics with the learned CCS further improves performance, indicating that CCS provides a more informative description of the input color cast. Moreover, our method outperforms direct concatenation, showing that feature-wise modulation utilizes the condition more effectively than leaving the network to implicitly learn its use.

Analysis of the Learned CCS Space and Guidance. Figure 4 analyzes the learned CCS space and its effect on white balance correction. The input image is fixed throughout the experiment. In each panel, the first row shows the images used to extract CCS representations, while the second row shows the correction results obtained by applying these representations to the fixed input.

In (A), the input CCS and its nearest neighbor in the learned space are extracted from different scenes but produce nearly identical correction results, suggesting that nearby representations mainly reflect similar color cast states rather than image content. In (B), CCS representations extracted from the same scene under WB settings ranging from 2850 K to 7500 K are applied to the fixed input. As the supplied CCS changes from cool to warm, the correction response varies smoothly. These results indicate that the learned CCS space preserves continuous color cast relationships and provides consistent guidance for adaptive white balance correction.

5 Conclusion

In this paper, we introduced CCSG-WB, a CCS-guided framework for sRGB white balance correction that explicitly represents color casts before performing correction. At its core, the proposed color cast state (CCS) captures correction-relevant information about the tendency and severity of the input color cast, while preserving continuous relationships among different color cast states. The learned CCS is then used as an explicit condition to modulate intermediate features of the correction network, enabling a shared backbone to adapt its correction behavior to different color casts while preserving intrinsic scene colors. Extensive experiments indicate that our method learns a correction-relevant CCS space and achieves SOTA performance on multiple benchmark datasets.

AI USE STATEMENT

Generative AI tools were solely utilized to assist in drafting and revising manuscript content, refining language expression, correcting grammar, and improving the overall clarity and readability of this paper. All AI-assisted texts, equations and other manuscript contents have been thoroughly reviewed, revised and verified manually by the authors. The authors take full and final responsibility for all content and findings presented in the submitted manuscript.

REPRODUCIBILITY STATEMENT

The model architecture and training objectives are described in Section 3, while the datasets, evaluation metrics, optimization settings, and training schedule are provided in Section 4. Additional ablation studies and qualitative comparisons are included in the supplementary material. These details are provided to facilitate independent reproduction of the reported results. Upon acceptance, we plan to publicly release the source code and trained model weights to further support reproducibility and future research.

REFERENCES

Mahmoud Afifi and Michael S Brown. What else can fool deep learning? addressing color constancy errors on deep neural network performance. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 243–252, 2019.

Mahmoud Afifi and Michael S Brown. Deep white-balance editing. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1394–1403. IEEE, 2020.

Mahmoud Afifi, Brian Price, Scott Cohen, and Michael S Brown. When color constancy goes wrong: Correcting improperly white-balanced images. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1535–1544. IEEE, 2019.

Mahmoud Afifi, Jonathan T Barron, Chloe LeGendre, Yun-Ta Tsai, and Francois Bleibel. Cross-camera convolutional color constancy. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1961–1970. IEEE, 2021.

Mahmoud Afifi, Marcus A Brubaker, and Michael S Brown. Auto white-balance correction for mixed-illuminant scenes. In *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 934–943. IEEE, 2022.

Nikola Banić, Karlo Koščević, and Sven Lončarić. Unsupervised learning for color constancy. *arXiv preprint arXiv:1712.00436*, 2017.

Jonathan T Barron. Convolutional color constancy. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 379–387, 2015.

Jonathan T Barron and Yun-Ta Tsai. Fast fourier color constancy. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 6950–6958. IEEE, 2017.

Gershon Buchsbaum. A spatial processor model for object colour perception. *Journal of the Franklin institute*, 310(1):1–26, 1980.

Yang Cheng, Ziteng Cui, Lin Gu, Shenghan Su, and Zenghui Zhang. Perception-inspired color space design for photo white balance editing. In *2026 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 3741–3749. IEEE, 2026.

Yu-Cheng Chiu, Guan-Rong Chen, Zihao Chen, and Yan-Tsung Peng. Abc-former: Auxiliary bimodal cross-domain transformer with interactive channel attention for white balance. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21258–21266. IEEE, 2025.

Julien Denize, Jaonary Rabarisoa, Astrid Orcesi, Romain Hérault, and Stéphane Canu. Similarity contrastive estimation for self-supervised soft contrastive learning. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2705–2715. IEEE, 2023.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=YicbFdNTTy.

Yihua Fan, Yongzhen Wang, Dong Liang, Yiping Chen, Haoran Xie, Fu Lee Wang, Jonathan Li, and Mingqiang Wei. Low-facenet: Face recognition-driven low-light image enhancement. *IEEE Transactions on Instrumentation and Measurement*, 73:1–13, 2024.

Graham D Finlayson and Elisabetta Trezzi. Shades of gray and colour constancy. In *Color and imaging conference*, volume 12, pp. 37–41. Society of Imaging Science and Technology, 2004.

David A Forsyth. A novel algorithm for color constancy. *International Journal of Computer Vision*, 5(1):5–35, 1990.

Brian Funt, Kobus Barnard, and Lindsay Martin. Is machine colour constancy good enough? In *European Conference on Computer Vision*, pp. 445–459. Springer, 1998.

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. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Yuanming Hu, Baoyuan Wang, and Stephen Lin. Fcˆ 4: Fully convolutional color constancy with confidence-weighted pooling. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 330–339. IEEE, 2017.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673, 2020.

Furkan Kınlı, Doğa Yılmaz, Barış Özcan, and Furkan Kıraç. Modeling the lighting in scenes as style for auto white-balance correction. In *2023 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 4892–4902. IEEE, 2023.

Chunxiao Li, Xuejing Kang, and Anlong Ming. Wbflow: Few-shot white balance for srgb images via reversible neural flows. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pp. 1026–1034, 2023a.

Chunxiao Li, Xuejing Kang, Zhifeng Zhang, and Anlong Ming. Swbnet: a stable white balance network for srgb images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 1278–1286, 2023b.

Shuwei Li and Robby T Tan. Nightcc: Nighttime color constancy via adaptive channel masking. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 25522–25531. IEEE, 2024.

Shuwei Li, Lei Tan, and Robby T. Tan. White-balance first, adjust later: Cross-camera color constancy via vision-language evaluation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1331–1341, June 2026.

Dong Liang, Zhengyan Xu, Ling Li, Mingqiang Wei, and Songcan Chen. Pie: Physics-inspired low-light enhancement. *International Journal of Computer Vision*, 132(9):3911–3932, 2024.

Dong Liang, Yuanhang Gao, Ling Li, Zhengyan Xu, Sheng-Jun Huang, and Songcan Chen. Aesthetics-guided low-light enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(7):5866–5883, 2025.

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 *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8049–8059. IEEE, 2021.

Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=Skq89Scxx.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Yan-Tsung Peng and Guan-Rong Chen. Hvdualformer: Histogram-vision dual transformer for white balance. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 6506–6513, 2025.

Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proceedings of the AAAI conference on artificial intelligence*, 2018.

Yanlin Qian, Joni-Kristian Kamarainen, Jarno Nikkanen, and Jiri Matas. On finding gray pixels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8062–8070, 2019.

Gaurav Sharma, Wencheng Wu, and Edul N Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color Research & Application: Endorsed by Inter-Society Color Council, The Colour Group (Great Britain), Canadian Society for Color, Color Science Association of Japan, Dutch Society for the Study of Color, The Swedish Colour Centre Foundation, Colour Society of Australia, Centre Français de la Couleur*, 30(1):21–30, 2005.

Joost Van De Weijer, Theo Gevers, and Arjan Gijsenij. Edge-based color constancy. *IEEE Transactions on image processing*, 16(9):2207–2214, 2007.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17662–17672. IEEE, 2022.

Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5728–5739, 2022.

Yuyan Zhou, Dong Liang, Songcan Chen, and Sheng-Jun Huang. Image lens flare removal using adversarial curve learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47 (9):7396–7409, 2025.

SUPPLEMENTARY MATERIAL

OVERVIEW

This supplementary material provides additional experiments and analyses that further support the findings in the main manuscript, including:

- Overall algorithm workflow;
- Extended ablation results across all evaluation datasets;
- Additional analyses of correction consistency and the learned CCS space;
- Additional qualitative comparisons across diverse scenes and WB settings.

A OVERALL ALGORITHM WORKFLOW

The complete training and inference procedure of CCSG-WB is summarized in Algorithm 1. The framework is optimized in two stages. Stage I learns a structured color cast state (CCS) representation from paired incorrectly white-balanced and ground-truth images, while Stage II uses the learned CCS as an explicit condition for white balance correction. At inference time, the frozen CCS encoder extracts the color cast state directly from the input sRGB image, which then conditions the correction network to produce the final white-balanced result .

Stage I: CCS Representation Learning. Given an incorrectly white-balanced image and its ground-truth image, the CCS encoder extracts their representations. Since the paired images originate from the same scene but differ in WB rendering, their chromatic discrepancy provides supervision for learning color casts. Tendency loss encourages the representation to encode the color cast tendency, while severity loss organizes the representation according to the color cast severity. Soft similarity contrastive learning further structures the representation space by preserving continuous relationships among different color cast states. After Stage I, the auxiliary regression head is discarded and the trained CCS encoder is frozen for the subsequent correction stage.

Stage II: CCS-Guided WB Correction. For each input image, the frozen CCS encoder first extracts its CCS representation. The same image is processed by the correction backbone, while the CCS representation is introduced into multiple CCSG blocks as an explicit condition. Each CCSG block transforms the CCS into feature-wise modulation parameters and uses them to adapt intermediate feature responses. In this way, different color cast states induce different correction behaviors within the shared backbone. The correction network is then optimized against the ground-truth image using the reconstruction loss.

Inference. During inference, an input sRGB image is first passed through the frozen CCS encoder to obtain its color cast state. The resulting CCS representation directly conditions the correction backbone to produce the final white-balanced image.

B EXTENDED ABLATION RESULTS

Table S1 extends the ablation studies in the main paper to Set 1-Test, Set 2, and Rendered Cube+ (Afifi et al., 2019; Banić et al., 2017), reporting both mean and quartile statistics for MSE, MAE, and ΔE_{2000} (Sharma et al., 2005). All variants use the same correction backbone and training settings.

The extended results further support the observations in the main paper. Combining tendency and severity losses with the soft similarity loss (Denize et al., 2023) yields the strongest overall performance profile across the evaluated datasets, demonstrating that tendency and severity supervision complement soft similarity learning by jointly characterizing individual color cast states and their relationships in the learned CCS space. Explicit conditioning generally improves over the unconditioned baseline, while replacing handcrafted Lab/YUV statistics motivated by ABC-Former (Chiu et al., 2025) with the learned CCS provides further gains. CCS-guided feature modulation achieves the best mean performance across all evaluated datasets, further validating the effectiveness of explicitly using the learned color cast state to adapt intermediate correction features.

Algorithm 1 Training and inference workflow of CCSG-WB.

Require: Paired training images $\{(\boldsymbol{I}, \boldsymbol{I}_{\mathrm{gt}})\}$; CCS encoder E_ϕ; correction network G_θ

Stage I: Color Cast State Representation Learning
for each training batch **do**
  Extract $\boldsymbol{z} = E_\phi(\boldsymbol{I})$ and $\boldsymbol{z}_{\mathrm{gt}} = E_\phi(\boldsymbol{I}_{\mathrm{gt}})$
  Compute tendency target $\boldsymbol{t}_{\mathrm{CCS}}$ from Lab/YUV chromatic offsets
  Compute severity s_{CCS} from the Lab difference
  Compute $\mathcal{L}_{\mathrm{t}}$, $\mathcal{L}_{\mathrm{s}}$, and $\mathcal{L}_{\mathrm{c}}$
  Update E_ϕ using $\mathcal{L}_{\mathrm{CCS}} = \lambda_{\mathrm{t}}\mathcal{L}_{\mathrm{t}} + \lambda_{\mathrm{s}}\mathcal{L}_{\mathrm{s}} + \lambda_{\mathrm{c}}\mathcal{L}_{\mathrm{c}}$
end for
Freeze E_ϕ and discard the auxiliary regression head

Stage II: CCS-Guided White Balance Correction
for each training pair $(\boldsymbol{I}, \boldsymbol{I}_{\mathrm{gt}})$ **do**
  $\boldsymbol{z}_{\mathrm{CCS}} \leftarrow E_\phi(\boldsymbol{I})$
  for each CCSG block i **do**
    $[\boldsymbol{\gamma}_i, \boldsymbol{\beta}_i] \leftarrow \mathrm{Linear}_i(\boldsymbol{z}_{\mathrm{CCS}})$
    Modulate intermediate features using $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$
  end for
  $\hat{\boldsymbol{I}} \leftarrow G_\theta(\boldsymbol{I}, \boldsymbol{z}_{\mathrm{CCS}})$
  Update G_θ using $\mathcal{L}_{\mathrm{rec}} = \|\hat{\boldsymbol{I}} - \boldsymbol{I}_{\mathrm{gt}}\|_1$
end for

Inference
$\boldsymbol{z}_{\mathrm{CCS}} \leftarrow E_\phi(\boldsymbol{I})$
$\hat{\boldsymbol{I}} \leftarrow G_\theta(\boldsymbol{I}, \boldsymbol{z}_{\mathrm{CCS}})$

C Additional Analysis of Correction Results and the CCS Space

Correction Consistency across WB Settings. Figure S1 provides a complementary view of the evaluation under different WB settings in the main paper by visualizing the chromaticity of corrected results in the CIELAB a^*–b^* plane. Although the five input images exhibit substantially different color casts, the outputs of CCSG-WB converge to a compact region close to the ground-truth chromaticity. In comparison, competing methods show larger deviations from the ground truth and are less consistent across WB settings. This result demonstrates that CCSG-WB produces more consistent correction outcomes across diverse WB settings, indicating that the learned CCS provides stable correction guidance under substantially different color cast conditions.

Visualization of the Learned CCS Space. Figure S2 visualizes the structure of the learned CCS representations. Despite substantial variations in scene content, samples with similar color cast states tend to occupy nearby regions in the representation space. Warm and yellowish casts are mainly located toward the left and lower-left regions, whereas cool and bluish casts are concentrated toward the right and lower-right, with smooth transitions between different states. This organization indicates that the learned representation is structured primarily by color cast characteristics rather than image content, while preserving continuous relationships among different color cast states.

D Additional Visual Results

Figures S3–S12 present additional qualitative comparisons on Set 1-Test, Set 2, and Rendered Cube+ (Afifi et al., 2019; Banić et al., 2017). The Set 1-Test results additionally include multiple WB renderings of the same scene, providing a direct visualization of correction consistency across different color cast states. Results on Set 2 further examine correction behavior on images captured by unseen cameras, while Rendered Cube+ provides additional examples across diverse indoor and outdoor scenes. Across these settings, CCSG-WB consistently removes color casts while preserving intrinsic scene colors. The corresponding ΔE_{2000} values are reported for all corrected results.

Table S1: Detailed ablation of the CCS learning objectives and conditioning strategies on the Rendered WB (Afifi et al., 2019) and Rendered Cube+ (Afifi et al., 2019; Banić et al., 2017) benchmark datasets. Lower values are better, and the best result in each column is shown in bold.

Variant	MSE				MAE				ΔE_{2000}			
	Mean	Q1	Q2	Q3	Mean	Q1	Q2	Q3	Mean	Q1	Q2	Q3
Rendered WB Dataset Set 1-Test (21,046 images)												
$\mathcal{L}_t + \mathcal{L}_s$	22.92	4.47	9.74	21.31	1.85°	1.11°	1.53°	2.17°	2.04	1.19	1.69	2.46
$\mathcal{L}_c$	19.45	3.16	8.05	18.12	1.55°	0.95°	1.33°	1.87°	1.74	1.05	1.46	2.12
No condition	26.14	5.64	12.81	27.79	1.94°	1.23°	1.64°	2.35°	2.29	1.31	1.81	2.63
Lab/YUV statistics	23.47	3.51	8.58	20.49	1.64°	1.01°	1.42°	2.01°	1.90	1.13	1.58	2.32
CCS concatenation	18.25	3.36	8.18	18.08	1.56°	0.96°	1.33°	1.88°	1.73	1.07	1.47	2.10
Ours	**17.73**	**2.80**	**6.84**	**16.40**	**1.50°**	**0.91°**	**1.28°**	**1.81°**	**1.67**	**0.99**	**1.37**	**2.03**
Rendered WB Dataset Set 2 (2,881 images)												
$\mathcal{L}_t + \mathcal{L}_s$	109.52	27.29	61.30	137.64	3.13°	1.80°	2.64°	3.91°	4.40	2.88	4.06	5.46
$\mathcal{L}_c$	98.99	24.73	54.59	137.18	2.70°	1.61°	2.32°	3.41°	4.05	2.70	3.86	5.24
No condition	119.49	27.25	62.47	146.97	3.23°	1.78°	2.62°	4.05°	4.56	2.98	4.07	5.63
Lab/YUV statistics	116.50	26.99	62.37	144.98	3.28°	1.76°	2.65°	4.11°	4.49	2.87	4.06	5.57
CCS concatenation	97.28	**22.98**	**53.48**	132.68	2.68°	**1.60°**	2.32°	3.33°	4.03	2.69	**3.80**	5.13
Ours	**94.97**	24.67	54.57	**124.54**	**2.67°**	1.62°	**2.29°**	**3.31°**	**4.02**	**2.66**	3.84	**5.10**
Rendered Cube+ Dataset with Different WB Settings (10,242 images)												
$\mathcal{L}_t + \mathcal{L}_s$	61.80	13.59	28.08	58.87	3.02°	1.63°	2.45°	3.75°	3.96	2.41	3.38	4.81
$\mathcal{L}_c$	54.15	13.08	26.45	54.88	2.73°	1.54°	2.23°	3.32°	3.73	2.32	3.18	4.52
No condition	69.26	13.06	28.42	66.32	3.20°	1.59°	2.50°	4.04°	4.21	2.40	3.51	5.18
Lab/YUV statistics	63.10	**11.46**	25.85	59.47	3.08°	1.55°	2.41°	3.89°	3.99	2.28	3.29	4.88
CCS concatenation	56.02	13.89	26.22	51.92	2.74°	1.57°	2.23°	3.29°	3.74	2.33	3.18	4.48
Ours	**51.70**	11.53	**23.51**	**48.78**	**2.70°**	**1.53°**	**2.22°**	**3.26°**	**3.68**	**2.24**	**3.16**	**4.45**

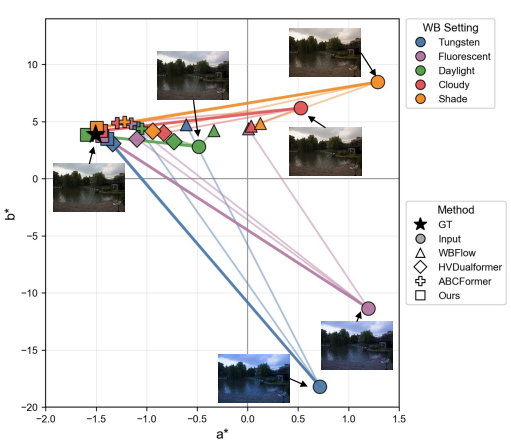


Figure S1: Chromaticity of corrected results in the CIELAB a^*–b^* plane for one scene under five WB settings. Colors indicate different WB settings, and marker shapes denote different methods. Line segments connect results from the same WB setting. Our outputs remain closest to the ground truth across all settings, indicating more consistent chromaticity recovery.



Figure S2: t-SNE visualization of the learned CCS representations. Each sample is displayed using its corresponding input-image thumbnail at the projected location. Despite diverse image content, samples with similar color cast characteristics tend to form coherent neighborhoods, with warm and cool appearances occupying different but gradually varying regions.

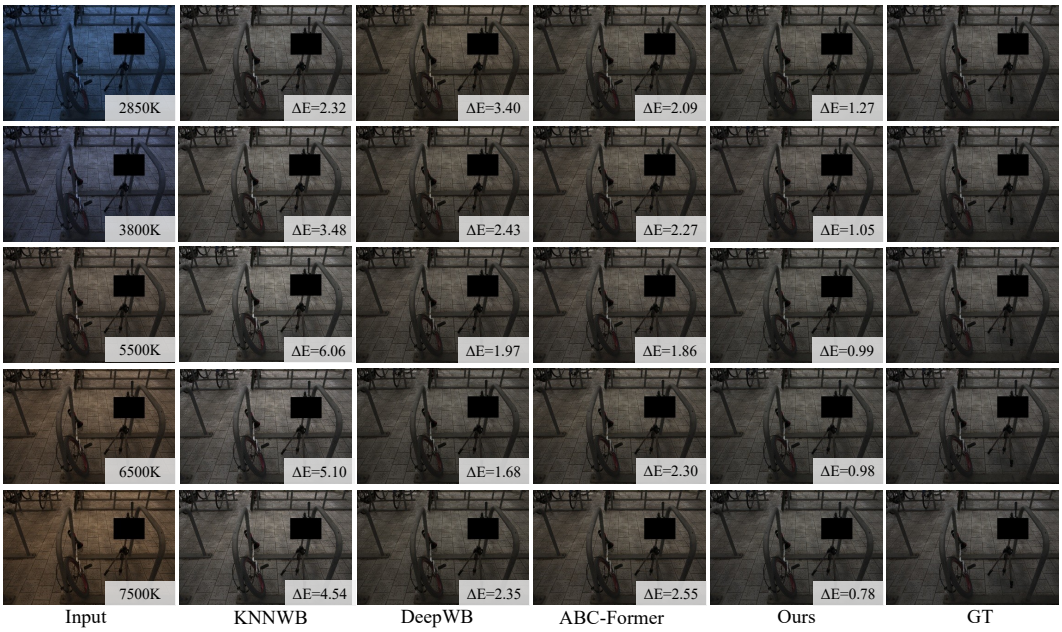


Figure S3: Additional qualitative comparisons on Set 1-Test of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

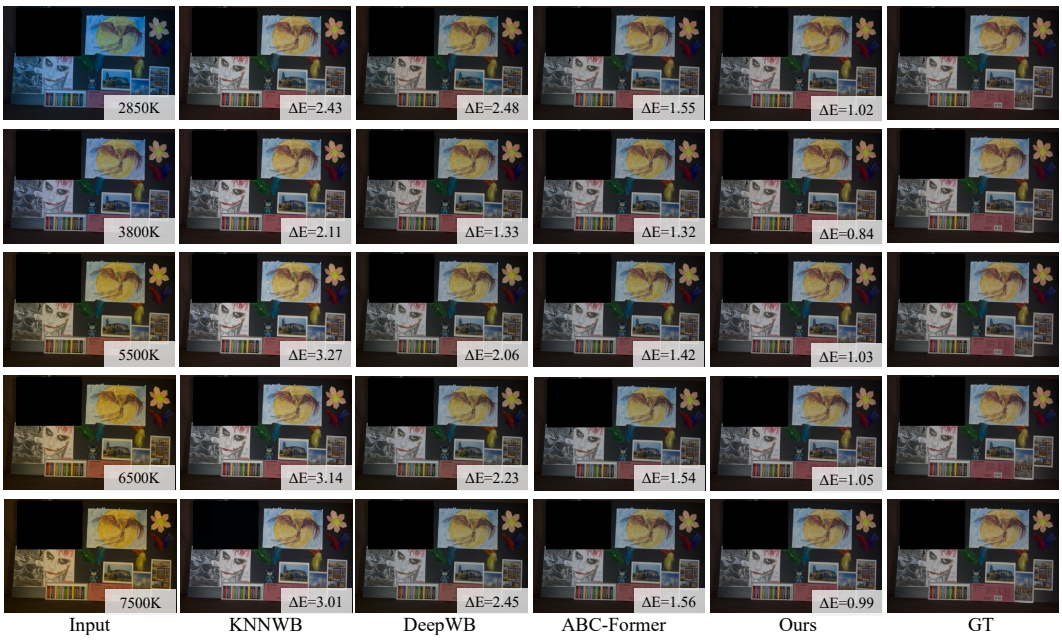


Figure S4: Additional qualitative comparisons on Set 1-Test of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

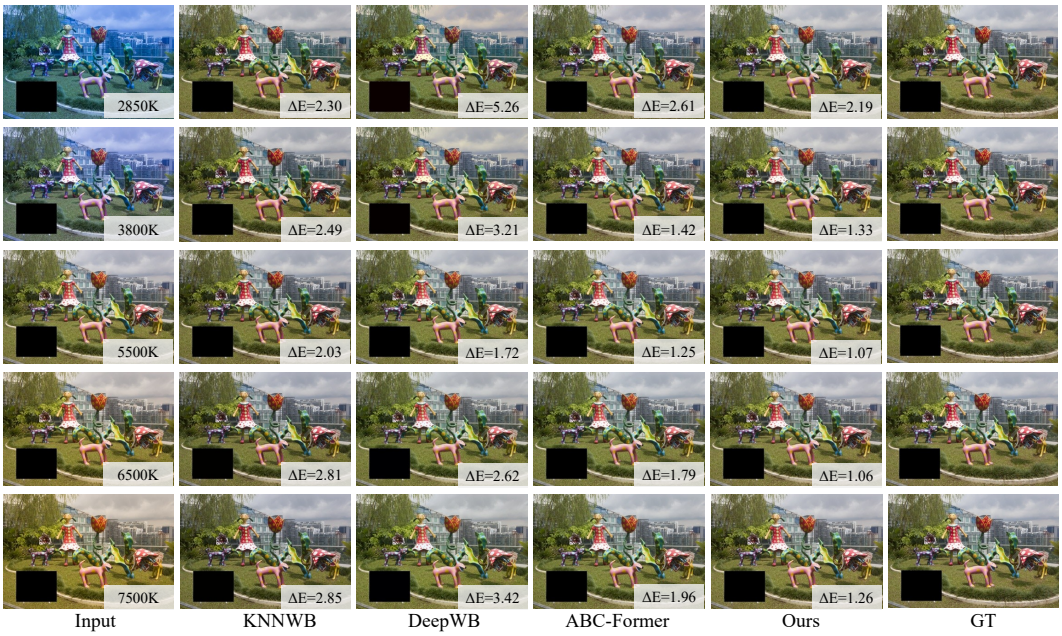


Figure S5: Additional qualitative comparisons on Set 1-Test of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

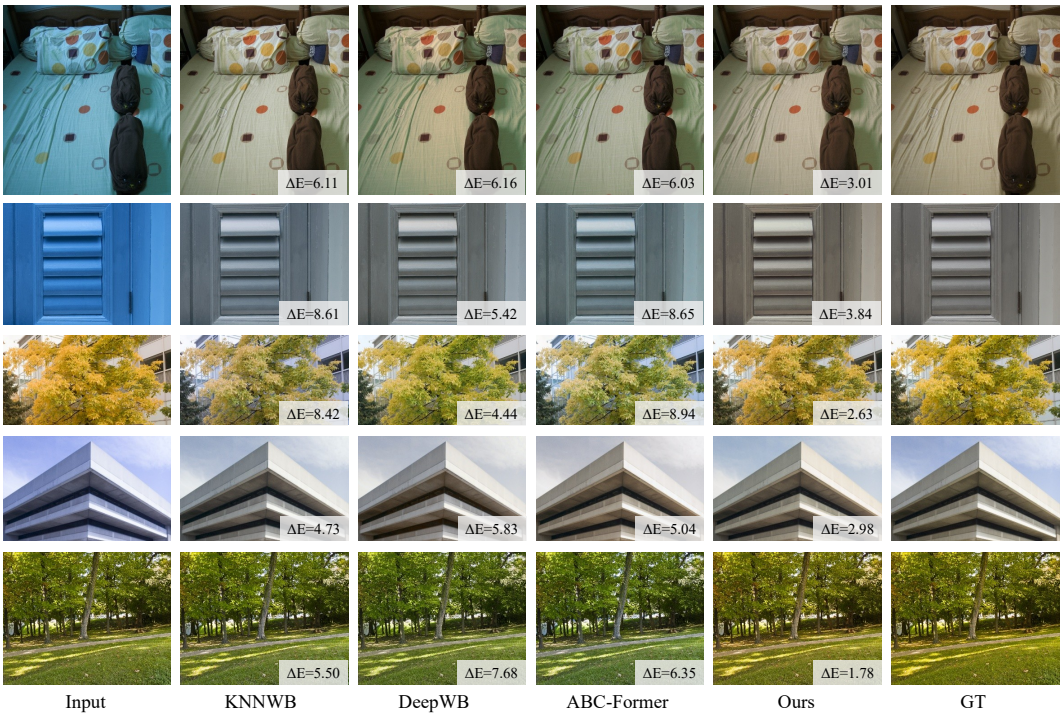


Figure S6: Additional qualitative comparisons on Set 2 of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

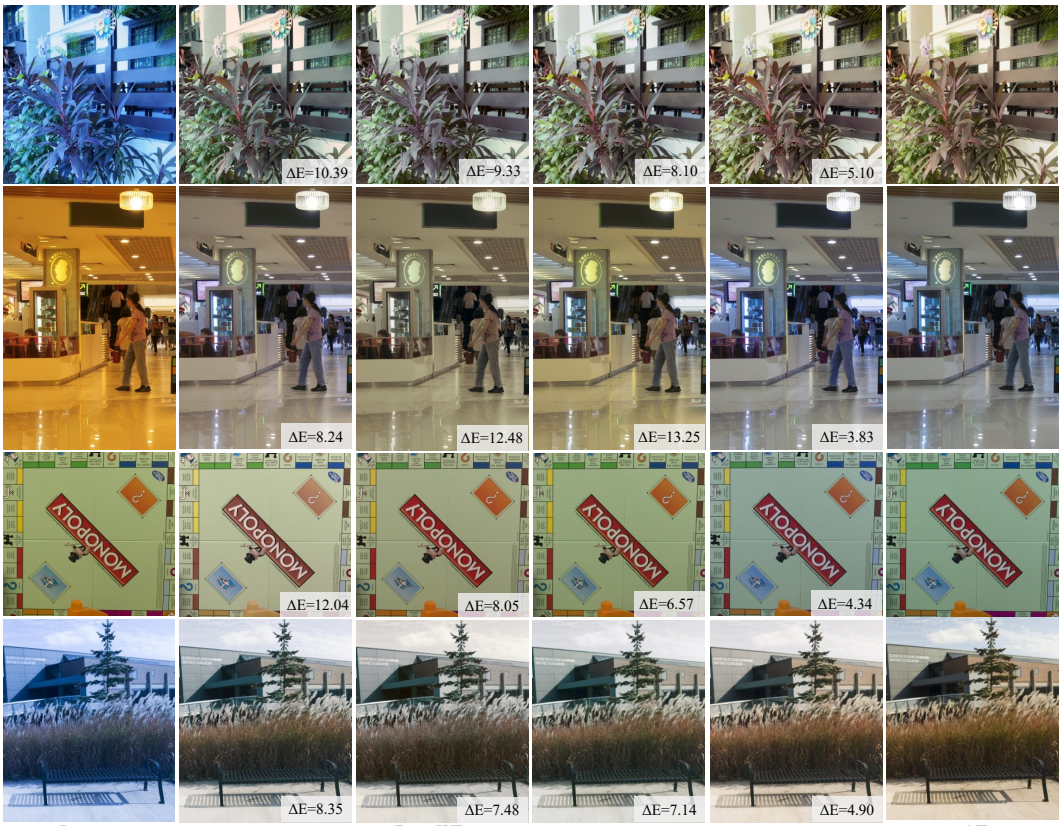


Figure S7: Additional qualitative comparisons on Set 2 of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

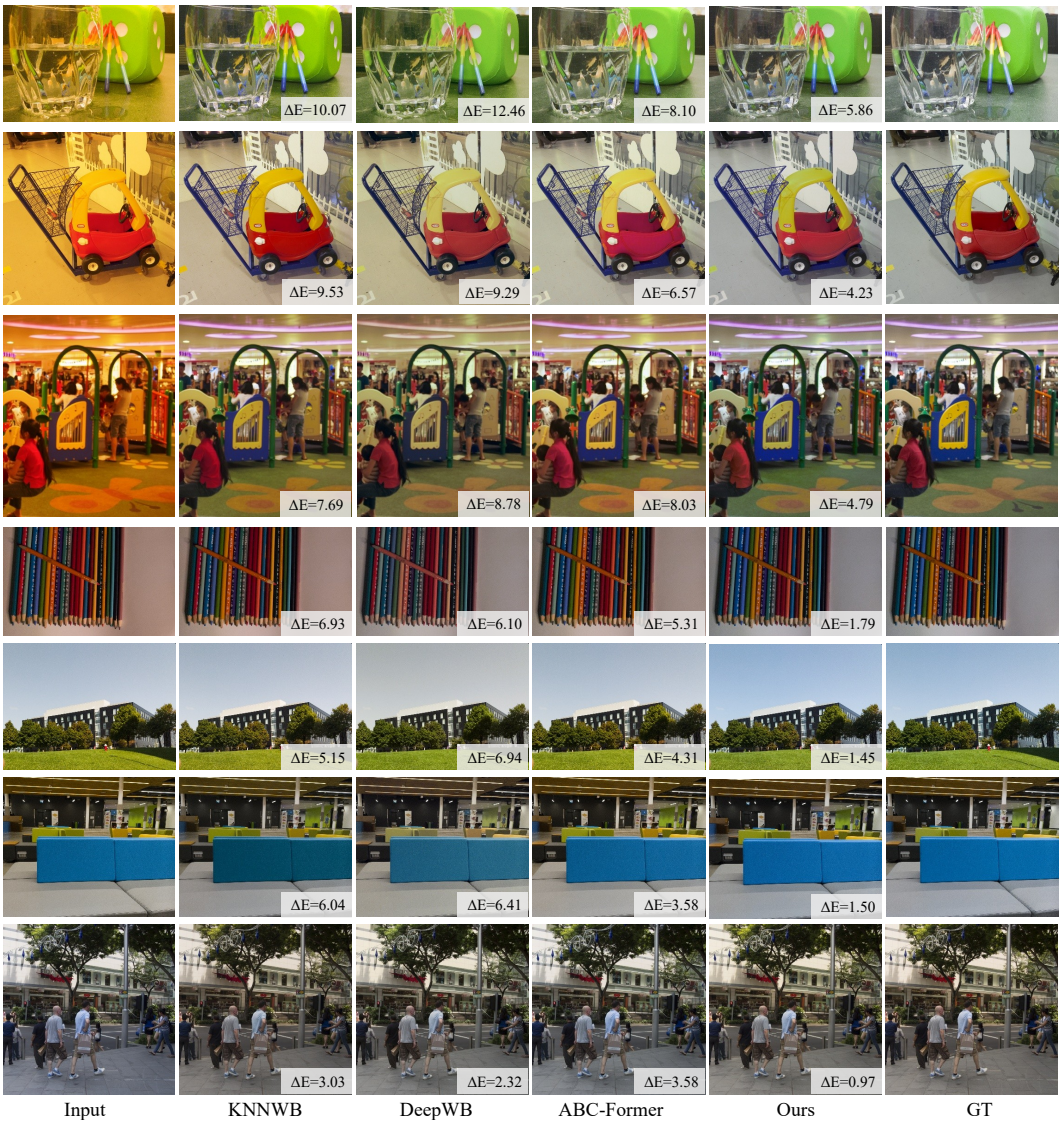


Figure S8: Additional qualitative comparisons on Set 2 of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

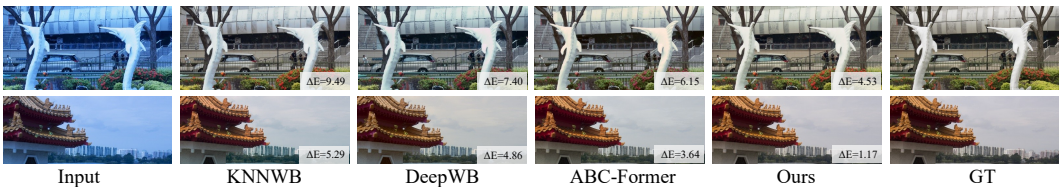


Figure S9: Additional qualitative comparisons on Set 2 of the Rendered WB Dataset (Afifi et al., 2019). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

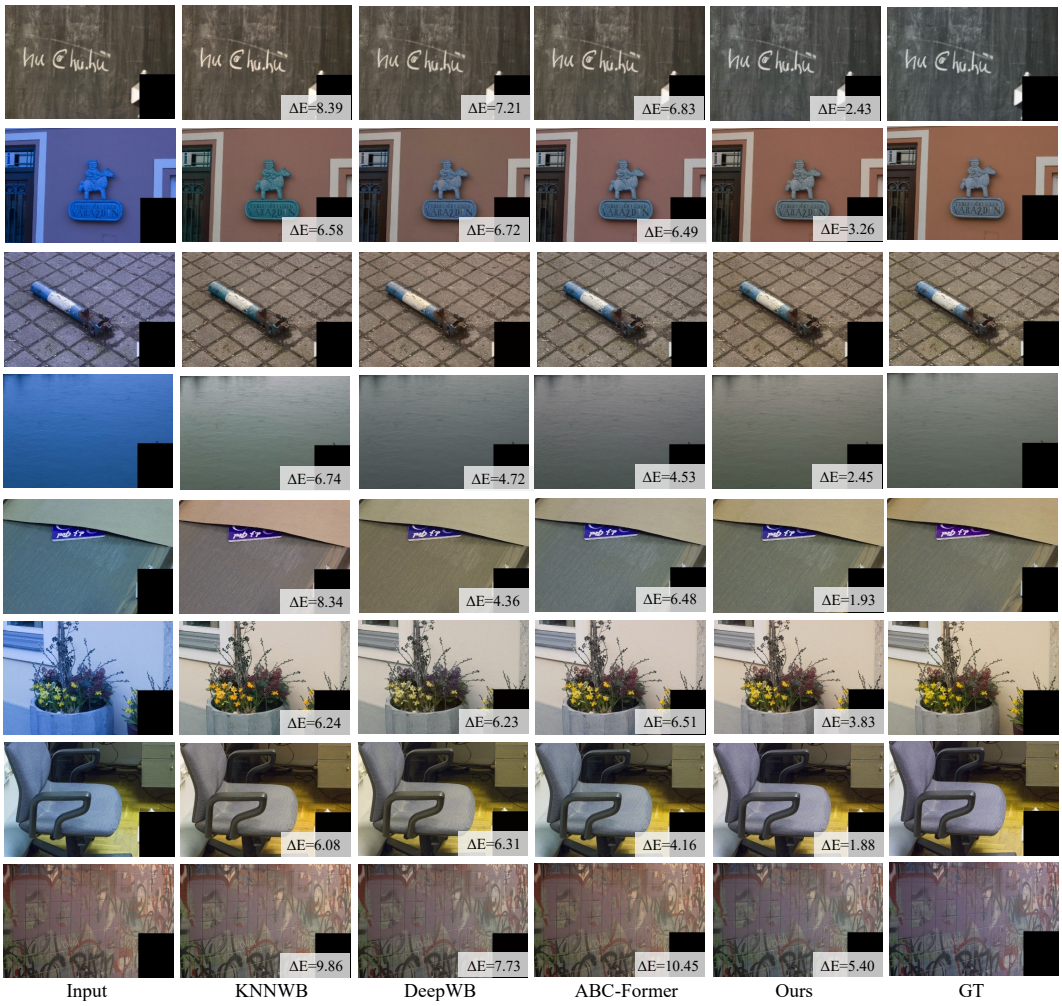


Figure S10: Additional qualitative comparisons on the Rendered Cube+ dataset (Afifi et al., 2019; Banić et al., 2017). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

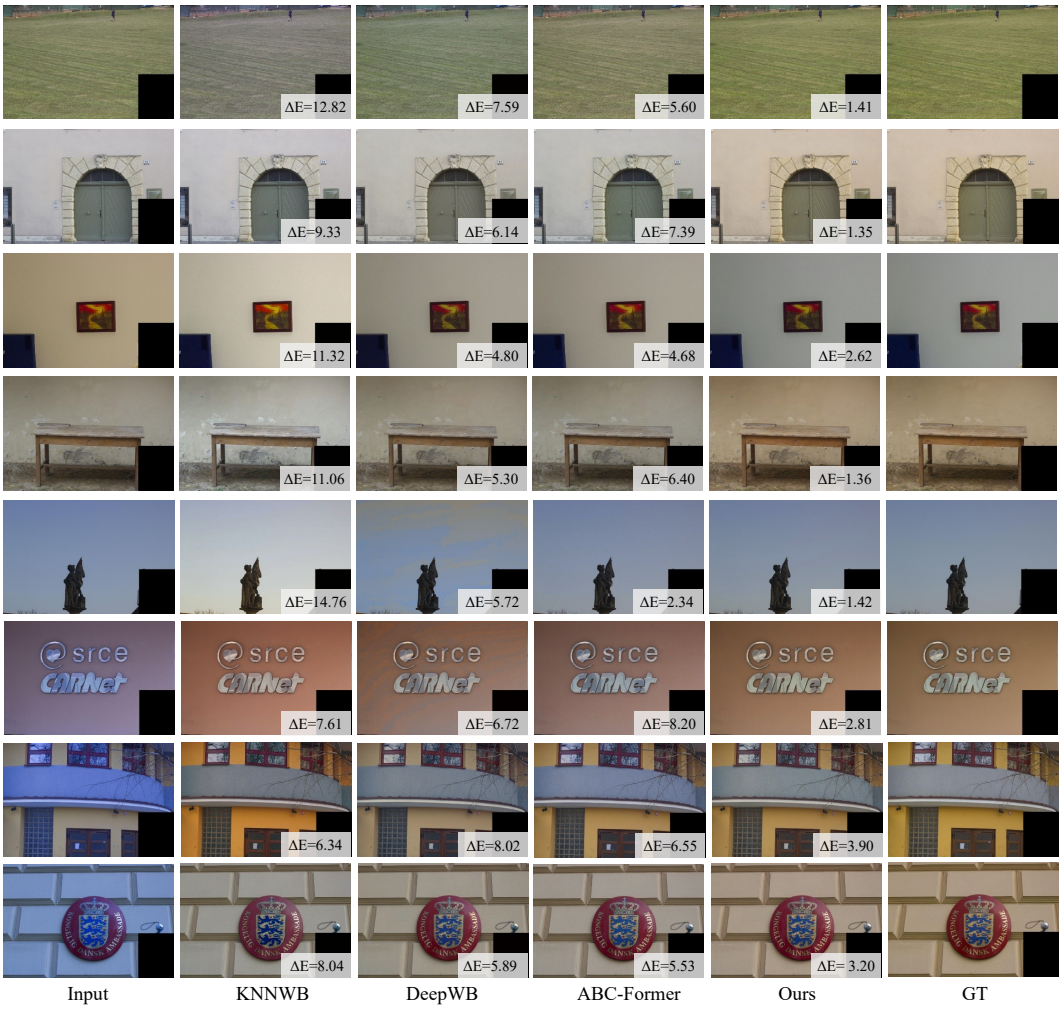


Figure S11: Additional qualitative comparisons on the Rendered Cube+ dataset (Afifi et al., 2019; Banić et al., 2017). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.

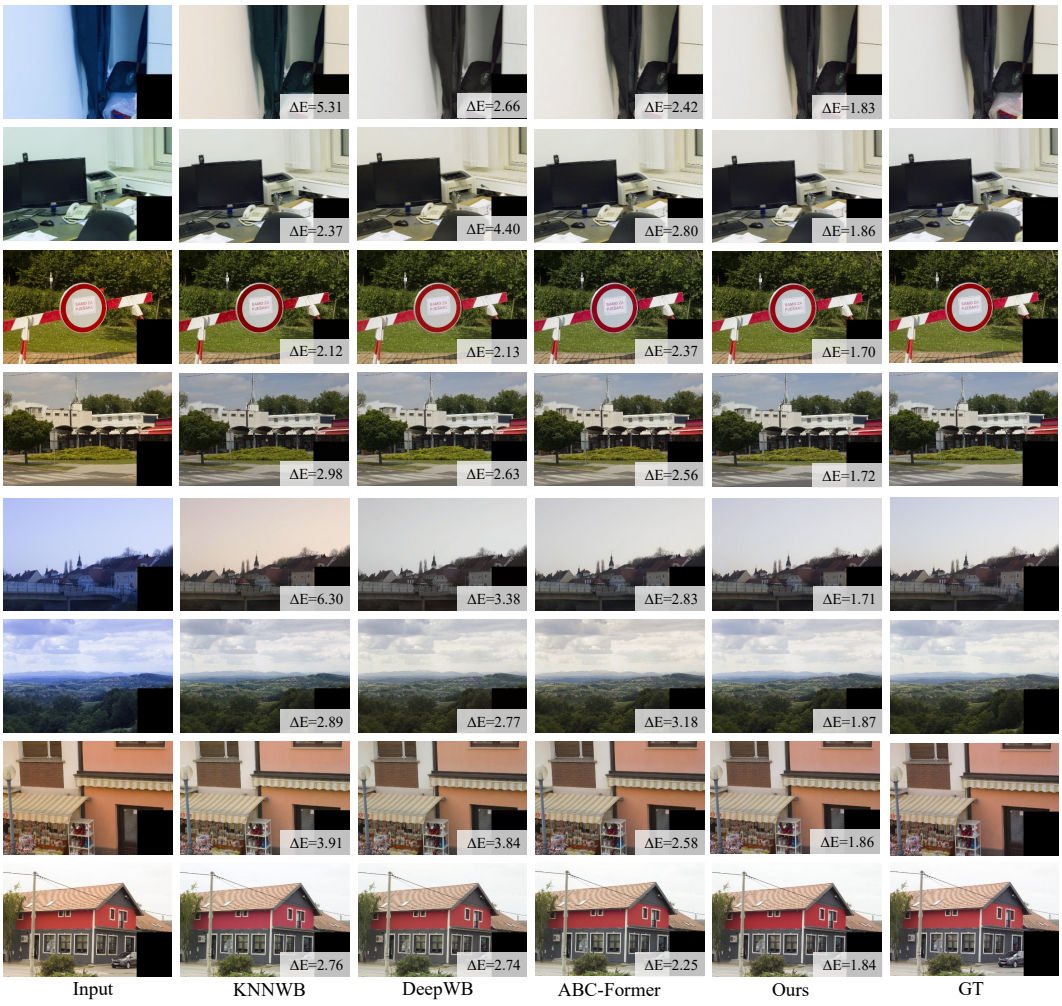


Figure S12: Additional qualitative comparisons on the Rendered Cube+ dataset (Afifi et al., 2019; Banić et al., 2017). The columns show the input, KNNWB (Afifi et al., 2019), DeepWB (Afifi & Brown, 2020), ABC-Former (Chiu et al., 2025), our result, and the ground truth. The ΔE_{2000} error is reported for each competing result.