Supplemental Material for When Color Constancy Goes Wrong: Correcting Improperly White-Balanced Images

Mahmoud Afifi
York University
mafifi@eecs.yorku.ca

Brian Price
Adobe Research
bprice@adobe.com

Scott Cohen
Adobe Research
scohen@adobe.com

Michael S. Brown
York University
mbrown@eecs.yorku.ca

Our supplemental materials are organized as follows: Sec. S.1 provides additional examples to show how diagonal white balance (WB) correction cannot work properly with camera sRGB images; Sec. S.2 defines the error metrics used for evaluation in the main paper; Sec. S.3 provides additional details regarding our dataset; Sec. S.4 provides additional details regarding our selection of k for the k-nearest neighbor searching and the kernel function Φ used for our correction matrix $\mathbf{M}$; Sec. S.5 provides details for a comparison of the principal component analysis (PCA) features used in the main paper with an alternative feature derived from an autoender; Sec. S.6 provides an analysis of number of PCA components used and effect on the accuracy and performance; Sec. S.7 provides additional experiments to study the impact of using different camera models and picture styles in the training set. Sec. S.8 concludes with additional results, which include examples that illustrate the effect of fall-off factor σ, and additional qualitative and quantitative results.

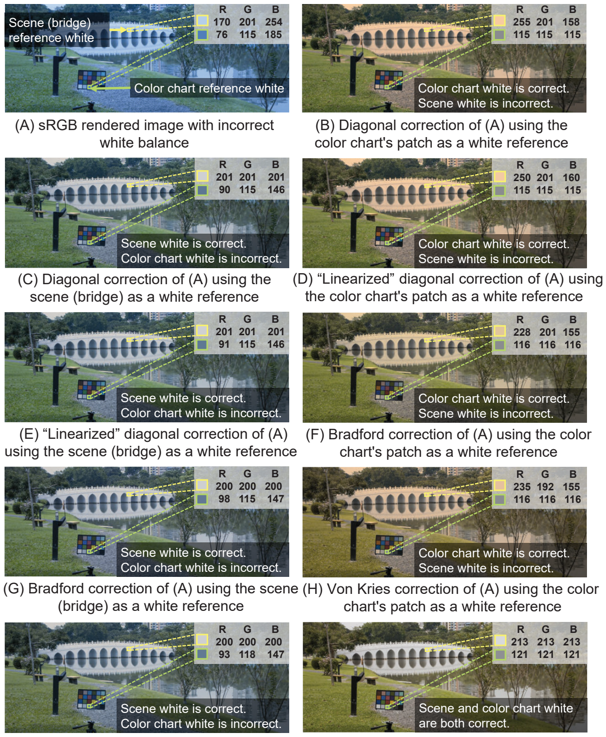
S.1. Diagonal White Balance Correction

As explained in the main paper, WB correction is applied using a 3×3 diagonal matrix to normalize the illumination's colors by mapping them to the achromatic line in the camera's raw-RGB color space.

There is a pervasive misconception that the camera-rendered image can be corrected by simply applying the diagonal WB correction. This erroneously ignores the nonlinear color manipulations that are applied onboard cameras after the essential WB procedure. Mathematically, this incorrect attempt at WB correction in the sRGB color space can be applied as: [12]:

$$\mathbf{I}_{\text{corr}} = \texttt{diag}\left(\hat{\boldsymbol{\ell}}\right)\, \mathbf{I}_{\text{in}}, \tag{1}$$

where $\mathbf{I}_{\text{in}}$ is the sRGB input image represented as an $3 \times N$ matrix, $\texttt{diag}\left(\hat{\boldsymbol{\ell}}\right)$ is a 3×3 diagonal matrix constructed based on the given normalized scene's illuminant vector $\boldsymbol{\ell}_{\text{e}}$ and the target illuminant (in this case, it is considered to be $[\boldsymbol{\ell}_{\texttt{e(G)}}, \boldsymbol{\ell}_{\texttt{e(G)}}, \boldsymbol{\ell}_{\texttt{e(G)}}]^T$ [3,4,15]), and $\mathbf{I}_{\text{corr}}$ is the sRGB "white-balanced" image. Specifically, the diagonal matrix is con-


Scene (bridge) reference white
R G B
170 201 254
76 115 185
Color chart reference white
R G B
255 201 158
115 115 115
Color chart white is correct.
Scene white is incorrect.
R G B
201 201 201
90 115 146
Scene white is correct.
Color chart white is incorrect.
R G B
250 201 160
115 115 115
Color chart white is correct.
Scene white is incorrect.
R G B
201 201 201
91 115 146
Scene white is correct.
Color chart white is incorrect.
R G B
228 201 155
116 116 116
Color chart white is correct.
Scene white is incorrect.
R G B
200 200 200
98 115 147
Scene white is correct.
Color chart white is incorrect.
R G B
235 192 155
116 116 116
Color chart white is correct.
Scene white is incorrect.
R G B
200 200 200
93 118 147
Scene white is correct.
Color chart white is incorrect.
R G B
213 213 213
121 121 121
Scene and color chart white are both correct.

(A) sRGB rendered image with incorrect white balance

(B) Diagonal correction of (A) using the color chart's patch as a white reference

(C) Diagonal correction of (A) using the scene (bridge) as a white reference

(D) "Linearized" diagonal correction of (A) using the color chart's patch as a white reference

(E) "Linearized" diagonal correction of (A) using the scene (bridge) as a white reference

(F) Bradford correction of (A) using the color chart's patch as a white reference

(G) Bradford correction of (A) using the scene (bridge) as a white reference

(H) Von Kries correction of (A) using the color chart's patch as a white reference

(I) Von Kries correction of (A) using the scene (bridge) as a white reference

(J) sRGB rendered image with correct white balance (ground truth image)

Figure 1. (A) A camera sRGB image that has the wrong white balance applied. There are two achromatic regions highlighted in red and yellow. (B) and (C) show diagonal white balance correction applied to the camera image using different reference achromatic points manually selected from the image. (D) and (E) show the results of applying the "linearization" process [1, 8] using the same reference achromatic points. (F) and (G) show the Bradford chromatic adaption transform [16] results using the same reference achromatic points. (H) and (I) show the von Kries chromatic adaption transform [9] results using the same reference achromatic points (see details of the Bradford and von Kries transforms in the supp material). (J) Ground truth camera sRGB image with the correct white balance applied.

structed as follows:

$$\mathtt{diag}\left(\hat{\boldsymbol{\ell}}\right) = \mathtt{diag}\left(\left[\frac{\boldsymbol{\ell}_{\mathtt{e(G)}}}{\boldsymbol{\ell}_{\mathtt{e(R)}}}, \frac{\boldsymbol{\ell}_{\mathtt{e(G)}}}{\boldsymbol{\ell}_{\mathtt{e(G)}}}, \frac{\boldsymbol{\ell}_{\mathtt{e(G)}}}{\boldsymbol{\ell}_{\mathtt{e(B)}}}\right]^T\right). \quad (2)$$

Typically, $\boldsymbol{\ell}_{\mathtt{e}}$ is unknown and there are many methods developed to estimate the scene's illuminant of the linear raw-RGB images (see Sec. 2 in the paper for representative examples). However, it can be defined manually using an achromatic region in the scene, as shown in Fig. 1-(A) (also see Fig. 2-[A] in the main paper). As shown, there are two reference achromatic points in the scene: (i) a gray patch in the color rendition chart $p(1)$ and (ii) a white patch from the bridge in the scene $p(2)$. In Fig. 1-(B), we applied Eq. 1 to correct the input image in Fig. 1-(A) using the color chart's patch as a reference achromatic point. By plugging the values of $diag\left(\hat{\boldsymbol{\ell}}\right)$ into Eq. 1, we can correct the reference achromatic point $p(1)$ as follows:

$$p(1) = \begin{bmatrix} 1.5132 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0.6216 \end{bmatrix} \begin{bmatrix} 76 \\ 115 \\ 185 \end{bmatrix} = \begin{bmatrix} 115 \\ 115 \\ 115 \end{bmatrix}. \quad (3)$$

However, applying the same diagonal correction matrix to the second reference white point $p(2)$ results in incorrect WB (i.e., $p(2) = [255, 201, 158]^T$). Fig. 1-(C) shows another attempt using the bridge scene region as a reference achromatic point. As shown, the same problem appears in the color rendition chart's reference point.

In the main paper, Eq. 1 represents the nonlinear function that generates the final sRGB image (including various operations, such as color enhancement, tone-manipulation) by $f(\cdot)$. The goal of the pre-linearization step [1, 8] is to undo the effect of $f(\cdot)$. However, $f(\cdot)$ cannot be represented by a simple inverse gamma operation, as we have illustrated in the main paper. As shown in Fig. 1-(D) and 1-(E), we encounter the same problem even after applying the pre-linearization step.

Another misconception that is suggested by Matlab is to also consider applying additional chromatic adaption—namely, the Bradford transform [16] and von Kries transform [9].

These models apply WB correction in a post-adaptation cone responses related to biological vision (i.e., tristimulus responses of the long [L], medium [M], short [S] cone cells in the human eye [21]).

The Bradford and von Kries transforms work under the assumption that the proper linearization has been applied, which can be done only with careful radiometric calibration of the function $f(\cdot)$ (see Sec. 2 in the main paper). Since the linearization of the camera image is not correct, the Bradford and von Kries transforms are completely ineffective. As a result, there is no notable improvement in the attempt of correcting improperly white-balanced image colors. Fig. 1-(F–I) shows examples. In some cases, it makes the result worse.

S.2. Evaluation Metrics

As described in the main paper (Sec. 4.2), we have used three different error metrics to evaluate the error per pixel.

The first metric is the mean square error (MSE), which is used to measure the average deviation per color channel between the corrected image $\mathbf{I}_{\text{corr}}$ and the ground truth image $\mathbf{G} \in \mathbf{I}_{\text{gt}}$. The MSE is given by the following equation:

$$e_{\text{MSE}} = \frac{1}{N \times 3} \sum_{i=1}^{N} \sum_{j=1}^{3} \left(\mathbf{I}_{\text{corr}(j,i)} - \mathbf{G}_{(j,i)}\right)^2, \quad (4)$$

where N is the total number of pixels.

The second error metric is the mean angular error (MAE), which calculates the average distance measure of angles between the color vectors of $\mathbf{I}_{\text{corr}}$ and $\mathbf{G}$. This is done to account for the effects of differences in brightness between compared color values. The MAE is calculated using the following equation:

$$e_{\text{MAE}} = \frac{1}{N} \sum_{i=1}^{N} \cos^{-1}\left(\frac{\mathbf{c}_{\mathbf{I}_{\text{corr}}(i)} \cdot \mathbf{c}_{\mathbf{G}_{(i)}}}{\parallel \mathbf{c}_{\mathbf{I}_{\text{corr}}(i)} \parallel \parallel \mathbf{c}_{\mathbf{G}_{(i)}} \parallel}\right), \quad (5)$$

where $\mathbf{c}_{\mathbf{I}_{\text{corr}}(i)}$ and $\mathbf{c}_{\mathbf{G}_{(i)}}$ are the i^{th} sRGB color vectors of the corrected image and the ground truth image, respectively.

Lastly, the $\triangle\mathtt{E}$ (also written Delta E) is used to measure the average changes in visual perception. There are several variants of $\triangle\mathtt{E}$. One of them is the $\triangle\mathtt{E}$ 2000 [20] (which we reported in the main paper). In the supplemental materials, we include also $\triangle\mathtt{E}$ 76 [19], which is based directly on the CIE $\mathtt{L^*a^*b^*}$ space where Euclidian distances better represent perceptual differences.

S.3. Proposed Dataset

As described in the paper, we have generated a dataset of 65,416 sRGB images that were divided into two sets: intrinsic set (Set 1) and extrinsic set (Set 2). Table 1 shows more details of the camera makes and models used for each set. The size of the original dataset is ∼1.14 TB, which was down-sampled by bicubic interpolation to 48.7 GB. For Set 1, the average image width and height are 890.1 and 687.2 pixels, respectively. For Set 2, the average width and height are 1,219.5 and 1,129.9 pixels, respectively.

For both sets, we have used the following WB presets: Fluorescent, Shade, Incandescent, Cloudy, and Daylight. For Set 1, we have used the following camera picture styles: Adobe Standard, Faithful, Landscape, Neutral, Portrait, Standard, Vivid, soft, D2X (mode 1, 2, and 3), and

Table 1. Camera models used in the proposed dataset. The intrinsic set (Set 1) comprises 62,535 sRGB images (48.7 GB) for seven different cameras. The extrinsic set (Set 2) comprises 2,881 sRGB images (5.43 GB) for one DSLR camera and four different mobile phone cameras. For each image in the dataset, there is a corresponding ground truth sRGB image rendered with a correct white balance in Adobe Standard.

	Intrinsic set (Set 1)							
Camera	Canon EOS-1Ds Mark III	Canon EOS 600D	Fujifilm X-M1	Nikon D40	Nikon D5200	Canon 1D	Canon 5D	Total
# of images	10,721	9,040	5,884	10,826	8,953	2,284	14,827	62,535
Size	11.00 GB	8.27 GB	4.78 GB	3.4 GB	10.3 GB	1.27 GB	9.68 GB	48.7 GB
	Extrinsic set (Set 2)							
Camera	Olympus E-PL6		Mobile phone cameras: Galaxy S6 Edge, iPhone 7, LG G4, and Google Pixel					Total
# of images	1,874		1,007					2,881
Size	3.5 GB		1.93 GB					5.43 GB

ACR (4.4 and 3.7). For Set 2, we have used the following camera picture styles: (for the Olympus camera) Muted, Portrait, Vivid, Adobe Standard, and (for the mobile cameras) the camera's "embedded style".

S.4. Selecting the Value of k and the Color Correction Matrix

In order to select the number of nearest neighbors k and the most appropriate color correction transform, we have evaluated the accuracy of different color correction approaches between an incorrectly white-balanced camera image and its correctly white-balanced target image.

This study was conducted for quality assessment rather than performance. Accordingly, we have used the RGB-uv histogram features with bandwidth $m = 180$ without dimensionality reduction applied.

Taking the square root after normalizing $\mathbf{H}$ in Eq. 5 in the main paper makes the classic Euclidean L_2 distance applicable as a symmetric similarity metrics to measure the similarity between two distributions [2]. Consistently with the PCA feature similarity measurement used in the main paper, the Hellinger distance [18] was used as a similarity metric in this study. The Hellinger distance [18] between two histograms $h\left(\mathbf{I}_1\right)$ and $h\left(\mathbf{I}_2\right)$ can be represented as $\left(1/\sqrt{2}\right)\ L_2\left(h\left(\mathbf{I}_1\right), h\left(\mathbf{I}_2\right)\right)$.

We tested eight different color correction matrices on Set 1 using three-fold validation. The matrices are: 3×3 full color correction matrices, 3×9, 3×11, 3×19, 3×34 polynomial color correction (PCC) matrices, and 3×6, 3×13, 3×22 root polynomial color correction (RPC) matrices [10, 14]. For each color correction matrix, we tested different values of k. Note that the 3×3 color correction matrix is computed using Eq. 2 in the main paper using $\Phi\left(\mathbf{I}_\text{t}^{(i)}\right) = \mathbf{I}_\text{t}^{(i)}$—that is, the kernel function here is an identity function.

Table 2 shows the kernel functions used to generate each color correction matrix. In Fig. 2, we report the obtained results using the four error metrics described in Sec. S.2.

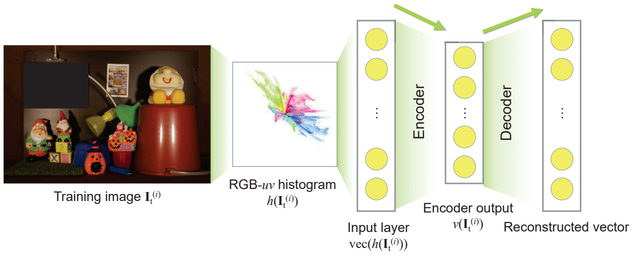

Training image $\mathbf{I}_\text{t}^{(i)}$
RGB-uv histogram $h(\mathbf{I}_\text{t}^{(i)})$
Input layer $\text{vec}(h(\mathbf{I}_\text{t}^{(i)}))$
Encoder
Encoder output $v(\mathbf{I}_\text{t}^{(i)})$
Decoder
Reconstructed vector

Figure 3. Autoencoder feature is generated after training an autoencoder with a single encoder/decoder layer.

As shown, the 3×11 color correction matrix, described by Hong et al. [14], provided the best results for our task. Also, it is shown that the accuracy increases as the value of k increases. At a certain point, however, increasing k negatively affects the accuracy.

In this set of experiments, adopting the RGB-uv histogram features requires approximately 14.9 GB of memory having ∼41K training samples represented as single-precision floating-point values, and runs in 44.6 seconds to correct a 12 mega-pixel image on average. This process includes the RGB-uv histogram feature extraction, the brute-force search of the k nearest neighbors, blending the correction matrix, and the final image correction. In the main paper, we extract a compact feature representing each RGB-uv histogram. This compact representation improves the performance (requiring less than 1.5 seconds on a CPU to correct a 12 mega-pixel image) and achieves on-par accuracy compared to employing the original RGB-uv histogram features.

S.5. Autoencoder Features

In order to extract a compact representation of our RGB-uv histogram feature, denoted as $h(\mathbf{I})$, our proposed method relies on PCA. We also explored the use of an autoencoder to map the vectorized histogram $\text{vec}(h\left(\mathbf{I}\right)) \in \mathbb{R}^{m\times m\times 3}$ to a c-dimensional space, where $c \ll m\times m\times 3$. Our goal was to

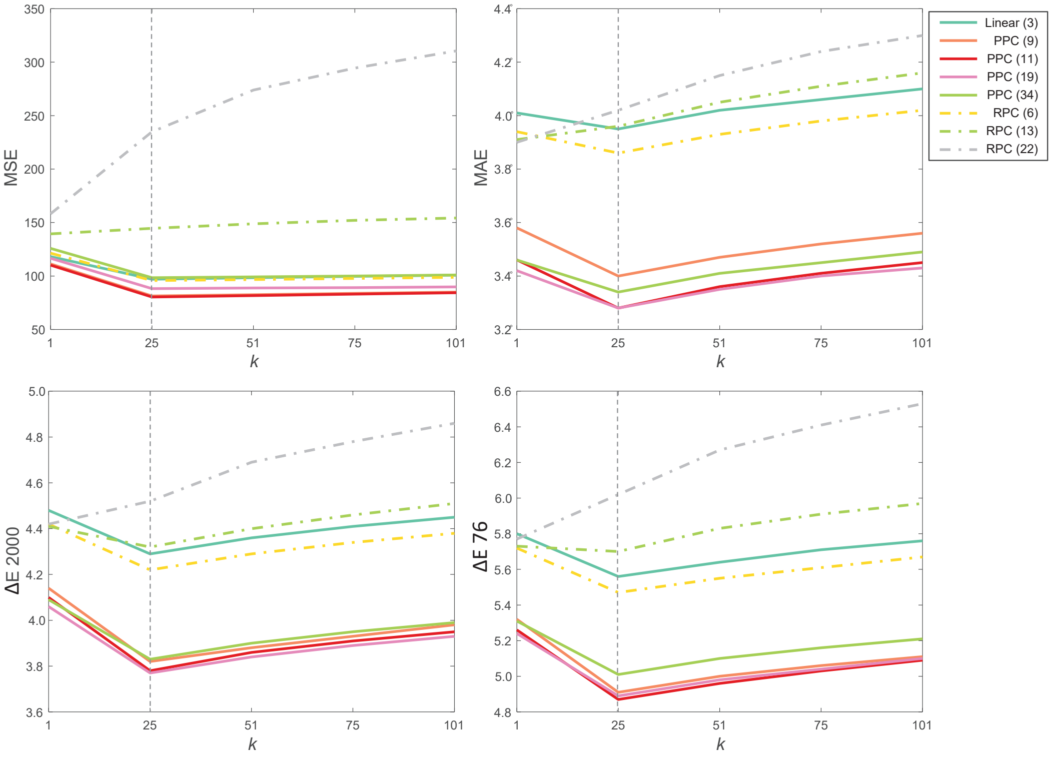


Figure 2. A study of the accuracy obtained using different color correction matrices, which are: (i) linear 3×3 full matrix, (ii) 3×9, (iii) 3×11, (iv) 3×19, (v) 3×34 polynomial color correction (PCC) matrices [10, 14], (vi) 3×6, (vii) 3×13, and (viii) 3×22 root-polynomial color correction (RPC) matrices [10]. The horizontal axis represents the number of nearest neighbors k and the vertical axis represents the error between the corrected images and the ground truth images using different error metrics. The error metrics are: the average of the mean squared error (MSE), the average of the mean angular error (MAE), the average of the $\triangle$E 2000 [20], and the average of the $\triangle$E 76 [19].

Table 2. Different kernel functions used to study the most suitable color correction matrix for our problem. The first column represents the dimensions of the output vector of the corresponding kernel function in the second column.

Dimensions	**Kernel function output**
3 (linear)	$[\mathrm{R, G, B}]^T$ (identity)
9 (PCC) [10]	$[\mathrm{R, G, B, R^2, G^2, B^2, RG, RB, GB}]^T$
11 (PCC) [14]	$[\mathrm{R, G, B, RG, RB, GB, R^2, G^2, B^2, RGB,1}]^T$
19 (PCC) [10]	$[\mathrm{R, G, B, RG, RB, GB, R^2, G^2, B^2, R^3, G^3, B^3, RG^2, RB^2, GB^2, GR^2, BG^2, BR^2, RGB}]^T$
34 (PCC) [10]	$[\mathrm{R, G, B, RG, RB, GB, R^2, G^2, B^2, R^3, G^3, B^3, RG^2, RB^2, GB^2, GR^2, BG^2, BR^2, RGB, R^4, G^4, B^4, R^3G, R^3B, G^3R, G^3B, B^3R, B^3G, R^2G^2, G^2B^2, R^2B^2, R^2GB, G^2RB, B^2RB}]^T$
6 (RPC) [10]	$[\mathrm{R, G, B}, \sqrt{\mathrm{RG}}, \sqrt{\mathrm{GB}}, \sqrt{\mathrm{RB}}]^T$
13 (RPC) [10]	$[\mathrm{R, G, B}, \sqrt{\mathrm{RG}}, \sqrt{\mathrm{GB}}, \sqrt{\mathrm{RB}}, \sqrt[3]{\mathrm{RG^2}}, \sqrt[3]{\mathrm{RB^2}}, \sqrt[3]{\mathrm{GB^2}}, \sqrt[3]{\mathrm{GR^2}}, \sqrt[3]{\mathrm{BG^2}}, \sqrt{\mathrm{BR^2}}, \sqrt[3]{\mathrm{RGB}}]^T$
22 (RPC) [10]	$[\mathrm{R, G, B}, \sqrt{\mathrm{RG}}, \sqrt{\mathrm{GB}}, \sqrt{\mathrm{RB}}, \sqrt[3]{\mathrm{RG^2}}, \sqrt[3]{\mathrm{RB^2}}, \sqrt[3]{\mathrm{GB^2}}, \sqrt[3]{\mathrm{GR^2}}, \sqrt[3]{\mathrm{BG^2}}, \sqrt{\mathrm{BR^2}}, \sqrt[3]{\mathrm{RGB}}, \sqrt[4]{\mathrm{R^3G}}, \sqrt[4]{\mathrm{R^3B}}, \sqrt[4]{\mathrm{G^3R}}, \sqrt[4]{\mathrm{G^3B}}, \sqrt[4]{\mathrm{B^3R}}, \sqrt[4]{\mathrm{B^3G}}, \sqrt[4]{\mathrm{R^2GB}}, \sqrt[4]{\mathrm{G^2RB}}, \sqrt[4]{\mathrm{B^2RG}}]^T$

achieve a similar accuracy of using the original histogram feature with a much smaller amount of memory (i.e., < 25 MB for the entire model including the coefficients and bias parameters). Accordingly, our experiments were performed using only linear transformations with a single coefficient matrix and a bias vector.

We examined training an autoencoder with a single fully connected encoder layer followed by a sigmoid activation function, and a single fully connected decoder layer followed by a sigmoid activation function; see Fig. 3. We have also examined a decoder linear transfer function, and found that the sigmoid decoder activation function works better for our data. The encoder layer contains c neurons, while the decoder layer contains $m \times m \times 3$ neurons. The autoencoder was trained for 1,500 epochs using scaled conjugate gradient descent [17] with weight decay of 0.001 for

Table 3. Results were obtained using the extrinsic set (**Set 2**) by employing different camera models with two setups. In the first setup, we used only a single picture style (Standard). In the second setup, we combined all picture styles (which is equivalent to the reported results in the main paper). The terms MSE and MAE stand for mean square error and mean angular error, respectively. The top results are indicated with yellow and bold.

Camera model	**Picture style**	**MSE**				**MAE**				**△E 2000**			
		Mean	**Q1**	**Q2**	**Q3**	**Mean**	**Q1**	**Q2**	**Q3**	**Mean**	**Q1**	**Q2**	**Q3**
Canon EOS-1Ds Mark III	**Standard**	258.94	59.42	138.74	292.59	6.26°	3.18°	5.09°	8.04°	7.16	4.33	6.39	9.07
	All	194.94	53.39	113.64	233.93	5.53°	2.95°	4.59°	7.16°	6.43	4.08	5.74	8.02
Canon EOS 600D	**Standard**	233.46	72.62	125.57	250.37	5.60°	2.87°	4.52°	7.21°	6.72	4.12	5.79	8.43
	All	157.63	45.77	84.97	171.68	4.59°	2.47°	3.85°	5.76°	5.69	3.64	4.98	7.02
Fujifilm X-M1	**Standard**	363.16	76.28	180.55	389.89	6.97°	3.09°	5.40°	9.56°	7.51	4.64	6.58	9.62
	All	246.57	51.51	121.58	264.58	5.40°	2.60°	4.29°	7.11°	6.49	4.09	5.66	8.10
Nikon D40	**Standard**	625.20	251.98	372.86	705.06	6.63°	3.27°	5.14°	8.44°	9.37	6.94	8.62	11.08
	All	243.79	80.02	164.22	299.12	4.82°	2.49°	3.97°	6.34°	6.82	4.68	6.27	8.41
Nikon D5200	**Standard**	293.34	68.68	157.39	348.11	5.96°	2.93°	4.93°	7.95°	6.91	4.21	6.31	8.97
	All	192.46	41.82	103.01	221.65	4.69°	2.35°	3.85°	6.20°	5.89	3.53	5.33	7.52
Canon 5D	**Standard**	242.04	52.88	132.68	296.95	6.05°	2.94°	4.81°	8.20°	7.11	4.23	6.37	9.26
	All	259.20	55.60	139.38	316.24	6.01°	3.01°	4.91°	7.93°	7.21	4.35	6.50	9.45
Canon 1D	**Standard**	600.42	221.19	396.37	715.97	8.62°	4.61°	7.47°	11.58°	10.01	6.74	9.16	12.40
	All	281.92	91.73	163.07	325.77	6.25°	3.48°	5.31°	8.00°	7.71	5.18	6.91	9.52
All	**Standard**	206.04	48.25	103.28	220.69	5.25°	2.61°	4.36°	6.92°	6.18	3.80	5.43	7.76
	All	**171.09**	**37.04**	**87.04**	**190.88**	**4.48°**	**2.26°**	**3.64°**	**5.95°**	**5.60**	**3.43**	**4.90**	**7.06**

Table 4. Results obtained using the extrinsic set (**Set 2**) by employing different camera models with two setups. In the first setup, we used only a single picture style (Standard). In the second setup, we combine all picture styles (which is equivalent to the reported results in the main paper). This table shows △**E** 76 results; for other error metrics, please see Table 3. The top results are indicated with yellow and bold.

Camera model	**Picture style**	**△E 76**			
		Mean	**Q1**	**Q2**	**Q3**
Canon EOS-1Ds Mark III	**Standard**	9.52	5.41	7.96	11.67
	All	8.57	5.14	7.10	10.40
Canon EOS 600D	**Standard**	8.88	5.18	7.24	10.67
	All	7.52	4.46	6.22	9.12
Fujifilm X-M1	**Standard**	10.44	5.8	8.44	12.95
	All	8.93	4.97	7.17	11.30
Nikon D40	**Standard**	12.29	8.47	10.72	14.35
	All	8.89	5.71	7.74	10.90
Nikon D5200	**Standard**	9.39	5.30	7.87	12.10
	All	7.88	4.43	6.99	9.95
Canon 5D	**Standard**	9.76	5.22	8.02	12.68
	All	9.84	5.32	8.10	12.69
Canon 1D	**Standard**	13.25	8.41	11.68	16.05
	All	10.35	6.32	8.76	12.42
All	**Standard**	8.22	4.70	6.81	10.16
	All	**7.46**	**4.19**	**6.04**	**9.38**

the L_2 weight regularization.

Fig. 4 shows the obtained results of the trained autoencoder compared to utilizing the PCA features. As shown, both autoencoder and PCA features outperform the diagonal WB correction and achieve on-par accuracy compared to employing the original RGB-uv histogram features. In the end, PCA proved to be the better choice. Exploring this in more detail is an interesting area for future work.

S.6. Performance Analysis

In the main paper, we reported the results using PCA features extracted from RGB-uv histograms with bandwidth $m = 60$ and represented by $c = 55$ dimensions. In this section, we provide more analysis on the relationship between the performance (i.e., the runtime and required memory size) and the accuracy. Fig. 5 shows the average MSE using three-fold validation on Set 1 with different values of m and c. As shown, the dimensionality of the feature vectors (in this example, we use the PCA features) is inversely proportional to the error, whereas it is directly proportional to the required memory size and runtime. Note that all the reported runtimes in Fig. 5 are after excluding the time required to generate the histogram, which takes ~0.54 seconds, and computed on an Intel Xeon E5-1607 @ 3.10 GHz machine.

S.7. The Impact of Using Different Camera Models and Picture Styles

Our final color transformation $\mathbf{M}$ is represented as a weighted linear combination of different correction transformations associated with training images taken from different cameras and render styles (Eq. 7 in the main paper). We can interpret $\mathbf{M}$ correction as mapping the input image to a meta-camera's output composed from the most similar images to the input.

In order to show the effect of having rendered sRGB images taken from different camera models with different picture styles, we performed the following study using Set 2 as a testing set and all samples of Set 1 as training data. We recomputed our results using a single picture style (in this experiment, we used the Standard style). Additionally, we recomputed our results using each training camera individually. Table 3 and Table 4 show the MSE, MAE, and △**E** 2000 (Table 3), and △**E** 76 (Table 4) values obtained by using a single picture style against all picture styles. As

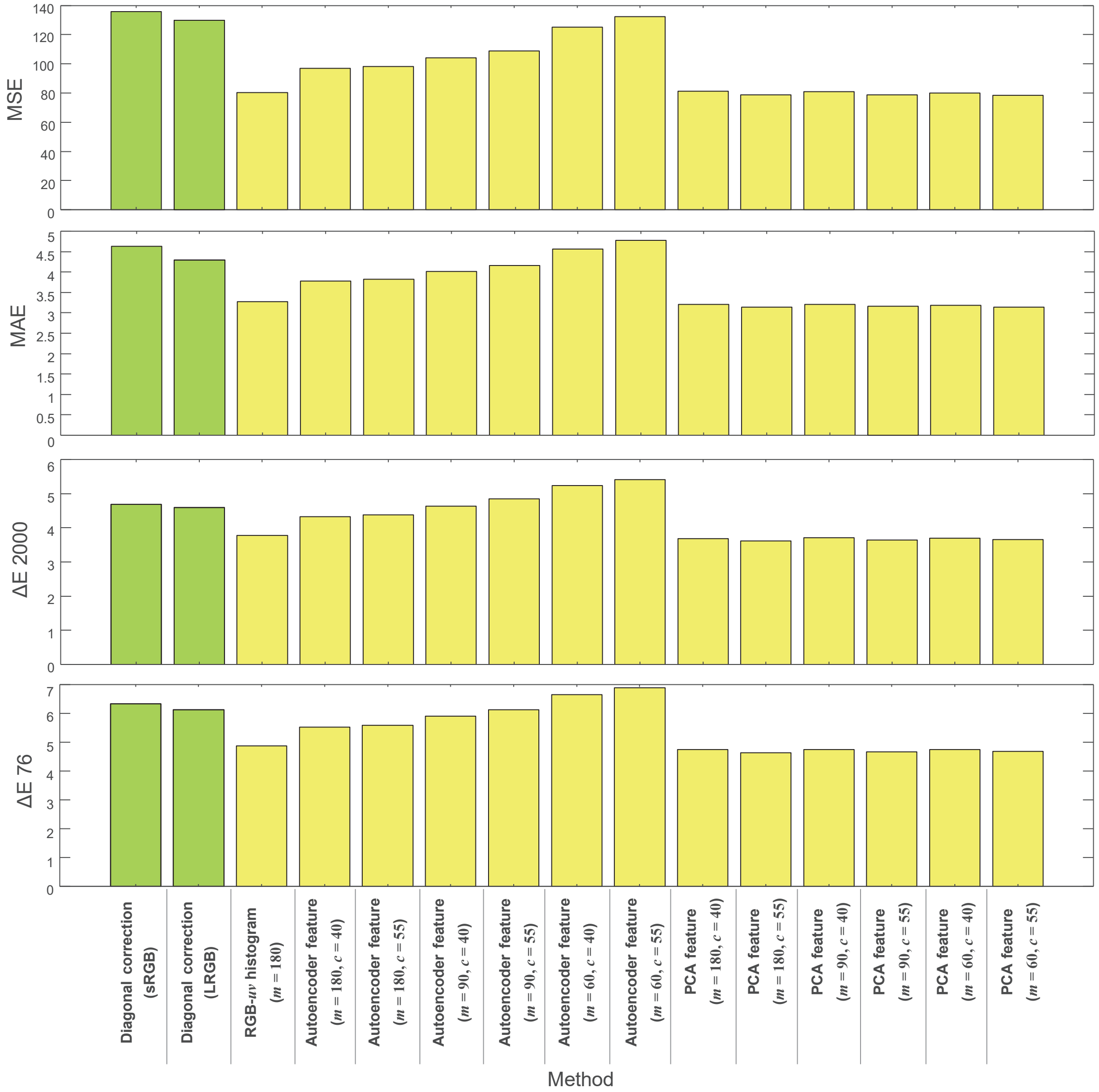


Figure 4. Quantitative results of the diagonal white balance correction using the exact achromatic reference point (sRGB and "linearized" sRGB, denoted as LRGB) and our method using: RGB-uv histogram, autoencoder, and PCA features. All results obtained using three-fold validation on **Set 1**. We report the average error using four different metrics, which are: mean squared error (MSE), mean angular error (MAE), $\triangle$E 2000, $\triangle$E 76. For our method, we report the results using different histogram bandwidth (m) and dimensions of the feature vector (c).

illustrated in Table 3 and Table 4, having different camera models and camera styles improves the results by combining different color correction matrices of several rendering functions.

S.8. Additional Results

In this section, we provide additional quantitative and qualitative results.

Additional Quantitative Results In the main paper, we compared our method against the diagonal WB correction using the "exact achromatic" reference point, denoted as (ER), which refers to a manual illumination determination based on the color checker in each testing image. **Note that ER represents a best possible achromatic reference** as it is an actual achromatic reference.

We also performed comparisons against the diagonal correction using representative examples of illuminant estimation methods—the diagonal matrix is automatically computed and not derived from a selected achromatic patch in the scene. As mentioned in Sec. 2 in the main paper, illumination estimation methods are intended to be ap-

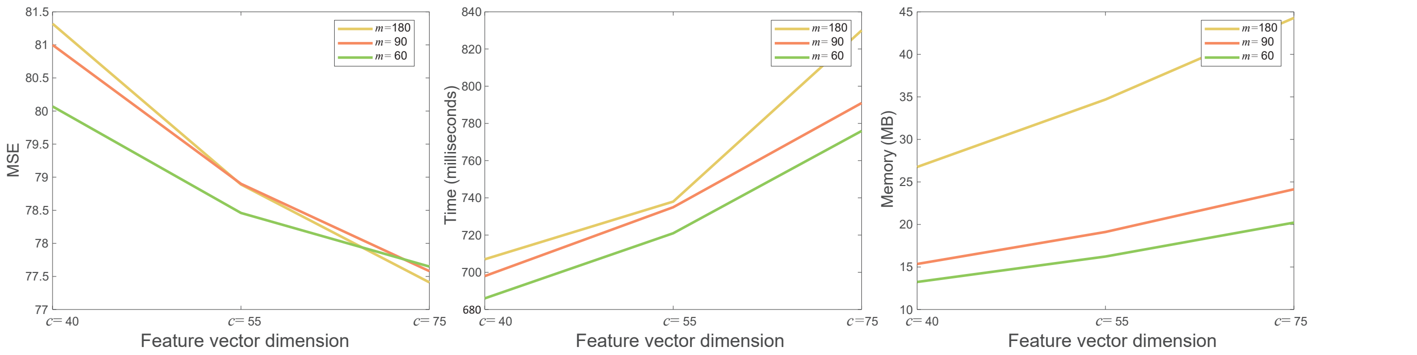
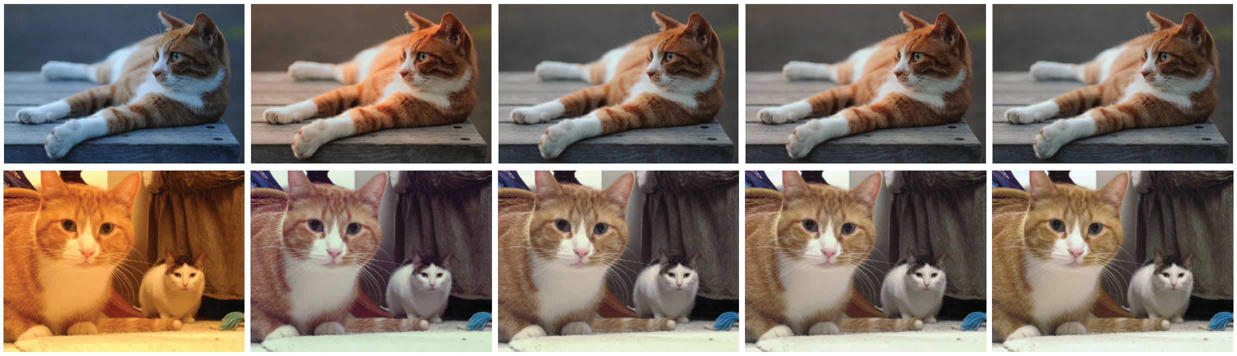


Figure 5. This figure shows the relationship between performance (runtime and required memory) and accuracy, represented by the average mean squared error (MSE) using different dimensions of the feature vector. As shown, there is an inverse relationship between performance and accuracy. The results are reported using PCA features and three-fold validation on **Set 1** with single-precision floating-point representation. The training samples were ∼41K PCA features.



(A) Input image

(B) Corrected image
k=25, σ = 0.1
(~ equivalent to k=1)

(C) Corrected image
k= 25, σ = 0.25

(D) Corrected image
k= 101, σ = 0.25

(E) Corrected image
k= 101, σ = 1.00

Figure 6. The effect of k and fall-off factor σ on the corrected images. (A) Input image. (B)-(E) corrected images using different values of k and σ. Note that when $\sigma = 0.01$, it gives approximately the same effect when $k = 1$. Photo credits: Hisashi and Franfeeley Flickr (CC BY-SA 2.0).

plied on raw-RGB images; however, in this case we apply them on the sRGB image. We use five well-known statistical methods for illumination estimation—namely, the gray world (GW) [6], gray edges (GE) [22], weighted gray edges (wGE) [13], max-RGB [5], and shades of gray (SoG) [11]. The Minkowski norm (p) was set to 5 for GE, wGE, and SoG. For GE, we computed the results using the first and second differentiations. For each method, we calculated the diagonal correction with and without the pre-linearization process. We have excluded learning-based illuminant estimation methods since they were mostly trained on the linear RGB space and re-training them in the sRGB space would use the ERs as ground truth illuminants, which were already included in our comparisons in the main paper and supplemental materials. Table 5 shows the results obtained by illuminant estimation algorithms. Table 6 reports complementary results for both Table 1 in the main paper and Table 5 in this supplemental material using $\triangle\mathtt{E}$ 76.

Additional Qualitative Results Fig. 6 illustrates the effect of the fall-off factor σ (Eq. 8 in the main paper). Also it shows the effect of different k values. Fig. 7 illustrates our results using different WB settings and picture styles. As shown, our results are consistent against different settings and have lower error values (reported using $\triangle$E 2000 error metric). In this example, the original raw-RGB image was taken from the MIT-Adobe FiveK dataset [7]. Note that the raw-RGB image was captured by a Nikon D700, which is not included in our training set camera models. Fig. 8 shows another example taken from the MIT-Adobe FiveK dataset—again, the shown images here were captured by different cameras than our dataset cameras. This figure shows that our method produces similar images to the input images if they were initially rendered with the correct WB setting. Lastly, additional qualitative results are provided in Fig. 9.

Table 5. Comparisons between our method with diagonal white balance correction using an *exact achromatic* reference point (ER) and a set of statistical-based illuminant estimation methods, which are: gray world (GW) [6], gray edges (GE) [22] using the first and second differentiations, weighted gray edges (wGE) [13], max-RGB [5], and shades of gray (SoG) [11]. The diagonal correction is computed from and applied directly to the sRGB images, denoted as (sRGB), and "linearized" RGB [1, 8], denoted as (LRGB). The terms Q1, Q2, and Q3 denote the first, second (median), and third quartile, respectively. The terms MSE and MAE stand for mean square error and mean angular error, respectively. The top results are indicated with yellow and bold.

Method	**MSE**				**MAE**				**△E 2000**			
	Mean	**Q1**	**Q2**	**Q3**	**Mean**	**Q1**	**Q2**	**Q3**	**Mean**	**Q1**	**Q2**	**Q3**
Intrinsic set (Set 1): DSLR multiple cameras (62,535 images)												
GW (sRGB) [6]	282.76	70.50	180.81	380.89	7.23°	4.14°	6.40°	9.51°	7.99	5.08	7.47	10.40
GW (LRGB) [6]	285.51	73.37	184.72	384.67	7.97°	4.36°	6.91°	10.53°	8.48	5.38	7.91	11.03
GE-1 (sRGB) [22]	193.99	43.38	119.93	267.84	6.81°	3.22°	5.49°	9.31°	6.86	3.92	6.30	9.17
GE-1 (LRGB) [22]	190.59	42.51	116.75	261.6	6.54°	3.22°	5.37°	8.78°	6.82	3.91	6.20	9.06
GE-2 (sRGB) [22]	208.20	44.82	126.83	285.31	7.03°	3.30°	5.67°	9.62°	7.06	4.01	6.44	9.44
GE-2 (LRGB) [22]	204.93	43.86	123.19	279.15	6.76°	3.31°	5.54°	9.12°	7.02	4.01	6.35	9.32
wGE (sRGB) [13]	225.87	42.76	122.39	300.48	7.15°	3.20°	5.60°	9.69°	7.07	3.78	6.29	9.50
wGE (LRGB) [13]	223.30	42.40	119.47	294.22	6.90°	3.17°	5.42°	9.26°	7.05	3.76	6.21	9.43
max-RGB (sRGB) [5]	285.07	47.20	160.6	397.79	8.48°	3.49°	7.03°	12.35°	8.05	3.97	7.25	11.34
max-RGB (LRGB) [5]	280.86	46.21	156.09	390.99	8.17°	3.41°	6.76°	11.83°	7.96	3.92	7.12	11.21
SoG (sRGB) [11]	171.30	38.31	104.85	235.57	6.06°	2.99°	5.08°	8.21°	6.19	3.52	5.72	8.25
SoG (LRGB) [11]	169.33	38.10	102.21	231.50	5.96°	3.03°	5.03°	7.91°	6.24	3.56	5.71	8.27
ER (sRGB)	135.77	20.20	71.74	196.15	4.63°	1.99°	3.56°	6.14°	4.69	2.25	4.00	6.68
ER (LRGB)	130.01	19.73	68.54	183.65	4.29°	1.85°	3.35°	5.70°	4.59	2.24	3.89	6.51
Ours	**77.79**	**13.74**	**39.62**	**94.01**	**3.06°**	**1.74°**	**2.54°**	**3.76°**	**3.58**	**2.07**	**3.09**	**4.55**
Extrinsic set (Set 2): DSLR and mobile phone cameras (2,881 images)												
GW (sRGB) [6]	500.18	173.69	332.75	615.40	8.89°	5.82°	8.32°	11.33°	10.74	7.92	10.29	13.12
GW (LRGB) [6]	469.86	163.07	312.28	574.85	8.61°	5.44°	7.94°	10.93°	10.68	7.70	10.13	13.15
GE-1 (sRGB) [22]	791.10	235.37	524.94	1052.38	12.90°	7.98°	12.41°	17.50°	13.09	9.17	12.98	16.68
GE-1 (LRGB) [22]	779.27	225.36	510.03	1038.71	12.55°	7.55°	11.87°	17.00°	12.98	9.06	12.86	16.57
GE-2 (sRGB) [22]	841.83	239.58	542.07	1114.86	13.16°	7.94°	12.55°	17.76°	13.31	9.20	13.20	17.09
GE-2 (LRGB) [22]	831.01	231.42	530.52	1099.75	12.84°	7.64°	12.13°	17.45°	13.22	9.00	13.13	17.01
wGE (sRGB) [13]	999.95	236.46	587.55	1350.41	13.80°	8.08°	12.99°	18.80°	14.05	9.13	13.80	18.56
wGE (LRGB) [13]	990.20	230.38	577.62	1345.52	13.52°	7.76°	12.62°	18.56°	14.00	9.00	13.70	18.56
max-RGB (sRGB) [5]	791.99	263.00	572.23	1087.14	13.47°	8.44°	12.93°	18.50°	13.01	9.12	13.44	17.15
max-RGB (LRGB) [5]	780.63	256.40	560.58	1073.22	13.18°	8.15°	12.57°	18.12°	12.93	9.02	13.36	17.08
SoG (sRGB) [11]	429.35	147.05	286.84	535.72	9.54°	5.72°	8.85°	12.65°	10.01	7.09	9.85	12.69
SoG (LRGB) [11]	393.85	137.21	267.37	497.40	8.96°	5.31°	8.26°	11.97°	9.81	6.87	9.67	12.46
ER (sRGB)	422.31	110.70	257.76	526.16	7.99°	4.36°	7.11°	10.57°	8.53	5.52	8.38	11.11
ER (LRGB)	385.23	99.05	230.86	475.72	7.22°	3.80°	6.34°	9.54°	8.15	5.07	7.88	10.68
Ours	**171.09**	**37.04**	**87.04**	**190.88**	**4.48°**	**2.26°**	**3.64°**	**5.95°**	**5.60**	**3.43**	**4.90**	**7.06**

References

[1] Matthew Anderson, Ricardo Motta, Srinivasan Chandrasekar, and Michael Stokes. Proposal for a standard default color space for the Internet - sRGB. In *Color and Imaging Conference*, pages 238–245, 1996.

[2] Relja Arandjelovic and Andrew Zisserman. Three things everyone should know to improve object retrieval. In *CVPR*, 2012.

[3] Çağlar Aytekin, Jarno Nikkanen, and Moncef Gabbouj. A data set for camera-independent color constancy. *IEEE Transactions on Image Processing*, 27(2):530–544, 2018.

[4] Jonathan T Barron and Yun-Ta Tsai. Fast fourier color constancy. In *CVPR*, 2017.

[5] David H Brainard and Brian A Wandell. Analysis of the retinex theory of color vision. *Journal of the Optical Society of America A*, 3(10):1651–1661, 1986.

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

[7] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *CVPR*, 2011.

[8] Marc Ebner. *Color Constancy*, volume 6. John Wiley & Sons, 2007.

[9] Mark D Fairchild. *Color Appearance Models*. John Wiley & Sons, 2013.

[10] Graham D Finlayson, Michal Mackiewicz, and Anya Hurlbert. Color correction using root-polynomial regression. *IEEE Transactions on Image Processing*, 24(5):1460–1470, 2015.

[11] Graham D Finlayson and Elisabetta Trezzi. Shades of gray and colour constancy. In *Color and Imaging Conference*, 2004.

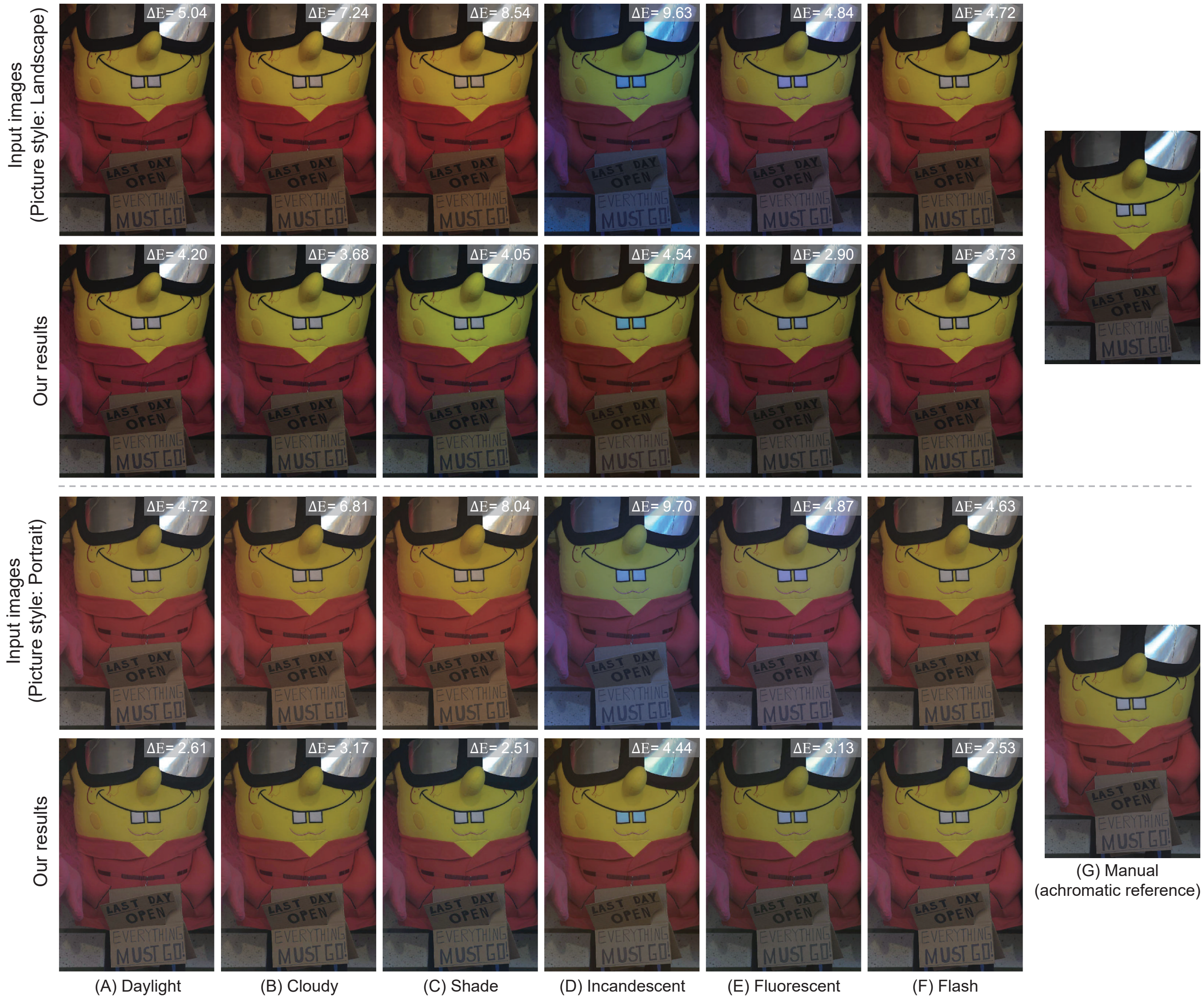


Figure 7. This figure shows our correction of an input image with different white balance settings and picture styles. The first and third rows show the camera-rendered images using two different picture styles—namely, Landscape and Portrait. The second and fourth rows show our results. Each column represents a different white balance setting that has been applied to the raw-RGB image before rendering the final sRGB image. The white balance settings are: (A) Daylight, (B) Cloudy, (C) Shade, (D) Incandescent, (E) Fluorescent, (F) Flash, and (G) manual correction (ground truth). We reported the $\triangle \mathbf{E}$ 2000 error between each image and the corresponding ground truth image. The original raw-RGB image is taken from the MIT-Adobe FiveK dataset [7] and captured by a Nikon D700.

[12] Arjan Gijsenij, Theo Gevers, and Joost Van De Weijer. Computational color constancy: Survey and experiments. *IEEE Transactions on Image Processing*, 20(9):2475–2489, 2011.

[13] Arjan Gijsenij, Theo Gevers, and Joost Van De Weijer. Improving color constancy by photometric edge weighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(5):918–929, 2012.

[14] Guowei Hong, M Ronnier Luo, and Peter A Rhodes. A study of digital camera colorimetric characterisation based on polynomial modelling. *Color Research & Application*, 26(1):76–84, 2001.

[15] Edmund Y Lam. Combining gray world and retinex theory for automatic white balance in digital photography. In *International Symposium on Consumer Electronics*, 2005.

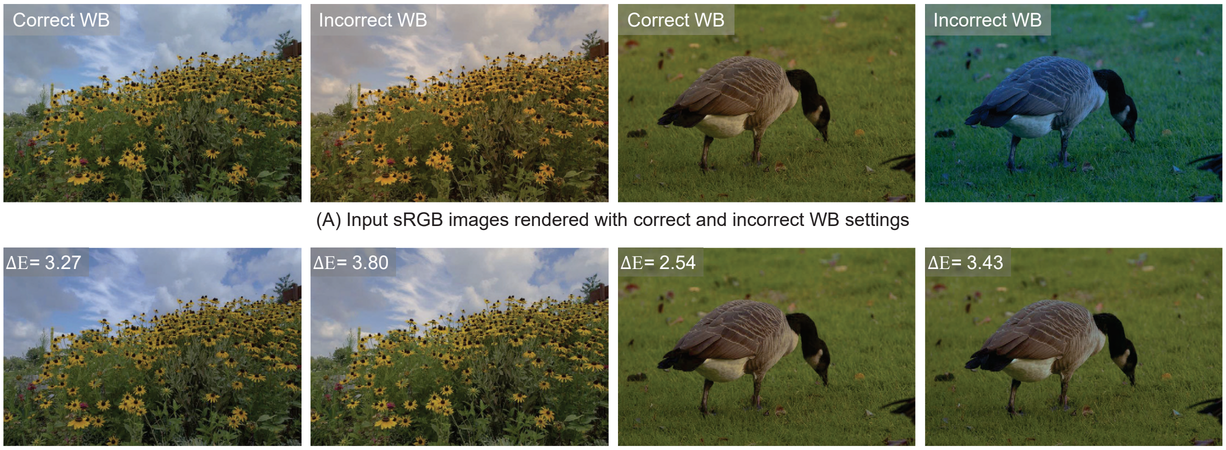
[16] King Man Lam. Metamerism and Colour Constancy. *Ph. D. Thesis, University of Bradford*, 1985.

[17] Martin Fodslette Møller. A scaled conjugate gradient algorithm for fast supervised learning. *Neural networks*, 6(4):525–533, 1993.

[18] David Pollard. *A User's Guide to Measure Theoretic Probability*, volume 8. Cambridge University Press, 2002.

[19] Gaurav Sharma and Raja Bala. *Digital Color Imaging Handbook*. CRC press, 2017.

[20] Gaurav Sharma, Wencheng Wu, and Edul N Dalal. The CIEDE2000 color-difference formula: Implementation



(A) Input sRGB images rendered with correct and incorrect WB settings

(B) Our results

Figure 8. Our method can deal with correctly and incorrectly white-balanced images. (A) sRGB rendered images with correct/incorrect white balance settings. (B) Our results. The original raw-RGB images are taken from the MIT-Adobe FiveK dataset [7].

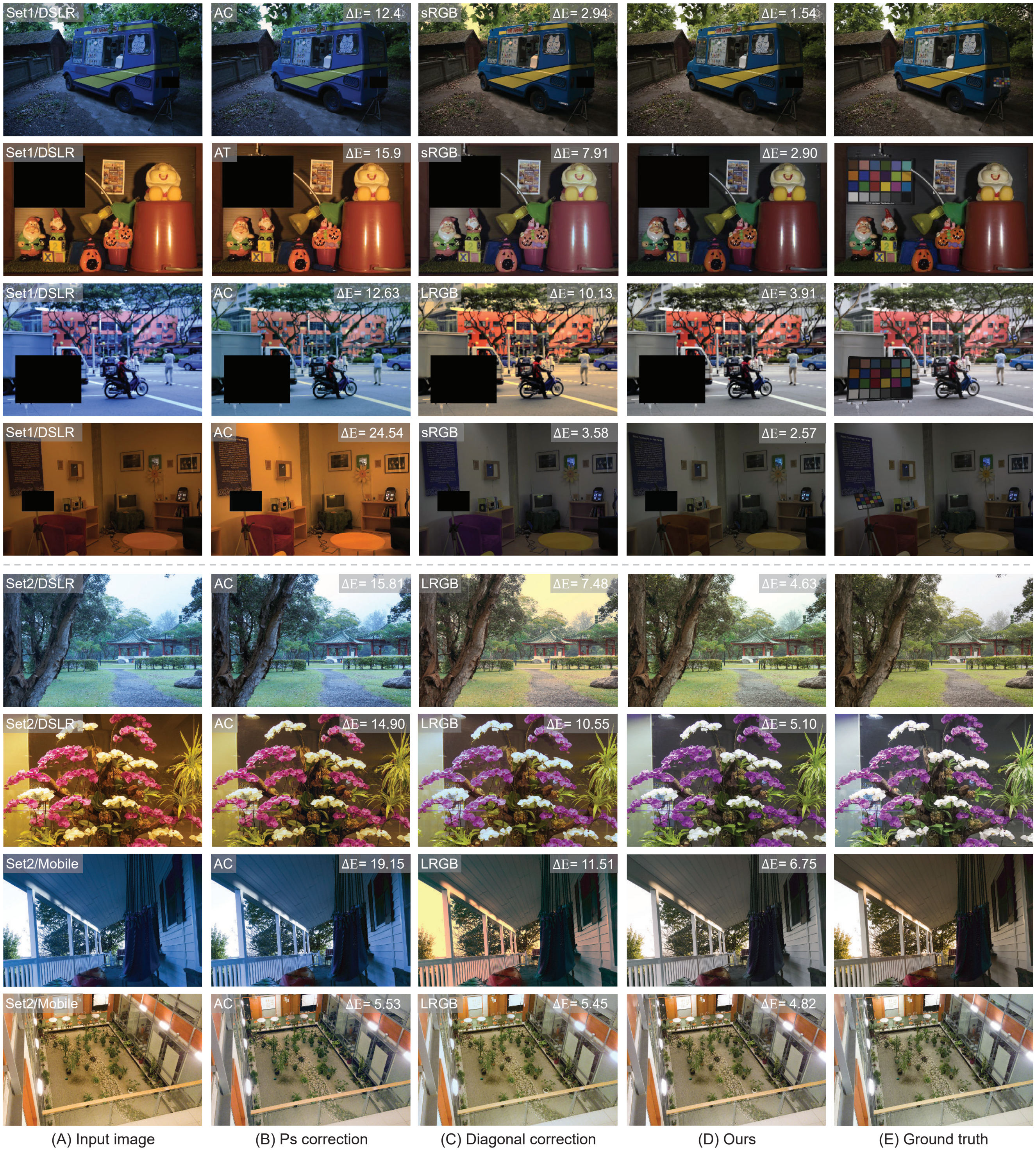
Table 6. △**E** 76 results: complementary results for Table 1 in the main paper and Table 5 in this supplemental material. The top results are indicated with yellow and bold.

Method	**△E 76**			
	Mean	**Q1**	**Q2**	**Q3**
Intrinsic set (Set 1): DSLR multiple cameras (62,535 images)				
Photoshop auto-color	14.13	7.35	11.73	18.43
Photoshop auto-tone	15.42	8.83	14.24	20.53
GW (sRGB) [6]	8.48	5.38	7.91	11.03
GW (LRGB) [6]	11.12	6.38	9.96	14.57
GE-1 (sRGB) [22]	9.09	4.75	8.01	12.13
GE-1 (LRGB) [22]	8.98	4.69	7.85	11.89
GE-2 (sRGB) [22]	9.47	4.86	8.26	12.63
GE-2 (LRGB) [22]	9.35	4.81	8.08	12.43
wGE (sRGB) [13]	9.68	4.71	8.18	12.90
wGE (LRGB) [13]	9.60	4.66	8.08	12.71
max-RGB (sRGB) [5]	11.52	5.03	9.73	16.34
max-RGB (LRGB) [5]	11.35	4.98	9.51	16.04
SoG (sRGB) [11]	8.10	4.26	7.28	10.83
SoG (LRGB) [11]	8.11	4.30	7.19	10.78
ER (sRGB)	6.34	2.82	5.34	9.07
ER (LRGB)	6.13	2.80	5.18	8.75
Ours	**4.56**	**2.51**	**3.92**	**5.87**
Extrinsic set (Set 2): DSLR and mobile phone cameras (2,881 images)				
Photoshop auto-color	17.86	9.65	16.11	23.63
Photoshop auto-tone	20.53	13.84	19.42	26.42
GW (sRGB) [6]	15.13	10.07	13.68	18.91
GW (LRGB) [6]	14.89	9.77	13.40	18.86
GE-1 (sRGB) [22]	19.99	12.36	18.77	26.38
GE-1 (LRGB) [22]	19.76	12.02	18.55	26.17
GE-2 (sRGB) [22]	20.54	12.75	19.06	26.98
GE-2 (LRGB) [22]	20.33	12.44	18.68	26.85
wGE (sRGB) [13]	22.48	12.48	20.37	30.31
wGE (LRGB) [13]	22.32	12.28	20.13	30.16
max-RGB (sRGB) [5]	20.52	13.31	19.96	26.99
max-RGB (LRGB) [5]	20.34	13.15	19.77	26.77
SoG (sRGB) [11]	14.15	9.26	13.21	18.10
SoG (LRGB) [11]	13.74	8.88	12.84	17.61
ER (sRGB)	12.83	7.45	11.80	16.54
ER (LRGB)	12.12	6.91	10.97	15.63
Ours	**7.46**	**4.19**	**6.04**	**9.38**

notes, supplementary test data, and mathematical observations. *Color Research & Application*, 30(1):21–30, 2005.

[21] Andrew Stockman. Physiologically-based color matching functions. In *Color and Imaging Conference*, 2008.

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



(A) Input image (B) Ps correction (C) Diagonal correction (D) Ours (E) Ground truth

Figure 9. Additional qualitative comparisons between the proposed approach and other techniques on **Set 1** (first four rows) and **Set 2** (last four rows). (A) Input image in sRGB. (B) Results of Adobe Photoshop (Ps) color correction functions. (C) Results of diagonal correction using the *exact* reference point obtained directly from the color chart. (D) Our results. (E) Ground truth images. In (B) and (C), we pick the best result between the auto-color (AC) and auto-tone (AT) functions and between the sRGB (sRGB) and "linearized" sRGB (LRGB) [1,8] based on △E values, respectively.