Tunable color correction for noisy images

Ryo Yamakabe,[a] Yusuke Monno,[a,*] Masayuki Tanaka,[a,b] and Masatoshi Okutomi[a]

[a]Tokyo Institute of Technology, School of Engineering, Department of Systems and Control Engineering, Meguro-ku, Tokyo, Japan

[b]National Institute of Advanced Industrial Science and Technology, Artificial Intelligence Research Center, Koto-ku, Tokyo, Japan

Abstract. Color correction is one of the most essential camera imaging operations that transforms a camera-specific RGB color space to a standard color space, typically the XYZ or the sRGB color space. Linear color correction (LCC) and polynomial color correction (PCC) are two widely used methods; they perform the color space transformation using a color correction matrix. Owing to the use of high-order terms, PCC generally achieves lower colorimetric errors than LCC. However, PCC amplifies noise more severely than LCC. Consequently, for noisy images, there exists a trade-off between LCC and PCC regarding color fidelity and noise amplification. We propose a color correction framework called tunable color correction (TCC) that enables us to tune the color correction matrix between the LCC and the PCC models. We also derive a mean squared error calculation model of PCC that enables us to select the best trade-off balance in the TCC framework. We experimentally demonstrate that TCC effectively balances the trade-off for noisy images and outperforms LCC and PCC. We also generalize TCC to multispectral cases and demonstrate its effectiveness by taking the color correction for an RGB-near-infrared sensor as an example. © *2020 SPIE and IS&T* [DOI: 10.1117/1.JEI.29.3.033012]

Keywords: color correction; colorimetric characterization; RGB sensors; multispectral sensors; noise.

Paper 191180 received Dec. 27, 2019; accepted for publication May 21, 2020; published online Jun. 4, 2020.

1 Introduction

In color digital cameras, the spectral sensitivity functions of RGB color filters are device-dependent and usually differ from those of the human visual system. Therefore, color correction or colorimetric characterization is an essential camera imaging operation that transforms a device-dependent RGB color space to a standard or a desired color space, typically the device-independent XYZ or the display sRGB color space.[1,2]

The research on color correction has a long history, and many color correction methods, including least-squares regression-based methods,[2–6] look-up-table-based methods,[7–12] and neural network-based methods,[13–16] have been proposed. Among these methods, linear color correction (LCC)[2–4] and polynomial color correction (PCC)[5] are the two most widely used methods. LCC is performed by multiplying camera RGB values by a linear 3×3 color correction matrix. In contrast, PCC exploits high-order terms in addition to the first-order linear terms used in LCC, resulting in a larger matrix size depending on the polynomial order. In LCC and PCC, the color correction matrix is calculated by least-squares regression to minimize the mean colorimetric error for training samples, typically color patches in a colorchecker.

It is known that, when a captured image contains noise, LCC and PCC amplify the noise. Therefore, one challenge of the color correction is to suppress the noise amplification while preserving high color fidelity. Owing to the use of high-order terms, PCC generally achieves lower colorimetric errors than LCC. However, PCC amplifies the noise more severely than LCC because the high-order terms usually cause larger noise amplification. Consequently, for noisy

*Address all correspondence to Yusuke Monno, E-mail: ymonno@ok.sc.e.titech.ac.jp

1017-9909/2020/$28.00© 2020 SPIE and IS&T

images, there exists a trade-off between LCC and PCC regarding color fidelity and noise amplification.

In this paper, we propose a color correction framework called tunable color correction (TCC) that enables us to tune the color correction matrix between the LCC and the PCC models by a tuning parameter. We also derive a mean squared error (MSE) calculation model of PCC that enables us to select an optimal tuning parameter value for TCC, providing the best trade-off balance between LCC and PCC. We experimentally demonstrate that TCC effectively balances the trade-off regarding color fidelity and noise amplification and outperforms LCC and PCC. We also generalize TCC to multispectral cases and demonstrate its effectiveness by taking the color correction for an RGB-near-infrared (RGB-NIR) sensor as an example.

The effect of noise amplification on color correction has been analyzed in some works.[17–21] These analyses are generally based on the LCC model to mathematically derive an optimal color correction matrix for noisy images. One exception is the work by Burns and Berns[22] in which noise amplification effects on nonlinear transformations are analyzed in a general form using a Taylor series. Unlike these previous works, our TCC framework and MSE calculation model provide a method and analysis to derive an optimal color correction matrix across the LCC and the PCC models.

The preliminary version of this paper is published in the conference proceedings.[23] In this paper, we provide three major extensions as follows: (i) we describe the detailed formulation of the TCC framework, including a more complete derivation of the MSE calculation model of PCC. (ii) We generalize the TCC framework to multispectral cases and demonstrate its effectiveness for an RGB-NIR sensor. (iii) We report extensive experimental results, including the performance comparisons with state-of-the-art color correction methods other than LCC and PCC.

The remainder of this paper is organized as follows. Section 2 briefly reviews existing color correction methods. Section 3 introduces the trade-off between LCC and PCC. Section 4 details our proposed TCC framework and MSE calculation model. Section 5 reports experimental results, and Sec. 6 concludes the paper.

2 Related Works

Existing color correction methods can roughly be classified into three categories: (i) least-squares regression-based methods,[2–6] (ii) look-up-table-based methods,[7–12] and (iii) neural network-based methods.[13–16] The focus of this paper is least-squares regression-based methods, which are widely adopted in camera imaging pipelines due to the simplicity of the algorithm and implementation. In what follows, we briefly review existing regression-based methods based on a color correction matrix.

2.1 *Methods Without Considering Noise*

LCC is the most widely used method; it performs the color space transformation by multiplying camera RGB values by a linear 3×3 color correction matrix.[2–4] Many existing works have extended LCC to address specific problems. Finlayson and Drew[24] proposed a constrained least-squares regression, which constrains such that the color of a particular surface, typically a white surface, is reproduced without errors. Bianco et al.[25–27] proposed an improved color correction pipeline that considers white balancing errors. Zhang and Liu[28] and Vazquez-Corral et al.[29] proposed a color correction model that considers the human color perception or preference. Funt and Bastani[30] and Finlayson et al.[31,32] proposed an intensity independent or compensated method to alleviate the shading effect. Andersen and Hardeberg,[33] Andersen and Connah,[34] and Mackiewicz et al.[35] proposed a piece-wise LCC model based on hue planes to improve color reproduction accuracy. Finlayson and Drew[36] proposed a method without the need for training samples based on a statistical assumption of the spectral reflectance, which was also exploited in a spectral reflectance estimation-based color correction method.[37]

PCC[5] is another widely used method; it extends LCC by exploiting high-order terms in addition to the first-order linear terms used in LCC. The matrix size of PCC is larger than that of LCC

and depends on the used polynomial terms. Several works have extended PCC. Bianco et al.[38,39] proposed a genetic algorithm-based method to automatically find the best set of polynomial terms. Finlayson et al.[6] proposed root-polynomial color correction (RPCC) that modified PCC by taking the k'th root of each k-degree term to achieve exposure independence of the color correction matrix.

The above-mentioned methods have successfully addressed each problem of color correction. However, these methods do not consider the noise effect. Although some recent methods[6,34] experimentally demonstrate the robustness to noise, they still do not explicitly model the noise effect.

2.2 *Methods Considering Noise*

Some existing works have analyzed the noise effect on a color measurement system, mostly in the literature of spectral sensitivity designs[40–44] and color correction.[17–22] The purpose of spectral sensitivity designs is to mathematically derive an optimal set of color filters in the presence of noise, whereas the purpose of color correction is to derive an optimal color correction matrix for a given set of color filters or measurements. The latter is our focus and is reviewed below.

Tan and Acharya[17] and Barnhöfer et al.[18] mathematically described the trade-off between color fidelity and noise amplification in LCC and presented a method of calculating an optimal LCC matrix in terms of MSE. Quan[19] derived a comprehensive error metric for LCC considering the signal-dependent Poisson noise. Lim and Silverstein[20] proposed a method to derive an optimal LCC matrix in a spatially varying manner based on the covariance matrix calculation of observed color signals within each local window. Trussell and Vrhel[21] proposed a locally adaptive LCC method considering the signal-dependent Poisson noise. Another class of methods addressed the noise issue by solving an optimization problem[45,46] or incorporating additional signal processing steps such as average filtering,[47,48] discrete cosine transform,[49] and denoising.[50]

As an analytical approach, some works[17–21] mathematically modeled the color correction problem considering the noise. However, they are not beyond the scope of LCC. To the best of our knowledge, the only exception is the work by Burns and Berns,[22] in which the noise effect on a nonlinear transformation was analyzed in a very general form using a Taylor series. Unlike these works, our TCC framework provides a mathematical method and analysis to explicitly derive an optimal color correction matrix across the LCC and the nonlinear PCC models.

3 Trade-Off between LCC and PCC

In this section, we first introduce a general formulation of LCC and PCC and present our color correction setup. We then exemplify the trade-off between LCC and PCC using simulated colorchecker images.

3.1 *General Formulation*

LCC and PCC are generally formulated in matrix form as

$$\mathbf{q} = \mathbf{M}\mathbf{p}, \tag{1}$$

where $\mathbf{p} \in \mathbb{R}^N$ is an input N-dimensional vector formed by camera RGB values, $\mathbf{q} \in \mathbb{R}^3$ is an output three-dimensional color-corrected vector in a target color space, and $\mathbf{M} \in \mathbb{R}^{3\times N}$ is a color correction matrix. The dimension number N depends on the color correction model and how many terms are used for the color correction. In this work, we consider that LCC and PCC include the bias term. In that case, the input vector of LCC is formed as

$$\mathbf{p}_{\text{lcc}} = [1, p_R, p_G, p_B]^{\text{T}}, \tag{2}$$

where p_R, p_G, and p_B are the camera intensity values of R, G, and B channels, respectively. PCC uses higher-order terms in addition to the first-order linear terms. For example, the input vector of the second-order PCC is formed as

$$\mathbf{p}_{\mathrm{pcc}} = [1, p_R, p_G, p_B, p_R p_G, p_R p_B, p_G p_B, p_R^2, p_G^2, p_B^2]^{\mathrm{T}}. \tag{3}$$

In LCC and PCC, the color correction matrix is typically calculated using training color patches of a colorchecker by least-squares regression as

$$\hat{\mathbf{M}} = \arg\min_{\mathbf{M}} \|\mathbf{Q}_t - \mathbf{M}\mathbf{P}_t\|_F^2 = \mathbf{Q}_t \mathbf{P}_t^*, \tag{4}$$

where $\|\cdot\|_F^2$ is the Frobenius norm, $\mathbf{Q}_t \in \mathbb{R}^{3\times K}$ is a matrix containing the color vectors of K training patches in the target color space, $\mathbf{P}_t \in \mathbb{R}^{N\times K}$ is a matrix containing the corresponding input vectors formed by the camera RGB values of the patches, and $\mathbf{P}_t^* \in \mathbb{R}^{K\times N}$ is the pseudoinverse matrix of $\mathbf{P}_t$. The color correction matrix $\hat{\mathbf{M}} \in \mathbb{R}^{3\times N}$ is calculated to minimize the mean colorimetric error for the training patches, typically by assuming noise-free camera RGB values obtained in a color calibration phase. However, such a precalculated matrix amplifies noise when applied to noisy images in a real situation.

3.2 *Color Correction Setup*

In this work, we focus on the second-order PCC to design our proposed TCC for two reasons: (i) As experimentally reported in some existing works,[16,27] higher-order PCC does not necessarily provide lower colorimetric errors due to the over-fitting problem caused by the statistical differences between the training and the test samples. (ii) We observe that higher-order PCC amplifies the noise too significantly and is not beneficial when noisy images are considered color correction inputs. In what follows, we denote the second-order PCC as PCC for notation simplicity.

We consider the target color space to be the sRGB color space for the displaying purpose. We omit auto white balancing and directly transform the camera RGB values under some recording illumination to the sRGB values under the canonical CIE D65 illumination. As in some existing works,[20,40] we also assume white Gaussian noise due to its simplicity for model derivation, though real sensor noise is better modeled Poisson–Gaussian noise.[51]

3.3 *Trade-off Examples*

To show the trade-off between LCC and PCC, we performed an experiment using simulated X-Rite ColorChecker SG images, where 96 patches (We removed white, gray, and black patches repeated at the borders of the colorchecker.) of 200×200 pixels were simulated as shown in Fig. 1. The target sRGB image of Fig. 1(a) was simulated using the spectral power distribution of the CIE D65 illumination[52] and the spectral reflectance profiles of the 96 patches.[53] The XYZ values were first calculated using the XYZ color matching functions[52] and then transformed to the sRGB values by the XYZ-to-sRGB transformation matrix. The camera RGB image of Fig. 1(b) was simulated using the camera sensitivity functions of an Olympus E-PL2 camera[54] and the CIE A (incandescent) illumination.[52] Additive white Gaussian noise was added to each color channel to simulate the noisy camera RGB image.

Color correction was then performed to transform the noisy camera RGB image to the target sRGB image. The results of LCC and PCC are shown in Figs. 1(c) and 1(f), respectively. To evaluate the color accuracy and the amplified noise of the results, we calculated color bias and color standard deviation. For all pixels in each color patch, the mean intensity of each channel $(\overline{\mu}_R, \overline{\mu}_G, \overline{\mu}_B)$ and the standard deviation of each channel $(\hat{\sigma}_R, \hat{\sigma}_G, \hat{\sigma}_B)$ were calculated using the 200×200 color corrected samples. Then the color bias and the color standard deviation for each patch was calculated as

$$\begin{aligned} \text{color bias} &= \sqrt{(I_R - \overline{\mu}_R)^2 + (I_G - \overline{\mu}_G)^2 + (I_B - \overline{\mu}_B)^2}, \\ \text{color standard deviation} &= \sqrt{\hat{\sigma}_R^2 + \hat{\sigma}_G^2 + \hat{\sigma}_B^2}, \end{aligned} \tag{5}$$

where (I_R, I_G, I_B) are the ground-truth sRGB values for the patch. Using the above equations, MSE for each patch was then calculated as

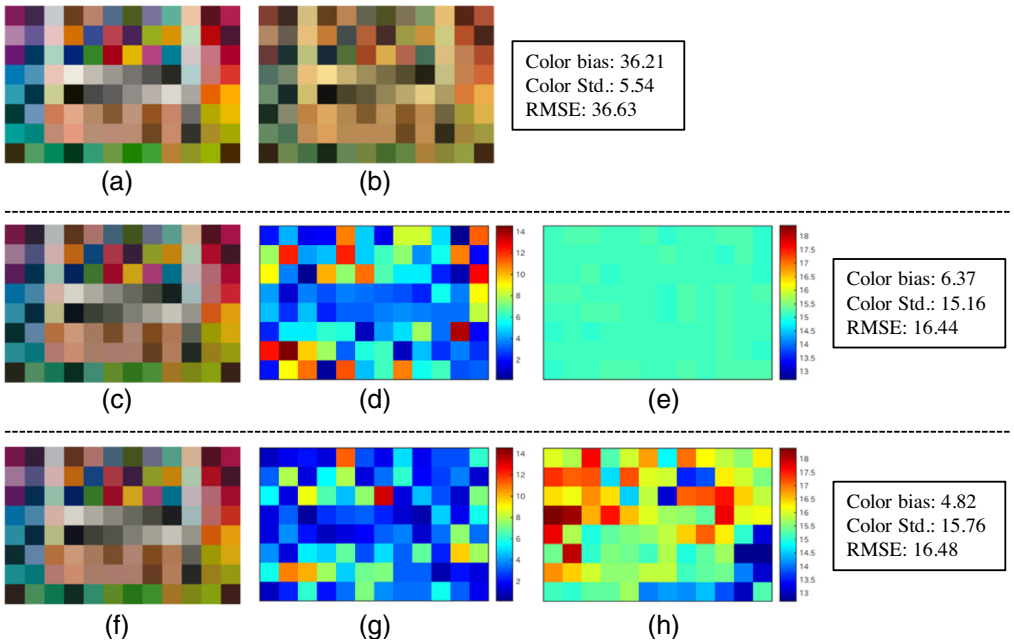

Fig. 1 Trade-off between LCC and PCC regarding color fidelity and noise amplification. Color correction was performed to transform the noisy camera RGB image (b) to the target sRGB image (a). The images (c) and (f) show the color correction results by LCC and PCC. As we observe in (d) and (g), PCC generally achieves lower color bias errors. In contrast, as we observe in (e) and (h), PCC causes higher color standard deviation (Std.). The mean values are shown in the boxes on the right.

$$\begin{aligned}\text{MSE} &= \text{mean}\left(\sum \|\text{ground truth sRGB values} - \text{color corrected values}\|^2\right), \\ &= \text{color bias}^2 + \text{color standard deviation}^2, \end{aligned} \tag{6}$$

where MSE can be derived from the square of the color bias and the square of the color standard deviation assuming zero-mean Gaussian error distribution.[55]

As we observe from Figs. 1(d) and 1(g), PCC generally achieves lower color bias errors. In contrast, as we observe from Figs. 1(e) and 1(h), PCC causes higher color standard deviation, i.e., larger noise amplification. The mean of all patches are shown in the boxes on the right in Fig. 1. From those mean values, we observe the trade-off between LCC and PCC regarding the color fidelity and the noise amplification. To select the best trade-off balance between LCC and PCC by our TCC framework, we evaluate MSE or root MSE (i.e., RMSE $= \sqrt{\text{MSE}}$), which is considered a general metric that includes the evaluation of both the color bias error and the color standard deviation, as expressed in Eq. (6). Later in this paper, we will show how MSE is modeled in the TCC framework.

4 Proposed Tunable Color Correction

4.1 *TCC Framework Overview*

Figure 2 shows the overview of our proposed TCC framework. The keys of TCC are two models: the TCC matrix calculation model and the MSE calculation model. The TCC matrix model is proposed to make the color correction matrix tunable between the LCC and the PCC models by a tuning parameter λ. The MSE model is derived to select an optimal parameter value $\hat{\lambda}$ that provides the best trade-off balance between the LCC and the PCC models. The color correction matrix calculated with the optimal parameter value $\hat{\mathbf{M}}(\hat{\lambda})$ is used to transform the input noisy camera RGB image. We detail each calculation model below.

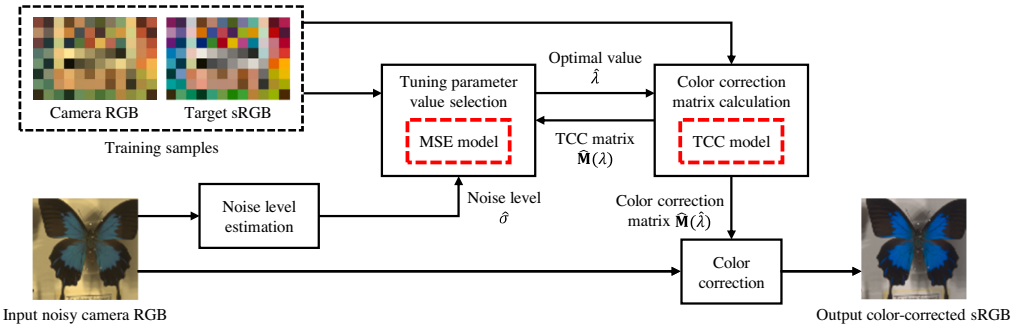

Fig. 2 The overview of our proposed TCC framework in which we propose two models: the TCC matrix calculation model (TCC model) and the MSE calculation model (MSE model). The TCC model is proposed to make the color correction matrix tunable between the LCC and the PCC models by a tuning parameter λ. The MSE model is derived to select an optimal parameter value $\hat{\lambda}$ that provides the best trade-off balance between the LCC and the PCC models. The color correction matrix calculated with the optimal parameter value $\hat{\mathbf{M}}(\hat{\lambda})$ is used to transform the input noisy camera RGB image.

4.2 *TCC Matrix Calculation Model*

Our proposed TCC is motivated by the observation that large color correction coefficients for high-order terms of PCC cause large noise amplification. This observation leads us to calculate our proposed TCC matrix as

$$\hat{\mathbf{M}}(\lambda) = \arg\min_{\mathbf{M}} \left(\|\mathbf{Q}_t - \mathbf{M}\mathbf{P}_t\|_F^2 + \frac{1}{\lambda}\|\mathbf{W} \circ \mathbf{M}\|_F^2 \right), \tag{7}$$

where $\hat{\mathbf{M}}(\lambda) \in \mathbb{R}^{3\times N}$ is the TCC matrix. The resultant TCC matrix by our formulation takes the same form and size as the PCC matrix. As described in Sec. 3.2, in this paper, we focus on designing the TCC matrix tunable between the LCC matrix and the second-order PCC matrix with the input forms of Eqs. (2) and (3), respectively. In this case, the size of the derived TCC matrix is the same as that of the second-order PCC matrix, i.e., 3×10 ($N = 10$), according to the second-order PCC input form of Eq. (3). The first term of Eq. (7) is the data fidelity term, which takes the same form as Eq. (4) assuming the second-order PCC input form of Eq. (3). The second term is our proposed constrain term, where $\circ$ represents the element-wise product and $\mathbf{W} \in \mathbb{R}^{3\times N}$ is a binary weighting matrix designed to constrain the coefficients for the high-order terms. Specifically, in the case of the considered second-order PCC input form of Eq. (3), the weighting matrix is designed as

$$\mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}, \tag{8}$$

where the binary weights are applied to the coefficients for the second-order terms.

Our TCC matrix has the following interesting properties:

$$\begin{aligned} \lim_{\lambda\to 0} \hat{\mathbf{M}}(\lambda) &= [\hat{\mathbf{M}}_{\text{lcc}} \quad \mathbf{0}], \\ \lim_{\lambda\to \infty} \hat{\mathbf{M}}(\lambda) &= \hat{\mathbf{M}}_{\text{pcc}}, \end{aligned} \tag{9}$$

where $\hat{\mathbf{M}}_{\text{lcc}} \in \mathbb{R}^{3\times 4}$ is the LCC matrix derived by the standard formulation of Eq. (4) using the input form of Eq. (2) and $\hat{\mathbf{M}}_{\text{pcc}} \in \mathbb{R}^{3\times N}$ is the second-order PCC matrix derived by Eq. (4) using the input form of Eq. (3). The first equation indicates that, if $\lambda \to 0$, the resultant TCC matrix $[\hat{\mathbf{M}}_{\text{lcc}} \quad \mathbf{0}] \in \mathbb{R}^{3\times N}$, in which $\mathbf{0} \in \mathbb{R}^{3\times(N-4)}$ is the zero matrix corresponding to the coefficients for the second-order terms, identically works as the standard LCC matrix $\hat{\mathbf{M}}_{\text{lcc}}$. By contrast, the

second equation indicates that, if $\lambda \to \infty$, the resultant TCC matrix is identical to the standard PCC matrix $\hat{\mathbf{M}}_{\text{pcc}}$. Therefore, the properties in Eq. (9) indicate that the TCC matrix is tunable between the LCC and the PCC models by a tuning parameter λ.

Equation (7) is solved in the vectorized form as

$$\hat{\mathbf{m}}_v(\lambda) = \left(\mathbf{P}_v^{\mathrm{T}}\mathbf{P}_v + \frac{1}{\lambda}\mathbf{W}_d^{\mathrm{T}}\mathbf{W}_d\right)^{-1}\mathbf{P}_v^{\mathrm{T}}\mathbf{q}_v, \tag{10}$$

where $\hat{\mathbf{m}}_v(\lambda) \in \mathbb{R}^{3N}$ and $\mathbf{q}_v \in \mathbb{R}^{3K}$ are the vectorized form of $\hat{\mathbf{M}}(\lambda)$ and $\mathbf{Q}_t$, respectively. $\mathbf{P}_v = [\mathbf{P}_1; \mathbf{P}_2; \ldots; \mathbf{P}_K] \in \mathbb{R}^{3K\times 3N}$ is the matrix containing the input PCC vectors of K training samples, where $\mathbf{P}_k \in \mathbb{R}^{3\times 3N}$ is the block diagonal matrix with diagonal elements that are the input vector for k'th training sample and $\mathbf{W}_d \in \mathbb{R}^{3N\times 3N}$ is the diagonal matrix with diagonal elements that are the weights of $\mathbf{W}$.

4.3 *MSE Calculation Model*

We here derive the MSE calculation model for selecting an optimal tuning parameter value that provides the best trade-off balance in terms of MSE. Based on the MSE calculation for each training sample, the optimal value $\hat{\lambda}$ is selected as

$$\hat{\lambda} = \arg\min_{\lambda} \sum_{k=1}^{K} \text{MSE}_k(\lambda), \tag{11}$$

where $\hat{\lambda}$ minimizes the sum of MSEs for all training samples. In the presence of noise, MSE for the k'th sample is estimated as

$$\text{MSE}_k(\lambda) = E(\|\mathbf{q}_k - \hat{\mathbf{M}}(\lambda)\mathbf{p}_k\|_2^2), \tag{12}$$

where $E(\cdot)$ is the expectation operator, $\mathbf{q}_k$ is the target sRGB vector, and $\mathbf{p}_k$ is a random variable representing the input vector with noise. Let $\boldsymbol{\mu}_{\mathbf{p}_k} \in \mathbb{R}^N$ be the expectation of $\mathbf{p}_k$, i.e., $\boldsymbol{\mu}_{\mathbf{p}_k} = E(\mathbf{p}_k)$ and $\Sigma_{\mathbf{p}_k}$ be the variance-covariance matrix for the input noisy vector. Then Eq. (12) is rewritten as

$$\text{MSE}_k(\lambda) = \|\mathbf{q}_k - \hat{\mathbf{M}}(\lambda)\boldsymbol{\mu}_{\mathbf{p}_k}\|_2^2 + \text{tr}[\hat{\mathbf{M}}(\lambda)\Sigma_{\mathbf{p}_k}\hat{\mathbf{M}}(\lambda)^{\mathrm{T}}], \tag{13}$$

where tr[·] is the trace operator. The first term represents the expected color bias error, and the second term represents the noise variance of the color-corrected values.

Equation (13) indicates that MSE for each training sample is calculated from the expectation vector $\boldsymbol{\mu}_{\mathbf{p}_k}$ and the variance-covariance matrix $\Sigma_{\mathbf{p}_k}$. Such calculations have been performed for LCC.[17,19] However, to the best of our knowledge, none of the existing works have explicitly extended the calculations to PCC. In what follows, we describe the calculation of $\boldsymbol{\mu}_{\mathbf{p}_k}$ and $\Sigma_{\mathbf{p}_k}$ for the second-order PCC.

We first derive the expectation of the input vector of the form of the second-order PCC in Eq. (3). If an input image contains noise, the noisy camera RGB values are represented as

$$p_R = g_R + n_R, \quad p_G = g_G + n_G, \quad p_B = g_B + n_B, \tag{14}$$

where g_R, g_G, and g_B are latent noise-free camera RGB values and n_R, n_G, and n_B represent the noise of each color channel.

In the following derivation, we assume that the noise of each channel is signal-independent zero-mean Gaussian noise and the noise variance of each channel is σ_R^2, σ_G^2, or σ_B^2. We also assume that the noise of each channel is independent of the latent noise-free camera RGB values and independent of each other. These assumptions have often been made in the literature.[20,40] Based on the assumptions, the expectation vector $\boldsymbol{\mu}_{\mathbf{p}_k} = E(\mathbf{p}_{\text{pcc}})$ is derived as

$$\boldsymbol{\mu}_{\mathbf{p}_k} = [1, g_R, g_G, g_B, g_R g_G, g_R g_B, g_G g_B, g_R^2 + \sigma_R^2, g_G^2 + \sigma_G^2, g_B^2 + \sigma_B^2]^{\mathrm{T}}. \tag{15}$$

In Eq. (15) and all of the following derivations included in the Appendix, we omit the training sample index k from all vector and matrix elements (e.g., g_R, g_G, g_B, and so on) for notation simplicity without loss of generality. The detailed derivations of the expectation of each linear, cross, and squared term of Eq. (15) are found in the Appendix.

We next derive the variance-covariance matrix $\boldsymbol{\Sigma}_{\mathbf{p}_k}$. Since the bias term does not affect the calculation of the variance-covariance matrix, we rewrite the second term of Eq. (13) as

$$\mathrm{tr}[\hat{\mathbf{M}}(\lambda)\boldsymbol{\Sigma}_{\mathbf{p}_k}\hat{\mathbf{M}}(\lambda)^{\mathrm{T}}] = \mathrm{tr}[\tilde{\mathbf{M}}(\lambda)\tilde{\boldsymbol{\Sigma}}_{\mathbf{p}_k}\tilde{\mathbf{M}}(\lambda)^{\mathrm{T}}], \tag{16}$$

where $\tilde{\mathbf{M}}(\lambda) \in \mathbb{R}^{3\times N-1}$ and $\tilde{\boldsymbol{\Sigma}}_{\mathbf{p}_k} \in \mathbb{R}^{(N-1)\times(N-1)}$ are the matrices with the removal of the bias term-related components. With this notation, the variance-covariance matrix $\tilde{\boldsymbol{\Sigma}}_{\mathbf{p}_k}$ is derived as Eqs. (18) and (19), where $V(\cdot)$ is the variance operator, $C(\cdot,\cdot)$ is the covariance operator, and the variables of the input vector are rewritten as

$$\begin{aligned}\mathbf{p}_{\mathrm{pcc}} &= [1, p_R, p_G, p_B, p_R p_G, p_R p_B, p_G p_B, p_R^2, p_G^2, p_B^2]^{\mathrm{T}} \\ &= [1, p_R, p_G, p_B, p_{RG}, p_{RB}, p_{GB}, p_{R^2}, p_{G^2}, p_{B^2}]^{\mathrm{T}}.\end{aligned} \tag{17}$$

The diagonal elements of $\tilde{\boldsymbol{\Sigma}}_{\mathbf{p}_k}$ are the variance of the variables, and the off-diagonal elements are the covariances between all possible pairs of the variables. The detailed derivations of each variance or covariance component are found in the Appendix. Using Eqs. (13), (15), (16), and (19), we calculate MSE for each training sample to find an optimal tuning parameter value:

$$\begin{aligned}\boldsymbol{\Sigma}_{p_k} = &\begin{bmatrix}
V(p_R) & C(p_R,p_G) & C(p_R,p_B) & C(p_R,p_{RG}) & C(p_R,p_{RB}) & C(p_R,p_{GB}) & C(p_R,p_{R^2}) & C(p_R,p_{G^2}) & C(p_R,p_{B^2}) \\
C(p_R,p_G) & V(p_G) & C(p_G,p_B) & C(p_G,p_{RG}) & C(p_G,p_{RB}) & C(p_G,p_{GB}) & C(p_G,p_{R^2}) & C(p_G,p_{G^2}) & C(p_G,p_{B^2}) \\
C(p_R,p_B) & C(p_G,p_B) & V(p_B) & C(p_B,p_{RG}) & C(p_B,p_{RB}) & C(p_B,p_{GB}) & C(p_B,p_{R^2}) & C(p_B,p_{G^2}) & C(p_B,p_{B^2}) \\
C(p_R,p_{RG}) & C(p_G,p_{RG}) & C(p_B,p_{RG}) & V(p_{RG}) & C(p_{RG},p_{RB}) & C(p_{RG},p_{GB}) & C(p_{RG},p_{R^2}) & C(p_{RG},p_{G^2}) & C(p_{RG},p_{B^2}) \\
C(p_R,p_{RB}) & C(p_G,p_{RB}) & C(p_B,p_{RB}) & C(p_{RG},p_{RB}) & V(p_{RB}) & C(p_{RB},p_{GB}) & C(p_{RB},p_{R^2}) & C(p_{RB},p_{G^2}) & C(p_{RB},p_{B^2}) \\
C(p_R,p_{GB}) & C(p_G,p_{GB}) & C(p_B,p_{GB}) & C(p_{RG},p_{GB}) & C(p_{RB},p_{GB}) & V(p_{GB}) & C(p_{GB},p_{R^2}) & C(p_{GB},p_{G^2}) & C(p_{GB},p_{B^2}) \\
C(p_R,p_{R^2}) & C(p_G,p_{R^2}) & C(p_B,p_{R^2}) & C(p_{RG},p_{R^2}) & C(p_{RB},p_{R^2}) & C(p_{GB},p_{R^2}) & V(p_{R^2}) & C(p_{R^2},p_{G^2}) & C(p_{R^2},p_{B^2}) \\
C(p_R,p_{G^2}) & C(p_G,p_{G^2}) & C(p_B,p_{G^2}) & C(p_{RG},p_{G^2}) & C(p_{RB},p_{G^2}) & C(p_{GB},p_{G^2}) & C(p_{R^2},p_{G^2}) & V(p_{G^2}) & C(p_{G^2},p_{B^2}) \\
C(p_R,p_{B^2}) & C(p_G,p_{B^2}) & C(p_B,p_{B^2}) & C(p_{RG},p_{B^2}) & C(p_{RB},p_{B^2}) & C(p_{GB},p_{B^2}) & C(p_{R^2},p_{B^2}) & C(p_{G^2},p_{B^2}) & V(p_{B^2})
\end{bmatrix},\end{aligned} \tag{18}$$

$$\begin{bmatrix}
\sigma_R^2 & 0 & 0 & g_G\sigma_R^2 & g_B\sigma_R^2 & 0 & 2g_R\sigma_R^2 & 0 & 0 \\
0 & \sigma_G^2 & 0 & g_R\sigma_G^2 & 0 & g_B\sigma_G^2 & 0 & 2g_G\sigma_G^2 & 0 \\
0 & 0 & \sigma_B^2 & 0 & g_R\sigma_B^2 & g_G\sigma_B^2 & 0 & 0 & 2g_B\sigma_B^2 \\
g_G\sigma_R^2 & g_R\sigma_G^2 & 0 & g_R^2\sigma_G^2+g_G^2\sigma_R^2+\sigma_R^2\sigma_G^2 & g_Gg_B\sigma_R^2 & g_Rg_B\sigma_G^2 & 2g_Rg_G\sigma_R^2 & 2g_Rg_G\sigma_G^2 & 0 \\
g_B\sigma_R^2 & 0 & g_R\sigma_B^2 & g_Gg_B\sigma_R^2 & g_R^2\sigma_B^2+g_B^2\sigma_R^2+\sigma_R^2\sigma_B^2 & g_Rg_G\sigma_B^2 & 2g_Rg_B\sigma_R^2 & 0 & 2g_Rg_B\sigma_B^2 \\
0 & g_B\sigma_G^2 & g_G\sigma_B^2 & g_Rg_B\sigma_G^2 & g_Rg_G\sigma_B^2 & g_G^2\sigma_B^2+g_B^2\sigma_G^2+\sigma_G^2\sigma_B^2 & 0 & 2g_Gg_B\sigma_G^2 & 2g_Gg_B\sigma_B^2 \\
2g_R\sigma_R^2 & 0 & 0 & 2g_Rg_G\sigma_R^2 & 2g_Rg_B\sigma_R^2 & 0 & 4g_R^2\sigma_R^2+2\sigma_R^4 & 0 & 0 \\
0 & 2g_G\sigma_G^2 & 0 & 2g_Rg_G\sigma_G^2 & 0 & 2g_Gg_B\sigma_G^2 & 0 & 4g_G^2\sigma_G^2+2\sigma_G^4 & 0 \\
0 & 0 & 2g_B\sigma_B^2 & 0 & 2g_Rg_B\sigma_B^2 & 2g_Gg_B\sigma_B^2 & 0 & 0 & 4g_B^2\sigma_B^2+2\sigma_B^4
\end{bmatrix}. \tag{19}$$

4.4 *Generalization to Multispectral Cases*

Recent progress of imaging technology has encountered the development of various multispectral imaging systems, such as a four-band RGB-NIR camera[53] and a five-band or an eight-band multispectral camera.[56,57] In these systems, the color correction, i.e., the mapping from the sensor-specific multispectral responses to the target color space, is necessary to realize high-fidelity color visualization.[58] In what follows, we generalize the TCC framework to multispectral cases.

As observed from Eqs. (15) and (18), the expectation vector and the variance-covariance matrix has a clear structure that can naturally be extended to multispectral cases. If we consider a multispectral case with H bands, the expectation vector is generalized as

$$\boldsymbol{\mu}_{\mathbf{p}_k} = [1, g_{X_1}, \ldots, g_{X_H}, g_{X_1}g_{X_2}, \ldots, g_{X_{H-1}}g_{X_H}, g_{X_1}^2 + \sigma_{X_1}^2, \ldots, g_{X_H}^2 + \sigma_{X_H}^2]^{\mathrm{T}}, \tag{20}$$

where $\{X_1, \ldots, X_H\}$ represents the band indices. Similarly, the variance-covariance matrix is generalized as

$$\tilde{\boldsymbol{\Sigma}}_{\mathbf{p}_k} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} & \mathbf{C}_{13} \\ \mathbf{C}_{21} & \mathbf{C}_{22} & \mathbf{C}_{23} \\ \mathbf{C}_{31} & \mathbf{C}_{32} & \mathbf{C}_{33} \end{bmatrix}, \tag{21}$$

where each submatrix is structured as

$$\mathbf{C}_{11} = \begin{bmatrix} V(p_{X_1}) & \cdots & C(p_{X_1}, p_{X_H}) \\ \vdots & \ddots & \vdots \\ C(p_{X_1}, p_{X_H}) & \cdots & V(p_{X_H}) \end{bmatrix},$$

$$\mathbf{C}_{22} = \begin{bmatrix} V(p_{X_1X_2}) & \cdots & C(p_{X_1X_2}, p_{X_{H-1}X_H}) \\ \vdots & \ddots & \vdots \\ C(p_{X_1X_2}, p_{X_{H-1}X_H}) & \cdots & V(p_{X_{H-1}X_H}) \end{bmatrix},$$

$$\mathbf{C}_{33} = \begin{bmatrix} V(p_{X_1^2}) & \cdots & C(p_{X_1^2}, p_{X_H^2}) \\ \vdots & \ddots & \vdots \\ C(p_{X_1^2}, p_{X_H^2}) & \cdots & V(p_{X_H^2}) \end{bmatrix},$$

$$\mathbf{C}_{12} = \mathbf{C}_{21}^{\mathrm{T}} = \begin{bmatrix} C(p_{X_1}, p_{X_1X_2}) & \cdots & C(p_{X_1}, p_{X_{H-1}X_H}) \\ \vdots & \ddots & \vdots \\ C(p_{X_H}, p_{X_1X_2}) & \cdots & C(p_{X_H}, p_{X_{H-1}X_H}) \end{bmatrix},$$

$$\mathbf{C}_{13} = \mathbf{C}_{31}^{\mathrm{T}} = \begin{bmatrix} C(p_{X_1}, p_{X_1^2}) & \cdots & C(p_{X_1}, p_{X_H^2}) \\ \vdots & \ddots & \vdots \\ C(p_{X_H}, p_{X_1^2}) & \cdots & C(p_{X_H}, p_{X_H^2}) \end{bmatrix},$$

$$\mathbf{C}_{23} = \mathbf{C}_{32}^{\mathrm{T}} = \begin{bmatrix} C(p_{X_1X_2}, p_{X_1^2}) & \cdots & C(p_{X_1X_2}, p_{X_H^2}) \\ \vdots & \ddots & \vdots \\ C(p_{X_{H-1}X_H}, p_{X_1^2}) & \cdots & C(p_{X_{H-1}X_H}, p_{X_H^2}) \end{bmatrix}. \tag{22}$$

The derivations of each variance or covariance component is similar to the case of RGB color imaging and are found in the Appendix.

5 Experimental Results

5.1 *Results on Colorchecker Data*

We here demonstrate the property of TCC using simulated X-Rite ColorChecker SG images. As explained in Sec. 3.3, the target sRGB values (to construct $\mathbf{Q}_t$) and the noise-free camera RGB values under the CIE A illumination (to construct $\mathbf{P}_t$) were simulated using the spectral reflectance profiles of the 96 patches in the colorchecker. The LCC and the PCC matrices were calculated by Eq. (4), and the TCC matrices were calculated by Eq. (7) using a set of tuning parameter values. To focus on the clarification of the TCC property, we here assumed an ideal condition that the training and the test samples are the same and the noise variance is known. The same noise variance was used for all color channels, i.e., $\sigma_R^2 = \sigma_G^2 = \sigma_B^2 = \sigma^2$.

We first show the effect of different tuning parameter values on TCC. Figure 3 shows the results of the MSE calculation by Eq. (13) for the TCC matrices with different parameter values. Each result in Fig. 3 shows the mean of the 96 patches for the case of $\sigma = 4$. Figures 3(a)–3(c) show the color bias error, the color standard deviation, and RMSE, respectively. These values were calculated by taking the root of the first term, the second term, and the entire equation of Eq. (13). In Fig. 3, LCC and PCC are equivalent to the cases of $\lim \lambda \to 0$ and $\lim \lambda \to \infty$, respectively. Figure 3(a) shows that the color bias error of PCC ($\lim \lambda \to \infty$) is lower than that of LCC ($\lim \lambda \to 0$). This is because of the advantage of using the high-order terms in PCC. In contrast, Fig. 3(b) shows that the color standard deviation of PCC is larger than that of LCC. This is because the high-order terms in PCC cause large noise amplification. Figures 3(a) and 3(b) show that our proposed TCC transitionally moves between LCC and PCC by changing the parameter value. Figure 3(c) further indicates that there is an optimal parameter value that balances the trade-off between LCC and PCC and provides the result with the best RMSE performance. In TCC, that parameter value is selected based on the derived MSE calculation model in Sec. 4.3.

Table 1 and Fig. 4 show the RMSE results with different noise levels. For the noise-free and the low-noise cases, TCC provides results similar to those of PCC, which are better than LCC. With the increase of the noise level, RMSE of PCC becomes larger than that of LCC due to the noise amplification. In contrast, TCC consistently provides better results than both LCC and PCC for various noise levels by taking the best trade-off balance.

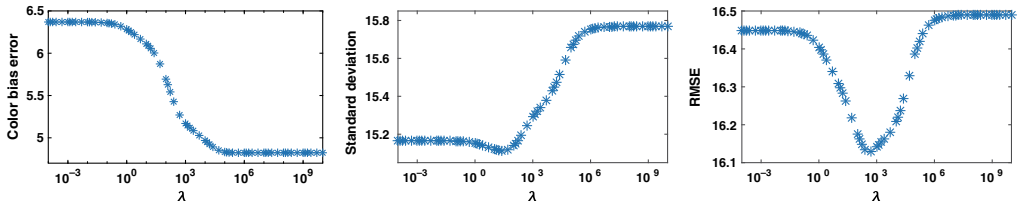

Fig. 3 Results of the MSE calculation by Eq. (13) for the TCC matrices with different tuning parameter values. Each result shows the mean of the 96 patches with $\sigma = 4$. (a) Color bias error, (b) color standard deviation, and (c) RMSE. Note that LCC and PCC are equivalent to the cases of $\lim \lambda \to 0$ and $\lim \lambda \to \infty$, respectively. As shown in (c), there is an optimal point that provides the minimium RMSE. That point is selected by our TCC framework.

Table 1 RMSE comparison for the 96 patches of the colorchecker.

σ	0	2	4	6	8	10
LCC	6.37	9.91	16.45	23.62	31.00	38.43
PCC	**4.81**	9.24	16.49	24.17	31.99	39.88
TCC	**4.81**	**9.17**	**16.13**	**23.41**	**30.81**	**38.23**

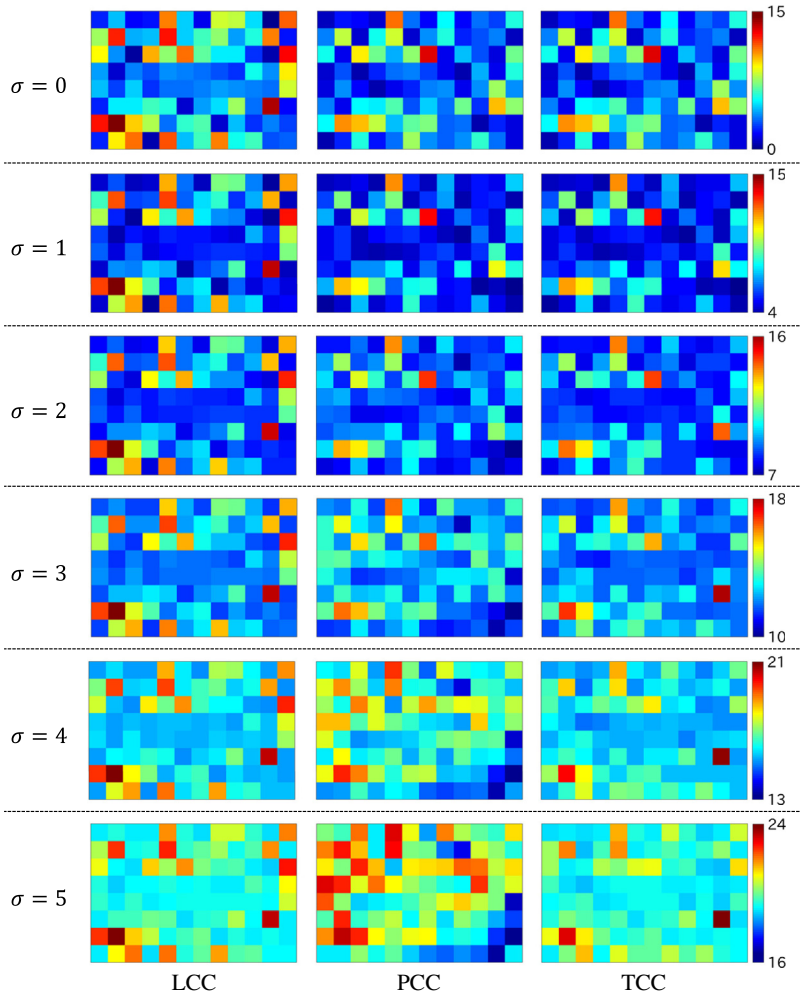

Fig. 4 RMSE maps for the colorchecker data. For the noise-free and the low-noise cases, TCC provides results similar to those of PCC, which are better than LCC. With the increase of the noise level, RMSE of PCC becomes larger than that of LCC due to the noise amplification. In contrast, TCC consistently provides better results than both LCC and PCC for various noise levels by taking the best trade-off balance.

5.2 *Results on Hyperspectral Image Dataset*

We next evaluate the performance of TCC in comparison with LCC, PCC, and other state-of-the-art methods. For this purpose, we used a hyperspectral image dataset, which contains 40 scenes of various objects,[53] and considered a more realistic condition that the training and the test samples are different and the noise variance is estimated from the input noisy camera RGB image. To make the training and the test samples, we divided the dataset into 20 training scenes and 20 test scenes. We used the CIE A (incandescent), the CIE D65 (standard), and the CIE F12 (fluorescent) illuminations[52] and the Olympus E-PL2 camera sensitivity[54] to generate the camera RGB image. The noise variance was estimated using the noise level estimation method by Liu et al.[59] We compared TCC with seven methods: LCC, PCC, second-order RPCC,[6] hue-plane preserving color correction (HPPCC),[35] homography color correction (HoCC),[32] angular minimization color correction (AMCC),[30] and maximum ignorance color correction (MICC),[36] which are implemented in a color correction toolbox.[60]

Table 2 shows the RMSE comparison for the hyperspectral image dataset, where we evaluated the mean, the median, and the max RMSE of the 20 test scenes. In Table 2, we observe that TCC provides the lowest mean RMSE for almost all tested situations with various noise levels and illumination conditions. This result validates the effectiveness of our TCC framework that tries to take the best trade-off balance based on the MSE calculation model, even in the case that

Table 2 RMSE comparison for the hyperspectral image dataset in the case of RGB color imaging.

	Mean								Median								Max							
σ	LCC	PCC	TCC	RPCC	HPPCC	HoCC	AMCC	MICC	LCC	PCC	TCC	RPCC	HPPCC	HoCC	AMCC	MICC	LCC	PCC	TCC	RPCC	HPPCC	HoCC	AMCC	MICC
CIE A (incandescent illumination)																								
0	11.06	**10.46**	**10.46**	10.64	10.92	13.52	13.11	11.13	8.55	8.56	8.56	7.82	**7.72**	8.87	8.56	8.43	27.56	26.28	26.28	**24.79**	26.71	34.41	33.22	27.15
2	14.04	13.90	**13.51**	14.39	14.34	16.01	15.71	14.06	12.18	12.43	12.14	12.34	**11.99**	12.33	12.20	12.06	28.91	27.93	26.89	**26.75**	28.17	35.48	34.35	28.49
4	20.29	20.95	**19.70**	21.98	21.22	21.61	21.49	20.23	19.13	19.58	**18.76**	20.64	19.79	19.26	19.25	18.93	32.60	32.40	**30.77**	31.83	32.25	38.47	37.50	32.16
6	27.43	29.02	**26.90**	30.53	28.94	28.32	28.35	27.30	26.82	28.31	**26.57**	29.63	28.20	26.79	26.96	26.67	37.77	38.62	**36.18**	38.68	37.95	42.85	42.06	37.31
8	34.73	37.43	**34.20**	39.35	36.78	35.34	35.48	34.54	34.57	37.14	34.27	38.78	36.68	34.25	34.55	**34.24**	43.78	45.90	**42.30**	46.41	44.49	48.15	47.55	43.29
10	41.98	46.05	**41.44**	48.20	44.55	42.38	42.63	41.73	42.01	46.08	**41.48**	47.82	44.96	41.73	42.13	41.76	50.07	55.21	**48.66**	55.78	51.31	53.84	53.41	49.54
CIE D65 (standard illumination)																								
0	11.19	**10.45**	**10.45**	10.90	10.83	14.12	12.76	11.32	9.12	**8.67**	**8.67**	8.86	9.24	10.51	9.18	8.95	27.62	25.73	25.73	**25.27**	26.45	35.82	32.01	27.27
2	12.24	11.63	**11.58**	12.09	11.97	15.10	13.78	12.35	10.34	10.10	**10.04**	10.26	10.49	11.76	10.50	10.19	28.07	26.20	26.24	**25.80**	26.83	36.23	32.44	27.73
4	14.89	14.54	**14.39**	15.06	14.81	17.67	16.38	14.97	13.26	13.24	**12.94**	13.52	13.46	14.82	13.63	13.12	29.36	27.57	28.69	**27.31**	28.04	37.40	33.66	29.01
6	18.39	18.29	**17.96**	18.91	18.51	21.18	19.89	18.43	17.14	17.44	16.95	17.88	17.48	18.78	17.61	**16.91**	31.38	29.69	31.15	**29.65**	30.03	39.27	35.60	31.04
8	22.30	22.41	**21.91**	23.15	22.59	25.19	23.86	22.31	21.55	21.96	**21.30**	22.78	22.04	23.44	22.23	21.32	34.01	**32.41**	33.84	32.65	32.69	41.77	38.15	33.67
10	26.39	26.69	**26.03**	27.56	26.83	29.45	28.05	26.38	26.08	26.63	**25.79**	27.42	26.71	28.30	27.01	25.85	36.97	**35.49**	36.81	35.99	35.71	44.59	41.04	36.63

Table 2 (*Continued*).

	Mean								Median								Max							
σ	LCC	PCC	TCC	RPCC	HPPCC	HoCC	AMCC	MICC	LCC	PCC	TCC	RPCC	HPPCC	HoCC	AMCC	MICC	LCC	PCC	TCC	RPCC	HPPCC	HoCC	AMCC	MICC
CIE F12 (fluorescent illumination)																								
0	11.57	**10.42**	**10.42**	10.97	11.10	13.90	13.11	11.60	8.63	8.49	8.49	8.34	**7.67**	9.28	8.93	8.66	32.68	**26.59**	**26.59**	28.36	30.44	40.58	37.85	32.32
2	16.00	15.56	**15.02**	16.46	16.30	17.55	17.03	15.98	14.12	14.26	**14.00**	14.74	14.11	14.16	14.15	14.07	34.57	29.26	**28.07**	31.22	32.84	42.02	39.43	34.20
4	24.48	25.12	**23.80**	26.61	25.77	25.14	24.96	24.39	23.40	24.08	23.18	25.48	24.75	**23.00**	23.22	23.38	39.48	35.95	**33.71**	38.26	38.71	45.90	43.65	39.08
6	33.70	35.68	**33.23**	37.62	35.88	33.74	33.81	33.54	33.29	35.21	33.12	36.93	35.59	**32.41**	32.83	33.06	46.19	44.78	**41.19**	47.40	46.45	51.46	49.61	45.76
8	42.91	46.65	42.60	48.82	45.96	**42.48**	42.74	42.69	42.91	46.50	42.42	48.44	46.18	**41.84**	42.38	42.68	53.76	55.67	**49.41**	57.43	55.07	57.98	56.50	53.29
10	51.98	58.04	51.82	60.10	55.91	**51.16**	51.59	51.70	52.31	57.69	51.90	60.01	56.08	**50.92**	51.58	52.01	61.73	69.78	**58.88**	68.24	64.00	65.05	63.89	61.20

the training and the test samples are different and the noise level is unknown. For the median RMSE, TCC provides the best performance for 9 situations among the total 18 situations (6 noise levels × 3 illuminations), which is followed by HoCC providing the best performance for 4 situations. For the max RMSE, TCC provides the best performance for 10 situations, which is followed by RPCC providing the best performance for 6 situations. Figure 5 shows the visual comparison of the RMSE maps for LCC, PCC, and TCC. For the pink and white flower paint region at the center of the image, LCC generates larger errors than PCC in the case of $\sigma = 0$ and 1, whereas PCC generates larger errors than LCC in the case of $\sigma = 2$ and 3, due to the trade-off

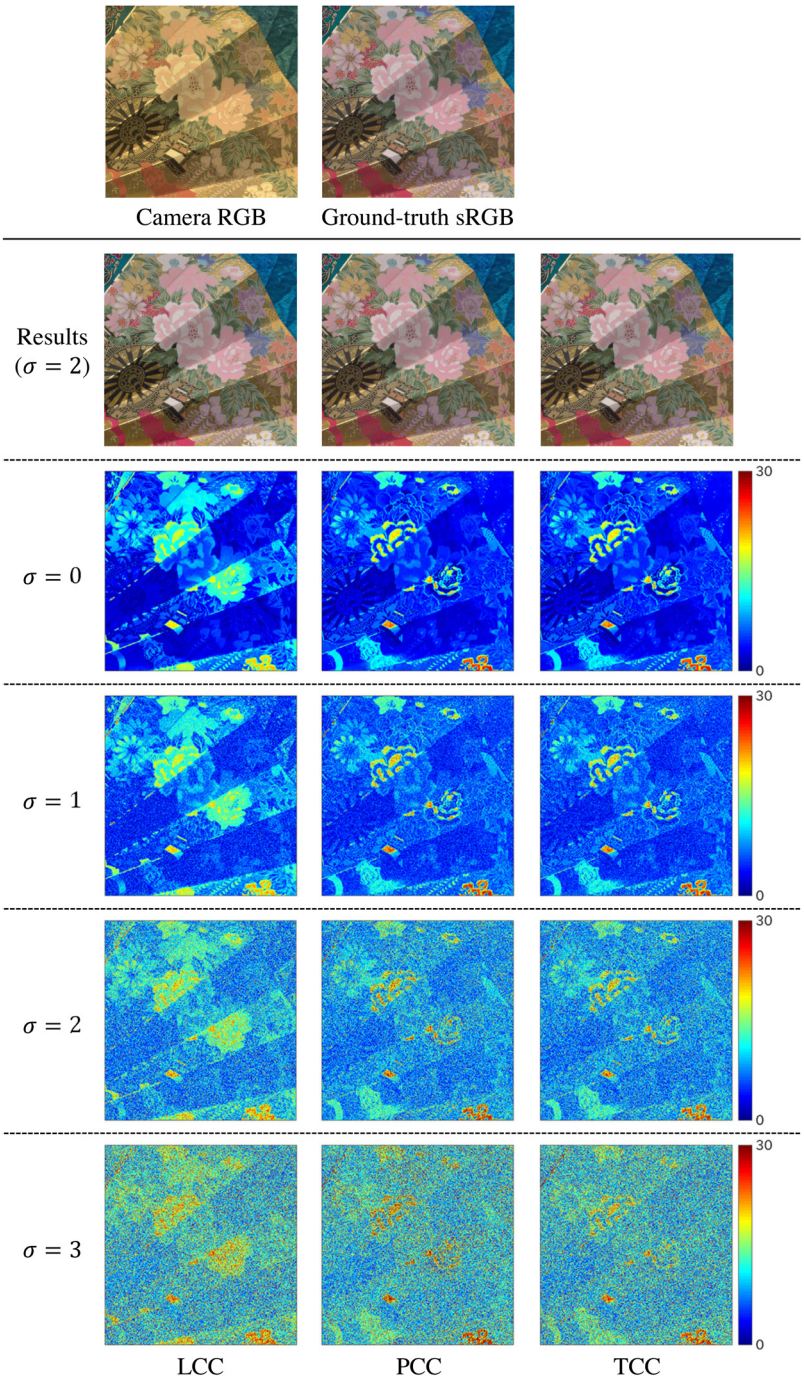

Fig. 5 Color correction results and RMSE maps for the hyperspectral image data in the case of RGB color imaging under the CIE A illumination.

regarding color fidelity and noise amplification. In contrast, TCC consistently generates errors lower than (or comparable to) LCC and PCC for all noise levels.

5.3 *Results on an RGB-NIR Imaging Case*

We next evaluate the performance of the generalized TCC for a multispectral case. We took the four-band RGB-NIR imaging case as an example and considered the mapping from the four-band RGB-NIR values to the target sRGB values. The same hyperspectral image dataset,[53] which covers the spectral range from the visible to the NIR domains (420 to 1000 nm), was used to simulate the camera RGB-NIR values assuming the RGB-NIR camera sensitivity of a real sensor.[53] Since there is no CIE-defined illumination for the RGB-NIR wavelength domain, we used the complete white illumination, which has the spectral power of one for all considered wavelengths. The input vector for the RGB-NIR case was formed as

$$\mathbf{p}_{\text{pcc}} = [1, p_R, p_G, p_B, p_I, p_R p_G, p_R p_B, p_R p_I, p_G p_B, p_G p_I, p_B p_I, p_R^2, p_G^2, p_B^2, p_I^2]^{\text{T}}, \tag{23}$$

where I represents the NIR band. The corresponding binary weighting matrix was formed as

$$\mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}. \tag{24}$$

The expectation vector $\boldsymbol{\mu}_{\mathbf{p}_k}$ and the variance-covariance matrix $\tilde{\Sigma}_{\mathbf{p}_k}$ were calculated using our derived MSE calculation model. The same 20 training scenes and 20 test scenes were used to evaluate the performance, and the same noise level estimation method[59] was applied in TCC.

Table 3 shows the RMSE comparison with LCC and PCC for the RGB-NIR imaging case. We observe that TCC generally provides the best performance also for the RGB-NIR imaging case. Figure 6 shows the visual comparison of the RMSE maps. As observed for the case of RGB color imaging, TCC can balance the trade-off between LCC and PCC, especially for the wing region of the butterfly.

5.4 *Limitations*

Although our TCC framework can successfully balance the trade-off between LCC and PCC, there are several limitations. The first limitation is that we currently can optimize MSE in the XYZ or the sRGB color space, though the LAB color space is desirable for colorimetric evaluation. Unfortunately, the current MSE calculation model cannot handle the LAB color space since the transformation from XYZ to LAB is highly nonlinear. Although we have tried to use

Table 3 RMSE comparison for the hyperspectral image dataset in the case of RGB-NIR imaging under the white illumination.

	Mean			Median			Max		
σ	LCC	PCC	TCC	LCC	PCC	TCC	LCC	PCC	TCC
0	15.94	14.55	**14.54**	14.74	12.94	**12.93**	**25.45**	27.48	27.46
2	30.44	32.58	**29.66**	29.83	31.62	**29.41**	36.32	40.57	**35.66**
4	54.12	61.11	**53.40**	53.95	59.45	**53.24**	57.85	71.81	**57.22**
6	78.82	92.69	**78.00**	78.86	90.86	**78.09**	81.88	109.51	**81.16**
8	103.60	127.52	**102.66**	104.29	125.51	**103.24**	106.94	150.99	**106.11**
10	128.30	165.99	**127.22**	129.71	164.76	**128.47**	132.04	194.76	**131.09**

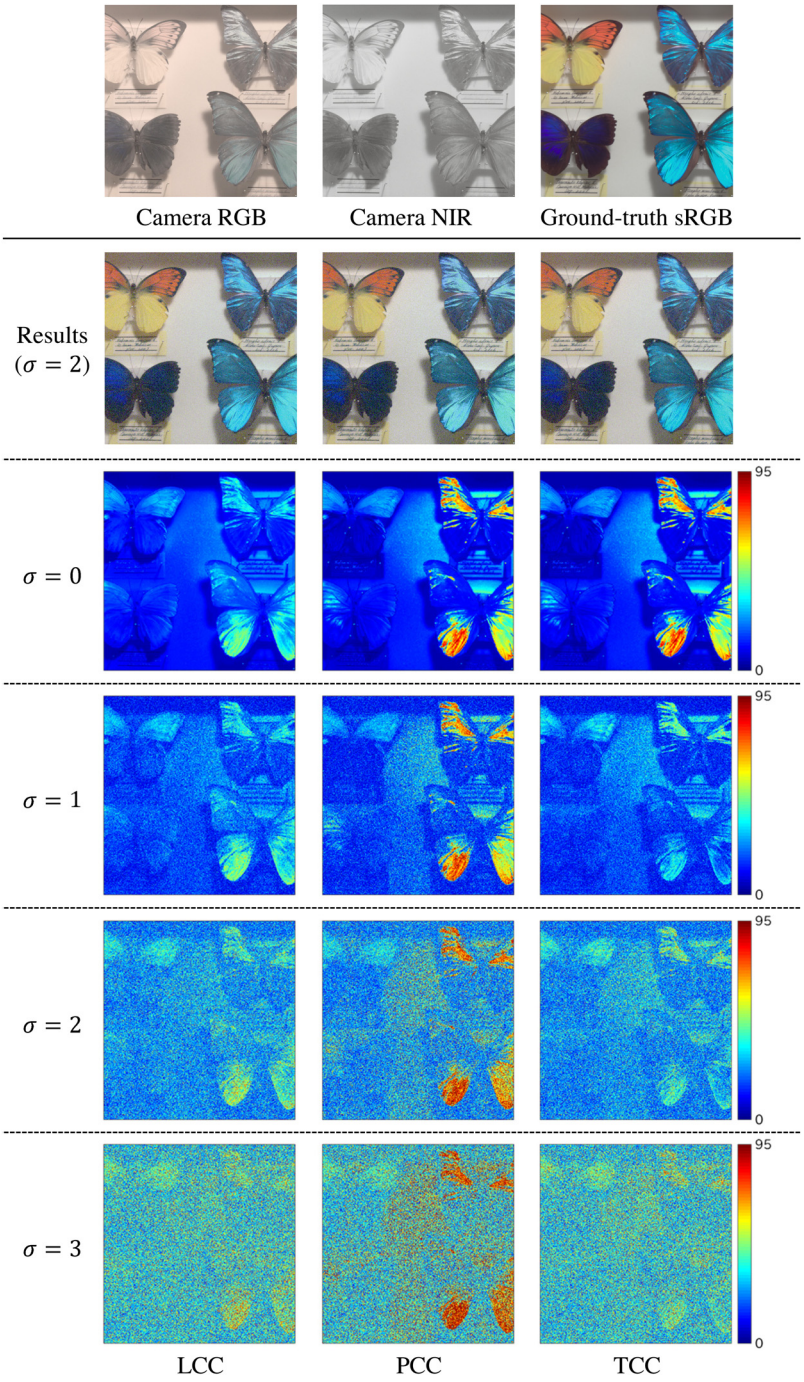

Fig. 6 Color correction results and RMSE maps for the hyperspectral image data in the case of RGB-NIR imaging under the white illumination.

the linearized LAB transformation,[18,41] we found that its approximation error is not trivial for selecting an optimal tuning parameter value in the TCC framework.

The second limitation is that TCC holds the negative sides of PCC such as irradiance dependence[30–32] and hue-plane distortion.[33–35] This cannot be avoided because TCC uses the PCC model in nature. The third limitation is that we currently assume signal-independent Gaussian noise due to its simplicity for the MSE model derivation, whereas the real noise is better modeled by signal-dependent Poisson noise.[19,21] The accuracy of the noise model and the complexity of the MSE calculation model may be the trade-off in the TCC framework. The fourth limitation is that we currently do not consider the mosaic nature of a color filter

array for single-sensor cameras. Our future work is to address this limitation by considering the pseudo-four-channel division manner proposed by Akiyama et al.[61]

6 Conclusion

In this paper, we have proposed the TCC framework that enables us to tune the color correction matrix between the LCC and the PCC models. We also have derived the MSE calculation model of PCC that enables us to select an optimal tuning parameter value to provide the best trade-off balance between LCC and PCC. Experimental results for noisy images have demonstrated that TCC effectively balances the trade-off regarding color fidelity and noise amplification and outperforms LCC, PCC, and other state-of-the-art color correction methods. We also have generalized the TCC framework to multispectral imaging cases and have demonstrated its effectiveness by experimental results.

7 Appendix

Here we describe the detailed calculation for deriving each component of Eqs. (15) and (19), i.e., each component of the expectation vector $\boldsymbol{\mu}_{\mathbf{p}_k}$ and the variance-covariance matrix $\tilde{\boldsymbol{\Sigma}}_{\mathbf{p}_k}$. For this purpose, we use the input polynomial vector representation of Eq. (17). We also introduce a symbolic representation and let $X \in \{R, G, B\}$ be the index set of the first-order linear terms, $XY \in \{RG, RB, GB\}$ be that of the second-order cross terms, and $X^2 \in \{R^2, G^2, B^2\}$ be that of the squared terms. Similarly, we use (X, Y), (XY, XZ), (X^2, Y^2), (X, XY), (X, YZ), (X, X^2), (X, Y^2), (XY, X^2), and (XY, Z^2) to represent the index sets of all possible combination patterns of the linear, cross, and squared terms. The following sections contain the series of calculations to derive all necessary components.

7.1 *Assumptions*

Based on the assumption of the signal-independent zero-mean Gaussian noise, the following equations are defined:

$$E(n_X) = 0, \quad V(n_X) = E(n_X^2) = \sigma_X^2$$
$$C(n_X, n_Y) = 0, \quad C(g_X, n_X) = 0.$$

These equations are used for deriving expectation, variance, and covariance components in the following calculation. In the appendix, we use the mathematical description $f(\cdot)_{\to 0}$ to represent the component that reduces to zero based on the assumption of the noise we made.

7.2 *Expectation of Linear, Cross, and Squares Terms*

The expectation of each linear, cross, and squared term for deriving the expectation vector $\boldsymbol{\mu}_{\mathbf{p}_k}$ of Eq. (15) is calculated as follows:

$$\begin{aligned}
E(p_X &= g_X + n_X) \\
&= E(g_X) + E(n_X)_{\to 0}, \\
&= g_X. \\
E[p_{XY} &= (g_X + n_X)(g_Y + n_Y)] \\
&= E(g_X g_Y) + E(g_X n_Y)_{\to 0} + E(g_Y n_X)_{\to 0} + E(n_X n_Y)_{\to 0}, \\
&= g_X g_Y. \\
E[p_{X^2} &= (g_X + n_X)^2] \\
&= E(g_X^2) + E(2 g_X n_X)_{\to 0} + E(n_X^2), \\
&= g_X^2 + \sigma_X^2.
\end{aligned}$$

7.3 *Variance of Linear, Cross, and Squared Terms*

The variance of each linear, cross, and squared term for deriving the diagonal elements of the variance-covariance matrix $\tilde{\Sigma}_{\mathbf{p}_k}$ of Eq. (19) is calculated as follows:

$$\begin{aligned}
V(p_X &= g_X + n_X) \\
&= V(g_X)_{\to 0} + V(n_X) + 2C(g_X, n_X)_{\to 0}, \\
&= \sigma_X^2. \\
V[p_{XY} &= (g_X + n_X)(g_Y + n_Y)] \\
&= V(g_X g_Y)_{\to 0} + V(g_X n_Y) + V(g_Y n_X) + V(n_X n_Y) \\
&\quad + 2C(g_X g_Y, g_X n_Y)_{\to 0} + 2C(g_X g_Y, g_Y n_X)_{\to 0} \\
&\quad + 2C(g_X g_Y, n_X n_Y)_{\to 0} + 2C(g_X n_Y, g_Y n_X)_{\to 0} \\
&\quad + 2C(g_X n_Y, n_X n_Y)_{\to 0} + 2C(g_Y n_X, n_X n_Y)_{\to 0}, \\
&= g_X^2 \sigma_Y^2 + g_Y^2 \sigma_X^2 + \sigma_X^2 \sigma_Y^2. \\
V[p_{X^2} &= (g_X + n_X)^2] \\
&= V(g_X^2)_{\to 0} + V(2 g_X n_X) + V(n_X^2) \\
&\quad + 2C(g_X^2, 2 g_X n_X)_{\to 0} + 2C(g_X^2, n_X^2)_{\to 0} \\
&\quad + 2C(2 g_X n_X, n_X^2)_{\to 0}, \\
&= 4 g_X^2 \sigma_X^2 + 2\sigma_X^4,
\end{aligned}$$

where we use the following derivation:

$$\begin{aligned}
V(n_X^2) &= E[(n_X^2)^2] - E(n_X^2)E(n_X^2), \\
&= 3\sigma_X^4 - \sigma_X^4, \\
&= 2\sigma_X^4. \\
C(2 g_X n_X, n_X^2) & \\
&= E(2 g_X n_X^3)_{\to 0} - E(2 g_X n_X)_{\to 0} E(n_X^2), \\
&= 0.
\end{aligned}$$

7.4 *Covariance of Linear Terms*

The covariance of linear terms used in the submatrix $\mathbf{C}_{11}$ is calculated as follows:

$$\begin{aligned}
C(p_X, p_Y) &= E(p_X p_Y) - E(p_X)E(p_Y), \\
&= 0,
\end{aligned}$$

where we use the following derivation:

$$\begin{aligned}
E(p_X p_Y &= g_X g_Y + g_X n_Y + g_Y n_X + n_X n_Y) \\
&= E(g_X g_Y) + E(g_X n_Y)_{\to 0} + E(g_Y n_X)_{\to 0} + E(n_X n_Y)_{\to 0}, \\
&= g_X g_Y. \\
E(p_X &= g_X + n_X) E(p_Y = g_Y + n_Y) \\
&= g_X g_Y.
\end{aligned}$$

7.5 *Covariance of Cross Terms*

The covariance of cross terms used in the submatrix $\mathbf{C}_{22}$ is calculated as follows:

$$\begin{aligned} C(p_{XY}, p_{XZ}) &= E(p_{XY}p_{XZ}) - E(p_{XY})E(p_{XZ}), \\ &= g_Y g_Z \sigma_X^2, \end{aligned}$$

where we use the following derivation:

$$\begin{aligned} E(p_{XY}p_{XZ}) &= E(g_X^2 g_Y g_Z) + E(g_X^2 g_Z n_Y)_{\to 0} \\ &\quad + E(2g_X g_Y g_Z n_X)_{\to 0} + E(2g_X g_Z n_X n_Y)_{\to 0} \\ &\quad + E(g_Y g_Z n_X^2) + E(g_Z n_X^2 n_Y)_{\to 0} \\ &\quad + E(g_X^2 g_Y n_Z)_{\to 0} + E(g_X^2 n_Y n_Z)_{\to 0} \\ &\quad + E(2g_X g_Y n_X n_Z)_{\to 0} + E(2g_X n_X n_Y n_Z)_{\to 0} \\ &\quad + E(g_X n_X^2 n_Z)_{\to 0} + E(n_X^2 n_Y n_Z)_{\to 0}, \\ &= g_X^2 g_Y g_Z + g_Y g_Z \sigma_X^2. \\ E(p_{XY})E(p_{XZ}) &= g_X g_Y \cdot g_X g_Z, \\ &= g_X^2 g_Y g_Z. \end{aligned}$$

7.6 *Covariance of Squared Terms*

The covariance of squared terms used in the submatrix $\mathbf{C}_{33}$ is calculated as follows:

$$\begin{aligned} C(p_{X^2}, p_{Y^2}) &= E(p_{X^2}p_{Y^2}) - E(p_{X^2})E(p_{Y^2}), \\ &= 0, \end{aligned}$$

where we use the following derivation:

$$\begin{aligned} E(p_{X^2}p_{Y^2}) &= E(g_X^2 g_Y^2) + E(2g_X^2 g_Y n_Y)_{\to 0} + E(g_X^2 n_Y^2) \\ &\quad + E(2g_X g_Y^2 n_X)_{\to 0} + E(4g_X g_Y n_X n_Y)_{\to 0} \\ &\quad + E(2g_X n_X n_Y^2)_{\to 0} + E(g_Y^2 n_X^2) \\ &\quad + E(2g_Y n_X^2 n_Y)_{\to 0} + E(n_X^2 n_Y^2), \\ &= g_X^2 g_Y^2 + g_X^2 \sigma_Y^2 + g_Y^2 \sigma_X^2 + \sigma_X^2 \sigma_Y^2. \\ E(p_{X^2})E(p_{Y^2}) &= (g_X^2 + \sigma_X^2)(g_Y^2 + \sigma_Y^2), \\ &= g_X^2 g_Y^2 + g_X^2 \sigma_Y^2 + g_Y^2 \sigma_X^2 + \sigma_X^2 \sigma_Y^2. \end{aligned}$$

7.7 *Covariance Between Linear and Cross Terms*

The covariance between linear and cross terms used in the submatrices $\mathbf{C}_{12}$ and $\mathbf{C}_{21}$ is calculated as follows:

$$\begin{aligned} C(p_X, p_{XY}) &= E(p_X p_{XY}) - E(p_X)E(p_{XY}), \\ &= g_Y \sigma_X^2, \end{aligned}$$

where we use the following derivation:

$$\begin{aligned}
E(p_X p_{XY}) &= E(g_X^2 g_Y) + E(g_X^2 n_Y)_{\to 0} + E(2 g_X g_Y n_X)_{\to 0} \\
&\quad + E(2 g_X n_X n_Y)_{\to 0} + E(g_Y n_X^2) + E(n_X^2 n_Y)_{\to 0}, \\
&= g_X^2 g_Y + g_Y \sigma_X^2. \\
E(p_X) E(p_{XY}) &= g_X \cdot (g_X g_Y), \\
&= g_X^2 g_Y.
\end{aligned}$$

In the case that the channel is not shared, the covariance is calculated as follows:

$$\begin{aligned}
C(p_X, p_{YZ}) &= E(p_X p_{YZ}) - E(p_X) E(p_{YZ}), \\
&= 0 .
\end{aligned}$$

where we use the following derivation:

$$\begin{aligned}
E(p_X p_{YZ}) &= E(g_X g_Y g_Z) + E(g_X g_Y n_Z)_{\to 0} + E(g_X g_Z n_Y)_{\to 0} \\
&\quad + E(g_X n_Y n_Z)_{\to 0} + E(g_Y g_Z n_X)_{\to 0} + E(g_Y n_X n_Z)_{\to 0} \\
&\quad + E(g_Z n_X n_Y)_{\to 0} + E(n_X n_Y n_Z)_{\to 0}, \\
&= g_X g_Y g_Z. \\
E(p_X) E(p_{YZ}) &= g_X \cdot (g_Y g_Z), \\
&= g_X g_Y g_Z.
\end{aligned}$$

7.8 *Covariance Between Linear and Squared Terms*

The covariance between linear and squared terms used in the submatrices $\mathbf{C}_{13}$ and $\mathbf{C}_{31}$ is calculated as follows:

$$\begin{aligned}
C(p_X, p_{X^2}) &= E(p_X p_{X^2}) - E(p_X) E(p_{X^2}), \\
&= 2 g_X \sigma_X^2,
\end{aligned}$$

where we use the following derivation:

$$\begin{aligned}
E(p_X p_{X^2}) &= E(g_X^3) + E(3 g_X^2 n_X)_{\to 0} + E(3 g_X n_X^2) + E(n_X^3)_{\to 0}, \\
&= g_X^3 + 3 g_X \sigma_X^2 . E(p_X) E(p_{X^2}) \\
&= g_X (g_X^2 + \sigma_X^2), \\
&= g_X^3 + g_X \sigma_X^2.
\end{aligned}$$

In the case that the channel is not shared, the covariance is calculated as follows:

$$\begin{aligned}
C(p_X, p_{Y^2}) &= E(p_X p_{Y^2}) - E(p_X) E(p_{Y^2}), \\
&= 0,
\end{aligned}$$

where we use the following derivation:

$$\begin{aligned}
E(p_X p_{Y^2}) &= E(g_X g_Y^2) + E(2 g_X g_Y n_Y)_{\to 0} + E(g_X n_Y^2) \\
&\quad + E(g_Y^2 n_X)_{\to 0} + E(2 g_Y n_X n_Y)_{\to 0} + E(n_X n_Y^2)_{\to 0}, \\
&= g_X g_Y^2 + g_X \sigma_Y^2 . E(p_X) E(p_{Y^2}) \\
&= g_X (g_Y^2 + \sigma_Y^2), \\
&= g_X g_Y^2 + g_X \sigma_Y^2.
\end{aligned}$$

7.9 *Covariance Between Cross and Squared Terms*

The covariance between cross and squared terms used in the submatrices $\mathbf{C}_{23}$ and $\mathbf{C}_{32}$ is calculated as follows:

$$\begin{aligned} C(p_{XY}, p_{X^2}) &= E(p_{XY}p_{X^2}) - E(p_{XY})E(p_{X^2}), \\ &= 2g_X g_Y \sigma_X^2, \end{aligned}$$

where we use the following derivation:

$$\begin{aligned} E(p_{XY}p_{X^2}) &= E(g_X^3 g_Y) + E(g_X^3 n_Y)_{\to 0} + E(3g_X^2 g_Y n_X)_{\to 0} \\ &\quad + E(3g_X^2 n_X n_Y^2)_{\to 0} + E(3g_X g_Y n_X^2) \\ &\quad + E(3g_X n_X^2 n_Y)_{\to 0} + E(g_Y n_X^3)_{\to 0} \\ &\quad + E(n_X^3 n_Y)_{\to 0}, \\ &= g_X^3 g_Y + 3g_X g_Y \sigma_X^2. \\ E(p_{XY})E(p_{X^2}) &= g_X g_Y (g_X^2 + \sigma_X^2), \\ &= g_X^3 g_Y + g_X g_Y \sigma_X^2. \end{aligned}$$

In the case that the channel is not shared, the covariance is calculated as follows:

$$\begin{aligned} C(p_{XY}, p_{Z^2}) &= E(p_{XY}p_{Z^2}) - E(p_{XY})E(p_{Z^2}), \\ &= 0, \end{aligned}$$

where we use the following derivation:

$$\begin{aligned} E(p_{XY}p_{Z^2}) &= E(g_X g_Y g_Z^2) + E(g_X g_Z^2 n_Y)_{\to 0} + E(g_Y g_Z^2 n_X)_{\to 0} \\ &\quad + E(g_Z^2 n_X n_Y)_{\to 0} + E(2g_X g_Y g_Z n_Z)_{\to 0} \\ &\quad + E(2g_X g_Z n_Y n_Z)_{\to 0} + E(2g_Y g_Z n_X n_Z)_{\to 0} \\ &\quad + E(2g_Z n_X n_Y n_Z)_{\to 0} + E(g_X g_Y n_Z^2) \\ &\quad + E(g_X n_Y n_Z^2)_{\to 0} + E(g_Y n_X n_Z^2)_{\to 0} + E(n_X n_Y n_Z^2)_{\to 0}, \\ &= g_X g_Y g_Z^2 + g_X g_Y \sigma_Z^2. \\ E(p_{XY})E(p_{Z^2}) &= g_X g_Y (g_Z^2 + \sigma_Z^2), \\ &= g_X g_Y g_Z^2 + g_X g_Y \sigma_Z^2. \end{aligned}$$

Acknowledgments

This work was partly supported by the MIC/SCOPE #141203024.

References

1. M. Anderson et al., "Proposal for a standard default color space for the internet—sRGB," in *Proc. Color and Imaging Conf.*, pp. 238–245 (1996).
2. H. R. Kang, *Computational Color Technology*, SPIE Press, Bellingham, Washington (2006).
3. D. Marimont and B. Wandell, "Linear models of surface and illuminant spectra," *J. Opt. Soc. Am. A* **9**(11), 1905–1913 (1992).
4. P. M. Hubel et al., "Matrix calculations for digital photography," in *Proc. Color and Imaging Conf.*, pp. 105–111 (1997).
5. G. Hong, M. R. Luo, and P. A. Rhodes, "A study of digital camera colorimetric characterisation based on polynomial modelling," *Color Res. Appl.* **26**(1), 76–84 (2001).

6. G. D. Finlayson, M. Mackiewicz, and A. Hurlbert, "Colour correction using root-polynomial regression," *IEEE Trans. Image Process.* **24**(5), 1460–1470 (2015).
7. P. C. Hung, "Colorimetric calibration in electronic imaging devices using a look-up-table model and interpolations," *J. Electron. Imaging* **2**(1), 53–61 (1993).
8. J. M. Kasson et al., "Performing color space conversions with three-dimensional linear interpolation," *J. Electron. Imaging* **4**(3), 226–250 (1995).
9. R. Bala et al., "Two-dimensional transforms for device color correction and calibration," *IEEE Trans. Image Process.* **14**(8), 1172–1186 (2005).
10. V. Monga, R. Bala, and X. Mo, "Design and optimization of color lookup tables on a simplex topology," *IEEE Trans. Image Process.* **21**(4), 1981–1996 (2012).
11. E. Garcia, R. Arora, and M. R. Gupta, "Optimized regression for efficient function evaluation," *IEEE Trans. Image Process.* **21**(9), 4128–4140 (2012).
12. J. S. McElvain and W. Gish, "Camera color correction using two-dimensional transforms," *Proc. Color and Imaging Conf.*, pp. 250–256 (2013).
13. H. R. Kang and P. G. Anderson, "Neural network applications to the color scanner and printer calibrations," *J. Electron. Imaging* **1**(2), 125–135 (1992).
14. M. J. Vrhel and H. J. Trussell, "Color scanner calibration via a neural network," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, pp. 3465–3468 (1999).
15. V. Cheung and S. Westland, "Color camera characterisation using artificial neural networks," in *Proc. Color and Imaging Conf.*, pp. 117–120 (2002).
16. V. Cheung et al., "A comparative study of the characterisation of colour cameras by means of neural networks and polynomial transforms," *Color. Technol.* **120**(1), 19–25 (2004).
17. Y. P. Tan and T. Acharya, "Method for color correction with noise consideration," *Proc. SPIE* **3963**, 329–337 (2000).
18. U. Barnhöfer et al., "Color estimation error trade-offs," *Proc. SPIE* **5017**, 263–273 (2003).
19. S. Quan, "Analytical approach to the optimal linear matrix with comprehensive error metric," *Proc. SPIE* **5292**, 243–253 (2004).
20. S. Lim and A. Silverstein, "Spatially varying color correction (SVCC) matrices for reduced noise," in *Proc. Color and Imaging Conf.*, pp. 76–81 (2004).
21. H. J. Trussell and M. J. Vrhel, "Color estimation under Poisson noise," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process.*, pp. 1914–1918 (2013).
22. P. D. Burns and R. S. Berns, "Error propagation analysis in color measurement and imaging," *Color Res. Appl.* **22**(4), 280–289 (1997).
23. R. Yamakabe et al., "Tunable color correction between linear and polynomial models for noisy images," in *Proc. IEEE Int. Conf. Image Process.*, pp. 3125–3129 (2017).
24. G. D. Finlayson and M. S. Drew, "Constrained least-squares regression in color spaces," *J. Electron. Imaging* **6**(4), 484–493 (1997).
25. S. Bianco et al., "Color space transformations for digital photography exploiting information about the illuminant estimation process," *J. Opt. Soc. Am. A* **29**(3), 374–384 (2012).
26. S. Bianco et al., "Color correction pipeline optimization for digital cameras," *J. Electron. Imaging* **22**(2), 023014 (2013).
27. S. Bianco and R. Schettini, "Error-tolerant color rendering for digital cameras," *J. Math. Imaging Vision* **50**(3), 235–245 (2014).
28. H. Zhang and H. Liu, "Hue constrained matrix optimization for preferred color reproduction," *J. Electron. Imaging* **21**(3), 033021 (2012).
29. J. Vazquez-Corral, D. Connah, and M. Bertalmío, "Perceptual color characterization of cameras," *Sensors* **14**(12), 23205–23229 (2014).
30. B. Funt and P. Bastani, "Irradiance-independent camera color calibration," *Color Res. Appl.* **39**(6), 540–548 (2014).
31. G. D. Finlayson, M. M. Darrodi, and M. Mackiewicz, "The alternating least squares technique for nonuniform intensity color correction," *Color Res. Appl.* **40**(3), 232–242 (2015).
32. G. D. Finlayson, H. Gong, and R. B. Fisher, "Color homography: theory and applications," *IEEE Trans. Pattern Anal. Mach. Intell.* **41**(1), 20–33 (2019).
33. C. F. Andersen and J. Y. Hardeberg, "Colorimetric characterization of digital cameras preserving hue planes," in *Proc. Color and Imaging Conf.*, pp. 141–146 (2005).

34. C. F. Andersen and D. Connah, "Weighted constrained hue-plane preserving camera characterization," *IEEE Trans. Image Process.* **25**(9), 4329–4339 (2016).
35. M. Mackiewicz, C. F. Andersen, and G. D. Finlayson, "Method for hue plane preserving color correction," *J. Opt. Soc. Am. A* **33**(11), 2166–2177 (2016).
36. G. D. Finlayson and M. S. Drew, "The maximum ignorance assumption with positivity," in *Proc. Color and Imaging Conf.*, pp. 202–205 (1996).
37. M. J. Vrhel and H. J. Trussell, "Color correction using principal components," *Color Res. Appl.* **17**(5), 328–338 (1992).
38. S. Bianco et al., "Polynomial modeling and optimization for colorimetric characterization of scanners," *J. Electron. Imaging* **17**(4), 043002 (2008).
39. S. Bianco, R. Schettini, and L. Vanneschi, "Empirical modeling for colorimetric characterization of digital cameras," in *Proc. IEEE Int. Conf. Image Process.*, pp. 3469–3472 (2009).
40. M. J. Vrhel and H. J. Trussell, "Optimal color filters in the presence of noise," *IEEE Trans. Image Process.* **4**(6), 814–823 (1995).
41. G. Sharma and H. J. Trussell, "Figures of merit for color scanners," *IEEE Trans. Image Process.* **6**(7), 990–1001 (1997).
42. P. Vora and C. Herley, "Trade-offs between color saturation and noise sensitivity in image sensors," in *Proc. IEEE Int. Conf. Image Process.*, pp. 196–200 (1998).
43. N. Shimano, "Suppression of noise effects in color correction by spectral sensitivities of image sensors," *Opt. Rev.* **9**(2), 81–88 (2002).
44. H. Kuniba and R. S. Berns, "Spectral sensitivity optimization of color image sensors considering photon shot noise," *J. Electron. Imaging* **18**(2), 023002 (2009).
45. X. Zhang and D. H. Brainard, "Bayesian color correction method for non-colorimetric digital image sensors," in *Proc. Color and Imaging Conf.*, pp. 308–314 (2004).
46. S. Bianco et al., "A new method for RGB to XYZ transformation based on pattern search optimization," *IEEE Trans. Consum. Electron.* **53**(3), 1020–1028 (2007).
47. L. Kharitonenko, S. Twelves, and C. Weerasinghe, "Suppression of noise amplification during colour correction," *IEEE Trans. Consum. Electron.* **48**(2), 229–233 (2002).
48. C. Weerasinghe et al., "Novel color processing architecture for digital cameras with CMOS image sensors," *IEEE Trans. Consum. Electron.* **51**(4), 1092–1098 (2005).
49. I. Kharitonenko and W. Li, "Image color correction in DCT domain," in *Proc. IEEE Int. Conf. Consum. Electron.*, pp. 245–248 (2013).
50. K. Takahashi et al., "Effective color correction pipeline for a noisy image," *Proc. IEEE Int. Conf. Image Process.*, pp. 4002–4006 (2016).
51. A. Foi et al., "Practical Poissonian–Gaussian noise modeling and fitting for single-image raw-data," *IEEE Trans. Image Process.* **17**(10), 1737–1754 (2008).
52. "Munsell color science laboratory: usefull color data," https://www.rit.edu/cos/colorscience/rc_useful_data.php.
53. Y. Monno et al., "Single-sensor RGB-NIR imaging: high-quality system design and prototype implementation," *IEEE Sens. J.* **19**(2), 497–507 (2019).
54. J. Jiang et al., "What is the space of spectral sensitivity functions for digital color cameras?" in *Proc. Workshop Appl. Comput. Vision*, pp. 168–179 (2013).
55. S. Geman, E. Bienenstock, and R. Doursat, "Neural networks and the bias/variance dilemma," *Neural Comput.* **4**(1), 1–58 (1992).
56. Y. Monno et al., "A practical one-shot multispectral imaging system using a single image sensor," *IEEE Trans. Image Process.* **24**(10), 3048–3059 (2015).
57. J. B. Thomas et al., "Spectral characterization of a prototype SFA camera for joint visible and NIR acquisition," *Sensors* **16**(7), 993 (2016).
58. Y. Monno, M. Tanaka, and M. Okutomi, "N-to-sRGB mapping for single-sensor multispectral imaging," in *Proc. Color and Photometry Comput. Vision Workshop*, pp. 66–73 (2015).
59. X. Liu, M. Tanaka, and M. Okutomi, "Single-image noise level estimation for blind denoising," *IEEE Trans. Image Process.* **22**(12), 5226–5237 (2013).
60. F. Fang et al., "Colour correction toolbox," in *Proc. 13th AIC Cong.*, pp. 13–18 (2017).
61. H. Akiyama, M. Tanaka, and M. Okutomi, "Pseudo four-channel image denoising for noisy CFA raw data," in *Proc. IEEE Int. Conf. Image Process.*, pp. 4778–4782 (2015).

Ryo Yamakabe received his BE and ME degrees from Tokyo Institute of Technology, Tokyo, Japan, in 2017 and 2019, respectively. He is currently a researcher at NEC Corporation, Kanagawa, Japan. His research interests are in imaging processing and computational imaging.

Yusuke Monno received his BE and ME degrees and his PhD from Tokyo Institute of Technology, Tokyo, Japan, in 2010, 2011, and 2014, respectively. He is currently a postdoctoral researcher in the Department of Systems and Control Engineering of the School of Engineering at Tokyo Institute of Technology. From November 2013 to March 2014, he joined the Image and Visual Representation Group at École Polytechnique Fédérale de Lausanne as a research internship student. His research interests are in both theoretical and practical aspects of image processing, computer vision, and biomedical engineering. He received the NF Foundation R&D Encouragement Award in 2015, the Inoue Research Award for Young Scientists and the Seiichi Tejima Doctoral Dissertation Award in 2016, the IEEE Signal Processing Society Japan Best Paper Award in 2017, and the Telecom System Technology Award in 2020.

Masayuki Tanaka received his bachelor's and master's degrees in control engineering and his PhD from Tokyo Institute of Technology in 1998, 2000, and 2003, respectively. He was a software engineer at Agilent Technology from 2003 to 2004. He was a research scientist at Tokyo Institute of Technology from 2004 to 2008. He was an associate professor at the Graduate School of Science and Engineering, Tokyo Institute of Technology, from 2008 to 2016. He was a visiting scholar in the Department of Psychology at Stanford University from 2013 to 2014. He was an associate professor at the School of Engineering of Tokyo Institute of Technology from 2016 to 2017. He was a senior researcher at National Institute of Advanced Industrial Science and Technology from 2017 to 2020. Since 2020, he has been an associate professor in the School of Engineering, Tokyo Institute of Technology.

Masatoshi Okutomi received his BEng degree from the Department of Mathematical Engineering and Information Physics, University of Tokyo, Tokyo, Japan, in 1981 and his MEng degree from the Department of Control Engineering, Tokyo Institute of Technology, Tokyo, in 1983. He joined the Canon Research Center, Canon Inc., Tokyo, in 1983. From 1987 to 1990, he was a visiting research scientist at the School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA. He received his Dr. Eng. degree from the Tokyo Institute of Technology, in 1993, for his research on stereo vision. Since 1994, he has been at the Tokyo Institute of Technology, where he is currently a professor in the Department of Systems and Control Engineering of the School of Engineering.