Adaptive characterization method for desktop color printers

Hui-Liang Shen
Zhi-Huan Zheng
Chong-Chao Jin
Xin Du
Zhejiang University
Department of Information and Electronic Engineering
Hangzhou 310027, China
E-mail: shenhl@zju.edu.cn

Si-Jie Shao
John H. Xin
The Hong Kong Polytechnic University
Institute of Textiles and Clothing, Hong Kong

Abstract. *With the rapid development of multispectral imaging technique, it is desired that the spectral color can be accurately reproduced using desktop color printers. However, due to the specific spectral gamuts determined by printer inks, it is almost impossible to exactly replicate the reflectance spectra in other media. In addition, as ink densities can not be individually controlled, desktop printers can only be regarded as red-green-blue devices, making physical models unfeasible. We propose a locally adaptive method, which consists of both forward and inverse models, for desktop printer characterization. In the forward model, we establish the adaptive transform between control values and reflectance spectrum on individual cellular subsets by using weighted polynomial regression. In the inverse model, we first determine the candidate space of the control values based on global inverse regression and then compute the optimal control values by minimizing the color difference between the actual spectrum and the predicted spectrum via forward transform. Experimental results show that the proposed method can reproduce colors accurately for different media under multiple illuminants.* © 2013 SPIE and IS&T [DOI: 10.1117/1.JEI.22.2.023012]

1 Introduction

Faithful color acquisition and reproduction, as the basis of color quality control, is a critical demand in the textile, printing, and other industries. Multispectral imaging, for its ability in accurate reconstruction of spectral spectrum, has been extensively studied in recent years.[1–3] With the acquired reflectance spectrum, the colors of the object surfaces can be reproduced with high fidelity on output devices, provided that the devices are properly characterized. For a color printer, which is a typical output device, the characterization consists of forward and inverse models.[4,5] The forward model seeks to convert the device-dependent control values to the device-independent reflectance spectrum or colorimetric stimulus.[6] The inverse model, which is more useful in color applications, transforms from a requested reflectance spectrum to the optimal control values that, when printed, could reproduce that spectrum or colorimetric stimulus with greatest possible accuracy.[7]

In the color imaging field, much work has been conducted to characterize color printers using physics models, assuming ink intensities are individually controllable.[5] However, this assumption does not hold for the widely available desktop printers with solely red-green-blue (RGB) operation interface. In fact, the embedded conversion between RGB values and cyan-magenta-yellow-black (CMYK) is unknown to users. To deal with this problem, we propose a unified framework, which consists of both forward and inverse models, to characterize desktop printers.

1.1 *Background*

In spectral characterization, the Yule-Nielsen modified spectral Neugebauer (YNSN) model[8] is particularly effective, which, despite its simplicity, could produce high accuracy in color prediction.[5] Compared with the original Neugebauer model, the YNSN model uses an additional factor, namely n-factor, such that the scattering of light within the paper substrate can be account for. In the YNSN model, the spectrum of a four-ink halftone print is the weighted summation of $16(=2^4)$ different colors referred as Neugebauer primaries, given by all the possible overprints of inks.[5] For CMYKcm (CMYK plus dilute cyan and magenta) printers, the characterization becomes more difficult, as there is no unique mapping from the four-dimensional (4-D) CMYK space to the six-dimensional (6-D)CMYKcm space without additional constraints.[9] Note that the YNSN model is based on the assumption that the ink densities of the printer can be individually controlled.

The YNSN parameters, such as n-factor and dot-gain factor, can be estimated using genetic algorithms[6] or total least squares.[10] The recent work shows that, by deducing ink spreading curves and imposing additional constraints, the

Paper 12169 received May 7, 2012; revised manuscript received Oct. 15, 2012; accepted for publication Apr. 4, 2013; published online May 10, 2013; corrected date May 22, 2013.

spectral characterization can be conducted using image tiles within the the printed color images, without printing special calibration patches.[11] To improve spectrum prediction, the 4-D CMYK input space can be divided into a grid of cells, and then the YNSN model can be applied in individual cells instead of the whole space.[12]

In addition to the mentioned forward transforms, the YNSN model can also be employed in the inverse transform, namely color separation. However, as the spectral characteristics of print colors are determined by the inks accompanied, in general, color printers cannot exactly reproduce reflectance spectra of other media. Such an example is shown in Fig. 1. The print spectra exhibit a particular distribution in wavelength from 600 to 700 nm, which is totally distinct to the highlighted spectrum of ColorChecker SG color. Therefore, in previous work, the target colors are spectrally mapped to the printer gamut before color separation.[13,14] However, as spectral gamut mapping results in changed spectra, it is unfeasible if the requirement is to exactly reproduce color appearances under various illuminants.

Colorimetric characterization provides an alternative solution. In this category, a multidimensional lookup table (LUT) is probably the most straightforward and frequently adopted technique.[15,16] While this technique does not require knowledge of device physics, its application is largely affected by disadvantages such as large measurement number and multidimensional interpolation.[17] In this regard, optimal sampling is necessary to reduce LUT size and improve implementation efficiency.[18] In additional to LUT, polynomial regression[19,20] and neural network[21] have been employed in printer characterization. Local linear regression using enclosing neighborhood[22] and constrained least-squares regression[23] have also showed their effectiveness in colorimetric characterization. We note that, however, in most previous work, the colorimetric characterization is conducted under a single illuminantand thus cannot reproduce faithful colors under multiple illuminants.

1.2 *Motivation of this Work*

Our motivation is to characterize desktop printers such that accurate color appearances of multispectral scenes can be reproduced under multiple illuminants. This requirement implies that the original reflectance spectra should not be spectrally mapped to the printer gamut. In addition, as the ink densities cannot be controlled individually and independently, the YNSN model or its modified versions are inappropriate to desktop printers. We propose a unified framework that establishes the locally adaptive forward and inverse models for desktop printers. The objective is to obtain the lowest colorimetric errors under multiple illuminants, while the resultant print spectra are not necessarily the closest replica of the originals. Experiments have been conducted to evaluate the proposed method.

1.3 *Paper Organization*

The rest of the paper is organized as follows. Section 2 overviews the proposed framework of printer characterization, and Sec. 3 introduces the forward model that transforms RGB values to reflectance spectrum. The inverse model, which incorporates global regression and numerical optimization, is elaborated in Sec. 4. Section 5 presents experimental results and relevant discussion, and finally, Sec. 6 concludes this paper.

2 Overview of the Proposed Method

The forward and inverse models of the printer characterization framework is shown in Fig. 2. As illustrated, the control inputs of the desktop printers are in RGB format, represented in the integer range [0,255]. The device-independent color measurement is represented by either reflectance spectrum or the corresponding CIEXYZ values under various illuminants, including D65, A, F11, and others if needed.

In the forward model, we employ polynomial regression to model the transform between RGB inputs and reflectance spectrum. Considering the nonlinear nature of printing process, this transform is computed in individual cellular subsetsin order to achieve high accuracy. When predicting, we first locate the cell surrounding the given RGB valuesand then use the corresponding transform to compute the reflectance spectrum. The details of the forward model will be presented in Sec. 3.

As mentioned in Sec. 1, due to the spectral characteristics of the print colors, it is impossible to exactly reproduce the reflectance spectra of other media. The compromise adopted in this work is that the reproduced stimulus values should closely match the original ones under various illuminants. Accordingly, in the inverse model, we first establish the global transforms between CIEXYZ and RGB values for individual illuminants using high order polynomial regression. Then, for a given reflectance spectrum, we obtain a RGB candidate under each illuminant. As can be expected, each RGB candidate is the best to the corresponding

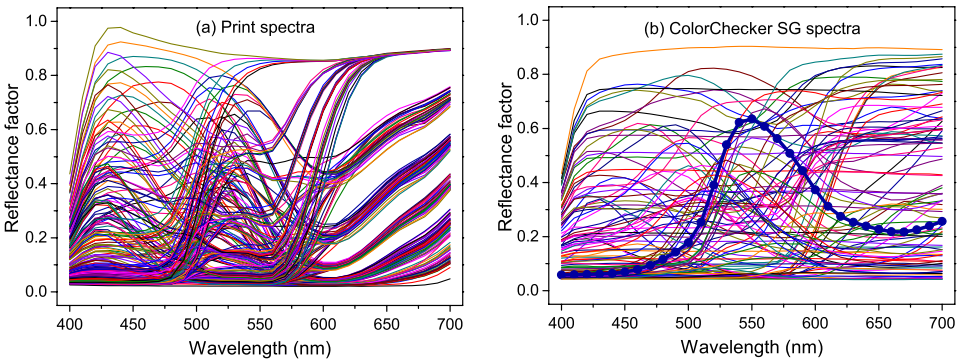

Fig. 1 The reflectance spectra of (a) print colors and (b) ColorChecker SG colors. Due to the quite different spectral characteristics, the highlighted SG spectrum (in blue) could not be accurately reproduced by the printer.

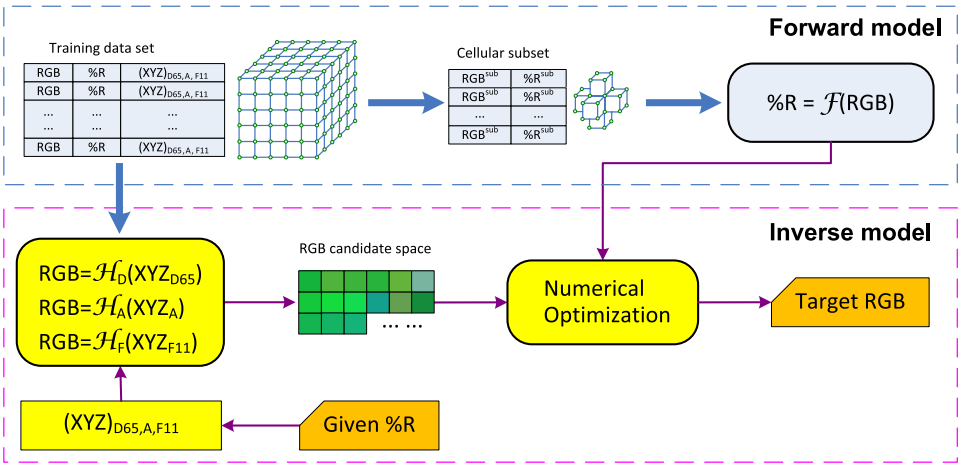

Fig. 2 The unified framework of the proposed printer characterization method. Note that %R denotes reflectance spectrum. In the forward model, the adaptive transforms are defined on individual cellular subsets. In the inverse model, the RGB candidate space is first determined based on global regression under individual illuminants, and then the optimal RGB values are computed by using the forward model and numerical optimization. See text for details.

illuminant, but is usually not optimal to all illuminants. Nevertheless, it is reasonable to assume that the optimal RGB values should be in or close to the space determined by these RGB candidates. In this regard, we deploy numerical optimization to compute the optimal target RGB values by minimizing the color difference between the actual and predicted spectra via the forward model, as will be elaborated in Sec. 4.

3 Forward Model

The three-dimensional (3-D) RGB cubic space is equally divided into $K \times K \times K$ cells, each containing a set of eight lattice nodes. These nodes are referred as supporting points in the following. We additionally include the lattice nodes of the sixneighboring cellsand refer to these nodes as neighboring points. By grouping the supporting and neighboring points, we construct a cellular subset for each cell, based on which the adaptive transform is computed. The adoption of neighboring points can benefit the polynomial regression in increasing generality and avoiding over-fitting.

Determined by locations, there are four types of cellular subsets in the whole RGB cubic grid if not considering orientation, as shown in Fig. 3. The full cellular subset locates inside the RGB cubic, while the corner, edge, and plane cellular subsets reside at corresponding cubic boundaries. As can be seen, the number of supporting points is eight for each subset type, and the number of neighboring points are 12, 16, 20, and 24 for the corner, edge, plane, and full cellular subsets.

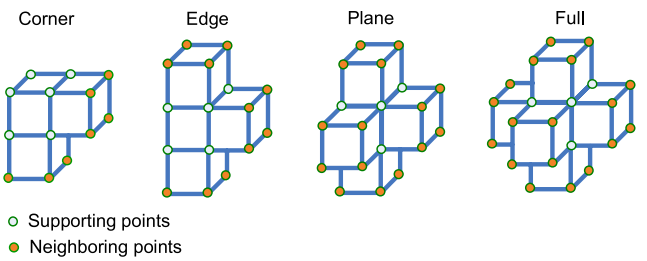

Fig. 3 Four types of cellular subsets. The number of supporting points is eight for any cellular subsets. The numbers of neighboring points are 12, 16, 20, and 24 for the corner, edge, plane, and full cellular subsets.

In the following, we use $\mathbf{u} = (R, G, B)^\top$ to denote the control vector (also referred as RGB vector hereinafter) fed to the printer, and $\mathbf{r} \in \mathbb{R}^{N\times 1}$, where $N = 31$, to denote the spectrum vector. The tristimulus value vector is represented by $\mathbf{q} = (X, Y, Z)^\top$.

For the cellular subset, the transform between RGB inputs and reflectance spectrum is nonlinear, and the polynomial regression is employed in the forward model. As each subset contains at most 32 points, the third-order polynomial regression will become over-fitting, and hence we choose the second-order one. The polynomial extended RGB vector, $\tilde{\mathbf{u}} \in \mathbb{R}^{P_1\times 1}$, where $P_1 = 10$, is

$$\tilde{\mathbf{u}} = (R, G, B, RG, RB, GB, R^2, G^2, B^2, 1)^\top. \tag{1}$$

Suppose the number of neighboring points for a given cellular subset is L. Note that the value of L is different for the four types of cellular subset illustrated in Fig. 3. We further denote the extended polynomial RGB vectors for supporting and neighboring points as $\tilde{\mathbf{u}}_i^a$ and $\tilde{\mathbf{u}}_j^b$, respectively. For polynomial modeling, these extended vectors are assembled in a RGB matrix, $\tilde{\mathbf{U}} \in \mathbb{R}^{(L+8)\times P_1}$, as

$$\tilde{\mathbf{U}} = (\tilde{\mathbf{u}}_1^a, \ldots, \tilde{\mathbf{u}}_8^a, \tilde{\mathbf{u}}_1^b, \ldots, \tilde{\mathbf{u}}_L^b)^\top, \tag{2}$$

and accordingly, the spectra matrix, $\mathbf{R} \in \mathbb{R}^{N\times P_1}$, of the cellular subset is written as

$$\mathbf{R} = (\mathbf{r}_1^a, \ldots, \mathbf{r}_8^a, \mathbf{r}_1^b \ldots, \mathbf{r}_L^b)^\top, \tag{3}$$

where $\mathbf{r}_i^a$ and $\mathbf{r}_j^b$ represent the spectrum vectors of supporting and neighboring points, respectively.

As the supporting and neighboring points locate in different positions, it is reasonable to assume that their contributions to the forward model are also different. In this regard, we use the weighted least-squares, in which the neighboring points are less weighted than the supporting points. The weight matrix, $\mathbf{W} \in \mathbb{R}^{(L+8)\times(L+8)}$, is defined as

$$\mathbf{W} = \begin{pmatrix} \mathbf{I}_8 & \mathbf{0} \\ \mathbf{0} & w\mathbf{I}_L \end{pmatrix}, \tag{4}$$

where $\mathbf{I}_8 \in \mathbb{R}^{8\times 8}$ and $\mathbf{I}_L \in \mathbb{R}^{L\times L}$ denote identity matrices with respective sizes. Our investigation indicates that the characterized color accuracy is not sensitive to w when $w \geq 0.1$, and we adopt $w = 0.1$ in this work.

Inspired by the YNSN model, we further use the n-factor to account for the nonlinear light scattering effect within the paper substrate. Then the forward transform matrix, $\mathbf{F} \in \mathbb{R}^{P_1\times N}$, should minimize the following cost function

$$\|\mathbf{W}\mathbf{R}^{1/n} - \mathbf{W}\tilde{\mathbf{U}}\mathbf{F}\|^2, \tag{5}$$

where $\|\cdot\|$ denotes Frobenius norm and the exponential $1/n$ operates on element-wise manner. The forward transform $\mathbf{F}$ can be solved as

$$\mathbf{F} = (\tilde{\mathbf{U}}^\top\mathbf{W}^\top\mathbf{W}\tilde{\mathbf{U}})^{-1}\tilde{\mathbf{U}}^\top\mathbf{W}^\top\mathbf{W}\mathbf{R}^{1/n}. \tag{6}$$

With the computed $\mathbf{F}$, the reflectance spectrum, $\tilde{\mathbf{r}} \in \mathbb{R}^{N\times 1}$, can be predicted from the given input vector $\mathbf{u}$ as

$$\hat{\mathbf{r}} \triangleq \mathcal{F}(\mathbf{u}) = (\mathbf{F}^\top\tilde{\mathbf{u}})^n, \tag{7}$$

where the exponential n operates element-wise. The accuracy of spectrum prediction is insensitive to different n values when $n \geq 2$, and in this paper we set $n = 2$.

4 Inverse Model

As mentioned, the spectral characteristic of print colors are determined by the equipped inks, and thus the printer cannot exactly reproduce the reflectance spectra of target colors in other media. Consequently, it is not feasible to establish the inverse transform from the original reflectance spectrum to RGB values. As a compromise, we instead require that the color appearances of the printed samples should closely match those of target samples under various illuminants. Therefore, in the inverse model, we first compute the individual RGB candidates from the CIEXYZ values under individual illuminants using global regression, and then compute the desired RGB values that are optimal to all illuminants by numerical optimization.

4.1 *Regression Under Individual Illuminants*

Unlike the RGB cubic, the nodes in CIEXYZ space are irregularly distributed, making adaptive transform not straightforward in the inverse model. We employ third-order global polynomial regression to compute the RGB candidate from the CIEXYZ values under each illuminant. Our investigation indicates that the third-order polynomial regression can produce high color accuracy and avoid the over-fitting problem at the same time.

Instead of using CIEXYZ values directly, we compute their logarithm counterparts $X_l = \log(X)$, $Y_l = \log(Y)$, and $Z_l = \log(Z)$. As indicated by our investigation and also others,[20,24] the latter exhibits a better linearity with RGB values. We construct the third-order polynomial vector $\tilde{\mathbf{q}} \in \mathbb{R}^{P_2\times 1}$, where $P_2 = 20$, as

$$\tilde{\mathbf{q}} = (X_l, Y_l, Z_l, X_lY_l, X_lZ_l, Y_lZ_l, X_l^2, Y_l^2, Z_l^2, X_l^2Y, X_l^2Z_l, Y_l^2Z_l, Y_l^2X_l, Z_l^2X_l, Z_l^2Y_l, X_lY_lZ_l, X_l^3, Y_l^3, Z_l^3, 1)^\top. \tag{8}$$

Let the number of training samples be M, we assemble the CIEXYZ matrix, $\tilde{\mathbf{Q}} \in \mathbb{R}^{M\times P_2}$, as

$$\tilde{\mathbf{Q}} = (\tilde{\mathbf{q}}_1, \tilde{\mathbf{q}}_2, \ldots, \tilde{\mathbf{q}}_M)^\top, \tag{9}$$

and assemble the RGB matrix, $\mathbf{U} \in \mathbb{R}^{M\times 3}$, as

$$\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_M)^\top. \tag{10}$$

Then the objective of global polynomial regression is to find the inverse transform, $\mathbf{H} \in \mathbb{R}^{P_2\times 3}$, that minimizes the following cost function

$$\|\mathbf{U} - \tilde{\mathbf{Q}}\mathbf{H}\|^2, \tag{11}$$

which yields the transform $\mathbf{H}$ under the least-squares sense,

$$\mathbf{H} = (\tilde{\mathbf{Q}}^\top\tilde{\mathbf{Q}})^{-1}\tilde{\mathbf{Q}}^\top\mathbf{U}. \tag{12}$$

With the computed transform $\mathbf{H}$, the RGB vector, $\hat{\mathbf{u}} \in \mathbb{R}^{3\times 1}$, corresponding with the given CIEXYZ vector $\mathbf{q}$ under a certain illuminant, is predicted as

$$\hat{\mathbf{u}} \triangleq \mathcal{H}(\mathbf{q}) = \mathbf{H}^\top\tilde{\mathbf{q}}. \tag{13}$$

Without loss of generality, suppose that three illuminants, namely, D65, A, and F11, are considered. Accordingly, we obtain three global transforms, represented by $\mathcal{H}_\mathrm{D}(\cdot)$, $\mathcal{H}_\mathrm{A}(\cdot)$, and $\mathcal{H}_\mathrm{F}(\cdot)$ for these illuminants, and the RGB candidates are computed as $\hat{\mathbf{u}}_{\mathrm{D65}} = \mathcal{H}_\mathrm{D}(\mathbf{q})$, $\hat{\mathbf{u}}_\mathrm{A} = \mathcal{H}_\mathrm{A}(\mathbf{q})$, $\hat{\mathbf{u}}_{\mathrm{F11}} = \mathcal{H}_\mathrm{F}(\mathbf{q})$. We note that these RGB candidates should be the best solution to the specific illuminant (e.g., D65), but may result in higher errors under other illuminants (e.g., A and F11). To balance the errors under various illuminants, the straightforward way is to average these RGB values. However, due to the nonlinearity between RGB and CIEXYZ, this simple averaging is not optimal. Therefore, we deploy numerical optimization to compute the target RGB values, as will be elaborated in the following. The accuracy of simple averaging and numerical optimization will be compared in the experiment.

4.2 *RGB Optimization*

With the computed RGB candidates under various illuminants, we can obtain the minimum RGB values $R_{\min}$, $G_{\min}$, $B_{\min}$ and maximum values $R_{\max}$, $G_{\max}$, $B_{\max}$ in the respective dimensions. With these extreme values, a local RGB candidate space Ω_{RGB} can be constructed. As the RGB candidates are computed from individual illuminants, it can be expected that the optimal target RGB vector belongs to, or at least is close to, the local space Ω_{RGB}.

According to the forward model, the reflectance spectrum corresponding to the RGB vector $\mathbf{u}$ is predicted as $\hat{\mathbf{r}} = \mathcal{F}(\mathbf{u})$. The CIELAB color difference between actual spectrum $\mathbf{r}$ and the predicted spectrum $\hat{\mathbf{r}}$ is denoted as $\Delta E_m(\mathbf{r}, \mathcal{F}(\mathbf{u}))$, where

$m \in \{\mathrm{D65}, \mathrm{A}, \mathrm{F11}\}$. The inverse model requires that the optimal RGB vector produces low color differences under all these illuminants, and accordingly, the cost function is defined as

$$J(\mathbf{u}) = \sum_m \Delta E_m(\mathbf{r}, \mathcal{F}(\mathbf{u})). \tag{14}$$

For the printer used in this work, the span of Ω_{RGB} at each dimension is approximately 10 in range [0,255]. This means that exhaustive searching of optimal input $\mathbf{u}_{\mathrm{opt}}$ requires about 1000 evaluations of cost function, which is computationally expensive for practical applications. Therefore, we resort to differential evolution to compute the optimal RGB vector.

Differential evolution, since its introduction by Storn and Price[25,26] has been successfully applied in mechanical engineering[27] and pattern recognition.[28] As with other evolution algorithms, differential evolution generally involves three steps, i.e., mutation, crossover, and selection. Accordingly, typical parameters include population size, mutation scaling factor, and crossover rate. Recent studies show that its performance can be further improved by balancing exploration and exploitation abilities, self-adaptation, and parameter control.[29,30] Considering the RGB space is of low dimension and integer data type, we accordingly modify the algorithm so that it adapts well to our problem.

In the algorithm, let N_P be the population size and $g \in \{1, 2, \ldots, g_{\max}\}$ the generation, then the j'th population in evolution is represented by $\mathbf{u}_{j,g}$, where $1 \le j \le N_P$. In this work we set $N_P = 9$ and $g_{\max} = 3$. When using large N_P and $g_{\max}$ values, the improvement of colorimetric accuracy is not obvious. This indicates that the proposed method is insensitive to population size and generation number.

The initial RGB vector, $\mathbf{u}_{j,0} = (R_{j,0}, G_{j,0}, B_{j,0})^{\top}$, for the j'th population is generated randomly in the local RGB candidate space Ω_{RGB}:

$$R_{j,0} = R_{\min} + \lfloor \mathrm{rand} \cdot (R_{\max} - R_{\min}) \rfloor, \tag{15}$$

$$G_{j,0} = G_{\min} + \lfloor \mathrm{rand} \cdot (G_{\max} - G_{\min}) \rfloor, \tag{16}$$

$$B_{j,0} = B_{\min} + \lfloor \mathrm{rand} \cdot (B_{\max} - B_{\min}) \rfloor, \tag{17}$$

where the operator rand generates uniformly distributed random number in range (0,1), and $\lfloor \cdot \rfloor$ rounds the variable to the nearest integer.

After population initialization, trial vector is generated by adding the difference between two population vectors to a third one, which is referred as mutation. More specifically, the target vector is computed as

$$\mathbf{v}_{j,g+1} = \mathbf{u}_{\mathrm{best},g} + F \cdot (\mathbf{u}_{r_1} - \mathbf{u}_{r_2}), \tag{18}$$

where the random indices $r_1, r_2 \in \{1, 2, \ldots, N_P\}$ are mutually exclusive, and the scaling factor $F \in (0, 2)$. Note that, in Eq. (18), the trial vectors are generated from the current best vector $\mathbf{u}_{\mathrm{best},g}$. This is because the local minima are always closely distributed in the RGB space, and their number is not larger than three. Furthermore, as the local space Ω_{RGB} is only 3-D and of relatively limited span, we use full mutation by setting $F = 1$. After mutation, a crossover procedure is usually adopted to increase the potential diversity of the population. In this work full crossover is adopted.

To determine whether $\mathbf{v}_{j,g+1}$ should become a member of generation $g + 1$, the trial vector $\mathbf{v}_{j,g+1}$ is compared with the target vector $\mathbf{u}_{j,g}$ according to the cost function defined in Eq. (14). If the trial vector $\mathbf{v}_{j,g}$ produces a cost lower than $\mathbf{u}_{j,g}$, then $\mathbf{u}_{j,g}$ is set as $\mathbf{v}_{j,g}$; otherwise $\mathbf{u}_{j,g}$ keeps unchanged. After $g_{\max}$ generation, we get the optimal RGB vector

$$\mathbf{u}_{\mathrm{opt}} = \arg \min_{\mathbf{u}} J(\mathbf{u}). \tag{19}$$

We note that, some reflectance spectra may exceed the printer gamut, i.e., the values of $\mathbf{u}_{\mathrm{opt}}$ are out of range [0,255]. In this case, we simply clip the RGB values into the range [0,255].

5 Experiments and Discussion

In the experiment, we employed a CMYKcm printer (Epson Stylus Photo 1390) to evaluate the proposed printer characterization method. The substrate used for printing was Epson Gloss Photo Paper. As it is not feasible to control ink densities directly with the accompanied drivers, the printer was treated as a RGB device.

Totally 216 ($=6 \times 6 \times 6$) print colors were used to establish the forward and inverse models, and other 100 random print colors were used to evaluate the forward model. To evaluate the inverse model, we employed color samples in three different media, including 96 patches of the X-Rite ColorChecker SG (referred as SG hereafter), 100 randomly selected fabric Pantone patches (referred as Pantone hereafter), and multispectral images of textile fabrics. These color targets and their corresponding usage are listed in Table 1. The spectra of these solid samples were measured by GretagMacbeth ColorEye XTH spectrophotometer. The spectral images of textile fabrics were acquired using the multispectral imaging system developed in our laboratory.[3]

5.1 *Accuracy of Forward Model*

The proposed forward model is compared with the commonly adopted 3-D LUT method on the same 216 training samples. Spline interpolation was adopted in the LUT method. Table 2 shows the spectral root-mean-square (rms) errors and CIEDE 2000 color difference errors,[31] ΔE_{00}, for the proposed method and LUT method. The spectral mean

Table 1 The color targets and their corresponding usage.

Target	Sample number	Usage
Print training samples	216	Establish forward and inverse models
Print test samples	100	Evaluate forward model
SG samples	96	Evaluate inverse model
Pantone samples	100	Evaluate inverse model
Multispectral image	—	Evaluate inverse model

Table 2 Spectral rms errors and ΔE_{00} errors of the proposed forward model and the LUT method.

Method	Spectral rms			ΔE_{00} under D65			ΔE_{00} under A			ΔE_{00} under F11		
	Mean	Std.	Max.	Mean	Std.	Max.	Mean	Std.	Max.	Mean	Std.	Max.
LUT	0.0058	0.0030	0.0200	0.85	0.43	2.54	0.82	0.40	2.36	0.91	0.48	2.70
Proposed	0.0034	0.0021	0.0115	0.50	0.29	1.46	0.48	0.28	1.25	0.50	0.30	1.45

rms error is 0.0034, and the mean ΔE_{00} errors under various illuminants are about 0.5. These error magnitudes are clearly lower than those of the LUT method.

As the second-order polynomial regression is applied on each cellular subset, it is necessary to evaluate the color discontinuity between neighboring subsets. Figure 4 shows the lightness discontinuity of the center point $(R = 127, G = 127, B = 127)$ across the cell boundaries in the red, green, and blue dimensions. As can be found, the discontinuity is visible only for a small number of cell boundaries. Our investigation indicates that the average ΔE_{00} at all cell boundaries is around 0.12 under the D65 illuminant. This magnitude is considerably lower than the modeling error of the polynomial regression.

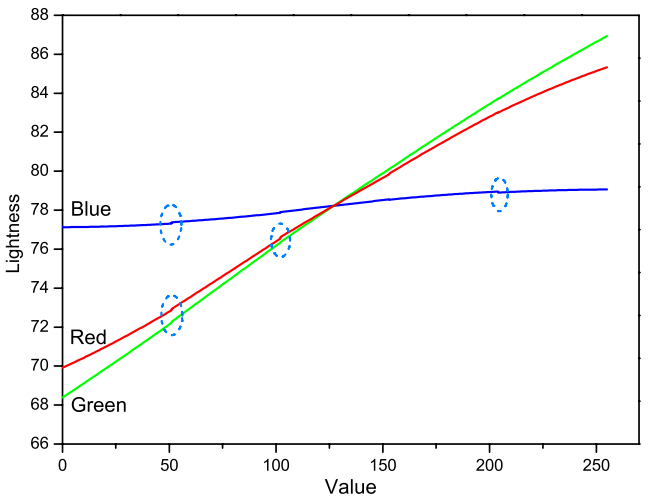

Fig. 4 The lightness discontinuity of the center point $(R = 127, G = 127, B = 127)$ across the cell boundaries in the red, green, and blue dimensions. The dotted ellipses mark the most visible discontinuities.

5.2 *Accuracy of Inverse Model*

Table 3 shows the ΔE_{00} errors when the inverse modeling is applied on a single individual illuminant. As expected, the colorimetric differences are quite small when the illuminants in the modeling and evaluation procedures are identical, but are considerably large otherwise. For example, for the SG target, when the modeling illuminant is D65, the average ΔE_{00} errors are 1.37, 2.60, and 2.71 for the evaluation illuminants D65, A, and F11, respectively. The unbalanced error distribution is generally unacceptable in practical applications, and this consequently motivates the adoption of numerical optimization strategy in this work.

Note that, the cost function [Eq. (14)] of the inverse model is defined on the predicted colorimetric errors between the input spectrum and the predicted spectrum using forward model. When evaluating the accuracy of the inverse model, however, the actual colorimetric errors are computed between the input spectrum and the spectrum of the actually printed color. It is expected that, due to unperfect accuracy of the forward model, these two errors are not identical. Figure 5 shows the correlations of the predicted and actual errors of the Pantone samples. It is observed that, although the magnitudes of these two errors are not identical, they are statistically proportional. This in turn validates that the cost function [Eq. (14)] is reliable for RGB optimization.

In the inverse model, we used Vrhel's method,[15] which is based on multidimensional LUT, as the baseline method. Both Vrhel's method and the proposed method were evaluated on the same data sets. Table 4 shows the ΔE_{00} errors of

Table 3 The ΔE_{00} errors under a single illuminant for SG samples and Pantone samples.

Target	Illuminant	ΔE_{00} under D65			ΔE_{00} under A			ΔE_{00} under F11		
		Mean	Std.	Max.	Mean	Std.	Max.	Mean	Std.	Max.
SG	D65	1.37	0.64	3.78	2.60	1.67	7.43	2.71	1.45	6.86
	A	3.01	2.13	8.43	1.22	0.60	3.25	3.05	1.97	9.88
	F11	2.62	1.64	8.55	3.31	2.02	9.60	1.51	0.83	4.64
Pantone	D65	1.61	1.26	8.89	2.11	1.43	6.92	2.51	1.56	9.22
	A	2.46	1.56	8.54	1.44	1.02	6.32	2.82	1.45	8.80
	F11	2.50	1.38	8.13	2.98	1.61	6.89	1.88	1.33	8.37

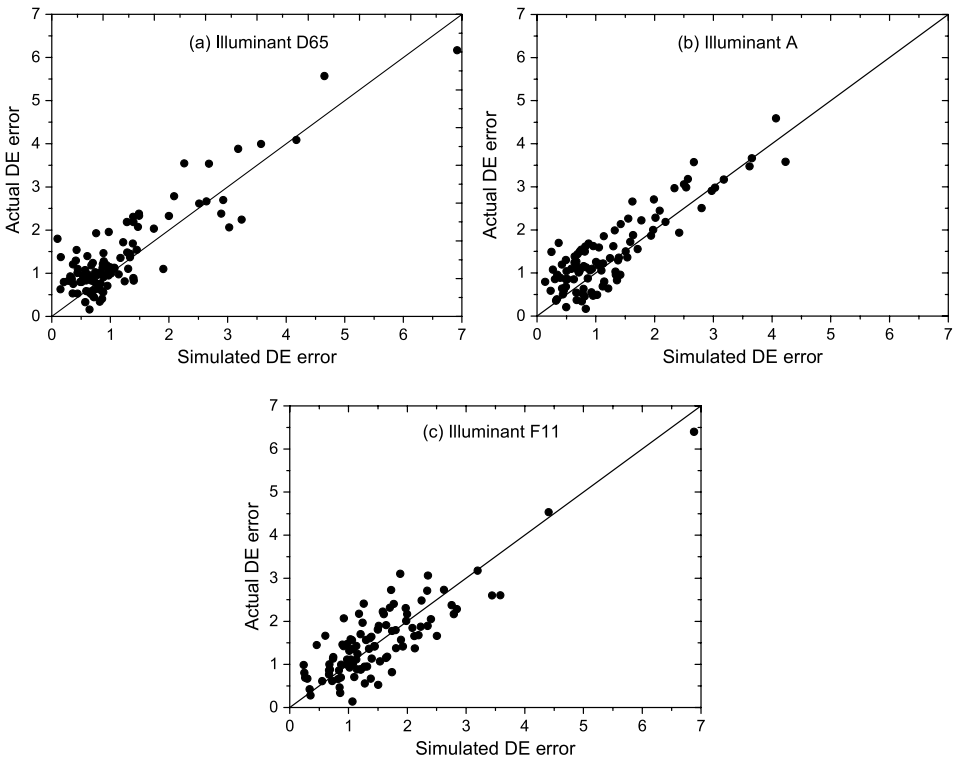

Fig. 5 The distribution of the simulated ΔE_{00} errors under various illuminants with respect to the actual ones of the Pantone samples.

Table 4 The ΔE_{00} errors of SG samples using Vrhel's method[15] and the proposed method.

		ΔE_{00} under D65			ΔE_{00} under A			ΔE_{00} under F11		
Method	**Strategy**	**Mean**	**Std.**	**Max.**	**Mean**	**Std.**	**Max.**	**Mean**	**Std.**	**Max.**
Vrhel's	Averaging	4.13	2.72	13.1	4.88	2.77	13.7	5.02	3.01	14.9
	Optimization	2.20	1.74	9.68	2.59	1.79	8.59	2.53	1.54	9.23
Proposed	Averaging	1.87	1.15	6.07	1.93	1.13	5.56	1.94	0.96	4.29
	Optimization	1.56	1.02	5.27	1.62	1.11	5.70	1.60	0.83	3.92

Table 5 The ΔE_{00} errors of Pantone samples using Vrhel's methodx[15] and the proposed method.

		ΔE_{00} under D65			ΔE_{00} under A			ΔE_{00} under F11		
Method	**Strategy**	**Mean**	**Std.**	**Max.**	**Mean**	**Std.**	**Max.**	**Mean**	**Std.**	**Max.**
Vrhel's	Averaging	3.83	2.71	12.2	4.14	2.57	11.9	4.31	2.69	12.1
	Optimization	1.89	1.38	8.47	2.06	1.39	10.2	2.17	1.35	8.20
Proposed	Averaging	1.65	1.06	7.04	1.83	1.10	6.96	1.84	1.13	7.55
	Optimization	1.43	1.04	6.17	1.47	0.90	4.59	1.55	0.90	6.40

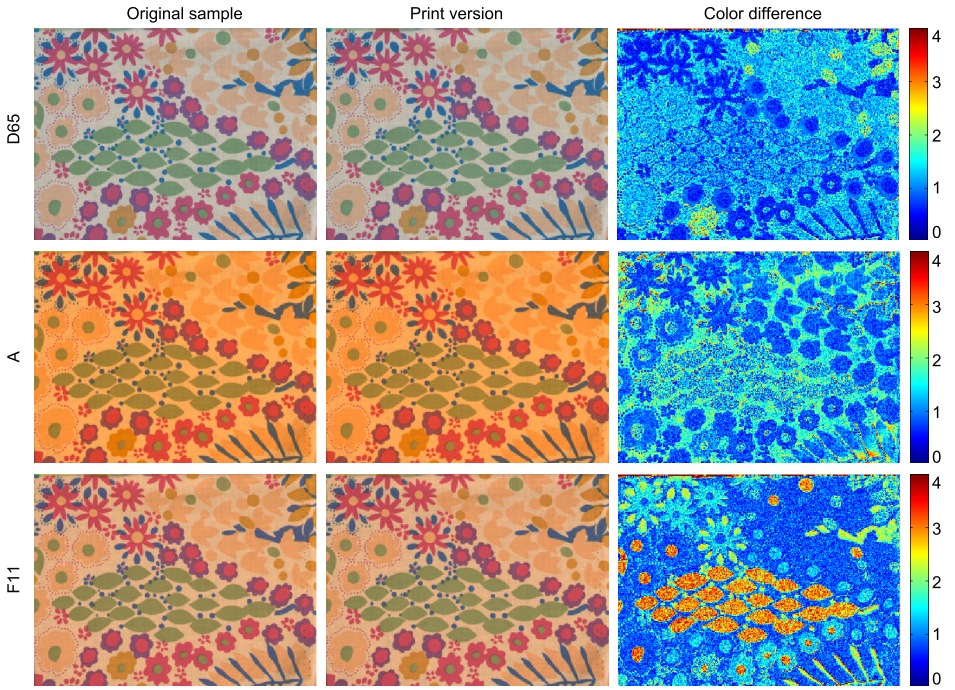

Fig. 6 An acquired multispectral image of an textile fabric sample and its print version, displayed under various illuminants (D65, A, and F11). The ΔE_{00} error maps are in the range of 0 and 4.0. The images are transformed to sRGB color space for visualization.

the two methods for the SG targets, with averaging and optimization strategies. It is clear that, for both methods, the optimization strategy produces lower colorimetric errors. Under the optimization strategy, the average ΔE_{00} errors of the proposed method are around 1.5, while those of Vrhel's method are above 2.2. Further evaluation on the Pantone color target (Table 5) also validates the superiority of the proposed method over the baseline one.

We also applied the proposed inverse method for the reproduction of multispectral images. Figure 6 shows the actual and printed images of a textile fabric under three illuminationsas well as their colorimetric errors. For visualization purpose, these images are transformed into the sRGB color space.[32] Note that, as actual spectrum measurement for each pixel is unfeasible, the print image is generated by first computing the printer control values via the inverse model, and then predicting spectra via the forward model. It is observed that, for illuminants D65 and A, the color difference errors are relatively low and widely distributed; while for illuminant F11, the green regions exhibits high errors. Overall, the original sample and print versions under these illuminants are visually undistinguishable, which further validates the effectiveness of the proposed method.

6 Conclusion

For most desktop printers, the ink densities are not individually controllable, and thus the YNSN model cannot be directly applied. To cope with this problem, this paper proposes a unified method, which consists of forward and inverse models, to characterize desktop printers. Considering the nonlinear nature of the printing process, the modeling is carried out in a locally adaptive manner. In the forward model, the transform is computed using adaptive polynomial regression for each cellular subset. In the inverse model, the RGB candidates of an input spectrum are first obtained using global regression, and then the target RGB values are estimated using numerical optimization. Experimental results indicate that the proposed method produces high color accuracy, which is sufficient to most practical applications.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (NSFC) under Grant Nos. 60778050 and 61271339, and the Hong Kong Research Institute of Textiles and Apparel (HKRITA) under Grant No. ITP/001/10TP.

References

1. Y. Murakami, T. Obi, and M. Yamaguchi, "Spectral reflectance estimation from multi-band image using color chart," *Opt. Comm.* **188**(1), 47–54 (2001).
2. N. Shimano, "Recovery of spectral reflectance of objects being imaged without prior knowledge," *IEEE Trans. Image Process.* **15**(7), 1848–1856 (2006).
3. H. L. Shen et al., "Correcting cross-media instrument metamerism for reflectance estimation in multispectral imaging," *J. Opt. Soc. Am. A* **28**(4), 511–516 (2011).
4. R. Bala and G. Sharma, "System optimization in digital color imaging," *IEEE Signal Proc. Mag.* **22**(1), 55–63 (2005).
5. D. R. Wyble and R. S. Berns, "A critical review of spectral models applied to binary color printing," *Color Re. App.* **25**(1), 4–19 (2000).
6. S. Zuffi, R. Schettini, and G. Mauri, "Spectral-based printer modeling and characterization," *J. Electronic Im.* **14**(2), 023008 (2005).
7. I. S. Jang et al., "Improved inverse characterization of multicolorant printer using colorant correlation," *J. Imag. Sci. Tech.* **51**(2), 175–184 (2007).
8. J. A. C. Yule and W. J. Neilsen, "The penetration of light into paper and its effect on halftone reproduction," in *TAGA Proc.*, pp. 65–76, Technical Association of the Graphic Arts, Sewickley, American (1951).
9. M. R. Rosen, E. F. Hattenberger, and N. Ohta, "Spectral redundancy in a six-ink ink jet printer," *J. Imag. Sci. Tech.* **48**(3), 194–202 (2004).
10. M. Xia et al., "End-to-end color printer calibration by total least squares regression," *IEEE Trans. Image Proc.* **8**(5), 700–716 (1999).
11. T. Bugnon and R. D. Hersch, "Constrained acquisition of ink spreading curves from printed color images," *IEEE Trans. Image Proc.* **20**(2), 513–5223 (2011).

12. R. Balasubramanian, "Optimization of the spectral Neugebauer model for printer characterization," *J. Electron. Im.* **8**(2), 156–166 (1999).
13. P. Urban, M. R. Rosen, and R. S. Berns, "Accelerating spectral-based color separation within the Neugebauer subspace," *J. Electron. Im.* **16**(4), 043014 (2007).
14. J. Gerhardt and J. Y. Hardeberg, "Spectral color reproduction minimizing spectral and perceptual color differences," *Color Re. App.* **33**(6), 494–504 (2008).
15. M. J. Vrhel and H. J. Trussell, "Color printer characterization in MATLAB," in *IEEE Inter. Conf.on Image Proc.*, Vol. 1, pp. 457–460, IEEE Signal Processing Society, Rochester, New York (2002).
16. R. Balasubramanian, "Reducing the cost of lookup table based on color transforms," *J. Imag. Sci. Tech.* **44**(4), 321–327 (2000).
17. A. Artusi and A. Wilkie, "Novel color printer characterization model," *J. Electron. Im.* **12**(3), 448–458 (2003).
18. J. Morovic et al., "Sampling optimization for printer characterization by greedy search," *IEEE Trans. Image Proc.* **19**(10), 2705–2711 (2010).
19. J. Morovic and M. R. Luo, "Characterising desktop colour printers without full control over all colorants," in *The Fourth Color Imaging Conference: Color Science, System, and Applications*, IS&T, pp. 70–74, Society for Imaging Science and Technology, Scottsdale, Arizona (1996).
20. M. S. Drew et al., "Colour printer characterization by regression with greyspace constraint," in *The Sixth Color Imaging Conference: Color Science, System, and Applications*, IS&T, pp. 193–196, Society for Imaging Science and Technology, Scottsdale, Arizona (1998).
21. A. Verikas, K. Malmqvist, and L. Bergman, "Neural networks based colour measuring for process monitoring and control in multicoloured newspaper printing," *Neural Comput. Appl.* **9**(3), 227–242 (2000).
22. M. R. Gupat, E. K. Garcia, and E. Chin, "Adaptive local linear regression with application to printer color management," *IEEE Trans. Image Proc.* **17**(6), 936–945 (2008).
23. G. D. Finlayson and M. S. Drew, "Constrained least-squares regression in color spaces," *J. Electron. Im.* **6**(4), 484–493 (1997).
24. T. Johnson, "Methods for characterizing colour printers," *Displays* **16**(4), 193–202 (1996).
25. R. Storn and K. V. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous space," *J. Global Opt.* **11**(4), 341–359 (1997).
26. R. Storn, K. V. Price, and J. Lampinen, *Differential Evolution -A Practical Approach to Global Optimization*, Springer-Verlag, Berlin, Germany (2005).
27. R. Joshi and A. C. Sanderson, "Minimum representation multi-sensor fusion using differential evolution," *IEEE Trans. Syst. Man Cybern. Part A* **29**(1), 63–76 (1999).
28. S. Das, A. Abraham, and A. Konar, "Automatic clustering using improved differential evolution algorithm," *IEEE Trans. Syst. Man Cybern. Part A* **38**(1), 218–237 (2008).
29. S. Das et al., "Differential evolution using a neighborhood-based mutation operator," *IEEE Trans. Evol. Comp.* **13**(3), 526–553 (2009).
30. A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comp.* **13**(2), 398–417 (2009).
31. M. R. Luo, G. Cui, and B. Rigg, "The development of the CIE 2000 colour-difference formula: CIEDE2000," *Color Res. App.* **26**(5), 340–350 (2001).
32. M. Stokes et al., "A standard default color space for the internet: sRGB," Version 1.10, International Color Consortium, http://www.color.org/sRGB.xalter (1996).

Biographies and photographs of the authors are not available.