

# Polynomial modeling and optimization for colorimetric characterization of scanners

Simone Bianco  
Francesca Gasparini  
Raimondo Schettini  
Leonardo Vanneschi

University of Milan-Bicocca  
DISCo, Department of Computer Science, Systems and Communication  
viale Sarca 336  
20126 Milan, Italy  
E-mail: simone.bianco@disco.unimib.it

---

**Abstract.** We present different computational strategies for colorimetric characterization of scanners using multidimensional polynomials. The designed strategies allow us to determine the coefficients of an a priori fixed polynomial, taking into account different color error statistics. Moreover, since there is no clear relationship between the polynomial chosen for the characterization and the intrinsic characteristics of the scanner, we show how genetic programming could be used to generate the best polynomial. Experimental results on different devices are reported to confirm the effectiveness of our methods with respect to others in the state of the art. © 2008 SPIE and IS&T. [DOI: 10.1117/1.2982004]

---

## 1 Introduction

Scanners have their own reference systems for the specification of color (device-dependent spaces). Since the description of a color in these spaces is based on the characteristics of the device concerned, it does not constitute an objective definition of that color. To facilitate the reproduction of colors on various devices and supports, we must employ a system of description that allows us to define the color in an unequivocal fashion, i.e., in a device-independent space, separating the way colors are defined from the way the various devices represent them. A point in RGB space indicates how a color stimulus is produced by a given device, while a point in a colorimetric space, such as CIELAB space, indicates how the color is perceived in standard viewing conditions. Let us consider the function that associates at every point in the device-dependent space the colorimetric value of the corresponding color. The colorimetric characterization of a scanner device consists in rendering this function explicitly. It must take into account the peculiar characteristics of the device; consequently, every device calls for specific conversion functions.

A simple transformation can be used to relate RGB values of a scanner to CIE tristimulus values  $XYZ$  if and only if the digitizer channel sensitivities satisfy the Luther conditions by which they are a linear transformation of the color matching functions that define the CIE colorimetric

observer. Relatively few of the scanners and cameras that can be found on the current market have been designed to meet this constraint. This circumstance motivates several different approaches to the problem from spectral and colorimetric<sup>1</sup> points of view. Spectral characterization approaches try to recover reflectance information from the scanner responses and compute the colorimetric values from the recovered reflectance information.<sup>2,3</sup> These approaches assume that the spectral scanner sensitivity can be accurately measured or recovered mathematically. The advantage of these techniques over colorimetric ones for traditional RGB scanners using a single illuminant is not evident.<sup>4</sup> Among various colorimetric approaches, neural networks, 3-D interpolation, and polynomial regression have been widely investigated.<sup>5</sup> Kang and Anderson,<sup>6</sup> Schettini, Barolo, and Boldrin,<sup>7</sup> Vrhel and Trussel,<sup>8</sup> and Cheung *et al.*<sup>9</sup> applied artificial neural networks to scanner calibration and characterization. The results are promising, but the fact that neural networks would need large training sets that are often not available should be taken into account. The characterization function found can be rather complicated and computationally expensive. For this reason, in the design of the colorimetric profile of the device, this characterization function is usually applied to produce a finely sampled lookup table (LUT) to be used together with an interpolation scheme.<sup>5</sup> The main advantage of using polynomial regression with respect to other characterization methods is that they require smaller training sets not necessarily uniformly distributed. Since there is no clear relationship between polynomials adopted and imaging device characteristics, they must be empirically determined and defined for each device and for a specific device each time a change is made in any component of the system.<sup>10</sup>

Usually, the accuracy of the characterization increases as the number of terms in the polynomial increases; however, there is not a simple way to avoid data overfitting. Moreover, polynomial regression usually minimizes only the average color error, and therefore it does not guarantee a uniform accuracy across the entire color gamut. Using an average color error as the objective to be minimized, the polynomial coefficients can be found by using the least square method (LS) or the total least squares method

---

Paper 07193RR received Oct. 2, 2007; revised manuscript received May 20, 2008; accepted for publication Jul. 16, 2008; published online Oct. 20, 2008.

1017-9909/2008/17(4)/043002/13/\$25.00 © 2008 SPIE and IS&T.

(TLS).<sup>11</sup> More powerful methods, such as total color difference minimization (TCDM) and CIELAB least squares minimization (LAB-LS), have been recently proposed<sup>12</sup> to minimize nonlinear functions that take into account the average CIELAB colorimetric error.

Taking these methods and results as our point of departure, we try to solve the colorimetric characterization problem defining a set of novel functions that take into account not only the mean perceptual colorimetric error but also other error statistics. We adopt genetic algorithms (GAs)<sup>13,14</sup> to minimize these functions and to find the corresponding coefficients from the given polynomial terms.

In a second phase of our work, we try to further improve the accuracy of the characterization by generating new polynomials. For this task, we use genetic programming (GP),<sup>15,16</sup> a domain-independent evolutionary method that automatically breeds a population of expressions, functions, or more generally, computer programs to solve a problem. To the best of our knowledge, this work represents the first attempt to use GP for this task.

This work is structured as follows. In Sec. 2 we introduce the problem of colorimetric/characterization of scanners. Section 3 contains a description of the three novel functions defined and of the corresponding optimization methods adopted in this work. Section 4 describes how to model new polynomials using GP (a more detailed description of evolutionary algorithms and of GP in particular is contained in Appendix A). In Sec. 5 we present the experimental results that we have obtained both using *a priori* fixed polynomial and the polynomials that we have found using GP. Finally, Sec. 6 concludes the work and offers some hints for future research.

## 2 Scanner Characterization

The basic model that describes the response  $\rho = [R, G, B]$  of a three-channel color scanner can be formulated as:

$$\rho = \int_{\omega} I(\lambda)R(\lambda)S(\lambda)d\lambda + \mathbf{n}, \quad (1)$$

where  $I(\lambda)$  is the scanner-illuminant spectral power distribution,  $R(\lambda)$  is the reflectance of the surface being scanned,  $S(\lambda)$  represents the three scanner sensor spectral sensitivities stacked row-wise,  $\mathbf{n}$  is the three-channel additive noise, and  $\lambda$  is the integration variable representing the wavelengths within the visible spectrum  $\omega$ . Equation (1) represents a good model if the scanner response is linear. In a more general situation, the scanner response may be subject to an input-output nonlinearity that can be represented by an optoelectronic conversion function  $\mathcal{F}$ . Equation (1) then becomes:

$$\rho = \mathcal{F}\left(\int_{\omega} I(\lambda)R(\lambda)S(\lambda)d\lambda + \mathbf{n}\right), \quad (2)$$

where  $\mathcal{F}$  models the three-channel nonlinearity functions.

Using vector space notation, and sampling the visible spectrum in  $N$  equally sampled wavelengths, Eq. (2) can be rewritten as:

$$\rho = \mathcal{F}(\mathbf{SIR} + \mathbf{n}), \quad (3)$$

where  $\rho$  is a  $3 \times 1$  vector,  $\mathbf{S}$  is the  $3 \times N$  matrix formed by stacking the scanner spectral sensitivities row-wise,  $\mathbf{I}$  is the  $N \times N$  diagonal matrix whose elements are the samples of the scanner-illuminant spectral power distribution,  $\mathbf{R}$  is the  $N \times 1$  vector of the reflectance of the surface being scanned, and  $\mathbf{n}$  is the  $3 \times 1$  noise vector.

Similarly, the CIEXYZ tristimulus values, denoted by a  $3 \times 1$  vector  $\mathbf{s}$ , are defined as:

$$\mathbf{s} = \mathbf{CLR}, \quad (4)$$

where  $\mathbf{C}$  is the  $3 \times N$  matrix of the CIEXYZ color matching functions and  $\mathbf{L}$  is the  $N \times N$  diagonal matrix whose elements are the samples of the viewing-illuminant spectral power distribution.

The characterization problem is to find the mapping  $\mathcal{M}$  that transforms the recorded values  $\rho$  to their corresponding CIEXYZ values  $\mathbf{s}$ :

$$\mathbf{s} = \mathcal{M}(\rho). \quad (5)$$

We address this problem using a two-step procedure: first the optoelectronic conversion function  $\mathcal{F}$  is estimated and  $\mathcal{F}^{-1}$  is applied to the  $\rho$  data to linearize them; then an  $m$ 'th-order polynomial mapping  $\mathbf{M}$  is applied to the linearized data  $\mathcal{F}^{-1}(\rho)$  to obtain  $\mathbf{s}$ . The general  $m$ 'th-order polynomial  $P(R, G, B)$  with three variables can be given as follows:

$$P(R, G, B) = \sum_{i=0}^m \sum_{j=0}^m \sum_{k=0}^m R^i G^j B^k \text{ with } i + j + k \leq m. \quad (6)$$

Given the scanner response  $\rho$ , their linearized values  $\mathcal{F}^{-1}(\rho)$  and the polynomial model  $P$  to use, we can calculate the polynomial expansion  $\mathbf{r}$  of  $\mathcal{F}^{-1}(\rho)$  as  $\mathbf{r} = P[\mathcal{F}^{-1}(\rho)]$ . Using the polynomial modeling, Eq. (5) then becomes:

$$\mathbf{s} = \mathbf{MP}[\mathcal{F}^{-1}(\rho)] = \mathbf{Mr}. \quad (7)$$

The first step to find the matrix  $\mathbf{M}$  is to select a collection of color patches that spans the device gamut. The reflectance spectra of these  $N_c$  color patches is denoted by  $\mathbf{R}_k$  for  $k \in \{1, \dots, N_c\}$ . These patches are measured using a spectrophotometer or a colorimeter, which provides the device-independent values:

$$\mathbf{s}_k = \mathbf{CLR}_k \text{ for } k \in \{1, \dots, N_c\}. \quad (8)$$

Without loss of generality,  $\mathbf{s}_k$  can be transformed into any colorimetric or device-independent values. The same  $N_c$  patches are also acquired with the scanner to be characterized, providing  $\rho_k = \mathcal{F}(\mathbf{SIR}_k + \mathbf{n})$  with calculated polynomial expansions  $\mathbf{r}_k$ , for  $k \in \{1, \dots, N_c\}$ .

Algebraically, the characterization problem is to find the matrix  $\mathbf{M}$ :

$$\mathbf{M} = \arg \left[ \min_{\mathbf{M} \in \mathbb{R}^{3 \times q}} \sum_{k=1}^{N_c} \|\mathbf{M}\mathbf{r}_k - \mathcal{L}(\mathbf{s}_k)\|^2 \right], \quad (9)$$

where  $\mathcal{L}(\cdot)$  is the transformation from CIEXYZ to the appropriate standard color space chosen, and  $\|\cdot\|$  is the error metric in the chosen color space.

$\mathbf{M}$  is a  $3 \times q$  matrix, where  $q$  is the number of terms of the polynomial  $P(R, G, B)$ ; the number of terms  $q$  is related to the order  $m$  of the polynomial by:

$$q = \left[ \sum_{k=1}^m \binom{k+2}{2} + 1 \right].$$

$\mathbf{M}$  being a  $3 \times q$  matrix, the problem of finding  $\mathbf{M}$  consists of determining  $3q$  coefficients; in having enough equations to solve for the  $3q$  unknowns and to deal with a less ill-posed problem, then we have to use  $N_c \geq q$  different color patches. In other words, the higher the order of the polynomial, the larger the number of its terms, and consequently the larger the number of different color patches that we have to use.

Different functions to be minimized can be defined to find the unknown matrix  $\mathbf{M}$ . Depending on the function adopted, different optimization methods have to be considered.

The easiest way to find the unknown matrix  $\mathbf{M}$  in Eq. (9) is to minimize the function:

$$\mathcal{H}_{LS} = \|\mathbf{s} - \hat{\mathbf{s}}\|_2 \quad \text{with } \hat{\mathbf{s}} = \mathbf{M}\mathbf{r}, \quad (10)$$

using the least squares minimization (LS). Analytically, the matrix  $\mathbf{M}$  can be easily found by:

$$\mathbf{M} = \mathbf{s}\mathbf{r}^T(\mathbf{r}\mathbf{r}^T)^{-1}. \quad (11)$$

The LS method assumes that errors are present only in the matrix  $\mathbf{s}$ , while the matrix  $\mathbf{r}$  is assumed free of error.<sup>11</sup>

The total least squares (TLS) method<sup>11</sup> is a generalization of the LS method: it assumes that both the matrices  $\mathbf{s}$  and  $\mathbf{r}$  are affected by error. It searches for the solution  $\mathbf{M}$  that minimizes the function:

$$\mathcal{H}_{TLS} = \|[\mathbf{r}; \mathbf{s}] - [\hat{\mathbf{r}}; \hat{\mathbf{s}}]\|_F \quad \text{with } \hat{\mathbf{s}} = \mathbf{M}\hat{\mathbf{r}}, \quad (12)$$

where  $\|\cdot\|_F$  is the Frobenius norm, defined for a generic  $m \times n$  matrix  $\mathbf{A}$  as:

$$\|\mathbf{A}\|_F = \left( \sum_{i=1}^m \sum_{j=1}^n \|a_{ij}\|^2 \right)^{1/2}. \quad (13)$$

The analytic solution for the minimization of  $\mathcal{H}_{TLS}$  exists and can be found using singular value decomposition.

Since the color accuracy of the characterization methods is evaluated using a color error in the CIELAB color space, total color difference minimization (TCDM)<sup>12</sup> has been proposed to search for the matrix  $\mathbf{M}$  that minimizes the sum of the CIELAB  $\Delta E_{94}$  color error between the measured and the predicted CIELAB values (respectively  $\mathbf{s}_{Lab}$  and  $\hat{\mathbf{s}}$ ) for the  $n$  patches, i.e.:

$$\mathcal{H}_{TCDM} = \sum_{i=1}^n \Delta E_{94}(\mathbf{s}_{Lab}, \hat{\mathbf{s}}) \quad \text{with } \hat{\mathbf{s}} = \mathbf{M}\mathbf{r}. \quad (14)$$

The minimization of  $\mathcal{H}_{TCDM}$  does not have an analytic solution, and has to be solved with an iterative method. In this work following Ref. 12, it has been solved using a downhill simplex method, as it does not require the calculation of derivatives of the objective function.<sup>17</sup>

The CIELAB least squares minimization (LAB-LS)<sup>12</sup> method employs a preprocessing consisting of a  $p$ 'th root correction of the scanner responses  $\mathbf{r}$  before calculating the LS regression with the CIELAB values  $\mathbf{s}_{Lab}$  of the measured patches. The LAB-LS method then minimizes:

$$\mathcal{H}_{LAB-LS} = \|\mathbf{s}_{Lab} - \hat{\mathbf{s}}\|_2 \quad \text{with } \hat{\mathbf{s}} = \mathbf{M}\mathbf{r}^{1/p}. \quad (15)$$

This  $p$ 'th root correction has the aim to compensate for the cubic root relationship between the RGB scanner color space and the CIELAB color space. It can be easily found that if the RGB data have been properly linearized, the best choice for the  $p$ 'th root is  $p=3m$ , where  $m$  is the order of the polynomial used. Such a choice for  $p$  permits the cancellation of the cubic relationship whatever the polynomial used is. Performing a LS regression, the LAB-LS method admits an analytic solution that can be found as:

$$\mathbf{M} = \mathbf{s}_{Lab}(\mathbf{r}^{1/p})^T[\mathbf{r}^{1/p}(\mathbf{r}^{1/p})^T]^{-1}. \quad (16)$$

### 3 Characterization Procedures

In this section, we describe the characterization procedures that we have designed to assess matrix  $\mathbf{M}$  given *a priori* fixed polynomial. These procedures are based on GAs (see Appendix A). Among the optimization methods that can be used, we adopted GAs, as they are a well known and widely accepted optimization method. Moreover, GAs are intrinsically parallel: most other optimization algorithms (like Hill Climbing,<sup>18</sup> Simulated Annealing,<sup>19</sup> Tabu Search<sup>20,21</sup>) are serial and can only explore the solution space to a problem one direction at a time. If that direction leads to a local- or suboptimal solution, another direction has to be taken by means of some backtracking technique or by restarting the algorithm from the beginning. On the other hand, since GAs have multiple offspring, they can explore the solution space in multiple directions at once. This makes GAs particularly well suited to solving problems where the space of all potential solutions is complex.<sup>15,16</sup> In particular, we introduce three novel functions that are minimized by GAs.

The CIELAB genetic algorithm (LAB-GA) procedure is a hybrid procedure based on both TCDM and LAB-LS. It uses the same preprocessing  $p$ 'th root correction of the scanner responses calculated by LAB-LS, but then minimizes the CIELAB  $\Delta E_{94}$  color error between these values  $\hat{\mathbf{s}}$  and the CIELAB measured values  $\mathbf{s}_{Lab}$ :

$$\mathcal{H}_{LAB-GA} = \sum_{i=1}^n \Delta E_{94}(\mathbf{s}_{Lab}, \hat{\mathbf{s}}) \quad \text{with } \hat{\mathbf{s}} = \mathbf{M}\mathbf{r}^{1/p}. \quad (17)$$

The LAB-GA procedure inherits the nonexistence of an analytical solution from the TCDM method.

**Table 1** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score (on the test set) of the seven procedures considered using the full third polynomial on the Macbeth ColorChecker DC dataset.

Method	$\Delta E_{94}$ training			$\Delta E_{94}$ test			$\Delta E_{94}$ total			WST score
	Mean	Max	Std Dev	Mean	Max	Std Dev	Mean	Max	Std Dev	
LS	1.71	12.92	1.83	1.61	6.39	1.53	1.68	12.92	1.71	1
TLS	2.74	51.65	6.24	2.14	15.92	2.74	2.55	51.65	5.33	0
TCDM	1.53	8.52	1.52	1.55	8.30	1.51	1.54	8.52	1.52	1
LAB-LS	1.33	6.66	1.13	1.23	3.16	0.80	1.29	6.66	1.05	3
LAB-GA	1.25	8.21	1.16	1.16	3.66	0.81	1.21	8.21	1.00	3
LAB-GAp	1.23	8.18	1.16	1.15	3.64	0.80	1.20	8.18	1.00	5
LAB-GApw	1.31	5.83	1.09	1.18	2.84	0.79	1.26	5.83	0.98	5

The CIELAB genetic algorithm  $p$ 'th root (LAB-GAp) procedure is a variant of the LAB-GA procedure: it does not use the same  $p=3m$  value of LAB-LS and LAB-GA for preprocessing, but lets the GA look for it. This is justified by the fact that the  $p=3m$  is the best choice if the scanner responses are linear: even if the scanner responses have been linearized, some nonlinearity may remain. The use of the GA to look for the best value of  $p$  permits us to correct a part of the residual nonlinearity if present. The function to be minimized is now defined as:

$$\mathcal{H}_{\text{LAB-GAp}} = \sum_{i=1}^n \Delta E_{94}(s_{\text{Lab}}, \hat{s}_p) \quad \text{with } \hat{s}_p = \mathbf{M}\mathbf{r}^{1/p}, \quad (18)$$

where the notation  $\hat{s}_p$  instead of  $\hat{s}$  of Eq. (17) is to underline that  $p$  is not a fixed value now, but an added unknown to look for.

The CIELAB genetic algorithm  $p$ 'th root weighted (LAB-GApw) procedure is a variant of the LAB-GAp procedure. As LAB-GAp, it uses the GA to search for the best value of  $p$  to perform the preprocessing  $p$ 'th root correction. The difference is the objective function that it minimizes: two more terms are added in Eq. (18), one for the maximum  $\Delta E_{94}$  error and one for the  $\Delta E_{94}$  standard deviation. The three terms of the function are multiplied for a weight vector  $w=[w_1, w_2, w_3]$ , which reflects the relative importance to be given to the various terms. In this work we have heuristically adopted the following weights  $w=[w_1, w_2, w_3]=[0.35 \ 0.50 \ 0.15]$ . The LAB-GApw method then minimizes:

$$\begin{aligned} \mathcal{H}_{\text{LAB-GApw}} = & w_1 \cdot \text{mean}[\Delta E_{94}(s_{\text{Lab}}, \hat{s}_p)] \\ & + w_2 \cdot \max[\Delta E_{94}(s_{\text{Lab}}, \hat{s}_p)] \\ & + w_3 \cdot \text{std}[\Delta E_{94}(s_{\text{Lab}}, \hat{s}_p)] \quad \text{with } \hat{s}_p = \mathbf{M}\mathbf{r}^{1/p}. \end{aligned} \quad (19)$$

The first term has been changed with respect to Eqs. (17) and (18), but the problem to minimize the sum of some quantities or their mean is equivalent.

#### 4 Polynomial Modeling Using Genetic Programming

Since there is no clear relationship between the best polynomial for a given imaging device and the imaging device characteristics, a common solution is to use full-order polynomials or to determine it by a trial-and-error process.<sup>10,12</sup>

We automatically determine the best polynomial for a given imaging device using genetic programming (GP).<sup>15,16</sup>

GP is a domain-independent evolutionary method that genetically breeds a population of functions, or more generally, computer programs to solve a problem. For a brief introduction of GP, see Appendix A; for a more detailed presentation, see for instance Refs. 15, 16, and 22.

Chosen the polynomial order  $m$  and the maximum number  $n$  of polynomial terms, the polynomials evolved by GP have been built using the set of functionals (or nonterminal) symbols  $F=\{\text{Join}\}$  and the set of terminal symbols:  $T=\{R^i G^j B^k \mid 0 \leq i+j+k \leq m\}$ . Given two terminal symbols  $t_1 \in T$  and  $t_2 \in T$ , the Join function concatenates them in a new list, i.e.:

$$\text{Join}(t_1, t_2) = [t_1 \ t_2];$$

thus, the expressions evolved by GP may be seen as a list of polynomial terms, each one of the form  $R^i G^j B^k$  with  $0 \leq i+j+k \leq m$ . In our experiments, we have set the polynomial order  $m$  to 4 and 7.

GP individuals, corresponding to different polynomials, have been evaluated using a fitness function composed of the measure of the  $\Delta E_{94}$  color error on a chosen training set, weighted by the value of the leave-one-out cross-validation (LOOCV).<sup>23</sup> The LOOCV uses a single observation from the original sample as validation data, and the remaining observations as training data. This process is repeated to allow each observation in the sample to be used exactly once as validation data. The use of the LOOCV permits us to understand if the polynomial chosen has good generalization capability.

## 5 Experimental Results

In this section we present the results of our characterization procedures and compare them with those presented by Shen, Xin, and Mou<sup>3,4,12</sup> using their own acquisition data: the 198 patches of the Macbeth ColorChecker DC (MDC), the 144 patches of the Kodak Q60 photographic standard (IT8), and the 20 patches of the Kodak Gray Scale Q-14. The spectral reflectance values of MDC and Q14 were measured by Shen *et al.* using a GretagMacbeth Spectrophotometer 7000A, and those of IT8 were measured using a GretagMacbeth Spectrolino spectrophotometer. The CIEXYZ and CIELAB values under the CIE D65 standard illuminant were then calculated from these reflectance data for scanner characterization. The three color targets have been scanned using an Epson GT-10000+. During the scanning process, all the color adjustment functions of the scanner have been disabled. The target Q14 is used to calculate the inverse optoelectronic conversion function  $\mathcal{F}$  in Eq. (3). The targets MDC and IT8 are used for the colorimetric characterization of the scanner.

In the first part of our experiments (Sec. 5.1), we test the performance of the seven optimizing procedures described in Secs. 2 and 3 using the full third-order polynomial suggested for the considered device by Shen, Mou, and Xin:<sup>12</sup>

$$\begin{aligned} \text{full } 3d = & 1 + R + G + B + R^2 + RG + RB + G^2 + GB + B^2 \\ & + R^3 + R^2G + R^2B + G^3 + RG^2 + G^2B + B^3 + RB^2 \\ & + GB^2 + RGB. \end{aligned}$$

Then, in the second part of the experiments (Sec. 5.2), we try to further improve the characterization performance of the optimizing procedures, generating new polynomials by means of GP.

For all the experiments, we study both the performance during the training phase and the generalization capability. Following Shen *et al.*, we have partitioned both the Macbeth ColorChecker DC and the IT8 datasets into a training set and a test set using the following algorithm.

```

Initialize an empty training set and an empty test set
(let their names be Train and Test, respectively, and let
 $N_c$  be the total number of patches in the whole dataset);
 $i := 1$ ;
repeat
    Insert the  $i$ 'th and the  $(i+1)$ 'th patches of the dataset
    into Train;
    Insert the  $(i+2)$ 'th patch of the dataset into Test;
     $i = i + 3$ ;
until  $i \geq N_c - 2$ .

```

In this way, training and test sets are interleaved parts of the same dataset, and the training set is approximately two times larger than the test set. For all the patches of the training and test sets, we calculate the  $\Delta E_{94}$  color error between the predicted and measured CIELAB values. We report the mean, maximum, and standard deviation for the  $\Delta E_{94}$  color error for each of the minimization procedures considered, both in the case of *full third* and GP-generated polynomials.

As we are interested in comparing performances of the considered procedures, together with single summary statistics, it is worthwhile to compare the whole error distribu-

tion of the methods. Since the underlying error distribution cannot be well modeled by a standard distribution, we need a test that does not require any assumptions about it. An appropriate test in this case is the Wilcoxon sign test,<sup>24,25</sup> as it is a nonparametric test for assessing whether two samples of observations come from the same distribution. The null hypothesis is that the two samples are drawn from a single population, and therefore that their probability distributions are equal. It requires the two samples to be independent, and the observations to be ordinal or continuous measurements, i.e., one can at least say which one of any two observations has the larger value. In a less general formulation, the Wilcoxon-Mann-Whitney two-sample test may be thought of as testing the null hypothesis that the probability of an observation from one population exceeding an observation from the second population is equal to 0.5. Formally, let  $X$  and  $Y$  be random variables representing the  $\Delta E_{94}$  colorimetric error of the methods  $X$  and  $Y$ . The Wilcoxon test is used to test the hypothesis that the random variables  $X$  and  $Y$  are such that  $p = P(X > Y) = 0.5$ . We test the null hypothesis  $H_0: p = 0.5$  (i.e., we hypothesize that the methods  $X$  and  $Y$  have the same performance) against the alternative hypothesis  $H_1: p < 0.5$  (which if true implies that colorimetric errors for method  $X$  are lower than those for method  $Y$ ). To test the hypothesis  $H_0$ , we consider independent pairs  $(X_1, Y_1), \dots, (X_n, Y_n)$  of errors for  $N_p$  different patches. We denote by  $W$  the number of patches for which  $X_i > Y_i$ . When  $H_0$  is true,  $W$  is binomially distributed  $[b(N_p, 0.5)]$  and the Wilcoxon test is based on this statistic. We accept or reject the null hypothesis at the significance level  $\alpha = 0.05$ . For a better comprehension, we have summarized the result of the Wilcoxon sign test in a score: this score is the number of times that we have rejected the null hypothesis for the considered procedure, i.e., the number of procedures in respect to which the results of the procedure being considered are statistically better. This permits us to obtain an overall rank among the methods despite the fluctuations in the single error statistics adopted (mean, maximum, and standard deviation).

### 5.1 Full Third-Order Polynomial

In this part of the experiments, the procedures considered are applied to the full third-order polynomial. The four literature procedures that we have considered (LS, TLS, TCDM, and LAB-LS) do not need any particular parameter setting (except for the  $p$ 'th root of the LAB-LS chosen, as indicated in Sec. 2). The GAs procedures (LAB-GA, LAB-GAp, and LAB-GApw) instead need the setting of some parameters, the meaning of which can be found in Appendix A or in Refs. 13 and 14. The results that we present in this section have been obtained using the following set of parameters:

- population size  $N = 1000$
- size of the individuals = 25
- maximum number of generations = 100,000
- selection type: tournament selection, with tournament size = 100
- crossover rate  $p_c = 0.8$
- mutation rate  $p_m = 0.2$
- presence of elitism.

**Table 2** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score (on the test set) of the seven procedures considered using the full third polynomial on the IT8 dataset.

Method	$\Delta E_{94}$ training			$\Delta E_{94}$ test			$\Delta E_{94}$ total			WST score
	Mean	Max	Std Dev	Mean	Max	Std Dev	Mean	Max	Std Dev	
LS	1.25	6.74	1.07	1.45	7.78	1.25	1.31	7.78	1.14	0
TLS	1.40	6.75	1.25	1.59	7.69	1.37	1.47	7.69	1.30	0
TCDM	1.18	5.60	0.93	1.43	6.48	1.12	1.27	6.48	1.02	2
LAB-LS	0.85	2.64	0.49	1.13	3.68	0.67	0.96	3.68	0.57	3
LAB-GA	0.79	1.81	0.42	1.05	2.48	0.55	0.87	2.48	0.47	4
LAB-GAp	0.79	1.72	0.42	1.04	2.48	0.56	0.88	2.48	0.47	5
LAB-GApw	0.83	1.57	0.35	1.03	2.54	0.52	0.90	2.54	0.41	5

The mutation rate has usually a large impact on time to converge to a solution and on the quality of that solution. The parameter value adopted here is the one that has returned the best results both on the training and tests sets used in this work. The genetic operators we have used are the standard one-point crossover and point mutation defined in Refs. 13 and 14. The results of the GAs that we report in this work are the best ones obtained over five independent runs for each GA version. The choice of performing a limited number of runs (five) for a large number of generations (100,000) is justified in Ref. 26, where the authors show that under a constant cost constraint, executing a small number of large runs can reach better solutions faster than executing many small runs.

Table 1 shows the results returned by the considered procedures for Macbeth ColorChecker DC with the full third polynomial. Column one identifies the methods; columns 2, 3, and 4 report the results on the training set; columns 5, 6, and 7 report the results on the test set; while columns 8, 9, and 10 report the results of the models learned on the training set executed on both the training and test sets together. Column 11 reports the Wilcoxon sign test score on the test set, which is representative of the number of the procedures in respect to which the procedure-considered results are statistically better.

As Table 1 shows, our three functions, optimized by the GAs, produce lower error statistics with respect to all the other methods both on the training and test sets. In particular, LAB-GApw outperforms all the state of the art procedures (first four rows of Table 1) both on the training and test sets for all the error measures that we have studied (even though it is outperformed by the other two GAs versions for the mean error). Furthermore, the standard deviation of the error of LAB-GApw is smaller than those of all the other methods.

In Table 2 we report the results on IT8 (the columns of this table have to be interpreted as those of Table 1). Also in this experiment, our functions perform better than the other methods.

## 5.2 Generating New Polynomials by Genetic Programming

The second phase of our study consists in defining new polynomials that can improve the characterization performance, compared with those obtained using the *full third* polynomial.

The results presented have been obtained using the fitness function defined in Sec. 4, i.e.:

$$\text{fitness} = \text{LOOCV} \cdot \text{mean}(\Delta E_{94}),$$

and using the following set of parameters:

- functional (or nonterminal) symbols  $F = \{\text{Join}\}$  (see Sec. 4)
- terminal symbols  $T = \{R^i G^j B^k \mid 0 \leq i + j + k \leq m\}$
- population size  $N = 200$
- maximum number of tree nodes  $= 2n - 1$  (where  $n$  is the maximum number of polynomial terms)
- maximum number of generations  $= 20,000$
- algorithm used to initialize the population: ramped half-and-half
- selection algorithm: tournament, with tournament size  $= 50$
- crossover rate  $p_c = 0.5$
- the mutation rate  $p_m = 0.5$
- presence of elitism.

Also in this case, the mutation rate used here is the one that has allowed us to find the best results both in the training and test sets used in this work.

First of all, we look for new polynomials with similar characteristics to the *full third* polynomial: thus we look for a fourth degree polynomial ( $m=4$ ), composed of a maximum of 20 terms ( $n=20$ , the same number of terms of the *full third*). The fitness of the polynomials evolved by GP has been set equal to the error obtained by LAB-LS, since it is faster than the other techniques. The best polynomial obtained is:

**Table 3** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score (on the test set) of the four procedures considered using the best fourth order polynomial found by GP (with a maximum number of terms equal to 20). The dataset is the Macbeth ColorChecker DC.

Method	$\Delta E_{94}$ training			$\Delta E_{94}$ test			$\Delta E_{94}$ total			WST score
	Mean	Max	Std Dev	Mean	Max	Std Dev	Mean	Max	Std Dev	
LAB-LS	1.18	6.86	1.05	1.13	3.06	0.73	1.16	6.86	0.96	0
LAB-GA	1.15	7.78	1.06	1.08	3.57	0.74	1.12	7.78	0.97	0
LAB-GAp	1.14	7.69	1.03	1.07	3.60	0.75	1.11	7.69	0.96	0
LAB-GApw	1.18	5.85	1.04	1.12	2.81	0.71	1.15	5.85	0.94	3

$$R + G + B + RG + RB + GB + R^2 + G^2 + R^3 + RG^2 + G^2B + B^3 + R^4 + R^2G^2 + R^2B^2 + RG^3 + RG^2B + G^4 + G^2B^2 + GB^3.$$

Tables 3 and 4 report the results obtained using the three procedures proposed (LAB-GA, LAB-GAp, and LAB-GApw) on the Macbeth ColorChecker DC and on IT8, respectively, using this fourth order polynomial. These procedures are also compared on the same polynomial with LAB-LS, which is the best characterization procedure that we have found in the literature for these color targets. As for the prior tables, column one identifies the method; columns 2, 3, and 4 report the results on the training set; columns 5, 6, and 7 report the results on the test set; and columns 8, 9, and 10 report the results on the training and test set together. Column 11 reports the Wilcoxon sign test (WST) score on the test set, which is representative of the number of the methods in respect to which the method-considered results are statistically better.

If we compare these results with those of Tables 1 and 2, we can observe that using the polynomial found by GP allows all the methods to find a better mean error with smaller standard deviations both on the training and test sets for the Macbeth ColorChecker DC. For the IT8 dataset, the mean error that we have found using the polynomial returned by GP has always been smaller than the one that we have found using the *full third* on the training set. On

the test set of the IT8 dataset, LAB-GAp has returned the same mean error as in the case when the *full third* has been used, while all the other methods have returned a smaller mean error. The GAs versions still perform generally better than LAB-LS, and LAB-GApw seems to be globally the best method.

Once these results have been obtained, the next step in our study consisted in relaxing the polynomial constraints. In other words, we have used GP to allow a higher degree for the polynomial (i.e., 7) and no constraint on the maximum number of terms, except that it does not exceed the cardinality of the training set minus one.

The best seventh order polynomial obtained has 28 terms and it is reported as:

$$1 + R^2 + GB + R^3 + RGB + R^4 + RG^2B + RB^3 + G^4 + G^3B + R^4G + R^2B^3 + G^4B + G^2B^3 + B^5 + R^4G^2 + R^2G^4 + R^2G^2B^2 + RB^5 + G^4B^2 + G^3B^3 + G^2B^4 + B^6 + R^4G^3 + G^4B^3 + G^3B^4 + G^2B^5 + B^7.$$

Once again, the fitness of the polynomials evolved by GP has been set equal to the error obtained by LAB-LS, since it is faster than the other techniques.

In Tables 5 and 6 we report the results of the methods studied on the Macbeth ColorChecker DC and on the IT8 respectively, using the new polynomial found by GP.

**Table 4** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score (on the test set) of the four procedures considered using the best fourth order polynomial found by GP (with a maximum number of terms equal to 20). The dataset is the IT8.

Method	$\Delta E_{94}$ training			$\Delta E_{94}$ test			$\Delta E_{94}$ total			WST score
	Mean	Max	Std Dev	Mean	Max	Std Dev	Mean	Max	Std Dev	
LAB-LS	0.79	1.66	0.36	1.06	2.39	0.52	0.87	2.39	0.41	0
LAB-GA	0.75	1.67	0.32	1.04	2.39	0.53	0.85	2.39	0.39	0
LAB-GAp	0.74	1.65	0.33	1.04	2.41	0.52	0.85	2.41	0.39	0
LAB-GApw	0.78	1.55	0.33	1.02	2.34	0.50	0.86	2.34	0.38	3

**Table 5** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score (on the test set) of the four procedures considered using the best seventh order polynomial found by GP. The dataset is the Macbeth ColorChecker DC.

Method	$\Delta E_{94}$ training			$\Delta E_{94}$ test			$\Delta E_{94}$ total			WST score
	Mean	Max	Std Dev	Mean	Max	Std Dev	Mean	Max	Std Dev	
LAB-LS	1.09	5.96	1.01	0.98	2.78	0.70	1.05	5.96	0.88	0
LAB-GA	1.04	6.57	1.02	0.93	3.42	0.72	1.00	6.57	0.89	2
LAB-GAp	1.02	6.63	1.05	0.91	3.46	0.74	0.98	6.63	0.91	0
LAB-GApw	1.06	5.42	0.98	0.95	2.11	0.67	1.02	5.42	0.85	3

Comparing these results with the ones of Tables 3 and 4, we observe that when using this new polynomial, all the methods return a smaller mean error both on the training and test sets. Furthermore, if we compare these results to the ones initially obtained using the *full third* polynomial (Tables 1 and 2), we observe that on the Macbeth ColorChecker DC test set, the mean error decreases by 28% (from 1.18 to 0.85), the maximum error decreases by 22% (from 2.84 to 2.21), and the standard deviation decreases by 15% (from 0.79 to 0.67). For what concerns the IT8 test set, we observe that the mean error decreases by 9% (from 1.03 to 0.94), the maximum error decreases by 12% (from 2.54 to 2.23), and the standard deviation decreases by 23% (from 0.52 to 0.40).

To further validate our characterization approach, we tested two additional scanners (Agfa Duoscan and HP Scanjet G3010) obtaining similar experimental behavior. The experimental results are summarized and reported in Appendix B.

Finally, in Table 7, we report the  $p$ 'th root values used in our experiments for all the methods and polynomials considered.

### 5.3 Behavior with Respect to the Neutral Axis

The higher order polynomials sometimes break the monotonic conversion curves, and this might be a serious drawback of the polynomial technique. We report the RGB to XYZ transformation curves for the neutral axis ( $R=G=B$ ),

respectively, for the third order polynomial [Fig. 1(a)], the fourth order polynomial [Fig. 1(b)], and for the seventh order polynomial [Fig. 1(c)].

We note that for all the polynomials used, the monotonic behavior of the conversion curves is preserved. We can also observe a nonlinear behavior for very low gray values. However, this nonlinearity is not severe because these out-of-gamut values are clipped.

## 6 Conclusions and Future Work

We address the problem of scanner characterization using multidimensional polynomials. We show that genetic algorithms are a class of optimization methods well suited for the colorimetric characterization problem, since they can minimize functions that take into account the mean perceptual colorimetric error, together with other error statistics. Furthermore, we show how the search for the best characterization polynomial can be automatized using genetic programming. Experimental results on different devices have shown that our functions, minimized by the GAs, globally outperform the state of the art methods. Moreover, relaxing the constraints on the polynomial degree and number of terms, and modeling the best polynomial with genetic programming, can be further improved.

Currently, we are trying to modify our genetic algorithms to add some constraints to our functions. In particular, we would like to preserve the neutral axis to avoid possible color artifacts due to nonoptimal RGB to XYZ

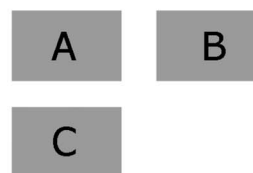
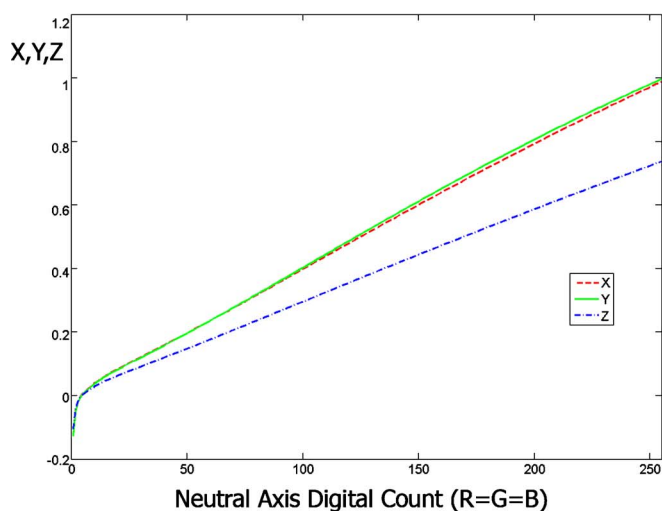
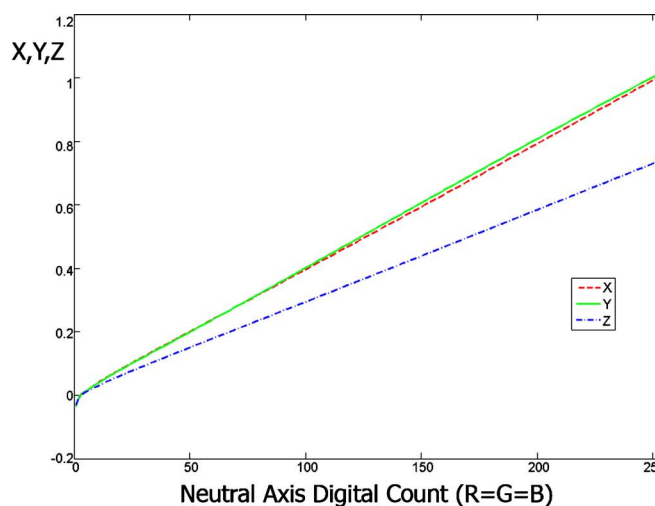
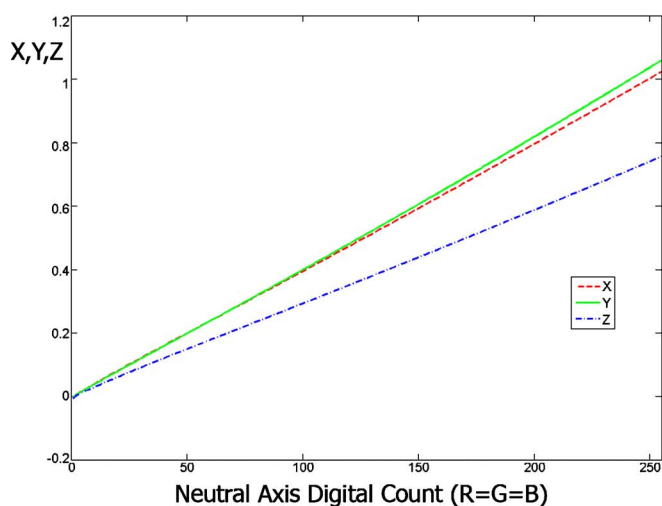
**Table 6** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score (on the test set) of the four methods considered using the best seventh order polynomial found by GP. The dataset is the IT8.

Method	$\Delta E_{94}$ training			$\Delta E_{94}$ test			$\Delta E_{94}$ total			WST score
	Mean	Max	Std Dev	Mean	Max	Std Dev	Mean	Max	Std Dev	
LAB-LS	0.69	1.61	0.36	0.95	2.36	0.45	0.77	2.36	0.39	0
LAB-GA	0.66	1.68	0.36	0.92	2.51	0.51	0.75	2.51	0.41	0
LAB-GAp	0.64	1.70	0.37	0.89	2.53	0.49	0.72	2.53	0.41	2
LAB-GApw	0.68	1.38	0.33	0.94	2.23	0.40	0.76	2.23	0.35	3



**Table 7**  $p$ 'th root values used in our experiments for all the methods and polynomials considered.

Method	Polynomial		
	Third order	Fourth order	Seventh order
LS	NA	NA	NA
TLS	NA	NA	NA
TCDM	NA	NA	NA
LAB-LS	9	12	21
LAB-GA	9	12	21
LAB-GAp	9.82	12.51	21.33
LAB-GApw	9.78	12.40	21.19



**Fig. 1** RGB to XYZ transformation curves for the neutral axis ( $R=G=B$ ): (a) The third order polynomial; (b) the fourth order polynomial, and (c) the seventh order polynomial. X curve is plotted with a red line, Y with a green one, and Z with a blue one.

transformation. We are also trying to improve the rendering of some specific color classes (such as skin tone).

## Appendix A: Evolutionary Algorithms

Evolutionary algorithms (EAs) are a broad class of stochastic optimization algorithms. They maintain a *population* of candidate solutions (or *individuals*) for the problem at hand, and make it evolve by iteratively applying a set of stochastic operators, which are usually *selection*, *mutation*, and *recombination* (or *crossover*). Selection replicates the most successful solutions found in a population at a rate bound to their relative quality, measured by a user-defined function called a *fitness function*. Mutation and recombination transform selected solutions into new solutions, providing exploration of the search space. In particular, mutation randomly perturbs a candidate solution; recombination decomposes two distinct solutions and then randomly mixes their parts to form novel solutions. An iteration of this algorithm is usually called a *generation*. The initial population may be either a random sample of the solution space or may be seeded with solutions found by simple local search procedures or by some *a priori* information about the problem, if these are available. We have used two kinds of EAs: genetic algorithms and genetic programming, respectively, described as follows.

### Genetic Algorithms

Genetic algorithms (GAs)<sup>13,14</sup> are the oldest and most known kind of EA. Their peculiarity is that potential solutions that undergo evolution are represented as fixed length strings of characters or numbers. The iterative process of GAs can be summarized by the following pseudocode:

- Generate a population  $P$  composed of an even number  $N$  of individuals.
- Generation := 0.
- Repeat until a termination condition is satisfied:
  - calculate the fitness of all the individuals in population  $P$
  - create a new empty population  $P$
  - repeat until population  $P'$  is composed of exactly  $N$  individuals:
    - Select two individuals  $i_1$  and  $i_2$  from population  $P$  using the chosen selection algorithm.
    - Perform the crossover between  $i_1$  and  $i_2$  with probability  $p_c$ , and let  $j_1$  and  $j_2$  be the offspring (if crossover is not applied, let  $j_1 = i_1$  and  $j_2 = i_2$ ).
    - Mutate each character of  $j_1$  and  $j_2$  with a certain probability  $p_m$ , and let  $k_1$  and  $k_2$  be the offspring.
    - Insert  $k_1$  and  $k_2$  into population  $P'$ .
  - Perform the copy:  $P := P'$  and delete  $P'$ .
  - Generation := generation + 1.

Examples of termination conditions are: a predetermined number of generations or time has elapsed, a satisfactory solution has been found, or no improvement in solution quality has been taking place for a predetermined number of generations. In some cases, another genetic operator is added to crossover and mutation: *elitism*, i.e., the copy of the best individual unchanged into the newly generated

population at each generation. In that case, either an odd number of individuals  $N$  is chosen or another individual in  $P$  is selected, mutated, and inserted into  $P'$  at each generation.

The parameters the user has to set before running this process are:

- the population size  $N$
- the size of the individuals (i.e., number of digits of each potential solution)
- the maximum number of generations
- the selection algorithm
- the crossover rate  $p_c$
- the mutation rate  $p_m$
- presence or absence of elitism.

No formal method to properly set these parameters exists, only some heuristics. Nevertheless, independently from the setting of the parameters used, the probability that a GA with elitism finds a globally optimal solution at time step  $t$  asymptotically converges to 1 when  $t$  tends to infinity.<sup>27</sup>

### Genetic Programming

The major difference between genetic programming (GP)<sup>15,16</sup> and the other EAs is that potential solutions to be evolved are generally speaking *computer programs*. They can be represented as trees, lines of code, expressions in prefix or postfix notations, strings of variable length, etc. We use the representation first introduced in Ref. 15: potential solutions are represented as *LISP-like* tree structures built using a set of *terminal* symbols and a set of *nonterminal* or *functional* symbols. The iterative process of GP is similar to the one of GAs presented in the previous section, with the difference that crossover and mutation are redefined to act on trees instead of strings.<sup>15</sup> Before running GP, the user has to set a larger number of parameters than for GAs, mainly due to the variable size of GP individuals. These parameters are as follows.

- The sets of functional (or nonterminal) and terminal symbols that are used to build the potential solutions.
- The population size.
- The maximum size of the individuals (typically expressed as the maximum number of tree nodes or the maximum tree depth).
- The maximum number of generations.
- The algorithm used to initialize the population (the algorithm used to initialize a GA population is usually simple: each digit composing each individual is generated randomly with uniform probability; in GP a population of random computer programs has to be generated. A set of algorithms to accomplish this goal can be found in Refs. 15 and 16).
- The selection algorithm.
- The crossover rate.
- The mutation rate.
- Presence or absence of elitism.

**Table 8** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score of the four procedures considered using the full third polynomial for the Agfa Duoscan.

Method	Macbeth ColorChecker DC				IT8			
	$\Delta E_{94}$ test			WST Score	$\Delta E_{94}$ test			WST Score
	Mean	Max	Std Dev		Mean	Max	Std Dev	
LAB-LS	1.39	3.25	1.15	0	0.85	2.78	0.49	0
LAB-GA	1.38	3.46	1.15	1	0.85	2.79	0.51	0
LAB-GAp	1.35	3.49	1.15	1	0.81	2.80	0.50	0
LAB-GApw	1.37	3.05	1.13	3	0.83	2.58	0.49	3

**Appendix B: Additional Results**

The devices investigated are the Agfa Duoscan and the HP Scanjet G3010. The experiments are performed in the same way as described in Sec. 5. For the Agfa Duoscan, the best fourth order polynomial found is:

$$1 + B^2 + B^3 + G + G^2 + G^2B + G^3 + G^4 + R^1B^3 + RG + RGB + RG^2B + RG^3 + R^2 + R^2G + R^2GB + R^2G^2 + R^3 + R^3G + R^4,$$

and the best seventh order polynomial found is:

$$B + B^3 + B^5 + G + G^2 + G^3 + G^3B^4 + G^4B^3 + G^5B^2 + R^1B^5 + RGB^2 + RGB^4 + RG^2B + RG^2B^2 + RG^3B + R^2 + R^2B + R^2B^3 + R^2GB + R^2GB^4 + R^2G^3 + R^2G^5 + R^3 + R^3B + R^3B^3 + R^3B^4 + R^4B^3 + R^4G^2 + R^4G^3 + R^5 + R^5B + R^6 + R^7.$$

For the HP Scanjet G3010 the best fourth order polynomial found is:

$$B^2 + B^4 + GB + GB^3 + G^2B + G^2B^2 + G^3B + R + RB + RGB + RGB^2 + RG^2B + R^2B + R^2B^2 + R^2G + R^2GB + R^2G^2 + R^3B + R^3G + R^4,$$

and the best seventh order polynomial found is:

$$1 + B^5 + B^6 + GB + B^3 + GB^5 + GB^6 + G^2 + G^2B^3 + G^3B^2 + G^4B + G^6B + G^7 + RGB^3 + RGB^4 + RG^2 + RG^3B^3 + RG^4 + RG^4B^2 + RG^6 + R^2 + R^2B + R^2GB^3 + R^2G^2B^3 + R^2G^3B^2 + R^3 + R^3B^2 + R^4B^2 + R^4G^1B^2 + R^6G^1 + R^7.$$

Experimental results are summarized in Tables 8–13.

**Acknowledgments**

The authors are grateful to Hui-Liang Shen of the Department of Information and Electronic Engineering of the Zhejiang University for providing the data used in the experiments that have permitted a fair comparison with some of the state of the art methods.

**Table 9** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score of the four procedures considered using the best fourth order polynomial found by GP (with a maximum number of terms equal to 20) for the Agfa Duoscan.

Method	Macbeth ColorChecker DC				IT8			
	$\Delta E_{94}$ test			WST Score	$\Delta E_{94}$ test			WST Score
	Mean	Max	Std Dev		Mean	Max	Std Dev	
LAB-LS	1.35	3.00	1.14	0	0.83	2.75	0.47	0
LAB-GA	1.33	3.05	1.14	0	0.81	2.75	0.47	0
LAB-GAp	1.33	3.07	1.13	0	0.81	2.76	0.47	0
LAB-GApw	1.33	2.84	1.12	3	0.82	2.55	0.45	3

**Table 10** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score of the four methods considered using the best seventh order polynomial found by GP for the Agfa Duoscan.

Method	Macbeth ColorChecker DC				IT8			
	$\Delta E_{94}$ test			WST Score	$\Delta E_{94}$ test			WST Score
	Mean	Max	Std Dev		Mean	Max	Std Dev	
LAB-LS	1.14	2.95	1.13	0	0.61	2.73	0.46	0
LAB-GA	1.13	3.00	1.14	0	0.59	2.73	0.47	0
LAB-GAp	1.12	3.01	1.14	0	0.58	2.73	0.46	0
LAB-GApw	1.12	2.72	1.10	3	0.59	2.53	0.45	3

**Table 11** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score of the four procedures considered using the full third polynomial for the HP Scanjet G3010.

Method	Macbeth ColorChecker DC				IT8			
	$\Delta E_{94}$ test			WST Score	$\Delta E_{94}$ test			WST Score
	Mean	Max	Std Dev		Mean	Max	Std Dev	
LAB-LS	2.38	5.12	1.30	0	1.10	4.45	0.72	0
LAB-GA	2.37	5.24	1.30	2	1.11	4.48	0.72	0
LAB-GAp	2.35	5.25	1.31	0	1.06	4.48	0.73	0
LAB-GApw	2.36	4.90	1.28	3	1.08	4.06	0.70	3

**Table 12** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score of the four procedures considered using the best fourth order polynomial found by GP (with a maximum number of terms equal to 20) for the HP Scanjet G3010.

Method	Macbeth ColorChecker DC				IT8			
	$\Delta E_{94}$ test			WST Score	$\Delta E_{94}$ test			WST Score
	Mean	Max	Std Dev		Mean	Max	Std Dev	
LAB-LS	2.34	5.06	1.30	0	1.03	4.31	0.72	0
LAB-GA	2.30	5.15	1.29	0	1.00	4.32	0.72	0
LAB-GAp	2.29	5.16	1.28	0	1.01	4.32	0.73	0
LAB-GApw	2.30	4.83	1.26	3	1.01	4.01	0.70	3

**Table 13** Mean, maximum, and standard deviation for  $\Delta E_{94}$  errors and Wilcoxon sign test score of the four methods considered using the best seventh order polynomial found by GP for the HP Scanjet G3010.

Method	Macbeth ColorChecker DC				IT8			
	$\Delta E_{94}$ test				$\Delta E_{94}$ test			
	Mean	Max	Std Dev	WST Score	Mean	Max	Std Dev	WST Score
LAB-LS	2.04	4.78	1.29	2	0.88	4.36	0.67	0
LAB-GA	2.02	4.81	1.27	0	0.86	4.37	0.68	0
LAB-Gap	2.01	4.82	1.26	0	0.86	4.38	0.68	0
LAB-Gapw	2.02	4.56	1.24	3	0.87	3.98	0.66	3

## References

- V. Cheung, S. Westland, C. Li, J. Hardeberg, and D. Connah, "Characterization of trichromatic color cameras by using a new multispectral imaging technique," *J. Opt. Soc. Am. A* **22**, 1231–1240 (2005).
- R. S. Berns and M. J. Shyu, "Colorimetric characterization of a desktop drum scanner using a spectral model," *J. Electron. Imaging* **4**(4), 360–372 (1995).
- H. L. Shen, J. H. Xin, "Spectral characterization of a color scanner by adaptive estimation," *J. Opt. Soc. Am. A* **21**(7), 1125–1130 (2004).
- H. L. Shen and J. H. Xin, "Colorimetric and spectral characterization of a color scanner using local statistics," *J. Imaging Sci. Technol.* **48**(4), 342–346 (2004).
- G. Sharma, *Digital Color Imaging Handbook*, CRC Press, Boca Raton, FL (2003).
- H. R. Kang and P. G. Anderson, "Neural network application to color scanner and printer calibrations," *J. Electron. Imaging* **1**(2), 125–135 (1992).
- R. Schettini, B. Barolo, and E. Boldrin, "Colorimetric calibration of color scanners by back-propagation," *Pattern Recogn. Lett.* **16**(10), 1051–1056 (1995).
- M. J. Vihel and H. J. Trussell, "Color scanner calibration via a neural networks," *Proc. IEEE Intl. Conf. Acoustics, Speech Signal Process.* **6**, 3465–3468 (1999).
- T. L. V. Cheung, S. Westland, D. R. Connah, and C. Ripamonti, "Characterization of colour cameras using neural networks and polynomial transforms," *J. Coloration Technol.* **120**(1), 19–25 (2004).
- H. R. Kang, *Computational Color Technology*, Vol. PM159, SPIE Press, Bellingham, WA (2006).
- S. V. Huffel and J. Vandewalle, "The total least squares problem: computational aspects and analysis," *Soc. Industrial Applied Mathematics*, Philadelphia (1991).
- H. L. Shen, T. S. Mou, and J. H. Xin, "Colorimetric characterization of scanners by measures of perceptual color error," *J. Electron. Imaging* **15**(4), 1–5 (2006).
- D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, New York (1989).
- J. H. Holland, *Adaptation in Natural and Artificial Systems* The Univ. of Michigan Press, Ann Arbor, MI (1975).
- J. R. Koza, *Genetic Programming*, The MIT Press, Cambridge, MA (1992).
- L. Vanneschi, "Theory and practice for efficient genetic programming," Ph.D. Thesis, Faculty of Science, Univ. of Lausanne, Switzerland (2004). Downloadable version at <http://personal.disco.unimib.it/Vanneschi>.
- J. A. Nelder and R. Mead, "A simplex method for function minimization," *Comput. J.* **7**(4), 308–313 (1965).
- S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, Chap. 4, Prentice-Hall, Englewood Cliffs, NJ (1995).
- S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science* **220**, 4598 (1983).
- F. Glover, "Tabu search-part i," *ORSA J. Comput.*, **1**(3), 190–206 (1989).
- F. Glover, "Tabu search-part ii," *ORSA J. Comput.* **2**(1), 4–32 (1990).
- J. Koza and R. Poli, "A genetic programming tutorial," in *Introductory Tutorials in Optimization, Search and Decision Support*, E. Burke, Ed., Chap. 8, see <http://www.genetic-programming.com/jkpdf/burke2003tutorial.pdf>. (2003).
- R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," *Proc. 14 Intl. Joint Con. Arti. Intell.*, pp. 1137–1143, (1995).
- R. V. Hogg and E. A. Tanis, *Probability and statistical Inference*, Prentice Hall, Englewood Cliffs, NJ (2001).
- S. D. Hordley and G. D. Finlayson, "Re-evaluating color constancy algorithms," in *Proc. 17th Intl. Conf. Patt. Recog.*, pp. 76–79 (2004).
- E. Cantu-Paz and D. E. Goldberg, "Are multiple runs of genetic algorithms better than one?" in *Proc. Genetic and Evolutionary Computation Conference, GECCO 2003, Proc. Part 1, Lect. Notes Comput. Sci.* **2723**, 801–812 (2003).
- G. Rudolph, "Convergence analysis of canonical genetic algorithms," *IEEE Trans. Neural Netw.* **5**(1), 96–101 (1994).

Biographies and photographs of the authors not available.