Pareto-Optimal Formulations for Cost versus Colorimetric Accuracy Trade-Offs in Printer Color Management

D. J. LITTLEWOOD
University of Colorado
P. A. DRAKOPOULOS
IBM Corporation
and
G. SUBBARAYAN
University of Colorado

Color management for the printing of digital images is a challenging task, due primarily to nonlinear ink-mixing behavior and the presence of redundant solutions for print devices with more than three inks. Algorithms for the conversion of image data to printer-specific format are typically designed to achieve a single predetermined rendering intent, such as colorimetric accuracy. In the present paper we present two CIELAB to CMYK color conversion schemes based on a general Pareto-optimal formulation for printer color management. The schemes operate using a 149-color characterization data set selected to efficiently capture the entire CMYK gamut. The first scheme uses artificial neural networks as transfer functions between the CIELAB and CMYK spaces. The second scheme is based on a reformulation of tetrahedral interpolation as an optimization problem. Characterization data are divided into tetrahedra for the interpolation-based approach using the program Qhull, which removes the common restriction that characterization data be well organized. Both schemes offer user control over trade-off problems such as cost versus reproduction accuracy, allowing for user-specified print objectives and the use of constraints such as maximum allowable ink and maximum allowable ΔE^*_{ab}. A formulation for minimization of ink is shown to be particularly favorable, integrating both clipping and gamut compression features into a single methodology. Codes developed as applications of these schemes were used to convert several CIELAB Tiff images to CMYK format, providing both qualitative and quantitative verification of the Pareto-optimal approach. Prints of the MacBeth ColorChecker^{tm} chart were accurate within approximately to 3 ΔE^*_{ab} for in-gamut colors. Modifications to this approach are presented that offer user control over grey component replacement and provide additional options for rendering intent.

Categories and Subject Descriptors: I.4.0 [**Image Processing and Computer Vision**]: General—*Image processing software*

General Terms: Algorithms, Theory, Performance, Experimentation

Additional Key Words and Phrases: Artificial Neural Networks, Color Conversion, Color Fidelity, Color Management, Color Matching, CMYK, Color Printing, Color Space Transformation, Optimization, Pareto-optimization, Tetrahedral Interpolation

This work was funded by IBM through the IBM University Partnership Project and by the National Science Foundation under contract number ECS 9734349.

Authors' addresses: David J. Littlewood, University of Colorado, Campus Box 427, Boulder, CO 80309-0427; e-mail: littlewo@sprocket.colorado.edu; Paul A. Drakopoulos, IBM Software Group, 590 Madison Avenue, 14th floor, New York, NY, 10022; e-mail: drako@us.ibm.com; Ganesh Subbarayan, University of Colorado, Campus Box 427, Boulder, CO 80309-0427; e-mail: ganesh@colorado.edu.

1. INTRODUCTION

A prevalent color conversion requirement is the transfer of an image displayed on a monitor to CMYK format for printing on a four-color print device. This conversion is commonly achieved through the use of a third, device-independent, color space. Typically, RGB values are transformed into a device-independent color space from which conversion into a device-dependent CMYK color space is performed [Stone et al. 1988]. This strategy is the basis for international standards for device specification, namely the International Color Consortium (ICC) profile [ICC 1998; Wallner 1998]. The choice of a device-independent color space for this study was based primarily on the need to quantify color differences. A device-independent color space well suited for this purpose is the CIE 1976 ($L^*a^*b^*$) color space [CIE 1986; Hunter and Harold 1987; Kasson and Plouffe 1992; Xu and Holub 1992]. The CIE 1976 ($L^*a^*b^*$) color space, abbreviated CIELAB, aims towards perceptual uniformity, the condition where equal distances in a color space correspond to approximately equal perceptual differences as seen by the human eye [Wyszecki and Stiles 1982]. The use of CIELAB is widespread, and CIELAB is included as a standard color space in ICC profiles [ICC 1998; Wallner 1998]. The difference between two color stimuli represented by points in the CIELAB color space is quantified by the Euclidean distance between the two points, ΔE^*_{ab} [CIE 1986].

The characterization of displays in terms of tristimulus values has been well studied (see, for example, Kasson and Plouffe [1992], Widdel and Post [1992], Cowan [1983; 1987], or Berns [1996]), and the calculation of CIELAB values from tristimulus values is simply a matter of applying colorimetric formulae [CIE 1986; Wyszecki and Stiles 1982; Hunt 1991]. The CIELAB to CMYK conversion of a monitor image, however, is nontrivial for several reasons. One difficultly arises from the fact that color gamuts of both monitors and print devices are restricted by the physical nature of the systems. Printer gamuts generally do not match monitor gamuts, leading to monitor images that cannot be accurately reproduced by a printer [Gordon et al. 1987; Stone et al. 1988; Gentile et al. 1990]. This problem is addressed by gamut mapping, the process by which the range of colors produced by a given monitor is projected onto a printer gamut [Gentile et al. 1990]. A second challenge is mathematical in origin, CIELAB to CMYK conversion is a 3-space to 4-space transformation having multiple solutions in general, and thus requires an additional constraint. This constraint is typically formulated in terms of grey component replacement [Yule 1967; Sayanagi 1987; SWOP 1988].

Transformation into a CMYK color space from CIELAB is commonly performed using a look-up table (LUT) and interpolation. Generally, interpolation inside a LUT is well suited for implementation due to its speed, but error prone due to its inability to model the nonlinearity of ink mixing. Interpolation errors decrease as the number of points in the LUT increases, thus LUTs are commonly built from a large characterization data set [Kang 1995b; Johnson 1996]. Construction of a LUT entails printing a set of color patches and obtaining colorimetric data by physical measurement of the device output [Johnson 1996]. These data are arranged in a nxnxn matrix, where n typically falls between 6 and 16. For detailed information regarding interpolation inside a look-up table (LUT), and the errors associated with it, the reader is referred to Nin et al. [1992], Hung [1993], Jennings et al. [1994], Kasson [1994], Kasson et al. [1995], Kang [1995a], or Kanamori [1999]. Due to its efficiency and accuracy relative to other interpolation schemes, tetrahedral interpolation is considered best suited for interpolation inside a LUT [Kasson et al. 1995].

Various formulae have also been used for conversion into printer color spaces, including regression models and models based on optics and ink mixing [Kang 1995b]. Models based on optics and ink mixing include the Neugebauer equations, the Yule-Nielsen model, the Clapper-Yule model, the Kubelka-Munk theory, and the Beer-Bouguer law [Neugebauer 1937; Yule and Neilsen 1951; Clapper and Yule 1953; 1955; Kubelka and Munk 1931; Kubelka 1948; Bouguer 1729; Beer 1852; Wyszecki and Stiles 1982].

A number of studies had applied these theories to modern printing devices [Heuberger et al. 1992; Kang 1993; Rolleston and Balasubramanian 1993; Kang 1994; Kim et al. 1997; Praefcke 1999]. In general, ink-mixing theory is insufficient to predict the complex interactions of ink and paper in four-color printing [Kang 1995b]. Regression models typically involve finding model parameters (such as polynomial coefficients) which minimize the difference between a numerical model and a set of characterization data [Vachon 1988; Mongeon 1996]. Of particular relevance to this study are nonlinear regression models based on artificial neural networks [Kang 1992; Marcu and Iwata 1993; Abe and Marcu 1994; Arai et al. 1993; Tominaga 1993, 1996; Drakopoulos 1997a; Tominaga 1998a, 1998b].

The effectiveness of color conversions can be measured in several ways. The cost of reproducing a given image and the colorimetric accuracy with which it is reproduced provide quantitative measures of quality. Judgments regarding out-of-gamut color conversion and the preservation of gradients must be made considering the overall appearance of the image. This must be done, at least in part, qualitatively, and not based simply on the accuracy with which individual tones are reproduced by the print device [Gentile et al. 1990]. At the present time, systematic approaches to characterizing printers and thereby converting images with arbitrary, user-defined rendering intents are largely missing in the literature.

In this paper we present an approach based on a Pareto-optimal formulation of the color conversion problem to achieve arbitrary, user-defined rendering intents [Littlewood 2001]. An important aspect of the proposed approach is a small set of input characterization data, together with carefully built artificial neural network models. The smallness of the data set is expected to enable efficient calibration. The remainder of Section 1 contains background material relating to grey component replacement and tetrahedral interpolation. Section 2 presents the mathematical foundation of the Pareto-optimal approach. Section 2 also introduces the artificial neural network and interpolation transfer functions used in this study to convert between the CIELAB and CMYK color spaces. Implementation details are covered in Section 3, including the selection of a characterization data set and development of the color management systems NeuralColor and OptInterpol. NeuralColor is based on the use of artificial neural networks as global models of the relationship between CMYK and CIELAB. OptInterpol operates on a generalized Pareto-optimal formulation of tetrahedral interpolation, and utilizes the code Qhull [Barber and Huhdanpaa 1998] as a means to construct tetrahedra from any arbitrary set of characterization data. Evaluation methodology and results are presented in Sections 4 and 5. Colorimetric data was acquired by printing the MacBeth ColorChecker^{tm} color rendition chart [McCamy et al. 1976]. Additional images were printed to allow for qualitative evaluation. Finally, a discussion of the results and conclusions drawn from this study are presented in Section 6. Section 6 includes a discussion of creating ICC profiles with the color management systems developed in this study, a practice which allows for real-time image conversion and integration of the Pareto-optimal approach with commercially available software packages on standard operating systems.

1.1 The Black Printer and Grey Component Replacement

The use of black ink in addition to chromatic inks is favorable in several respects [SWOP 1988; Yule 1967; Johnson 1996; Jung 1984; Holub et al. 1989; Birkenshaw et al. 1986]:

—To reduce overall colorant deposition.

—To substitute a relatively inexpensive black ink for a part of the more costly colored inks.

—To expand the printer gamut.

—To make the grey balance of the cyan, magenta, and yellow inks less critical.

—To produce denser blacks and better shadow detail than cyan, magenta, and yellow inks alone can produce.

These benefits have led to the commonality of four-color printing in the image reproduction industry.

The output of a four-color printer, expressed in terms of CIELAB values, is a function of 4 variables.

$$\begin{aligned} L^* &= L^*(C, M, Y, K) \\ a^* &= a^*(C, M, Y, K) \\ b^* &= b^*(C, M, Y, K) \end{aligned} \tag{1}$$

Equation (1) illustrates that four-color printing is an indeterminate problem of three equations with four unknowns. For any printed color with a grey component, that is, for any color that lies within the outer boundary of the printer gamut, infinite combinations of cyan, magenta, yellow, and black exist as solutions to Equation 1 [Yule 1967; Holub et al. 1989; Schwartz et al. 1985; Holub and Kearsley 1989]. The constraint required to determine a unique combination of the four inks is commonly expressed in terms of grey component replacement (GCR) [SWOP 1988; Holub et al. 1989; Holub and Kearsley 1989; Field 1986].

GCR can be well described in terms of three components, the black printer, under color removal (UCR), and under color addition (UCA) [Sayanagi 1987]. The black printer is simply the amount of black ink added to a CMY image. UCR refers to the reduction of cyan, magenta, and yellow inks (or any chromatic ink in general) in compensation for black ink added to the image [Yule 1967]. UCA refers to the addition of chromatic inks to a CMYK image to counteract undesirable consequences of UCR, such as the loss of density in dark regions of the print [SWOP 1988].

It should be noted that the above terminology implies that any CMYK image can be produced by combinations of cyan, yellow, and magenta inks. This is not strictly true, as the use of black in addition to cyan, magenta, and yellow increases the printer gamut [Holub et al. 1989; Birkenshaw et al. 1986]. Thus, the practice of starting with a CMY image and subsequently adding black while maintaining colorimetric consistency does not allow for realization of the full CMYK gamut.

1.2 Tetrahedral Interpolation

The implementation of a LUT requires an interpolation scheme to approximate the transfer function between the input space and the output space. A desirable interpolation method is both accurate and computationally inexpensive. Tetrahedral interpolation meets both of these criteria [Kasson and Plouffe 1992; Hung 1992; Kanamori and Kotera 1992].

A transformation function $F(x, y, z)$ can be approximated through tetrahedral interpolation using four neighboring points which surround a given input point. Figure 1 shows an input point falling inside a tetrahedron of arbitrary geometry. The transformation function $F(x, y, z)$ is approximated by the well known interpolation formula [Kasson and Plouffe 1992]

$$F(x, y, z) = N_1F_1 + N_2F_2 + N_3F_3 + N_4F_4 \tag{2}$$

where F_i are the known function values for vertices P_i, and N_i are volume coordinates which specify the unknown point's location in the tetrahedron. Formulae for the volume coordinates N_i for a given point are found as the solution of Equation Set (3).

$$\begin{aligned} x &= N_1x_1 + N_2x_2 + N_3x_3 + N_4x_4 \\ y &= N_1y_1 + N_2y_2 + N_3y_3 + N_4y_4 \\ z &= N_1z_1 + N_2z_2 + N_3z_3 + N_4z_4 \\ 1 &= N_1 + N_2 + N_3 + N_4 \end{aligned} \tag{3}$$

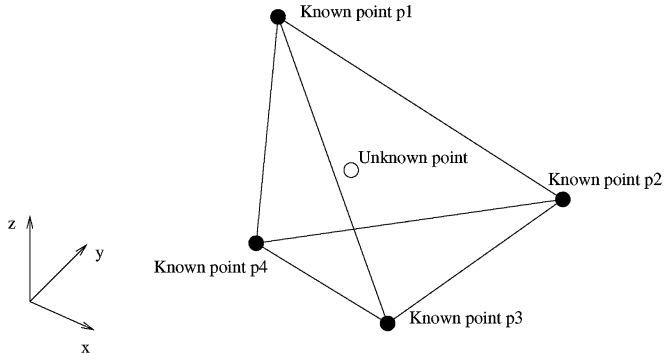


Fig. 1. Illustration of tetrahedral interpolation.

2. METHODS

At the present time, color conversion algorithms are based on fixed, predetermined print objectives (e.g. colorimetric match). Criteria such as GCR and restrictions on total ink deposition are incorporated into typical color management systems in such a way that they are fixed to the user. For this reason, a color management system designed to achieve optimal print quality may not be practical for situations where the cost of printing is a major concern. Rendering intents suitable for a given printing application may not be appropriate for another. A color management system allowing for user control over trade-off problems such as cost versus print quality represents a significant improvement over "black box" systems which are fixed to the user. It was the goal of this study to develop both on a conceptual basis and as functioning software, a color management system offering user flexibility over a variety of color conversion objectives.

The following sections present an optimization-based approach to color management, first in terms of minimizing ΔE^*_{ab}, then in terms of cost minimization, and finally in terms of a generalized Pareto-optimal formulation allowing for compromise among any number of print objectives.

2.1 Optimization Approach for Minimization of ΔE^*_{ab}

Consider a set of transfer functions which provide a relation between the CIELAB and CMYK color spaces (i.e. predict the CIELAB values of a printed image given CMYK values).

$$\begin{aligned} L^*_{\text{prediction}} &= f_L(CMYK) \\ a^*_{\text{prediction}} &= f_a(CMYK) \\ b^*_{\text{prediction}} &= f_b(CMYK) \end{aligned} \tag{4}$$

An optimization problem can be constructed based on minimizing ΔE^*_{ab} which yields CMYK values for a given CIELAB tone [Drakopoulos 1997a; 1997b; 1998; Iino and Berns 1998; Nakauchi et al. 1998; Nakauchi et al. 1999; Littlewood 2001].

$$\begin{aligned} &\text{minimize the function} \\ &f(CMYK) = \sqrt{(L - f_L)^2 + (a - f_a)^2 + (b - f_b)^2} \\ &\text{where } (L, a, b) = (L^*, a^*, b^*)_{input} \end{aligned} \tag{5}$$

In this formulation, $f(CMYK)$ is simply the distance (Euclidean norm) in the CIELAB space separating the given (input) color from the predicted (output) color, ΔE^*_{ab}. The solution $f(CMYK) = 0$ is found when the predicted output CIELAB values match exactly the input CIELAB values, which is possible for any in-gamut color. The optimization-based solution for any out-of-gamut color is simply the closest in-gamut color, found by minimizing the objective function $f(CMYK)$. The conversion of out-of-gamut colors to the closest in-gamut color is commonly termed clipping, and has been shown to be a preferable technique for gamut mapping [Gentile et al. 1990].

The formulation of color conversion as an optimization problem is the foundation of the Pareto-optimal approach to color conversion. This formulation allows for the conversion of both out-of-gamut and in-gamut colors, and can be expanded with the specification of alternative conversion objectives.

2.2 Optimization Approach for Cost Minimization

While the approach described in Section 2.1 allows for color conversion based on the minimization of ΔE^*_{ab}, the general approach is much broader and can be modified in accordance with alternative objectives such as the minimization of cost. The minimization of cost can be achieved either by minimizing total ink or by maximizing black ink, as the maximization of black ink typically reduces chromatic ink deposition (as in GCR). In the case of total ink minimization, a color in the CIELAB space is converted into a CMYK space using as little ink as possible while maintaining a prescribed accuracy $\Delta E^*_{ab\,\max}$.[1] The optimization problem takes the form

$$\begin{aligned} &\text{minimize the function} \\ &\quad f(CMYK) = C + M + Y + K \\ &\text{subject to the constraint} \\ &\quad h(CMYK) = \Delta E^*_{ab\,\max} - \Delta E^*_{ab} \geq 0 \\ &\text{where } \Delta E^*_{ab} \text{ is found as} \\ &\quad \Delta E^*_{ab} = \sqrt{(L - f_L)^2 + (a - f_a)^2 + (b - f_b)^2} \end{aligned} \tag{6}$$

This approach is practical in cases where cost is a key factor. Users of this approach can specify the maximum allowable ΔE^*_{ab} between the original and converted image, and then minimize cost based on that chosen maximum ΔE^*_{ab}.

The minimization of ink approach can be interpreted as gamut compression where input CIELAB values are mapped a distance $\Delta E^*_{ab\,\max}$ towards the point of least cost. In addition to savings in cost, this mapping provides protection against the loss of gradients at the gamut edge. In-gamut colors and colors lying a distance less than $\Delta E^*_{ab\,\max}$ from the gamut surface yield an in-gamut solution, while colors lying a distance greater than $\Delta E^*_{ab\,\max}$ from the gamut boundary are clipped to the boundary surface. Thus the minimization of ink formulation unifies gamut compression and clipping schemes into a single methodology.

An alternative approach for lowering the cost of color printing is to maximize the amount of black ink in the image. This approach is analogous to GCR implementations which seek to lower the usage of chromatic inks by increasing the percentage of black ink. Maximization of black ink for a prescribed

[1]Due to gamut mismatch, prescribed accuracies may not be achievable for out-of-gamut colors. This is an important consideration when implementing Equation (6). Throughout this study, optimization routines were provided with a clause allowing for the replacement of Equation (6) with Equation (5) (i.e. a clipping algorithm) for cases where the prescribed ΔE^*_{ab} constraint could not be met.

maximum allowable ΔE^*_{ab} leads to a problem of the form

$$\begin{aligned} &\text{minimize the function} \\ &\quad f(CMYK) = -K \\ &\text{subject to the constraint} \\ &\quad h(CMYK) = \Delta E^*_{ab\,\max} - \Delta E^*_{ab} \geq 0 \end{aligned} \tag{7}$$

2.3 Generalized Pareto-optimal Formulation

In general, the cost and accuracy measures discussed in Sections 2.1 and 2.2 are competing objectives. In other words, a general color management problem involves trade-offs between competing quantities of cost, accuracy, and perhaps other user specified objectives (e.g. a rendering intent that prefers saturated colors). Optimization theory provides a very natural framework for decision making in the presence of competing objectives through multi-objective optimization or Pareto-optimization [Stadler 1988]. A Pareto-optimal problem in general is a mapping problem which possesses a vector of objectives expressed as a function of the unknown variables

$$\underline{f}(\underline{x}) : R^n \rightarrow R^m \tag{8}$$

It is easy to show [Stadler 1988] that this vector-objective problem is equivalent to minimizing a scalar objective of the form

$$\begin{aligned} &\underline{c}^T \underline{f}(\underline{x}) \\ &\underline{c} = (c_1, c_2, \ldots, c_m) \end{aligned} \tag{9}$$

for each combination of the constraints c_i in the range [0,1]. Furthermore, an objective equation may be divided by any one of the parameters c_i resulting in the removal of a degree of freedom and an objective function with parameters in the range [0,∞]. The family of Pareto-optimal solutions therefore represents a *m-1* dimensional hyper-surface for a m dimensional objective. Thus, the problem formulations discussed in Sections 2.1 and 2.2 may be combined in the unified Pareto-optimal problem

$$\begin{aligned} &\text{minimize the function} \\ &\quad f(CMYK) = c_1 \Delta E^*_{ab}(CMYK) + c_2 I(CMYK) - c_3 K + c_4 R(CMYK) \\ &\text{subject to the constraints} \\ &\quad h_1(CMYK) = \Delta E^*_{ab\,\max} - \Delta E^*_{ab} \geq 0 \\ &\quad h_2(CMYK) = I_{max} - I \geq 0 \\ &\quad h_3(CMYK) \geq 0 \\ &\text{where} \\ &\quad I(CMYK) = C + M + Y + K \\ &\quad R(CMYK) \text{ is any arbitrary user defined print objective} \\ &\quad h_3(CMYK) \text{ is any arbitrary user specified restriction to color conversion} \\ &\quad \text{and } c_i \text{ are user defined coefficients specifying the overall print objective} \end{aligned} \tag{10}$$

Any solution in the Pareto-optimal family is "non-dominated" since, at any solution, it is not possible to further reduce all of the functions simultaneously. Preferred solutions may be based on a single term in the Pareto-optimal formulation or on a compromise between any number of them. Proper selection of the parameters c_i and the bounds enforced by the constraint equations are sufficient to achieve

any preferred solution. The following list provides examples of possible conversion objectives and the corresponding parameter values:

—Reproduce the image as accurately as possible ($c_1 = 1$, $c_2 = c_3 = 0$, $\Delta E^*_{ab\,\max} = I_{max} = \infty$).
—Minimize total ink with a restriction on allowable error ($c_2 = 1$, $c_1 = c_3 = 0$, $\Delta E^*_{ab\,\max}$ = prescribed value, $I_{max} = \infty$).
—Minimize error with a restriction on total ink ($c_1 = 1$, $c_2 = c_3 = 0$, $\Delta E^*_{ab\,\max} = \infty$, I_{max} = prescribed value).
—Maximize black ink with a restriction on allowable error ($c_3 = 1$, $c_1 = c_2 = 0$, $\Delta E^*_{ab\,\max}$ = prescribed value, $I_{max} = \infty$).
—Combined objective function: minimize error while reducing total ink (c_1 = prescribed value, c_2 = prescribed value, $c_3 = 0$, $\Delta E^*_{ab\,\max} = I_{max} = \infty$).
—Any arbitrary combination of objectives and constraints.

Problem (10) could be easily modified to incorporate alternative conversion objectives, such as the minimization of perceptual difference as studied by Nakauchi et al. [1999; 1998]. Stollnitz et al. [1998] and Power et al. [1996] also demonstrate the versatility of optimization in color management by finding optimal ink selections for printing. The constraint equation set can also be modified for inclusion of alternative mapping strategies. Constant hue angle, for example, can be enforced using the constraint equation

$$h(CMYK) = atan\left(\frac{b}{a}\right) - atan\left(\frac{f_b}{f_a}\right) = 0 \tag{11}$$

A wide variety of user requirements related to GCR can be also be met using the Pareto-optimal approach. A powerful expansion of the Pareto-optimal approach allowing for GCR control was developed in conjunction with this study and is the subject of current work as described in Section 6.5 of this paper.

2.4 Artificial Neural Networks as Transfer Functions

The success of the Pareto-optimal approach to color conversion is dependent on the choice of transfer functions presented in Equation Set (4). As was described in Section 1, LUT-based conversions use interpolation to define a relation between CIELAB and CMYK. Interpolation formulae require a relatively dense LUT to accurately capture the behavior of four-color printers, resulting in a large number of required characterization measurements [Johnson 1996; Kanamori 1999]. An alternative to interpolation-based transformations is found in artificial neural networks (ANNs) [Kang and Anderson 1992; Marcu and Iwata 1993; Arai et al. 1993; Abe and Marcu 1994; Tominaga 1993; 1996; 1998a; Drakopoulos 1997a; 1997b; 1998; Tominaga 1998a; 1998b; Littlewood 2001]. ANNs are capable of modeling any arbitrary function [Hornik et al. 1989; Leshno et al. 1993], and thus provide a means of capturing the nonlinearity of ink mixing, a significant source of error in interpolation schemes [Kanamori and Kotera 1992; Kanamori 1999]. In addition to their general ability to model nonlinear processes, ANNs are distinct from LUT-based schemes in that they provide a global model of the process, as opposed to combining many local basis functions.

Consider the transfer functions

$$f_L = \tilde{f}_L(C, M, Y, K), \quad f_a = \tilde{f}_a(C, M, Y, K), \quad f_b = \tilde{f}_b(C, M, Y, K) \tag{12}$$

where $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$ are ANNs trained using a characterization data set (a characterization data set is constructed by printing a set of CMYK combinations and measuring the CIELAB values of the output).

By utilizing ANNs, the creation of a LUT is circumvented, as characterization data are required only for the training of the ANNs. The size of the characterization set required for equivalent accuracy is substantially smaller than the size of a LUT [Drakopoulos 1997a; 1997b; 1998]. A properly selected characterization set spans the entire printer gamut and reflects its nonlinearity, yielding ANNs which also possess these qualities.

2.5 Formulation of Tetrahedral Interpolation as an Optimization Problem

As described earlier, the formulation of color space conversion as a Pareto-optimal problem enables considerable generality in terms of specifying the conversion objectives and constraints. This generality is not easily achieved with classical tetrahedral interpolation. However, it is shown here that the volume coordinates N_i found explicitly using the tetrahedral interpolation formulae can also be found through the solution of an optimization problem. This allows a generalization of tetrahedral interpolation to include arbitrary, user-specified objectives.

Consider a point inside a tetrahedron in the CIELAB space, where the vertices of the tetrahedron are calibrated points as in standard interpolation. If the point's location is specified by the volume coordinates N_i, then the following relations can be established.

$$\begin{aligned} L_p &= N_1L_1 + N_2L_2 + N_3L_3 + N_4L_4 \\ a_p &= N_1a_1 + N_2a_2 + N_3a_3 + N_4a_4 \\ b_p &= N_1b_1 + N_2b_2 + N_3b_3 + N_4b_4 \\ C_p &= N_1C_1 + N_2C_2 + N_3C_3 + N_4C_4 \\ M_p &= N_1M_1 + N_2M_2 + N_3M_3 + N_4M_4 \\ Y_p &= N_1Y_1 + N_2Y_2 + N_3Y_3 + N_4Y_4 \\ K_p &= N_1K_1 + N_2K_2 + N_3K_3 + N_4K_4 \\ \sum_{i=1}^{4} N_i &= 1 \end{aligned} \tag{13}$$

The subscript p refers to the point specified by N_i, and the numeric subscripts refer to the tetrahedron's vertices. The N_i and CMYK values can be determined for in-gamut colors as described earlier (Equations (2) and (3)). However, this interpolation fails for out-of-gamut colors.

An equivalent, but more general optimization problem can then be constructed as follows: treat the volume coordinates N_i as variables and solve the problem

minimize the function

$$f(N_i) = c_1\Delta E^*_{ab}(N_i) + c_2 I(N_i) - c_3 K(N_i) + c_4 R(N_i) \tag{14}$$

subject to the constraints

$$\begin{aligned} h_1(N_i) &= \sum_{i=1}^{4} N_i - 1 = 0 \\ h_2(N_i) &= \Delta E^*_{ab\,\max} - \Delta E^*_{ab} \geq 0 \\ h_3(N_i) &= I_{max} - I \geq 0 \\ h_4(N_i) &\geq 0 \end{aligned}$$

where

$$I(N_i) = \sum_{i=1}^{4}(C_i N_i + M_i N_i + Y_i N_i + K_i N_i)$$

$$K(N_i) = \sum_{i=1}^{4} K_i N_i$$

$R(N_i)$ is any arbitrary user defined print objective

$h_4(CMYK)$ is any arbitrary user specified restriction to color conversion

and c_i are user defined coefficients specifing the overall print objective

This approach differs from the general Pareto-optimal approach in that the independent variables are the intermediate variables N_i and not the CMYK values themselves.

Equation (14) is equivalent to tetrahedral interpolation for the case where $c_1 = 1$, $c_2 = c_3 = 0$, and all constraints other than h_1 in Equation 14 are removed. The solution is found when the first order necessary conditions for minimum are satisfied. For an in-gamut color with no restriction on ink usage ($I_{\max} = 4$) this occurs when:

$$\begin{aligned} L_p &= L = N_1 L_1 + N_2 L_2 + N_3 L_3 + N_4 L_4 \\ a_p &= a = N_1 a_1 + N_2 a_2 + N_3 a_3 + N_4 a_4 \\ b_p &= b = N_1 b_1 + N_2 b_2 + N_3 b_3 + N_4 b_4 \\ &\text{and } N_1 + N_2 + N_3 + N_4 = 1 \end{aligned} \tag{15}$$

which is identical to Equation set (3), the equations from which the standard tetrahedral interpolation formulae are derived.

Unlike standard tetrahedral interpolation, however, the Pareto-optimal formulation is not limited to cases where the input point lies inside a tetrahedron. For cases when the input CIELAB values fall outside the print device gamut, no tetrahedron exists that encompasses the input point.[2] As with the general Pareto-optimal problem, the Pareto-optimal formulation of tetrahedral interpolation yields a solution for out-of-gamut colors which is simply the closest in-gamut color. Hence, the Pareto-optimal formulation of tetrahedral interpolation is equivalent to a clipping algorithm for out-of-gamut colors.

For the purpose of implementation, it is important to note that in general the Pareto-optimal problem must be solved for all tetrahedra filling the printer gamut. The repeated solving of the Pareto-optimal problem for each tetrahedron can be avoided for the case where $c_1 = 1$, $c_2 = c_3 = 0$, and all constraints other than h_1 in Equation 14 are removed (i.e. the case where the Pareto-optimal problem reduces to standard tetrahedral interpolation). In this case, N_i can be solved for directly in each tetrahedron using the standard formulae for calculating volume coordinates. The tetrahedron encompassing the input point is identified when $0 \leq N_i \leq 1$. If no tetrahedron is found to encompass the input point, the corresponding color is necessarily out-of-gamut and the optimization-based scheme can be applied to generate a clipping gamut mapping.

The reformulation of tetrahedral interpolation as a Pareto-optimal problem may be extended to include interpolation between any given number of known points as

minimize the function

$$f(N_i) = c_1 \Delta E^*_{ab}(N_i) + c_2 I(N_i) - c_3 K(N_i) + c_4 R(N_i)$$

[2]This assumes all characterization points were created by measuring printed patches and not artificially created for the purpose of gamut mapping.

subject to the constraints

$$h_1(N_i) = \sum_{i=1}^{n} N_i - 1 = 0$$
$$h_2(N_i) = \Delta E^*_{ab\,\max} - \Delta E^*_{ab} \geq 0$$
$$h_3(N_i) = I_{max} - I \geq 0$$
$$h_4(N_i) \geq 0 \tag{16}$$

where n is the number of points over which the interpolation is made. In this way, interpolation in arbitrarily shaped polyhedra can be expressed as a Pareto-optimal problem. The n points used in the Pareto-optimal problem need not form a polyhedron about the input point, however. Any number of known points in any arbitrary configuration is suitable for use in the Pareto-optimal problem. The entire characterization set, in fact, may be used as the set of n points. This removes the restriction that the characterization set be subdivided, and allows for interpolation over the entire characterization set. This type of interpolation procedure is analogous to the "meshless methods" of the finite element method [Belytschko et al. 1996], just as tetrahedral interpolation for color conversion is analogous to the standard tetrahedral finite element method [Zienkiewicz and Taylor 1989]. Increasing the number of points n over which the Pareto-optimal formulation of interpolation is applied is not without cost. As the number of points n is increased, the optimization routine must manipulate a greater number of independent variables N_i, which results in greater computational expense.

3. IMPLEMENTATION

The ANN- and tetrahedral-interpolation-based schemes presented in Sections 2.4 and 2.5 were implemented in the programs NeuralColor and OptInterpol, respectively. The codes were evaluated both quantitatively and qualitatively through the conversion of several Tiff images. All image conversions were carried out on a Dell personal computer with a 400 MHz Pentium processor and 128 MB of RAM running Windows NT 4.0. The converted Tiff images and the characterization data set were printed on a Tektronix Phaser 740 printer. Colorimetric data were measured using a X-Rite Digital Swatchbook spectrophotometer with D65 illuminant and 2 degree standard observer. Consistent paper stock and ambient light were maintained throughout the project.

Key issues related to the implementation of the Pareto-optimal approach, as well as the basic architectures of the codes NeuralColor and OptInterpol, are presented in the following sections. In particular, Sections 3.1 and 3.2 cover the construction of the characterization data set and the selection of an optimization routine, two issues strongly affecting the performance of NeuralColor and OptInterpol.

3.1 The Characterization Data Set

A 149-color characterization data set was developed by the authors to efficiently and accurately capture the nonlinearity of color mixing while spanning the entire gamut of the print device. The data set was used for the training of NeuralColor's ANNs and for the tetrahedral interpolation of OptInterpol. The data were selected based on ANN training requirements, as well as knowledge of ink mixing as studied by Drakopoulos [1997a, 1997b, 1998]. The practice of calculating CMYK values from CMY values was strictly avoided, as this practice neglects dark areas of the gamut which can only be produced using CMYK.

To evaluate the nonlinearity of a printer gamut quantitatively, Drakopoulos trained the ANNs $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$ using the basic ANSI IT8.7/3 data set [ANSI 1993] printed on an IBM Network Color Printer. A 4^6 full factorial design was then used to test the accuracy of the ANNs as a function of the amount

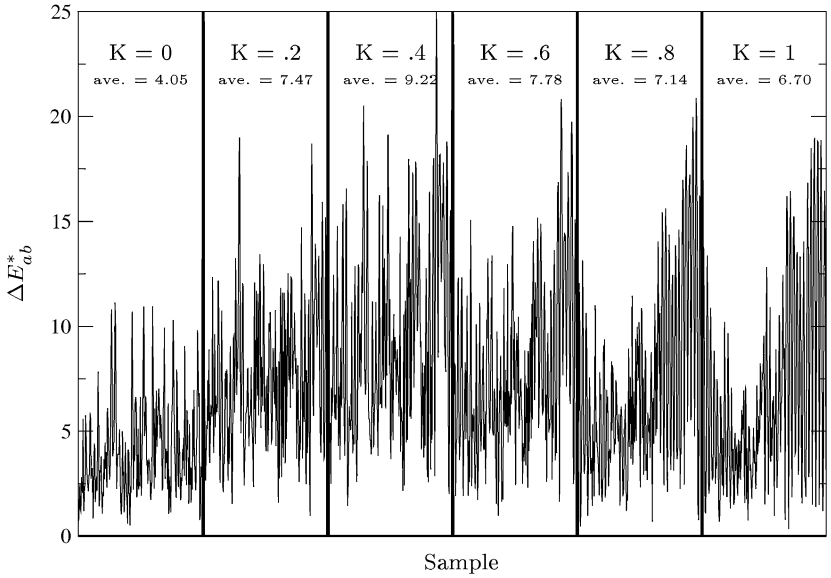


Fig. 2. ΔE^*_{ab} histogram illustrating the relationship between ΔE^*_{ab} and black ink.

of black ink in the printed tone. The results of this experiment are presented in Figure 2, which shows that the error is greatest in the region where the dot fraction for black ink equals .4. The K values for the colors in the characterization data set are based on this knowledge. The six chromatic primaries[3] at various levels of saturation with black dot fractions of .2, .4, and .7 constitute the first 96 colors of the characterization set. Colors 97 through 101 are made up of several levels of grey obtained solely with black ink, including the white point (zero black). Colors 102 through 131 are mixtures of the chromatic primaries with various levels of black. The final 18 data are mixtures of primaries and black. Table I gives a more detailed description of the logic behind the 149-color characterization set. Figure 3 illustrates the 149-color characterization set in the CIELAB space. Appendix A contains a complete listing of the characterization data set, including the measured CIELAB values for each color.

Chromaticity diagrams are often used to visualize printer gamuts. Figure 4 shows the six chromatic primaries of four-color printing with various dot fractions of black. The figure may be interpreted as cross sections of the printer gamut taken at different locations on the lightness scale. The change in shape of the cross sections as black is added illustrates again the nonlinear behavior of ink mixing.

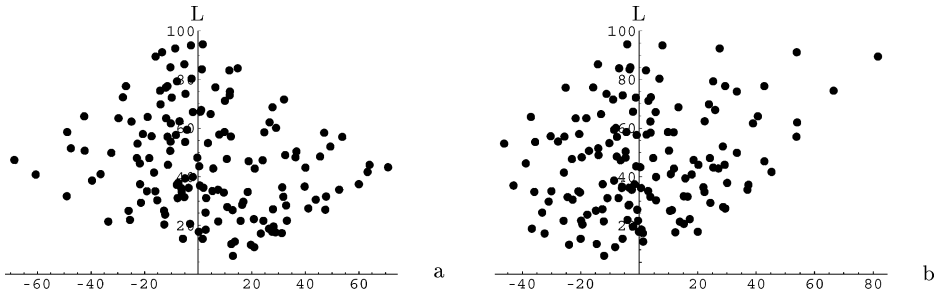
3.2 The Sequential Quadratic Programming Routine NLPQL

The Pareto-optimal problem dictating the conversion from CIELAB to CMYK can be solved efficiently and accurately using a nonlinear programming algorithm. In any direct optimization code, two issues are critical, namely the efficiency of search direction and the guarantee of convergence from any starting point (global convergence) [Dennis and Schnabel 1983]. Sequential quadratic programming (SQP) algorithms are efficient in search direction determination (superlinear convergence), and have proved to be effective in solving equality- and inequality-constrained optimization problems when combined with appropriate globalization schemes [Rustem 1998; Lootsma 1985]. Therefore, the SQP code NLPQL [Schittkowski 1985] was selected for use in the present study.

[3]The chromatic primaries for four-color printing are cyan, green, yellow, red, magenta, and blue.

Table I. Construction Methodology for the 149-Color Characterization Set

Points	Description
1–6	Primaries
7–12	.2 (Primaries)
13–18	.4 (Primaries)
19–24	.7 (Primaries)
25–30	Primaries + .2 Black
31–36	.2 (Primaries) + .2 Black
37–42	.4 (Primaries) + .2 Black
43–48	.7 (Primaries) + .2 Black
49–54	Primaries + .4 Black
55–60	.2 (Primaries) + .4 Black
61–66	.4 (Primaries) + .4 Black
67–72	.7 (Primaries) + .4 Black
73–78	Primaries + .7 Black
79–84	.2 (Primaries) + .7 Black
85–90	.4 (Primaries) + .7 Black
91–96	.7 (Primaries) + .7 Black
97	White Point
98–101	Greyscale Using Black
102–111	Mixtures of Primaries
112–121	Mixtures of Primaries + .3 Black
122–131	Mixtures of Primaries + .7 Black
132–149	Disproportional Mixing of Primaries and Black



(a) L-a plane. (b) L-b plane.

Fig. 3. The 149-color characterization set.

The code NLPQL allows for a reverse-communication implementation in which the driving program supplies the values of the objective function, objective function derivatives, constraint equations, and constraint equation derivatives. This is unlike standard implementations where the optimization code directly calls the function routines. In addition to the constraint equations, NLPQL maintains upper and lower bounds for the variables $\underline{x}$. The implementation of NLPQL is illustrated in Figure 5.

3.3 NeuralColor

The color management system NeuralColor, first developed in the thesis by Drakopoulos [1997a, 1997b, 1998], was expanded to fully implement the Pareto-optimal problem presented in Section 2.3 [Littlewood

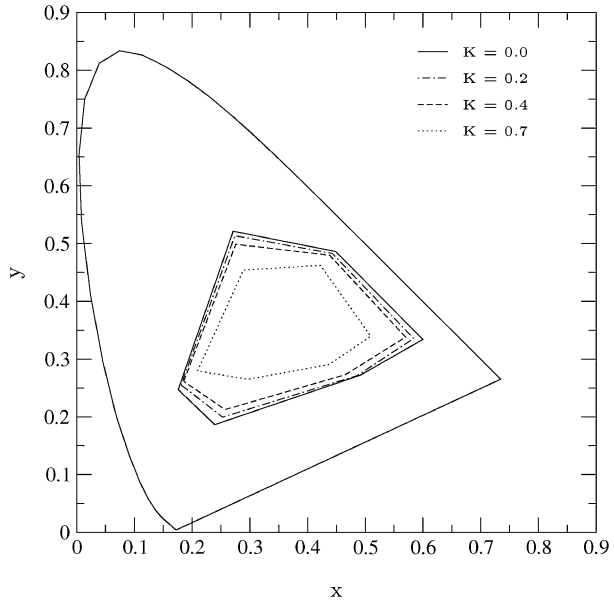


Fig. 4. Primaries with various amounts of black added.

2001]. As described in previous sections, NeuralColor operates by minimizing the objective function for each pixel using the NLPQL routine, yielding a set of CMYK values. A flow chart for NeuralColor is presented in Figure 6.

A crucial detail in the development of NeuralColor is the creation of the ANNs $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$. The ANNs used in the program NeuralColor are regression models that predict either L^*, a^*, or b^* as a function of CMYK. Specifically, one-hidden-layer feedforward ANNs were trained by backpropagation [Haykin 1999] to model the relationship between CMYK and CIELAB. ANNs are well suited for capturing the nonlinear behavior of color mixing. It has been proven that ANNs are universal approximators, and that in particular a standard multilayer feedforward network with a locally-bounded piecewise-continuous activation function can approximate any continuous function to any degree of accuracy if and only if the network's activation function is not a polynomial [Hornik et al. 1989; Leshno et al. 1993]. This property guarantees that ANNs are capable of capturing the nonlinearity of the CMYK to CIELAB conversion.

Figure 7 illustrates the general structure of the ANN models. The ANNs are comprised of an input layer, a hidden layer, and an output layer. Each input neuron feeds each hidden-layer neuron. As data travels from input neuron j to a hidden-layer neuron k, it is multiplied by the weight ω_{1jk} and summed with bias b_{1j}. At each hidden neuron, the activation function $Act()$ is called and the resulting value is sent to the output layer. As data travels to the output layer from hidden-layer neuron j, it is multiplied by weight ω_{2j1} and summed with bias b_{2j}. The activation function $Act()$ is again called by the output neuron, and the resulting value is output by the network. Sigmoid functions are the most common form of activation function [Haykin 1999]. In this study, hyperbolic tangent was used as the ANN-activation function. Hyperbolic tangent is a monotonically increasing, non-polynomial function with upper and lower bounds of 1 and -1, respectively.

The values for ω_{ijk} and b_{ij} were determined by a backpropagation procedure. Backpropagation was achieved by implementing the LMDIF1 subroutine, which solves the least squares problem with the

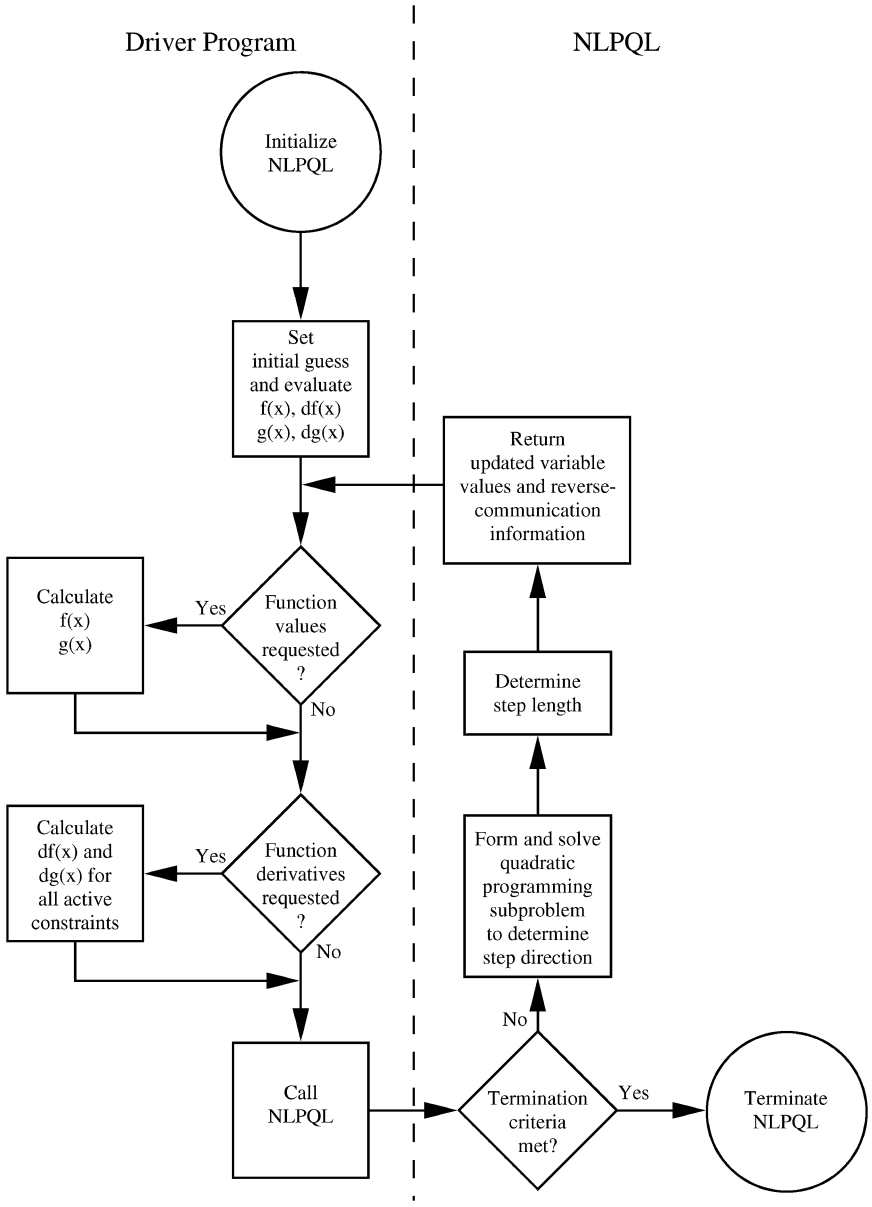


Fig. 5. Flow diagram for reverse-communication interface with the optimization routine NLPQL. The objective function is denoted as f(x), the constraint equations as g(x), and their respective derivatives as df(x) and dg(x).

Levenberg-Marquardt algorithm [More et al. 1980; Dennis and Schnabel 1983]. In this procedure, the mean square error (MSE) between a known characterization set and the ANN predictions is minimized. The 149-color characterization set was used for training the ANNs. The proper number of hidden neurons was determined by an exhaustive comparison of possible structures, with the number of hidden neurons ranging from one to nine.

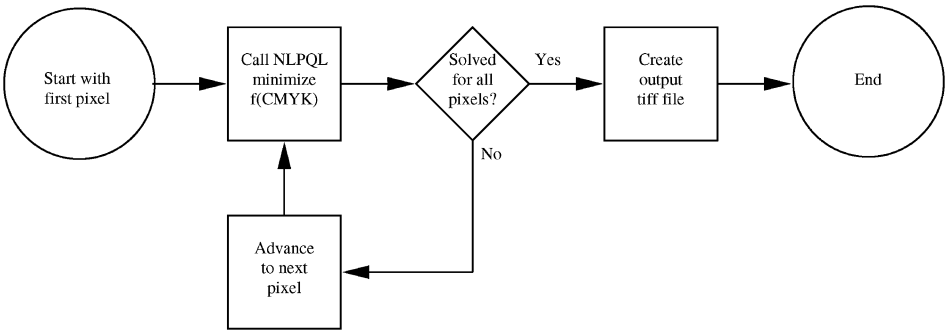


Fig. 6. Flow diagram for program NeuralColor.

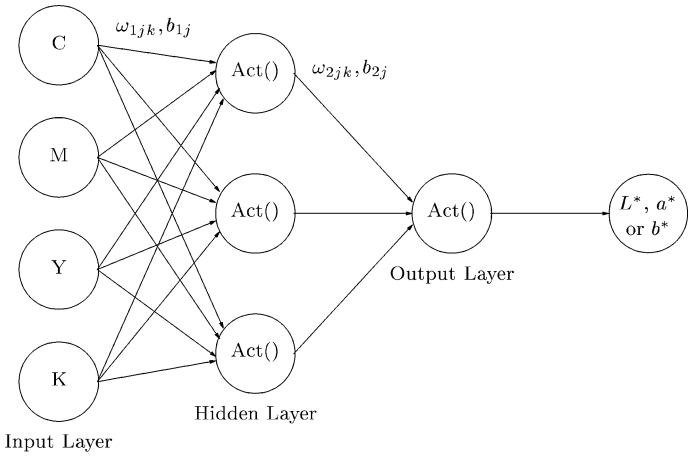
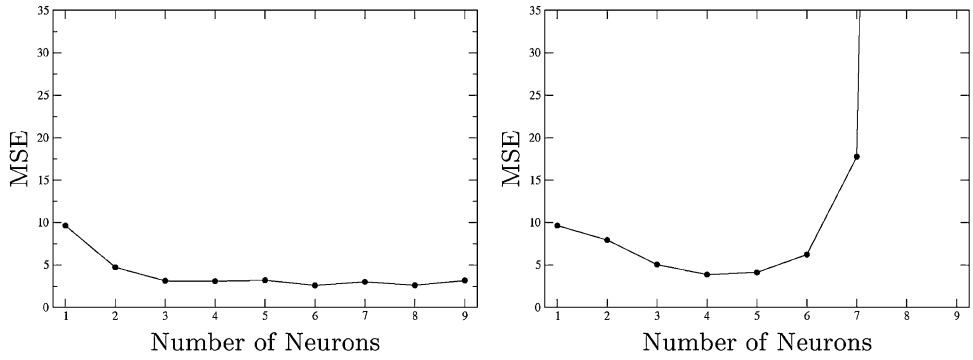


Fig. 7. An illustration of a four-input, one-hidden-layer, one-output ANN with three neurons in the hidden layer. The ANNs $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$ used by NeuralColor contained 5, 6, and 7 hidden neurons, respectively.

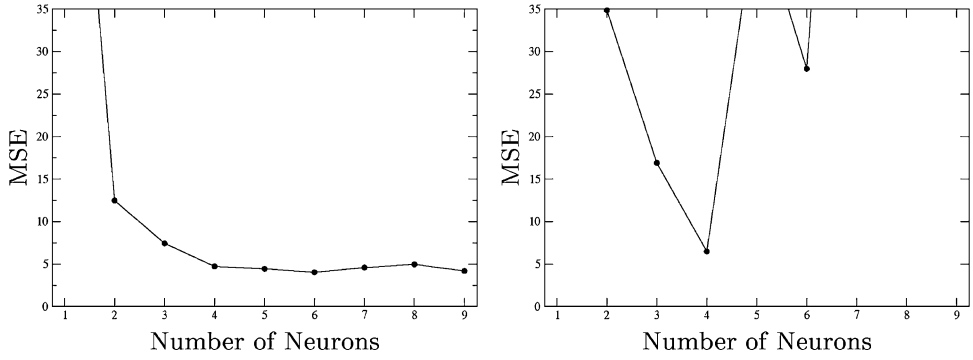
For training of the ANNs $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$, six copies of the 149-color set were printed and measured. Colorimetric data were then averaged over the six measurements. This was done to reduce characterization errors, which result from such factors as measurement error, color variation within a printed patch, and color variation between prints. In a preliminary study, five patches were randomly chosen from the 149-color set and measured on each of 25 separate prints. From these measurements, it was estimated that the standard deviation when printing on the Tektronix Phaser 740 printer is equal to 1.145 ΔE^*_{ab}. This correlates to a 95% confidence interval of $\pm.77$ ΔE^*_{ab} when averaging over 6 measurements. Averaging over three measurements, which was done elsewhere in this study, corresponds to a 95% confidence interval of ± 1.09 ΔE^*_{ab}.

For the purpose of identifying suitable ANN structures for $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$, a systematic experiment was carried out that compared the performances of ANNs with different numbers of hidden neurons. ANNs were trained with structures that varied from one hidden neuron to nine hidden neurons. The comparison of ANN structures utilized two distinct sets of data. The 149-color characterization set was used for training, and the extended ANSI IT8.7/3 set was used for testing the ANNs. Measurements were averaged over three copies for the ANN-structure-comparison experiments. The extended



(a) Least error solution. (b) Mean solution.

Fig. 8. ANN testing results for $\tilde{f}_L$.

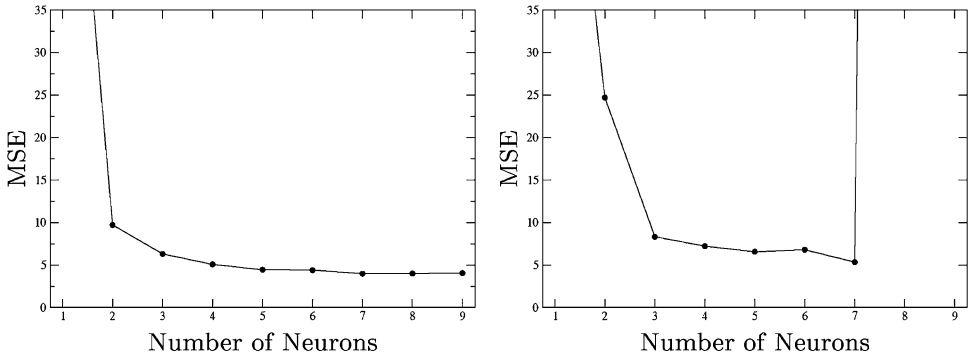


(a) Least error solution. (b) Mean solution.

Fig. 9. ANN testing results for $\tilde{f}_a$.

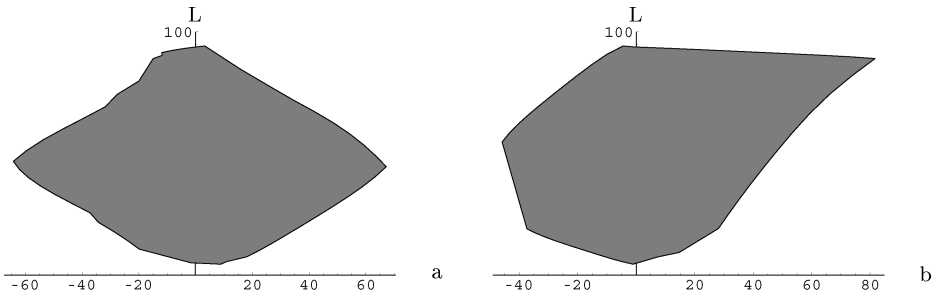
ANSI IT8.7/3 set was modified by removing from it any color which also appears in the 149-color training set, resulting in a testing set of 851 colors. This modification ensured that the ANNs were tested on data that they were not exposed to during the training process. Testing of this type measures an ANNs ability to generalize the transformation, and also reveals overfitting. In every case, 10 repetitions of the training procedure were carried out with random initial values assigned to the weights ω_{ijk} and initial values of zero assigned to the biases b_{ij}. For each structure, the best test MSE was recorded, along with the average test MSE. Complete testing results are presented in Figures 8, 9, and 10.

Of primary importance is the best test MSE obtained over the 10 training runs. The results for $\tilde{f}_L$ show that accurate prediction is possible when the number of neurons in the hidden layer exceeds two. Even in the case of a single hidden-layer neuron, the MSE for $\tilde{f}_L$ is not exceedingly large. The minimum number of hidden-layer neurons required for accurate prediction of a^* and b^* is four neurons. These data suggest that the relationship between CMYK and lightness is less complex than the relationship between CMYK and hue.



(a) Least error solution. (b) Mean solution.

Fig. 10. ANN testing results for $\tilde{f}_b$.



(a) L-a plane. (b) L-b plane.

Fig. 11. The printer gamut as calculated by the ANNs $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$.

Comparison of the best MSE values and the mean MSE values shows clearly that ANN training is generally not repeatable when the weights are assigned random initial values. This behavior may be explained in terms of local minima [Sontag and Sussmann 1989; Gori and Tesi 1992]. The training algorithm begins with randomly assigned weights and makes iterative adjustments until a minimum is found (i.e. until the derivatives of the MSE function with respect to ω_{ijk} and b_{ij} are zero). There is no guarantee that the minimum found will be the global minimum. This causes the wide variation in solutions. The solution variation was taken into consideration when selecting the best ANN structures for $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$. ANN structures capable of obtaining low MSE values are clearly desirable, but also desirable is a structure that is well-behaved during the training process.

A trade-off exists in the determination of ANN structures for $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$. As more neurons are added to the hidden layer, neural networks are better able to handle complex relationships. Larger ANNs, however, are more computationally expensive to evaluate and exhibit inconsistency in the training process. Based on the data presented in Figures 8, 9 and 10, ANN structures of 5, 6, and 7 hidden-layer neurons were selected for $\tilde{f}_L$, $\tilde{f}_a$, and $\tilde{f}_b$, respectively. The printer gamut as calculated by the final ANNs is shown in Figure 11. The gamut images for this figure were generated by inputting a 4^{17} full factorial design in CMYK into the ANNs.

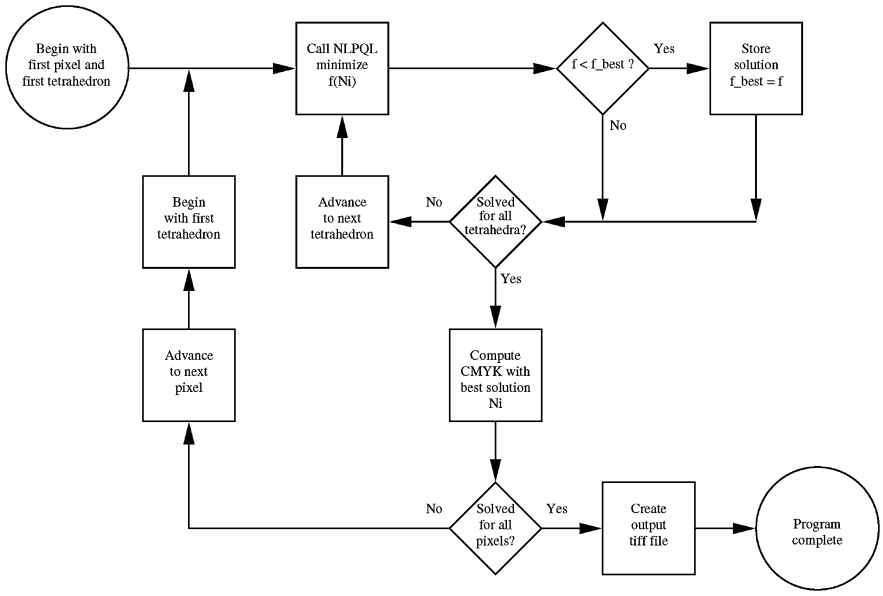


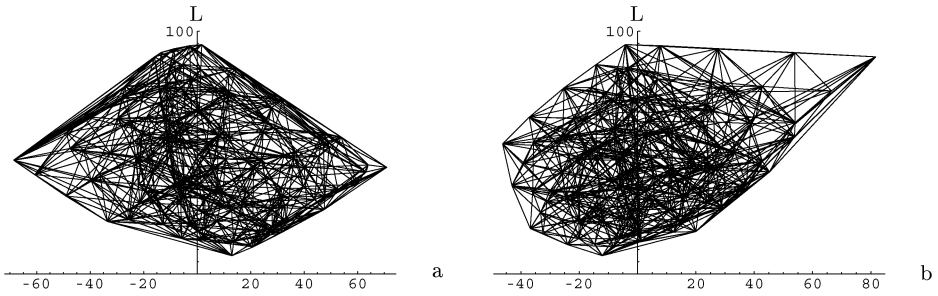
Fig. 12. Flow diagram for program OptInterpol.

3.4 OptInterpol

The tetrahedral-interpolation-based Pareto-optimal scheme was implemented through creation of the color management system OptInterpol. As with NeuralColor, OptInterpol utilizes the optimization routine NLPQL. A flow chart illustrating the architecture of OptInterpol is presented in Figure 12. Unlike NeuralColor, OptInterpol solves the Pareto-optimal problem multiple times for each pixel, considering each tetrahedron individually. The use of a Pareto-objective function does not allow the proper tetrahedron to be selected prior to the solution of the Pareto-optimal problem. This is due to the fact that input and output CIELAB values do not match in general. For example, in the solution for minimization of total ink with a specified $\Delta E^*_{ab\,\max}$, the input and output CIELAB values do not match, as colorimetric accuracy was sacrificed for a reduction in cost. Without a priori knowledge of the output CIELAB values, a search for the proper tetrahedron cannot be made.

Typical interpolation-based color management schemes operate using a characterization set organized in a lattice structure [Hung 1992; 1993]. A well organized structure of this type allows for straightforward division into cubes or tetrahedra [Kanamori and Kotera 1992]. The 149-color characterization set created in this study, however, is not well organized. As described in Section 3.1, the characterization set was chosen to efficiently capture gamut nonlinearity and is not evenly distributed in the CIELAB color space. The division of the characterization set into tetrahedra is therefore a non-trivial problem requiring the application of more advanced mathematics.

A solution to this problem was found in the code Qhull, a public domain program provided by The Geometry Center, Minneapolis, Minnesota [Barber and Huhdanpaa 1998; Barber et al. 1996]. In the context of this research, Qhull is used to compute a three-dimensional Delaunay triangulation, resulting in a set of tetrahedra which completely fill the printer gamut and do not overlap one another. In total, 826 tetrahedra were formed to fill the printer gamut, as illustrated in Figure 13. It is noted that Figure 13 gives a depiction of the printer gamut similar to that of Balasubramanian and Dalal [1997], who used a modified convex hull algorithm to estimate printer gamut surfaces. After running the Qhull



(a) L-a plane. (b) L-b plane.

Fig. 13. Qhull output: tetrahedra filling the 149-color characterization set.

Table II. Print Objectives and Corresponding Parameter Values

c1	c2	c3	$\Delta E^*_{ab\,\max}$	$\text{Ink}_{\max}$	Print Objective
1	0	0	400	4	Reproduce image as accurately as possible
1	0	0	400	1	minimize error with a restriction on total ink
0	1	0	20	4	minimize total ink with a high allowable ΔE^*_{ab}
0	1	0	5	4	minimize total ink with a low allowable ΔE^*_{ab}
0	0	1	20	4	maximize black with a high allowable ΔE^*_{ab}
0	0	1	5	4	maximize black with a low allowable ΔE^*_{ab}
2.5	1	0	400	4	unconstrained solution of Pareto-optimal problem.
8	0	1	400	4	unconstrained solution of Pareto-optimal problem.

program, the output was reformatted and the CMYK data was added for each point. The result is a data file containing the CIELAB and CMYK data for each point in a number of tetrahedra. The data file is read by OptInterpol, providing the tetrahedra required for interpolation.

4. EVALUATION METHODOLOGY

Quantitative evaluations of NeuralColor and OptInterpol were made through conversion of the MacBeth ColorChecker^{tm} color rendition chart. A 24-pixel image, with each pixel corresponding to a patch in the MacBeth ColorChecker^{tm} chart, was converted from CIELAB to CMYK and then printed at a resolution of 2 pixels per inch. True error, $\Delta E^*_{ab\,\text{true}}$, for each patch in the chart was calculated as

$$\Delta E^*_{ab\,\text{true}} = \sqrt{(L^*_{input} - L^*_{output})^2 + (a^*_{input} - a^*_{output})^2 + (b^*_{input} - b^*_{output})^2} \tag{17}$$

where $(L^*a^*b^*)_{input}$ are the input values supplied to the conversion programs and $(L^*a^*b^*)_{output}$ were determined by physical measurement of the printed image after conversion. Average ink usage for cyan, magenta, yellow, and black was tabulated, as well as program run time (CPU time).

The MacBeth ColorChecker^{tm} chart was converted using the programs NeuralColor and OptInterpol for eight sets of parameter values. The parameter values were selected for comparison of different output objectives, as described in Section 2.3. The set of conversion objectives, along with the corresponding parameter values,[4] are presented in Table II. The values of c_i for the unconstrained solutions of the Pareto-optimal problem (listed at the bottom of Table II) were chosen to create a balance between the

[4]Note that the objective function, as it appears in the code for NeuralColor and OptInterpol, is $f = \frac{1}{375}c_1\Delta E^*_{ab} + \frac{1}{4}c_2 I - c_3 K$.

Table III. Ink Usage and CPU Time for Conversion of the MacBeth ColorChecker^{tm} Chart (24 Pixels) using NeuralColor

c1	c2	c3	$\Delta E^*_{ab\,\max}$	$\text{Ink}_{\max}$	C	M	Y	K	Total Ink	CPU Time (sec)
1	0	0	400	4	0.31	0.37	0.46	0.10	1.24	0.12
1	0	0	400	1	0.17	0.24	0.32	0.17	0.90	0.11
0	1	0	20	4	0.11	0.19	0.23	0.13	0.66	0.09
0	1	0	5	4	0.19	0.26	0.33	0.18	0.96	0.29
0	0	1	20	4	0.16	0.24	0.37	0.48	1.25	0.11
0	0	1	5	4	0.20	0.27	0.36	0.26	1.09	0.27
2.5	1	0	400	4	0.14	0.22	0.27	0.19	0.83	0.09
8	0	1	400	4	0.15	0.26	0.40	0.54	1.35	0.08

competing conversion objectives. Typical solution values for ink usage were in the neighborhood of 30% of the maximum possible ink usage ($I_{\text{max possible}} = 4$, $I_{\text{typical}} \approx 1.2$, $K_{\text{max possible}} = 1$, $K_{\text{typical}} \approx .3$). Typical solution values for ΔE^*_{ab}, however, were in the neighborhood of 5% of the maximum possible ΔE^*_{ab} (the maximum possible value for ΔE^*_{ab} is based on the ranges of L^*, a^*, and b^* used in ICC profiles [ICC 1998], $\Delta E^*_{ab\,\text{max possible}} = \sqrt{256^2 + 256^2 + 100^2} \approx 375$, $\Delta E^*_{ab\,\text{typical}} \approx 20$). Due to the nature of these solution values, careful selection of c_i is required to avoid dominance of a single term in the objective function. Note that setting $\Delta E^*_{ab\,\max}$ equal to 400 and $\text{Ink}_{\max}$ equal to 4 is equivalent to removing the constraints.

In addition to the parameter values listed in TableII, the MacBeth ColorChecker^{tm} chart was converted with the objective of minimizing total ink with the constraint on ΔE^*_{ab} ranging from 0 to 20, and also for the maximization of black ink with the constraint on ΔE^*_{ab} ranging from 0 to 20. These conversions were used to analyze ink usage as a function of the constraint on ΔE^*_{ab} and were not analyzed for colorimetric accuracy, as is discussed in Section 5.2.

In addition to the quantitative data obtained through conversion of the MacBeth ColorChecker^{tm} chart, several Tiff images were converted for the purpose of subjectively evaluating the conversion schemes. Four images, each with its dominant tones falling in different regions of the printer gamut, were converted using both NeuralColor and OptInterpol. These results provide a qualitative feel for the effects of different conversion objectives, something which is not provided by the numerical results.

5. RESULTS

The programs NeuralColor and OptInterpol were successful in converting the MacBeth ColorChecker^{tm} color rendition chart with different sets of conversion parameters as described in the preceding section. Quantitative data from these conversions are included in Section 5.1. The colorimetric data presented in Section 5.1 were averaged over three prints. Ink usage as a function of the constraint on ΔE^*_{ab} for the minimization of total ink and the maximization of black ink is presented in Section 5.2. Section 5.3 describes the image conversions used to evaluate the Pareto-optimal approaches qualitatively.

5.1 Quantitative Evaluations

The code NeuralColor was used to convert the MacBeth ColorChecker^{tm} chart for each of the eight prescribed parameter sets. Quantitative results for these conversions are presented in Tables III and IV. Ink usage (CMYK) was tabulated in terms of average dot fraction ($0 \leq \text{CMYK} \leq 1$). To determine which of the colors in the MacBeth ColorChecker^{tm} chart are in-gamut, NeuralColor was used to convert the image for minimization of ΔE^*_{ab} with the constraint that ΔE^*_{ab} be equal to zero. For any case in which

Table IV. Colorimetric Accuracy of the Results Obtained with the Program NeuralColor

c1	c2	c3	$\Delta E^*_{ab\,max}$	Ink_{max}	Overall $\Delta E^*_{ab\,true}$	In-gamut (14) $\Delta E^*_{ab\,true}$	Out-of-gamut (10) $\Delta E^*_{ab\,true}$
1	0	0	400	4	5.3	2.8	8.8
1	0	0	400	1	10.8	8.3	14.2
0	1	0	20	4	19.3	19.1	19.7
0	1	0	5	4	6.5	4.8	8.9
0	0	1	20	4	20.8	20.3	21.5
0	0	1	5	4	7.5	6.3	9.2
2.5	1	0	400	4	10.5	8.3	13.6
8	0	1	400	4	22.0	20.0	24.7

Table V. Ink Usage and CPU Time for Conversion of the MacBeth ColorChecker^{tm} Chart (24 Pixels) using OptInterpol

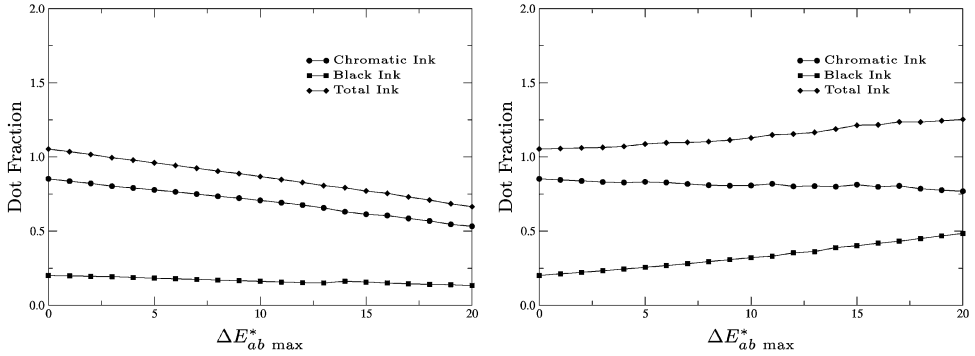
c1	c2	c3	$\Delta E^*_{ab\,max}$	Ink_{max}	C	M	Y	K	Total Ink	CPU Time (sec)
1	0	0	400	4	0.26	0.31	0.39	0.17	1.13	23.03
1	0	0	400	1	0.20	0.23	0.28	0.16	0.87	21.76
0	1	0	20	4	0.13	0.19	0.20	0.13	0.64	57.13
0	1	0	5	4	0.21	0.26	0.32	0.18	0.98	15.86
0	0	1	20	4	0.17	0.25	0.38	0.47	1.27	62.38
0	0	1	5	4	0.23	0.28	0.36	0.25	1.12	78.98
2.5	1	0	400	4	0.16	0.20	0.26	0.17	0.80	17.54
8	0	1	400	4	0.19	0.23	0.35	0.34	1.11	15.72

Table VI. Colorimetric Accuracy of the Results Obtained using the Program OptInterpol

c1	c2	c3	$\Delta E^*_{ab\,max}$	Ink_{max}	Overall $\Delta E^*_{ab\,true}$	In-gamut (15) $\Delta E^*_{ab\,true}$	Out-of-gamut (9) $\Delta E^*_{ab\,true}$
1	0	0	400	4	5.7	2.7	10.7
1	0	0	400	1	12.1	9.1	17.1
0	1	0	20	4	21.6	21.0	22.6
0	1	0	5	4	7.0	5.2	10.1
0	0	1	20	4	20.9	20.3	21.8
0	0	1	5	4	7.9	6.5	10.3
2.5	1	0	400	4	13.8	12.7	15.5
8	0	1	400	4	11.4	8.5	17.0

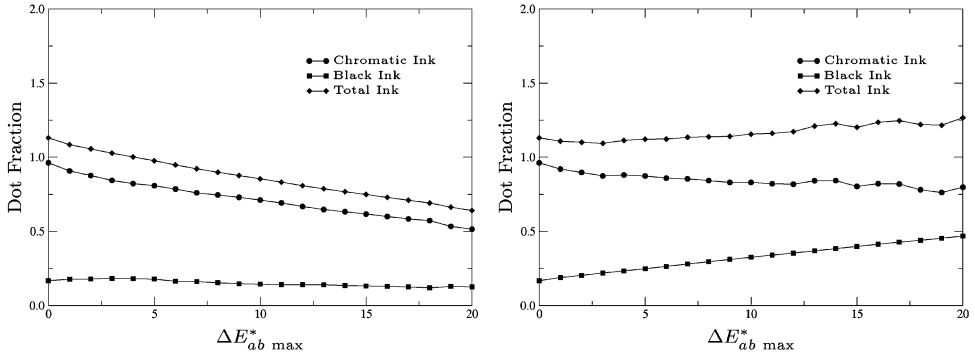
NeuralColor returned no solution, the color was determined to be out-of-gamut. Ten of the twenty-four colors were found to be out-of-gamut. Colorimetric error is listed separately for in-gamut and out-of-gamut colors in Table IV.

Results for the conversions of the MacBeth ColorChecker^{tm} chart using each of the eight parameter sets are presented in Tables V and VI for OptInterpol. To determine which colors are in-gamut, OptInterpol was also used to convert the image for minimization of ΔE^*_{ab} with the constraint that ΔE^*_{ab} be equal to zero. Using OptInterpol in this way, nine colors were found to be out-of-gamut. This result differs from the result obtain using NeuralColor by one color, and illustrates that, in practice, the color-management system affects the obtainable gamut of a printing system. It is again worth noting that the first conversion in Tables V and VI is equivalent to standard tetrahedral interpolation over an unevenly distributed data set in conjunction with a clipping algorithm for out-of-gamut colors.



(a) Minimization of total ink with various $\Delta E^*_{ab\ \max}$ constraints using NeuralColor.

(b) Maximization of black ink with various $\Delta E^*_{ab\ \max}$ constraints using NeuralColor.



(c) Minimization of total ink with various $\Delta E^*_{ab\ \max}$ constraints using OptInterpol.

(d) Maximization of black ink with various $\Delta E^*_{ab\ \max}$ constraints using OptInterpol.

Fig. 14. Ink usage for conversion of the MacBeth ColorChecker^{tm} chart.

5.2 Further Analysis of Ink Reduction Techniques

The results presented in Tables III and V provide physically measured data for the analysis of the Pareto-optimal approach to color management. Specifying the $\Delta E^*_{ab\,\max}$ constraint to be both 5 and 20 for the minimization of total ink and maximization of black ink approaches provides information about the behavior of these schemes as the $\Delta E^*_{ab\,\max}$ constraint is increased (i.e. as the trade-off problem of cost versus reproduction accuracy moves in the direction of lowering cost). In an effort to further document the behavior of these algorithms as cost becomes a more dominant factor, the minimization of total ink and the maximization of black ink approaches were analyzed by varying $\Delta E^*_{ab\,\max}$ from 0 to 20 on increments of 1 for conversion of the MacBeth ColorChecker^{tm} chart. The ink usage for each case is presented in Figures 14(a) through 14(d).

5.3 Qualitative Results

The numerical results presented in Section 5.1 allow for an analysis of converted images in terms of colorimetric match and printing cost. A rigorous evaluation of the Pareto-optimal formulations for image conversion requires qualitative results as well. Several test images were selected for this purpose,

each having dominant tones in a different region of the printer gamut. Images were acquired from the Kodak web page and from the Adobe Photoshop CD [Kodak; Adobe Photoshop 1996]. Appendix B and Appendix C contain these test images converted with a variety of conversion objectives.

6. DISCUSSION AND CONCLUSIONS

A number of common color management objectives including colorimetric accuracy, cost minimization, and gamut mapping were unified under the general Pareto-optimal formulation. Two distinct implementations of this formulation, namely the ANN-based NeuralColor and the interpolation-based OptInterpol, allowed for printing with various user-controlled color management objectives. A 149-color characterization set was used both as an ANN training set and for the creation of a tetrahedral mesh, as facilitated by the Qhull program. The converted test images demonstrate the validity of the Pareto-optimal color management systems developed in this study.

Prior to directly addressing the image conversion results in Section 6.2, the general behavior of NeuralColor and OptInterpol when solving indeterminate problems are addressed in Section 6.1. In Section 6.3, a comparison of the color management systems developed in this study to other existing methods is made. Finally, the creation of ICC profiles for real-time image conversion using the Pareto-optimal approach is covered in Section 6.4.

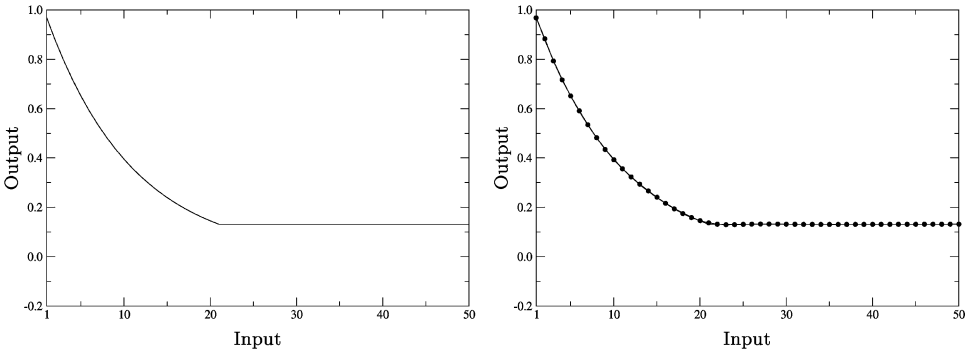
6.1 Uniqueness of Solutions

As was described in preceding sections, a transformation from CIELAB to CMYK is, in general indeterminate, having infinite solutions for any color inside the printer gamut. In the context of the Pareto-optimal approach, an example of an indeterminate transformation is that of Equation (5). The interpolation-based color management scheme is capable of finding only one of the solutions to indeterminate transforms of this type. Factors such as GCR are inherently determined by the construction of the characterization set. Desirable traits such as smooth GCR must therefore be considered during the construction of the characterization set if they are to be implemented by interpolation-based color management schemes.

ANN behavior in relation to uniqueness of solutions for CIELAB to CMYK conversion is far less rigid than that of interpolation-based schemes. There is no constraint explicitly guiding the ANN scheme to a particular solution of an intermediate problem. The code NeuralColor, however, returns unique solutions without exception when given a CIELAB image for conversion into CMYK.

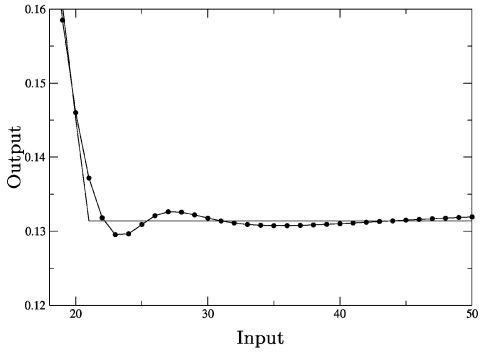
The convergence of NeuralColor to a single solution in the case of indeterminate problems is a result of the imperfect fit provided by the ANNs. The ANNs created for use in NeuralColor are locally oscillatory, leading to a unique local minimum in cases where a multitude of minima actually exist. When given input CIELAB values, NeuralColor finds a unique CMYK solution from the theoretically infinite set of solutions as a result of this local oscillatory behavior. Figures 15(a) through 15(c) were constructed using a simple 1-D example, and are presented as an illustration of this behavior. The training function is single valued for training points 21 through 50. The ANN, however, is oscillatory, as can be seen clearly in Figure 15(c).

Although this behavior leads to a unique solution for the Pareto-optimal problems developed in this study, it is possible to expand the Pareto-optimal approach to allow for user control over the chosen solution. Rather than allowing local oscillations of the ANNs to dictate the choice of solution, an approach has been developed which allows for a user-specified solution. This expansion is outlined in Section 6.5 along with its most promising application, user-controlled GCR.



(a) The function used to generate the ANN training set.

(b) The ANN overlaid on the training function.



(c) ANN behavior in the neighborhood of training points 18 through 50.

Fig. 15. Illustration of the locally oscillatory behavior of ANNs.

6.2 Analysis of Image Conversions

The following conclusions are drawn from the numerical results presented in Section 5 and from the qualitative image conversions presented in Appendix B and Appendix C.

(1) Prints of the MacBeth ColorChecker^{tm} chart obtained by unconstrained minimization of ΔE^*_{ab} had an average error for in-gamut colors of 2.7 ΔE^*_{ab} and 2.8 ΔE^*_{ab} for NeuralColor and OptInterpol, respectively. Average errors across all colors in the MacBeth ColorChecker^{tm} chart were 5.3 ΔE^*_{ab} and 5.7 ΔE^*_{ab}, respectively. These results confirm NeuralColor and OptInterpol's ability to reproduce images with colorimetric accuracy.

(2) Minimization of ΔE^*_{ab} in conjunction with the constraint $I_{\max}$ allows for image conversion for a given upper bound on ink usage. With a constraint of 1 on total ink (i.e. dot fractions $C+M+Y+K \leq 1$), the MacBeth ColorChecker^{tm} chart was reproduced with an average error across all colors of 10.8 ΔE^*_{ab} for NeuralColor and 12.1 ΔE^*_{ab} for OptInterpol.

(3) The lowest possible total ink usage for a specified maximum allowable ΔE^*_{ab} can be found using the minimization of total ink approach with the $\Delta E^*_{ab\,\max}$ constraint. When solving for the minimization of ink with $\Delta E^*_{ab\,\max} = 5$, NeuralColor found a solution that used an average ink per pixel of .96 and had a measured error of 4.8 ΔE^*_{ab}. By increasing $\Delta E^*_{ab\,\max}$ to 20, the average ink per pixel was reduced

to .66 with an average measured error of 19.1 ΔE^*_{ab}. Similar results were found with OptInterpol, as shown in Tables V and VI.

(4) Solutions for the minimization of total ink effectively lower the cost of printing by reducing total ink deposition. The savings in cost is off-set by an increase in ΔE^*_{ab}. The inverse relationship between cost and colorimetric accuracy for the ink minimization approach is evidenced in Figures 14(a) and 14(c).

(5) The maximization of black ink approach in conjunction with the constraint $\Delta E^*_{ab\,\max}$ is not effective in reducing cost. As illustrated in Figures 14(a) and 14(d), the total ink deposition is not reduced as $\Delta E^*_{ab\,\max}$ is increased. This behavior is explained by the fact that the maximum GCR solution was found when the $\Delta E^*_{ab\,\max}$ constraint was set to zero. Increasing $\Delta E^*_{ab\,\max}$ did not have the effect of increasing GCR, because the maximum GCR level had already been reached. Instead, increasing $\Delta E^*_{ab\,\max}$ simply resulted in the increased deposition of black. This both degraded the image and increased total ink deposition (cost).

(6) Tighter constraints (e.g. lower $\Delta E^*_{ab\,\max}$ or $I_{\max}$) increase computational time.

(7) The use of a mixed objective function balancing the minimization of ΔE^*_{ab} and the minimization of ink allows for compromise between the objectives of colorimetric accuracy and cost, without the introduction of constraints into the Pareto-optimal problem. The exclusion of constraints from the problem reduces computational time, but does not allow for strict upper bounds such as $I_{\max}$. The parameters c_i governing these conversions must be chosen carefully to avoid dominance of a single term in the Pareto-optimal solution.

(8) The tetrahedral-interpolation-based code OptInterpol requires extensive computational time in comparison to the ANN-based code NeuralColor. This is a result of OptInterpol's repeated solving of the optimization problem for each tetrahedron in the characterization set.

(9) The results obtained using OptInterpol are comparable to the results obtained by NeuralColor, indicating that the 149-color characterization set is well selected for capturing the nonlinear nature of color printing. The characterization data are densely arranged in areas of high nonlinearity, and sparsely arranged in more linear regions, allowing the local linear models of interpolation to give results comparable to the global nonlinear ANN model despite the small overall size of the characterization set.

(10) Undesirable artifacts were created in several of the images in Appendices B and C. Bands are present, particularly when the $\Delta E^*_{ab\,\max}$ constraint is loosened to 20 ΔE^*_{ab}. These artifacts are most evident in the capsule image and the balloons image. These bands or jumps are the result of the programs NeuralColor and OptInterpol finding significantly different solutions for very similar input colors. Figure 16 illustrates this phenomenon in more detail. Figure 16 shows the input CIELAB image and the output CMYK image for two pixels located in the thumb in the capsule image. The output image was created using NeuralColor for maximization of black ink with $\Delta E^*_{ab\,\max}$ equal to 20. The neighboring input colors are nearly identical, with CIELAB values equal to (68, 28, 37) and (68, 28, 35), respectively. They are part of a smooth transition in the input image. A significant color jump is present in the output image. The CMYK values for the output image are (0, .43, .77, .27) and (0, .31, .67, .27), respectively. The ANNs used in NeuralColor predict corresponding output CIELAB values of (50.7, 18.3, 33.4) and (56.6, 11.9, 30.7). This gives a colorimetric error for the first pixel of 20.16 ΔE^*_{ab} and a colorimetric error for the second pixel of 20.19 ΔE^*_{ab}. The two output pixels were calculated from very similar input colors, and both are $\approx 20 \Delta E^*_{ab}$ away from their respective input values in the CIELAB space, but the output pixels differ from each other by 9.11 ΔE^*_{ab}. This type of artifact is reduced as the constraint on ΔE^*_{ab} is tightened; the worst case scenario creates a false jump of magnitude $2\Delta E^*_{ab\,\max}$. The color management formulations in the present study regard each pixel in an image as a separate color conversion problem and do not take into consideration the relationships between neighboring pixels.

(a) The input $L^*a^*b^*$ image.

(b) The output CMYK image.

Fig. 16. An example of artifacts present in the converted images. Both pixels reproduce the originals with an error of ≈20 ΔE^*_{ab}, but the output pixels differ from each other by ≈9 ΔE^*_{ab}.

It would be possible, however, to develop such relationships and include them in the Pareto-optimal formulation.

The Pareto-optimal approach to color conversion investigated in this study offers a novel approach to color printing. Important contributions include the unification of multiple printing objectives in a single methodology and an ink minimization formulation combining clipping and gamut compression. The formulation of interpolation as an optimization problem and the utilization of the program Qhull for division of arbitrarily arranged characterization data into tetrahedra, are novel approaches offering new flexibility in color management. The Pareto-optimal approach to printer color management could be applied to any color-space transform, such as those required for display on a monitor or for image scanning. Printing was a natural choice for the application of the Pareto-optimal approach due to the trade-off of cost versus reproduction accuracy, however any number of rendering intents for image display or scanning could be included in the formulation. The success of this project has provided the groundwork for current studies that shall be presented in follow-up papers.

6.3 Comparison to other Methods

A variety of methods are available for color conversion into a printer color space, including LUTs and continuous models. Comparison of the methods developed in this study to other methods of printer color management can be made in terms of flexibility, practicality, and the ability to accurately reproduce an image.

Flexibility is an important strong point of the Pareto-optimal approach. The Pareto-optimal approach unifies any number of rendering intents into a single methodology, including cost and reproduction accuracy. Color-mixing models attempt to predict the color of a print given the dot fractions of ink laid down on the paper. The use of these models alone does not allow for any flexibility in rendering intent. Likewise, a color management system that operates using a limited set of LUTs, is limited to

the conversions defined in those LUTs. The ability to print with arbitrary rendering intent, sets the Pareto-optimal approach apart from other existing methods.

A shortcoming of NeuralColor and OptInterpol is the extensive computational time required for printing. The Pareto-optimal approach requires the solution of a general equality- and inequality-constrained nonlinear optimization problem. This type of optimization problem must be solved iteratively, requiring extensive computation in comparison to simply evaluating an equation or performing interpolation inside a LUT. The program WriteICC was developed for the creation of ICC profiles in order to facilitate real-time image conversion using the Pareto-optimal approach. Section 6.4 covers the creation of ICC profiles with arbitrary rendering intent using the Pareto-optimal approach. For any given rendering intent, an ICC profile can be created and stored in a library for later use. The computational time for creating an ICC profile is equivalent to the time required for conversion of an equivalently-sized image, but once the profile is created it can be used for real-time conversion of any image and is suitable for use in a variety of commercial image processing codes.

To evaluate the ability of the color management systems developed in this study to accurately reproduce an image, a number of studies have been selected from the literature for comparison. Comparisons are best made with studies in which results were presented for measured print data, as opposed to print simulations or comparison to other mathematical models of the printing process. The literature cited for comparison presents data for image conversion with the intent of colorimetric match. This intent was implemented in the present study through the unconstrained minimization of ΔE^*_{ab}, for which NeuralColor and OptInterpol had average errors of 2.8 ΔE^*_{ab} and 2.7 ΔE^*_{ab} for in-gamut colors in the MacBeth ColorChecker^{tm} chart, and errors of 5.3 ΔE^*_{ab} and 5.7 ΔE^*_{ab} across all colors in the MacBeth ColorChecker^{tm} chart. The data cited in the following paragraphs show that the ability of the color management systems developed in this study is comparable to, and often exceeds, that of other existing methods.

Kang [1994] presented an extensive evaluation of various color mixing models including the Neugebauer equations, the Yule-Nielsen model, the Clapper-Yule model, the Beer-Bouguer law, the Kubelka-Monk model, and several hybrid approaches. A range of values was used in the model parameters in each case, for a total of 52 cases. Modal parameters were determined using 10-level color wedges for each primary. A Canon Color Laser Copier 500 (CLC-500) was used for printing in the study. The best results for modeling CMY printing over 64 test patches was 5.01 ΔE^*_{ab} for a single-constant Kubelka-Munk equation with a halftone correction factor. The same model had an average error of 7.15 ΔE^*_{ab} when applied to CMYK printing and tested over 58 patches. Iino and Berns [1998] evaluated the accuracy of the Murry-Davies-Yule-Nielsen and the Omatsu models. Model parameters were calculated using data from 57 print samples. Average testing errors over an independent data set were 4.2 and 3.9 ΔE^*_{ab} for the Murray-Davies-Yule-Nielsen and the Omatsu models, respectively. Rolleston and Balasubramanian [1993] evaluated the accuracy of various types of Neugebauer models using a Xerox 5775 color printer. Results were obtained by printing 1000 random samples; average errors ranged from 2.600 ΔE^*_{ab} for cellular Yule-Nielsen modified Neugebauer Equations, to 7.414 ΔE^*_{ab} for the basic (unmodified) Neugebauer equations. The various Neugebauer equations required 16 characterization measurements. Mongeon [1996] presented results for image transformations using 4th-order polynomial regression models. A Xerox 5760 printer was modeled using a 360-color test print, and test results were presented for the Kodak Q60 print (236 colors). Average error over the entire test print was 3.56 ΔE^*_{ab}, and the average error for in-gamut colors was reported to be 2.47 ΔE^*_{ab}. Nin, Kasson, and Plouffe [1992] evaluated a tri-linear interpolation scheme for conversion from CIELAB to CMY. An interpolation table was constructed for a DuPont 4Cast using colors at constant hue angles of 30, 90, 150, 210, 270, and 330 degrees, with squares 5 units apart in luminance and chroma. The MacBeth ColorChecker^{tm} Chart was then reproduced with an average error of 2.8 ΔE^*_{ab} across all colors.

Results from other studies that used ANNs to create printer models are highly relevant to the present study. A study by Kang and Anderson [1992] investigated the use of ANNs for scanner and printer calibration. To examine the use of ANNs as printer models, a Kodak Q60-C print (236 colors) was used to train an ANN for a Canon Color Laser Copier-500 (CLC-500). A number of the data were removed from the training set and used as testing data, with average errors of about 6 ΔE^*_{ab}. Arai, Nakano, and Iga [1993] used a neural network for modeling the transformation from CIELAB to CMY. A sample set of 9261 colors was measured, and a neural network was then trained repeatedly, using randomly selected subsets of between 8 and 125 colors until the 9261-color set was exhausted. Final testing using 125 colors chosen at random from the 9261-color set gave an average error of 2.91 ΔE^*_{ab} for the most accurate of the ANNs. Tominaga [1993, 1996, 1998a] has authored several papers on the use of ANNs for modeling color conversion into a printer color space. In one case, Tominaga trained an ANN using 216 pairs of RGB and XYZ data to model a Shinko Electric model CHC-S443 dye sublimation printer [Tominaga 1993]. The ANN model was then tested over 125 colors with an average error of 2.61 ΔE^*_{ab}. Tominaga [1996] also conducted a study using a thermal transfer-dye sublimation printer. An ANN was trained with 4096 samples, and results showed a mean error of 2.5 ΔE^*_{ab} over 150 test colors. In a later study, Tominaga [1998a] conducted experiments with a dye sublimation printer and an ink jet printer. An ANN was trained using a 6561-color data set, and then tested using 148 color patches. The resulting RMS ΔE^*_{ab} was 2.24.

For a comparison of our method to readily available methods, the MacBeth Chart was printed using the Tektronix Phaser 740 ICC profile (Tkph7401.icm) with relative colorimetric rendering intent from Adobe Photoshop 4.0.1. The average error was 11.4 ΔE^*_{ab}. This relatively poor result is presumed to be due largely to the fact that the ICC profile is intended to capture some average performance of the Tektronix Phaser 740 under various printing conditions, with various paper stock, and so on. The color management systems developed in the present study were calibrated using identical paper stock and similar environmental conditions to those present during testing and thus have a distinct advantage over the use of the ICC profile provided by Tektronix.

6.4 ICC Profile Generation

A shortcoming of the color management systems developed in this study is the extensive computational time required for image conversion. This issue was overcome through the development of the program WriteICC, which creates ICC profiles for use with a wide variety of commercial software. WriteICC was developed using guidelines set by the ICC, and was based largely on code provided by D. Wallner [ICC 1998; Wallner 1998].

By inputting a regular grid in the CIELAB space into NeuralColor or OptInterpol, LUTs for CIELAB to CMYK conversion were created. Likewise, a regular grid in CMYK was used for the creation of LUTs for CMYK to CIELAB conversion. The Pareto-optimal formulation was not used in the case of creating LUTs for CMYK to CIELAB conversion; the functions $f_L(CMYK)$, $f_a(CMYK)$, and $f_b(CMYK)$ were simply used to predict CMYK given CIELAB. In the case of NeuralColor, the ANNs were used to predict CIELAB values given a set of CMYK values. In the case of OptInterpol, CIELAB values were predicted using interpolation inside a 4-D LUT constructed using the 149-color characterization set.

By using the Pareto-optimal approach in conjunction with the program WriteICC, any number of LUTs can be created with arbitrary rendering intent. These LUTs can then be stored in a custom library for later use. The computational effort required to create an ICC profile is equivalent to that of directly reproducing an image using NeuralColor or OptInterpol, but once an ICC profile has been created, image conversion may be achieved in real time. The accuracy lost in the creation of an ICC profile is a function of the profile size. As the density of the profile is increased, interpolation error

decreases. Large profiles, however, require a significant amount of memory for storage, thus users must make a decision regarding the trade-off of accuracy versus memory usage.

The creation of ICC profiles using the Pareto-optimal approach allows for real-time image conversion with arbitrary rendering intent and allows for integration with a variety of commercial software. The loss of accuracy associated with building ICC profiles can be made negligible by creating a profile that is significantly dense. ICC profile generation successfully removes the computational restrictions inherent in the Pareto-optimal approach to printer color management.

6.5 Current Work

A shortcoming of the ANN-based color management scheme is convergence to a single solution for indeterminate problems based on the local behavior of the ANNs. This behavior reduces control of GCR and can lead to inconsistent GCR across the image. An expansion of the Pareto-optimal approach has been developed that allows for user control over GCR.

Indeterminate problems (e.g. Equation (5)) have a set of solutions for any in-gamut color. The solutions within this set are uniquely described in terms of GCR, that is to say the solutions differ only by the dot fraction of black ink. The solution for 0% GCR can be found using NeuralColor by minimizing K with $\Delta E^*_{ab\,\max} = 0$. Likewise, the solution for 100% GCR can be found by maximizing K with with $\Delta E^*_{ab\,\max} = 0$. Maximizing or minimizing total ink with $\Delta E^*_{ab\,\max} = 0$ also yields the 0%-GCR and 100%-GCR solutions, respectively. Once the 0%- and 100%-GCR solutions have been obtained for a given color, any arbitrary GCR solution may be found by setting a constraint on black ink that forces the dot fraction of black ink to fall at some prescribed location between the dot fractions of black ink for the 0%-GCR and 100%-GCR solutions. For example, if a 0%-GCR solution is found for a given color in which the black dot fraction is equal to 0, and a 100%-GCR solution is found in which the black dot fraction is equal to .7, then a 50%-GCR solution may be found by minimizing ΔE^*_{ab} with the constraint that $K = .35$. Alternatively, this methodology may be applied to the smallest chromatic dot fraction (i.e. a constraint may be placed on $MIN(CMY)$.

The method of specifying the dot fraction of black or $MIN(CMY)$ based on the 0%- and 100%-GCR solutions overrides NeuralColor's tendency to converge to a solution based on local ANN behavior and allows for arbitrary, used-specified GCR. Users may specify GCR as a constant value, resulting in consistent GCR across an image, or users may supply a "GCR map," which specifies GCR on a pixel by pixel basis. An example application of the "GCR map" is the assignment of 100% GCR to text within an image. The expanded Pareto-optimal approach for user-controlled GCR has been verified using a modified version of NeuralColor and is the subject of current research. Image conversions with consistent GCR across the image have been free of the gradient jumps that are present in many of the prints obtained in the current study.

A second promising aspect of the ANN-based Pareto-optimal approach is its ability to model a CIELAB to CMYK conversion based on a relatively small number of characterization measurements. This trait lends itself to the field of print device calibration. The remeasurement of the characterization set and retraining of the ANNs would yield a recalibrated printing system with significant savings in cost in comparison to remeasuring a typical LUT. ANNs may also be recalibrated by adding several heavily weighted recalibration points to the training set followed by a retraining of the ANNs. A set of recalibration points might be comprised of Neugebauer's [1937] primaries: red, green, blue, cyan, magenta, yellow, black, and white. A third recalibration scheme is that of modeling printer drift directly, using a new set of ANNs. The original characterization set, or any LUT in general, could be modified using the recalibration ANNs to compensate for printer drift. ANN-based approaches to printer calibration are research topics currently under investigation.

APPENDIX

A. 149-COLOR CHARACTERIZATION SET

Table VII. The 149-Point Characterization Data Set. $L^*a^*b^*$ Data were Averaged over Six Measurements and have a 95% Confidence Interval of $\pm.77\ \Delta E^*_{ab}$

Point No.	C	M	Y	K	L*	a*	b*	Point No.	C	M	Y	K	L*	a*	b*
1	1	0	0	0	53.6	−22.7	−46.3	76	1	1	0	0.7	7.4	13	−12.1
2	0	1	0	0	43.9	70.9	0.3	77	1	0	1	0.7	21.6	−33.6	13.9
3	0	0	1	0	89.5	−15.9	81.7	78	0	1	1	0.7	17.4	27.6	20
4	1	1	0	0	18.6	26.5	−36.8	79	0.2	0	0	0.7	35.4	−3.5	−6
5	1	0	1	0	46.8	−68.6	18.6	80	0	0.2	0	0.7	34.7	7.5	−2.6
6	0	1	1	0	42	63.2	45.3	81	0	0	0.2	0.7	39.9	−2	5.7
7	0.2	0	0	0	86.3	−5.2	−14.2	82	0.2	0.2	0	0.7	31.2	3	−7.2
8	0	0.2	0	0	84.6	14.8	−7	83	0.2	0	0.2	0.7	35.4	−6.4	0.6
9	0	0	0.2	0	94.1	−2.7	7.9	84	0	0.2	0.2	0.7	34.1	5.4	3
10	0.2	0.2	0	0	76.8	6.4	−15.8	85	0.4	0	0	0.7	31.2	−7.7	−10.9
11	0.2	0	0.2	0	85	−10.4	−3.1	86	0	0.4	0	0.7	28.5	16.4	−3.5
12	0	0.2	0.2	0	83.7	11.6	2.2	87	0	0	0.4	0.7	39.3	−4.9	15.8
13	0.4	0	0	0	76.7	−12.1	−25.2	88	0.4	0.4	0	0.7	21.6	7.5	−12.2
14	0	0.4	0	0	71.8	32.1	−9	89	0.4	0	0.4	0.7	30.3	−15.3	5.7
15	0	0	0.4	0	92.8	−8.6	27.5	90	0	0.4	0.4	0.7	26.3	12.9	11.1
16	0.4	0.4	0	0	56.5	12.2	−25.6	91	0.7	0	0	0.7	24.4	−12.3	−17.8
17	0.4	0	0.4	0	72.7	−28.1	3.9	92	0	0.7	0	0.7	19.6	28.1	−2.1
18	0	0.4	0.4	0	68.6	27.8	13.4	93	0	0	0.7	0.7	37.5	−7.4	30.1
19	0.7	0	0	0	64.6	−18.9	−37.1	94	0.7	0.7	0	0.7	12.3	12.4	−14.6
20	0	0.7	0	0	56.4	53.8	−7.7	95	0.7	0	0.7	0.7	26	−26	10.5
21	0	0	0.7	0	91.2	−13.5	53.9	96	0	0.7	0.7	0.7	22.6	20.8	17.3
22	0.7	0.7	0	0	33.8	18.5	−35.9	97	0	0	0	0	94.5	1.7	−4.2
23	0.7	0	0.7	0	58.3	−48.9	9.8	98	0	0	0	0.2	84.2	1.4	−3.4
24	0	0.7	0.7	0	52.4	49.4	28.6	99	0	0	0	0.4	66.7	0.8	−2.2
25	1	0	0	0.2	45.5	−21.8	−38.9	100	0	0	0	0.7	44.3	0.4	−1.1
26	0	1	0	0.2	36.9	60	−1.7	101	0	0	0	1	17.4	0.2	−0.2
27	0	0	1	0.2	75.4	−14.2	66.5	102	0.5	0.5	0.5	0	47.8	−0.3	−4.7
28	1	1	0	0.2	16.6	23.4	−32.4	103	1	0.5	0.5	0	34.1	−16	−20.8
29	1	0	1	0.2	40.9	−60.6	17.9	104	0.5	0.5	1	0	44.9	−11.3	29.2
30	0	1	1	0.2	34.7	52.7	37	105	0.5	1	0.5	0	28.2	32.8	−3.7
31	0.2	0	0	0.2	74.1	−4.8	−11.3	106	1	0.5	0	0	36.4	0.7	−43.1
32	0	0.2	0	0.2	73.5	11.8	−5.7	107	1	0	0.5	0	51.7	−47.5	−14.1
33	0	0	0.2	0.2	80.3	−2.5	6.9	108	0.5	0	1	0	64.9	−42.5	40.6
34	0.2	0.2	0	0.2	65.7	4.6	−13.1	109	0	0.5	1	0	62.4	26.7	54.1
35	0.2	0	0.2	0.2	72.6	−9.9	−1.2	110	0	1	0.5	0	44.9	64	20.2
36	0	0.2	0.2	0.2	71.2	10	3.1	111	0.5	1	0	0	30.6	43.9	−22.8
37	0.4	0	0	0.2	64	−12	−21.8	112	0.5	0.5	0.5	0.3	35.5	2	−3.5
38	0	0.4	0	0.2	60.1	29.1	−8	113	1	0.5	0.5	0.3	26	−12.8	−14.9
39	0	0	0.4	0.2	79.2	−7.9	25.3	114	0.5	0.5	1	0.3	33.8	−6.2	22.3
40	0.4	0.4	0	0.2	47.3	10.7	−23	115	0.5	1	0.5	0.3	21.9	24.4	−0.4
41	0.4	0	0.4	0.2	62.7	−24.9	3.8	116	1	0.5	0	0.3	25.3	2.7	−33.3
42	0	0.4	0.4	0.2	58.2	24.7	11.7	117	1	0	0.5	0.3	38.4	−39.6	−8.6
43	0.7	0	0	0.2	56.7	−17.4	−30.3	118	0.5	0	1	0.3	49.8	−32.4	33.4
44	0	0.7	0	0.2	48.3	45.5	−7.8	119	0	0.5	1	0.3	46.3	18.8	42.8
45	0	0	0.7	0.2	77.3	−11.5	42.8	120	0	1	0.5	0.3	31.9	47.8	16.5
46	0.7	0.7	0	0.2	29.8	16.9	−31	121	0.5	1	0	0.3	21.9	33.1	−19.9
47	0.7	0	0.7	0.2	50.7	−42.4	10.2	122	0.5	0.5	0.5	0.7	18.3	2.8	0.8
48	0	0.7	0.7	0.2	43.7	40	25.3	123	1	0.5	0.5	0.7	14.5	−5.7	−5.8
49	1	0	0	0.4	34.1	−19.1	−30.1	124	0.5	0.5	1	0.7	20.6	−3	15.2
50	0	1	0	0.4	26.4	47.5	−0.7	125	0.5	1	0.5	0.7	13.3	13.7	1.3
51	0	0	1	0.4	56.5	−11.5	53.8	126	1	0.5	0	0.7	14.5	1.7	−20.1
52	1	1	0	0.4	12	19.7	−24	127	1	0	0.5	0.7	22.2	−25.5	−4.1
53	1	0	1	0.4	32	−49.1	15.2	128	0.5	0	1	0.7	29.3	−21.4	24.3
54	0	1	1	0.4	27	41.1	29.4	129	0	0.5	1	0.7	27.6	10.9	28.7
55	0.2	0	0	0.4	59.3	−4.1	−8.7	130	0	1	0.5	0.7	17.2	28.9	12.3
56	0	0.2	0	0.4	58.4	9.8	−4.4	131	0.5	1	0	0.7	11.1	20.9	−8.3
57	0	0	0.2	0.4	66.6	−2	5.6	132	0.3	0.7	0	0	48	36.3	−19.7
58	0.2	0.2	0	0.4	53.8	3.7	−10.3	133	0.7	0.3	0	0	54.3	−4.9	−35.7
59	0.2	0	0.2	0.4	57.1	−8.3	−0.9	134	0.3	0	0.7	0	77.3	−27	29.4
60	0	0.2	0.2	0.4	57.3	7.9	2.5	135	0.7	0	0.3	0	64.1	−29.7	−18.2
61	0.4	0	0	0.4	50.6	−10.4	−17.3	136	0	0.3	0.7	0	75	11.9	33.4
62	0	0.4	0	0.4	46.8	24.1	−6.4	137	0	0.7	0.3	0	58.1	47.1	3.8
63	0	0	0.4	0.4	62.8	−7	22.4	138	0.3	0.7	0	0.5	26.7	27.9	−12.6
64	0.4	0.4	0	0.4	33.5	9.6	−20.3	139	0.7	0.3	0	0.5	31.6	−5.2	−24.6
65	0.4	0	0.4	0.4	47.7	−22.9	4.7	140	0.3	0	0.7	0.5	47.6	−18.3	24.1
66	0	0.4	0.4	0.4	43.3	21.1	11.8	141	0.7	0	0.3	0.5	36.9	−21.7	−12.6
67	0.7	0	0	0.4	41.7	−16.5	−25.8	142	0	0.3	0.7	0.5	43.4	5.6	27.1
68	0	0.7	0	0.4	36	39.8	−5.9	143	0	0.7	0.3	0.5	31.8	31.8	3.8
69	0	0	0.7	0.4	61.9	−10.2	38.9	144	0.3	0.7	0.2	0	48.9	32.5	−14
70	0.7	0.7	0	0.4	21.9	15.8	−25.9	145	0.7	0.3	0.2	0	54.5	−10.3	−27.8
71	0.7	0	0.7	0.4	41.1	−36.4	10.8	146	0.3	0.2	0.7	0	69.8	−14.1	23.8
72	0	0.7	0.7	0.4	35.6	31.4	21.9	147	0.7	0.2	0.3	0	57.6	−20.7	−20.6
73	1	0	0	0.7	20.4	−12.7	−19.5	148	0.2	0.3	0.7	0	67.4	1.1	26
74	0	1	0	0.7	16.8	31.2	1.2	149	0.2	0.7	0.3	0	50.4	36.7	−4.7
75	0	0	1	0.7	36.7	−8	37.3								

B. IMAGES CONVERTED USING NEURALCOLOR

This appendix presents four images, each converted using five different conversion objectives, for the purpose of qualitative analysis. Image conversion parameters were chosen to illustrate unconstrained minimization of ΔE^*_{ab}, as well as minimization of total ink and maximization of black ink with both a tight and a loose constraint on $\Delta E^*_{ab\,\max}$. Table VIII lists average combined ink deposition (C + M + Y + K) for each conversion.

Table VIII. Ink Usage for Images Converted using NeuralColor

Image	Conversion Objective	Average Ink Deposition per Pixel (C + M + Y + K)
Balloons	Unconstrained minimization of ΔE^*_{ab}	1.27
Balloons	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.15
Balloons	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	0.84
Balloons	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.22
Balloons	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.35
Corn	Unconstrained minimization of ΔE^*_{ab}	1.72
Corn	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.54
Corn	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	1.02
Corn	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.70
Corn	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.80
Flower	Unconstrained minimization of ΔE^*_{ab}	1.89
Flower	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.23
Flower	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	0.82
Flower	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.54
Flower	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.96
Capsule	Unconstrained minimization of ΔE^*_{ab}	1.35
Capsule	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.18
Capsule	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	0.81
Capsule	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.37
Capsule	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.61

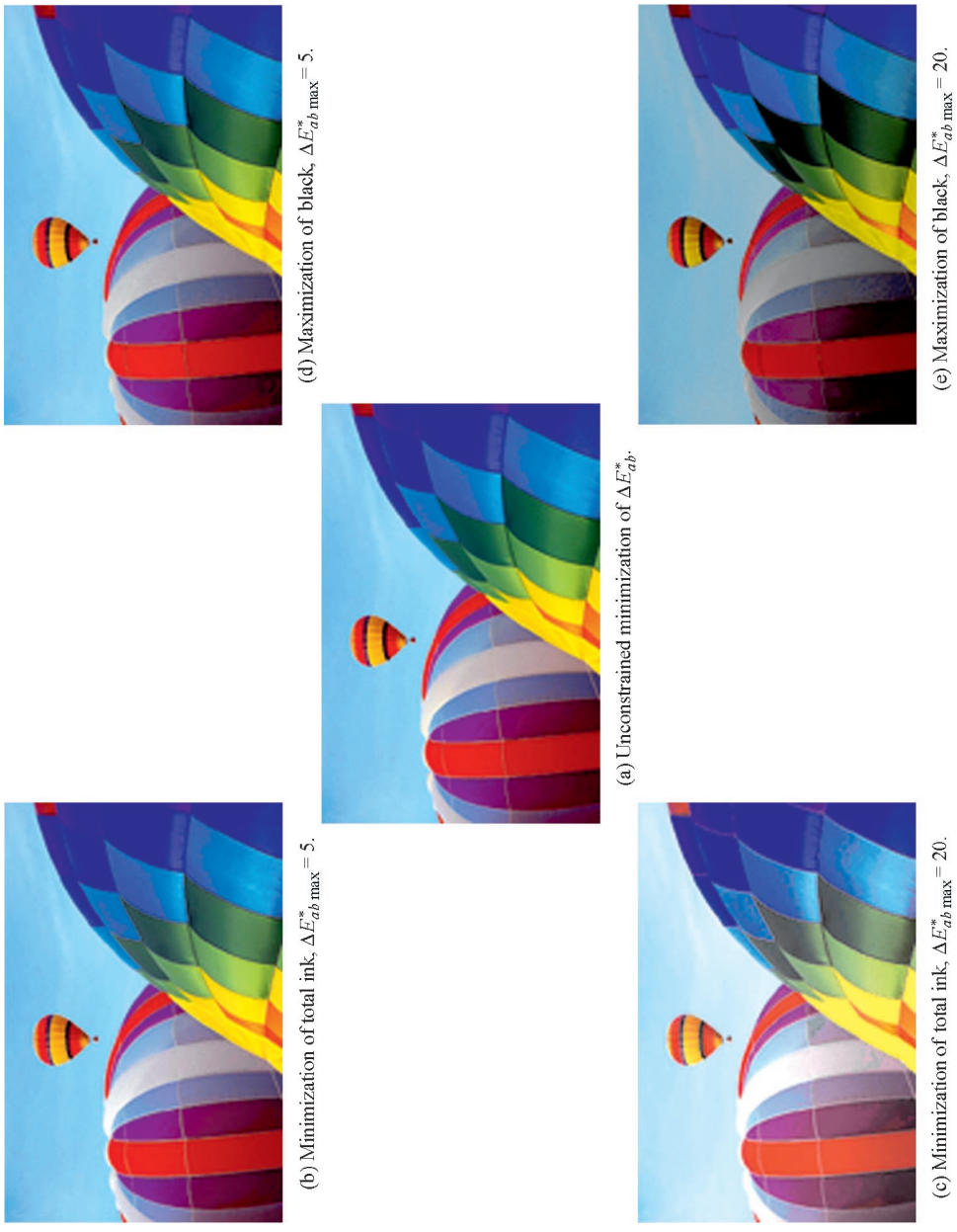
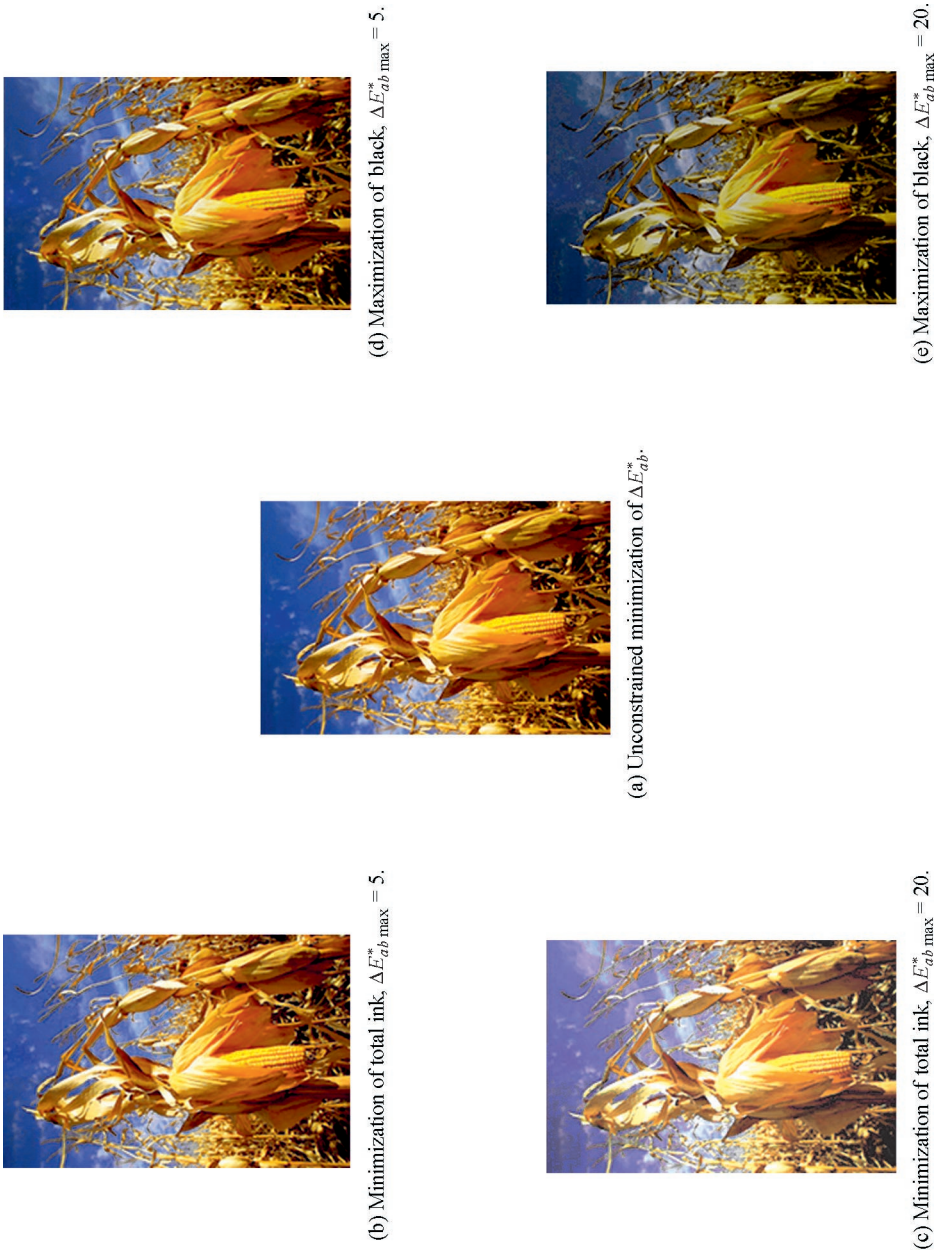


Fig. 17. Balloon prints obtained using NeuralColor.



(a) Unconstrained minimization of ΔE^*_{ab}.

(b) Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$.

(c) Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$.

(d) Maximization of black, $\Delta E^*_{ab\,\max} = 5$.

(e) Maximization of black, $\Delta E^*_{ab\,\max} = 20$.

Fig. 18. Corn prints obtained using NeuralColor.

(d) Maximization of black, $\Delta E^*_{ab\,\max} = 5$.

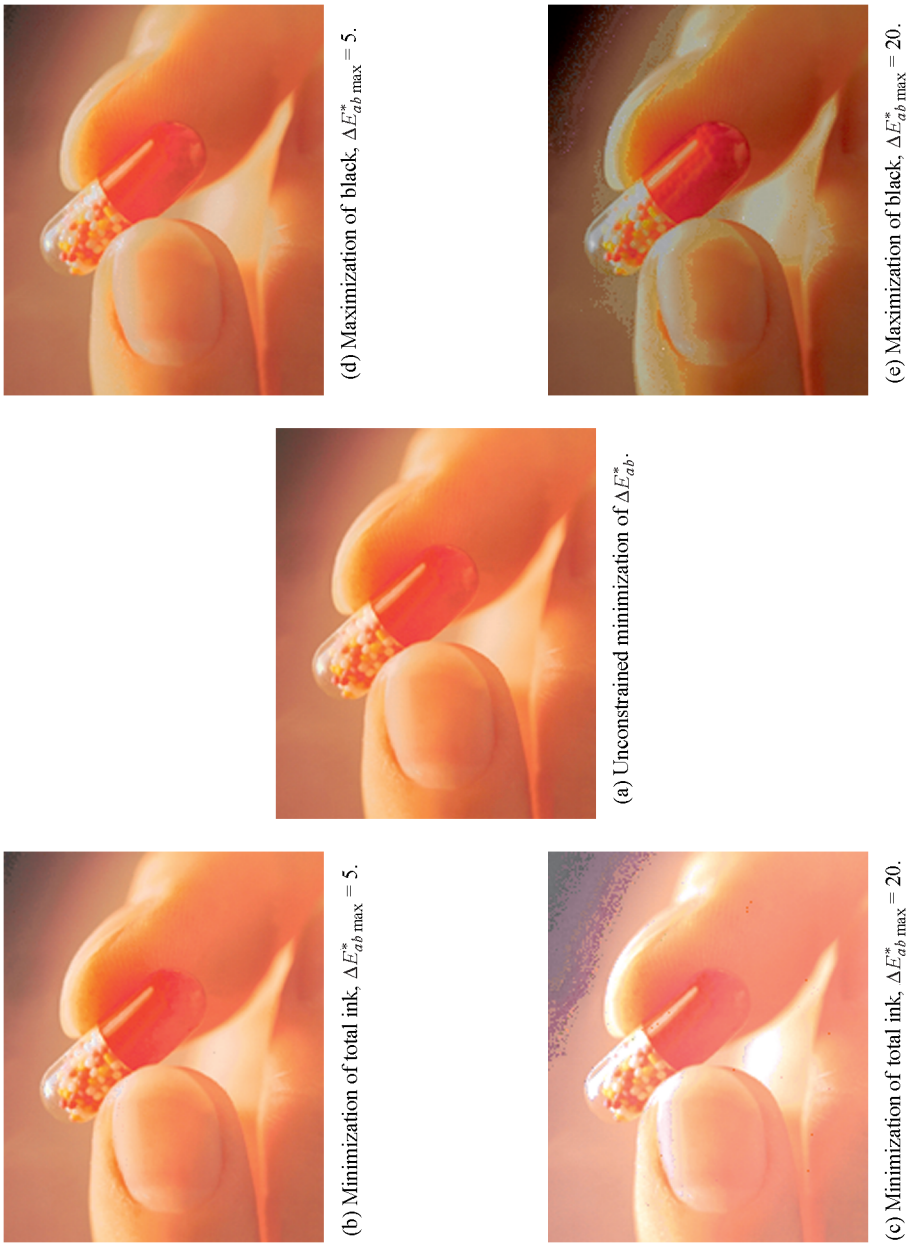
(b) Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$.

(a) Unconstrained minimization of ΔE^*_{ab}.

(e) Maximization of black, $\Delta E^*_{ab\,\max} = 20$.

(c) Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$.

Fig. 19. Flower prints obtained using NeuralColor.



(d) Maximization of black, $\Delta E^*_{ab\,\max} = 5$.

(e) Maximization of black, $\Delta E^*_{ab\,\max} = 20$.

(a) Unconstrained minimization of ΔE^*_{ab}.

(b) Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$.

(c) Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$.

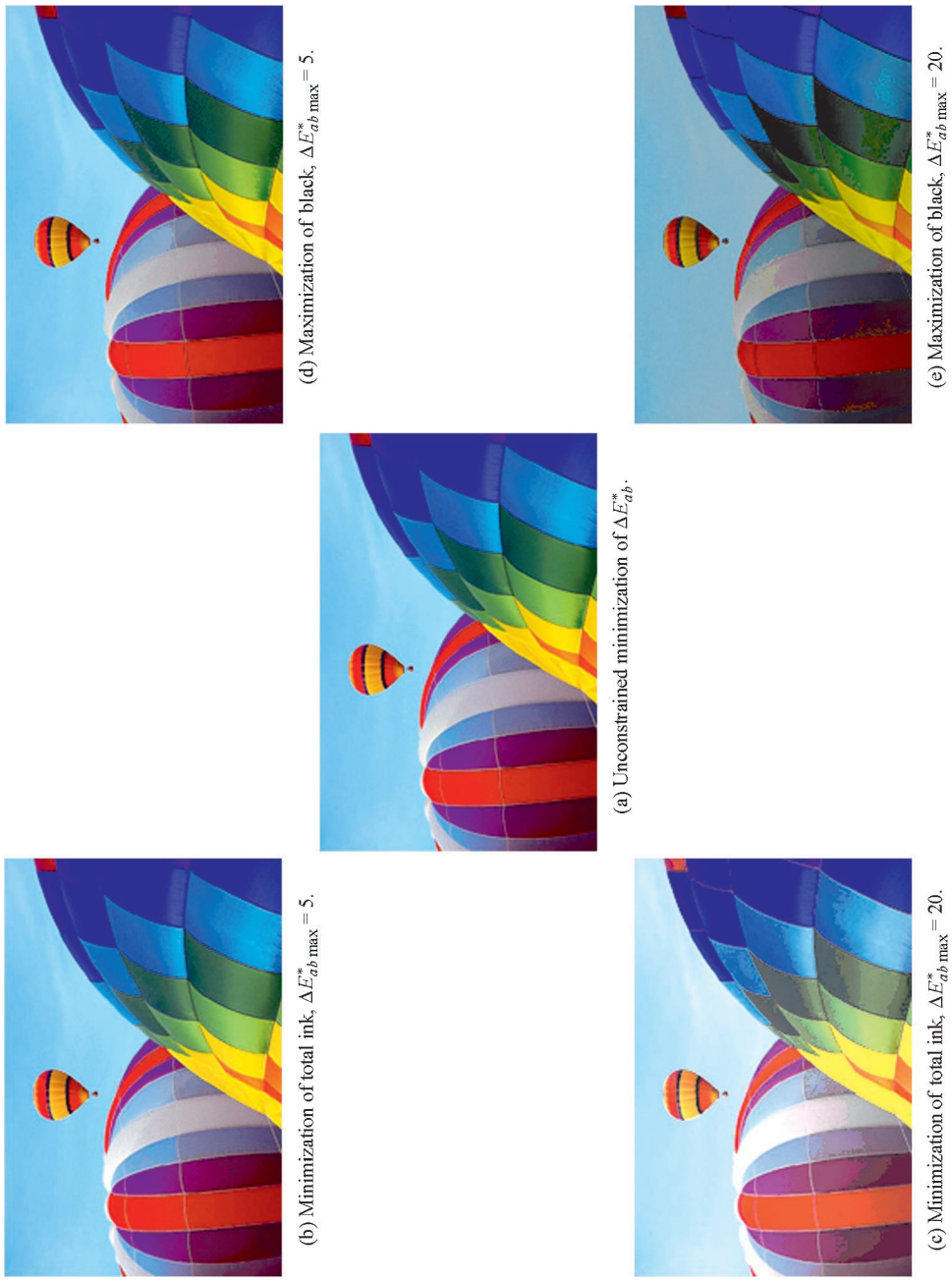
Fig. 20. Capsule prints obtained using NeuralColor.

C. IMAGES CONVERTED USING OPTINTERPOL

This appendix presents four images, each converted using five different conversion objectives, for the purpose of qualitative analysis. Image conversion parameters were chosen to illustrate unconstrained minimization of ΔE^*_{ab}, as well as minimization of total ink and maximization of black ink with both a tight and a loose constraint on $\Delta E^*_{ab\,\max}$. Table IX lists average combined ink deposition (C + M + Y + K) for each conversion.

Table IX. Ink Usage for Images Converted using OptInterpol

Image	Conversion Objective	Average Ink Deposition per Pixel (C + M + Y + K)
Balloons	Unconstrained minimization of ΔE^*_{ab}	1.22
Balloons	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.16
Balloons	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	0.84
Balloons	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.23
Balloons	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.28
Corn	Unconstrained minimization of ΔE^*_{ab}	1.67
Corn	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.55
Corn	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	1.01
Corn	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.70
Corn	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.72
Flower	Unconstrained minimization of ΔE^*_{ab}	1.70
Flower	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.29
Flower	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	0.85
Flower	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.42
Flower	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.25
Capsule	Unconstrained minimization of ΔE^*_{ab}	1.28
Capsule	Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$	1.14
Capsule	Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$	0.77
Capsule	Maximization of black ink, $\Delta E^*_{ab\,\max} = 5$	1.35
Capsule	Maximization of black ink, $\Delta E^*_{ab\,\max} = 20$	1.66



(d) Maximization of black, $\Delta E^*_{ab\,\max} = 5$.

(e) Maximization of black, $\Delta E^*_{ab\,\max} = 20$.

(a) Unconstrained minimization of ΔE^*_{ab}.

(b) Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$.

(c) Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$.

Fig. 21. Balloon prints obtained using OptInterpol.

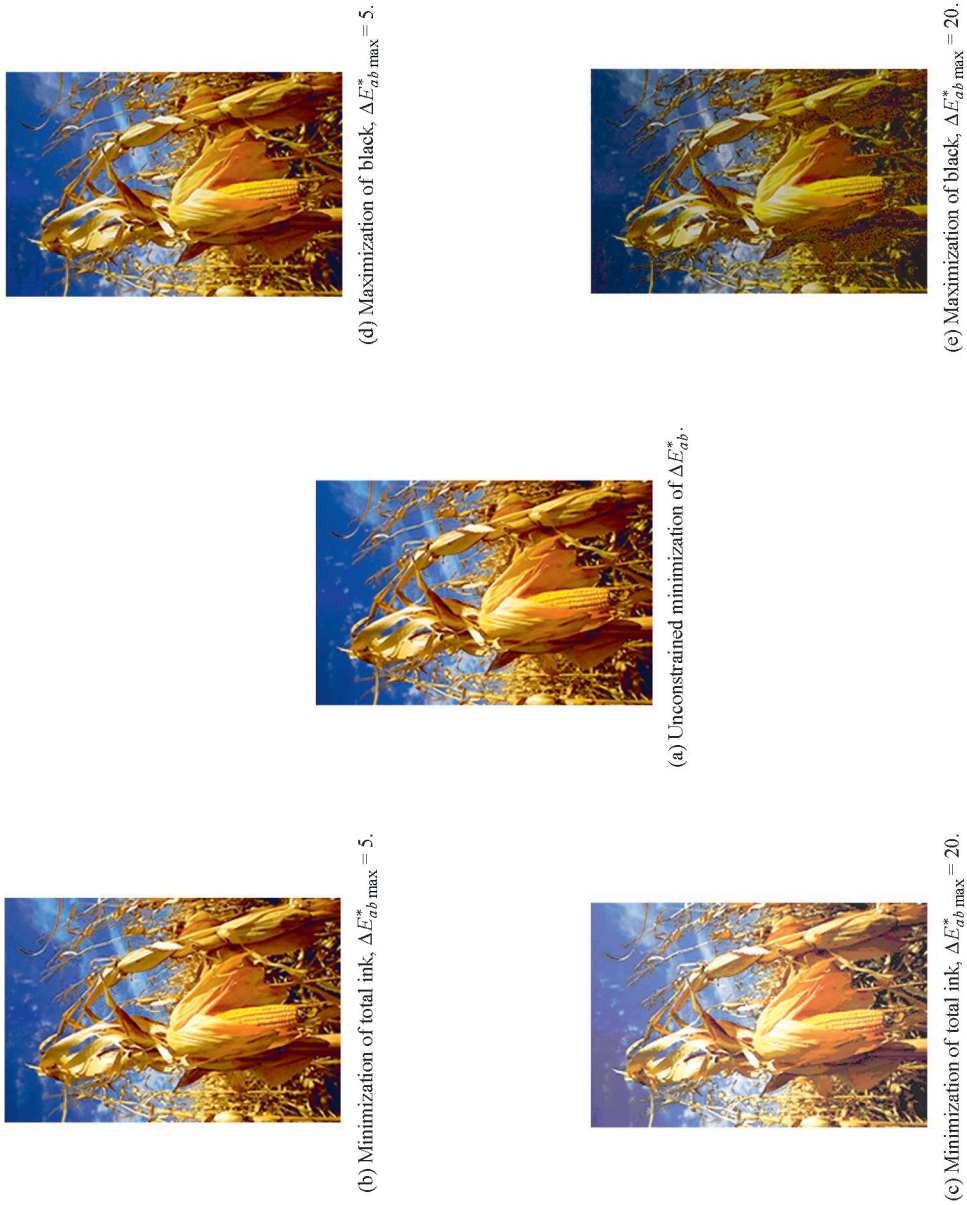


Fig. 22. Corn prints obtained using OptInterpol.

(d) Maximization of black, $\Delta E^*_{ab\,\max} = 5$.

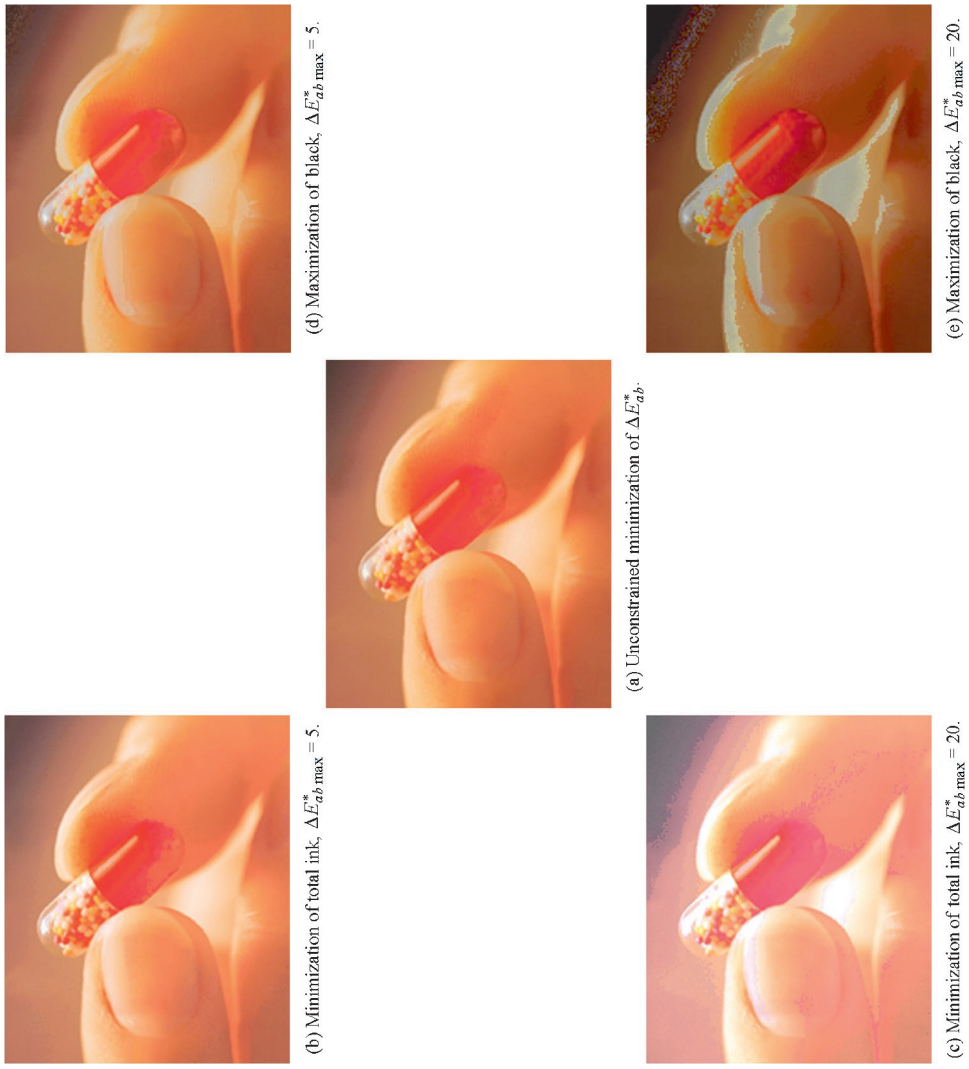
(e) Maximization of black, $\Delta E^*_{ab\,\max} = 20$.

(a) Unconstrained minimization of ΔE^*_{ab}.

(b) Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$.

(c) Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$.

Fig. 23. Flower prints obtained using OptInterpol.



(d) Maximization of black, $\Delta E^*_{ab\,\max} = 5$.

(b) Minimization of total ink, $\Delta E^*_{ab\,\max} = 5$.

(a) Unconstrained minimization of ΔE^*_{ab}.

(e) Maximization of black, $\Delta E^*_{ab\,\max} = 20$.

(c) Minimization of total ink, $\Delta E^*_{ab\,\max} = 20$.

Fig. 24. Capsule prints obtained using OptInterpol.

ACKNOWLEDGMENTS

Professor J. L. Zable, Mr. W. C. Decker Jr., and Dr. H. C. Lee are gratefully acknowledged for their support throughout this study.

REFERENCES

ABE, S. AND MARCU, G. 1994. A neural network approach for RGB to YMCK color conversion. In *Proceedings of IEEE Region 10's Ninth Annual International Conference: Frontiers of Computer Technology*. IEEE, 6–9.

Adobe Photoshop 1996. Photoshop 4.0 application cd.

ANSI. 1993. Graphic technology—input data for characterization of 4-color process printing. ANSI IT8.7/3-1993.

ARAI, Y., NAKANO, Y., AND IGA, T. 1993. A method of transformation from CIELAB to CMY value by a three-layered neural network. In *IST and SID's Color Imaging Conference: Transforms and Transportability of Color*, 41–44.

BALASUBRAMANIAN, R. AND DALAL, E. 1997. A method for quantifying the color gamut of an output device. In *Color Imaging: Device-Independent Color, Color Hard Copy, and Graphic Arts II*. Vol. 3018. SPIE, 110–116.

BARBER, C., DOBKIN, D., AND HUHDANPAA, H. 1996. The quickhull algorithm for convex hulls. *ACM Trans. Math. Softw. 22*, 4 (Dec.).

BARBER, C. AND HUHDANPAA, H. 1998. Qhull version 2.5. For access to the program and documentation the following URL is provided: http://www.geom.umn.edu/software/qhull/.

BEER, A. 1852. Bestimmung der absorption des rothen lichts in farbigen flussigkeiten. *Ann. Phys. Chem. 86*, 2.

BELYTSCHKO, T., KRONGAUZ, Y., ORGAN, D., FLEMING, M., AND KRYSL, P. 1996. Meshless methods: An overview and recent developments. *Comput. Meth. Appl. Mech. Eng. 139*, 1–47.

BERNS, R. 1996. Methods for characterizing CRT displays. *Displays 16*, 4, 173–182.

BIRKENSHAW, J., SCOTT-TAGGART, M., AND TRITTON, K. 1986. The black printer. *Proc. Tech. Assoc. Graph. Arts 38*, 403–429.

BOUGUER, P. 1729. *Essai d'optique sur la gradation de la lumiere*. Claude Tombert, Paris.

CIE. 1986. Colorimetry, second edition. Vienna, Austria. Publication CIE No. 15.2.

CLAPPER, F. AND YULE, J. 1953. The effect of multiple internal reflections of the densities of half-tone prints on paper. *J. Optical Soc. Am. 43*, 7 (July), 600–603.

CLAPPER, F. AND YULE, J. 1955. Reproduction of color with halftone images. In *Proc. Tech. Assoc. Graph. Arts*. TAGA, 1–14.

COWAN, W. 1983. An inexpensive scheme for calibration of a colour monitor in terms of CIE standard coordinates. *Comput. Graph. 17*, 3 (July), 315–321.

COWAN, W. 1987. CIE calibration of video monitors. Tech. rep., Division of Physics, National Research Council of Canada, Ottawa, Ontario.

DENNIS, J. J. AND SCHNABEL, R. 1983. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

DRAKOPOULOS, P. 1997a. Color printer characterization using optimization theory and artificial neural networks. M.S. dissertation, University of Colorado Department of Mechanical Engineering, Boulder, Colorado.

DRAKOPOULOS, P. 1997b. Color printer characterization using optimization theory and neural networks. Provisional U.S. Patent 60/066844.

DRAKOPOULOS, P. 1998. Color printer characterization using optimization theory and neural networks. U.S. Patent 09/199626.

FIELD, G. 1986. Color variability associated with printing GCR color separations. *Proceedings Tech. Assoc. Graph. Arts 38*, 145–157.

GENTILE, R., WALOWIT, E., AND ALLEBACH, J. 1990. A comparison of techniques for color gamut mismatch compensation. *J. Imag. Tech. 16*, 5 (Oct.), 176–181.

GORDON, J., HOLUB, R., AND POE, R. 1987. On the rendition of unprintable colors. *Proceedings of the Tech. Assoc. Graph. Arts 39*, 186–195.

GORI, M. AND TESI, A. 1992. On the problem of local minima in backpropagation. *IEEE Trans. Pattern Anal. Mach. Intell. 14*, 1 (Jan.), 76–86.

HAYKIN, S. 1999. *Neural Networks A Comprehensive Foundation*. Prentice Hall, Upper Saddle River, NJ.

HEUBERGER, K., JING, Z., AND PERSIEV, S. 1992. Color transformations and lookup tables. In *Proc. Tech. Assoc. Graph. Arts*. TAGA, 863–881.

HOLUB, R. AND KEARSLEY, W. 1989. Color to colorant conversions in a colorimetric separation system. In *Neugebauer Memorial Seminar on Color Reproduction*. Vol. 1184. SPIE, 24–35.

HOLUB, R., PEARSON, C., AND KEARSLEY, W. 1989. The black printer. *J. Imag. Tech. 15*, 4 (Aug.), 149–158.

HORNIK, K., STINCHCOMBE, M., AND WHITE, H. 1989. Multilayer feedforward networks are universal approximators. *Neural Networks 2*, 183–192.

HUNG, P. 1992. Tetrahedral division technique applied to colorimetric calibration for imaging media. *Proc. Soc. Imag. Sci. Tech.*, 419–422.

HUNG, P. 1993. Colorimetric calibration in electronic imaging devices using a look-up-table model and interpolations. *J. Elect. Imaging 2*, 1 (Jan.), 53–61.

HUNT, R. 1991. *Measuring Colour, Second Edition*. Ellis Horwood Limited, West Sussex, England.

HUNTER, R. AND HAROLD, R. 1987. *The Measurement of Appearance, Second Edition*. John Wiley and Sons, New York, NY.

ICC. 1998. Specification ICC.1:1998-09 file format for color profiles. http://www.color.org/.

IINO, K. AND BERNS, R. 1998. Building color management modules using linear optimization I. desktop color system. *J. Imaging Sci. Tech. 42*, 1 (January/February), 79–94.

JENNINGS, E., HOLLAND, R., AND LEE, C. 1994. Error analysis of lookup table implementations in device-independent color imaging systems. In *Device-independent color imaging*. Vol. 2170. SPIE, 98–107.

JOHNSON, A. 1996. Methods for characterizing colour printers. *Displays 16*, 4, 193–202.

JUNG, E. 1984. Programmed and complementary color reduction. *Proc. Tech. Assoc. Graph. Arts 36*, 135–150.

KANAMORI, K. 1999. A study on interpolation errors and ripple artifacts of 3D lookup table method for nonlinear color conversions. In *Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts IV*. Vol. 3648. SPIE, 167–178.

KANAMORI, K. AND KOTERA, H. 1992. Color correction technique for hard copies by 4-neighbors interpolation method. *J. Imaging Sci. Tech. 36*, 1, 73–80.

KANG, H. AND ANDERSON, P. 1992. Neural network applications to the color scanner and printer calibrations. *J. Elect. Imaging 1*, 2 (Apr.), 125–135.

KANG, H. 1993. Comparisons of color mixing theories for use in electronic printing. In *Color Imaging Conference: Transforms & Trasportability of Color*. IS&T/SID, 78–82.

KANG, H. 1994. Applications of color mixing models to electronic printing. *J. Elect. Imaging 3*, 3 (July), 276–287.

KANG, H. 1995a. Comparisons of three-dimensional interpolation techniques by simulations. In *Device-independent color imaging II*, E. Walowit, Ed. Vol. 2414. SPIE, 104–114.

KANG, H. 1995b. Printer-related color processing techniques. In *Color Hard copy and Graphic Arts IV*. Vol. 2413. SPIE, 410–419.

KASSON, J. 1994. Tetrahedral interpolation algorithm accuracy. In *Device-independent color imaging*. Vol. 2170. SPIE, 24–35.

KASSON, J., NIN, S., PLOUFFE, W., AND HAFNER, J. 1995. Performing color space conversions with three-dimensional linear interpolation. *J. Elect. Imaging 4*, 3 (July), 226–250.

KASSON, J. AND PLOUFFE, W. 1992. An analysis of selected computer interchange color spaces. *ACM Trans. Graph. 11*, 4 (Oct.), 373–405.

KIM, C., KWEON, I., AND SEO, Y. 1997. Color and printer models for color halftoning. In *Selected Papers on Digital Halftoning*, J. Allebach, Ed. Vol. MS 154. SPIE, 710–724.

Kodak. Kodak. http://www.Kodak.com/.

KUBELKA, P. 1948. New contributions to the optics of intensely light-scattering materials. part I. *J. Opt. Soc. Am. 38*, 448–457.

KUBELKA, P. AND MUNK, F. 1931. Ein beitrag zur optik der farbanstriche. *Z. Techn. Physik 12*, 593.

LESHNO, M., LIN, V., PINKUS, A., AND SCHOCKEN, S. 1993. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks 6*, 861–867.

LITTLEWOOD, D. 2001. Printer color management using Pareto-optimization and efficient calibration techniques. Ph.D. dissertation, University of Colorado Department of Mechanical Engineering, Boulder, Colorado.

LOOTSMA, F. A. 1985. Comparative performance evaluation, experimental design and generation of test problems in nonlinear optimization. In *Computational Math Programming*, K. Schittkowski, Ed. Springer-Verlag.

MARCU, G. AND IWATA, K. 1993. RGB-YMCK color conversion by application of the neural networks. In *Color Imaging Conference: Transforms & Trasportability of Color*. IS&T/SID, 27–32.

MCCAMY, C., MARCUS, H., AND DAVIDSON, J. 1976. A color-rendition chart. *J. App. Photog. Eng. 2*, 3 (summer), 95–99.

MONGEON, M. 1996. Image transformation into device dependent color printer description using 4th-order polynomial regression and object oriented programming development of image processing modules. In *Color Imaging: Device-independent Color, Color Hard Copy, and Graphic Arts*. SPIE, 341–352.

MORE, J., GARBOW, B., AND HILLSTROM, K. 1980. User guide for minpack-1. Argonne National Laboratory Report ANL-80-74.

NAKAUCHI, S., HATANAKA, S., AND USUI, S. 1999. Color gamut mapping based on a perceptual image difference measure. *Color Research and Application 24*, 4 (Aug.), 280–291.

NAKAUCHI, S., IMAMURA, M., AND USUI, S. 1998. Color gamut mapping by minimizing perceptual differences between images. *Systems and Computers in Japan 29*, 10, 46–55.

NEUGEBAUER, H. 1937. Die theoretischen grundlagen des mehrfarbenbuchdrucks. *Zeitschrift fur wissenschaftliche Photographie Photophysik und Photochemie 36*, 4 (Apr.), 73–89.

NIN, S., KASSON, J., AND PLOUFFE, W. 1992. Printing CIELAB images on a CMYK printer using tri-linear interpolation. In *Color Hard Copy and Graphic Arts*, J. Bares, Ed. Vol. 1670. SPIE, 316–324.

POWER, J., WEST, B., STOLLNITZ, E., AND SALESIN, D. 1996. Reproducing color images as duotones. In *Proceedings of SIGGRAPH 96, in Computer Graphics Proceedings, Annual Conference Series*, 237–248.

PRAEFCKE, W. 1999. Robust and fast numerical color separation for mathematical printer models. In *IS&T/SPIE Conference on Color Imaging: Device-independent Color, Color Hardcopy, and Graphic Arts IV*. Vol. 3648. SPIE, 2–10.

ROLLESTON, R. AND BALASUBRAMANIAN, R. 1993. Accuracy of various types of neugebauer model. In *Color Imaging Conference: Transforms & Trasportability of Color*. IS&T/SID, 32–37.

RUSTEM, B. 1998. *Algorithms for Nonlinear Programming and Multiple-objective Decisions*. John Wiley and Sons, New York, NY.

SAYANAGI, K. 1987. Black printer, UCR and UCA—gray component replacement—. *Proceedings of the Technical Association of the Graphic Arts 39*, 711–724.

SCHITTKOWSKI, K. 1985. NLPQL: A FORTRAN subroutine solving constrained nonlinear programming problems. *Ann. Oper. Res. 5*, 485–500.

SCHWARTZ, M., HOLUB, R., AND GILBERT, J. 1985. Measurements of gray component reduction in neutrals and saturated colors. *Proc. Tech. Assoc. Graph. Arts 37*, 16–27.

SONTAG, E. AND SUSSMANN, H. 1989. Backprogagation can give rise to spurious local minima even for networks without hidden layers. *Complex Systems 3*, 1 (Feb.), 91–106.

STADLER, W. 1988. *Multi Criteria Optimization in Engineering and the Sciences*. Plenum Press.

STOLLNITZ, E., OSTROMOUKHOV, V., AND SALESIN, D. 1998. Reproducing color images using custom inks. In *Proceedings of SIGGRAPH 98, in Computer Graphics Proceedings, Annual Conference Series*.

STONE, M., COWAN, W., AND BEATTY, J. 1988. Color gamut mapping and the printing of digital color images. *ACM Trans. Graph. 7*, 4 (Oct.), 249–292.

SWOP. 1988. Recommended specifications web offset publications.

TOMINAGA, S. 1993. A neural network approach to color reproduction in color printers. In *IST and SID's Color Imaging Conference: Transforms and Transportability of Color*, 173–177.

TOMINAGA, S. 1996. Color control using neural networks and its application. In *Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts*. Vol. 2658. SPIE, 253–260.

TOMINAGA, S. 1998a. Color conversion using neural networks. In *Color Imaging: Device-Independent Color, Color Hardcopy, and Graphic Arts III*. Vol. 3300. SPIE, 66–75.

TOMINAGA, S. 1998b. Color scheme for printers using more than three color inks. In *Electronic Imaging: Processing, Printing, and Publishing in Color*. Vol. 3409. SPIE, 286–293.

VACHON, G. 1988. Modeling the mixing behavior of inks with polynomials. *Color Research and Application 13*, 1 (Feb.), 46–49.

WALLNER, D. 1998. Building ICC profiles—the mechanics and engineering. This document is available for downloading at www.color.org, the web site for the International Color Consortium.

WIDDEL, H. AND POST, D., Eds. 1992. *Color in Electronic Displays*. Plenum Press, New York, NY.

WYSZECKI, G. AND STILES, W. 1982. *Color Science: Concepts and Methods, Quantitative Data and Formula*. John Wiley and Sons, New York, NY.

XU, L. AND HOLUB, R. 1992. Color spaces for image representation and image processing. *Proc. Tech. Assoc. Graph. Arts 45*, 1, 114–137.

YULE, J. 1967. *Principles of Color Reproduction*. John Wiley and Sons, New York, NY.

YULE, J. AND NEILSEN, W. 1951. The penetration of light into paper and its effect on halftone reproduction. In *Proc. Tech. Assoc. Graph. Arts*. TAGA, 65–76.

ZIENKIEWICZ, O. AND TAYLOR, R. 1989. *The Finite Element Method*. McGraw-Hill, New York, NY.

Received January 2000; revised August 2001; accepted November 2001