Journal of Electronic Imaging 7(3), 664–671 (July 1998).

Color control of printers by neural networks

Shoji Tominaga
Osaka Electro-Communication University
Department of Engineering Informatics
Neyagawa, Osaka 572, Japan
E-mail: shoji@tmlab.osakac.ac.jp

Abstract. *A method is proposed for solving the mapping problem from the three-dimensional color space to the four-dimensional CMYK space of printer ink signals by means of a neural network. The CIE-L*a*b* color system is used as the device-independent color space. The color reproduction problem is considered as the problem of controlling an unknown static system with four inputs and three outputs. A controller determines the CMYK signals necessary to produce the desired L*a*b* values with a given printer. Our solution method for this control problem is based on a two-phase procedure which eliminates the need for UCR and GCR. The first phase determines a neural network as a model of the given printer, and the second phase determines the combined neural network system by combining the printer model and the controller in such a way that it represents an identity mapping in the L*a*b* color space. Then the network of the controller part realizes the mapping from the L*a*b* space to the CMYK space. Practical algorithms are presented in the form of multilayer feedforward networks. The feasibility of the proposed method is shown in experiments using a dye sublimation printer and an ink jet printer.* © 1998 SPIE and IS&T. [S1017-9909(98)02003-0]

1 Introduction

The concept of device-independent color reproduction of images has received widespread attention since the advent of desk-top publishing systems. Color reproduction requires color conversion between the color signals, depending on the device used, and the standard color coordinates like the CIE-$L^*a^*b^*$ color system[1,2] representing color appearance. Color reproduction on a CRT monitor is based on an additive mixture of three primaries, for which color conversion is usually performed with a matrix for the primary transformation and a look-up table for the nonlinear correction. On the other hand, the color reproduction of print is based on the subtractive mixture of the three primaries Cyan (C), Magenta (M), and Yellow (Y), or four primaries with the addition of Black (K). It is usually difficult to predict from the knowledge of the amounts of ink for these primaries what color stimulus will be generated on paper. In other words, because of the complicated nonlinear relationship between the input and output signals of a printer, it is difficult to control the CMYK color signals in a simple operation using a matrix and a look-up table.

Paper 97-005 received Feb. 3, 1997; revised manuscript received Oct. 10, 1997; accepted for publication Oct. 17, 1997.
1017-9909/98/$10.00 © 1998 SPIE and IS&T.

Several methods for estimating the amounts of the primary inks necessary to produce a desired color stimulus have been proposed. These are (1) an analytical method using the Neugebauer equations, (2) matrix transformations, (3) a look-up table and three-dimensional interpolation, and (4) neural network methods. The analytical method often suffers from an inevitable discrepancy between the printer outputs and the model equations, and a precise color reproduction can not be expected. In (2)–(4), a color printer is regarded as a black box. Method (2) describes the input-output relationship by a matrix with nonlinear elements. Method (3) makes a data table of measured color values and interpolates this table to determine the input signal for generating a desired color output.[3,4] Method (4) models the mapping between the printer color signal and the output color stimulus values.[5,6]

A neural network is suitable for modeling a nonlinear transformation between two color spaces, for which a mathematical description is difficult to provide. In fact, a neural network can describe much more complex nonlinearity than a polynomial expression. This ability may motivate us to use a neural network for modeling the printer's nonlinearity based on the subtractive color mixture. Such a network consists of collections of connected processing units (neurons). It can adaptively learn the mapping and store this knowledge in the network structure. The mapping from one space to another is then realized by using this simple structure in which nonlinear units are linked in parallel and in layers. Thus the mapping is realized without keeping a large data table such as the one used in method (3).

Previous neural network methods were developed for printers using the three color signals CMY as printer primaries. However, the four inks of CMYK are needed for high quality color reproduction. In this case, the conversion from color stimulus values such as CIE-$L^*a^*b^*$ or CIE-XYZ to the printer color signals CMYK is not unique because a mapping from a three-dimensional color space to a higher four-dimensional space has to be determined. Previous network approaches failed to realize such a mapping.

The present paper describes a method for solving the mapping problem from the three-dimensional color space of color stimuli to the higher dimensional color space of printer signals. The CIE-$L^*a^*b^*$ color system is used as the device-independent color space. The mapping from the $L^*a^*b^*$color space to the printer CMYK color space is

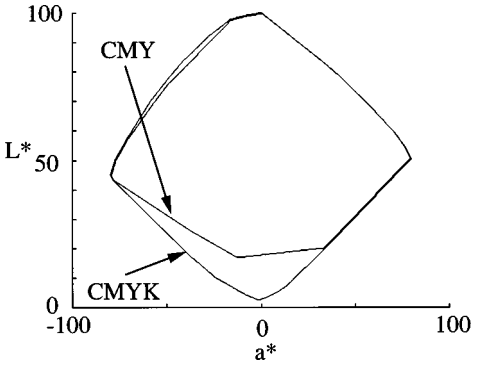

Fig. 1 Color gamut of four-color printing in comparison with three-color printing for a dye sublimation printer.

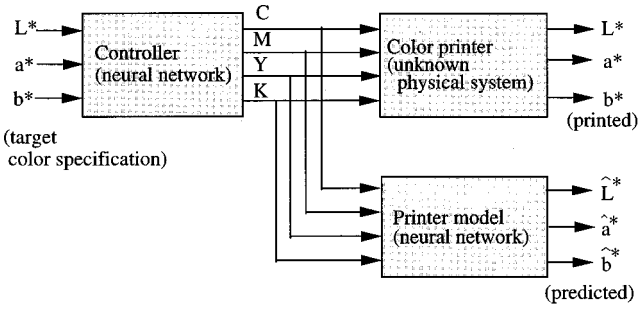

Fig. 2 Color control of a printer by neural networks.

constructed using a neural network. This mapping does not use such techniques as undercolor removal (UCR) and gray component replacement (GCR), which are often used in methods in category (3).[7]

We view a color printer as an unknown static system with four inputs and three outputs. The color reproduction problem can then be considered as the problem of controlling a system with unknown characteristics. We determine the CMYK values of the input control signals such that the printer system outputs the desired $L^*a^*b^*$ values. A system for the signal determination is called the controller in this paper. The controller attempts to realize an inverse mapping of the unknown system.

This paper is organized as follows. Section 2 describes the basic concepts for color control of a printer. Section 3 presents practical algorithms implemented as multilayered neural networks. Experimental results are shown in Sec. 4. Conclusions are given in Sec. 5.

2 Basic Concepts of Color Control

In developing a theoretical treatment of four-color printing, there is no unique solution.[8] In three-color printing, there is only one combination of the three inks which will match the color of the original, but in four-color printing, a unique solution is obtained only if some assumption about the relationship between the amounts of black ink and the inks of the other three colors is made. In this paper, we neither make assumptions about this relationship nor any other constraints on the ink amounts. It is even assumed that the amounts of CMYK inks can be controlled independently without using the conventional UCR and GCR techniques.

Figure 1 shows a typical example of the color gamut of four-color printing in comparison with the gamut of three-color printing. These are the gamuts for a nonimpact printer of the dye sublimation type. Each color printer primary value is expressed as having 256 levels (8-bits). When the four-dimensional printer inputs CMYK were varied in the full range [0,0,0,0]−[255,255,255,255], the measurements of color patches on the printer outputs were specified in the CIE-$L^*a^*b^*$ color space. Note that the reproducible color region in four-color printing is expanded into the dark part of the lightness dimension, as is indicated on the (L^*,a^*) plane in Fig. 1.

Color reproduction requires determining the digital values of the ink signals CMYK necessary to generate any desired $L^*a^*b^*$ color specification within the gamut as shown in Fig. 1, that is, to realize the mapping from the CIE-$L^*a^*b^*$ color space to the printer CMYK color space. Neural network methods[5,6] have been applied to this task, but the associated iterative learning algorithm for determining the network parameters does not converge, because the necessary mapping scheme is a projection from a three-dimensional space to a four-dimensional space. Mathematically, there is no one-to-one correspondence in such cases. The previous methods are only applicable to establishing a one-to-one correspondence between the CMY printer space and the three-dimensional color space within the gamut.

We consider the color reproduction problem as the problem of controlling a system with unknown characteristics (see Fig. 2). The color printer is an unknown physical system with four inputs and three outputs. A controller, which is constructed with a neural network, tries to realize the inverse mapping of the system, i.e., the controller transforms the target $L^*a^*b^*$ values into the CMYK values, which are then transformed into the actual $L^*a^*b^*$ values by the unknown printing system. Therefore, the controller must determine the printer ink signals in such a way that the color stimulus errors between the targets and the printed colors are minimized in the $L^*a^*b^*$ color space. In this process, a model is used for the physical printer. The printer model is constructed as a neural network with the same input-output relationship as the physical printer. We propose a procedure for finding this control solution based on the following two-phase procedure:

The first phase is devoted to identifying the unknown printer system [Fig. 3(a)]. The printer system is modeled by a neural network, that is, the neural network is trained to act like a printer. We first measure the $L^*a^*b^*$ values of a set of color patches which were printed with the printer and which correspond to grid points sampled uniformly in the whole CMYK space. The data set of these measured $L^*a^*b^*$ values and their corresponding CMYK values from the printer inputs is used as the training data for the neural network. The printer model is determined by a learning procedure which is based on minimizing the color difference between the target $L^*a^*b^*$ values in the training data and the corresponding $L^*a^*b^*$ values ($\hat{L}^*\hat{a}^*\hat{b}^*$) predicted by the network.

The second phase is devoted to the determination of a combined neural network system (combined NN system) including the controller and the printer model [Fig. 3(b)].

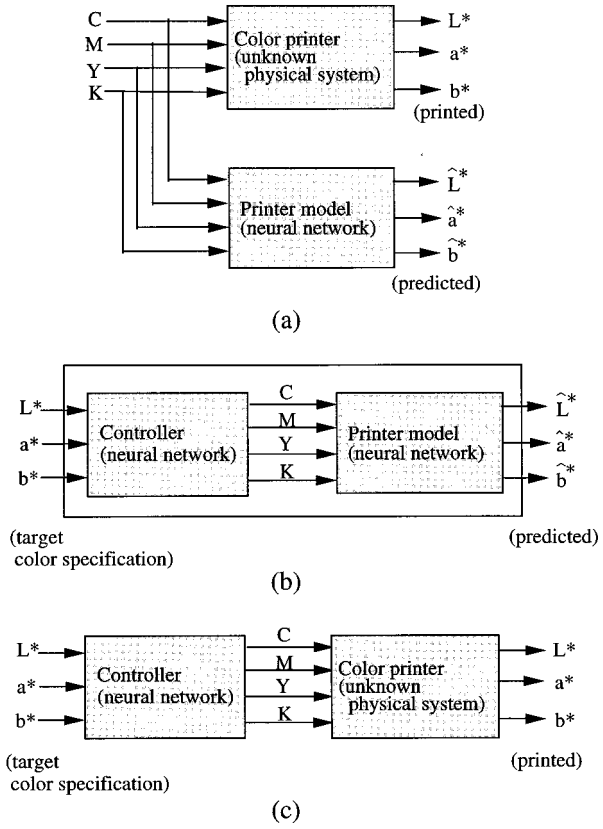

Fig. 3 Procedure for finding a solution for the color control. (a) Creating a model of the unknown printing system in the first phase. (b) Optimization of the controller within the combined NN system in the second phase. (c) Final control of the printer.

Once the printer system has been learned in the first place, the control system for mapping a color specification vector into an ink vector can be determined using a neural network. We adopt an indirect approach for doing this. Let us consider the combined NN system as shown in Fig. 3(b). The CMYK values are the internal signals of the combined NN system. From a global point of view, this combined system represents a mapping from the target $L^*a^*b^*$ values to the predicted $L^*a^*b^*$ values describing the printer outputs. It should be noted that this mapping is always required to produce the same color values as the input color values. In mathematical terms, this is called a one-to-one identity mapping. Therefore, the combined NN system is trained to realize a one-to-one mapping. In this training, the network structure of the printer model and its network parameters are fixed, while the network parameters of the controller are being adjusted to minimize the error.

Figure 3(c) indicates the phase of actually controlling the physical printer. When the combined NN system completes the one-to-one mapping in the second phase, the network of the controller part must realize an inverse mapping of the printer model that is equivalent to the physical printer. Thus the controller can be used to determine the CMYK signals so that the desired $L^*a^*b^*$ color specifications are produced by the printer.

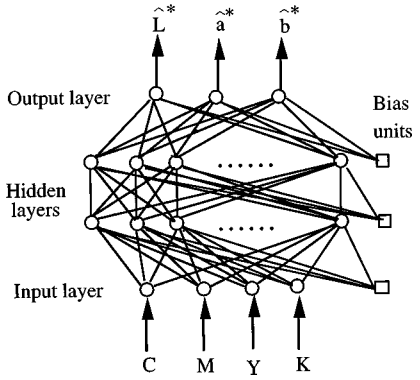

Fig. 4 Network structure of the printer model.

3 Practical Algorithms

3.1 *Determination of the Printer Model*

Figure 3 depicts the network structure used for modeling a printer. The mapping from the input space of the four ink signals CMYK to the output $L^*a^*b^*$ space is implemented as a four-layered neural network. Each of the two hidden layers has 10 units. The network size was determined by empirically selecting the most effective numbers of hidden layers and units. In Fig. 4, each unit receives its input signals from the prior layer, computes their weighted sum, and outputs the unit's level of activation by weighting this sum with a nonlinear function.

Let o_i be the output of the ith unit in the prior layer, w_{ji} be the weighting coefficient for the connection from the ith unit to the jth (target) unit, and b_j be the bias term of the unit. The input to the jth unit is then described as the sum of the weighted outputs from the prior layer

$$\text{net}_j = \sum_i w_{ji} o_i + b_j . \tag{1}$$

The nonlinear output of the jth unit is given by

$$o_j = f(\text{net}_j), \tag{2}$$

where f is the sigmoidal activation function

$$f(\text{net}_j) = 1/[1+\exp(-4\alpha\ \text{net}_j)]. \tag{3}$$

This function takes any real number in the interval [0,1], and the positive constant α represents the slope of f at net=0. The operations of (1) and (2) are executed at all units except the input layer.

All input/output signals of the network are normalized. First, the CMYK values, which lie in the region of 0 to 255, are scaled to the range [0,1]. Concerning the output signals, we assume that the ranges for three quantities L^*, a^*, and b^* are given as [0,100], [−100,100], and [−100,100], respectively. The color gamut of the physical printer is contained within the rectangular prism delimited by these ranges. All of these ranges are then normalized into the range [0,1] to form a cube.

Modeling a printer by such a 4-10-10-3 type network (Fig. 4) requires 170 weighting coefficients and 23 biases,

or a total of 193 parameters, to be adjusted. This set of parameters is determined through a learning procedure based on the measurement data. We adopt error backpropagation as the learning rule. The backpropagation method is well established for training multilayer feedforward networks. The method has been published in a large number of text books on neural networks.[9]

Now let us define the pth vector of the normalized $L^*a^*b^*$ values in the training data set as $\mathbf{t}_p \equiv [t_{p1}, t_{p2}, t_{p3}]$ and define the corresponding output vector of the network as $\mathbf{o}_p \equiv [o_{p1}, o_{p2}, o_{p3}]$. All the weights and biases in the network are then adjusted such as to minimize the squared error between the target $\mathbf{t}_p$ and the actual output $\mathbf{o}_p$, $E_p = \|\mathbf{t}_p - \mathbf{o}_p\|^2$. The change Δw_{ji} in the weight w_{ji} is computed by the following recursive algorithm including a momentum term:

$$\Delta w_{ji}(n+1) = \eta \delta_{pj} o_{pi} + \beta \Delta w_{ji}(n), \tag{4}$$

where n indicates the nth learning step. The notation $\Delta w_{ji}(n)$ represents the amount of correction in the weight from the ith unit to the jth unit in the next layer at the nth step. δ_{pj} is an error term in the jth unit, which is calculated recursively from the output layer by using the equations of error backpropagation (see Ref. 10). η and β are positive constants called the ''learning constant'' and ''momentum constant,'' respectively. The change Δb_j in the bias b_j is computed in the same manner as the other weights.

In the iterative learning process, the initial values $w_{ij}(0)$ and $b_j(0)$ of the weights and biases are set to random numbers. One period of presenting the entire training data to the network is defined as an epoch, and the mean squared error E_p over the data set is evaluated as the system error. Corrections of all the parameters are repeated for as many epochs as are necessary to decrease the system error to an acceptable level. The input/output signals of the network are normalized. Therefore, the network models the printer accurately when the system error becomes sufficiently small, that is, it realizes a system equivalent to the printer in the input-output relationship.

3.2 *Optimization of the Controller*

The network structure of the controller is assumed to have the same complexity as the printer model, because it should be modeling the inverse system of the printer. As shown in Fig. 5, the controller is constructed with a four-layered network of the 3-10-10-4 type with 3 units in the input and 4 units in the output.

First, the printer model and the controller network are combined into a single large network as shown in Fig. 5. This combined NN network is an eight-layered network, consisting of the input/output layers and six hidden layers. Note that the third and fourth hidden layers are double, since the same signals are copied simply from the lower layer to the upper one. Therefore, only five hidden layers are effective for the complete mapping. Both the input and output signals of the combined network are the normalized $L^*a^*b^*$ values in the range $[0,1]$, compressed from the color specifications in the regions of $0 \leqslant L^* \leqslant 100$, $-100 \leqslant a^* \leqslant 100$, and $-100 \leqslant b^* \leqslant 100$.

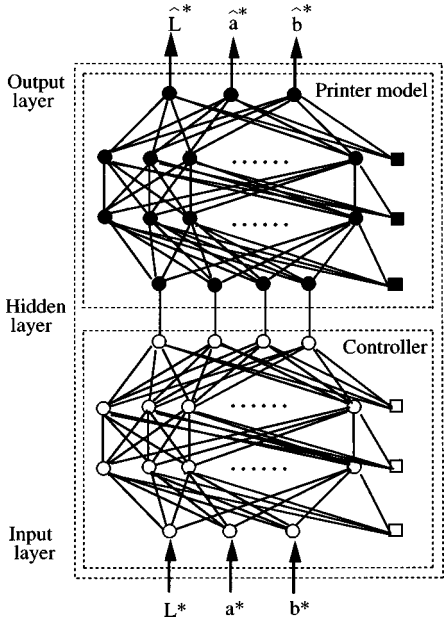

Fig. 5 Network structure for determining the identity mapping and the controller.

The above neural network is trained to realize an identity mapping from the $L^*a^*b^*$ color space to the $L^*a^*b^*$ color space. The computations in each unit are coincident with Eqs. (1)–(3). The network learning is performed according to the following procedure.

(1) Because of the one-to-one mapping, the same $L^*a^*b^*$ values are presented to both the input and output layers. CMYK values are not used as training data. Moreover, the $L^*a^*b^*$ values are not necessarily selected from the measured data set, but any $L^*a^*b^*$ values sampled in the color gamut can be used.

(2) The combined NN system is trained based on the error-backpropagation rule. At the same time, the weights and biases in the printer model part are kept fixed as previously estimated according to Sec. 3.1; they never change as $w_{ij}(n) = w_{ij}(0)$ and $b_j(n) = b_j(0)$. Thus, the parameters to be newly determined in the combined NN system are limited to a total of 194 parameters (170 weights and 24 biases). In Fig. 5, the filled circles and squares (●,■) indicate, respectively, the fixed units and biases, while the units and biases indicated by (○,□) are determined by training the combined network, starting from a random number set.

(3) The error is propagated backward through the combined network from the output layer to the input layer. The upper half of the network does not change its network parameters, and the error passes only through the hidden layers. On the other hand, the lower half corrects its parameters due to the recursive computations, accompanied by the error backpropagation.

The above learning procedure is repeated for the entire gamut in the $L^*a^*b^*$ color space. When the system error of the combined network becomes sufficiently small, the squared error

$$\|L^* - \hat{L}^*\|^2 + 4\|a^* - \hat{a}^*\|^2 + 4\|b^* - \hat{b}^*\|^2 \tag{5}$$

between the target $L^*a^*b^*$ values of color specifications and the $\hat{L}^*\hat{a}^*\hat{b}^*$ values predicted by the network is minimized. Because of the error minimization over the entire region, the combined network realizes the identity mapping of the $L^*a^*b^*$ color space.

With respect to the structure of the completed total network, we note that the printer model part in the upper half remains unchanged, while the controller part in the lower half is newly trained. Because the upper half represents the mapping from the CMYK space to the $L^*a^*b^*$ space, and moreover the combined NN system performs the one-to-one mapping, the lower half must realize the inverse mapping of the printer model part, i.e., the mapping from the $L^*a^*b^*$ space to the CMYK space. In other words, the desired controller of the printer is obtained by utilizing the lower half of the combined network. This controller can determine the four ink signals necessary to generate any target $L^*a^*b^*$ color specifications on the printer.

4 Experimental Results

In order to examine the feasibility of the proposed algorithms for color control, we have conducted experiments with a dye sublimation printer and an ink jet printer. The printing resolution was 300 dpi for the former printer, and 360 dpi for the latter. A digital halftoning technique for the ink jet printer was developed based on the error diffusion algorithm.[11]

4.1 *Printer Model*

First, we constructed a training data set from the measurements of many color patches printed in four inks. Each digital value of CMYK was set to 0, 32, 64, 96, 128, 160, 192, 224, and 255 for uniformly sampling the entire four-dimensional space, resulting in a total number of $9^4 = 6561$ color patches. The measured color samples were specified in the CIE-$L^*a^*b^*$ color space under the CIE illuminant D_{65}. The tristimulus values of the printing paper used as reference white were $X_0 = 84.2$, $Y_0 = 88.0$, $Z_0 = 102.1$ for the dye sublimation printer, and $X_0 = 81.4$, $Y_0 = 83.1$, $Z_0 = 105.7$ for the ink jet printer. Thus the training data set consisted of 6561 pairs of the input ink signals and their corresponding printer-output $L^*a^*b^*$ color values.

The networks were simulated on a SUN SPARC Station 20. The initial values of the weights $w_{ij}(0)$ and the biases $b_j(0)$ were set to random numbers. The training data were selected based on uniformly random sampling so that every data pair was selected only once in random order in each epoch. The learning rate and the momentum constants were set to $\eta = \beta = 0.09$ at the start, and these constants were reduced to 0.08, 0.07, 0.06, ... at appropriate intervals in the iterative process.

The learning behaviors of the networks are indicated by the curves marked (a) in Figs. 6 and 7 for the both printers, respectively; the curve (a) represents the rate of decrease of the total system error as the number of iterative presentations (epochs) increases. The total system error is defined as the sum of the squared errors. After 40 000 epochs of the iterative learning process, the total system error reached a small value of 1.05 for the dye sublimation printer. This error corresponds to the average system error of 1.59 $\times 10^{-4}$ per one normalized data item. Moreover, it is equivalent to the average color difference of $\Delta E^*_{ab} = 1.32$. These errors become 1.55, 2.36×10^{-4}, and 1.61, respectively, for the ink jet printer. These color differences predicted by the model are repeatable for the two printers.

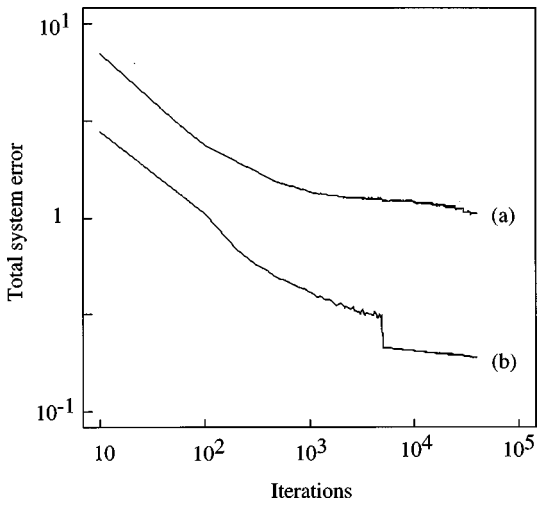

Fig. 6 Learning behaviors for the dye sublimation printer. (a) System error for modeling the printer. (b) System error for training the combined NN system.

In order to compare the learning behaviors, we applied a direct mapping scheme to the above data of 6561 color patches. A 3-10-10-3 type network used in the earlier work[6] by Tominaga was extended to a 3-10-10-4 type network. Here this network was used for trying to determine a direct mapping from the $L^*a^*b^*$ color space to the CMYK ink signal space, where the normalized values of the $L^*a^*b^*$ color specifications and the CMYK ink signals were assigned to three inputs and four outputs, respectively. In this case, the iterative learning algorithms did not converge at all. For instance, Fig. 8 shows the learning curve

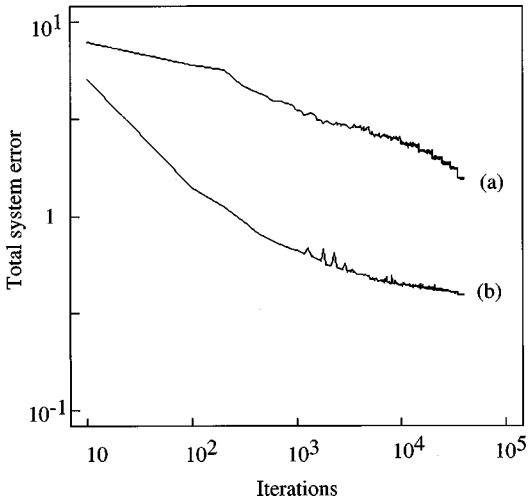

Fig. 7 Learning behaviors for the ink jet printer. (a) System error for modeling the printer. (b) System error for training the combined NN system.

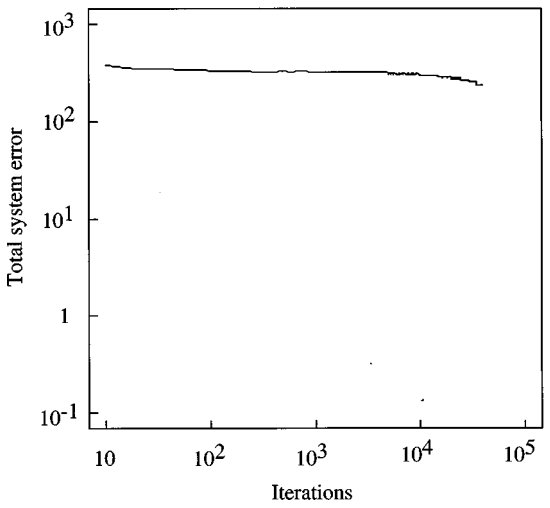

Fig. 8 Learning behavior for the dye sublimation printer when the known network method (see Ref. 6) was applied in order to directly map from the $L^*a^*b^*$ color space to the CMYK ink signal space.

for the dye sublimation printer. The total system error still remains at a high value of 235.1 even after 40 000 iterations. The direct mapping scheme failed.

4.2 *Controller*

The controller was constructed by training the total eight-layered network according to the procedure in Sec. 3.2. For realizing the identity mapping, the same $L^*a^*b^*$ values were presented to both the input and output layers, which were identical with the 6561 $L^*a^*b^*$ values of the training data used for modeling the printers. The training constants in the iterative process were set in the same manner as shown for the printer model. The curves marked with the (b) symbols in Figs. 6 and 7 indicate the learning behaviors of the combined NN system. During training, the squared error of the combined network converges at around 40 000 iterations in both cases, with the total system error, the average system error, and the equivalent color difference being 0.191, 2.91×10^{-5}, and 1.01, respectively, for the dye sublimation printer. For the ink jet printer, these errors are 0.394, 6.01×10^{-5}, and 1.41. Since the system errors became sufficiently small, the controller parts from each combined NN system could be utilized. Figure 6(b) shows the sharp drop in the learning behavior. It is supposed that the system found a steep valley of minimizing the error function in the drop point.

4.3 *Accuracy Test*

The accuracy of color reproduction achieved by the controller obtained above was examined as follows: First, we tested color samples from the printer outputs. These color samples consisted of a total of 148 color patches, which were produced from 132 coordinate points in the CMYK space by uniformly sampling the CMY scales at the fixed K values 0 and 128, and uniformly sampling the gray axis of the color space at 16 locations. The color specifications of the measured color patches are shown in Figs. 9 and 10 for the dye sublimation and ink jet printers, respectively, where

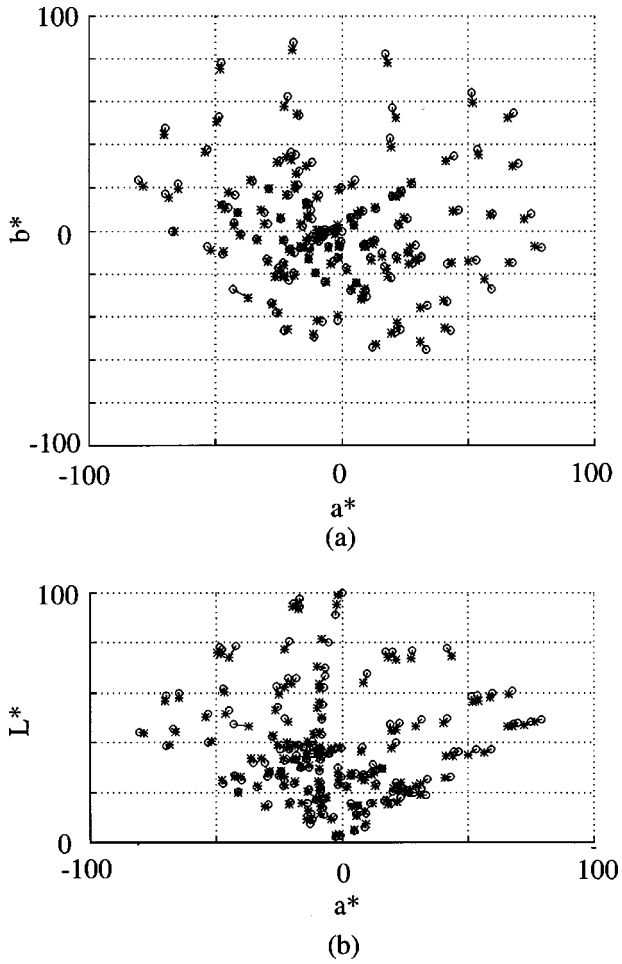

Fig. 9 Test results of color reproduction for the dye sublimation printer. ○ Target color coordinates. * Reproduced color coordinates.

the ○ symbols represent each coordinate point of the test data. In these figures, the upper diagram (a) and the lower diagram (b) represent the projections of the three-dimensional coordinates onto the (a^*,b^*) plane and (a^*,L^*) plane, respectively. The points marked with ○ symbols are the target color coordinates of color reproduction in the $L^*a^*b^*$ space. The color gamuts of printers range over about $5\leqslant L^*\leqslant100$, $-80\leqslant a^*\leqslant80$, $-60\leqslant b^*\leqslant80$ for the dye sublimation, and $20\leqslant L^*\leqslant100$, $-70\leqslant a^*\leqslant70$, $-60\leqslant b^*\leqslant85$ for the ink jet printer. The printing papers are assumed as the white point $(L^*,a^*,b^*)=(100,0,0)$.

Second, the $L^*a^*b^*$ color specifications of the target colors were fed into the controller for determining the ink amounts of the CMYK primaries. Third, color patches were reproduced from these CMYK signals with the real printers. The accuracy was examined by comparing the reproduced color specifications and the target color specifications in the $L^*a^*b^*$ space. The * symbols in Figs. 9 and 10 represent the reproduced color coordinate points. The color differences between the targets ○ and the reproduced colors * represent the overall error of color reproduction. The average color differences are 2.24 as Euclidean dis-

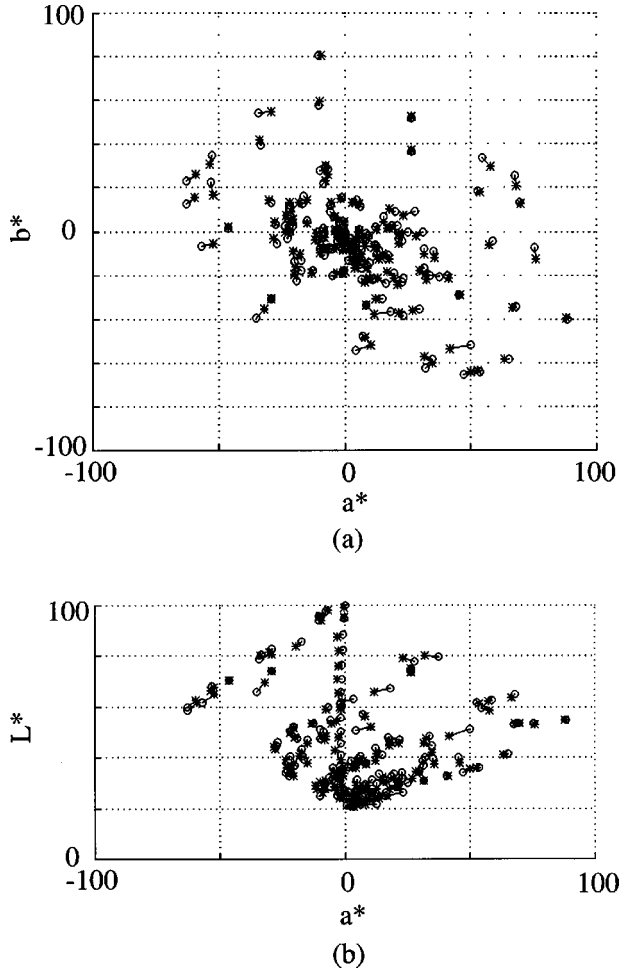

Fig. 10 Test results of color reproduction for the ink jet printer. ○ Target color coordinates. * Reproduced color coordinates.

tance ΔE^*_{ab} (rms Delta E) for the dye sublimation printer and 2.95 for the ink jet printer.

It looks as if there are noticeable trends in reproduction. For the dye sublimation printer, the reproduced chroma is generally a bit lower than the target in Fig. 9(a), particularly as the target chroma increases, and the L^* of the reproduced color is lower than the target L^* in Fig. 9(b). Those trends are a bit less noticeable for the inkjet in Fig. 10 but are still there.

5 Conclusion

In this paper we have proposed a method for solving the mapping problem from the three-dimensional color space to the four-dimensional CMYK space of printer ink signals by means of neural networks. The CIE-$L^*a^*b^*$ color system is used as the device-independent color space. We have realized the mapping from the $L^*a^*b^*$ color space to the printer CMYK color space without using the UCR and GCR techniques. This color reproduction problem was considered as the problem of controlling an unknown printing system with four inputs and three outputs. A controller finds the CMYK signals necessary to produce the desired $L^*a^*b^*$ values from a printer. Our solution for this control problem was based on a two-phase procedure. The first phase determines a neural network for modeling the printer, and the second phase determines the combined NN system combining the printer model and the controller so as to provide the identity mapping. The controller then realizes the desired mapping for color control.

Practical algorithms based on multilayer feedforward networks were presented. First, a printer is modeled with a 4-10-10-3 type neural network having the four CMYK inputs and the three $L^*a^*b^*$ outputs. Second, the controller is constructed with a 3-10-10-4 type network to realize an inversion system of the printer model. The effective learning procedure for training the networks is based on the error-backpropagation rule. The feasibility of the proposed algorithms were shown for a dye sublimation printer and an ink jet printer. Many printed color patches from these printers were measured for determining the printer models and evaluating the accuracy. The training of the entire network was stable. The experimental results suggest that the controller achieves a color reproduction accuracy of less than 3.0 in the average $L^*a^*b^*$ color difference for both printers.

In this paper, we were not running into any edge problems on a printer gamut in the estimation process. The above accuracy is performed in the region close to the gamut edge.

Acknowledgments

The author thanks David E. Rumelhart and Brian Wandell of Stanford University, Toshiaki Sakai of Kyoto University, Michael Hild of Osaka Electro-Communication University, and Jan Bares of Xerox Corporation for their useful discussions in this work.

References

1. M. C. Stone, W. B. Cowan, and J. C. Beatty, ''Color gamut mapping and the printing of digital color images,'' *ACM Trans. on Graphics* **7**, 249–292 (1988).
2. R. S. Berns, ''Color WYSIWYG: A combination of device colorimetric characterization and appearance modeling,'' *Proc. SID* **92**, 540–552 (1992).
3. H. Kotera *et al.*, ''A single chip color processor for device independent color reproduction,'' *Proc. 1st IS&T/SID Color Imaging Conf.*, pp. 133–137 (1993).
4. Po-Chieh Hung, ''Smooth colorimetric calibration technique utilizing the entire color gamut of CMYK printers,'' *J. Electron. Imaging* **3**(4), 415–424 (1994).
5. Y. Arai, Y. Nakano, and T. Iga, ''A method of transformation from CIE-$L^*a^*b^*$ to CMY value by a three-layered neural network,'' *Proc. 1st IS&T/SID Color Imaging Conf.*, pp. 41–44 (1993).
6. S. Tominaga, ''A neural network approach to color reproduction in color printers,'' *Proc. 1st IS&T/SID Color Imaging Conf.*, pp. 173–177 (1993).
7. H. Ogatsu, K. Murai, and S. Kita, ''A flexible GCR based on CIE-$L^*a^*b^*$,'' *Proc. SPIE* **2414**, 123–133 (1995).
8. J. A. C. Yule, *Principles of Color Reproduction*, Chap. 11, Wiley, New York (1967).
9. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, ''Learning internal representations by error propagation,'' in *Parallel Distributed Processing*, D. E. Rumelhart and J. E. McClelland, eds., Vol. 1, Chap. 8, MIT Press, Cambridge, MA (1986).
10. S. Tominaga, ''Color notation conversion by neural networks,'' *Color Research and Application* **18**(4), 253–259 (1993).
11. R. Ulichney, *Digital Halftoning*, Chap. 8, MIT Press, Cambridge, MA (1993).

Shoji Tominaga received his BE, MS, and PhD degrees in electrical engineering from Osaka University, Toyonaka, Osaka, Japan, in 1970, 1972, and 1975, respectively. From 1975 to 1976 he was with Electrotechnical Laboratory, Osaka. Since 1976 he has been with Osaka Electro-Communication University, Neyagawa, Osaka, where he is currently a professor with the Department of Engineering Informatics. During the 1987–1988 academic year he was a visiting scholar at the Department of Psychology, Stanford University. His research interests include computational color vision, color image analysis, and neural networks. He is a member of the Optical Society of America, IEEE, ACM, SID, and IS&T.