

Color and Printer Models for Color Halftoning

Chang-Yeong Kim

Samsung Advanced Institute of Technology, Signal Processing Laboratory

P.O. Box 111, Suwon, Korea 440-600

E-mail: cykim @ saitgw.sait.samsung.co.kr

In-So Kweon

Korea Advanced Institute of Science and Technology,

Department of Electrical and Electronic Engineering

207-43, CheongRyangRiDong, DongDaemooNGu, Seoul, Korea 130-012

Yang-Seock Seo

Samsung Advanced Institute of Technology, Signal Processing Laboratory

P.O. Box 111, Suwon, Korea 440-600

Abstract

Two error diffusion algorithms, based on Pappas's printer model accounting for dot-overlapping and ink distortion, are presented to achieve good color reproduction. The basic idea is to combine printer and color models on the perceptually uniform Commission Internationale de l'Eclairage (CIE) $L^*a^*b^*$ (CIE 1916) color space. The models, derived from the Neugebauer equations and color matching theories, are designed to achieve the minimization of the human visual color distortions between the colors of original pixels and those of a halftoned image. The effectiveness of our approaches is shown by comparison and examination of two error diffusion algorithms with previous methods: the error diffusion based models and the window based minimization algorithm. Experimental results of the error clipping technique, focused on the real application of the nonseparable algorithm, and the desired range of error clipping, where an image produced by the nonseparable algorithm can be stable without additional color distortion, are reported.

1 Introduction

Color halftoning is a method to simulate a continuous tone image with limited number of colors.^{1,2} One of the main drawbacks of halftoning techniques is the poor spatial resolution, which is caused from dither matrices. To overcome this problem, error diffusion methods are widely used^{1,3} and several research results were obtained that improved the performance of error diffusion method.⁴⁻¹⁰ Currently, color printers are widely used and there are efforts to improve the significant problem in color printers of the spreading of inks, which generates "color distortions" by dot overlap. To consider the problem and improve the image quality, a general framework

for the model-based approach in color error diffusion was proposed by Pappas.⁵ He also treated the imperfect ink case and showed examples that dot size is $\sqrt{2}$ times the ideal pixel size.⁶ Recently, the model-based approaches using Pappas's printer model^{1,5,6} have been exploited by several authors.^{11,12} As the result of these approaches, it has become clear that the model-based approach can be a powerful tool to improve the quality of the image produced by error diffusion. However, the effectiveness of color halftoning techniques considerably depends on the color space that is used. Previous methods have not adequately taken this characteristic into account in implementation.

Some important points to be considered in designing color error diffusion algorithm are the following. The first is to select standard color space like Commission Internationale de l'Eclairage (CIE) XYZ, CIE 1976 $L^*a^*b^*$, and CIE 1976 $L^*u^*v^*$ (Ref. 13) as the quantization domain in the error diffusion algorithm. The advantage of adapting a standard color space is that it is then possible to directly print image data composed of the standard color space. For example, the CIE 1976 $L^*a^*b^*$, which is referred to as LAB in the following sections, will be used as the standard color space for color facsimile in the future,¹⁴ and it must characterize color devices in terms of a standard color space in the color management system.^{15,16} The second point is the selection of the color space where the quantization process occurs, so that the error vector or distance measurement is perceptually uniform over that space.¹ We have shown the algorithm structure and an initial result that shows image quality could be improved by adopting the perceptual uniform color space LAB in the model-based approach. Based on Ref. 12 and experimental data, we report here the method, expanded and improved to focus on the application technology in practice. Especially,

we give the optimal range of “error clipping” to effectively prevent the divergence of an image in the non-separable algorithm. The effectiveness of the error clipping method is proved by applying it in the images of divergence and convergence. In these algorithms, we use the color model from Pappas’s printer model, which includes a color model and a printer model,^{5,6} to analyze color correction effects in terms of the standard color space LAB. The models are derived from a numerical calculation based on the Neugebauer equation when the fractional area is known, as in Pappas’s approach.^{5,6} The one difference between the models described here and Pappas’s printer model is that our model uses the perceptually uniform color space CIELAB and is designed to minimize the human visual color distortion. Hence, the color distortion caused by interdot overlapping and imperfect ink^{5,6} can be explained in terms of the standard color space LAB in this paper. The model-based error diffusion algorithm with the nonseparable printer model is sometimes unstable and diverges for some images.⁶ In the case of selecting the LAB color space as a quantization domain, the algorithm is also nonseparable and becomes unstable, especially it diverges when the outside colors of the printer gamut range are continuously input to the system. To correct this divergence problem, the error clipping method is used.

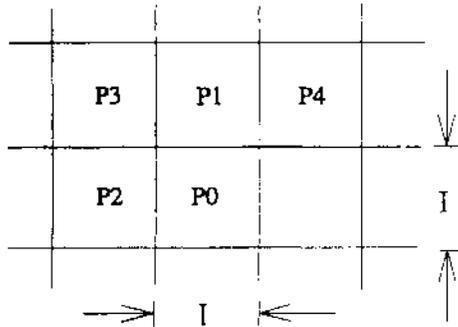


Figure 1. Ideal pixels of the printer model.

For the experiments we use a color scanner (CLC500 by Canon) and three ink jet printers (300 dpi PaintJet XL300, 180 dpi PaintJet XL180 by HP, and 360 dpi IBM 4079) each of which has different physical dot characteristics. A spectrophotometer is used to measure the color values of inks of the three printers, and a high scope system is used to measure the diameter of the physical printed dot of the color printers.

2 Printer Model

A printer model is independent of the color characteristics of a printer. Modeling printers can begin with the assumption of a raster printing model, from left to right and top to bottom, and the produced colored dots on a media (typically paper) make a Cartesian grid with a horizontal and vertical spacing of I inches. The term “ideal pixel” such as P_0 to P_4 in Figure 1 is defined as the square dot of I inches. Usually, the features of printed

dots do not have the shape of the ideal pixel. The real features of the printed dots are more closely round than square and the diameter of the dots produced by the color printer must be at least $2I$, a minimum dot size. Hence the smallest dot diameter of the printer must be designed as $\sqrt{2}I$. The reciprocal of I is the resolution of the printer in dots per inch (dpi). The HP PaintJet XL300 and PaintJet XL180 and the IBM Color Jet printer PS 4079, with 360 dpi color ink-jet printers with four color (cyan, magenta, yellow and black) nozzles are used as test vehicles. A specific color printer model, with target regions, which are shown as the solid lines in Figure 2, is developed. In this model, the area of each target region becomes the same as that of the ideal pixel. We also assume that the diameter of the dots produced by printer D satisfies⁵

$$\sqrt{2}I \leq D \leq 2I. \quad (1)$$

It is found that the dot diameter of the 300 dpi HP printer is $D = 2.2I$, and those of the IBM printer and the 180 dpi HP printer are $D = 1.7I$. Figure 3 shows eight different colored segments, C_0 to C_7 in the target region T_p (Ref. 5). Each segment’s color is determined from the colors of neighboring pixels P_1 to P_4 and the current pixel P_0 . In Figure 3, a , b , c and d are, respectively, fractional areas of segments and P_1 , P_2 , P_3 and P_4 are neighboring pixels that have an effect on the color of the current pixel P_0 . Here C_0 to C_7 represent the color of each segment and C_{P_0} to C_{P_4} represent the colors of the current pixel P_0 and neighboring pixels P_1 to P_4 , which have the shape of ideal pixel. The fractional areas of segments, a , b , c and d , are calculated by the geometrical interpretation of the target region as⁵

$$a = 1 - 2b - 4c - d, \quad (2)$$

$$b = \frac{1}{4I^2} \left[\Pi D^2 - 4I^2 + (2 + \sqrt{2})I(D^2 - I^2)^{1/2} - 6D^2 \cos^{-1}\left(\frac{I}{D}\right) \right], \quad (3)$$

$$c = \frac{1}{12I^2} \left[3\Pi D^2 - 12I^2 - 6I(D^2 - I^2)^{1/2} - 8D^2 \cos^{-1}\left(\frac{I}{D}\right) \right], \quad (4)$$

$$d = \frac{1}{I^2} \left[I^2 = \Pi D^2 - 4I(D^2 - I^2)^{1/2} + 4D^2 \cos^{-1}\left(\frac{I}{D}\right) \right]. \quad (5)$$

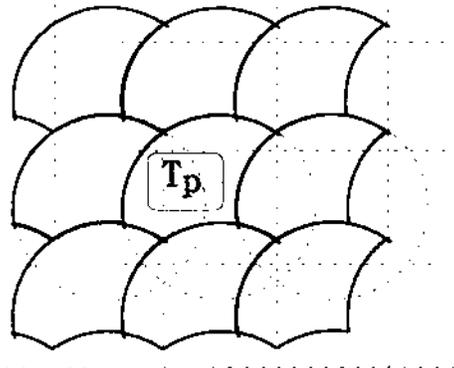


Figure 2. Target region of printer model.

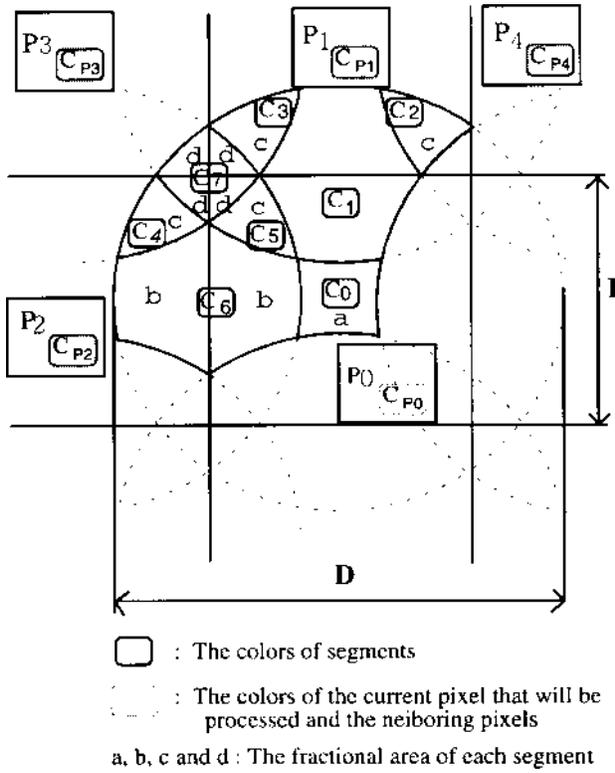


Figure 3. Colors and fractional area of the target region.

3 Color Model

A printer model using device colors R (red), G (green), B (blue), C (cyan), M (magenta), and Y (yellow) was proposed by Pappas,⁵ whose printer model includes both a color model and a printer model. The color model described here is based on it. The main differences between Pappas's model and our model described here are in the choice of the target region and the use of the perceptually uniform color space LAB as the quantization domain. If we define the target region T_p , as shown in Figure 2, under the condition of Eq. (1), the area of the target region is the same as that of the ideal pixel P and the number of neighboring pixels that affect the color of target region reduces from eight to four. Also, we do not need to consider other neighboring pixels, except the four neighboring pixels P_1 to P_4 in the quantization step. If we consider the target region composed of eight different colored segments and the current pixel P_0 and four neighboring pixels P_1 to P_4 , the number of available colors of the target region is 8^5 (Ref. 17). In this paper, the concept of device-independent color^{15,16} is also applied in the model-based error diffusion algorithm. The concept of device-independent color, which is introduced by Schreiber,¹⁸ is more suited to horizontally distributed systems enabled by networks than a stand-alone system. For example, for an ink-jet printer networked by other devices such as other printers, display devices, computers, and facsimile, the device independent color concept is important. One thing in particular distinguished in the device independent color is that standard color values are converted to the ink values of cyan, magenta, and

yellow. Hence, we try to modify the color space of Pappas's model from device color space^{5,6} to standard color space. If the fractional area of each primary color is determined, the Neugebauer equations specify the averaged color of the halftone screen.¹⁹ Therefore, in the model-based approach, once the tristimulus values and fractional area of eight segments of the target region are determined, the tristimulus values of the target region can be calculated.^{5,6} Let us review the printer model⁵ and estimate the color values of a target region in terms of LAB.

3.1 Review of Color Estimation Model

Binary color printers usually use cyan (C), magenta (M), yellow (Y), and black (B) inks to produce color dots. But if we consider that the different inks of C, M, Y are printed on top of each other to produce red (R), green (G), blue (B), and white (W), the primary colors of a color printer are defined by eight colors. Figure 3 shows eight segment colors C_0 to C_7 , and each color of effective pixels C_0 to C_4 , which affects the color values of the target region. Since the color of each effective pixel is specified as one of eight primary colors, the color of each segment can be represented by Eqs. (6) to (13).

Methods to decide the segment's color are classified into four categories by the number of effective pixels. A segment's color C_0 is determined only by one effective pixel C_{p0} . Some segments colors, such as C_1 and C_6 are determined by two effective pixels, C_{p0} and C_{p1} and C_{p0} and C_{p2} , respectively, by Eqs. (7) and (12). A third class of segment colors (C_2, C_3, C_4, C_5) is specified by three effective pixels by Eqs. (8), (9), and (10), and (11). The last class is determined by four effective pixels, as shown in Eq. (13).

$$C_0 = f(C_{p0}), \quad (6)$$

$$C_1 = f(C_{p0}, C_{p1}), \quad (7)$$

$$C_2 = f(C_{p0}, C_{p1}, C_{p4}), \quad (8)$$

$$C_3 = f(C_{p0}, C_{p1}, C_{p3}), \quad (9)$$

$$C_4 = f(C_{p0}, C_{p2}, C_{p3}), \quad (10)$$

$$C_5 = f(C_{p0}, C_{p1}, C_{p2}), \quad (11)$$

$$C_6 = f(C_{p0}, C_{p2}), \quad (12)$$

$$C_7 = f(C_{p0}, C_{p1}, C_{p2}, C_{p3}), \quad (13)$$

where $f(\bullet)$ is some function thereof. If all segment colors C_0 to C_7 are specified by Eqs. (6) to (13), the color of the target region C_{target} is determined by

$$C_{\text{target}} = f(C_0, C_1, C_2, C_3, C_4, C_5, C_6, C_7). \quad (14)$$

To evaluate segments' colors C_0 to C_7 , the remaining task is to find the relationship between the effective pixels' colors C_{p0} to C_{p4} and the segments' colors. To establish the color estimation model of each segment we assume that the segment color is independent of the print-

ing order. To approximate the segment color with various color combinations, the following color combination rules are used.

1. The order of color:
 zeroth order: white (W)
 first order: yellow (Y), magenta (M), cyan (C)
 second order: red (R), green (G), blue (B)
 third order: variations of black (K).
2. If a color is added to the same color, the resulting color is not changed.
3. A color added to the zeroth order makes the same color.
4. The first order color added to the second order except for complementary color makes the second order.
5. A combination of the second order colors, or the second order color and complementary color, makes the third order.
6. The third order color added to any color makes another third order.

As a result of the application of these rules, the color of each segment is computed in terms of the eight primary colors and one of the third order colors that have a different color.

3.2 Estimation of Color Values, LAB, of Target Region

In the following colorimetric computation, the CIE standard illuminant D_{65} and the color matching functions of the CIE 1931 standard observer have been adapted. Sixteen different colors are used for the estimation of each segment's color, but nine of these colors are variations of black. To obtain the XYZ values of the 16 different colors, a solid color chart has been designed and the printed samples produced by ink-jet printer (PaintJet XL 300) are measured with a spectrophotometer (CM2002-Minolta) in terms of XYZ, which refers to CIE XYZ (1931) in the following sections. We can observe from Table 1 that the color inks of PaintJet XL 300 are imperfect. For example, the Y component of R patch has the 50% value of the X component of R patch, and the C patch also has a nonnegligible X component, and so on. The model-based error diffusion algorithm with a nonseparable printer model is sometimes unstable and diverges for some images.⁶ In the case of adapting the standard color space as a quantization domain, the error diffusion algorithm is also nonseparable; moreover, the degree of imperfection can not be negligible. Once we know the segments' XYZ values and the fractional area of eight segments from Eqs. (2) to (5) as Pappas treated the imperfect printer model,^{5,6} the XYZ values of the target region can be specified by:

$$X_{\text{target}} = \sum_{i=0}^7 X_i f_i, \quad (15)$$

$$Y_{\text{target}} = \sum_{i=0}^7 Y_i f_i, \quad (16)$$

$$Z_{\text{target}} = \sum_{i=0}^7 Z_i f_i, \quad (17)$$

where f_i is fractional area, and X_i , Y_i , and Z_i are the CIE XYZ values (1931) of each segment from Table 1, and X_{target} , Y_{target} and Z_{target} are the CIE XYZ values (1931) of the target region, respectively. The XYZ values of the target region can be transformed into LAB color domain:

$$L_{\text{target}}^* = 116 \left(\frac{Y_{\text{target}}}{Y_n} \right)^{1/3} - 16, \quad (18)$$

$$a_{\text{target}}^* = 500 \left[\left(\frac{Y_{\text{target}}}{X_n} \right)^{1/3} - \left(\frac{Y_{\text{target}}}{Y_n} \right)^{1/3} \right], \quad (19)$$

$$b_{\text{target}}^* = 200 \left[\left(\frac{Y_{\text{target}}}{Y_n} \right)^{1/3} - \left(\frac{Z_{\text{target}}}{Z_n} \right)^{1/3} \right], \quad (20)$$

where X_n , Y_n , and Z_n are the XYZ values of the standard illuminant, CIE D_{65} and L^* , a^* , and b^* represent LAB values of the target region.

Table 1. The XYZ Values of Possible Segment Colors under CIE D, Standard Illuminant

No.	Color Patches	XValues	YValues	ZValues
1	R	22.21	11.92	3.57
2	G	6.51	15.26	5.98
3	B	5.76	4.81	17.8
4	C	14.77	21.08	58.0
5	M	23.4	12.17	11.4
6	Y	65.35	72.07	8.18
7	K	3.06	3.25	4.05
8	W	80.37	84.45	91.2
9	CMY	3.47	3.55	3.85
10	RB	2.58	2.44	2.8
11	RC	3.38	3.45	3.8
12	BY	3.37	3.45	3.82
13	RGB	2.47	2.5	2.65
14	RG	2.79	2.95	2.52
15	OB	2.37	2.63	9.93
16	GM	3.32	3.4	3.73

4 Model Based Halftoning

In this section, we consider two algorithms, the model-based error diffusion and the window-based error minimization algorithm: Two algorithms are based on the model-based approach⁵ and the concept of device-independent color.^{15,16} In Section 4.1, a model-based error diffusion on LAB color space is described and experimental data of the error clipping technique, focused on the real application, are reported. We can obtain the desired range of error clipping where an image produced by the nonseparable algorithm, can be stable without additional color distortion. In Section 4.2, a window-based minimization algorithm that uses the window primary colors and pixel reallocation technique is introduced. We use \mathbf{I}_{ij} to denote a color image, where i and j denote the location of a pixel at i th column and j th row in the Cartesian grid. Typical color image has three channels in which each channel has 256 gray levels. We

consider LAB image with 256 gray levels for each channel. The choice of gamut range is as follows: $L^* = [0,100]$, $a^* = [-125,125]$, $b^* = [-125,125]$.

4.1 Error Diffusion Based on Models (EDBM)

Error diffusion is superior in spatial resolution and continuous tone reproducibility to any other halftoning techniques.^{20,21} The color error diffusion algorithm is usually applied to each R, G, and B component independently, then it has a simple structure and the merit of a low computational cost. However, when the error diffusion algorithm is applied to a printing system, the image becomes dark and displays color distortion due to dot overlap. In addition, the inks used by most color printers are imperfect and present significant unwanted absorption. Moreover, the amount of ink absorption is variable and depends on colorants. To overcome these problems, a model-based approach to color error diffusion was proposed by Pappas.^{5,6} Another approach to make color correction and solve this problem is to generate the printer and color models using a standard color space such as XYZ or LAB. Moreover, the color space chosen for the error diffusion makes a significant difference in the color reproduction accuracy. Hence, the important design consideration for error diffusion is to select the color space, in which the quantization occurs, so that the error vector can be estimated in perceptually uniform color space. Uniform color spaces¹³ are designed with the objective of providing a color coordinate system in which the length of the color difference vector corresponds to a difference that is just noticeable and is the same throughout the color space regardless of location and angular orientation. The error diffusion algorithm on uniform color space minimizes the human visual color difference between the modified color image C_{ij} and the reproduced color image at the location (i,j) . The block diagram of the EDBM is shown in Figure 4, where I_{ij} represents the color of a pixel and is given as three values corresponding to the color components of perceptually uniform color space LAB:

$$I_{ij} = (L_{ij}^*, a_{ij}^*, b_{ij}^*), \quad 1 \leq i \leq N_w, \quad 1 \leq j \leq N_H, \quad (21)$$

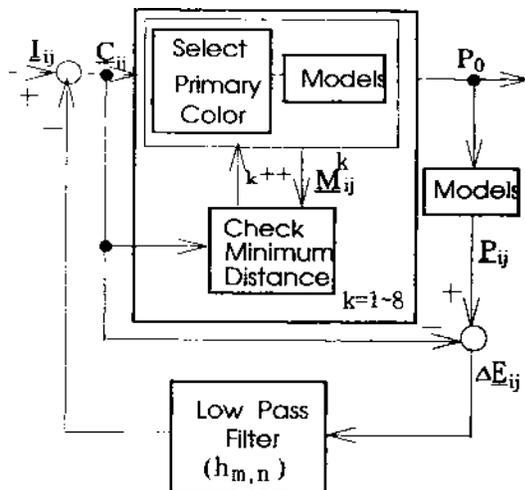


Figure 4. Block diagram of the proposed regular algorithm of color error diffusion on a uniform color space (CIE $L^*a^*b^*$ 1976).

where N_w and N_H represent the horizontal and vertical ranges of the printer, respectively. The regular algorithm of the EDBM consists of three steps:

First is the estimation of the modified input vector C_{ij} from the input color I_{ij} and an error vector that is diffused from neighboring pixels:^{1-7,22}

$$\begin{aligned} C_{ij} &= I_{ij} - \sum_{\min} (P_{i-m,j-n} - C_{i-m,j-n}) \cdot h_{m,n} \\ &= I_{ij} - \sum_{\min} \Delta E_{m,n}, \end{aligned} \quad (22)$$

where $h_{m,n}$ is a transfer function of a low-pass filter, proposed by Jarvis *et al.*³

Second is finding the closest primary color of the current pixel P_0 , to modified input color C_{ij} , by minimizing the norm of the error vector between the modified input color C_{ij} and target color M_{ij} :

$$P_{ij} = (M_{ij}^k)_{\|M_{ij}^k - C_{ij}\|_{\min}}, \quad (k = 0, 1, \dots, 7). \quad (23)$$

Third is estimation of the error vector:

$$\Delta E_{ij} = P_{ij} - C_{ij}. \quad (24)$$

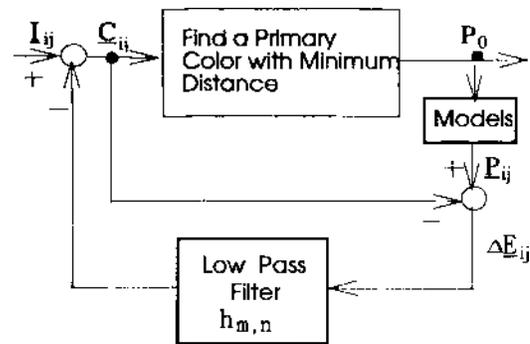


Figure 5. Block diagram of the proposed simple algorithm of color error diffusion on a uniform color space (CIE $L^*a^*b^*$ 1976).

The error ΔE_{ij} accounts for printer distortions as well as the quantization effect.^{5,6} The algorithm described here minimizes the squared color difference in the uniform LAB domain between the modified input and the color produced by the printer. But it requires an expensive computation cost to find the closest primary color of the current pixel with models. This problem is solved by another approach, as shown in Figure 5, where P_0 is assumed to be one of eight primary colors and has the closest primary color value to the modified input color C_{ij} , in uniform space (CIE $L^*a^*b^*$ 1976), which is performed without models. Here P_{ij} is estimated with the models, the information of the neighbors, and the selected primary. In the next step, the error vector ΔE_{ij} is the color difference between the modified input C_{ij} and the color of the target region P_{ij} , which is also estimated^{5,6} by Eqs. (15) to (20). While the simple version has a significant computational simplicity, the images obtained from the simple algorithm still show a quality approximately equivalent to those produced by the preceding error dif-

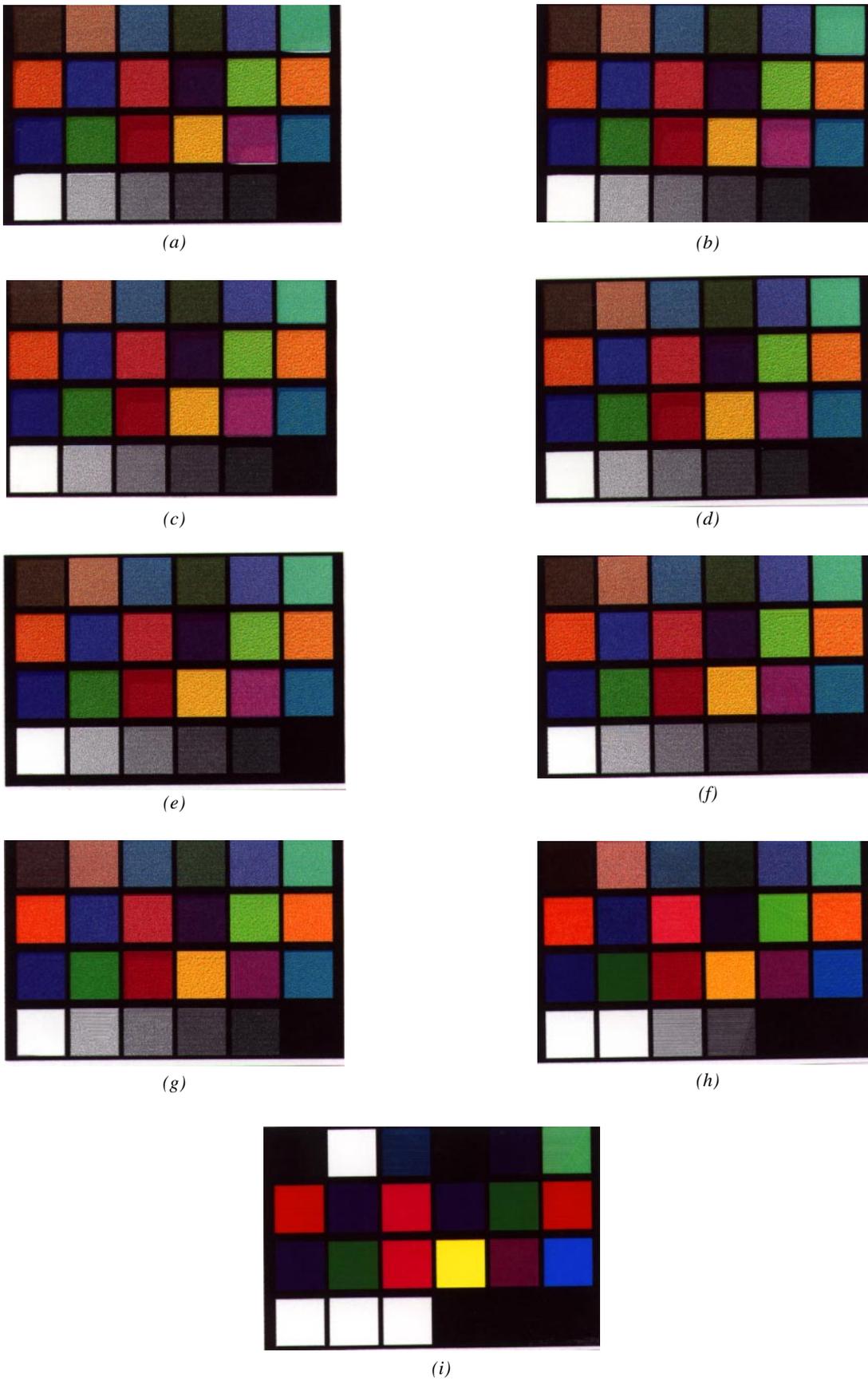


Figure 6. Printed image of the “Macbeth Color Checker” with an error clipping parameter of (a) no clipping, (b) 2.0, (c) 1.5, (d) 1.0, (e) 0.8, (f) 0.6, (g) 0.4, (h) 0.2, and (i) 0.0.

fusion algorithm, as shown in Figure 4. Therefore, the simple algorithm is used to obtain the experimental results. In a real application of a separable algorithm, the algorithm is stable because the modified input of three components (R, G, B) is clipped independently within the range of available printer input, 0 to 255. In the case of a nonseparable algorithm, the algorithm is unstable and diverges for some images,⁶ especially when the outside colors of the printer gamut are continuously input to the system and three components of the error vector have large values. Such characteristics also appear in EDBM and can be corrected by clipping the modified input within the 3-D printer gamut or the error clipping of the quantization step. In our experiment, the error clipping method and the lightness clipping of a gamut mapping are exploited to correct this divergence problem and focus on a real application. We know that the nonseparable algorithm can be made stable by the error clipping technique, but the merit of error diffusion, which enables us to represent a continuous image by a limited number of colors, goes down as the amount of error clipping increases. If we set error to zero, then using error diffusion a printer produces not a continuous tone image but an image quantized by eight colors. Hence, it is important to determine optimal values for error clipping through experiments to generate a color image without color distortion caused by the clipping. In our experiment, the gamut range of an ink-jet printer (PaintJet XL300) is selected as the reference values of error clipping: $EC_L = (L_{max} - L_{min})/2$, $EC_A = (A_{max} - A_{min})/2$, and $EC_B = (B_{max} - B_{min})/2$, where EC_L , EC_A , and EC_B represent the reference values of the L , A , and B components for error clipping, respectively. We experiment with an image of 933 × 632 pixels of the “Macbeth Color Checker,” which is scanned by a CLC 500 scanner. The parameters for error clipping are selected as: Figure 6(a) five times, Figure 6(b) two times, Figure 6(c) 1.5 times, Figure 6(d) 1.0 times, Figure 6(e) 0.8, Figure 6(1) 0.6, and Figure 6(g) 0.0 of reference values, respectively. Figures 6(a) to 6(h) shows the images produced by the PaintJet XL 300 using these parameters. The “Macbeth Color Checker” is composed of 24 different color patches. Let us number them from 1 to 24. In the case of no error clipping [Figure 6(a)], many patches are diverse; 6th and 17th patches show white stripes under each patch; and the 7th, 10th, 13th, 15th, 17th, 18th, 19th, and 23rd patches also diverge. Most patches that diverged without clipping change to stable ones except for 10th, 13th, and 15th patches, and the white stripes of the 6th and 17th patches also disappear after the error clipping parameter is set to 1.5. All patches are not changed for the parameter range from 1.5 to 0.6. We observe that the error diffused images are not as sensitive to the parameter within the range of 1.5 to 0.6. Hence, we can guess that the optimal range of the error clipping parameter is from 1.2 to 0.8, and the optimal value is 1.0. It is important to know that the 10th, 13th, and 15th patches diverge for all parameters except for critical parameter value 0.0. This reveals the limit of error clipping method applied in the nonseparable algorithm. As mentioned, the modified input must stay within the 3-D gamut range of a printer to be stable in the nonseparable

algorithm. But the 3-D gamut of the eight color printer (PaintJet XL 300) is unclear and inappropriate to treat the modified input of continuous color in the LAB domain, which becomes an obstacle to a simple algorithm for real applications. However, the gray direction of the LAB space is nearly consistent with that of the black and white points of a printer. Hence, the clipping for lightness value L can be applied in the modified input independently. The pseudo-algorithm of the lightness clipping is as following:

if $L > L_{max}$, $L = L_{max}$ and ERROR = 0;

else if $L < L_{min}$, $L = L_{min}$ and ERROR = 0;

else;

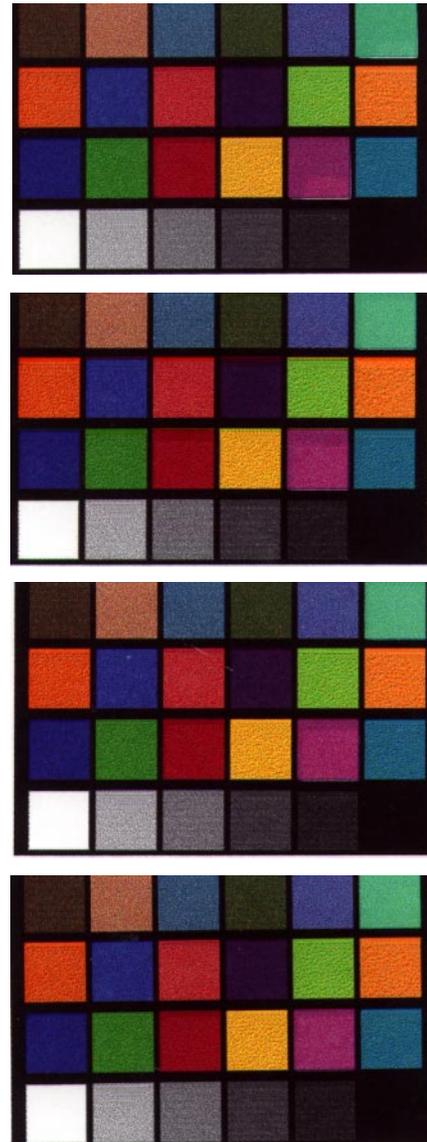
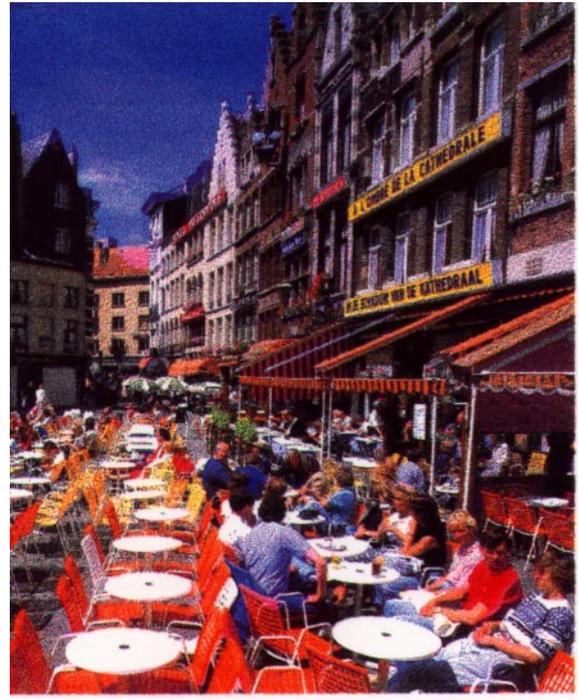


Figure 7. Printed image of the “Macbeth Color Checker” with (a) lightness clipping, (b) an error clipping parameter of 2.0 and lightness clipping, (c) an error clipping parameter of 1.5 and lightness clipping, and (d) an error clipping parameter of 1.0 and lightness clipping.



(a)



(b)



(c)

Figure 8. Printed image of a conventional image with (a) no clipping, (b) lightness clipping and (c) an error clipping parameter of 1.0 and lightness clipping.

If the L value goes off the gamut range, the error is meaningless because a printer cannot reproduce the color. Turning back the original objective of the error diffusion technique, to represent the gray scale using a binary printer within the printer gamut range, in any case, the outside color of the printer gamut can not be treated and reproduced by a printer. Figure 7(a) shows the case of lightness clipping without error clipping. We can observe that some patches also diverge. However, we can observe from Figures 7(b), 7(c), and 7(d) that the patches rapidly converge and all patches are converged in Figure 7(d) with an error clipping parameter 1.0. This is important because it shows that the instability problem in the nonseparable algorithm can be effectively solved by using error and lightness clipping simultaneously. Figures 8(a), 8(b), and 8(c) show the converged case in the nonseparable algorithm for Figure 8(a) no clipping, Figure 8(b) lightness clipping, and Figure 8(c) an error clipping parameter of 1.0 and lightness clipping, respectively. We cannot find any difference from these images. In stable circumstances, we can observe that error clipping and lightness clipping do not have an effect on an image.

4.2 Window-Based Error Minimization Algorithm (WBMA)

In a color error diffusion algorithm for a binary printer, the unit of the halftone cell is a pixel, the number of the available primary colors in quantization step is eight, and the error incurred at 1 pixel is propagated to the unprocessed neighboring pixels using a low-pass filter. This adaptive characteristic makes error diffusion superior to other dithering techniques. Nevertheless, in real applications, error diffusion can create a large number of nonprintable individual pixels, which results in a change in the tone reproduction curve (TRC). Fan²³ experimented with a halftone technique that combines traditional halftoning (ordered dithering) and the error diffusion algorithm and obtained an image that has no coarse quantization effect and a well formed dot structure like error diffusion. Moreover, since only eight colors are used in quantization, the error defined by Eq. (24) often becomes a large value, which distorts the colors of the neighboring pixels and makes it difficult to reproduce an accurate color at a pixel's original location. To solve these problems, we use additional primary colors by a window composed of more than 2 pixels in error diffusion. For example, if the window is composed of 2×1 pixels, the available window primary colors for quantization become 36 and in the case of 2×2 , 330 (Ref. 17). The WBMA has the effect of reducing the amount of error at each window location. Moreover, the error can be minimized and a pixel array has a well formed structure, like a pixel based error diffusion, by applying the pixel reallocation technique. In WBMA, by accommodating window-to-window error propagation with the pixel reallocation technique, we can obtain an image that is faithful to the original color at each window location and has no artifacts. Figure 9 shows the block diagram of the WBMA. Let us consider a window with 2×2 pixels and 330 primary colors. Then, the WBMA with the pixel reallocation technique can be described in the following five steps:

1. An input image \mathbf{I}_{ij} is transformed into the unit of a window, I_{kl} by Eq. (25). In the case of 2×2 pixels, \mathbf{I}_{kl} represents input color of the unit of window. The subscripts k and l denote the window located at (k, l) . Hence, the pixel position of the window will be $(2i - 1)$ 'th column and $(2j - 1)$ 'th row.

$$\mathbf{I}_{kl} = \sum_{i=2k-1}^{2k} \sum_{j=2l-1}^{2l} \mathbf{I}_{i,j} / (2 \times 2), \quad (25)$$

where, $1 \leq i \leq N_w$, $1 \leq j \leq N_H$, $1 \leq k \leq N_w/2$, $1 \leq l \leq N_H/2$.

2. The estimation of the modified input vector \mathbf{C}_{kl} is the same as that of step 1 in EDBM except for the subscripts.
3. To find the closest primary color with the modified input color and the indexes of pixels of the window, first, we pick one of window primary colors from the look-up table (LUT) and estimate the norm of the error vector between the modified input color \mathbf{C}_{kl} and the target window color \mathbf{M}_{kl} . The target window's color \mathbf{M}_{kl} can be calculated from Eqs. (26) to (31).

$$X_{\text{target}}^p = \sum_{i=0}^7 X_i f_i, \quad (26)$$

$$Y_{\text{target}}^p = \sum_{i=0}^7 Y_i f_i, \quad (27)$$

$$Z_{\text{target}}^p = \sum_{i=0}^7 Z_i f_i, \quad (28)$$

$$X_{\text{target}} = \sum_{p=0}^N X_{\text{target}}^p / N, \quad (29)$$

$$Y_{\text{target}} = \sum_{p=0}^N Y_{\text{target}}^p / N, \quad (30)$$

$$Z_{\text{target}} = \sum_{p=0}^N Z_{\text{target}}^p / N, \quad (31)$$

where N is the number of pixels that consist of a window: X_{target}^p , Y_{target}^p , and Z_{target}^p denote the XYZ values of each pixel; and X_{target} , Y_{target} , and Z_{target} represent the XYZ values of target window. The target window's XYZ values are converted into LAB values by Eqs. (18) and (20). We pick one of window primary colors from the LUT and estimate the norm of error vector $\mathbf{M}_{kl}^t - \mathbf{C}_{kl}$ between the modified input color \mathbf{C}_{kl} and the target window color \mathbf{M}_{kl} . If we iterate this step until the last primary color, then we can find P_{wtmp} , which has a minimum error among the window primary colors by

$$\mathbf{P}_{\text{wtmp}} = (\mathbf{M}_{kl})_{\|\mathbf{M}_{kl}^t - \mathbf{C}_{kl}\|_{\min}}, \quad (32)$$

where t indicates the number of primaries. But this algorithm has the disadvantage of reducing a reso-

lution, while the error defined by Eq. (24) is reduced, which supports the aim to reproduce accurate color at the pixel's original location. The pixel reallocation technique is applied in this algorithm to minimize an error again and prevent the resolution from decreasing.

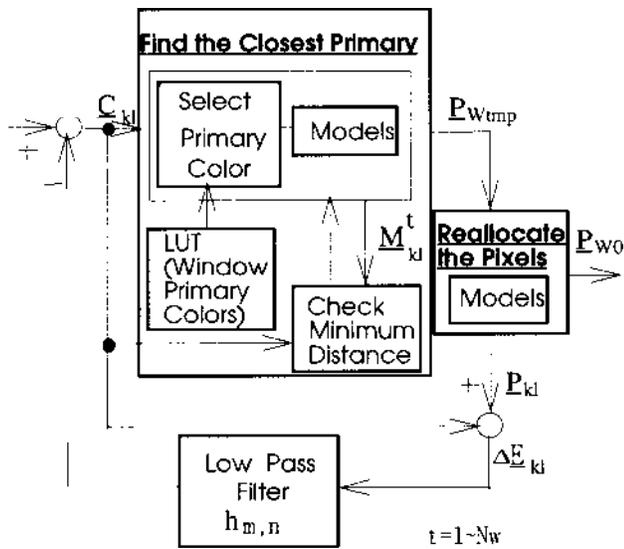
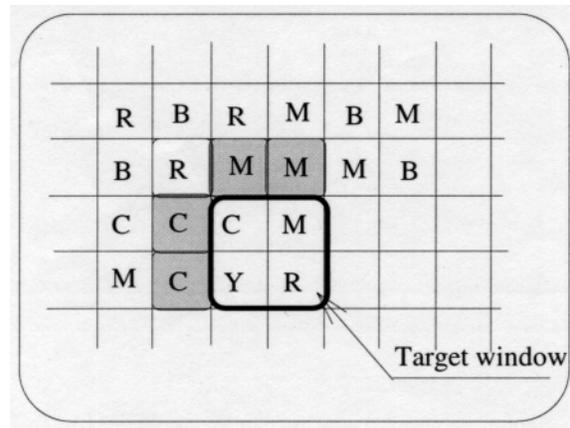


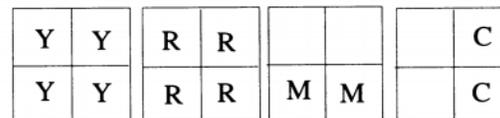
Figure 9. Block diagram of the proposed WBMA for the color error diffusion algorithm on uniform color space (CIE $L^*a^*b^*$ 1976).

4. Reallocate the pixels of the windows to find an array structure with the minimum error and that satisfies a constraint that prevents the resolution from decreasing. In step 3, the selected window consists of N pixels. The number of available colors becomes $N!$ by changing the position of each pixel in a window. Moreover, to reduce coarse quantization effect in an image and to have a well formed dot structure like the one dot based error diffusion algorithm, the constraint that the pixels of the same colors should not be neighbors in a window is used. The coarse quantization effect and decreases of resolution are reduced dramatically in an image by applying this position constraint. The available colors decrease to a value smaller than $N!$. Each window's color values are estimated by Eqs. (26) to (31) and we can select the array structure and color values from Eq. (32) and the constraint. In this step, we can minimize the error between the modified input of the window and the color of the reallocated window in the perceptually uniform color space. For example, assume that the neighboring pixels of a target window are already assigned by $M, M, C,$ and C , as in Figure 10(a) and the indices of the window primary color selected from the LUT are composed of $Y, C, M,$ and R . Since the colors of neighboring pixels in the upper row are assumed to be M and M , any pixel in the first row in the target window should not have color M . In a similar way, the first column of the target window cannot have color C . Hence, when we consider the colors of neighboring pixels in Fig-

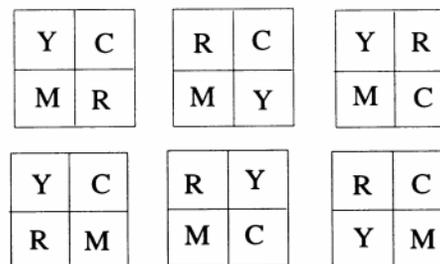
ure 10(a), the possible pixel positions of each $C, M, Y,$ and R pixel in the target window will be as shown in Figure 10(b). The number of available colors of the target window reduce from $4!$ to 6, as shown in Figure 10(c). Next, we can select an array structure, which has minimum error among the six structures in Figure 10(c). This approach causes WBMA to have an adaptive characteristic by considering the colors of neighboring pixels. We can determine the optimized pixel array that satisfies both having minimum error and preserving resolution. As a result of this step, a loss in resolution due to a pixel grouping (2×2) is corrected.



(a)



(b)



(c)

Figure 10 (a) Example of the target window and neighboring pixels in WBMA, (b) possible pixel positions of $C, M, Y,$ and R in the target window of (a), and (c) possible pixel array structures of $C, M, Y,$ and R in target window considering the colors of neighboring pixels in (a).

$$P_{ij} = (P_{ij}^m)_{\|P_{ij}^m - P_{ij}\|_{\min}}, \quad (m = 1, 2, \dots, N). \quad (33)$$

5. Estimation of the error vector is the same as that of step 3 in EDBM. The error vector is diffused to neighboring windows. The WBMA has an advantage that the quantization error is reduced and the error $\Delta \bar{E}_{ij}$ can be minimized again by the pixel real

location technique. The algorithm minimizes the squared color difference in LAB between the modified input and the printer output at the window of the current location.

5 Experimental Results and Discussions

5.1 Color Chart Experiment

The simple version of EDBM is tested on several images and the color precision are estimated in terms of LAB. For the experiments, we used a color scanner (CLC500 by Canon), and three kinds of ink-jet printers (a 180 dpi Paint-Jet XL180 and a 300 dpi PaintJet XL300 by HP and a 360 dpi IBM 4079) which have different physical dot characteristics. The Hi-Scope system (KH2200 by HiRox) is used to measure the actual diameters of the physical dots. We a spectrophotometer (CM2002 by Minolta) to measure the color values of the printing output. First, a color chart for estimating the feasibility of the models is designed. The color chart has 330 different color chips. Since each color chip has a number of halftone cells with 2×2 pixel arrays, the total number of colors of the color printer with a four color printing system becomes 330 (Ref. 21). The LAB values of the printed color chart, measured by the spectrophotometer, are compared with those estimated by Eqs. (18) to (20).

Figure 11(a) shows the relationship between the pixel diameters of the model and the color differences between the estimated and measured values of the 2×2 pixel array composed of a single color component for PaintJet XL180: cyan represents the average color difference of 2×2 pixel array composed of only cyan pixels. We can observe from Figure 11(a) that the minimum color difference ΔE_{ab} of the cyan array is 5.03 at $d = 0.2395$ mm, the magenta array is 4.90 at $d = 0.2435$ mm, and the yellow array is 5.53 at $d = 0.235$ mm. This also shows that the estimated diameters with a minimum color difference for three color components have close agreement with the value $d = 0.245$ mm measured by Hi-Scope system. Figure 11(b) illustrates the pixel diameter versus ΔE (LAB unit) plots in the PaintJet XL180 for 330 color chips. We classified the 330 color chips into three categories: the “CMY” class composed of only cyan, magenta, yellow, and white pixels; the “RGCMY” class composed of cyan, magenta, yellow, red, green, blue, and white pixels; and the “Others” class composed of one black pixel and three pixels of R, G, B, C, M, Y, R, G, or W. In the case of PaintJet XL 180, the CMY class has minimum error, and the next is the Others class, the last is the RGBCMY class, and the diameters with minimum error for each class are consistent with the measured values. Hence we are able to see that the color

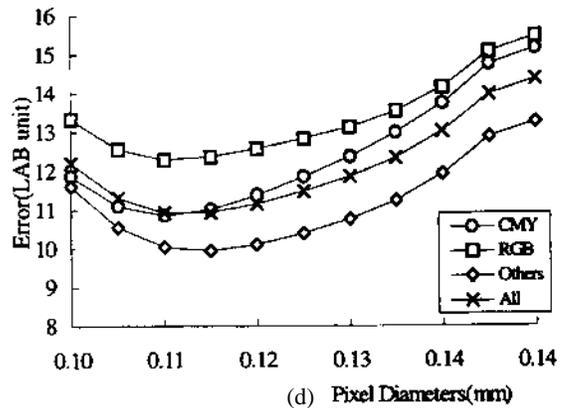
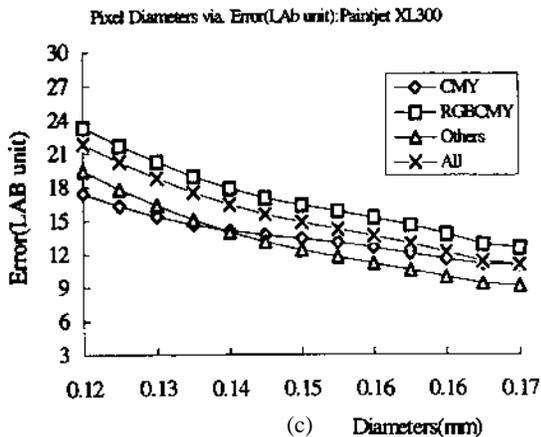
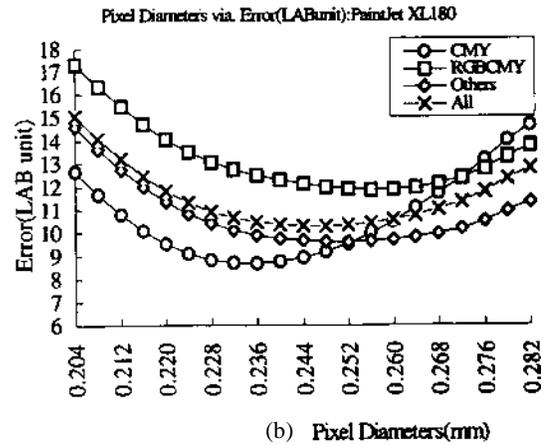
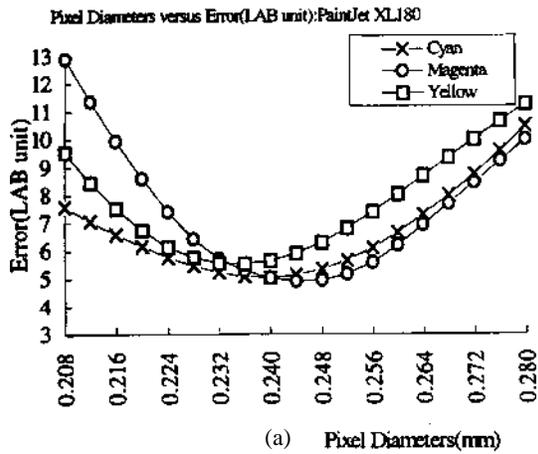


Figure 11. Pixel diameters of the model versus the color differences (a) of cyan, magenta, and yellow for the HP PaintJet XL 180, (b) for 330 colors reproduced by the PaintJet XL 180, (c) for the HP PaintJet XL 300, and (d) for the IBM 4079.

distortion is due to the misregistration of color pixels and the approximated color model. As pointed out in the printer model section, the measured diameters of the 300 dpi HP printer and IBM printer are $D_{\text{ave}} = 2.2I(0.19 \text{ mm})$ and $D_{\text{ave}} = 1.7I(0.12 \text{ mm})$, respectively. Unfortunately, the measured diameter ($D = 2.2I$) of the 300 dpi HP printer is out of the range of the our model [Eq. (1)]. Hence, three classes of Figure 11(c) show monotonic decreasing characteristics until $D = 2.0I$. In Figures 11(c) and 11(d), however, it can still be observed that the calculated diameters of the two printers are $2.0I$ (0.17 mm) and $1.7I$ (0.13 mm), which show a close agreement. We argue that the average color difference ($\Delta E = 10.5$) for the designed 330 color chips is due to such characteristics of the printer as misregistration and the variation of produced dot size and the approximated color model.

5.2 Real Image Experiment

To demonstrate the effectiveness of our methods, we tested several algorithms: the conventional error diffusion algorithm on LAB domain, EDBM, window-based error diffusion without the printer model, WBMA, and WBMA with the pixel reallocation technique, for a number of images. A PaintJet XL300 (HP, 300 dpi) is used as an image test vehicle. Figure 12(a) (see Color Plate) shows the photographic image of the color test chart (The Institute of Image Electronics Engineer of Japan, 11). Figure 12(b) shows the image of the conventional error diffusion without models and Figure 12(c) shows the image with the simple algorithm with an error clipping value of 1.0 and lightness clipping. Figure 12(d) is an example of the image quantized by 300 colors on the LAB domain, Figure 12(e) shows the image of the window-based error diffusion without the printer model, Figure 12(f) shows that of the WBMA, and Figure 12(g) is that of the WBMA with pixel reallocation. Figure 12(b) shows color distortion clearly due to the dot overlapping and Figures 12(c), 12(f), and 12(g) show the effectiveness of our approaches in color error diffusion. We can observe that the image is corrected significantly by the model based techniques. Visual comparisons of Figures 12(b), 12(c), 12(f), and 12(g) indicate that Figures 12(c), 12(f), and 12(g) exhibit a bright and acceptable color, while Figure 12(b) shows a dark and distorted color image. It is easy to observe that Figures 12(c), 12(f), and 12(g) show more faithful reproduction for hue and saturation than the image of Figure 12(b). The yellowish cloth in Figures 12(f) and 12(g) has more accurate color than the color produced by EDBM. By comparing Figure 12(c) with Figure 12(f), it is also perceived that the colors of apple, grape, cloth, rose, etc. shown in Figure 12(f) display more similarity to the original colors, in Figure 12(a), than those of Figure 12(c). Figures 12(b) and 12(e) represent the same case without the printer model. One difference between them is that Figure 12(b) is a one dot based error diffusion, while Figure 12(e) is the case of a 2×2 window. We can see from Figures 12(b) and 12(e) that the error of the window-based algorithm is smaller than that of the one dot case, as explained earlier. Hence, it also proves the effectiveness of using additional primary colors in a window. Figure 12(d) is

an image quantized by 330 colors on the LAB domain and shows very poor color reproduction with contouring and rough image quality. It is dramatically corrected by applying error diffusion based on a 2×2 window in Figure 12(e). In the case of large window size, window-based error diffusion without the printer model reproduces more accurate color than small window size. In the 2×2 window, color distortion due to dot overlapping is still observed and some color distortions in Figure 12(e) are corrected in Figure 12(f). Figure 12(g) is the case of WBMA with pixel reallocation technique, which has the objective of reducing the coarseness in window-based error diffusion by finding the optimal structure of pixels in a window. The pixel reallocation technique with the “position constraint” prevents the coarse quantization effect due to decreased spatial resolution and a changed initial structure of the LUT into fine structure in a window. As shown in Figure 12(f), there are some defects such as decreased resolution and the coarse quantization effect from the image produced by the window-based minimization algorithm.

However, we can see from Figures 12(f) and 12(g) that the coarse quantization effect due to the decreased image resolution is considerably corrected by applying WBMA with the pixel reallocation technique. While the pictures produced by WBMA in Figures 12(f) and 12(g) have the merit of reproducing accurate color at the original location, these unfortunately did not show better image quality than that of EDBM in our experiment. We believe the image quality of WBMA is related to the characteristics of the ink-jet printer, such as resolution, the number of physical colors produced by a window, the size of a printed dot, and the colorant. In the case of WBMA, the resolution is decreased to half of that in EDBM in our experiments; however, the available number of primary window colors did not actually increase to 330 colors, in which there are so many patches we can not discriminate the colors. The image quality in WBMA is the trade-off of the number of available colors and image resolution. WBMA will be more useful and give more faithful image quality than that of EDBM or the current status of WBMA under the condition that the actual number of primary window colors is nearly same as that of the predesigned ones, in our case 330 colors. The minimum window resolution for acceptable image quality is determined by human visual characteristic and the number of physical primary window colors, which are beyond the scope of this paper. In the case of a printer with low resolution, it is not so free to design the primary window colors, which strongly depend on the printer characteristics, as in our experiment, while we can design primary window colors by examining various dither matrices in a printer with high resolution. However, the number of physical primary window color can be increased by the pixel reallocation technique, which makes various primary window primary colors using 4 pixels, moreover, it considers the colors of neighboring pixels. We can interpret the EDBM as a special case of WBMA with a one dot window, in other word, WBMA is the expanded version of EDBM in the spatial domain.



(a)



(b)



(c)



(d)



(e)



(f)



Figure 12. (a) Photography of the original image, (b) printed image with the correction of the conventional error diffusion algorithm on the LAB domain (c) printed-image with the correction of the proposed simple algorithm with an error clipping of 1.0 and lightness clipping, (d) example of image quantized by 300 colors on the LAB domain, (e) printed image with the correction of the window-based error diffusion without the printer model, (f) printed image with the correction of the proposed WBMA, and (g) printed image with the correction of the proposed with WBMA with pixel reallocation.

6 Conclusions

We have examined an approach to reproduce a high quality image by using the printer model based on the measurements of a small number of patches and color model on the uniform color space (CIE $L^*a^*b^*$ 1976). The algorithms are designed to minimize the color distortion between the continuous tone image and its low-pass filtered halftone image at each pixel location (or window location), and the error has been analyzed in uniform human visual domain. In EDBM, we have performed an experiment focused on real applications and have shown that the image produced by the nonseparable algorithm also could be made stable by applying the error clipping and lightness clipping methods. Especially, the optimal range of "error clipping" to effectively prevent the divergence of an image in the nonseparable algorithm is given as from 0.8 to 1.2 times the limiting values of the ink-jet printer gamut. The effectiveness of the error clipping method has been proved by applying it in diverged and converged images respectively. By the algorithms, the color distortions caused by dot overlap and imperfect ink could be significantly corrected. To prevent decreased spatial resolution in WBMA, the pixel reallocation technique has been examined. Since the two algorithms described in this paper are performed in standard color space, it is easy to apply them to color printing devices directly, which require device independent color processing techniques, without extra color correction of a printer. The next step of our research is to investigate hardware implement action for high speed applications.

References

1. R. Ulichney, *Digital Halftoning*, The MIT Press, (1987).
 2. B. E. Payer, "An optimum method for two-level rendition of continuous tone pictures," in *Proc. IEEE Int. Conf. on Comm.*, pp. 11–26 (1973).
 3. J. F. Jarvis, C. N. Judice, and W. H. Ninke, "A survey of techniques for the display of continuous tone pictures on bilevel display." *Comput. Graph. Image Process.* **5**, 13–40 (1976).
 4. D. A. Carrara, M. Analoui, and J. P. Allebach, "Recent progress a digital halftoning," in *Proc. IS&T's 8th Int. Congr. on Advances in Non-Impact Printing Technologies*, pp. 25–30, Williamsburg VA (1992).
 5. T. N. Pappas, "Model based halftoning of color images," in *Proc. IS&T's 8th Int. Congr. on Advances in Non-impact Printing Technologies*, pp. 270–275, Williamsburg VA (1992).
 6. T. N. Pappas, "Printer models and color halftoning," in *Proc. IEEE ICASSP-93*, Vol. **5**, pp. 333–336, Minneapolis, MN (1993).
 7. T. N. Pappas and D. L. Neuhoff, "Model-based halftoning," in *Human Vision, Visual Proc. and Digital Display II, Proc. SPIE* **1453**, 244–255 (1991).
 8. T. N. Pappas and D. L. Neuhoff, "Least-squares model-based halftoning," *Human Vision, Visual Proc., and Digital Display III, Proc. SPIE* **1666**, 165–176 (1992).
 9. D. L. Neuhoff, T. N. Pappas, and N. Seshadri, "One-dimensional least-squares model-based halftoning," in *Proc. IEEE ICASSP-92*, San Francisco, CA (1992).
 10. C. J. Rosenberg, "Measurement based verification of an electrophotographic printer dot model for halftone algorithm tone correction," in *Proc. IS&T's 8th Int., Cong. on Adv. Non-impact Printing Technologies*, pp. 286–291 Williamsburg, VA (1992).
 11. Y. S. Rim and S. K. Rim, "Adaptive color error diffusion for natural color printing in binary color printer," in *Proc. IS&T's 9th Int. Cong. on Adv. Non-impact printing Technologies, Japan Hardcopy '93*, pp. 583–586, Yokohama, Japan (1993).
 12. C. Y. Rim, S. G. Kim, Y. S. See, and I. S. Kweon, "Model based color halftoning techniques on perceptually uniform color space," in *Proc. IS&T's 47th Annual Conference/CPS* 1994, pp. 494–499, Rochester (1994).
 13. G. Wyszecki and W. S. Stiles, *Color Science*, 2nd ed., Wiley, New York (1982).
 14. Commission Internationale de l'Eclairage (CIE), "Colorimetry," Publication CIE No. 15.2, Vienna (1986).
 15. R. Buckley, "A short history of device-independent color," in *Proc. 1st IS&T/SID Color Imaging Conference*, pp. 88–92 (1993).
 16. G. Murch, "Color management system," in *Proc. 1st IS&T/SID Color Imaging Conference*, pp. 95–99 (1993).
 17. P. G. Engeldrum, "Four color reproduction theory for dot formed imaging systems," *J. Imaging Technol.* **12**(126), 126–130 (1986).
 18. W. F. Schreiber, "A color prepress system using appearance variables," *J. Imaging Technol.* **12**, 200–211 (1986).
 19. G. G. Field "Color scanning and imaging System," *GATF*.
 20. R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial grey scale," *Proc. SID* **17**(2), 75–77 (1976).
 21. O. Bryngdahl, "Halftone images: spatial resolution and tone reproduction," *J. Opt. Soc. Am.* **68**, 416–422 (1978).
 22. J. C. Stoffel and J. F. Moreland, "A survey of electronic techniques for pictorial image representation," *IEEE Trans. Comm* **29**, 1898–1924 (1981).
 23. Z. Fan, "Halftoning by combining ordered dithering and error diffusion," in *Proc. IS&T's 8th Int. Cong. on Adv. Non-impact printing Technologies, Japan Hardcopy '92*, pp. 277–279, Williamsburg, VA (1992).
 24. A. H. Mutz and D. T. Lee, "An international standard for color facsimile," in *Proc. 2nd IS&T/SID Color Imaging Conference*, pp. 52–54 (1994).
- * Previously published in the *Journal of Electronic Imaging*, **6**(2) pp. 166–180, 1997.