

Color Image Quantization and Dithering Method Based on Human Visual System Characteristics*

Kyeong Man Kim, Chae Soo Lee, Eung Joo Lee, and Yeong Ho Ha

Department of Electronic Engineering, Kyungpook National University, Taegu 702-701, Korea

New methods for both color palette design and dithering based on human visual system (HVS) characteristics are proposed. Color quantization for palette design uses the relative visual sensitivity and spatial masking effect of HVS. The dithering operation for printing uses nonlinear quantization, which considers the overlapping phenomena among neighbor printing dots, and then a modified dot-diffusion algorithm is followed to compensate the degradation produced in the quantization process. The proposed techniques can produce high-quality images in low-bit color devices.

Journal of Imaging Science and Technology 40: 502–509 (1996)

Introduction

Recently, the use of color image data has been growing fast in the area of image processing.¹⁻⁷ To express natural color in conventional low-cost color devices, the color image should be quantized for monitoring and dithered for printing.

A video monitor displays a color image by modulating the intensity of three primary colors (red, green, and blue) at each pixel of the color image. In a digitized color image, each primary color is usually quantized with 8 bits of resolution to eliminate distinguishable quantization steps in trichromatic specification (luminance, hue, and saturation). Thus, full-color digital display systems use 24 bits to specify the color of each pixel on the screen. However, the cost of high-speed memory needed to support such a display on a high-resolution monitor makes many applications impractical. An alternative approach in currently available displays is to provide a limited number of bits, such as 8 bits, for specifying the color at each pixel. Each of these 2^8 values is then used as an index of a user-defined color table, i.e., color palette. Each entry in the table contains a 24-bit value that specifies each primary component of the color image. In this way, the user is allowed to select a small subset of the color palette from the full range of 2^{24} colors. The drawback of this scheme is that it restricts the number of colors that may be simultaneously displayed.¹⁻⁷

Because natural images typically contain a large number of distinguishable colors, displaying such images with a limited palette is difficult. Several techniques exist for color quantization, some of which are based on a more general class of vector quantization (VQ) techniques. One approach involves the iterative refinement of an initially

selected palette. Variations on this idea include the manner in which the initial palette is chosen and the color space in which the quantization is performed. The refinement algorithm, commonly known as the K-means or Linde–Buzo–Gray (LBG) algorithm,¹ is a vector extension of the Lloyd quantizer for scalars. The algorithm seeks to reduce the total squared error (TSE) between the original and the quantized image at each iteration until a (local) minimum is found. This method yields high-quality images and, with a properly chosen initial palette, will result in the lowest TSE for a given palette size. The method is, however, computationally intensive, and its performance is sensitive to the choice of the initial palette. Also there are splitting algorithms^{2,3,6} that divide the color space into disjoint regions and pick a representative color from each region as a palette color. The algorithms vary according to the methods used to split the color space. As an example of such an algorithm, the median-cut algorithm² invented by Heckbert was undertaken as an alternative to the popularity algorithm. The median-cut algorithm repeatedly subdivides the color space into smaller rectangular boxes until the desired number of boxes is generated. The split point is the median point, i.e., the plane that divides the box into two halves so that an equal number of colors is on each side. The main advantage of the split algorithm is lower computational time cost and memory space for the spatial storage scheme, mostly because it is simple to compute the split point. However, a number of problems are associated with this method. One problem is that partitioning a box by a plane passing through the median point does not necessarily lead to a lower quantization error.

As explained above, the conventional color quantization algorithms usually use the TSE as the distortion measure.¹⁻⁷ This measure is, however, perceptually insufficient when accurately estimating the perceptual difference between an original image and its quantized representation.⁵⁻⁷ The TSE does not take into account the spatial correlation linked with perceptually adjacent pixels. With such a measure we have no means of knowing whether the observed degradation is the result of several particularly noticeable degradations. The measure must be reconsidered, this time by duly integrating the notion of locally observable errors. Thus, we propose to use a new distortion measure that takes into account the spatial activity in local regions of the input image. The activity is computed as the mean of the sensitivity-weighted difference between the input and the local mean color in a 4×4 region. Then, using the distortion measure, a hierarchical quantization algorithm is proposed by considering the spatial masking effect,^{8,9} a characteristic of the HVS. The algorithm consists of initial and subdivision steps both to reduce the computational time and to minimize the measured distortion.

Digital dithering is the process for generating a pattern of dots with a limited number of gray levels to reproduce a

Original manuscript received January 24, 1996. Revised September 11, 1996.

*Presented in part at the 3rd IS&T/SID Color Imaging Conference, November 7–10, 1995, Scottsdale, Arizona.

© 1996, IS&T—The Society for Imaging Science and Technology

continuous tone image on the paper media for hard copy. Display of continuous tone images on the paper media is impossible without a dithering process. Thus, many dithering techniques can be found in the printing algorithms.¹⁰⁻¹³ Printers are able to produce marked dots on a part of a Cartesian grid with horizontal and vertical spacing. Conventional digital dithering methods use linear quantization in which input gray levels are equally divided by the printer resolution, dots per inch (dpi). Errors produced by the quantization are diffused to the neighborhood to compensate for the local gray level.¹⁴⁻¹⁷ In this process, accurate gray levels are not represented because of poor printer resolution, in general, and also because diffusion of the contrast in the edge region is not considered. Therefore, a new method using linear quantization and diffusion is required to reduce the degradation of images produced by the conventional dithering method.

For printing the monitor-displayed image on the paper media, we propose a nonlinear quantization that considers the overlapping of printing dots and a modified dot-diffusion method. For the nonlinear quantization, each quantization step is adjusted in proportion to the overlapping area that is subtracted from the total area of each neighboring printing dot. To compensate for errors produced in the quantization, the modified dot diffusion adjusts the quantization error to be diffused according to the characteristics of the input image. Filtering of the quantization error by a low-pass filter is carried out before dot diffusion so that the quantization error has to be diffused only in the smooth region. Also, to apply the proposed method to color printing, intensity and color change must be considered simultaneously. Thus quantization error is diffused in the Hue, Saturation, and Intensity (HSI) color coordinate that represents the HVS color coordinate system. The low-pass filtering must be considered to prevent color degradation due to the diffusion of quantization error in the color edge region.

This report is organized as follows: In the next section, which starts by defining the color space in which the quantization algorithm operates, the proposed distortion measure and hierarchical quantization algorithm are described. The succeeding section describes the printer model, the proposed nonlinear quantization, and the modified dot-diffusion method for color printing. The section that follows reports the simulation results of the proposed techniques. Finally we present our conclusion.

Proposed Color Quantization Algorithm

The color quantization for palette design is described as selecting the prescribed number of colors to display an image with almost no noticeable perceived difference. The process is usually performed by treating three color components (red, green, and blue) independently in the RGB color space. Although three color components can be decorrelated by transforming the color space to YIQ, Lab, or some other color spaces, independent quantization in these spaces is inefficient because a certain proportion of these spaces lies outside the RGB color space. In any event, color transformations are of little use in quantization for display; their proper place is in the image compression systems. Thus we quantize the colors of the original image into K (usually $K = 256$ or less) colors, called the color palette, in the RGB color space

The color image is assumed to be on the rectangular grid of $M \times M$ pixels. The set of all grid points is denoted by F and its members $f \in F$ may be explicitly written as $f = (i, j)$, where i and j are the row and column indices ($0 \leq i, j \leq M - 1$). The color value of the pixel at grid point f is denoted by $\bar{c}_f = [r_f, g_f, b_f]^T$, where the components are the red, green, and blue tristimulus values for the pixel in the

RGB color space, and superscript T means transpose. As explained above, the color quantization is selecting K colors to be displayed on the monitor. The K colors are usually composed of the centroids of K color clusters. In this report, the k -th cluster is denoted by Ω_k ($1 \leq k \leq K$), and the centroids of the clusters are denoted by $\bar{\mu}_k = [r_k, g_k, b_k]^T$, ($1 \leq k \leq K$), each of which composes the color palette. Thus the input image colors are mapped to the centroid colors after the quantization, and the mapped colors are displayed on the monitor simultaneously. Therefore the color quantization algorithm finds the optimal color cluster and its centroid with almost no difference between the original image and the reconstructed image.

Weighted-Distortion Measure of HVS Color Activity. In this report a new distortion measure based on the spatial masking effect of the HVS is proposed. The spatial masking effect means that human vision is more sensitive to quantization errors in the smooth region than to those in the edge region. To use this effect, the activity of color in the local region must be computed to judge whether the color is in the smooth region or in the edge region. Because human vision is sensitive to each color component differently, the proposed color activity is computed as the mean of sensitivity-weighted differences between input color in the local region of $16(4 \times 4)$ pixels and mean color ($\bar{c}_m = [\bar{r}_m, \bar{g}_m, \bar{b}_m]^T$) of all input colors in the region, as follows:

$$A(\bar{c}) = \frac{1}{16} \sum_{n=1}^{16} \Delta(\bar{c}_n, \bar{c}_m) \quad (1)$$

$$= \frac{1}{16} \sum_{n=1}^{16} \left\{ \alpha (r_n - \bar{r}_m)^2 + \beta (g_n - \bar{g}_m)^2 + \gamma (b_n - \bar{b}_m)^2 \right\}^{\frac{1}{2}},$$

where $\Delta(\bar{c}_n, \bar{c}_m)$ is the sensitivity-weighted difference and $\bar{c}_n = [r_n, g_n, b_n]^T$ is the n th input color in the local region. The coefficients (α , β , and γ) are the relative visual sensitivities of the HVS in the CIE standard, and they are the coefficients of the luminance equation. This computed activity is similarly assigned to each color included in the local region, that is, the same activity value to each color in the same local region. This is performed for all local regions of the image. Then we can obtain the activity ($A(\bar{c}_f)$) of a pixel f as the activity of the local region in which the pixel is included. According to the masking effect, a higher activity means that the color is in the edge region and its quantization error is less sensitive in human vision, whereas a lower activity means that the color is in the smooth region and its quantization error is more sensitive. Therefore, we can see that the activity is inversely proportional to the error sensitivity of the HVS.

Using this property, the new distortion measure is proposed to determine which color should be more finely quantized to decrease the distortion based on the masking effect. The total distortion is given by

$$D = \sum_{k=1}^K d_k$$

$$= \sum_{k=1}^K \sum_{\bar{c}_f \in \Omega_k} \frac{1}{A(\bar{c}_f)^2} \|\bar{c}_f - \bar{\mu}_k\|^2 \quad (2)$$

$$= \sum_{k=1}^K \sum_{\bar{c}_f \in \Omega_k} \frac{1}{A(\bar{c}_f)^2} E_q,$$

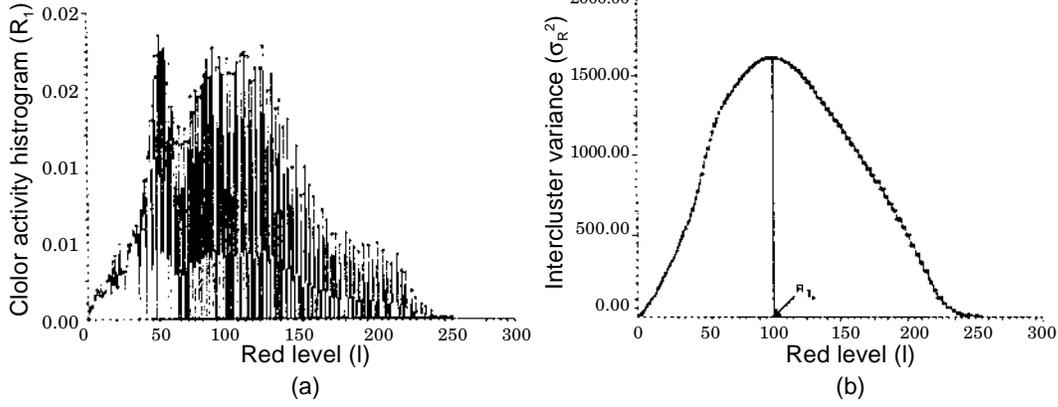


Figure 1. Activity histogram and its intercluster variance in the red component of the Peppers image: (a) activity histogram and (b) intercluster variance.

where d_k is the distortion of the k th color cluster and E_q is the quantization error. Because the input color is included in the k th cluster, the quantization error is inversely weighted by the activity of the input color. If so, the distortion error is decreased by the activity. Therefore, if the input color is in the edge region, the activity is much higher and the quantization error of the color is relatively much reduced, which means that the distortion of the cluster is decreased. If the input color is in the smooth region, the activity is much lower and the quantization error of the color is relatively less reduced, which means that the distortion of the cluster is increased. Thus, if a color cluster of maximum distortion is more finely quantized, the total distortion based on the masking effect can be much less.

The proposed quantization algorithm has a hierarchical structure, so that the color of lower activity should be more finely quantized.

Proposed Hierarchical Color Quantization Algorithm. Based on the activity and the distortion measure, the proposed color quantization algorithm can be performed in two hierarch steps. In the first step, the input colors are divided into eight initial color clusters by a thresholding method using intercluster variances.¹⁸ To compute the intercluster variance, an activity histogram (x_l , $x = R, G, \text{ or } B$ component) with respect to gray level (l , $0 \leq l \leq 255$) of each color component is obtained. The activity histogram of the red component of the Peppers image used in the experiment is shown in Fig. 1(a).

According to the activity histogram, the intercluster variance for each component is calculated as follows:

$$\sigma_x^2 = P_1 \left(\frac{1}{T_h + 1} \sum_{l=0}^{T_h} x_l - \bar{x} \right)^2 + P_2 \left(\frac{1}{255 - T_h} \sum_{l=T_h+1}^{255} x_l - \bar{x} \right)^2, \quad 1 \leq T_h \leq 255$$

$$P_1 = (x_0 + x_1 + \dots + x_{T_h}) / X$$

$$P_2 = (x_{T_h+1} + x_{T_h+2} + \dots + x_{255}) / X = 1 - P_1, \quad (3)$$

where $X = \sum_{l=0}^{255} x_l$ and $\bar{x} = X / 256$. The value T_h signifies threshold, and it is the gray value of the maximum intercluster variance. The calculated variance of Fig. 1(a) is shown in Fig. 1(b). The threshold values ($R_{T_h}, G_{T_h}, B_{T_h}$) for red, green, and blue can be selected as T_h . By using the thresholds, eight clusters are obtained, as given below.

$$\begin{aligned} \text{cluster 1 : } & r_f < R_{T_h}, g_f < G_{T_h}, \text{ and } b_f < B_{T_h} \\ \text{cluster 2 : } & r_f < R_{T_h}, g_f < G_{T_h}, \text{ and } b_f \geq B_{T_h} \\ \text{cluster 3 : } & r_f < R_{T_h}, g_f \geq G_{T_h}, \text{ and } b_f < B_{T_h} \\ \text{cluster 4 : } & r_f < R_{T_h}, g_f \geq G_{T_h}, \text{ and } b_f \geq B_{T_h} \\ \text{cluster 5 : } & r_f \geq R_{T_h}, g_f < G_{T_h}, \text{ and } b_f < B_{T_h} \\ \text{cluster 6 : } & r_f \geq R_{T_h}, g_f < G_{T_h}, \text{ and } b_f \geq B_{T_h} \\ \text{cluster 7 : } & r_f \geq R_{T_h}, g_f \geq G_{T_h}, \text{ and } b_f < B_{T_h} \\ \text{cluster 8 : } & r_f \geq R_{T_h}, g_f \geq G_{T_h}, \text{ and } b_f \geq B_{T_h} \end{aligned} \quad (4)$$

In the second step, the color cluster of maximum distortion is again divided into eight color clusters, using the proposed distortion measure and the thresholding method based on the masking effect. This process is repeatedly applied until K ($=256$ or less) color clusters are obtained. Then the color palette is composed of the centroids of the obtained color clusters, and, finally, the input colors are mapped to the color palette. The proposed algorithm is shown in Fig. 2.

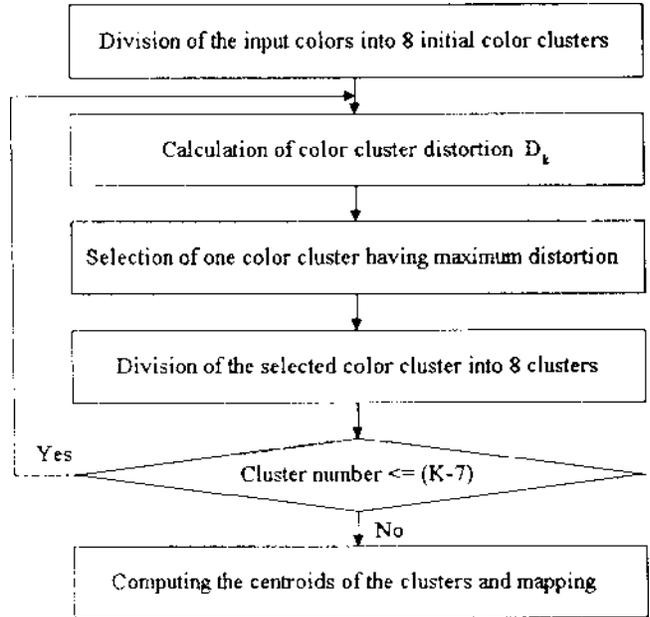


Figure 2. The proposed hierarchical color quantization algorithm.

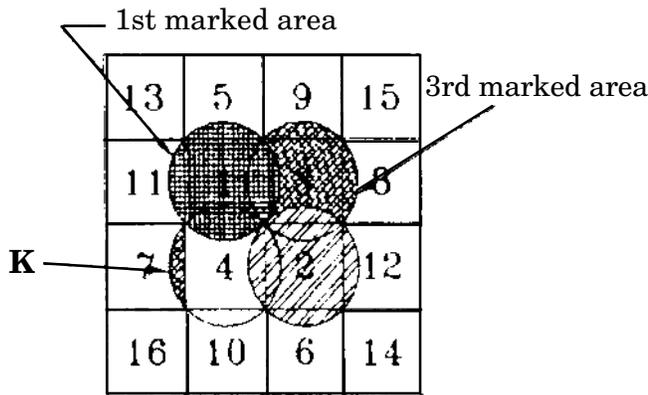


Figure 3. The marking area of dots in the dither matrix.

Proposed Color Dithering Method

Printer Model and Nonlinear Quantization. Most conventional dithering methods use linear quantization that equally divides input gray level according to printer resolution. This algorithm has simple processing and less computational time. However, because the algorithm does not consider the hardware characteristics of the printer, a gray-level difference between the printed image and the displayed image on the monitor is produced.

As shown in Fig. 3, printers produce circular black dots, rather than square ones. The most elementary degradation introduced by most printers is illustrated as follows: their dots are larger than the minimal covering size, as if “ink spreading” has occurred. Other degradations are caused by heat finishing, reflections of light within the paper, etc. Also the inks are not perfect. As a result, the gray level produced by the printer in the vicinity of pixel (i, j) depends in some complicated way on neighboring dots. However, due to the close spacing of dots and the limited spatial resolution of the eye, the gray level can be modeled as having a constant value within the area of the ideal circular pixel (i, j) .

Therefore, we developed a quantization method that improves the quality of printed images. The method decreases the quantization error and distributes the remaining quantization error more effectively, both spatially and across the intensity range.

We propose here a nonlinear quantization that accounts for the dot overlap degradation due to hardware characteristics. This approach is adequate to illustrate quantization errors produced in the multilevel quantization due to printer resolution.

TABLE I. Quantized Level and Reconstructed Gray Level Based on Marking Area.

Quantized level	N	P_N	Reconstructed gray level (r)	Increment
1	1	0	230	25
2	2	0	205	25
3	3	2	189	16
4	4	4	173	16
5	5	5.5	154	19
6	6	7	136	18
7	7	8.5	118	18
8	8	10	100	18
9	9	12.5	87	13
10	10	15	73	14
11	11	17.5	59	14
12	12	20	46	13
13	13	23	34	12
14	14	26	23	11
15	15	29	11	12
16	16	30	0	11

In black-and-white printing, all the overlapping areas have the same gray level as the dots, so that the reproduced gray level is proportional to the marked area. The marked area in the printing is changed according to marking position and dither matrix. If the difference in the marking area is not considered, the printed image is not equal in gray level to the image displayed on the monitor.

The proposed nonlinear quantization for dithering can improve this drawback. The quantization step is adjusted in proportion to the overlapping area, which is subtracted from the total area of each printed neighbor dot. The total area of each pixel is the area of the 4×4 dither matrix in the 300-dpi printer, and the area of each dot is the area of a circle. Therefore, the reconstructed gray level due to actual marked area is calculated as follows:

$$r = \left\{ 1 - N \times \left(\frac{\pi}{32} \right) + P_N \times K' \right\} \times 255, \quad (5)$$

where r is the reconstructed gray level, N is the number of marked dots, P_N is the sum of the overlapped area and half of the dots marked in the outer region of the 4×4 dither matrix among the marked dots, K' is $2 \times K$, and the area of K is $((\pi - 2)/32)/4$. The actual marked areas and their calculated gray levels are shown in Table I. In the printed image, this nonlinear quantization reduces the degradation that is generated due to hardware characteristics.

Modified Dot Diffusion. The ordered dither method produces a binary-recursive and computerized texture that is unsuitable for most applications. The Floyd–Steinberg error diffusion method usually gives more pleasing results but it has occasional problems with intrusive, snakelike patterns that call attention to themselves. It would be desirable to have a solution that does not retain both previous properties but has both the sharpness of the Floyd–Steinberg method and the parallelism of ordered dither. Dot diffusion has the desired property but it tends to blur the image in the edge regions through the diffusion of the quantization error. To prevent the diffusion of the quantization error between the neighboring different gray-level regions, edge detection and thresholding may be considered, but the methods need complex, lengthy computation. Therefore, a fast and adaptive diffusion algorithm is required.

The frequencies of the quantization errors are, in general, not homogeneously distributed. These frequencies can be divided into two components, one of which is dependent on the object frequencies and one that is independent. In printing, the separation of the object information from the quantization error during the reconstruction is desired. The linear component of the object-dependent error component lies at the same frequency as the object information and can be much stronger than all other error components. Also, most of the object-dependent components of the quantization error lie at the low frequency. Thus, control over the object-dependent error is desired to allow better reconstruction.

We propose a modified dot diffusion that diffuses the object-dependent components of the quantization error. To compensate local gray level in the smooth region and maintain sharpness in the edge regions, the input image should be modified. We divide the input image into two parts, one of which is quantized values and the other is quantization errors. To modify the input image, filtering the quantization error by a low-pass filter before dot diffusion is suggested. The filtered errors include most of the object-dependent components except for edge components; therefore, the input image is modified by the sum of the filtered

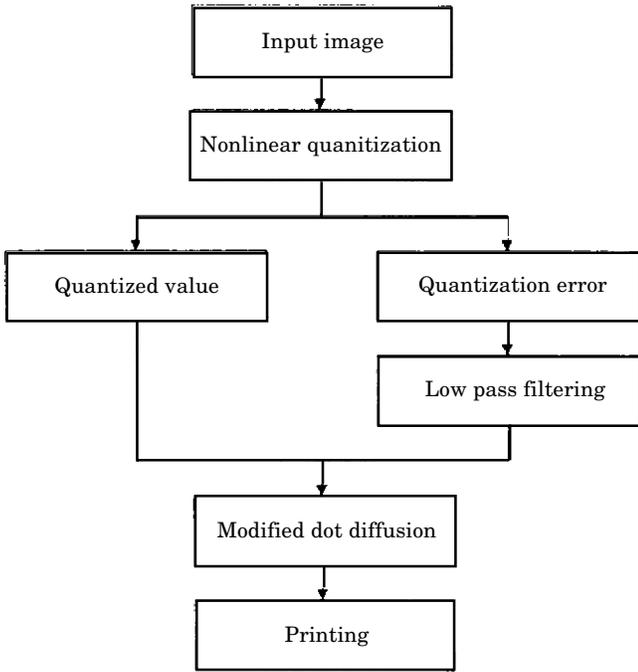


Figure 4. Block diagram for printing.

errors and the quantized values. In the dithering process, dot diffusion is applied to the modified image so that diffusion of the quantization error can be minimized in the edge regions and retained in the smooth region. Thus the proposed modified dot diffusion can compensate local gray level in the smooth region and retain sharpness in the edge regions. In this work, the 3×3 average filter is used for simple and fast processing. The block diagram for printing is shown in Fig. 4.

Modified Dot Diffusion for Color Printing. In this section we present models for color printers and modified dot diffusion for color printing. Such printers are capable of producing colored dots on a piece of paper, at any and all sites of a Cartesian grid with horizontal and vertical spacing. The reciprocal of spacing is the printer resolution in dots per inch (dpi). Color printers use cyan (C), magenta (M), and yellow (Y) inks to produce color dots. These colors form the basis for the subtractive system of colors. The relationship to the additive colors (RGB) follows:

$$\begin{aligned} R &= 1 - C \\ G &= 1 - M \\ B &= 1 - Y \end{aligned} \quad (6)$$

Cyan ink absorbs red light, magenta absorbs green, and yellow absorbs blue. Dots of different inks can be superimposed to produce red, green, blue, and black dots. In practice, however, the inks are not perfect, i.e., there are unwanted absorptions. Thus, many printers use a separate black ink (K) to produce better black dots. In the remainder of this discussion we will assume that the printer uses all four types of ink. The printer is controlled by an array of four-dimensional vectors with binary components,

$$b_{i,j} = (b_{i,j}^C, b_{i,j}^M, b_{i,j}^Y, b_{i,j}^K), \quad (7)$$

where $b_{i,j}^C = 1$ indicates that a cyan dot is to be placed at pixel (i, j) and $b_{i,j}^C = 0$ indicates that no cyan dot is to be placed at the site. The magenta $b_{i,j}^M$, yellow $b_{i,j}^Y$, and black $b_{i,j}^K$ components are defined similarly. When all components are zero, the site remains white. When more than

TABLE II. The Comparison of Peak Signal-to-Noise Ratio (PSNR), Quantization Errors (Q_e) in CIE (Lu^*v^*), and Computation Time Using PC in Each Algorithm

Image	Algorithm	PSNR [dB]	Q_e in Lu^*v^*	Time [s]
Girl	LBG	30.6	15.13	4150
	Heckbert	28.86	22.86	11
	The proposed	30.87	11.37	24
Lena	LBG	30.15	5.37	2307
	Heckbert	29.80	6.20	11
	The proposed	31.83	4.56	24
Pepper	LBG	28.92	8.72	3262
	Heckbert	25.88	19.57	11
	The proposed	29.28	8.42	24
Zelda	LBG	31.71	4.08	1828
	Heckbert	29.66	5.29	11
	The proposed	34.80	3.35	24

one component is equal to 1, different inks are printed on top of each other. In principle, we can specify $2^4 = 16$ different colors for each dot, but nine of these colors are variations of black. Usually, the black ink is used only in combination with the other three inks to produce solid black dots, thus reducing the number of colors that each dot can include to $2^3 = 8$.

As we can see in black-and-white printing, printers produce circular, rather than square, dots, and the quantization error produced in the quantization process must be compensated. To apply the proposed algorithm to color printing, color change, as well as intensity, must be considered. Thus, quantization error is diffused according to the characteristics of the input image in the HSI color-coordinate system, which represents HVS color-sensing properties. In the dithering process, dot diffusion is modified by adjusting the amount of hue and saturation diffusion according to the intensity and saturation of the input image in the HSI color-coordinate system. The HSI coordinate system has higher sensitivity in the highest and lowest intensities and lower saturation in the mid-range intensities. To compensate the nonuniform property in the HSI coordinate system, the amount of diffusion is given by

$$\begin{aligned} \Delta H' &= \begin{cases} \Delta H \times S \times \frac{I}{127}, & I \leq 127 \\ \Delta H \times S \times \frac{(255 - I)}{127}, & \text{otherwise} \end{cases}, \\ \Delta S' &= \begin{cases} \Delta S \times S \times \frac{I}{127}, & I \leq 127 \\ \Delta S \times S \times \frac{(255 - I)}{127}, & \text{otherwise} \end{cases} \end{aligned} \quad (8)$$

where $\Delta H'$ and $\Delta S'$ are the hue and saturation to be diffused, ΔH and ΔS are the hue and saturation errors that occurred in the quantization process, S is the saturation value, and I is the intensity value.

In color printing using modified dot diffusion, we suggest filtering the quantization error by a low-pass filter before dot diffusion, because the quantization error must be diffused only in the smooth region. Therefore, the printed image can prevent color degradation through the diffusion of quantization error between neighboring regions of different color.

Experimental

In the experiments, we used a PC with a VGA board of 256 colors and an HP DeskJet 560K 300-dpi ink-jet printer. Color quantization operations were performed separately



(a)



(b)



(c)



(d)

Figure 5. Original and result image of various quantization algorithms: (a) original image, (b) result of LBG algorithm, (c) result of Heckbert algorithm, and (d) result of the proposed algorithm.

Figure 6. Magnified images of Fig. 5: (a) Magnified portion original, (b) magnified image from LBG algorithm, (c) magnified image from Heckbert algorithm, and (d) magnified image from the proposed algorithm.



(a)



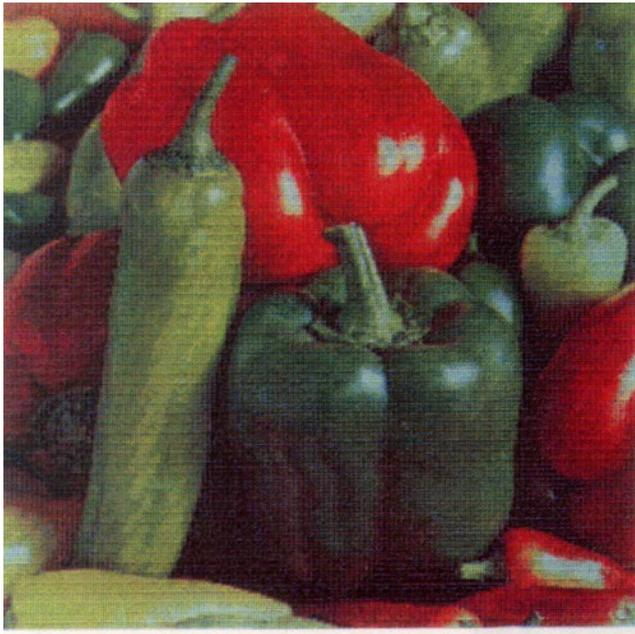
(b)



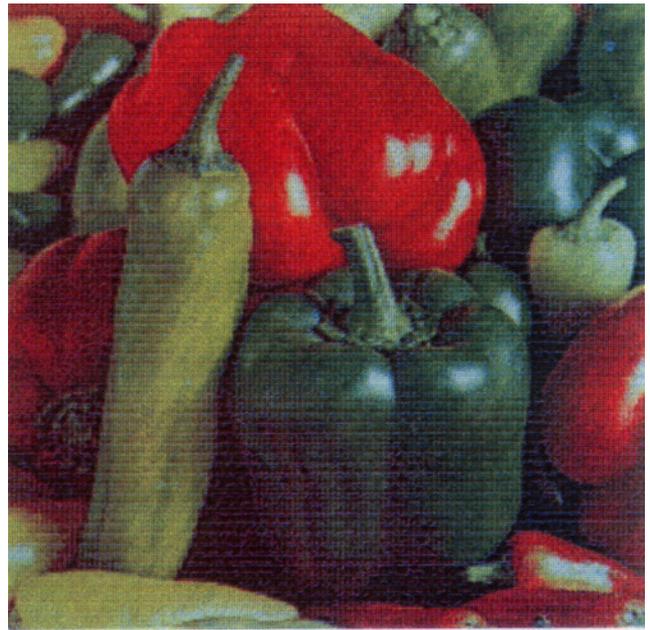
(c)



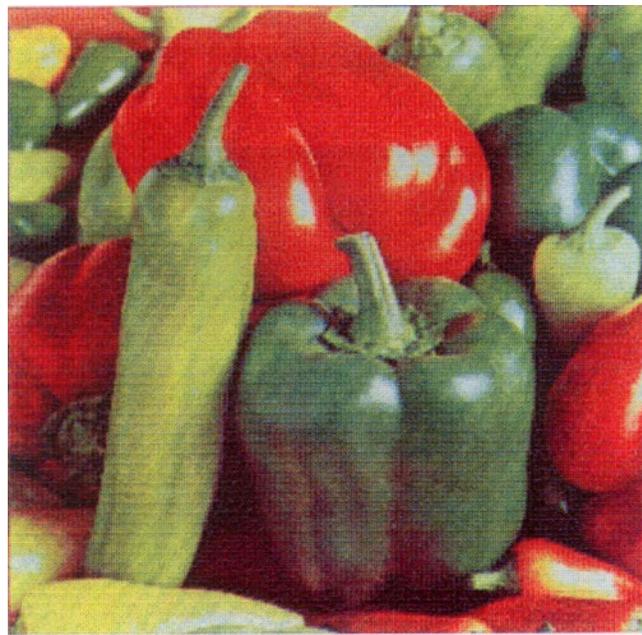
(d)



(a)



(b)



(c)

Figure 7. Printed image using various methods: (a) Printed image using linear quantization and ordered dithering, (b) printed image using linear and dot diffusion, and (c) printed image using the proposed method.

on each color component: R, G, and B. The images used in the experiment were 256×256 Girl, Lena, Peppers, and Zelda. Table II compares the peak signal-to-noise ratio (PSNR), quantization errors in uniform color coordinate system space, and computation time, using the PC in three algorithms (LBG, Heckbert, and the proposed one). In the table, the proposed algorithm takes a little longer computation time than Heckbert's algorithm, but it takes much shorter computation time than that for LBG. Figure 5 shows the results of the color quantization with the Girl image. Here Fig. 5(a) is the original image, 5(b) is the result of the LBG algorithm, 5(c) is the result of the Heckbert algorithm, and 5(d) is the result of the proposed algorithm. Figure 6 compares magnified images of Fig. 5. The displayed image on the monitor using the proposed color quantization method shows almost no noticeable difference from the result of the LBG method and from the original image. Figure 7 shows printed images using (a) linear quantization and ordered dithering, (b) linear and dot diffusion, and (c) the proposed method, along with the original image. The printed result using the proposed color dithering method is of high quality and shows less color degradation than that of the control image produced by a conventional printing method.

Summary and Conclusion

New methods for displaying and printing high-resolution full-color images on limited color output devices are proposed. The color quantization for palette design consists of a hierarchical method based on HVS characteristics. This technique uses an activity-weighted distortion measure based on the relative visual sensitivity and spatial masking effect of the HVS. In color printing, dithering of nonlinear quantization is proposed to reduce color degradation by considering the overlapping phenomena

of printing dots. In the dithering process, modified dot diffusion is also proposed. The proposed techniques enable limited-color-output devices to display and print high-quality color images. ▲

References

1. Y. Linde, A. Buzo, and R. Gray, An algorithm for vector quantizer design, *IEEE Trans. Commun.* **COM-28** (1): 84–95, (1980).
2. P. S. Heckbert, Color image quantization for frame buffer display, *Comput. Graph.* **16** (3), 297–307 (1982).
3. N. Goldbert, Color image quantization for high resolution graphics display, *Image Vision Comput.* **9** (5) (1991).
4. M. T. Ochar and C. A. Bouman, Color quantization of images, *IEEE Trans. Signal Process.* **39** (12) (1991).
5. A. Tremeau, M. Calonnier, and B. Laget, Color quantization error in terms of perceived image quality, in *1994 ICASSP*.
6. R. Balasubramanian, Color-image quantization with use of fast binary splitting techniques, *J. Opt. Soc. Amer. A*, **11** (11), 1994.
7. T. S. Liu and L.-W. Chang, Fast color image quantization with error diffusion and morphological operations, *Sig. Process.* **43**: 293–303 (1995).
8. A. Netravali and B. Prasada, Adaptive quantization of picture signals using spatial masking, in *Proc. IEEE* **65**: 536–548 (1977).
9. J. D. Mollon and L. T. Sharps, *Color Vision, Physiology and Psychophysics*, Academic Press Inc. (1983).
10. R. Eschbach and K. T. Knox, Error diffusion algorithm with edge enhancement, *J. Opt. Soc. Amer. A*, **8** (12): 1844–1850 (1991).
11. R. Ulichney, *Digital Halftoning*, MIT Press, Cambridge, Massachusetts, (1987).
12. R. S. Gentile, E. Walowit, and J. P. Allebach, Quantization and multi-level halftoning of color images for near-original image quality, *J. Opt. Soc. Amer. A* **7**: 1019–1026 (1990).
13. S. C. Wells, G. J. Williamson, and S. E. Carrie, Dithering for 12-bit true-color graphics, *IEEE Comput. Graph. Appl.*: 18–29 (1991).
14. D. E. Knuth, Digital halftones by dot diffusion, *ACM Trans. Graph.* **6** (4): 245–273 (1987).
15. F. Fettuer and O. Bryngdahl, Quantization noise and the error diffusion algorithm, *J. Elec. Imaging* **3**(1): 37–44 (1994).
16. H. G. Lamming and W. L. Rhodes, A simple method for improved color printing of monitor images, *ACM Trans. Graph.* **9** (4): 345–375 (1990).
17. M. C. Stone, W. B. Cowan, and J. C. Beatty, Color gamut mapping and the printing of digital color images, *ACM Trans. Graph.* **7**(4): 249–292 (1988).
18. N. Otsu, A threshold selection method for gray level histograms, *IEEE Trans. Sys. Man and Cybern.* **SMC-9**: 62–66 (1979).