Color Image Quantization and Dithering Method based on HVS 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
E-mail: yhha@ee.kyungpook.ac.kr

Abstract

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

Keywords: color quantization, distortion measure, color dithering, and dot diffusion

1. Introduction

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

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 in order to eliminate distinguishable quantization steps in tri-chromatic 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 display on the 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 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 color palette from the full range of 2^24 colors. The drawback of this scheme is that it restricts the number of colors which may be simultaneously displayed.[1-7]

Since 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. It 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. It is, however, computationally intensive and its performance is sensitive to the choice of the initial palette. Also there is a class of 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 algorithm, the median-cut algorithm[2] 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 - the plane which divides the box into two halves so that equal number of colors are on each side. The main advantage of the split algorithm is that it has lower computational time cost and memory space for the spatial storage scheme - mostly because it is simple to compute the split point. There remain, however, a number of problems associated with this method. One of those 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.\(^{[5,7]}\) It does not take into account the spatial correlation that linked with perceptually adjacent pixels. With such a measure we have no means to know whether the observed degradation is the result of several particularly noticeable degradation. 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 region of input image. The activity is computed as mean of sensitivity-weighted difference between input and local mean color in 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 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 continuous tone image on the paper media for hard copy. It is impossible to display continuous tone images on the paper media without the dithering process. Thus, many dithering techniques can be found in the printing algorithm.\(^{[10-13]}\) 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 local gray level.\(^{[14-17]}\) In this process, accurate gray levels are not represented because of printer’s characteristic, generally, and diffusion of the contrast in the edge region is not considered. Therefore, new method is required to reduce the degradation of image produced in the conventional dithering method using linear quantization and diffusion.

For printing the monitor-displayed image on the paper media, we propose nonlinear quantization that considers the overlapping of printing dots and modified dot-diffusion method. For the nonlinear quantization, quantization step is adjusted in proportion to the area which overlapping area is subtracted from total area of each neighboring printing dot. In order to compensate errors produced in the quantization, the modified dot-diffusion adjusts the quantization error to be diffused according to the characteristic of input image. Filtering of the quantization error by low pass filter before dot-diffusion is carried out so that the quantization error has to be diffused only in the smooth region. Also to apply the proposed method to the color printing, intensity and color change must be considered simultaneously. Thus, quantization error is diffused in the HSI color coordinate which 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 paper is organized as follows. In section 2, starting by defining the color space in which the quantization algorithm operates, the proposed distortion measure and hierarchical quantization algorithm are described. Section 3 describes the printer model, the proposed nonlinear quantization, and the modified dot-diffusion method for color printing. Finally, Section 4 reports the simulation results of the proposed techniques. And Section 5 describes the conclusion.

2. The proposed color quantization algorithm

The color quantization for palette design is selecting the prescribed number of colors to display an image with almost no noticeable perceived difference. The process is usually done by treating three color components (red, green, and blue) independently in 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 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 the image compression systems. Thus, we quantize the colors of original image into K (usually K=256 or less) colors, called 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 \( \tilde{c}_f = [r_f, g_f, b_f]^T \), 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 paper, the \( k \)-th cluster is denoted by \( \Omega_k \) \((1 \leq k \leq K)\) and the centroids of the clusters are denoted by \( \tilde{\mu}_k = [r_k, g_k, b_k]^T \), \((1 \leq k \leq K)\), 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 is to find the optimal color cluster and its centroid with almost no difference between the original image and the reconstructed image.

### 2.1. Weighted-distortion measure of HVS color activity

In this paper, a new distortion measure based on the spatial masking effect of HVS is proposed. The spatial masking effect means that human vision is more sensitive to quantization errors in the smooth region than 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 differently sensitive to each color component, the proposed color activity is computed as mean of sensitivity-weighted difference between input color in the local region of \(16 \times 16\) pixels and mean color \((\tilde{c}_n) = [\tilde{r}_n, \tilde{g}_n, \tilde{b}_n]^T\) of all input colors in the region, as follows.

\[
A(\tilde{c}) = \frac{1}{16} \sum_{m=0}^{15} \Delta(\tilde{c}_m, \tilde{c}_n)
\]

\[
= \frac{1}{16} \sum_{m=0}^{15} [\alpha(r_m - \tilde{r}_n) + \beta(g_m - \tilde{g}_n) + \gamma(b_m - \tilde{b}_n)]^T
\]  

(1)

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

\[
D = \sum_{k=1}^{K} d_k
\]

\[
= \sum_{k=1}^{K} \sum_{m=1}^{n} \frac{1}{A(\tilde{c}_f)} \| \tilde{c}_f - \tilde{\mu}_k \|^2
\]

\[
= \sum_{k=1}^{K} \sum_{m=1}^{n} \frac{1}{A(\tilde{c}_f)} E_q
\]

(2)

where \( d_k \) is the distortion of \( k \)-th color cluster and \( E_q \) is the quantization error. As the input color is included in \( k \)-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 decreased, 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 decreased, which means that the distortion of the cluster is increased. Thus if a color cluster of maximum distortion is more finely quantized, the whole distortion can be much more decreased based on the masking effect.

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

### 2.2. The proposed hierarchical color quantization algorithm
Based on the activity and the distortion measure, the proposed color quantization algorithm can be performed in hierarchically two steps. In the first step, the input colors are divided into 8 initial color clusters by thresholding method using inter-cluster variance. To compute the inter-cluster variance, activity histogram ($x_i$, $i = R, G, B$ 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 Peppers image used in the experiment is shown in Fig. 1(a).

\[
\sigma_i^2 = P_1 \left( \frac{1}{T_i} + \frac{1}{255} \sum_{l=0}^{255} x_i - \bar{x} \right)^2 + P_2 \left( \frac{255 - T_i}{255} \sum_{l=0}^{255} x_i - \bar{x} \right)^2, \quad l < T_i \leq 255
\]

\[
P_1 = \frac{1}{X} \sum_{i=0}^{X} x_i
\]

\[
P_2 = \frac{1}{X} \sum_{i=R, G, B} x_{i+1} \quad \text{for} \quad \text{cluster 1 to 8}
\]

where $X = \frac{255}{n}$, and $\bar{x} = X / 256$. $T_i$ means threshold value and it is the gray value of maximum inter-cluster variance. Then the calculated variance with Fig. 1(a) is shown in Fig. 1(b). The threshold values ($R_T, G_T, B_T$) for red, green, and blue can be selected as $T_i$. By the thresholds, 8 clusters are obtained, as given below.

cluster 1: $r_i < R_T, g_i < G_T, b_i < B_T$
cluster 2: $r_i < R_T, g_i < G_T, b_i \geq B_T$
cluster 3: $r_i < R_T, g_i \geq G_T, b_i < B_T$
cluster 4: $r_i < R_T, g_i \geq G_T, b_i \geq B_T$
cluster 5: $r_i \geq R_T, g_i < G_T, b_i < B_T$
cluster 6: $r_i \geq R_T, g_i < G_T, b_i \geq B_T$
cluster 7: $r_i \geq R_T, g_i \geq G_T, b_i < B_T$
cluster 8: $r_i \geq R_T, g_i \geq G_T, b_i \geq B_T$

In the second step, the color cluster of maximum distortion is again divided into 8 color clusters using the proposed distortion measure and 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.

3. The proposed color dithering method
3.1. The printer model and nonlinear quantization

Most of 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 it doesn’t consider hardware characteristic of printer, gray level difference between printed image and 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 following: their dots are larger than the minimal covering size, as if "ink spreading" is occurred. Other degradation are caused by the heat finishing, reflections of light within the paper, and so on. Also, the inks are not perfect. As a result, the gray level produced by the printer in the vicinity of pixel (i, j) depends on some complicated way and 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 develop a quantization method that improves the quality of printed images. It must decrease the quantization error and distribute the remaining quantization error more effectively, both spatially and across the intensity range.

![Fig. 3. The marking area of dots in the dither matrix.](image)

Table 1. Quantized level and reconstructed gray level based on marking area.

<table>
<thead>
<tr>
<th>Level</th>
<th>N</th>
<th>$P_N$</th>
<th>$r$ (255)</th>
<th>Increment</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>230</td>
<td>25</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>0</td>
<td>205</td>
<td>25</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2</td>
<td>189</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>4</td>
<td>173</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5.5</td>
<td>154</td>
<td>19</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td>7</td>
<td>136</td>
<td>18</td>
</tr>
<tr>
<td>7</td>
<td>7</td>
<td>8.5</td>
<td>118</td>
<td>18</td>
</tr>
<tr>
<td>8</td>
<td>8</td>
<td>10</td>
<td>100</td>
<td>18</td>
</tr>
<tr>
<td>9</td>
<td>9</td>
<td>12.5</td>
<td>87</td>
<td>13</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>15</td>
<td>73</td>
<td>14</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>17.5</td>
<td>59</td>
<td>14</td>
</tr>
<tr>
<td>12</td>
<td>12</td>
<td>20</td>
<td>46</td>
<td>13</td>
</tr>
<tr>
<td>13</td>
<td>13</td>
<td>23</td>
<td>34</td>
<td>12</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>26</td>
<td>23</td>
<td>11</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>29</td>
<td>11</td>
<td>12</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>30</td>
<td>0</td>
<td>11</td>
</tr>
</tbody>
</table>

In this paper, we propose a nonlinear quantization that accounts for the dot overlap degradation due to hardware characteristic. This is adequate to illustrate quantization errors produced in the multilevel quantization due to printer resolution.

In the black and white printing, all the overlapping areas have the same gray level as the dots. So, the reproduced gray level is proportional to marked area. Marked area in the printing is changed according to marking position and dither matrix. If the difference of marking area is not considered, the printed image doesn't equal with gray level of the image displayed on monitor.

The proposed nonlinear quantization for dithering can improve this drawback. Quantization step is adjusted in proportion to the area which overlapping area is subtracted from total area of each printed neighbor dot. Total area of each pixel is the area of $4 \times 4$ 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.

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

where $r$ is reconstructed gray level, $N$ is number of the marked dots, $P_N$ is the sum of overlapped area and half of dots marked in outer region of $4 \times 4$ dither matrix among the marked dots, $K’ = 2 \times K$, and the area of $K$ is $((\pi - 2)/32)/4$. The actual marked areas and their calculated gray levels are shown in Table 1. In the printed image, this nonlinear quantization reduces the degradation which is generated due to hardware characteristic.
3.2. Modified dot-diffusion

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

The frequencies of the quantization errors is, in general, not homogeneously distributed. The frequencies of the quantization errors can be divided into two components, one of which is dependent on the object frequencies and the other is independent one. 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 frequencies as the object information and can be much stronger than all other error components. And, most of object-dependent components of quantization error lies at the low frequency. Thus, control over the object-dependent error is desired to allow better reconstruction.

In this paper, we propose modified dot-diffusion that diffuses object-dependent component of quantization error. To compensate local gray level in the smooth region and maintain sharp property in the edge regions, input image should be modified. In this paper, input image is divided into two parts one of which is quantized values and the other is quantization errors. To modify input image, filtering the quantization error by low pass filter before dot diffusion is suggested. The filtered errors have most of object-dependent component except edge components, therefore, input image is modified by sum of filtered errors and quantized values. In the dithering process, dot diffusion is applied to modified image so that diffusion of quantization error can be minimized in the edge regions and retained in the smooth region. So, the proposed modified dot diffusion can compensate local gray level in the smooth region and retain sharpness in the edge regions. In this paper, the 3x3 average filter is used for simple and fast processing. The block diagram for printing is shown in Fig. 4.

![Block diagram for printing.](image)

3.3. 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) is followed.

\[
\begin{align*}
R &= 1 - C \\
G &= 1 - M \\
B &= 1 - Y
\end{align*}
\]

(6)

Cyan ink absorbs red light, magenta absorbs green, and yellow absorbs blue. Different dots of ink can be printed on top of each other 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 paper 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.
\begin{align*}
b_{i,j} = (b_{i,j}^C, b_{i,j}^M, b_{i,j}^Y, b_{i,j}^K)
\end{align*}

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 \(b_{i,j}^K\) component are defined similarly. When all components are zero, the site is the remain white. When more than one component is equal to 1, different inks are printed on top of each other. In principle, we can specify \(2^3 = 8\) different colors for each dots, but 9 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 take 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 has to 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 characteristic of 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 input image in the HSI color coordinate system. The HSI coordinate system has higher sensitivity in the top and bottom intensities and lower saturation than in the middle intensity and high saturation. To compensate non-uniform property in the HSI coordinate system, the amount of diffusion is given by

\begin{align*}
\Delta H' &= \begin{cases} 
\frac{\Delta H \times S \times I}{127} & , \quad 1 \leq I \\
\frac{\Delta H \times S \times (255 - I)}{127} & , \quad otherwise
\end{cases} \\
\Delta S' &= \begin{cases} 
\frac{\Delta S \times S \times I}{127} & , \quad 1 \leq I \\
\frac{\Delta S \times S \times (255 - I)}{127} & , \quad otherwise
\end{cases}
\end{align*}

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

In color printing using the modified dot-diffusion, filtering the quantization error by low pass filter before dot-diffusion is suggested because the quantization error has to be diffused only in the smooth region. Therefore, the printed image can prevent color degradation due to the diffusion of quantization error between the neighboring different color regions.

### 4. Experimental results

In the experiment, we used a PC with VGA board of 256 colors and a HP Deskjet 560K of 300 dpi inkjet printer. Color quantization operations are performed separately on each color component, R, G, and B. The images used in the experiment are 256\&\times 256 Girl, Lena, Peppers, and zelda. Table 2 has the comparison of the PSNR, quantization errors in uniform color coordinate system space, and computation time using the PC in three algorithms (LBG, Heckbert, and the proposed). In the table, the proposed algorithm takes a little longer computation time than Heckbert’s algorithm but it takes much shorter computation time than LBG. Fig. 5 shows the results of the color quantization with the Girl image. Here, (a) is the original image, (b) is the result of LBG algorithm, (c) is the result of Heckbert algorithm, and (d) is the result of the proposed algorithm. Fig. 6 shows the magnified images of Fig. 5, to compare them. The displayed image on the monitor using the proposed color quantization method shows almost no noticeable difference comparing to the result of LBG method and the original image. Fig. 7 shows printed image using (a) linear quantization and ordered dithering, (b) linear and dot diffusion, and (c) the proposed method, namely with the original image. The printed result using the proposed color dithering method shows high quality and less color degradation than the conventional printing method.

<table>
<thead>
<tr>
<th>Image</th>
<th>Algorithm</th>
<th>PSNR [dB]</th>
<th>(Q_e) in Lu*+v*</th>
<th>Time [sec]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Girl</td>
<td>LBG</td>
<td>30.6</td>
<td>15.13</td>
<td>4150</td>
</tr>
<tr>
<td></td>
<td>Heckbert</td>
<td>28.86</td>
<td>22.86</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>The proposed</td>
<td>30.87</td>
<td>11.37</td>
<td>24</td>
</tr>
<tr>
<td>Lena</td>
<td>LBG</td>
<td>30.15</td>
<td>5.37</td>
<td>2307</td>
</tr>
<tr>
<td></td>
<td>Heckbert</td>
<td>29.80</td>
<td>6.20</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>The proposed</td>
<td>31.83</td>
<td>4.56</td>
<td>24</td>
</tr>
<tr>
<td>Pepper</td>
<td>LBG</td>
<td>28.92</td>
<td>8.72</td>
<td>3262</td>
</tr>
<tr>
<td></td>
<td>Heckbert</td>
<td>25.88</td>
<td>19.57</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>The proposed</td>
<td>29.28</td>
<td>8.42</td>
<td>24</td>
</tr>
<tr>
<td>Zelda</td>
<td>LBG</td>
<td>31.71</td>
<td>4.08</td>
<td>1828</td>
</tr>
<tr>
<td></td>
<td>Heckbert</td>
<td>29.66</td>
<td>5.29</td>
<td>11</td>
</tr>
</tbody>
</table>
5. Conclusion

New methods for displaying and printing a full resolution color image on limited color output devices have been proposed. The color quantization for palette design consists of hierarchical method based on HVS characteristics. This technique uses the activity-weighted distortion measure based on the relative visual sensitivity and spatial meaking effect of HVS. In the 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 was also proposed. The proposed techniques enables the limited-color output devices to be used to display and print full resolution color images.

References

Fig. 5. Original and result image of various quantization algorithm. (a) Original image. (b) Result of LBG algorithm. (c) Result of Heckbert algorithm. (d) Result of the proposed algorithm.
Fig. 6. Magnified images of Fig. 5. (a) Magnified image of original. (b) Magnified image of LBG algorithm. (c) Magnified image of Heckbert algorithm. (d) Magnified image of the proposed algorithm.
Fig. 7. Printed image using various printing method. (a) Printed image using linear quantization and ordered dithering. (b) Printed image using linear and dot diffusion. (c) Printed image using the proposed method.