Optimized Universal Color Palette Design for Error Diffusion [†‡]

Bernd W. Kolpatzik
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-1285
email: *bernd@ecn.purdue.edu*

Charles A. Bouman
School of Electrical Engineering
Purdue University
West Lafayette, IN 47907-1285
(317) 494-0340
email: *bouman@ecn.purdue.edu*

November 15, 2004

Abstract

Currently, many low cost computers can only simultaneously display a palette of 256 colors. However, this palette is usually selectable from a very large gamut of available colors. For many applications, this limited palette size imposes a significant constraint on the achievable image quality.

In this paper, we propose a method for designing an optimized universal color palette for use with halftoning methods such as error diffusion. The advantage of a universal color palette is that it is fixed and therefore allows multiple images to be displayed simultaneously. In order to design the palette, we employ a new vector quantization method known as sequential scalar quantization (SSQ) to allocate the colors in a visually uniform color space. The SSQ method achieves near optimal allocation, but may be efficiently implemented using a series of look up tables. When used with error diffusion, SSQ adds little computational overhead, and may be used to minimize the visual error in an opponent color coordinate system.

We compare the performance of the optimized algorithm to standard error diffusion by evaluating a visually weighted mean squared error measure. Our metric is based on the color difference in CIEL*a*b*, but also accounts for the lowpass characteristic of human contrast sensitivity.

[†]This work was supported by an NEC Faculty Fellowship.
[‡]*Journal of Electronic Imaging*, vol. 4, no. 2, pp. 131-143, April 1995.

1 Introduction

In recent years, there has been a dramatic increase in the need for moderate and low cost equipment to display digital color images. While many of the applications which drive this need require the highest possible quality, cost considerations often restrict color displays to 8 bit video memory. This 8 bit restriction only allows the simultaneous display of 256 colors from the full gamut of 2^{24} possible colors. The selection of this palette of colors is then of critical importance.

There are a number of approaches for ameliorating the effects of a restricted color palette. "Palettization" techniques work by choosing a palette which best represents a particular image [1, 2, 3, 4]. This method yields high quality results, but does not allow for the simultaneous display of multiple images. This is because the palette required for each image will be different, so the combined palette for multiple images will generally be too large. Iverson and Riskin [5] have proposed a method for combining image palettes, but such a process must inevitably lead to degradation of image quality.

The alternative approach, studied in this paper, is to use an optimized universal palette. The advantage of a universal color palette is that multiple images can be displayed simultaneously, since each image uses the same palette. Unfortunately, direct quantization using a universal color palette generally yields much lower image quality then an image dependent palette. Therefore, halftoning algorithms such as multilevel dithering [6] or error diffusion [7, 8] must be used to improve the visual quality of the displayed images. These methods exploit the lowpass nature of the human visual system to hide color quantization artifacts.

A variety of approaches to universal color palette design have been previously studied. Goertzel and Thompson [9] examined separable color palettes in RGB for use with error diffusion. They found that image quality was improved by distributing quantization levels along the three primaries based on the L^* component of the CIE$L^*a^*b^*$ color space. However, the separable nature of their RGB structure restricts the optimality of the resulting palette. Alternatively, Gentile, Walowit and Allebach [6] found that a nonseparable universal color

palette designed in L*u*v* coordinates gave superior performance over one designed in RGB coordinates. However, since the quantizer is not separable with respect to L*u*v*, quantization of each color requires a computationally expensive search of the entire palette. More recently, Venable, Stinehour and Roetling [10] designed an optimized universal color palette based on uniform separable quantization of a scaled CIE$L^*a^*b^*$ color space. However, since the display gamut is no longer a cube in the $L^*a^*b^*$ color space, straight forward application of this method leads to waisted colors due to gamut mismatch. Also, use of this palette in error diffusion requires a transformation to the $L^*a^*b^*$ color space.

In this paper, we present a method for designing a universal color palette which minimizes visual error and allows very fast quantization. Our approach is based on vector quantization (VQ) methods in a uniform color space CIEL*a*b*, but it employs a recently developed VQ technique call sequential scalar quantization (SSQ) [11, 4, 12]. The SSQ method uses a structured codebook to uniquely combine the performance advantages of a vector quantizer with the speed of a separable scalar quantizer. In practice, the quantization of a pixel into an SSQ palette may be implemented by using a sequence of three lookup tables; thus, a complete search through the color palette can be avoided.

To improve the subjective quality of the quantized image we apply the optimized color palette in conjunction with a visually optimized error diffusion technique described in [13]. Generally, error diffusion distributes errors at higher spatial frequencies and thereby reduces human visual sensitivity to those errors. In addition, optimized error diffusion exploits the difference in modulation transfer functions for luminance and chrominance components of color, and thus further reduces the perceived error of the displayed images.

Furthermore, we show how our optimized color palette may be efficiently combined with color error diffusion. In order to produce accurate color matches, the error diffusion algorithm must be performed in color coordinates which are linearly related to intensity. Therefore, the L*a*b* coordinates of the color palette may not be used in the error diffusion filter. To eliminate the need for computationally expensive transformations, we introduce a new linear

color coordinate system called $Y_y c_x c_z$ [14]. Because this new coordinate system is carefully chosen to align with the $L^*a^*b^*$ system, it preserves the structure of the SSQ palette. Hence, error diffusion and SSQ quantization may be performed in the same coordinate system, thereby eliminating the additional computation of a transformation to Lab.

Finally, we investigate a visually weighted quality metric to evaluate the performance of our halftoning methods. Commonly, the quality of a match between two color patches of sufficiently large size is assessed by computing the color difference in the CIE$L^*a^*b^*$ color space. However, spatial frequency response must also be incorporated to account for the reduced visual sensitivity to the high frequency quantization noise. To account for both of these effects, our metric is calculated by passing the original and halftoned images through a spatial filter approximating the human contrast sensitivity in luminance and chrominance. These spatial filters are applied in color coordinates which are linear in intensity. The filtered signals are then transformed to the visually uniform space L*a*b* and the error energy is computed in each component.

We apply our optimized palette and error diffusion algorithm to a variety of color test images. For comparison, we also test separable RGB palettes and conventional Floyd-Steinberg error diffusion[7]. Both subjective evaluation and our proposed quality metric indicate that the new method produces substantial and consistent improvement in image quality.

In Section 2 we describe the sequential color palette design. In Section 3 we combine our color palette with error diffusion using the new color space $Y_y c_x c_z$. Section 4 explains our visually weighted error metric, and Section 5 contains experimental results.

2 Optimal Color Palette Design

The objective of this section is to employ vector quantization (VQ) methods to design an optimized universal color palette. Conventional VQ works by selecting N code words (in our case colors) which minimize some distance to the expected input values. We will assume that the input color is specified in terms of the visually uniform color space, (L, a, b) (we

will suppress the asterisk superscripts for notational simplicity). This coordinate system was specified by CIE as

$$\begin{aligned} L &= 116f(Y/Y_n) - 16 \\ a &= 500[f(X/X_n) - f(Y/Y_n)] \\ b &= 200[f(Y/Y_n) - f(Z/Z_n)] \end{aligned} \tag{1}$$

where

$$f(x) = \begin{cases} x^{1/3} & \text{if } 0.008856 < x \leq 1 \\ 7.787x + (16/116) & \text{if } 0 \leq x \leq 0.008856 \end{cases} \tag{2}$$

X, Y, Z are the standard color coordinates for a 2° observer, and X_n, Y_n, Z_n specify the white point.

The *Lab* coordinate system was designed so that a just noticeable difference in color corresponds to an approximately constant euclidean distance. However, this approximate uniformity of *Lab* only holds at low spatial frequencies because the design was based on large color patches. At higher spatial frequencies, the relative sensitivity to the luminance or L component of color is much greater, and the *Lab* color space becomes nonuniform. This is particularly problematic when evaluating the quality of halftoning algorithms such as error diffusion because artifacts generally occur at high spatial frequencies. To compensate for this effect, Venable, Stinehour and Roetling applied a weighting factor $w > 1$ to the luminance component of the distortion. Thus, the square of the distance between the two colors $L_1a_1b_1$ and $L_2a_2b_2$ will be

$$D = w|L_1 - L_2|^2 + |a_1 - a_2|^2 + |b_1 - b_2|^2 \tag{3}$$

where w is an experimentally determined constant.

A major disadvantage of conventional unstructured VQ is its computational complexity. Moreover, quantization using an unstructured palette requires a full search through the N code book entries. Recently, tree structured VQ methods have been employed in color quantization applications [15, 16, 3, 17] to reduce computation. However, these methods still require $\log N$ operations per input color, which is excessive in many applications.

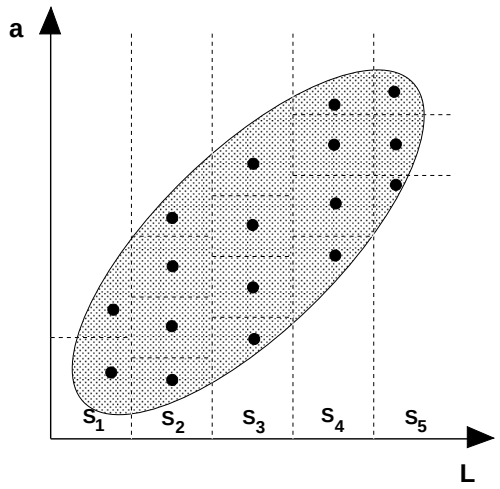

Figure 1: 2-D example for sequential scalar quantization (SSQ).The shaded area indicates a 2D device gamut. Bullets denote quantization levels, and dotted lines show cell boundaries. In this example, L is quantized first.

2.1 Overview of Sequential Scalar Quantization (SSQ)

Sequential scalar quantization (SSQ) is a VQ method which imposes structure on the color palette in order to minimize computation[11, 4, 12]. SSQ works by performing scalar quantization successively on each component of an input vector. Figure 1 illustrates the method for the two dimensional input vector (L, a). First, the scalar component L is quantized into $N_1 = 5$ regions denoted by the sets S_1 through S_5. These regions are formed by designing an optimal one dimensional quantizer for the marginal density of L

$$p_L(L) = \int_{a \in I\!R} \int_{b \in I\!R} p_{Lab}(L, a, b) da\, db \ .$$

Next, for each region $L \in S_i$, a different one dimensional quantizer is applied to the scalar a. Each of these quantizers is designed to be optimal for the conditional distribution of a given that $L \in S_i$

$$p_a(a|L \in S_i) = \int_{b \in I\!R} \int_{L \in S_i} p_{Lab}(L, a, b) dL\, db \ .$$

Since each conditional density is likely to be different, we would expect each quantizer to be different. In addition, each quantizer will differ due to the varying number of quantization levels n_i associated with each region S_i. For example, $n_4 = 4$ levels have been allocated to

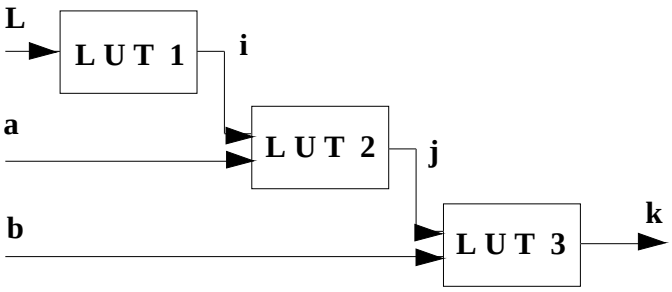

Figure 2: Block diagram of sequential look-up table (LUT) which performs SSQ quantization for the 3 dimensional (L, a, b) input vector. The order of quantization is L, then a, then b, and the output codeword is $Q[L, a, b]$.

region S_4; whereas, only $n_5 = 3$ levels have been allocated to region S_5. Notice, that the total number of levels allocated for both L and a is given by

$$N_2 = \sum_{i=1}^{N_1} n_i \ .$$

The extension of SSQ to three dimensions is straightforward. Each region of (L, a) formed by quantizing both components is denoted by $S_{i,j}$. For each set $S_{i,j}$, the third scalar component b is quantized to $n_{i,j}$ levels, and the total number of allocated colors is given by

$$N_3 = \sum_{i=1}^{N_2} \sum_{j=1}^{n_i} n_{i,j} \ .$$

Because the quantizers are different for each region S_i or $S_{i,j}$, the SSQ method exploits the dependencies among scalar components. For example, combinations of L and a which fall outside the gray area of Figure 1 are not in the device gamut and need not be quantized. We note that the order of quantization may vary, but the L, a, b order shown here will prove useful later.

The principal computational advantage of SSQ is that it may be implemented as a sequence of 1D look-up table (LUT) operations. Therefore, SSQ yields the performance benefits of VQ, but requires no more computation than conventional scalar quantization. Figure 2 illustrates the structure of this sequential LUT for the three dimensional input (L, a, b). The first LUT quantizes the L component and returns an index i corresponding to the quantization region $L \in S_i$. The second LUT then applies the appropriate scalar quantizer

based on this region S_i and returns the index (i, j) corresponding to the quantization region $(L, a) \in S_{i,j}$. The third LUT quantizes the component b based on $S_{i,j}$ and yields the quantized color vector, $Q[L, a, b] = (qL, qa, qb)$.

2.2 Color Palette Design with SSQ

In order to obtain $p_{Lab}(L, a, b)$, we will assume that image colors are uniformly distributed over the gamut of the monitor. Define the window function

$$w(L, a, b) = \begin{cases} 1 & \text{if } (L, a, b) \text{ is in gamut} \\ 0 & \text{if } (L, a, b) \text{ is out of gamut} \end{cases} .$$

Then $p_{Lab}(L, a, b)$ is given by

$$p(L, a, b) = \frac{w(L, a, b)}{\int_{L \in \mathbb{R}} \int_{a \in \mathbb{R}} \int_{b \in \mathbb{R}} w(L, a, b) dL\, da\, db} .$$

For the time being, we will also assume that N_1, N_2, and N_3 are known in advance. Generally, the total number of desired quantization levels, N_3, is specified, and we will present a method for estimating N_1 and N_2 in section 2.2.4.

Our method for designing the SSQ color palette is based on asymptotic quantization theory [18], and will closely parallel the method used for color quantization in [11, 4, 12]. One unique aspect of our problem is that we would like to preserve the maximum gamut of the display device. Since halftoning algorithms can produce the average of several palette colors, the effective gamut will be the convex hull spanned by the color palette. Moreover, error diffusion assumes that the input signal is contained in the convex hull of the available quantization levels and will otherwise pass accumulating errors forward to unquantized pixels, thus driving the quantizer into saturation. Therefore, our color palette design method will include heuristics to insure that colors extend to the boundaries of the device's gamut.

2.2.1 Luminance Quantizer

First, we design the quantizer for L. In order to preserve the maximum gamut, we temporarily fix the first and last quantization levels to be

$$\begin{aligned} \tilde{q}_1 &= \min_L\{L : p_L(L) \neq 0\} \\ \tilde{q}_{N_1} &= \max_L\{L : p_L(L) \neq 0\} \ . \end{aligned}$$

Since the computational cost of the quantizer design is not significant, we will use the Lloyd-Max algorithm to choose the quantization regions $S_1, \cdots, S_{N_1}$ and the quantization levels $q_2, \cdots, q_{N_1-1}$. In practice, we found the performance of the Lloyd-Max algorithm to be strongly dependent on the initial condition. Therefore, we initialize the Lloyd-Max algorithm by choosing the intermediate quantization levels according to the asymptotically optimal point density function, $\lambda(L)$, given by

$$\lambda(L) = \frac{\{p_L(L)\}^{1/3}}{\int_{L\in\mathbb{R}}\{p_L(x)\}^{1/3}dx} \ .$$

More specifically, we choose the initial values for $q_2, \cdots, q_{N_1-1}$ so that

$$\frac{i-1}{N_1-1} = \int_{-\infty}^{q_i} \lambda(L)dL \ .$$

Figure 3 illustrates a final step in which the quantization levels $\tilde{q}_1$ and $\tilde{q}_{N_1}$ are replaced with the centroids of their respective quantization regions.

$$\begin{aligned} q_1 &= \int_{L\in S_1} p_L(L)dL \\ q_{N_1} &= \int_{L\in S_{N_1}} p_L(L)dL \ . \end{aligned}$$

These values of $q_1, \cdots, q_{N_1}$ are then used as output values for L.

2.2.2 Chrominance Quantizer

The next step is to design the quantizer for the scalar a. We first calculate the conditional probability density of a given that $L \in S_i$,

$$p_a(a \mid L \in S_i) = \int_{b\in\mathbb{R}}\int_{L\in S_i} p_{Lab}(L,a,b)dL\,db \ .$$

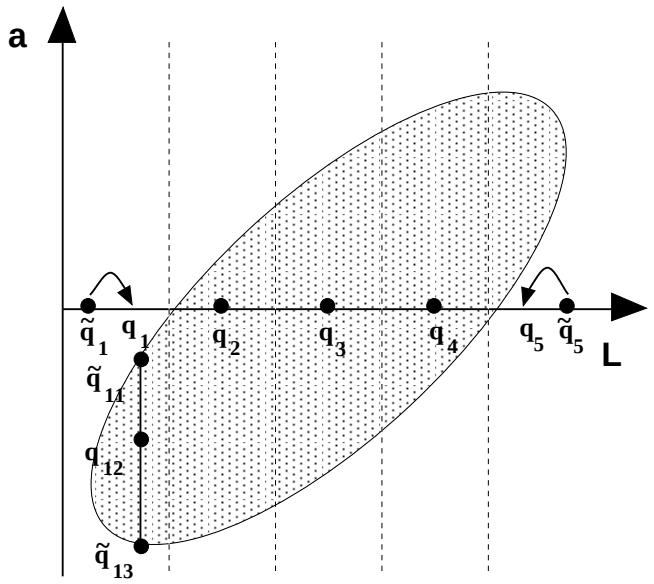

Figure 3: Quantization procedure for L and a components. Notice that $\tilde{q}_1$ and $\tilde{q}_5$ are initially placed at the most extreme locations, and are then moved to the centroids of their cells denoted by q_1 and q_5. Quantization levels $q_{1,j}$ are lined up along q_i.

Before we can design an optimal quantizer for each region S_i, we must determine the optimal number of quantization levels n_i. The optimal value for n_i can be derived using asymptotic quantization theory [11], and is given by

$$n_i = \text{int}\left(N_2 \frac{r_i}{\sum_{l=0}^{N_1-1} r_l} \right),$$

where

$$r_i = \{P(L \in S_i)\}^{1/3} \int_{a \in I\!R} \{p_a(a \mid L \in S_i)\}^{1/3} \, da,$$

$P(L \in S_i)$ is the probability that L is quantized to the region S_i, and $\text{int}(\cdot)$ denotes rounding to the nearest integer.

For each region, S_i, and the bit allocation, n_i, the quantizer for a is designed as before. The Lloyd-Max algorithm is applied after fixing the first and last quantization levels to be

$$\begin{aligned} \tilde{q}_{i,1} &= \min_a \{a : p_{La}(L = q_i, a) \neq 0\} \\ \tilde{q}_{i,n_i} &= \max_a \{a : p_{La}(L = q_i, a) \neq 0\}, \end{aligned}$$

and the Lloyd-Max algorithm is initialized using

$$\frac{j-1}{n_i-1} = \int_{-\infty}^{q_{i,j}} \lambda_i(a)da$$

where

$$\lambda_i(a) = \frac{\{p_a(a \mid L \in S_i)\}^{1/3}}{\int_{x\in\mathbb{R}} \{p_a(x \mid L \in S_i)\}^{1/3} dx} \ .$$

Once again, as a final step the first and last quantization levels are relocated to the centroids of their cells according to

$$\begin{aligned} q_{i,1} &= \int_{L\in S_{i,1}} p_a(a \mid L \in S_i)da \\ q_{i,N_1} &= \int_{L\in S_{i,n_i}} p_a(a \mid L \in S_i)da \ . \end{aligned}$$

The quantizer design for b is similar to that used for a. For each region $S_{i,j}$ in the (L,a) plane, we design a 1D quantizer for the b component. Using the conditional density, $p_b(b \mid (L,a) \in S_{i,j})$. we may compute the number quantization levels for region $S_{i,j}$ as

$$n_{i,j} = \text{int}\left(N_3 \frac{r_{i,j}}{\sum_{k=1}^{N_1}\sum_{l=1}^{n_k} r_{k,l}} \right),$$

where

$$r_{i,j} = \{P((L,a) \in S_{i,j})\}^{1/3} \int_{b\in\mathbb{R}} \{p_b(b \mid (L,a) \in S_{i,j})\}^{1/3} db,$$

and $P((L,a) \in S_{i,j})$ is the probability that (L,a) is quantized to the region $S_{i,j}$.

For each region $S_{i,j}$, we again set the first and last quantization levels to their most extreme values.

$$\begin{aligned} q_{i,j,1} &= \min_b\{b : p_{Lab}(L = q_i, a = q_{i,j}, b) \neq 0\} \\ q_{i,j,n_{i,j}} &= \max_b\{b : p_{Lab}(L = q_i, a = q_{i,j}, b) \neq 0\} \ . \end{aligned}$$

The remaining quantization levels, $q_{i,j,2}, \cdots, q_{i,j,n_{i,j}-1}$, are again allocated using the Lloyd-Max algorithm and an initial condition based on the optimal point density function

$$\lambda_{i,j}(b) = \frac{\{p_b(b \mid (L,a) \in S_{i,j})\}^{1/3}}{\int_{x\in\mathbb{R}} \{p_b(b \mid (L,a) \in S_{i,j})\}^{1/3} dx} \ .$$

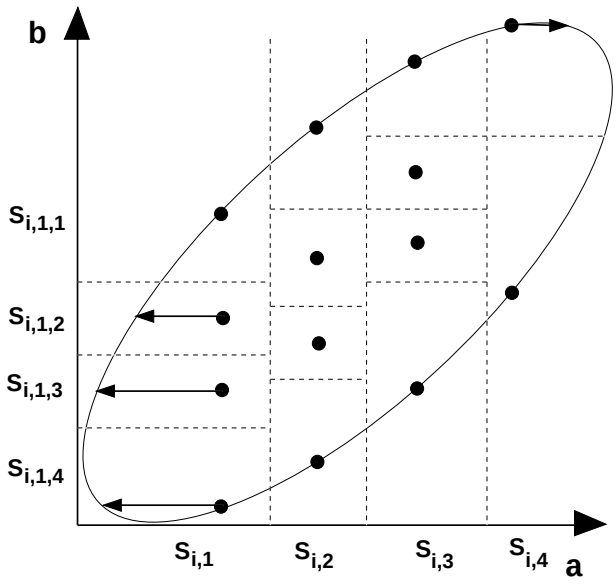

Figure 4: Expanding the palette range by moving the outer a-quantization levels to the edges of the display gamut.

This time we will not replace the first and last quantization levels. Instead, we will leave them at the boundary of the color space in order to preserve the maximum gamut.

At this point, we make some observations about the palette. If the gamut of the device is convex, then the palette colors will fall inside the gamut. However, the quantization levels corresponding to the minimum and maximum values of a will generally not be on the boundary of the gamut. In the next section, we will describe how these color can be moved slightly to expand the usable device gamut. Note that because the values of n_i and $n_{i,j}$ must be rounded to the nearest integer, the total number of allocated colors may not always be equal to the desired number of colors. The desired number of colors may be obtained by adding or subtracting single quantization levels according to an MSE criterion as described in [11]. However, for our application, we will simply accept a slightly smaller palette.

2.2.3 Expanding the Color Map Range

In order to maximize the displayable gamut formed by the convex hull of palette colors we will move some of the colors near the minimum and maximum values of a. Figure 4 illustrates how colors corresponding to the regions $S_{i,1}$ and S_{i,n_i} can be moved to expand the gamut.

More formally, for $1 < j < n_i$ we define the final (L, a, b) palette colors to be

$$C_{i,j,k} = (q_i, q_{i,j}, q_{i,j,k})$$

and for $j = 1$ or n_i we define

$$\begin{aligned} C_{i,1,k} &= (q_i, \min_{i,k}, q_{i,1,k}) \\ C_{i,n_i,k} &= (q_i, \max_{i,k}, q_{i,n_i,k}) \end{aligned}$$

where

$$\begin{aligned} \min_{i,k} &= \min_a \{a : p_{Lab}(L = q_i, a, b = q_{i,1,k}) > 0\} \\ \max_{i,k} &= \max_a \{a : p_{Lab}(L = q_i, a, b = q_{i,n_i,k}) > 0\} \ . \end{aligned}$$

Finally, we expand the color map to the maximum range along the L-axis, by adding one quantization level for perfect black, $(0, 0, 0)$, and another for perfect white, $(100, 0, 0)$.

As mentioned above, we note that in each cell $S_{i,j}$ the algorithm already placed the first and last quantization level of b on the boundaries of the display gamut.

2.2.4 Optimal Bit Allocation

A method must still be given for selecting the optimal values of N_1, and N_2. If N_1, and N_2, and N_3 are large numbers, then asymptotic quantization theory may be applied. Specifically, if D_L, D_a, and D_b are the mean squared error in the L, a, b components respectively, then the total distortion is given by $D = D_L + D_a + D_b$. Furthermore, each component distortion is approximately equal to

$$D_L \approx \tfrac{1}{N_1^2}\alpha, \quad D_a \approx \tfrac{N_1^2}{N_2^2}\beta, \quad D_b \approx \tfrac{N_2^2}{N_3^2}\gamma, \tag{4}$$

where α, β and γ are constants which depend on the probability density of the input signal[11, 4, 12]. If α, β and γ are known, then the optimal values for N_1 and N_2 are given by

$$N_1 = N_3^{1/3}(\frac{(w\alpha)^2}{\beta\gamma})^{1/6}$$

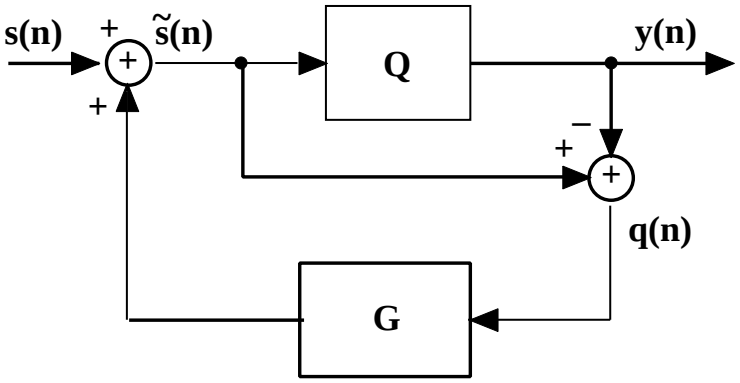

Figure 5: Block diagram of the error diffusion algorithm.

and

$$N_2 = N_3^{2/3}(\frac{w\alpha\beta}{\gamma^2})^{1/6}.$$

In practice, N_1, N_2, and N_3 will not be very large so the analytical formulas for α, β and γ will not be accurate. However, experiments have shown that, if the constants α, β and γ are properly adjusted, then (4) holds for a range of relatively small N_k. Adjusted values of these constants may be obtained by performing an initial quantization with arbitrary $\tilde{N}_k$, and measuring the values of D_L, D_a and D_b. The estimates for α, β and γ may then be obtained by equating terms in (4).

3 Application to Error Diffusion

In this section, we incorporate the optimized color palette into the error diffusion halftoning method. A naive implementation would require a transformation both to and from the *Lab* coordinate system used in the palette design. However, we will show that by careful selection of the coordinates used for the error diffusion algorithm, the SSQ quantizer may be implemented directly without any transformations.

Figure 5 illustrates the basic error diffusion algorithm as described by Floyd and Steinberg in [7]. In order to better understand how error diffusion works, we will review the frequency analysis of [3]. The error diffusion algorithm works by feeding back quantization error to reduce the low frequency component of the displayed error. Let $s(n)$ be an image indexed by

the two dimensional pixel locations $n = (n_1, n_2)$. Then, the equations which describe error diffusion are given by

$$\begin{aligned} y(n) &= Q[\tilde{s}(n)] \\ q(n) &= \tilde{s}(n) - y(n) \\ \tilde{s}(n) &= s(n) + g(n) * q(n) \end{aligned}$$

where $y(n)$ is the displayed image, $q(n)$ is the quantization error, and $*$ denotes two dimensional convolution. Substituting the third equation into the second yields the relationship for the display error $e(n)$

$$\begin{aligned} e(n) &= s(n) - y(n) \\ &= q(n) - g(n) * q(n) \ . \end{aligned}$$

Since this is a linear relationship, we may take the frequency transform to yield

$$E(\omega) = [1 - G(\omega)]\, Q(\omega) \ ,$$

where $E(\omega)$, $G(\omega)$, and $Q(\omega)$ denote the discrete space Fourier transforms of $e(n)$, $g(n)$, and $q(n)$ respectively. Thus, the display error spectrum can be shaped by selecting the proper error diffusion filter $G(\omega)$. Generally, $G(\omega)$ is chosen to be low pass so that the transfer function $1 - G(\omega)$ is high pass. This suppresses the low frequency error for which the visual system is most sensitive.

We will choose the error diffusion filter $G(\omega)$ to minimize the perceived error as described in [13]. This optimized error diffusion approach is appropriate for multilevel halftoning, and selects $G(\omega)$ based on an overall system model. This system model includes the monitor and the human visual system response, and incorporates the contrast sensitivity to both luminance and chrominance, and the reduced visual sensitivity to diagonal frequencies.

One difficulty with incorporating the optimized palette into error diffusion is the proper choice of color coordinates. In order for the perceived color of the original and halftone

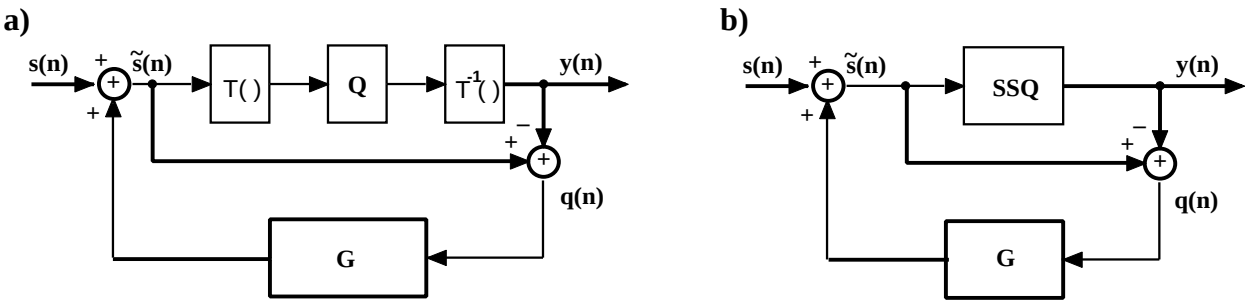

Figure 6: Block diagram of error diffusion with SSQ: a) matching color spaces with transformations T and T^{-1}, b) eliminating transformations in new Y, c_x, c_z color space.

images to match, the data should be processed in the same coordinate system used by the human visual system. Experiments show that spatial frequency response of the visual system is due to the combined effects of optical blur and the limited resolving power of the retina-brain system[19]. Recent studies have found that the fall off in contrast sensitivity at high spatial frequencies is mainly due to optical properties of the eye[20]. Since optical blurring effects are due to incoherent averaging of energy, these effects are properly modeled by filtering in a color coordinate system which is linear in intensity. In our application, this low pass behavior is the dominant effect; so we will adopt a linear color coordinate system when modeling the low pass nature of the human visual system. [1] This choice precludes the direct use of the *Lab* coordinates in the error diffusion algorithm since they are nonlinearly related to intensity.

Figure 6a shows a naive solution to this problem of mismatched coordinate systems. The input data $s(n)$ is processed by the error diffusion filter in a linear coordinate system. However, since the quantizer is designed in *Lab*, $T(\cdot)$ transforms the colors to *Lab* before quantization, and $T^{-1}(\cdot)$ converts back after quantization. We would like to eliminate these transformations because they are computationally expensive. The inverse transformation, $T^{-1}(\cdot)$, may be eliminated by precomputing the transformation for the palette of output colors.

[1] It is interesting to note that conventional color measurement instruments average the energy of the reflected signal over a color patch; and therefore, effectively operate in a linear color coordinate system.

We will eliminate the forward transformation, $T(\cdot)$, by judiciously selecting the linear coordinate system for error diffusion. Define the coordinate system which we call Y_y, c_x, c_z [14]

$$\begin{aligned} Y_y &= 116\frac{Y}{Y_n}, \\ c_x &= 500\Big[\frac{X}{X_n} - \frac{Y}{Y_n}\Big], \\ c_z &= 200\Big[\frac{Y}{Y_n} - \frac{Z}{Z_n}\Big], \end{aligned}$$

where as before (X_n, Y_n, Z_n) specifies the white point. Notice that the $Y_y c_x c_z$ coordinates are chosen so that they are aligned with the L, a, b system. In fact, one may easily verify that

$$\bigtriangledown_{(Y_y, c_x, c_z)}(L, a, b)\Big|_{white\ point} = f'(Y_y)\mathbf{I},$$

where $f'(Y_y)$ is the derivative of $f(\cdot)$ evaluated at Y_y and $\mathbf{I}$ is the identity matrix. Furthermore, Y_y, c_x, c_z is approximately an opponent color system, where variations in c_x correspond to changes along a red-green direction and variations in c_z reflect changes along a blue-yellow axis.

We next show that the SSQ structure of the optimized palette is approximately preserved in the coordinate system Y_y, c_x, c_z. This implies that a modified SSQ quantizer may be directly applied to the components of $\tilde{s}(n)$ in the error diffusion filter. Since the function f$(\cdot)$ in (2) is a monotone increasing function, quantization of $L = 116\text{f}(Y/Y_n) - 16$ may be replaced by equivalent quantization of Y_y,

$$Q_L[L] \equiv Q_Y[Y_y].$$

The component a is dependent on the value of Y_y and c_x, however the value of Y_y may be approximated by the quantized value $Q_Y[Y_y]$. After making this replacement, the quantization of a is equivalent to quantization of c_x. More specifically, define the function $a = g(c_x, Y_y)$. Then

$$Q_a[a] = Q_a[g(c_x, Y_y)]$$

$$\begin{aligned} &\approx Q_a[g(c_x, Q_Y[Y_y])] \\ &= Q_{cx}[c_x, Q_Y[Y_y]] \ . \end{aligned}$$

Therefore, for each quantized value $Q_Y[Y_y]$, the quantizer for a may be approximately replaced by a quantizer for c_x . We note that this approximation becomes more accurate as N becomes large.

The third component is similar. Define the function $b = h(c_z, Y_y)$. Then

$$\begin{aligned} Q_b[b] &= Q_b[h(c_z, Y_y)] \\ &\approx Q_b[h(c_z, Q_Y[Y_y])] \\ &= Q_{cz}[c_z, Q_Y[Y_y]] \ . \end{aligned}$$

This implies that the quantizer for b may be replaced by a quantizer for c_z for each quantized value $Q_Y[Y_y]$.

Figure 6b shows a block diagram of the resulting error diffusion algorithm, where we assume that the input image, $\mathbf{s}(\mathbf{n})$, is already transformed to Y_y, c_x, c_z coordinates. Notice that the transformations before and after the quantizer are eliminated and that the block diagram resembles basic error diffusion as in Figure 2, where the simple RGB quantizer is replaced by a sequential LUT similar to Figure 2 but with Y_y, c_x, c_z inputs.

4 Visually Weighted Error Metric

We would like to evaluate the quality of a halftoned image by some reproducible, perceptually relevant criterion. For large color patches, it is common to calculate the mean squared error,

$$\Delta E = \Delta L^2 + \Delta a^2 + \Delta b^2,$$

where $\Delta L, \Delta a$ and Δb are the differences between the original and the reproduced color patch in the visually uniform color space *Lab*. As mentioned above, error diffusion and other halftoning techniques introduce mostly high frequency noise to the quantized image. Therefore, any useful error metric must incorporate a model for human contrast sensitivity

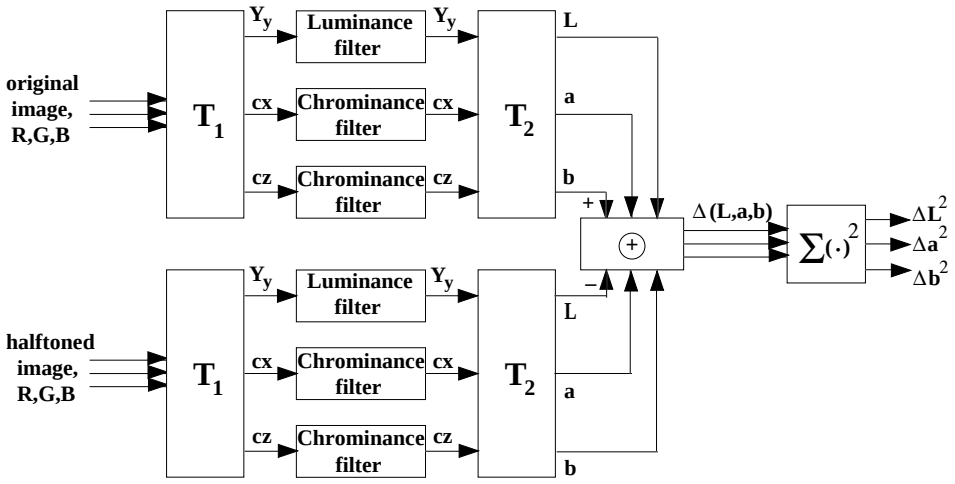

Figure 7: Block diagram of the visually weighted error metric. T_1, T_2 denote transformations from R,G,B to Y_y, c_x, c_z and from Y_y, c_x, c_z to L, a, b.

as a function of spatial frequency. Mitsa and Varkur have studied a variety of quality metrics for monochrome halftone images[21], and have found that the best correlation with subjective tests is achieved when the frequency response is chosen to be low pass instead of band pass. We take a similar approach, but also incorporate the *Lab* color metric to account for nonlinear visual effects.

Figure 7 shows a block diagram of our error metric. First, the halftoned and the original image are transformed from RGB to Y_y, c_x, c_z coordinates. Both images are then passed through a set of low pass modulation transfer functions, which model the contrast sensitivity of the human observer toward luminance and chrominance. The outputs of these filters are converted to *Lab* coordinates and the difference signals between the halftone and original are calculated. Finally, the energy in each component of the difference signal is determined. Before combining the three values of $\Delta L^2, \Delta a^2$ and Δb^2 to a ΔE related quantity, we examine the performance of our algorithm in the three components L, a, b separately.

Notice that the linear filtering operation is performed before the transformation to the *Lab* coordinates. This order of operations is critical, since the perceived color will be the average of the halftoned colors in a linear coordinate system. If the nonlinear transformation were performed first, the model would inaccurately predict the perceived color of the halftoned

image.

The spatial filters used in luminance and chrominance, are based on a visual model described by Sullivan et al. [22, 23] and data obtained by Mullen [24]. Both, the luminance and the chrominance model are of the form

$$W(\tilde{f}) = \begin{cases} \exp\left\{-\alpha(\tilde{f} - f_c)\right\} & f \geq f_c \\ 1 & f < f_c. \end{cases}$$

where the decay rates α and the cutoffs, f_c, are estimated from Mullen's data. For the luminance model we determined $\alpha = 0.1761$deg/cycle and $f_c = 2.2610$cycles/deg, and for the chrominance model we obtained $\alpha = 0.4385$deg/cycle and $f_c = 0.2048$cycles/deg.[2] Furthermore, $\tilde{f}$ is the weighted magnitude of the frequency vector $f = (f_1, f_2)$, where the weighting has an angular dependence as applied by Sullivan [22],

$$\tilde{f} = \frac{\sqrt{f_1^2 + f_2^2}}{s(\Theta)},$$

where

$$s(\Theta) = 0.15\cos(4\Theta) + 0.85 \ ,$$

and Θ is defined as

$$\Theta = \arctan\left(\frac{f_1}{f_2}\right) \ .$$

Thus, the model is also a function of the viewing angle and decreases faster for diagonal frequencies to account for reduced sensitivity to luminance changes in diagonal directions.

5 Experimental Results

The marginal distributions are obtained by numerical integration of $p_{Lab}(L, a, b)$ with respect to a and b. Figure 8 shows a histogram of this color distribution in the (L, a) plane for a display with SMPTE RGB primaries and a D65 white point. Bright areas correspond to high probability, while the dark regions along the boundaries of the gamut

[2]The original publication contained an error. It transposed the values of 0.1761 and 0.4385. The error has been corrected in this manuscript.

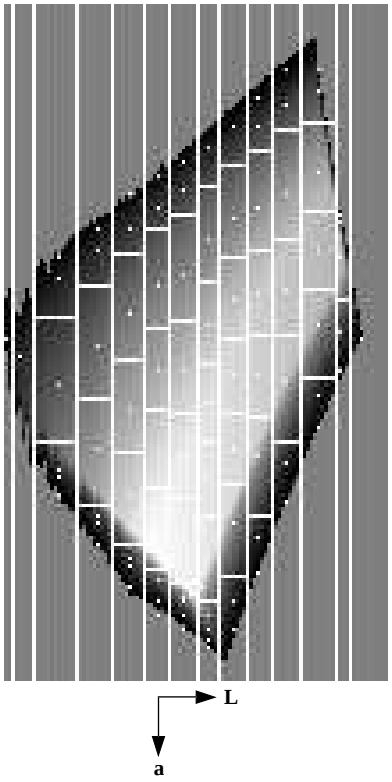

Figure 8: Histogram of the marginal color distribution in the (L, a) plane, where L increases from left to right and a increases from top to bottom. Bright areas indicate high probability, dark regions reflect low probability.

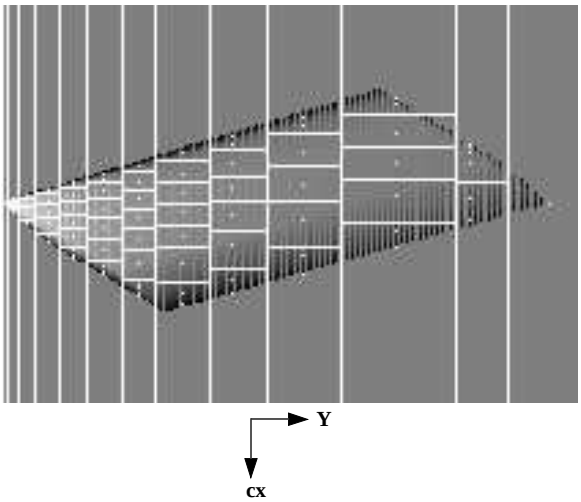

Figure 9: Histogram of the marginal color distribution in the (Y_y, c_x) plane, where Y_y increases from left to right and c_x increases from top to bottom. The Y_y-axis is magnified by a factor of eight.

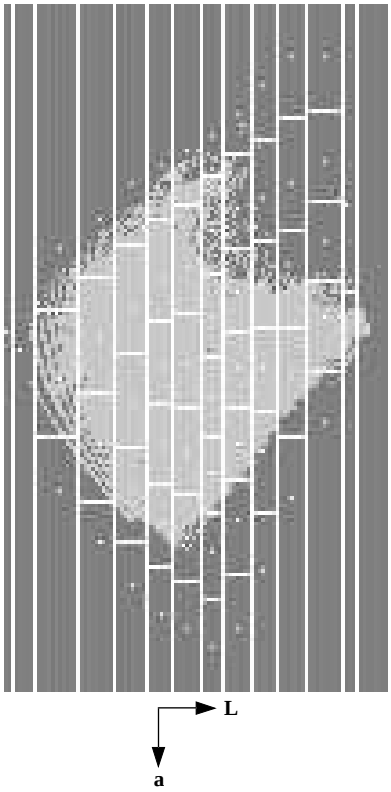

Figure 10: Range of color distribution in the L, a-plane, for image "Picnic".

reflect low probability. The white dots in Figure 8 indicate the positions of the quantization levels projected into the (L, a) plane, and the white lines show the boundaries between quantization cells. Notice that several quantization levels are shown for most of the boundary quantization cells. This shows the effect of expanding the range of the color map as described in Section 2.2.3. Using a weighting factor of $w = 8.0$ for the error metric, we assign 12 quantization levels to luminance, 57 levels to the chrominance component a and 245 colors to the entire palette. Adding perfect black and perfect white to the palette increases the palette size to 247 colors.

Figure 9 illustrates the transformation of the color map and the histogram to the new Y_y, c_x, c_z coordinate system, where the Y_y, c_x plane is shown. Notice that the sequential structure of the color palette is maintained after the transformation.

Figure 10 shows the range of colors in the (L, a) plane, which occur in the image "Picnic". Light gray areas indicate colors that are present in the image. The white lines and dots

again indicate the quantization cell boundaries and the palette colors. For each quantization region, we only show the one palette color which is closest to the gamut boundary. Notice how the colors of this image extend almost to the edges of the projected gamut. Although the distribution of colors in any one image is rarely uniform, the figure shows the variety of colors that may occur in one image. This indicates the value of minimizing the loss in gamut when designing the color map.

We will compare three algorithms: Floyd-Steinberg with a separable palette, Floyd-Steinberg with an optimized palette, and optimized error diffusion with an optimized palette. We will refer to this last combination as the optimized algorithm. Since a separable RGB color palette with linear spacings between quantization levels yields poor results, we used a separable RGB palette with colors allocated according to a power law with an exponent of 3.0. A similar approach was taken by Goertzel and Thompson in [9]. We assigned 8 levels to each of the red and green components, and 4 levels to the blue component for a total of 256 colors.

Tab.1 shows a measure of perceived error for a set of five test images shown in Figure 11. The error was measured using the procedure described in section 4 and assumed a normal viewing distance of 45.5cm (17.9in) and a display resolution of 100 dots/inch. We examined a variety of images, including images with very saturated colors such as "Picnic", images with more pastel colors like "Beach" and "Balloon", and test images like "Uchart". Except for the b-component in the image "Picnic", the optimized algorithm achieved consistently lower errors in all three components. Column 2 in Tab.1 shows the improvement that can be achieved by using the optimized color palette while retaining the Floyd-Steinberg filter coefficients. However, the best overall performance was achieved when combining the optimized color palette with optimized filter coefficients. These numerical results corresponded well with our subjective evaluations of image quality. Table 2 lists values of ΔE for the three algorithms which are obtained by adding the three quantities of $\Delta L^2, \Delta a^2, \Delta b^2$ for each image. In this case the optimized error diffusion together with an optimized color palette

Image:		Floyd-Steinberg RGB Palette	Floyd-Steinberg Opt. Palette	Opt. ED Opt. Palette
Balloon	ΔL^2	0.3176	0.1239	0.0872
	Δa^2	0.1203	0.0772	0.0510
	Δb^2	0.7894	0.3310	0.2940
Beach	ΔL^2	0.3584	0.1191	0.0740
	Δa^2	0.1064	0.0238	0.0116
	Δb^2	0.3889	0.0524	0.0357
Munich	ΔL^2	0.3190	0.1356	0.0968
	Δa^2	0.0662	0.0428	0.0191
	Δb^2	0.1092	0.0300	0.0168
Picnic	ΔL^2	0.4016	0.2680	0.1989
	Δa^2	0.1675	0.1726	0.1477
	Δb^2	0.4257	0.4031	0.4514
Uchart	ΔL^2	0.3298	0.2037	0.1535
	Δa^2	0.1247	0.0751	0.0355
	Δb^2	0.6661	0.3668	0.2614

Table 1: Estimates of perceived error of halftone images in L, a, b. The optimized error diffusion algorithm with an optimized universal palette produced the lowest errors in almost every case.

consistently produced the lowest errors.

Figure 12 shows the original "Balloon" image, the Floyd-Steinberg halftone with RGB palette, Floyd-Steinberg algorithm with optimized palette, and optimized error diffusion with optimized palette. Given the 2:1 zoom of the figures and the assumed viewing distance and resolution, these images are designed to be viewed at 7.0 times their height. The range of colors in this image and the numerical results for our error metric seem to be typical for the majority of images we worked with. Notice that the optimized algorithm breaks up contours and patterns which are typical artifacts of the Floyd-Steinberg algorithm. These effects can be most clearly seen in the slowly varying colors of the different balloons. Notice for example, how contouring effects in the (yellow) balloon on the left of the girl's face are broken up, when using the fully optimized algorithm.

Figure 13 shows the original "Picnic" image, the Floyd-Steinberg halftone with RGB palette, Floyd-Steinberg algorithm with optimized palette, and optimized error diffusion

Image:	Floyd-Steinberg RGB Palette ΔE^2	Floyd-Steinberg Opt. Palette ΔE^2	Opt. ED Opt. Palette ΔE^2
Balloon	1.2273	0.5321	0.4322
Beach	0.8537	0.1953	0.1213
Munich	0.4944	0.2084	0.1327
Picnic	0.9948	0.8437	0.7980
Uchart	1.1206	0.6456	0.4504

Table 2: Estimates of perceived error of halftone images in ΔE^2. Here the optimized error diffusion algorithm with an optimized universal palette yielded consistently the lowest errors.

with optimized palette. Similar effects may be seen in this image. For instance, examining the texture in the tree on the right side of the image, it can be seen that the optimized algorithm renders details with more accuracy than the old method. However, the improvement in this image was the least noticeable and the numerical results represent our worst case performance.

In a few image areas, for instance in the blue sky area of the "Picnic" image, the optimized algorithm seems to generate slightly more visible texture. We think that this effect is our trade off for achieving an overall performance improvement. The predominantly diagonal orientation of the texture may be due to the angle dependency of the human visual model, which was used to design our optimized error diffusion filter.

6 Conclusion

We examined the design of a universal color palette in a visually uniform color space. The palette is generated using a new vector quantization method known as sequential scalar quantization (SSQ). The SSQ approach reduces computation, by allowing vector quantization to be performed with a series of scalar quantizers. The resulting color palette was then combined with a previously developed multilevel error diffusion algorithm to give the best overall result.

To evaluate the quality of the displayed images, we developed a visually weighted error

a	b	c
d	e	

Figure 11: Test images: a)Balloon, b)Picnic, c)Uchart, d)Munich, e)Beach.

metric which uses models for human contrast sensitivity toward luminance and chrominance together with a nonlinear transformation of the color space. The measurement of this error metric for a variety of images substantiated our subjective conclusions that the optimized palette improved the displayed image quality.

Acknowledgement

We would like to thank the NEC corporation for their support of this work. Furthermore, we wish to thank Kodak and Reiner Eschbach form Xerox corporation for providing some of our test images.

References

[1] P. Heckbert, "Color image quantization for frame buffer display," *Computer Graphics,* vol. 16, no. 3, pp. 297-307, 1982.

[2] G. Braudaway, "A procedure for optimum choice of a small number of colors from a large color palette for color imaging," IBM internal report, RC 11367(#51227), Sept. 16, 1985.

[3] M. T. Orchard, C. A. Bouman, "Color Quantization of Images," *IEEE Trans. Signal Processing,* vol. 39, no.12, pp. 2677-2690, 1991.

[4] R. Balasubramanian, C. A. Bouman, and J. P. Allebach, "Sequential scalar quantization of color images", *IS&T's 46th Annual Conference*, Boston, MA, pp. 97-101, May 9-14, 1993.

[5] V. Iverson and E. Riskin, "A fast method for combining palettes of color quantized images," *Proc. of IEEE Int'l Conf. on Acoustic Speech an Signal Proc.*, Minneapolis, Minnesota, vol. 5, pp. 317-320, April 27-30, 1993.

[6] R. S. Gentile, E. Walowit, and J. P. Allebach, "Quantization and multilevel Halftoning of Color Images for Near Original Image Quality", *J. Opt. Soc. Am. A.,* vol.7, no.6, pp.1019-1026, 1990.

[7] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial greyscale," *Proc. SID,* vol. 17, no. 2, pp. 75-77, 1976.

[8] R. A. Ulichney, "Dithering with blue noise," *Proc. IEEE,* vol. 76, no. 1, pp. 56-79, 1988.

[9] G. Goertzel and G. R. Thompson, " 'Halftoning' techniques for displaying images with a limited color palette," *EI West 1990,* Pasadena, CA, pp. 102-108, 1990.

[10] D. Venable, J. Stinehour, and P. Roetling, "Selection and use of small color sets for pictorial display", *SPSE 43rd Annual Meeting,* Rochester, NY, pp. 90-92, May 20-25, 1990.

[11] R. Balasubramanian, C. A. Bouman, and J. P. Allebach, "Sequential scalar quantization of vectors: an analysis", submitted to the *IEEE Trans. on Image Processing.*

[12] R. Balasubramanian, C. A. Bouman, and J. P. Allebach, "Sequential scalar quantization of color images", *Journal of Electronic Imaging*, pp. 45-59, vol. 3, no. 1, 1994.

[13] B. W. Kolpatzik, C. A. Bouman, "Optimized error diffusion for image display," *J. Electronic Imaging,* vol. 1, no. 3, pp. 277-292, 1992

[14] T. J. Flohr, B. W. Kolpatzik, R. Balasubramanian, D. A. Carrara, C. A. Bouman, and J. P. Allebach, "Model based color image quantization," *Proc. of SPIE/IS&T Conf. on Human Vision, Visual Processing, and Digital Display*, pp. 270-281, San Jose, CA, January 31-February 4, 1993, Vol. 1993.

[15] J. Makhoul, S. Roucos, and H. Gish, "Vector quantization in speech coding", *Proc. of the IEEE*, vol. 73, no. 11, pp. 1551-1588, 1985.

[16] W. H. Equitz, "A new vector quantization clustering algorithm", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 10, pp. 1568-1575, 1989.

[17] R. Balasubramanian and J. P. Allebach, "A new approach to palette selection for color images," *J. Imaging Technol.,* vol. 17, no. 6, pp. 284-290, 1991.

[18] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression,* Kluwer Academic Publishers, 1992.

[19] F. W. Campbell, "The Human Eye as an Optical Filter," *Proceedings of the IEEE,* vol. 56, no. 6, pp. 1009-1014, 1968.

[20] "Relative Contribution of Optical and Neural Limitations to Human Contrast Sensitivity at Different Luminance Levels," *Vision Research,* vol. 33, no. 16, pp. 2321-2336, 1993.

[21] T. Mitsa and K. Varkur, "Evaluation of contrast sensitivity functions for the formulation of quality measures incorporated in halftoning algorithms," *Proc. of IEEE Int'l Conf. on Acoustic Speech an Signal Proc.*, Minneapolis, Minnesota, vol. 5, pp. 301-304, April 27-30, 1993.

[22] J. Sullivan, L. Ray, and R. Miller, "Design of minimum visual modulation halftone patterns," *IEEE Trans. Syst., Man, Cybernetics,* vol.21, no.1, pp. 33-38, 1991.

[23] J. Sullivan, R. Miller, G. Pios, "Image halftoning using visual error diffusion," *J. Opt. Soc. Am. A.*, vol.10, no.8, pp. 1714-1724, 1993.

[24] K. T. Mullen, "The contrast sensitivity of human color vision to red-green and blue-yellow Chromatic Gratings," *J. Physiol.* 359, pp. 381-400, 1985.

a	b
c	d

Figure 12: Comparison of 2:1 zoom of a) original "Balloon" image, b) Floyd-Steinberg algorithm with separable RGB palette, c) Floyd-Steinberg algorithm with optimized palette, d) optimized error diffusion with optimized palette.

a	b
c	d

Figure 13: Comparison of 2:1 zoom of a) original "Picnic" image, b) Floyd-Steinberg algorithm with separable RGB palette, c) Floyd-Steinberg algorithm with optimized palette, d) optimized error diffusion with optimized palette.