Accepted to CR&A (2002)

Garrett M. Johnson
Mark D. Fairchild

Munsell Color Science Laboratory
Chester F. Carlson Center for Imaging Science
Rochester Institute of Technology
Rochester, NY 14623
garrett@cis.rit.edu

A Top Down Description of S-CIELAB and CIEDE2000

Abstract

Recent work in color difference has led to the recommendation of CIEDE2000 for use as an industrial color difference equation. While CIEDE2000 was designed for predicting the visual difference for large isolated patches, it is often desired to determine the perceived difference of color images. The CIE TC8-02 has been formed to examine these differences. This paper presents an overview of spatial filtering combined with CIEDE2000, to assist TC8-02 in the evaluation and implementation of an image color difference metric. Based on the S-CIELAB spatial extension, the objective is to provide a single reference for researchers desiring to utilize this technique. A general overview of how S-CIELAB functions, as well as a comparison between spatial domain and frequency domain filtering is provided. A reference comparison between three CIE recommended color difference formulae is also provided.

Introduction

Industrial color difference equations have been the topic of tremendous research effort over the past 30 years. This effort has recently culminated with the recommendation by CIE TC1-47 of the CIEDE2000 color difference equation. This equation was derived from several data sets, though all the sets shared some commonality. Each data set was comprised of visual judgments of simple patches.[1] Thus the CIEDE2000 equation is able to accurately predict the perceived color differences between simple patches.

Often when dealing with image reproduction systems it is desired to determine the perceived differences between images. Traditionally this has been accomplished by using

a standard color difference formula on a pixel-by-pixel basis and then examining statistics such as mean, median, max, etc. This often produces undesirable results when dealing with images that have been spatially altered, such as those processed with a halftone algorithm. Although not designed for use with such complex stimuli, such as those found in digital color images, the CIEDE2000 formula can be extended using a series of spatial filters. Zhang and Wandell describe this technique with the extension of the standard CIELAB ΔE equation, known as the S-CIELAB metric.[2] This paper illustrates the use of similar spatial filtering with the CIEDE2000 color difference formula, as shown in Figure 1.

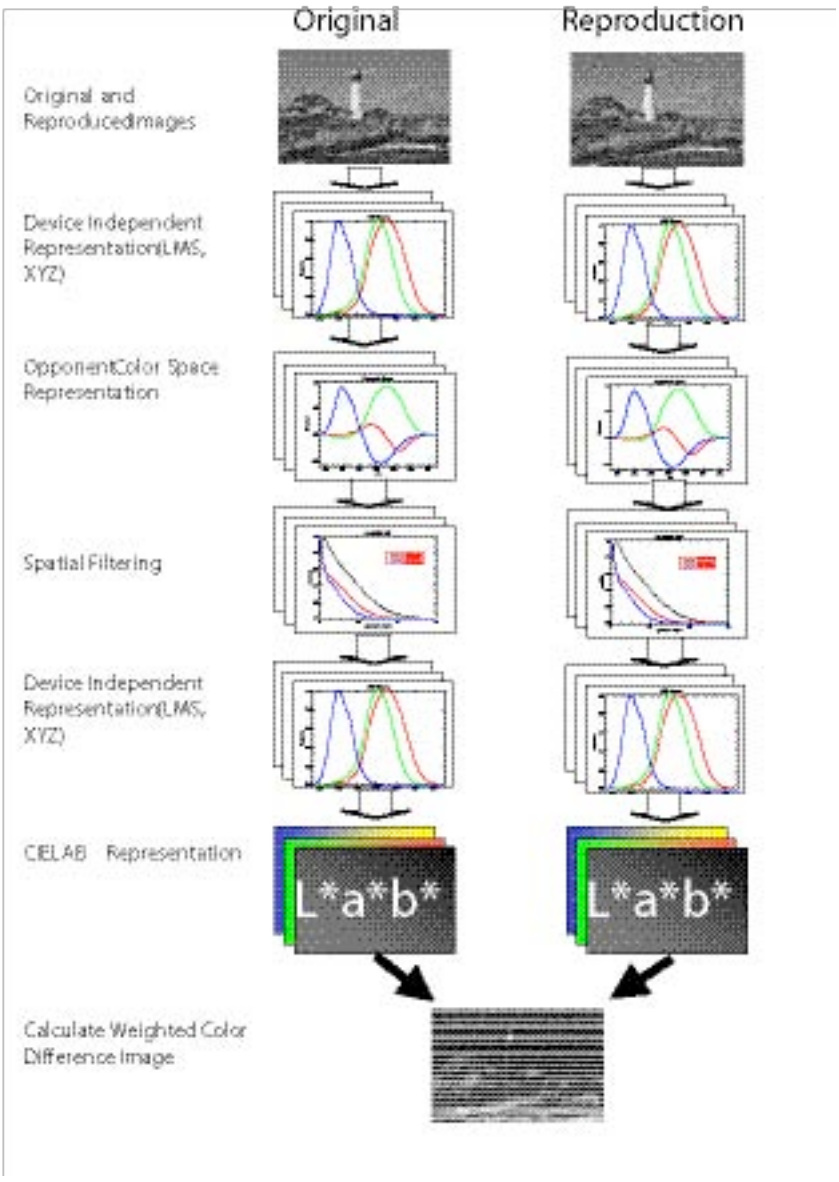

Figure 1. Flow Chart Illustrating Use of Spatial Filters with Color Difference Equations

Figure 2. Original Image and Cluster Dot Halftone Image

Implementation

Device Independent and Opponent Color Transformation

To determine the perceived color difference between image pairs, such as those shown in Figure 2, it is very important to have a well-characterized image device. The first step in using S-CIELAB is to transform the input images into a device independent space, such as LMS cone responsivities or CIE XYZ tristimulus values. This can be accomplished though device characterization, such as the GOG model for computer CRT displays.[3] The primary advantage S-CIELAB offers over a standard color difference formula is the spatial filtering pre-processing step. This filtering is performed in an opponent color space, containing one luminance and two chrominance channels.[2] These channels were determined though a series of psychophysical experiments testing for pattern color separability.[4] The opponent channels, AC_1C_2, are a linear transform from CIE 1931 XYZ or LMS as shown in Equation 1 and Figure 3. One important note about the AC_1C_2 opponent color space is that the three channels are not completely orthogonal. The chrominance channels do contain some luminance information, and vice-versa. This is illustrated in Figure 4, as the "white" lighthouse contains additional chroma information in both the red-green channel, and the blue-yellow channel. The lack of orthogonality has the potential to create color fringes when spatially filtered with different size filters for each channel. As S-CIELAB is not designed to render images, but rather to calculate color differences, this does not cause complications. The opponent color transform also creates negative response for the achromatic channel at certain wavelengths. This, along with the lack of orthogonality, indicates that the transform has potential for improvement,

$$\begin{bmatrix} A \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 0.297 & 0.72 & 0.107 \\ 0.449 & 0.29 & 0.077 \\ 0.086 & 0.59 & 0.501 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \qquad (1)$$

$$\begin{bmatrix} A \\ C_1 \\ C_2 \end{bmatrix} = \begin{bmatrix} 0.990 & 0.106 & 0.094 \\ 0.669 & 0.742 & 0.027 \\ 0.212 & 0.354 & 0.911 \end{bmatrix} \begin{bmatrix} L \\ M \\ S \end{bmatrix}$$

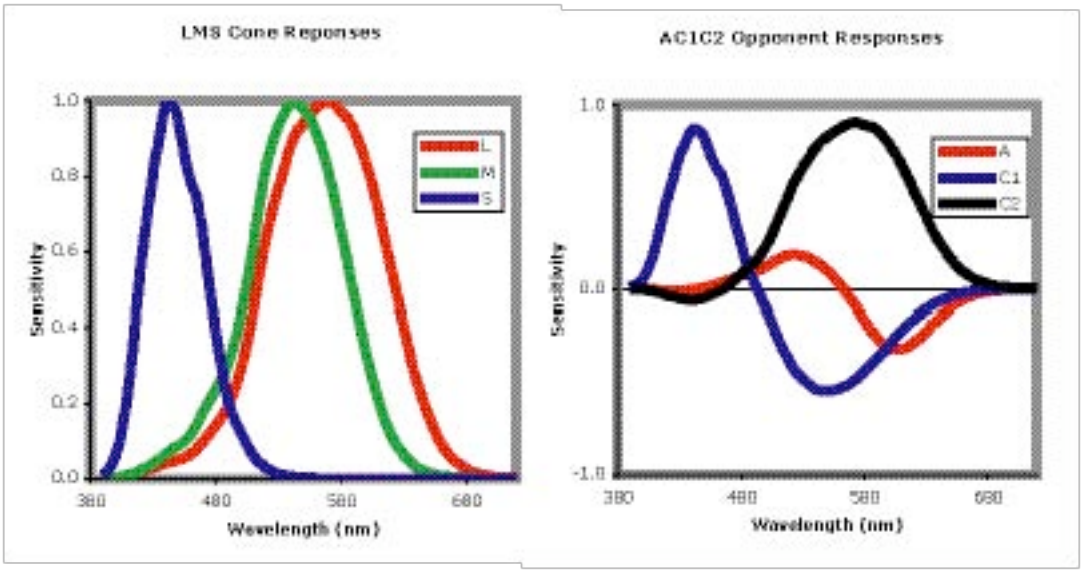

Figure 3. LMS and AC_1C_2 Reponse Functions

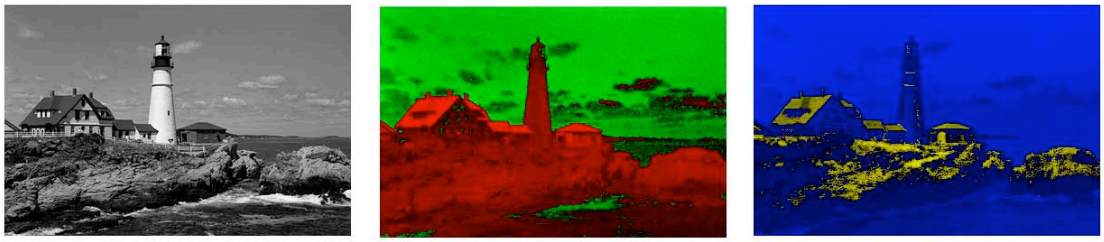

Figure 4. AC_1C_2 Representation of Image (Pseudo-Color)

Spatial Filtering via Convolution

After both images are transformed into the opponent color space, the independent channels can be spatially filtered, using filters that approximate the contrast sensitivity functions (CSF) of the human visual system. This can be accomplished using either a series of convolutions in the spatial domain, or by multiplications in the frequency domain. The traditional S-CIELAB model uses two-dimensional separable convolution kernels. These kernels are unit sum kernels, in the form of a series of Gaussian functions. The unit sum was designed such that for large uniform areas S-CIELAB predictions are identical to the corresponding CIELAB predictions. Equations 2 and 3 show the spatial form of the convolution kernels.

$$filter = k\sum_i w_i E_i \tag{2}$$

$$E_i = k_i e^{-\frac{x^2+y^2}{\sigma_i^2}} \tag{3}$$

The parameters k and k_i normalize the filters such that they sum to one, thus preserving the mean color value for uniform areas. The parameters w_i and σ_i represent the weight and the spread (in degrees of visual angle) of the Gaussian functions, respectively. Table I shows these values for the kernels used in S-CIELAB. It is important to note that these values differ slightly from the published values, as they are already adjusted to sum to one.[5]

Table I. Weight and Spread of Gaussian Convolution Kernel

Filter	*Weight (w_i)*	*Spread (σ_i)*
Achromatic (i=1)	1.00327	0.0500
Achromatic (i=2)	0.11442	0.2250
Achromatic (i=3)	-0.11769	7.0000
Red-Green (i=1)	0.61673	0.0685
Red-Green (i=2)	0.38328	0.8260
Blue-Yellow (i=1)	0.56789	0.0920
Blue-Yellow (i=2)	0.43212	0.6451

The separable nature of the kernels allows for the use of two relatively simple 1-D convolutions of the color planes, rather than a more complex 2-D convolution. This pre-

processing design feature allows S-CIELAB to be used as a spatial extension to existing color difference calculations.[2]

The combination of positive and negative weights in the achromatic channel creates a band-pass filter, as is traditionally associated with luminance contrast sensitivity functions. The positive weights used for the chrominance channels create two low-pass filters. Figure 5 illustrates the relative sensitivity of the three spatial filters, as a function of cycles per degree of visual angle, in both linear and log-log space. These plots were generated by performing a discrete Fourier transform on the convolution kernels.

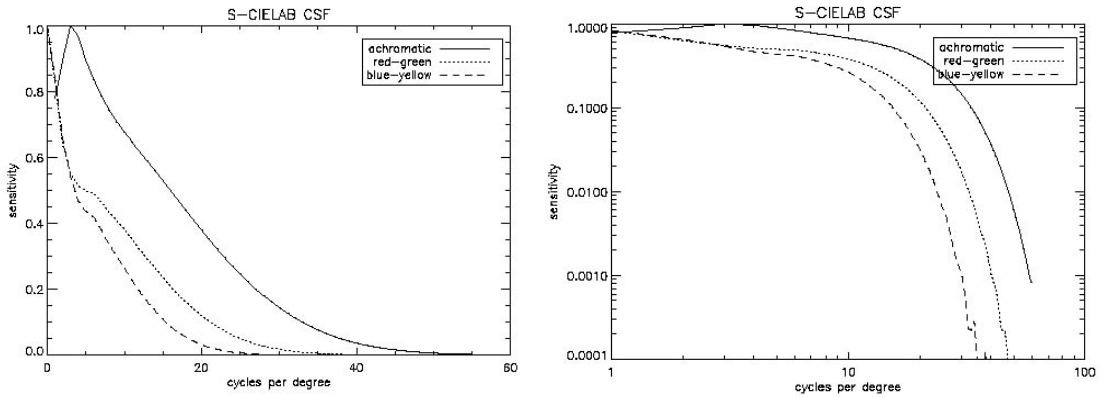

Figure 5. Spatial filters of the S-CIELAB model. These filters are designed to approximate the human Contrast Sensitivity Function (CSF).

The spatial filtering stage simulates the decrease in sensitivity that occurs in the human visual system. As shown in Figure 5, this blurring tends to increase as a function of cycles-per-degree of visual angle. In digital imaging applications, cycles-per-degree is a function of both addressability and viewing distance. For example, if a computer monitor is capable of displaying 72 pixels-per-inch (ppi) and is viewed at 18 inches then there are roughly 23 digital samples per degree of visual angle. This calculation is shown in Equation 4.

$$Cyc/\deg = \frac{ppi}{\frac{180}{\pi} * \tan^{-1}\left(\frac{1\ \text{inch}}{viewing\ \text{distance}}\right)} \tag{4}$$

To illustrate this point, the original and reproduced images were filtered for three viewing distances, which correspond to 10, 50, 100 pixels per degree. The results of the filtering on the achromatic channel are shown in Figure 6. All the images in Figure 6 are the same number of pixels. The image on the left is viewed at a distance that corresponds to 10 pixels per degree of visual angle. The other images are viewed at a distance of 5 and 10 times further away. Therefore the total image itself subtends a smaller visual angle, and the number of pixels per degree increases. The halftone dots of the reproduced image are clearly visible when viewed at 10 samples per degree, and become increasingly less visible as the number of pixels per degree of visual angle increase. This corresponds to actual perception of halftone images. At close distances the individual dots are noticeably visible, but as the viewing distance increases the dots gradually disappear until the image appears to be continuous tone. The blurring of the Red-Green color plane is slightly greater than the achromatic channel, while that of the Blue-Yellow color plane is greater still. This correlates with the known behavior of the human visual system[6]

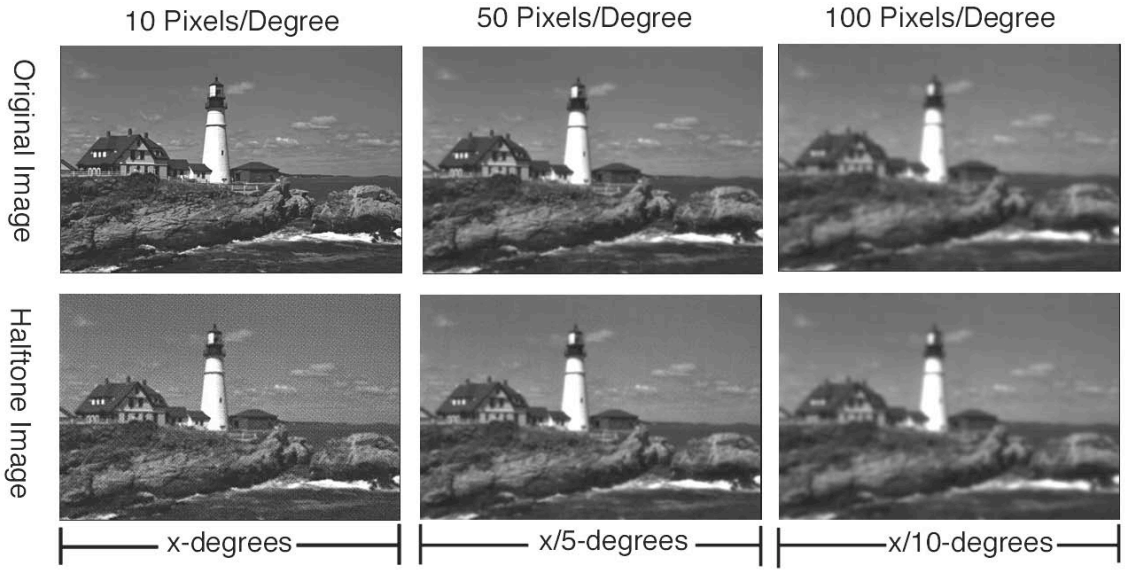

Figure 6. Spatial blurring as a function of samples-per-degree. This can be thought of as increasing the viewing distance of the image. If the images on the left span *x* degrees of visual angle, then the center image pair spans *x/5* degrees, which the images on the right span *x/10* degrees.

Spatial Filtering via Frequency Modulation

If computationally feasible it might be desirable to perform the spatial filtering in the frequency domain, rather than the spatial domain. This enables more precise specification of the shape of the filter with fewer terms. Filtering in the frequency domain is performed using a simple multiplication, rather than a series of convolutions. In order to filter the image, the opponent channels must be first transformed into their respective frequency representations. This can be accomplished using a Fourier transform. Figure 7 shows the log of the frequency spectrum of the original image, calculated via a discrete Fourier transform (DFT).

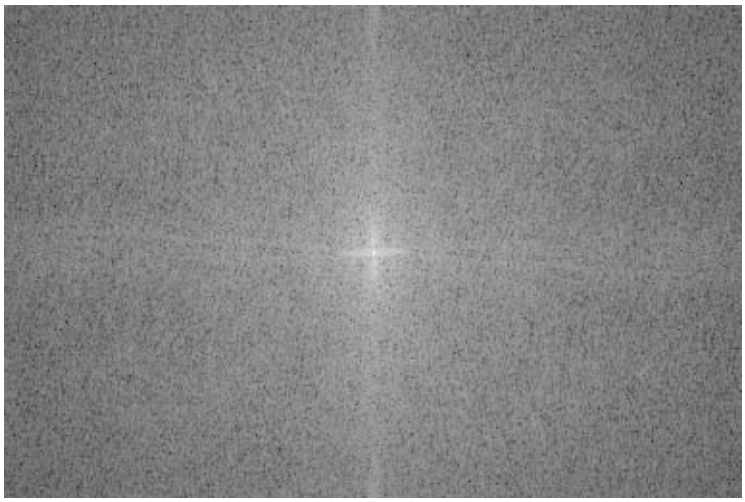

Figure 7. Fourier Transform of Original Image (log scale)

Once in frequency space, it is necessary to obtain frequency representations of the opponent spatial filters. These can be determined by taking the Fourier transform of the convolution kernels, as in Figure 5. Mathematically the convolution operator is identical to multiplication in the frequency domain, so the results of the frequency filtering would be identical to the results from the spatial convolution. The convolution kernels used in S-CIELAB are Gaussian approximations of the human contrast sensitivity functions. The relatively small size of the discrete convolution kernels become Cosine approximations in the frequency domain, rather than the Gaussian shape originally desired. Specifying the filters purely in the frequency domain allows for more precise control over the shape of the filter. Specification of the contrast sensitivity functions is another topic that is heavily researched, and many formulae exist.[7,8,9,10]

A three parameter exponential model, described by Movshon, is a simple description of the general shape of the luminance CSF, which behaves similarly to the S-CIELAB filter. This model is shown in Equation 5.

$$\mathrm{csf}_{\mathrm{lum}}(f) = a \cdot f^{c} \cdot e^{-b \cdot f} \tag{5}$$

The parameters, *a, b,* and *c* can be fit to existing experimental data, if available. Alternatively values of 75, 0.2, and 0.8 for *a*, *b*, and *c* respectively fit reasonably well with the S-CIELAB filters. Figure 8 plots the contrast sensitivity function calculated using the above values, and the corresponding frequency filter.

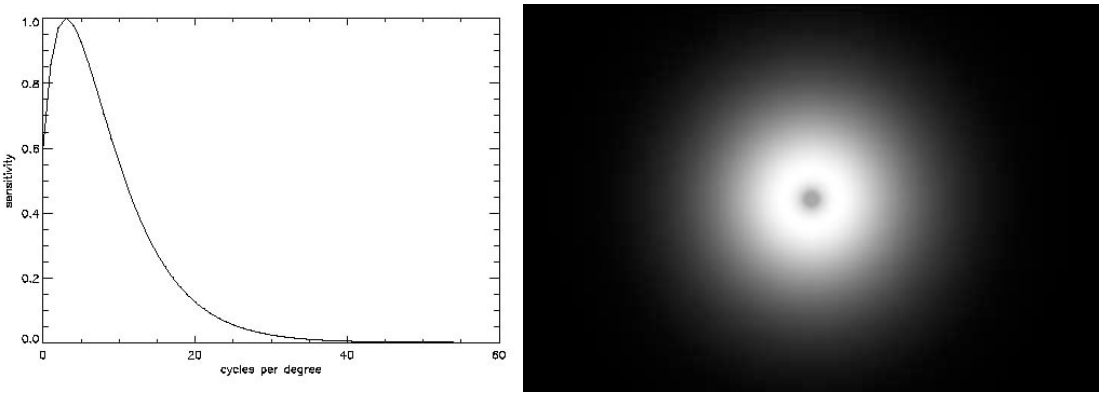

Figure 8. Frequency filter for luminance channel, approximating the human contrast sensitivity function (CSF).

It is important to note that the above filter behaves as a band-pass filter, peaking around 4 cycles-per-degree. Careful consideration needs to be taken regarding the DC component, 0 cycles-per-degree, of the filter. The DC component is essentially the mean value of the image channel. For simple patches, this mean value is the value of the patch itself. The existing color difference formulas are able to accurately predict color differences of simple patches, so it is important to keep this mean value constant. This can be accomplished in several ways. Either the luminance contrast sensitivity function can be truncated into a low-pass filter, or it can be normalized such that the DC component is equal to 1. If the latter method is chosen, then the spatial filtering behaves as a frequency enhancer as well as modulator.

To illustrate the importance of maintaining the DC component, the original image above was filtered with a traditional band-pass luminance CSF. Prior to filtering, the Mean L* value of the image was 47.94, with a standard deviation of 24.18. When filtered with a band-pass filter that modulates the DC component, the mean L* value is reduced to 34.42, with a standard deviation of 19.83. This same image, when filtered with a band-pass filter normalized to 1.0 for the DC component, maintains the original L* distribution. Figure 9 shows the histogram of the L* distribution of both filtered images. This large shift in overall lightness can cause an unwanted loss of contrast or poor predictability of large uniform patches. Therefore, it is strongly suggested that the DC component is preserved when performing frequency filtering.

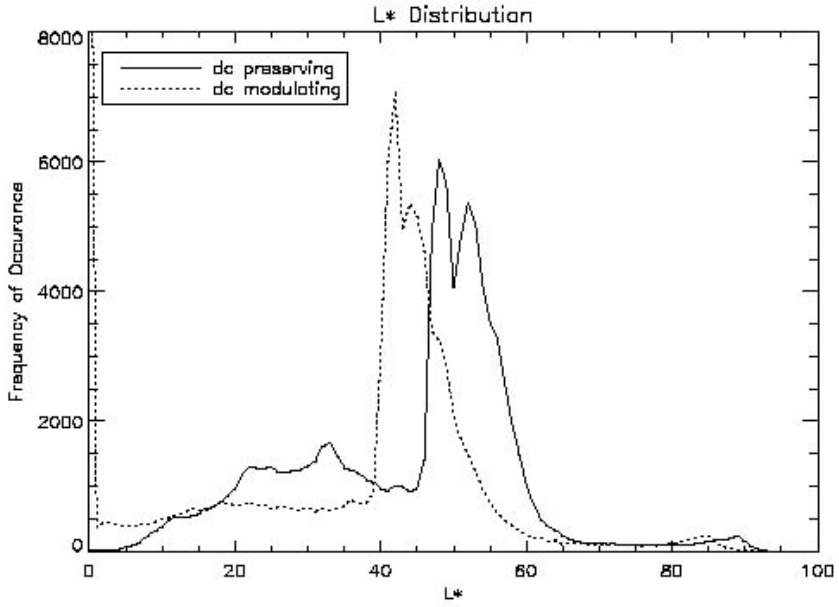

Figure 9. L* Distribution of DC Maintaining (solid) and Modulating (dashed) Filters. The DC Maintaining Filter is the Recommended Technique.

There is considerably less available data for the chrominance channel filters.[4,6,11] The available data can be fit with the sum of two Gaussian functions, as shown in Equation 6.

$$\mathrm{csf}_{\mathrm{chrom}}(f) = a_1 \cdot e^{-b_1 \cdot f^{c1}} + a_2 \cdot e^{-b_2 \cdot f^{c2}} \tag{6}$$

Table 2 shows the values of the six parameters that were fit to the Van der Horst and Poirson data sets. These data were fit using a Newton-Rhapson nonlinear regression

technique, minimizing the sum of squares error between the model and the data. The highest available spatial frequency in the data was 18 cycles-per-degree, so higher frequencies needed to be extrapolated. To help accommodate this extrapolation the data sets were padded with zeros at several higher frequencies. This padding also encourages the Gaussian tails to cut-off at higher frequencies. The Poirson data were first normalized so that they were on the same scale as the Van der Horst data. Figure 10 shows the normalized contrast sensitivity functions of the opponent color filters.

Table II. Parameters for Chrominance CSFs

Parameter	*Red-Green*	*Blue-Yellow*
a1	109.1413	7.0328
b1	-0.0004	0.0000
c1	3.4244	4.2582
a2	93.5971	40.6910
b2	-0.0037	-0.1039
c2	2.1677	1.6487

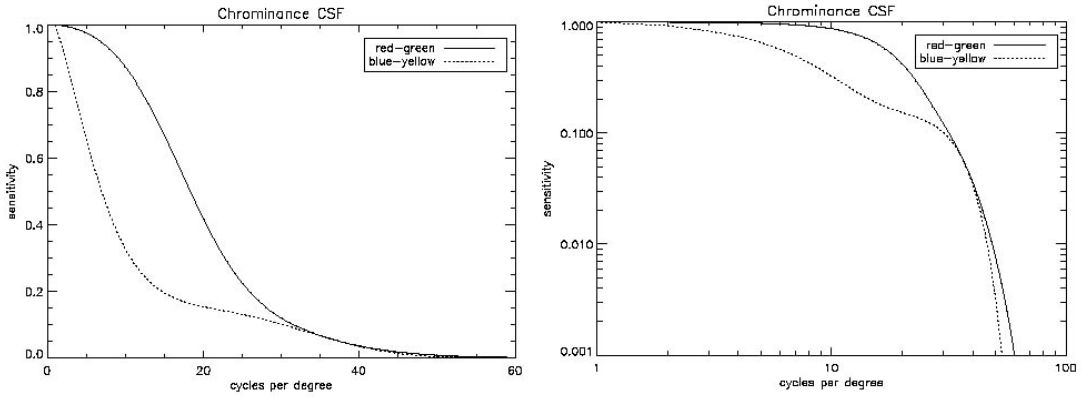

Figure 10. Normalized Opponent Color Contrast Sensitivity

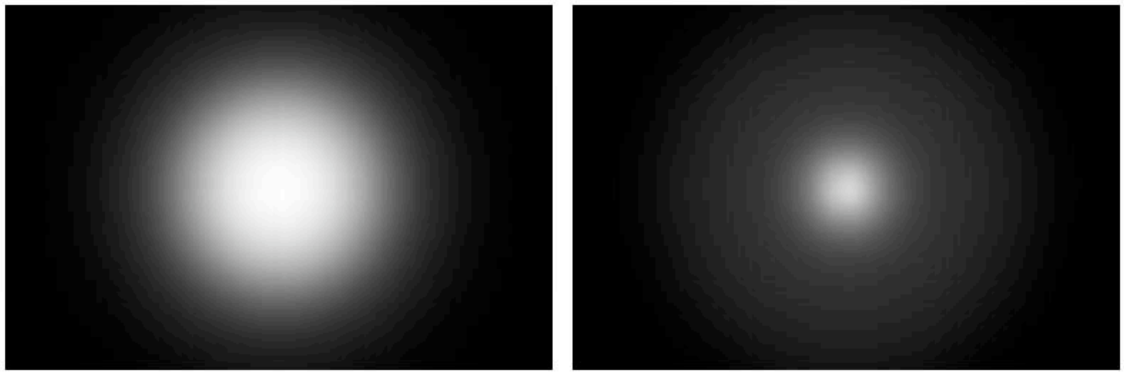

Figure 11. 2D Chrominance Filters. Red-Green on the Left, Blue-Yellow on the Right.

The two-dimensional filters, as shown in Figures 8 and 11, are then multiplied with the discrete Fourier transform of the individual opponent channels. The filtered images are then transformed back into the spatial domain via an inverse Fourier transform. This process can easily be extended to handle orientation-specific aspects of the human visual system, by using non-isotropic filters. Modulating the diagonal orientations more would be able to predict the decreased visibility of half-tone screens, as described by the oblique effect. This is a future direction for enhancements to the S-CIELAB model.

CIE XYZ and CIELAB

The filtered opponent channels are then transformed back into CIE XYZ space using the inverse of Equation 1, as shown in Equation 7.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.979 & 1.189 & 1.232 \\ 1.535 & 0.764 & 1.163 \\ 0.445 & 0.135 & 2.079 \end{bmatrix} \begin{bmatrix} A \\ C_1 \\ C_2 \end{bmatrix} \quad (7)$$

The filtered XYZ pixel values for both the original and the test image are then transformed into the CIELAB space, using Equations 8-10. It is necessary to know the tristimulus values of the white point for the display device, Xn, Yn, Zn, in order to calculate the CIELAB coordinates. This can be determined through the device characterization.

$$L^* = 116\left(\frac{Y}{Yn}\right)^{\frac{1}{3}} - 16 \quad , \text{if } \frac{Y}{Yn} > 0.008856$$
$$\text{else, } = 903.3\left(\frac{Y}{Yn}\right) \tag{8}$$

$$a^* = 500\left[\left(\frac{X}{Xn}\right)^{\frac{1}{3}} - \left(\frac{Y}{Yn}\right)^{\frac{1}{3}}\right], \text{ if } \frac{X}{Xn}, \frac{Y}{Yn} > 0.008856$$
$$\text{else, } 500^*\left[7.787\left(\frac{X}{Xn}\right) + \frac{16}{116}\right] - \left[7.787\left(\frac{Y}{Yn}\right) + \frac{16}{116}\right] \tag{9}$$

$$b^* = 200\left[\left(\frac{Y}{Yn}\right)^{\frac{1}{3}} - \left(\frac{Z}{Zn}\right)^{\frac{1}{3}}\right], \text{ if } \frac{Z}{Zn}, \frac{Y}{Yn} > 0.008856$$
$$\text{else, } 200^*\left[7.787\left(\frac{Y}{Yn}\right) + \frac{16}{116}\right] - \left[7.787\left(\frac{Z}{Zn}\right) + \frac{16}{116}\right] \tag{10}$$

Color Difference Formula

Once the CIELAB coordinates are calculated for the filtered images, color differences can be determined on a pixel-by-pixel basis. The standard CIE ΔE^*_{ab} color difference equation is traditionally used with S-CIELAB, though it is recommended that a more perceptually uniform metric is utilized. This section details the use of the newly recommended CIEDE2000 color difference equation on the filtered images.

In 1994 the CIE suggested a new formula for color difference calculations that included lightness, chroma, and hue weighting functions.[12] This formula has been met with much success. The CIE94 formula has recently been extended to include a interactive hue and chroma term to improve performance for the blue region of color space, correcting for perceived constant-hue nonlinearity.[13] An adjusted a* term, to improve performance of low-chroma (gray) colors, was also in this new extension.[1] In addition, this formula includes a hue dependent function to correct for perceived hue differences.[14] The CIE has recommended the use of this new color difference formula, CIEDE2000.

The first step of the CIEDE2000 formula is an adjustment of the a* axis to correct for the color difference perception of low chroma colors. This is accomplished using a modified Gaussian curve, similar to Morovic's GCUSP function, on the mean chroma difference, as shown in Equation 11-13.[15]

$$C^*_{[x,y]} = \sqrt{a^*_{[x,y]}{}^2 + b^*_{[x,y]}{}^2} \tag{11}$$

$$\overline{C}^*_{[x,y]} = \frac{C^*_{orig[x,y]} + C^*_{repro[x,y]}}{2} \tag{12}$$

$$G_{[x,y]} = 0.5\left(1 - \sqrt{\frac{\overline{C}^*_{[x,y]}{}^7}{\overline{C}_{[x,y]}{}^{*7} + 25^7}}\right) \tag{13}$$

Equation 13 is then used to calculate the lightness, chroma and hue rectangular differences on a per-pixel basis for the image pair.

$$a'_{[x,y]} = \left(1 + G_{[x,y]}\right) a^*_{[x,y]} \tag{14}$$

$$L'_{[x,y]} = L^*_{[x,y]} \tag{15}$$

$$C'_{[x,y]} = \sqrt{a'_{[x,y]}{}^2 + b^*_{[x,y]}{}^2} \tag{16}$$

$$h'_{[x,y]} = \tan^{-1}\left(\frac{b^*_{[x,y]}}{a'_{[x,y]}}\right) \tag{17}$$

In each equation, the subscript $_{[x,y]}$ refers to the x-y coordinates of each pixel for the image pairs. For this example, the image pairs are referred to as the original and reproduced images.

$$\Delta L'_{[x,y]} = L'_{orig[x,y]} - L'_{repro[x,y]} \tag{18}$$

$$\Delta C'_{[x,y]} = C'_{orig[x,y]} - C'_{repro[x,y]} \tag{19}$$

$$\Delta h'_{[x,y]} = h'_{orig[x,y]} - h'_{repro[x,y]} \tag{20}$$

$$\Delta H_{[x,y]} = 2 \cdot \sqrt{C'_{orig[x,y]} C'_{repro[x,y]}} \cdot \sin\left(\frac{\Delta h'_{[x,y]}}{2}\right) \tag{21}$$

Care must be taken when calculating hue angle differences between the original and reproduced images, if the hue angles reside in different hue quadrants. If the absolute difference between the two hue angles is greater than 180 degrees, then it is important to add 360 to the smaller of the hue angles, as shown in Equation 22.

$$\begin{aligned} &if \left| \Delta h'_{[x,y]} \right| > 180 \text{ then} \\ &\min(h'_{\text{orig}[x,y]}, h'_{rep[x,y]}) + 360 \end{aligned} \tag{22}$$

When coding Equation 22 in a computer program, it is often more efficient to split the calculations into two separate steps, as shown in Equation 23.

$$\begin{aligned} &if \ \Delta h'_{[x,y]} > 180 \text{ then } \Delta h'_{[x,y]} = \Delta h'_{[x,y]} - 360 \\ &elseif \ \Delta h'_{[x,y]} < -180 \text{ then } \Delta h'_{[x,y]} = \Delta h'_{[x,y]} + 360 \end{aligned} \tag{23}$$

Next, the arithmetic mean lightness, chroma, and hue-angle between the sample and batch images must be calculated. These calculations are shown in Equations 24-26.

$$\overline{L}'_{[x,y]} = \frac{L'_{orig[x,y]} + L'_{rep[x,y]}}{2} \tag{24}$$

$$\overline{C}'_{[x,y]} = \frac{C'_{orig[x,y]} + C'_{rep[x,y]}}{2} \tag{25}$$

$$\overline{h}'_{[x,y]} = \frac{h'_{orig[x,y]} + h'_{rep[x,y]}}{2} \tag{26}$$

Again, care must be taken when determining the mean hue angle, if the hue values for the pixels reside in different quadrants. Equation 22 should be applied when calculating mean hue as well.

Weighting functions are then calculated, to adjust for the perceived color differences between lightness, chroma, and hue in the CIELAB space. These weighting functions are also calculated on pixel-per-pixel basis. The lightness and chroma functions, S_L, S_C are shown in Equations 27 and 28.

$$S_{L[x,y]} = 1 + \frac{0.015\left(\overline{L}^*_{[x,y]} - 50\right)^2}{\sqrt{20 + \left(\overline{L}^*_{[x,y]} - 50\right)^2}} \tag{27}$$

$$S_{C[x,y]} = 1 + 0.045\overline{C}'_{[x,y]} \tag{28}$$

The hue weighting is a function of both hue angle and chroma. First, the hue angle dependency is determined, using Equation 29. This is then combined with the chroma dependency in Equation 30.

$$T_{[x,y]} = 1 - 0.17\cos(\bar{h}'_{[x,y]} - 30) + 0.24\cos(2\bar{h}'_{[x,y]}) + 0.32\cos(3\bar{h}'_{[x,y]} + 6) - 0.20\cos(4\bar{h}'_{[x,y]} - 63) \quad (29)$$

$$S_{H[x,y]} = 1 + 0.015\bar{C}_{[x,y]} \cdot T_{[x,y]} \quad (30)$$

The blue region of CIELAB is known to be highly nonlinear in regards to hue angle and chroma interaction. Thus, a rotation function has been created to compensate for this interaction. This rotation starts with a chroma GCUSP function, as shown in Equation 31.

$$R_{C[x,y]} = 2.0\sqrt{\frac{\bar{C}_{[x,y]}{}^{*7}}{\bar{C}_{[x,y]}{}^{*7} + 25^7}} \quad (31)$$

This is followed by a hue angle dependency, as shown in Equation 32.

$$\Delta\theta_{[x,y]} = 30e^{-\left(\frac{\bar{h}'_{[x,y]} - 275°}{25}\right)^2} \quad (32)$$

The rotation function, Equation 33, is applied.

$$R_{t[x,y]} = -\sin(2\Delta\theta_{[x,y]})R_{c[x,y]} \quad (33)$$

Finally, the total color difference for each pixel can be calculated, using Equation 34.

$$\Delta E_{[x,y]} = \sqrt{\left(\frac{\Delta L'_{[x,y]}}{K_L S_{L[x,y]}}\right)^2 + \left(\frac{\Delta C'_{[x,y]}}{K_C S_{C[x,y]}}\right)^2 + \left(\frac{\Delta H'_{[x,y]}}{K_H S_{H[x,y]}}\right)^2 + R_{T[x,y]}\left(\frac{\Delta C'_{[x,y]}}{K_C S_{C[x,y]}}\right)\left(\frac{\Delta H'_{[x,y]}}{K_H S_{H[x,y]}}\right)} \quad (34)$$

The parametric weights, K_L, K_C, K_H can be fit to existing data sets, if they exist. For most imaging applications, these weights are unknown and should be all set to 1.0.

Example and Data Reduction

Color Difference Formulae

The original design of S-CIELAB was a spatial pre-processing prior to standard color difference calculations. This allows for great flexibility with the choice of color difference formulas. The result of the calculations is an error image, where each pixel represents the perceived error at that given point. Figure 12 shows the scaled CIEDE2000 error image for the halftone image pair above, when viewed at 30 pixels per degree of visual angle. The error image can be very valuable for pinpointing systematic or gross errors in an imaging chain.

Figure 12. CIEDE2000 Error Image

Many times it is more desirable to reduce the error image into a single number, representing the overall perceived difference. This can be accomplished using image statistics, if care is taken. A common practice is to take the mean CIEDE2000 of the image. While useful in providing an overall idea, this can often lead to error masking. For example, when reproducing an original image one imaging system might cause large systematic shifts in hue while another system causes more random individual pixel shifts. It is possible for the two reproduced images to have equal mean error, despite the large systematic shifts being more perceptibly noticeable. The spatial filtering itself should give greater weight to the lower frequency large shifts and less weight to the individual pixels, but it is still possible to have identical mean errors. This can be somewhat overcome when comparing additional statistics such as the error variance, standard

deviation, median, and other percentiles. The error maximum is also sometimes examined, but the maximum is often not indicative towards the overall error trend. The error maximum can be very valuable for detecting threshold image differences.

Figure 12 shows the filtered CIE ΔE^*_{ab} and CIE ΔE^*_{94} error images for the same image pair.

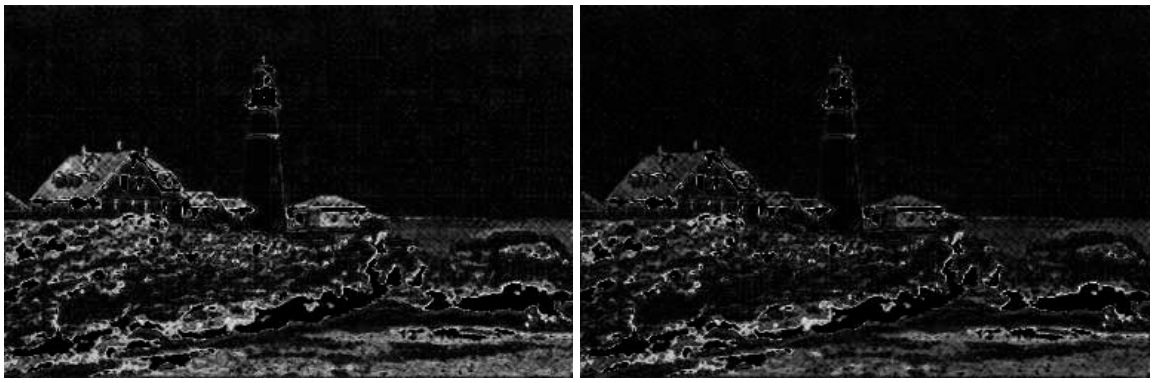

Figure 13. CIE ΔE_{ab} (left) and CIE ΔE_{94} (right) error images

For comparison, the image statistics produced from using three CIE color difference formulae are shown in Table 3. It should be noted that both CIE ΔE^*_{94} and CIEDE2000 errors tend to be smaller than the standard ΔE^*_{ab}, due to the parameter weighting functions. An interesting note is the closer grouping of all the CIEDE2000 errors, as shown by the lower error standard deviation.

Table 3. Image Statistics for CIE Color Difference Formula

Color Difference Formula	Mean	Std Dev	Median
ΔE^*_{ab}	6.45	10.48	2.40
CIE ΔE^*_{94}	5.07	8.50	2.03
CIEDE2000	4.21	5.49	2.08

Viewing Distance and Resolution Dependence

The spatial filtering pre-processing accounts for the spatial properties of the human visual system by removing error information that is not visible to the observer. As such, the

viewing conditions in which the images are viewed are very important and can greatly influence the color difference calculation. The error images for three different viewing conditions corresponding to 10, 50, and 100 pixels-per-degree were calculated, and are shown in Figure 14.

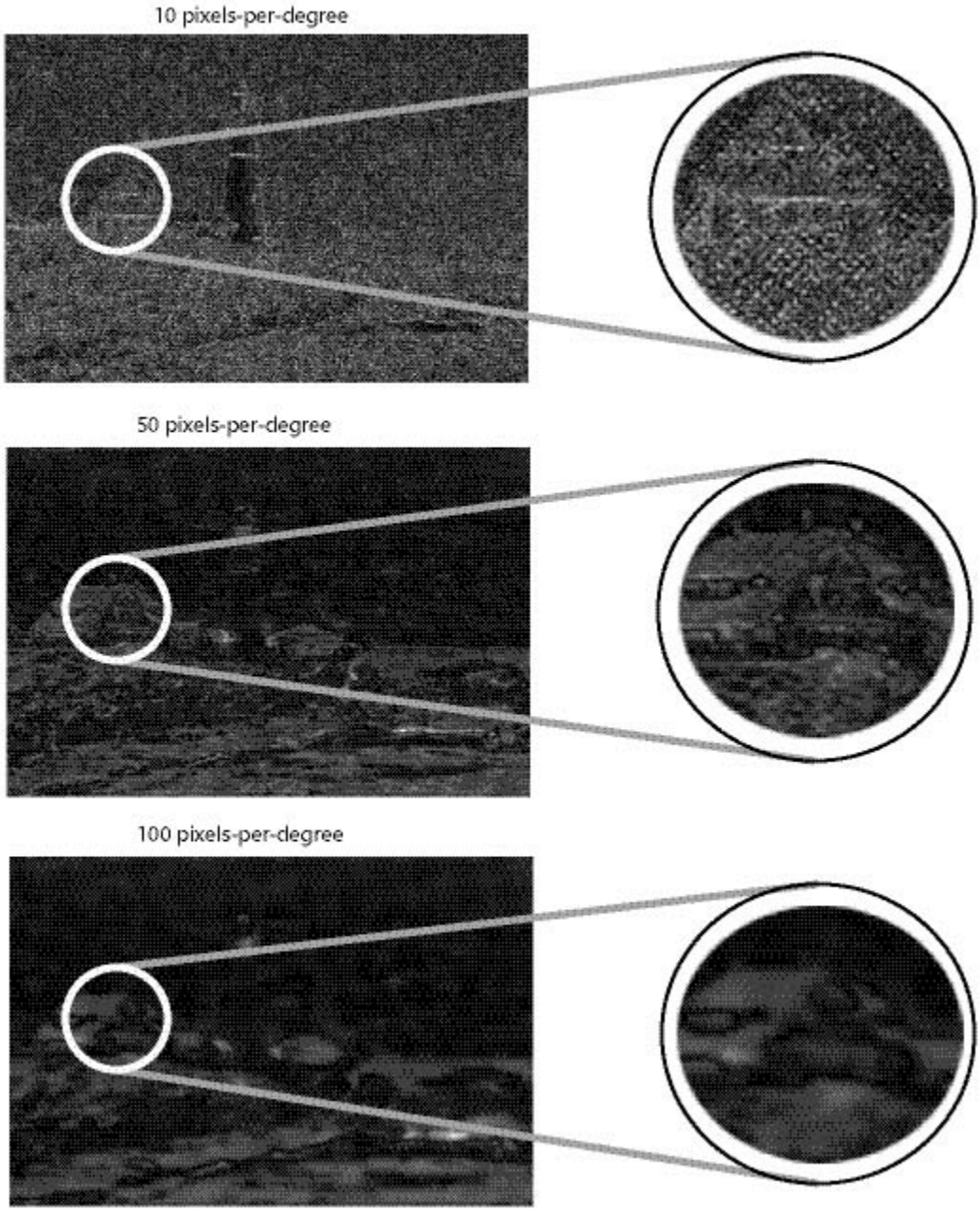

Figure 14. CIEDE2000 Error images calculated for three different viewing conditions (resolutions).

The error images are very different, depending on the viewing conditions. At 10 pixels-per-degree the halftone pattern is clearly noticeable. This pattern is much less noticeable at 50 pixels-per-degree, and practically disappears at 100 pixels-per-degree. This corresponds with the general behavior of halftone images. It is important to note that the error images shown in Figure 14 have been normalized by the maximum error to illustrate the error pattern. The image statistics also illustrate the importance of viewing distance in S-CIELAB calculations. Table 4 shows the mean and standard deviation of the CIEDE2000 color differences calculated for the three viewing conditions.

Table 4. Error image statistics for different viewing conditions

Viewing Condition	Mean CIEDE2000	Std. Dev CIEDE2000
10 pixels-per-degree	9.52	5.35
50 pixels-per-degree	2.11	1.37
100 pixels-per-degree	1.66	1.17

The average color difference between the original and the halftone image is very large when viewed at a distance corresponding to 10 pixels-per-degree. This difference is decreased greatly when viewed at 50 pixels-per-degree, and even more so when viewed at 100 pixels-per-degree.

Conclusion

An overview of using the state-of-the-art color difference formula along with a pre-processing spatial filtering for comparing image color difference has been presented. The basis of this technique is S-CIELAB along with CIEDE2000. The benefits and pitfalls of performing the filtering in the spatial domain, with a series of convolutions, or in the frequency domain, with Fourier multiplication have been described. When possible, it is advantageous to perform these calculations in the frequency domain. Care must be taken to maintain the DC component of the luminance channel, or large lightness shifts can occur. Finally, we have discussed data reduction and compared the results of three existing CIE color difference formula, and three different viewing conditions.

Acknowledgements

The authors wish to thank Kazuhiko Takemura and Fuji Photo Film for their support of this research work.

References

[1] M. R. Luo, G. Cui, and B. Rigg, "The development of the CIE 2000 Colour Difference Formula," *Color Research and Applications*, 26, 340-350 (2001).
[2] X. M. Zhang and B. A. Wandell, "A spatial extension to CIELAB for digital color image reproduction," *Society for Information Display Symposium Technical Digest*, 27, 731-734 (1996).
[3] R. B. Berns, *Billmeyer and Saltzman's Principles of Color Technology*, John Wiley & Co, NY (2000).
[4] A. B. Poirson and B. A. Wandell, "The appearance of colored patterns: pattern-color separability," *J. Opt. Soc. A.*, 10, 2458-2470 (1993).
[5] X. Zhang, *http://white.stanford.edu/~brian/scielab/scielab.html*
[6] K. Mullen, "The contrast sensitivity of human color vision to red-green and blue-yellow chromatic gratings," *J. Physiology*, 359, 381-400 (1985).
[7] P. G. J. Barten, *Contrast Sensitivity of the Human Eye and Its Effects on Image Quality*, SPIE, Bellingham, WA (1999).
[8] S. Daly, The Visible Differences Predictor: An algorithm for the assessment of image fidelity*, Ch. 14 in Digital Images and Human Vision*, A.B. Watson, Ed., MIT Press, Cambridge, 179-206 (1993).
[9] P. Lennie and M. D'Zmura, "Mechanisms of color vision," *CRC Critical Reviews of Neurobiology*, 333-400 (1988).
[10] T. Movshon and L. Kiorpes, "Analysis of the development of spatial sensitivity in monkey and human infants," *J. Opt. Soc. A.*, 5 2166-2172 (1988).
[11] G. J. C. Van der Horst and M. A. Bouman, "Spatiotemporal chromaticity discrimination," *J. Opt. Soc. A.*, 59 1482-1488 (1969).
[12] CIE Publication No. 101, "Parametric Effects in Colour-Difference Evaluation," (1993).
[13] G.J. Braun, F. Ebner, and M.D. Fairchild, "Color Gamut Mapping in a Hue-Linearized CIELAB Color Space," *Proceedings of the 6th IS&T/SID Color Imaging Conference*, 163-168 (1998).
[14] R. S. Berns, "Derivation of a hue-angle dependent, hue-difference weighting function for CIEDE2000," Proceedings of AIC Color '01, Rochester, 638-641 (2001).
[15] J. Morovic and M.R. Luo, "Gamut Mapping Algorithms Based on Psychophysical Experiment," *Proceedings of the 5th IS&T/SID Color Imaging Conference*, 44-49 (1997).