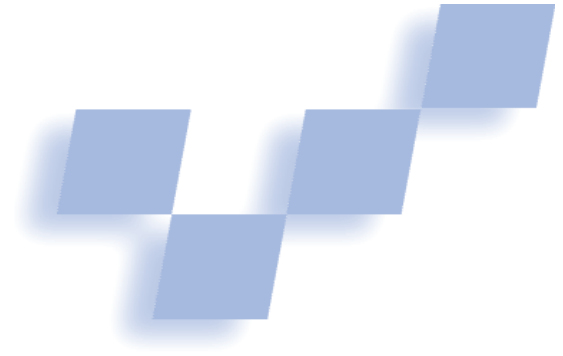


Detail Preserving Reproduction of Color Images for Monochromats and Dichromats



Karl Rasche, Robert Geist, and James Westall
Clemson University

An algorithm that transforms color to gray scale preserves image detail by maintaining distance ratios during the reduction process. An extension of the transformation can aid color-deficient observers.

In spite of the ever increasing prevalence of low-cost, color printing devices, gray-scale printers remain in widespread use. Authors producing documents with color images for any venue must account for the possibility that the color images might be reduced to gray scale before they are viewed. Because conversion to gray scale reduces the number of color dimensions, some loss of visual information is generally unavoidable. Ideally, we can restrict this loss to features that vary minimally within the color image. Nevertheless, with standard procedures in widespread use, this objective is not often achieved, and important image detail is often lost. Consequently, algorithms that convert color images to gray scale in a way that preserves information remain important.

Human observers with color-deficient vision may experience the same problem, in that they may perceive distinct colors to be indistinguishable and thus lose image detail. The same strategy that is used in converting color images to gray scale provides a method for recoloring the images to deliver increased information content to such observers.

Gray-scale conversion

A gray-scale conversion algorithm is a dimension-reducing function that maps points in color space—typically RGB tristimulus values $(R, G, B) \in \mathfrak{R}^3$ —to a subset of \mathfrak{R}^1 . Any such mapping assumes a color model, which includes specification of chromaticity values for red, green, and blue primaries as well as a reference white spectrum and a linearizing gamma value. The most common RGB model is probably the National Television Standards Committee (NTSC) standard, and the most common gray-scale conversion algorithm is mapping to luminance, Y , as follows:

$$Y = 0.299 \times R^{2.2} + 0.587 \times G^{2.2} + 0.114 \times B^{2.2} \quad (1)$$

Here the gamma value for the model is 2.2, and the coefficients are derived from the chromaticity values for the primaries as well as the reference white, which for this model is CIE Illuminant C, designed to represent daylight. Chromaticity values are defined in terms of the CIE XYZ color space of the standard observer. The XYZ color space was designed in 1931 by the Commission Internationale de l'Eclairage (International Commission on Illumination) for matching the perceived color of any given spectral energy distribution. Three primaries, \mathbf{X} , \mathbf{Y} , and \mathbf{Z} , and three associated matching functions, \bar{x}_λ , \bar{y}_λ , \bar{z}_λ , were identified with the property that, given any spectral energy distribution, $P(\lambda)$, the nonnegative weights

$$\begin{aligned} X &= k \int P(\lambda) \bar{x}_\lambda d_\lambda \\ Y &= k \int P(\lambda) \bar{y}_\lambda d_\lambda \\ Z &= k \int P(\lambda) \bar{z}_\lambda d_\lambda \end{aligned}$$

can be used to construct a color, $c = X\mathbf{X} + Y\mathbf{Y} + Z\mathbf{Z}$, that will match $P(\lambda)$ up to human perception. The scaling constant, k , depends on the spectrum of the reference white. Chromaticity values are then 2D, normalized coordinates (x, y) given by

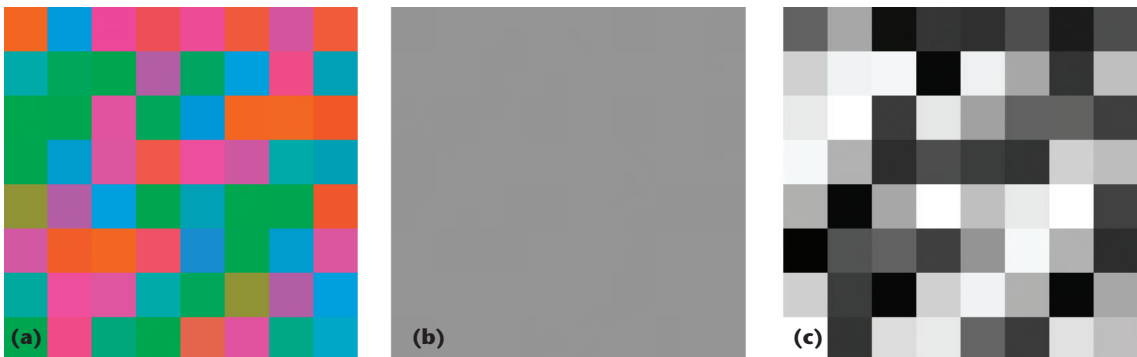
$$x = \frac{X}{X + Y + Z} \quad y = \frac{Y}{X + Y + Z}$$

Different color models—such as the sRGB model commonly used to describe color monitors—with different chromaticities (or reference white or gamma) will yield mappings to luminance that differ slightly from Equation 1. Details on color space definitions are available elsewhere.¹

When mappings such as Equation 1 are used, colors that have extremely small differences in luminance but



1 Conversion to gray scale: Substantial image detail is lost when the (a) color photo is converted to (b) gray scale using the standard NTSC map to luminance. (c) An alternative mapping preserves some of the detail.



2 Conversion to gray scale: All image detail is lost when the (a) color image is converted to (b) gray scale using the standard NTSC map to luminance. (c) An alternative mapping preserves some of the detail.

large differences in chrominance are mapped to similar shades of gray. Colors that were easily distinguishable in the original image, due to chrominance variation, can become indistinguishable in the gray-scale image. Figure 1 shows an example of this situation. Figure 1a is an RGB digital photograph. For Figure 1b, we applied Equation 1 to the image of Figure 1a to produce an encoded luminance, Y , and then linearly mapped the Y values to pixel gray levels. The red berries, which stand out markedly from the green leaves in the color image, are effectively lost in the gray image. Figure 1c shows an alternative mapping, which preserves some of the detail from Figure 1a.

Figure 2 shows a more dramatic, if contrived, example where the colored blocks in Figure 2a (the original image) were chosen to have nearly identical luminance. All image detail is lost (see Figure 2b) when we apply the luminance mapping of Equation 1. Figure 2c shows an alternative mapping, which preserves some of the detail from Figure 2a.

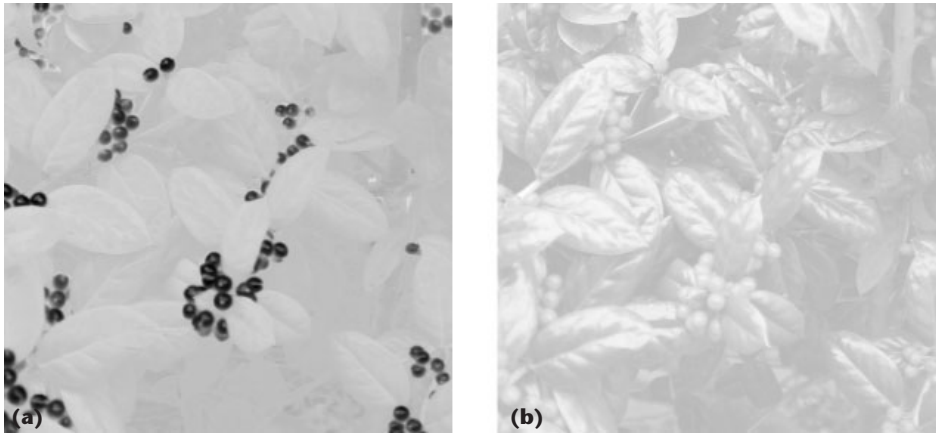
An effective gray-scale conversion method should preserve the detail in the original color image. Our fundamental premise is that this is best achieved by a perceptual match of relative color differences between the color and the gray-scale images. In particular, the perceived color difference between any pair of colors should be proportional to their perceived gray difference. In this article we describe a new approach to the gray-scale conversion problem that is built on this premise.

Our method automatically constructs a linear mapping from \mathfrak{R}^3 to \mathfrak{R}^1 that preserves, to the extent possible, the perceived distances between those points in \mathfrak{R}^3 that comprise a specific color image and their mapped values in \mathfrak{R}^1 . The derived mapping function depends on the characteristics of the input image and incorporates information from all three dimensions of the color space. After describing our detail-preserving gray-scale transform, we show how a straightforward extension recolors images in a way that preserves detail for viewers with color-deficient vision.

Work related to gray-scale conversion

The gray-scale conversion problem resembles the tone mapping problem of displaying high dynamic range images on low dynamic range displays.² In both cases, the objective is to preserve an abundance of visual information within the constraints of a limited gamut. Our approach to gray-scale conversion, like Tumblin and Rushmeier's tone mapping, seeks a perceptual match. Nevertheless, tone mapping is generally concerned with compression of the gamut range, whereas we are interested in compression of gamut dimensionality.

The problem might also be compared to color quantization, as it attempts to display a large number of colors with a small palette. Nevertheless, palette selection is an integral part of color quantization methods. Palette colors are most often selected from a large gamut, which at least includes the gamut of the original image. In our



3 Principal components: (a) Projection of the image of Figure 1 along the first principal component in CIE Lab space and (b) along the second principal component.

problem, the palette is fixed to a gray ramp, and so the principal task, as well as the flexibility available, for any color quantization method has been removed. Mapping image colors to the selected palette in color quantization methods is usually a variation on pixel error-diffusion,³ but such diffusion does not compensate for a problematic palette. When a standard error diffusion mapping is applied to the color image of Figure 1a, the result is visually indistinguishable from the image of Figure 1b.

Color gamut mapping for printing with a small set of inks also resembles our problem. Here, colors are reproduced using only one or two base ink colors. The entire color gamut of an image to be printed is generally not available, and an image must be mapped into the available gamut in a meaningful way. Stollnitz, Ostromoukhov, and Salesin⁴ examine the case of printing with an arbitrary number of base ink colors. For the case of a single ink, they seek to preserve luminance contrast by using a monotonic mapping of input luminance to ink density. This method ignores chrominance of the input image and thus will suffer the same loss of chrominance-based detail that Equation 1 does.

The reduction of high-dimensional data to a lower dimension is also a well-studied problem. A standard linear technique is principal component analysis (PCA). In PCA, a set of orthogonal vectors—the principal components—lying along the directions of maximal data variation is found. To reduce the dimensions of the data, a subset of the principal components form a subspace onto which the original data is projected. While preserving the color variation is necessary for pleasing gray-scale images, it is not sufficient. Consider again the example color image in Figure 1. If we have an image with most of the colors clustered in two regions, here red and green, we can envision a dumbbell-shaped distribution of pixels in color space. By preserving variance, one end of the dumbbell will be mapped to the light end of the gray-scale gamut while the other will be mapped to the dark end. If we then examine the histogram of the resulting gray-scale image, we will find a large number of empty gray bins in the center of the range. As such, any detail within the red and green dumbbell clusters will be reproduced over a small gray range and may not

be perceived. Figure 3 shows the projection of the colored photograph of Figure 1 along the first two principal components in CIE Lab space. CIE Lab color space is an alternative 3D (L , a , b) color model aimed at perceptual uniformity. That is, pairs of colors at equal measured distances from one another in CIE Lab space are perceived to be at approximately equal distances from one another. In this model, the L component is luminance, the a component is a red–green opponent value, and the b component is a blue–yellow opponent value.¹

In the first principal component, the red berries contrast with the green leaves, as expected. Neverthe-

less, we find significant detail in the leaves from the projection along the second principal component. A PCA-based approach would appear to require optimization between contrast and detail by mixing principal components.

We can characterize an alternative class of methods for dimensional reduction as triangulation techniques, in that selected sets of point triples are mapped to a lower dimensional space in a way that maintains exact distances within each triple. While this distance-preserving feature is attractive for preserving detail in our problem, distances across triples are not preserved, and so the end result depends on which triples are selected. We would like to preserve proportional differences between all pairs of points.

Another class of dimensional reduction techniques attempts to find a nonlinear transformation of the data into the lower dimensional space. Examples of these methods include local linear embedding (LLE)⁵ and Isomap.⁶ These transforms work well with data that has some inherent (although perhaps complex) parameterization. They attempt to maintain local distances, corresponding to the original data lying along a higher dimensional manifold. As such, they generally require a “magic number” defining the size of the neighborhood of local distances to observe, and the results can be somewhat sensitive to the neighborhood size. Further, as they only work at a local level, it is not clear whether they can reproduce both contrast and detail when reducing the dimension of the colors. It is also not clear that the colors from an arbitrary image lie along a single manifold. Roweis and Saul⁵ examined the performance of LLE with a dumbbell-shaped data set and achieved results no better than with PCA.

Color vision deficiencies

Those who suffer from deficiencies in color vision deal with reduced color gamuts on a daily basis. Our algorithm can be used to produce images that convey increased information content to such individuals.

Deficiencies in color vision arise from differences in pigmentation of optical photoreceptors.⁷ Normal vision has three distinct pigmentations of *cones*—the

Simulating Color Deficient Vision

Researchers have done significant work in simulating color deficient vision.¹⁻⁴ In these simulations, colors are transformed into a color space based on cone response. From this color space, we can simulate a cone deficiency by collapsing one dimension to a constant value. From empirical data, we can determine the value to which the deficient cone dimension should be collapsed.

Walraven and Alferdinck⁴ describe a color editor for simulating a color-deficient view of a color palette. Their editor determines pairs of colors in the palette whose perceived differences in a deficient color space are smaller than a critical threshold. If a difference is too small, the color pair is deemed indistinguishable to a potential color-deficient observer, and a different pair of colors should be used in the design. This editor can assist by selecting a default palette for which all distances meet the threshold criterion.

Daltonization is a procedure for recoloring an image for viewing by a color-deficient viewer. Details of the method are unpublished, but some discussion and an online demonstration are available at <http://www.vischeck.com>. In this method, contrast between red and green hues is stretched and used to modulate luminance and contrast between blue and yellow hues. Three user-specified parameters are required, one for stretching red–green contrast, one for modulation of luminance, and one for modulation of blue–yellow contrast. The online demonstration provides views of the original and recolored images for a simulated deuteranope. There is no automatic determination of optimal values for the required parameters, but three sets of parameters, labeled low, medium, and high (correction) are offered. The results depend highly on the parameters.

Ishikawa et al.⁵ describe manipulation of Web page color for color-deficient viewers. They first decompose the page

into a hierarchy of colored regions. These spatial relations determine important pairs of colors to be modified. An objective function is then chosen to maintain distances between pairs of colors, as well as minimize the extent of color remapping. This attempts to preserve both the original “naturalness” of the colors and the detail in the remapped color image. The objective function is minimized using a genetic algorithm. The authors have extended this method to full-color images.⁶ They first quantize the image to a small number of colors and construct a hierarchy of like-colored regions. As before, a fitness function—designed to preserve detail and minimize the distance between an input color and its corresponding remapped color—is minimized with an evolutionary algorithm.

References

1. G. Meyer and D. Greenberg, “Color-Defective Vision and Computer Graphics Displays,” *IEEE Computer Graphics and Applications*, vol. 8, no. 5, 1988, pp. 28–40.
2. H. Brettel, F. Viénot, and J. Mollon, “Computerized Simulation of Color Appearance for Dichromats,” *J. Optical Soc. of America A*, vol. 14, no. 10, 1997, pp. 2647–2655.
3. S. Kondo, “A Computer Simulation of Anomalous Color Vision,” *Color Vision Deficiencies*, Kugler & Ghedini, 1990, pp. 145–159.
4. J. Walraven and J.W. Alferdinck, “Color Displays for the Color Blind,” *Proc. IS&T and SID 5th Color Imaging Conf.*, Soc. for Imaging Science and Technology, 1997, pp. 17–22.
5. M. Ichikawa et al., *Web-Page Color Modification for Barrier-Free Color Vision with Genetic Algorithm*, LNCS 2724, Springer-Verlag, 2003, pp. 2134–2146.
6. M. Ichikawa et al., “Preliminary Study on Color Modification for Still Images to Realize Barrier-Free Color Vision,” *Proc. IEEE Int’l Conf. Systems, Man, and Cybernetics*, IEEE Press, 2004, pp. 36–41.

photoreceptors that contribute to color vision. *Anomalous trichromatopia* is a condition in which the pigment in one cone is not sufficiently distinct from the others. The viewer still has three distinct spectral sensitivities, but the separation is reduced. *Dichromatopia* results when the viewer has only two distinct pigments in the cones. For both dichromatopia and anomalous trichromatopia, there are three subclassifications depending on which cone has the abnormal pigmentation. Deficiencies in cones sensitive to long wavelengths are referred to as *protanopic*, while deficiencies in those sensitive to medium and short wavelengths are referred to as *deutanopic* or *tritanopic*, respectively. Protanopic and deutanopic deficiencies, the most common forms of color-deficient vision, are characterized by difficulty distinguishing between red and green tones. Tritanopic deficiencies are associated with confusion between blue and yellow tones. *Monochromatism* is another form of color-deficient vision, but it is rare. See the “Simulating Color Deficient Vision” sidebar for more information on human color vision deficiencies.

Monochromatic reproduction

Our fundamental premise dictates that the perceived color difference between any pair of colors should be proportional to their perceived gray difference. In formal terms, for each pair of colors, \mathbf{c}_u and \mathbf{c}_v , we wish to satisfy

$$\frac{\|\mathbf{c}_u - \mathbf{c}_v\|}{C_{\text{range}}} = \frac{\|T(\mathbf{c}_u) - T(\mathbf{c}_v)\|}{T_{\text{range}}}$$

where $T(\mathbf{x})$ is the gray-scale mapping function, C_{range} is the maximum distance between any pair of colors in the image, T_{range} is the maximum distance between any pair of transformed colors, and $\|\cdot\|$ is a perceptual color difference metric. We use the CIE94 color difference,⁸ as it is more efficient to compute than the CIEDE2000 color difference. CIE94 is a weighted Euclidean distance in LCH(ab) (luminance, chroma, hue) color space, which is closely related to CIELab color space:

$$L = L \quad C = \sqrt{a^2 + b^2} \quad H = \tan^{-1}(b/a)$$

Note that gray scale in either CIELab space or LCH(ab)

4 Initial positions: The gray-scale transform found when beginning at the first principal component.



space is given by $\{(L, 0, 0) \mid L \geq 0\}$.

A measure of the global error of the transformation is then

$$\varepsilon^2 = \sum_i \sum_{j=i+1} \left(\frac{\|\mathbf{c}_i - \mathbf{c}_j\|}{C_{\text{range}}} - \frac{\|T(\mathbf{c}_i) - T(\mathbf{c}_j)\|}{T_{\text{range}}} \right)^2 \quad (2)$$

where i and j range over image pixels, rather than distinct colors, to account for the frequency with which each color occurs. This is similar to the error term for multidimensional scaling, except that the differences are normalized over different ranges. Our objective is to minimize this total error.

For transformation to gray scale, we restrict our attention to linear transformations, T , within CIE Lab color space, in which case we determine $T(\mathbf{c}) = (\mathbf{g} \cdot \mathbf{c}, 0, 0)$ for some vector \mathbf{g} . Equation 2 is then

$$\begin{aligned} \varepsilon^2(\mathbf{g}) &= \sum_i \sum_{j=i+1} \left(\frac{\|\mathbf{c}_i - \mathbf{c}_j\|}{C_{\text{range}}} - \frac{\|(\mathbf{g} \cdot (\mathbf{c}_i - \mathbf{c}_j), 0, 0)\|}{T_{\text{range}}} \right)^2 \quad (3) \\ &= \sum_i \sum_{j=i+1} \left(\frac{\|\mathbf{c}_i - \mathbf{c}_j\|}{C_{\text{range}}} - \frac{|\mathbf{g} \cdot (\mathbf{c}_i - \mathbf{c}_j)|}{T_{\text{range}}} \right)^2 \end{aligned}$$

From the CIE94 distance metric and a color image, we can compute C_{range} directly, whereas T_{range} depends on \mathbf{g} . Nevertheless, when producing monochromatic images for a fixed display range, we can set T_{range} to the range of producible gray values.

To determine \mathbf{g} , we minimize the error in Equation 3 using the well-known, Fletcher–Reeves conjugate-gradient method. Such methods only ensure local minima, and so the choice of an initial position can affect the results. We have found, experimentally, that an initial position that selects luminance, $\mathbf{g} = (1, 0, 0)$, usually produces the best results. With this starting position, we

obtained the images of Figure 1c and Figure 2c. We conjecture that viewer expectations for gray-scale images include a bias toward luminance maps and that starting from this position provides the desired contrast without straying too far from expected values.

As an example of an alternative choice, Figure 4 shows, for the color image of Figure 1, the results of using the first principal component as the initial position.

The time required to evaluate Equation 2 for each iteration of the Fletcher–Reeves optimization can become excessive for images with a large number of colors. Our test images often had between 10,000 and 100,000 colors. Nevertheless, we have found that it is not necessary to evaluate the error function over the entire set of colors. We perform a simple equal-volume binning of colors, replacing each bin with its mean, to reduce the number of colors in each image to approximately 1,000. When evaluating Equation 2 with a reduced color image, care must be taken to weight each contribution by the number of colors that map to the reduced \mathbf{c}_i and \mathbf{c}_j .

With this color reduction, the method is relatively fast. On a 2.8-GHz Xeon PC, the optimization of Figure 1a required 2.96 seconds for a reduced color image containing 317 colors. For another version of the same image, containing 11,231 colors, the optimization required 1,931.44 seconds. The resulting gray-scale images for these two tests were visually indistinguishable. Figure 2a contains only 43 colors. Optimization required 0.06 second.

There is not an explicit constraint on the direction of \mathbf{g} . Use of \mathbf{g} or $-\mathbf{g}$ can yield vastly different results. If the wrong sign is chosen, the image will resemble a photographic negative. To prevent this, we check the sign of the L component of final vector \mathbf{g} . If this is negative, we use $-\mathbf{g}$ in the conversion to gray scale. This choice is consistent with our previous conjecture on viewer expectations.

Dichromatic reproduction

We can use the error function of Equation 2 to create a false color image for viewing by a color-deficient observer. The objective here is not to create an image that contains the same level of detail for all viewers—that is, for observers with normal vision and those with various color deficiencies. Rather, we want to create multiple versions of the image, each tailored to the viewer's visual characteristics. Unlike other researchers,^{9,10} we do not constrain how closely a remapped color must match the corresponding original color. Instead, we focus solely on the differences between pairs of colors. Walraven and Alferdinck¹¹ note that for many applications, the differentiation of colors is more important than the identification of colors. For a pair of distinct colors, minimizing the distance between the original and the remapped colors can be detrimental to maintaining adequate contrast. This is clearly the case if the original colors are nearly indistinguishable to the color-deficient viewer. Unlike Daltonization (see the sidebar), our method provides recolored images without user intervention.

For dichromatic reproduction, $\mathcal{T}(\mathbf{x})$ in Equation 2 is a composition of two transformations. First, a linear transformation in homogeneous CIE Lab coordinates, \mathcal{G} :

$\mathbb{R}^4 \rightarrow \mathbb{R}^3$, is applied to warp the original color distribution. Note that G is

$$G = \begin{pmatrix} g_1 & g_2 & g_3 & g_4 \\ g_5 & g_6 & g_7 & g_8 \\ g_9 & g_{10} & g_{11} & g_{12} \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

and so we ultimately search a 12-dimensional space. Nonzero values of $g_4, g_8,$ and g_{12} allow for color translations. The warped colors are then input to a simulator of a color-deficient viewer. From the simulator output, we can compute the perceived distance between the simulated deficient pairs of colors as before.

To simulate a color-deficient viewer, we followed Meyer and Greenberg's¹² procedure. They note that empirical studies on individuals who have normal vision in one eye and color-deficient vision in the other suggest that the 2D hue values of normal vision are reduced to a single dimension that is linear in terms of the CIE Uniform Chromaticity Scale coordinates:

$$u = \frac{4X}{X+15Y+3Z} \quad v = \frac{9Y}{X+15Y+3Z} \quad (4)$$

For each type of deficiency, a pair of line segments in uv space is identified, where each pair connects a reference white value to two specific spectral energies, shown in Table 1. To determine where, on such line segments, a color-deficient viewer will perceive any specific color, Meyer and Greenberg use another color space, called SML, so named because the three components correspond to cone sensitivities in the visible spectrum's short, medium, and long wavelength regions. They provide a linear transformation between CIE XYZ space and SML space, given by

$$\begin{bmatrix} S \\ M \\ L \end{bmatrix} = \begin{bmatrix} 0.0000 & 0.0000 & 0.5609 \\ -0.4227 & 1.1723 & 0.0911 \\ 0.1150 & 0.9364 & -0.0203 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (5)$$

We can then express color deficiency as an inability to recognize differences in one of these components. For example, viewers with protanopic deficiency would be unable to distinguish variations in the L component, and thus a line in SML color space determined by fixed values of S and M with varying values of L would represent a confusion line for such viewers.

To obtain the perceived color corresponding to any original color in RGB or CIE Lab or LCH(ab) space, the original color is first converted to CIE XYZ space and then mapped to SML space by Equation 5. This determines, per deficiency, a specific confusion line in SML space. By applying the inverse of the transformation (see Equation 5), we can then project the confusion line into uv space (see Equation 4) and find its intersection with one of the line segments, previously described, appropriate for this type of deficiency. The resulting uv coordinates of the intersection, together with the luminance of the original color, suffice to construct the perceived color.

Table 1. Spectral energies.

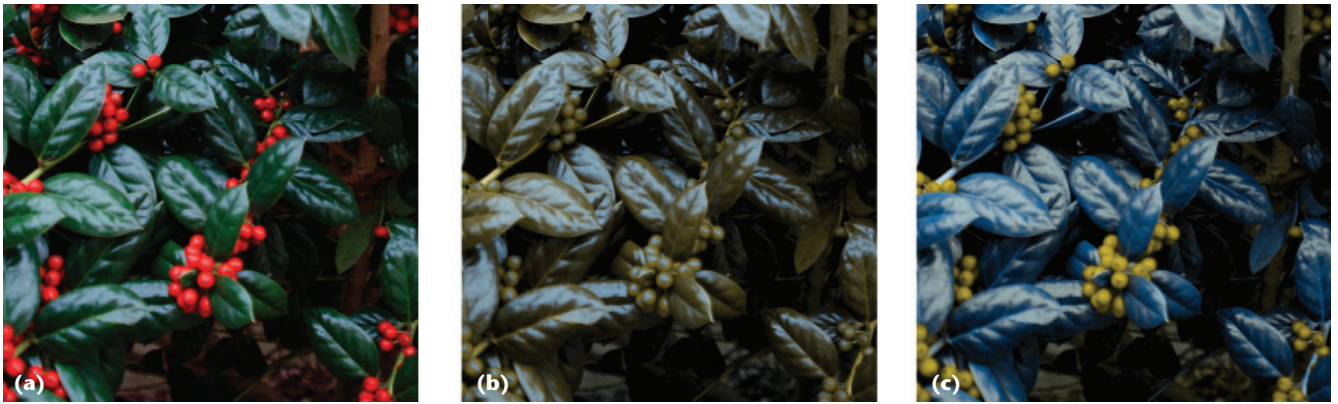
Deficiency	Energy 1 (nanometers)	Energy 2 (nanometers)
Protanopic	473	574
Deutanopic	477	578
Tritanopic	490	610



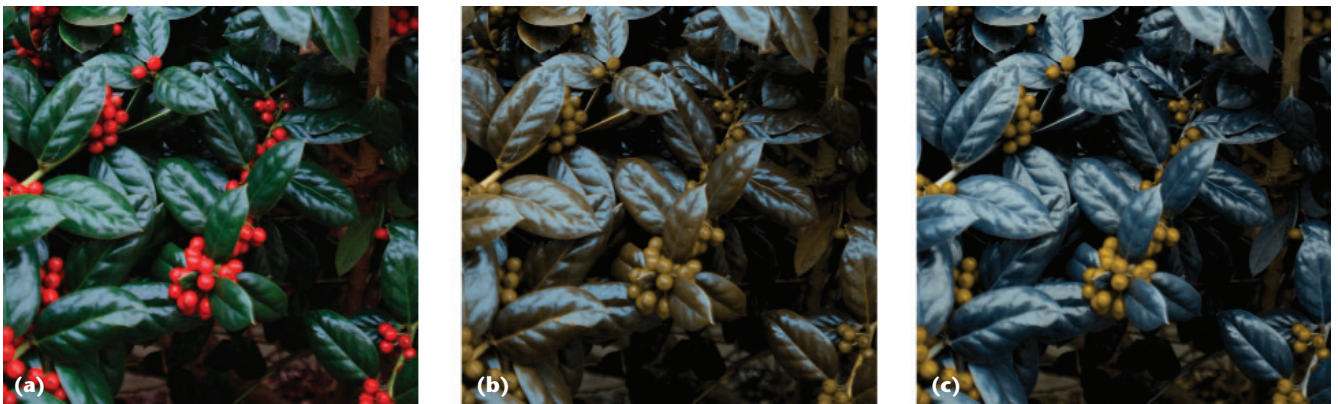
5 Confused CIE Lab gamut: CIE Lab chromaticity with perceived chromaticity plots for a simulated protanopic viewer (red), deutanopic viewer (green), and tritanopic viewer (blue).

Once again, to apply a conjugate-gradient search for an optimal value of the matrix G , we must supply an initial position for G . In Figure 5 we show a slice of CIE Lab space determined by fixing the L component. The a -axis is horizontal, and the b -axis is vertical. If we apply a color-deficient simulation to all colors shown, the 2D space collapses, in each case, to a 1D curve. The red curve shows the collapsed chromaticity for a simulated protanopic viewer, the green curve for a simulated deutanopic viewer, and the blue curve for a simulated tritanopic viewer. As expected, the protanopic and deutanopic viewers have an extremely small response in the a direction (horizontal axis) while the tritanopic viewer has a narrow response in the b direction (vertical axis). Based on this observation, we select an initial transformation, G , that maps the first principal component of the target image to b and the second principal component to a for the protanopic and deutanopic cases. The opposite mapping is selected for the tritanopic case. The L axis is mapped to itself. As in the transformation to gray scale, we then apply the Fletcher-Reeves conjugate-gradient method to Equation 2 to obtain an optimal G and therefore an optimal recoloring transformation, T .

In Figure 6 (next page), we show the results of applying this method to the image of Figure 1 for the case of protanopic deficiency. Figure 6a is the original, Figure 6b is the original as seen by a simulated observer with protanopic deficiency, and Figure 6c is the original after recoloring by T , also as seen by a simulated observer with protanopic deficiency. We see that the contrast between the berries and the leaves in Figure 6a—which is lost in



6 Protanopic deficiencies: (a) The color image from Figure 1, (b) as seen by a simulated protanopic viewer, and (c) recolored for a protanopic viewer as seen by a simulated protanopic viewer.



7 Deuteranopic deficiencies: (a) The color image from Figure 1, (b) as seen by a simulated deuteranopic viewer, and (c) recolored for a deuteranopic viewer as seen by a simulated deuteranopic viewer.

Figure 6b—is preserved in Figure 6c, the recolored image. We can choose T_{range} to span the distance of the color deficient gamut and maximize the range of colors available to reproduce the contrast of the original image.

In Figure 7, we show similar results for a viewer with deuteranopic deficiencies. The only input changes to our method include using a projected confusion line based on an inability to distinguish the M component of SML space and using the second row of Table 1 to determine the deuteranopic gamut. Again, we see that the color contrast between the berries and the leaves in Figure 7a—which is lost in Figure 7b—is preserved in Figure 7c, the recolored image.

Tritanopic deficiencies are associated with confusion between blue and yellow tones, rather than red and green tones, and as such, the color image of Figure 1 does not provide a good test case for our algorithm. (All variants are nearly indistinguishable from the original.) Instead, we select an image with significant blue-yellow information content. Figure 8 (next page) shows the results. As expected, here we use a confusion line determined by an inability to distinguish the S component in SML space and the third row of Table 1 to determine the tritanopic gamut.

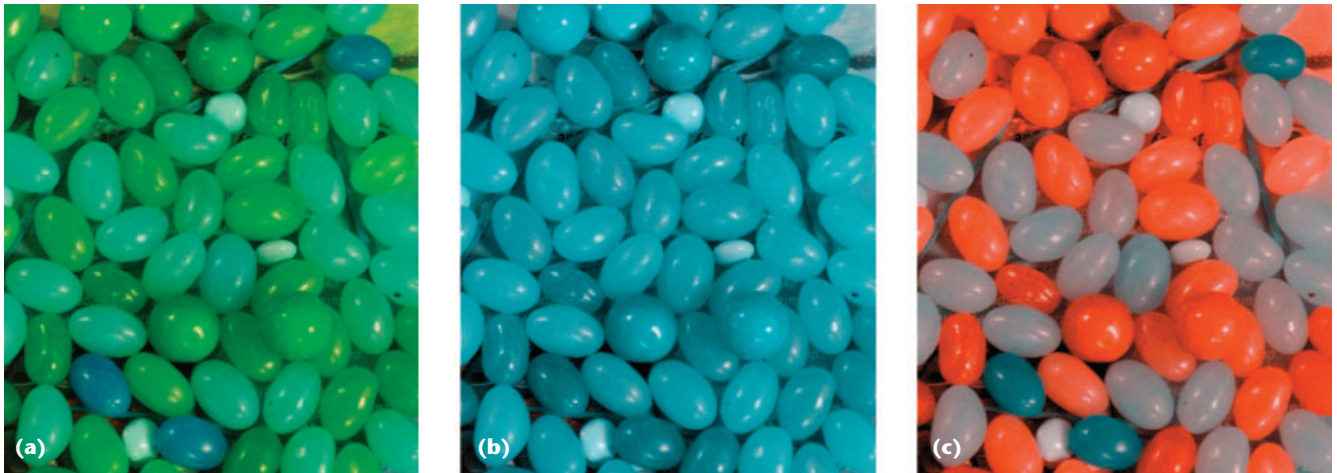
Like the monochromatic case, the time required for the Fletcher-Reeves optimization of Equation 2 in the dichromatic case can become excessive for images with large

numbers of colors. We use the same color reduction technique as a preprocessing step. For the 317-color image of Figure 6, optimization required 6.95 seconds on a 2.8-GHz Xeon. The 11,231-color version of the same image required 8,651.34 seconds. For Figure 7, the 317-color image required 6.91 seconds. The 11,231-color version required 8,617.85 seconds. We optimized a quantized version of the color image in Figure 8, containing 233 colors, in 23.51 seconds. A 6,515-color version of this image required 11,432.86 seconds. In all cases, the resulting pairs of images were visually indistinguishable.

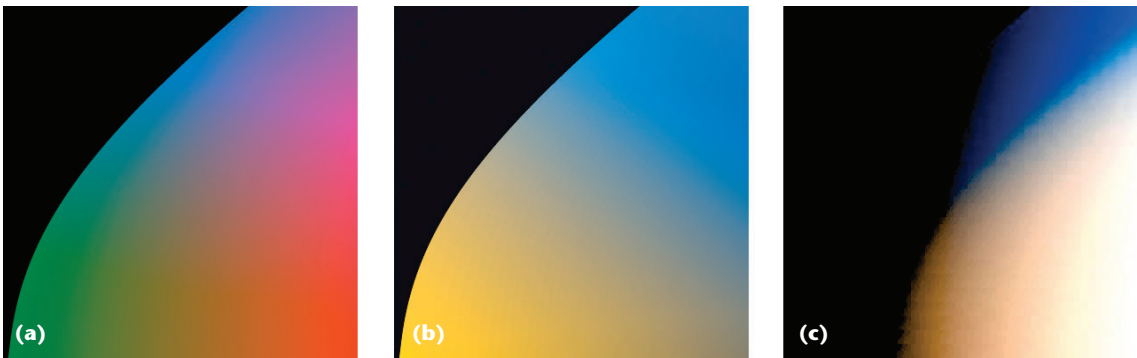
Direct comparison of our method with Daltonization is difficult in that, as noted in the “Simulating Color Deficient Vision” sidebar, the latter requires user-supplied parameters. For most images, parameters that yield recolored images comparable to those produced by our method can be found with modest effort. Nevertheless, there are cases for which suitable parameter choices are more difficult to find. In Figure 9 (next page) we show such a case. The Daltonized version selected, that with low correction, suffers from a loss of contrast along the curved edge.

Discussion

Our linear method for both monochromatic and dichromatic color image reduction has several advantages. First,



8 Tritanopic deficiencies: (a) A color image with significant blue-yellow content, (b) as seen by a simulated tritanopic viewer, and (c) recolored for a tritanopic viewer as seen by a simulated tritanopic viewer.



9 Comparison with Daltonization: (a) A multigradient color image, (b) recolored by our method for a deuteranopic viewer as seen by a simulated deuteranopic viewer. (c) Daltonized with low correction for a deuteranopic viewer as seen by a simulated deuteranopic viewer.

we can use color quantization to quickly and easily reduce the size of the data set. It is not necessary to operate on the entire data set, as it is with traditional nonlinear methods. Nevertheless, work is progressing to extend procedures such as local linear embedding (LLE) and Isomap to operate on partial data sets. Second, the linear operations are extremely fast to compute and easy to implement, as opposed to procedures, such as LLE, that require finding eigenvectors of very large, sparse matrices. It is also straightforward to incorporate alternative perceptual distance metrics. Finally, our linear method does not require tuning, in the form of a neighborhood-size parameter, which frequently appears in nonlinear methods.

In some cases, our method produces results of questionable value. If we apply the gray-scale transform to the CIE Lab gamut of Figure 5, the result is a smooth, gray ramp from the lower left to upper right of the image. The cyan corner is mapped to white, and the orange corner is mapped to black, but the green and magenta corners receive almost identical grays. It is impossible for a linear gray scale transform to handle multiple modes of contrasting gradients in the image. Nevertheless, this is a fairly contrived example, and it is unclear what a pleasing gray-scale version of this image would look like.

The linear gray-scale transform also breaks down

when the image contains large patches of black and white. Here the gray-scale transform maps black to low luminance and white to high luminance, resulting in a direct mapping from L to gray. To combat this problem, it is straightforward to allow the user to mask off regions of the image containing relevant color details.

Our method does not explicitly account for the extent to which the final gray-scale or recolored images differ from the originals. In the gray-scale transformation, we implicitly handle this by initializing the optimization with a gray vector pointing along the luminance axis. A generalization to the dichromatic case is still an open problem. This can result in extremely unnatural colors in the image, such as the blue leaves in Figure 6c, or the orange jelly beans in Figure 8c. There is also no consideration given to spatial surround or local color adaption. A colored box on a light surround and the same box on a dark surround are treated the same, even though they may be perceived differently. This would be useful to include for computing images for display on specific media. Finally, the perceptual color difference metrics we use are designed for cases where large areas of color are being compared to one another. There may be a more meaningful way to compare perceived color differences in high color frequency cases.

Future directions

We are currently examining the behavior of our conversion algorithm with respect to the luminance of the original image. We might need to introduce constraints on the ordering of some luminance values to preserve effects such as shadow gradients and specular highlights. Investigation into nonlinear transformations may also prove useful. While they are more difficult to use in conjunction with color quantization, they are better suited for handling complex variation, such as in Figure 5. ■

Acknowledgment

This work was supported in part by the Engineering Research Center Program of the US National Science Foundation under award EEC-9731680 and the Information Technology Research Program of the National Science Foundation under award ACI-0113139.

References

1. G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd ed., John Wiley & Sons, 2000.
2. J. Tumblin and H. Rushmeier, "Tone Reproduction for Realistic Images," *IEEE Computer Graphics and Applications*, vol. 13, no. 6, 1993, pp. 42-48.
3. R. Floyd and L. Steinberg, "An Adaptive Algorithm for Spatial Grey Scale," *Proc. Soc. for Information Display (SID 17)*, Soc. for Information Display, 1976, pp. 75-77.
4. E. Stollnitz, V. Ostromoukhov, and D. Salesin, "Reproducing Color Images Using Custom Inks," *Proc. ACM Siggraph*, ACM Press, 1998, pp. 267-274.
5. S. Roweis and L. Saul, "Nonlinear Dimensionality Reduction by Locally Linear Embedding," *Science*, vol. 290, no. 5500, 2000, pp. 2323-2326.
6. J. Tenenbaum, V. de Silva, and J. Langford, "A Global Geometric Framework for Nonlinear Dimensionality Reduction," *Science*, vol. 290, no. 5500, 2000, pp. 2319-2323.
7. B. Wandell, *Foundations of Vision*, Sinauer Associates, 1995.
8. *Industrial Colour-Difference Evaluation*, Commission Internationale de l'Eclairage (CIE), tech. report 116-1995, 1995.
9. M. Ichikawa et al., *Web-Page Color Modification for Barrier-Free Color Vision with Genetic Algorithm*, LNCS 2724, Springer-Verlag, 2003, pp. 2134-2146.
10. M. Ichikawa et al., "Preliminary Study on Color Modification for Still Images to Realize Barrier-Free Color Vision," *Proc. IEEE Int'l Conf. Systems, Man, and Cybernetics*, IEEE Press, 2004, pp. 36-41.
11. J. Walraven and J.W. Alferdinck, "Color Displays for the Color Blind," *Proc. IS&T and SID 5th Color Imaging Conf.*, Soc. for Imaging Science and Technology, 1997, pp. 17-22.
12. G. Meyer and D. Greenberg, "Color-Defective Vision and Computer Graphics Displays," *IEEE Computer Graphics and Applications*, vol. 8, no. 5, 1988, pp. 28-40.



Karl Rasche is a PhD candidate in computer science at Clemson University. His research interests include color reproduction and rendering techniques. Rasche received a BS in computer science from Hope College. Contact him at rkarl@cs.clemson.edu.



Robert Geist is a professor of computer science at Clemson University. His research interests include distributed rendering, rendering participating media, and computer graphics as a vehicle for instruction in the undergraduate computer science curriculum. Geist received a PhD in mathematics from the University of Notre Dame. Contact him at rmg@cs.clemson.edu.



James Westall is a professor of computer science at Clemson University. His research interests include distributed rendering and computer network performance analysis. Westall received a PhD in mathematics from the University of North Carolina at Chapel Hill. Contact him at westall@cs.clemson.edu.

For further information on this or any other computing topic, please visit our Digital Library at <http://www.computer.org/publications/dlib>.