

Reproducing Color Images as Duotones*

Joanna L. Power[†] Brad S. West Eric J. Stollnitz David H. Salesin

University of Washington

Abstract

We investigate a new approach for reproducing color images. Rather than mapping the colors in an image onto the gamut of colors that can be printed with cyan, magenta, yellow, and black inks, we choose the set of printing inks for the particular image being reproduced. In this paper, we look at the special case of selecting inks for duotone printing, a relatively inexpensive process in which just two inks are used. Specifically, the system we describe takes an image as input, and allows a user to select 0, 1, or 2 inks. It then chooses the remaining ink or inks so as to reproduce the image as accurately as possible and produces the appropriate color separations automatically.

CR Categories: I.3.4 [Computer Graphics]: Graphics Utilities

Additional Keywords: color reproduction, color printing, duotone, gamut mapping, Neugebauer model

1 Introduction

Modern color reproduction typically employs a fixed set of *process-color inks*: cyan, magenta, yellow, and sometimes black. Placed on top of one another and in juxtaposition, these inks can be used to reproduce a range of colors, called their *gamut*. Although the standard process colors were carefully chosen to provide a relatively large gamut, this gamut is nevertheless quite limited when compared to the full range of colors visible to the human eye. Thus, color fidelity must generally be compromised when reproducing images with process colors (or any other small, fixed set of inks).

A number of approaches have been suggested for mitigating this problem, including methods for smoothly mapping the original image colors to the process color gamut [21], and, in the case of three-dimensional computer-synthesized imagery, redefining the original object colors so that the rendered image will lie inside the process-color gamut [5]. However, to our knowledge, there has been no research to date on a very different but equally promising approach: allowing the inks used for printing to be selected for the particular image being reproduced.

In this paper, we investigate a first step toward this goal: selecting inks for *duotone* color reproduction, a less expensive printing process in which just two inks are used. In particular, the problem we address can be stated formally as follows:

Given: A color image C , a paper color, the set of all available printing inks \mathcal{I} , and a subset of $k = 0, 1, \text{ or } 2$ inks in \mathcal{I} chosen by the user.

Find: The best $2 - k$ additional inks for reproducing C on the selected paper, along with the appropriate color separations.

While duotones obviously have more limited color gamuts than three- and four-ink processes, they can nevertheless be used to reproduce a surprising range of color—especially when the inks are chosen for the particular image being reproduced, as we show in this paper. Indeed, a duotone print is really a combination of four colors: the paper color alone, the colors of each of the inks individually, and the color of the two inks superimposed on each another. These four colors define a bilinear surface in color space that may span a broad range of the full color gamut.

Whereas duotone printing has traditionally been used almost exclusively to enhance monochrome gray-scale images with a tint of color, the kind of duotone color reproduction we explore in this paper has a variety of new applications. First, duotone printing is significantly less expensive than process color, typically about two-thirds the cost. Thus, there is a clear economic advantage in using duotones for images that are adequately reproduced in this way. Second, the general form of the problem expressed above allows a user to select one or both of the two printing inks. This formulation is useful when other requirements of the page design, such as matching the precise colors of a company logo, already constrain the choice of inks. Also, because a large number of printing presses are two-color presses, printed documents are often designed for just two inks, generally black (for text) and one additional color. In these cases, we show how duotone separations can still be computed to match the colors of an image as well as possible with the inks available. Finally, our formulation also allows duotone separations to be computed for a colored paper, which may be useful in a variety of situations. As one example, this form of the problem could be useful for creating duotone separations of full-color images for a “yellow pages” telephone directory.

1.1 Related work

To our knowledge, there has been no previous work that proposes a general approach to reproducing images using duotones. The most closely related work in the field of computer graphics addresses the problem of gamut mapping, or smoothly mapping the colors of an original image to those available on an output device. Fundamentally, a duotone gamut is much more restricted than the gamut of a typical output device. Our duotone mapping differs from the approaches taken by Stone *et al.* [21] and Gentile and Allebach [3] primarily in that we require a mapping from three dimensions to two. Harrington *et al.* [8] describe a technique for creating a “highlight color image,” a specialized duotone in which one ink is black.

There are a few articles in the optical engineering literature on calculating halftone separations for inks other than the standard four-color process inks. Marcu and colleagues [11, 14] describe how to compute separations when printing with an arbitrary number of inks. However, the inks and the order in which they will be printed must be specified by the user. Furthermore, Marcu *et al.* do not specify

*Joanna L. Power, Brad S. West, Eric J. Stollnitz, and David H. Salesin. Reproducing color images as duotones. In *Proceedings of SIGGRAPH 96*. ACM, New York, 1996 (to appear).

[†]Department of Computer Science and Engineering, University of Washington, Box 352350, Seattle, WA 98195-2350.

how to handle a color that is out of gamut (the common case for duotone mapping). Ostromoukhov [16] describes how the traditional process-color printing gamut can be extended by introducing additional basic colors.

In the printing industry, a number of empirical studies have developed improved models for halftone color reproduction [1, 7, 19]. Instead of complicating our duotone mapping with the correction factors described in these works, we adjust the output of the model according to our own empirically determined correction factor. Kang [12] takes a similar approach, though he assumes that each ink can be corrected independently, while we adjust the duotone gamut as a whole.

1.2 Overview of paper

The remainder of the paper is organized as follows: Section 2 provides relevant background information on color, color printing, and the Neugebauer model of halftone printing. Section 3 describes the duotone mapping in detail, and Section 4 discusses the ink-selection process. Section 5 describes some variations on duotone printing that are easily accommodated by our algorithm. Section 6 presents some actual duotones as well as some rough timing results. We conclude in Section 7 with a discussion of possible directions for future work.

2 Background

Before we dive into the details of our duotone mapping, we present some background material on color and color spaces, color halftone printing, and the Neugebauer halftone model.

2.1 Color and color spaces

Color is determined by the intensity of light in the range of visible wavelengths from about 400nm to 700nm. According to the tristimulus theory of color perception [9], all colors can be reproduced using combinations of three primary wavelengths (roughly corresponding to red, green, and blue). Thus, color can be expressed as a function of wavelength, known as a *spectral reflectance*, or as a three-dimensional quantity. More information about color can be found in standard graphics texts [2, 4].

The *XYZ* color space was developed in 1931 by the Commission Internationale de l'Éclairage (CIE) to standardize color specification. The *XYZ* color space is additive, meaning that the color resulting from the superposition of two colored light sources can be calculated by simply adding the coefficients of the two known colors. A spectral reflectance can be converted to *XYZ* coordinates by integrating the spectral information against three functions \bar{x} , \bar{y} and \bar{z} [2, pages 579–580]. The computer graphics community is more familiar with the *RGB* color space, an additive color space that is device-dependent. Conversion between *RGB* and *XYZ* coordinates can be accomplished by a linear transform if the *XYZ* coordinates of the device's red, green and blue primaries are known [2, pages 585–587].

In contrast to additive color spaces, perceptually uniform color spaces allow the difference between two colors (as perceived by the human eye) to be measured as the distance between points. For example, two colors c_1 and c_2 separated by some distance d in a perceptually uniform color space appear about as different as two other colors c_3 and c_4 separated by the same distance d . The CIE developed two perceptually uniform spaces: $L^*a^*b^*$ and $L^*u^*v^*$. Both color spaces require the definition of a *reference white*, which is usually taken to be a standard light source defined by the CIE. In both spaces, L^* indicates brightness and has a value of 100 for reference white. Though neither $L^*a^*b^*$ nor $L^*u^*v^*$ is perfectly perceptually

uniform, both come close to satisfying the condition that colors separated by the same distance appear equally similar [4, pages 59–66].

2.2 Color halftone printing

In color halftone printing, a continuous-tone image is reproduced by printing a number of versions of the image atop one another. Each version, known as a *halftone separation*, consists of various sized dots of a single ink. Color halftone printing differs from color dithering on monitors in that subtractive effects as well as additive effects play a role. The subtractive effect of superimposing dots of different color produces the set of *printing primaries* for a particular set of inks. For example, for cyan, magenta, and yellow ink printed on white paper, the set of printing primaries is cyan, magenta, yellow, blue (cyan + magenta), green (cyan + yellow), red (magenta + yellow), black (cyan + magenta + yellow), and white (no ink). The additive effect of juxtaposing dots of different sizes produces the entire set, or *gamut*, of colors that can be achieved by printing halftone separations using a particular set of inks. Figure 1 illustrates the reproduction of a color image using cyan, magenta, yellow, and black inks. More information on color reproduction can be found in the classic texts by Hunt [10] and Yule [23].

2.3 Neugebauer halftone model

In 1937, Neugebauer developed a series of equations that, given ink and paper colors, describe the amount of each ink needed to reproduce a given color [15]. Intuitively, the model says that the overall color of a small area is a weighted average of the printing primaries, with each primary weighted by the relative area it covers. For example, in a square printed with cyan, magenta and yellow ink on white paper, the contribution of blue is given by the fraction of the square that is covered by cyan and magenta but not yellow. If α_1 , α_2 , and α_3 are the amounts of cyan, magenta and yellow ink printed, then the contribution of blue is $\alpha_1\alpha_2(1 - \alpha_3)$.

The “Neugebauer equations” express colors in terms of their coordinates in the *XYZ* color space. The model was originally designed to describe three-color printing, though it can be generalized to handle any number of inks. Let g_0 be the color of the paper, g_i the color of ink i on the paper, $g_{i,j}$ the color of inks i and j superimposed on the paper, $g_{i,j,k}$ the color of all three inks i, j and k superimposed on the paper, and α_i the amount of ink i (between 0 and 1). For three inks, the Neugebauer model describes c , a color in the printing gamut, in terms of the eight printing primaries and the amounts of the three inks required to achieve c :

$$c = \begin{bmatrix} g_0 & g_1 & g_2 & g_3 & g_{1,2} & g_{1,3} & g_{2,3} & g_{1,2,3} \end{bmatrix} \begin{bmatrix} (1 - \alpha_1)(1 - \alpha_2)(1 - \alpha_3) \\ \alpha_1(1 - \alpha_2)(1 - \alpha_3) \\ (1 - \alpha_1)\alpha_2(1 - \alpha_3) \\ (1 - \alpha_1)(1 - \alpha_2)\alpha_3 \\ \alpha_1\alpha_2(1 - \alpha_3) \\ \alpha_1(1 - \alpha_2)\alpha_3 \\ (1 - \alpha_1)\alpha_2\alpha_3 \\ \alpha_1\alpha_2\alpha_3 \end{bmatrix}$$

3 Duotone mapping

We are now prepared to tackle the following problem: Given 0, 1, or 2 inks out of a set of available inks, choose the remaining 0, 1, or 2 inks and compute the separations that most accurately reproduce a given image. In this section we present the duotone mapping, our technique for computing the separations for an image once the two inks are known. In Section 4 we will describe the process of choosing inks.

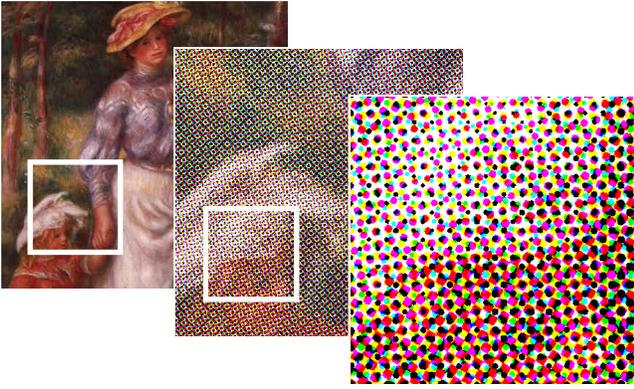


Figure 1 Color halftoning of an image.

3.1 Neugebauer model for duotone printing

For two-color printing, the Neugebauer model describes c , a color in the printing gamut, in terms of the four printing primaries and the amounts of the two inks:

$$c = \begin{bmatrix} g_0 & g_1 & g_2 & g_{1,2} \end{bmatrix} \begin{bmatrix} (1 - \alpha_1)(1 - \alpha_2) \\ \alpha_1(1 - \alpha_2) \\ (1 - \alpha_1)\alpha_2 \\ \alpha_1\alpha_2 \end{bmatrix}$$

The gamut described by the above equation is a bilinear surface in an additive color space such as XYZ or RGB , as shown in Figure 2.

Finding the duotone gamut

The duotone gamut is fully specified by the Neugebauer model given the printing primaries g_0 , g_1 , g_2 and $g_{1,2}$. We obtained spectral reflectance data for inks and papers using a Colortron™ spectrophotometer. The Colortron™ software also provides data for PANTONE® ink sets. In order to find the printing primaries, we must estimate the color of inks printed on the selected paper. We must also estimate the color of two inks superimposed on the selected paper.

We used a simple model to approximate the effect of printing ink on paper. The data for a layer of ink or paper can be expressed in two parts: an overall reflectance spectrum R and a Fresnel reflectance spectrum F . The overall reflectance spectrum indicates how much light of each wavelength is reflected back by the layer and can be directly measured with a spectrophotometer. The Fresnel information indicates how much light of each wavelength bounces off the surface without entering the layer.

Inks act very much like filters, which are purely subtractive layers. A paper acts like an opaque layer. This simple behavior allows us to approximate layer composition by multiplying reflectance spectra and adjusting for Fresnel effects [6]. To approximate the result of superimposing a subtractive layer i on another layer j , we remove the Fresnel component from each layer's overall reflectance, multiply the altered spectra, and add back the Fresnel reflectance of the top layer:

$$\begin{aligned} R_{i,j} &= (R_i - F_i)(R_j - F_j) + F_i \\ F_{i,j} &= F_i \end{aligned}$$

Since we cannot directly measure the Fresnel component of reflectance, we approximate the Fresnel spectrum of an ink by the reflectance spectrum of black ink. The intuition behind this approximation is that black absorbs nearly all incident light; thus, any reflectance from a layer of black ink is primarily due to a Fresnel effect

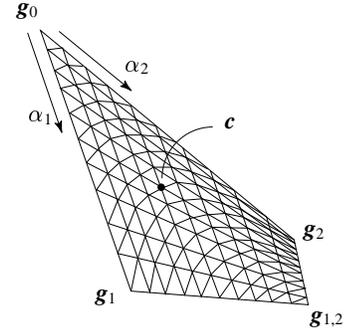


Figure 2 Bilinear surface described by the Neugebauer half-tone model.

at the surface. For simplicity, we use the same Fresnel spectrum for paper, and we ignore any Fresnel effect between ink and paper.

The layering model described above does not completely solve our problem, as the measured spectral reflectance data for an ink includes information about the stock paper on which the ink was originally printed. To compute the result of removing one layer from another, we simply invert the above equations:

$$\begin{aligned} F_i &= F_{i,j} \\ R_i &= \frac{(R_{i,j} - F_{i,j})}{(R_j - F_j)} + F_i \end{aligned}$$

Division by zero does not arise in practice, as we only remove inks from papers, and the papers have high overall reflectance spectra and low Fresnel components. Once we have the spectral reflectance information for the paper, the individual inks on paper, and the combination of both inks on paper, we can find the printing primaries g_0 , g_1 , g_2 , and $g_{1,2}$ by first converting the appropriate spectral quantities to XYZ or RGB coordinates [2, pages 579–587].

3.2 Goals of duotone mapping

The core of our algorithm is the mapping that transforms image colors onto the duotone gamut. There are many characteristics of colors and of sets of colors that one could seek to preserve in designing a mapping. The fundamental tradeoff is between mapping colors exactly and maintaining overall relationships between colors. Our approach places more importance on overall relationships than on exact matches, and weighs certain relationships more heavily than others.

The basic idea behind our mapping technique is to define an orthogonal axis system for each duotone gamut, then transform the image colors along two directions and use parallel projection along the third. The choice of directions is clearly critical to the effectiveness of the mapping and corresponds to preserving certain relationships between image colors at the expense of other relationships. Here are the choices we made:

1. **Preserve relative luminance.** The eye is more sensitive to light at some wavelengths than others. The *luminance-efficiency curve*, which describes the relationship between wavelength and visual sensitivity, corresponds to a direction in a three-dimensional color space. Preserving separation of image colors along this direction of greatest sensitivity, called the *luminance direction*, is the primary goal of our duotone mapping. The Y axis of the XYZ color space corresponds exactly to the luminance direction [2], and therefore we take Y to be the direction of our first transformation. (Stone *et al.* also used preservation of relative luminance in their gamut mapping work [21].)
2. **Preserve ink-spread separation.** The second relationship our

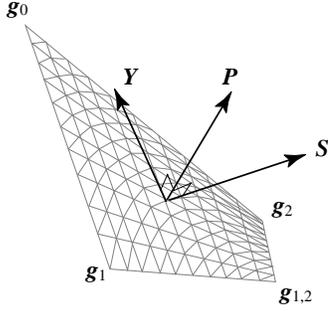


Figure 3 Axes associated with a typical duotone gamut.

mapping preserves is separation in the direction of most color variation on the duotone gamut. The curve on the gamut between the two individual ink primaries describes the widest variation in color achievable with the selected paper and inks. Thus, the vector $g_2 - g_1$ is the ideal direction for the second transformation. However, our duotone mapping requires an orthogonal axis system, and we have already chosen the Y axis. Therefore we orthogonalize $g_2 - g_1$ with respect to Y and use the resulting *ink-spread direction* S as the direction of the second transformation.

3. **Sacrifice normal separation.** Separation must be sacrificed in some direction in order to map points from three-dimensional space onto a surface. As we have chosen two axes and require mutual orthogonality, the *projection direction* P is already defined by $Y \times S$. The P axis approximates the average normal of the bilinear surface, which is the direction of least color variation on the gamut. It is therefore a good direction in which to sacrifice separation.

Given a duotone gamut, the constraints listed above completely define the orthogonal axis system we will use for our duotone mapping. A typical example is illustrated in Figure 3.

3.3 Computing the duotone mapping

The duotone mapping takes image colors c_1, \dots, c_n and maps them to colors $\hat{c}_1, \dots, \hat{c}_n$ on the duotone gamut. The mapping of image color c_i takes place in three steps, with each step affecting a different orthogonal component of c_i . We will denote the components as follows:

$$\begin{aligned} c_i^Y &\equiv c_i \cdot Y \\ c_i^S &\equiv c_i \cdot S \\ c_i^P &\equiv c_i \cdot P \end{aligned}$$

We will use the same superscript notation to indicate the orthogonal components of the printing primaries g_0, g_1, g_2 and $g_{1,2}$.

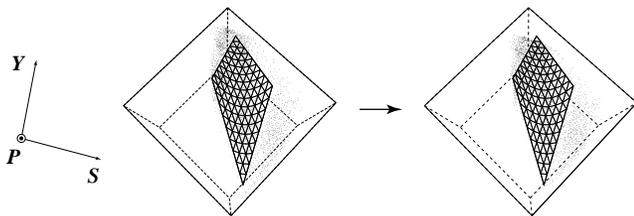


Figure 4 Image colors before and after a uniform transformation in the luminance direction Y .

3.3.1 Calculating the luminance transformation

The first transformation applied to the image colors is a mapping along Y to bring all colors within the luminance range of the current duotone gamut. Although this mapping could be any monotonically increasing function, for the sake of simplicity we use a linear function. If we let c_i^Y be the y -value of the original image color c_i , and \hat{c}_i^Y the y -value of the transformed color, the luminance transformation can be written as follows:

$$\begin{aligned} y_{\min} &= \min_i \{c_i^Y\} \\ y_{\max} &= \max_i \{c_i^Y\} \\ \hat{y}_{\min} &= \max\{y_{\min}, \min\{g_0^Y, g_1^Y, g_2^Y, g_{1,2}^Y\}\} \\ \hat{y}_{\max} &= \min\{y_{\max}, \max\{g_0^Y, g_1^Y, g_2^Y, g_{1,2}^Y\}\} \\ \hat{c}_i^Y &= \hat{y}_{\min} + \left(\frac{c_i^Y - y_{\min}}{y_{\max} - y_{\min}} \right) (\hat{y}_{\max} - \hat{y}_{\min}) \end{aligned}$$

Figure 4 shows a set of image colors before and after their luminance values have been transformed to lie within the range of luminance values available in a typical duotone gamut.

3.3.2 Calculating the ink-spread transformation

The second transformation is along the ink-spread axis S . This transformation depends upon luminance, since at some luminance values the duotone gamut is wide and at others it consists of a single point. Consider a particular luminance value \hat{y} between \hat{y}_{\min} and \hat{y}_{\max} . We define \hat{s}_0 and \hat{s}_1 to be the s -values of the points found by intersecting the edges of the duotone gamut with a plane of constant luminance \hat{y} . The ink-spread transformation at luminance value \hat{y} brings the s -values of all image colors with luminance value \hat{y} into the range $[\hat{s}_0, \hat{s}_1]$. Figure 5 illustrates the effect of an ink-spread transformation on a set of image colors.

In our current implementation, the non-uniformity along Y is handled by separating colors into bins according to y -value and calculating a different transformation for each bin. The coherence among colors in most natural images prevents this discrete approach from introducing noticeable discontinuities into a duotone.

The ink-spread transformation, like the luminance transformation, can be any monotonically increasing function that maps the s -values of image colors to the s -values available on the gamut. In what follows, we present both a simple linear mapping and a more complex mapping based on Bézier curves.

Linear mapping

The simplest transformation of s -values is a linear mapping, which we can define in a manner similar to the luminance transformation. If we consider only colors within a particular luminance bin, the lin-

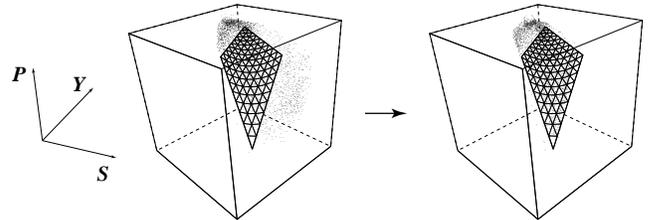


Figure 5 Image colors before and after a non-uniform transformation in the ink-spread direction S .

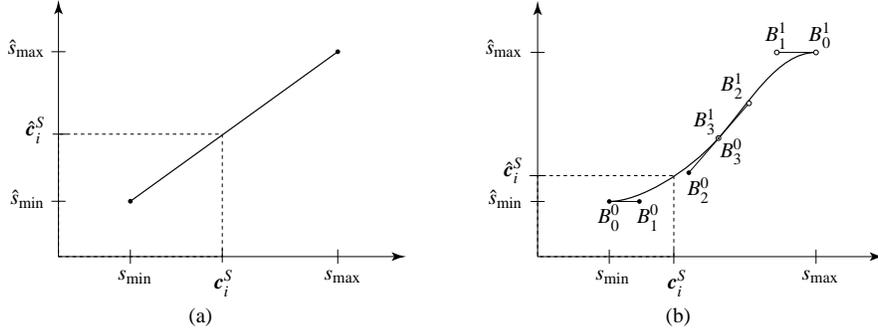


Figure 6 Two possible ink-spread transformations: (a) a linear mapping; (b) a Bézier curve mapping.

ear mapping is given by:

$$\begin{aligned}
 s_{\min} &= \min_i \{c_i^S\} \\
 s_{\max} &= \max_i \{c_i^S\} \\
 \hat{s}_{\min} &= \max\{s_{\min}, \min\{\hat{s}_0, \hat{s}_1\}\} \\
 \hat{s}_{\max} &= \min\{s_{\max}, \max\{\hat{s}_0, \hat{s}_1\}\} \\
 \hat{c}_i^S &= \hat{s}_{\min} + \left(\frac{c_i^S - s_{\min}}{s_{\max} - s_{\min}} \right) (\hat{s}_{\max} - \hat{s}_{\min})
 \end{aligned}$$

An example of a linear ink-spread transformation for a single luminance bin is illustrated in Figure 6(a).

Bézier-curve mapping

The linear mapping given above does not attempt to preserve the s -values of image colors, despite the fact that many such values may be available in the duotone gamut. Instead, the linear mapping preserves the relationships between s -values of the image colors. At the other extreme, we could map each s -value to the closest value in the interval of available values $[\hat{s}_0, \hat{s}_1]$, thereby accurately reproducing some colors while clamping others to the interval endpoints.

We are interested in obtaining some of the benefits of both these alternatives. As a compromise, we construct a mapping based on Bézier curves, as illustrated in Figure 6(b) for a typical luminance bin.

In the interest of brevity, we will only describe the Bézier-curve mapping qualitatively here. As with the linear mapping, the two endpoints of the curve are constrained to map s_{\min} to \hat{s}_{\min} and s_{\max} to \hat{s}_{\max} . However, because we are using two cubic Bézier curves that meet with C^1 continuity, we have five additional control points with which we can alter the behavior of the mapping. These additional degrees of freedom are utilized to meet the constraints that follow.

First, the slope of the Bézier-curve mapping at either end (controlled by B_1^0 and B_1^1) is constrained to be zero, while the tangent at the middle of the curve (controlled by the segment $B_2^1 - B_2^0$) is constrained

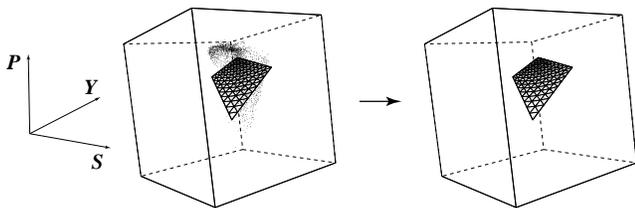


Figure 7 Image colors before and after being projected onto the duotone gamut in the direction P .

to be parallel to the line from B_1^0 to B_1^1 . Together, these constraints impose a nonlinearity on the mapping that tends to preserve separation of s -values for colors that are in gamut while squeezing together the colors that are out of gamut. This approach strikes a balance between a linear mapping and an approach that clamps values.

Second, the center of the curve is constrained to pass through the point $B_3^0 = B_3^1$, which is chosen to lie on the line $\hat{s} = s$ so as to guarantee that s -values near the center of the mapping are preserved. We determine the precise location of this center constraint by intersecting the line from g_0 to $g_{1,2}$ with a plane of constant luminance \hat{y} . As a result, the constraint at the center of the Bézier curves prevents colors that are closer in hue to one ink from mapping to a color on the opposite side of the gamut.

Our implementation of the Bézier-curve mapping relies on a lookup table in order to efficiently transform a value c_i^S to a new value \hat{c}_i^S . The table corresponding to each luminance bin is calculated before any colors are transformed. We evaluate a curve at a large number of parameter values, and store the resulting (s, \hat{s}) coordinates in a table. Then, to map a value c_i^S we need only find the correct interval of s -values and use linear interpolation to approximate the result \hat{c}_i^S .

3.3.3 Calculating the projection

Image colors transformed by the luminance and ink-spread mappings above are guaranteed to project onto the gamut in the P direction. The projection point corresponding to a color can be calculated analytically by solving for the intersection of a line and the bilinear surface defined by the Neugebauer equation. In the appendix, we derive a solution for α_1 and α_2 , the amounts of the two inks required to produce the projected color. Thus, in addition to computing the desired color, the projection step calculates the halftone separations required for the duotone printing process. Figure 7 illustrates the result of projecting a set of image colors onto a duotone gamut.

4 Selecting inks

The duotone mapping described in the previous section assumed that two inks were given. Now we consider more general cases in which neither ink or just one ink is specified by the user. Our goal is to find a few good pairs of inks for a given image, in addition to finding the halftone separations. We define a *good* pair of inks as one for which the duotone mapping algorithm produces an image that is as close as possible to the original full-color image. We define the closeness of two images as the pixel-wise L^2 distance measured in a perceptually uniform color space. Thus, the selection of one or two inks amounts to an optimization problem whose goal (or objective) function is a function of the original image and two inks.

Simulated annealing is a heuristic optimization technique designed to avoid local minima [18]. Transitions from the current state to an-

other state are generated randomly and accepted with some probability. This “acceptance probability” depends both on the relative scores of the two states, as rated by the objective function, and on the value of a control parameter T . Unfavorable moves are likely when T is high but are accepted with decreasing probability as T decreases. The “cooling schedule” describes the rate at which T decreases and the number of moves made at each value of T . In our problem, a state consists of a pair of inks, each of which has a set of neighbor inks. Taken together, the neighbor sets describe a fully connected, symmetric graph. The set of legal moves from a state is the set of all possible combinations of neighbor moves for each ink.

Simulated annealing is an ideal optimization technique for our particular problem. We require a heuristic technique because most interesting ink sets, such as the PANTONE® inks, contain hundreds of inks. In addition, our preliminary experiments indicated that our objective function has many local minima. Finally, the large number of parameters associated with simulated annealing allow us to tune the optimizer to our needs. In particular, by adjusting the cooling schedule and starting conditions, we can cause the optimizer to find several relatively deep local minima instead of a single global minimum. This adjustment is useful because it allows us to present several alternative choices of ink pairs to the user.

Evaluating the objective function tends to be costly because it requires that we transform every pixel color in the original image and compare the resulting color to its original value. In order to solve the optimization problem efficiently, the annealer uses a low-resolution version of the color information present in the original image. Such low-resolution information can be obtained either by quantizing the colors in the original image or by reducing the size of the original image. The low-resolution information is provided to the duotone mapping step instead of the original image colors.

5 Variations

There are several variations on the basic algorithm that lead to improvements in the duotones of certain images.

5.1 Outlying clusters

In our implementation, clustering of image colors is performed as a preprocessing step. Clusters of small size and large average distance from other clusters are marked as outlying clusters. Colors that have been identified as members of outlying clusters are ignored when calculating the ink-spread transformation. The rationale is that such colors matter less to the overall image, and that mapping them well at the expense of more prevalent colors is not justified.

Colors that are ignored in calculating the ink-spread transformation may not project onto the gamut. We transform each such color to a gray of the appropriate luminance, and then project that gray onto the gamut.

5.2 Black enhancement

Full-color printing relies on three colored inks—cyan, magenta, and yellow—that combine to make black. However, a black halftone separation is usually printed in addition to the three color separations. Adding a black separation permits denser blacks than three colors can produce, allows more detail to be expressed in shadowed areas, substitutes inexpensive black ink for more expensive colored inks, and avoids a thick accumulation of ink [23, page 282].

Adding a black separation to duotone printing offers a further benefit: for images with a wide range of luminance values, the two colored inks no longer need to be chosen so that their combination is

close to black. Instead, the first two inks can be chosen to reproduce the image’s hues, while the black ink permits fine gradations in luminance.

Extending the duotone process to support a black separation is simpler than the general problem of using three arbitrary inks (called a *tritone* process), thanks to the nearly accurate assumption that composing black ink with any other ink results in black. This assumption simplifies the problem in two ways. First, whereas a general tritone gamut is a trilinear volume according to the Neugebauer equations, a black-enhanced duotone is a volume bounded by four triangles, $(g_0, g_1, black)$, $(g_0, g_2, black)$, $(g_1, g_{1,2}, black)$, and one bilinear surface, as illustrated in Figure 8. Second, black enhancement can be implemented as a simple extension to the duotone algorithm using the following steps:

1. Subtract some amount of black from each color in the original image.
2. Apply the optimization and duotone mapping algorithm to produce two color separations.
3. Calculate a black separation that adds back the appropriate amount of black to each color.

These three steps are illustrated in Figure 9.

Removing black from an image in order to create a better duotone is similar to the four-color process of *under-color removal* (UCR), though slightly more complicated. Applying UCR to a color is fairly simple: a fraction of the minimum component of cyan, magenta, and yellow is removed from all three components, and then replaced by an equal amount of black [13]. The success of this technique hinges on the fact that cyan, magenta, and yellow combine to black, a property that is not necessarily true for the two inks used in a duotone.

For duotones, removal of black from an RGB color c is performed by moving c in a straight line away from the RGB position of black ink to a new position c' , as shown in Figure 10(a). The amount of shift depends on the color’s saturation: a fully saturated color does not benefit from using any black, while a fully desaturated (gray) color benefits most from using pure black ink. Performing this operation on every color in the original image yields a new target image for the duotone mapping.

Once we have applied the duotone mapping algorithm to obtain new colors, we can compute how much black to use by comparing the duotone colors to the original colors. Because we are primarily concerned with preserving the luminance of image colors, our goal is to match the luminance of the corresponding color in the original image. We cannot reproduce the complete range of original luminance values, though; instead we attempt to match a luminance scaled to lie between that of the paper and that of the black ink. For each color \hat{c} in our duotone, we create a line segment from that color to the position of black ink. We then shift \hat{c} toward black until it achieves the desired luminance at some position \hat{c}' , as illustrated in

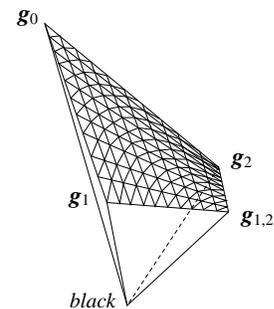


Figure 8 The gamut of two inks and black.

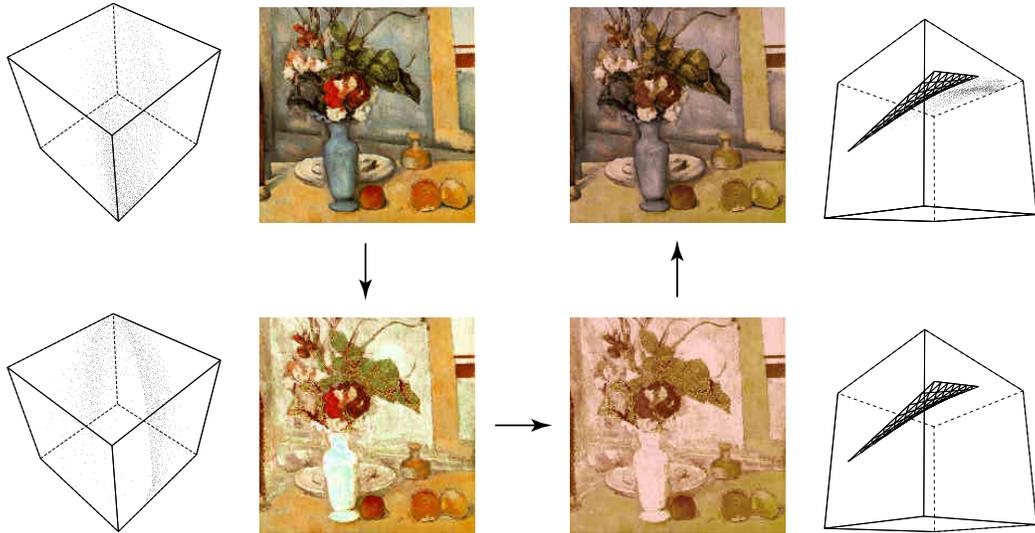


Figure 9 Steps of black enhancement (counterclockwise from top left): original, original with black removed, duotone without black, and duotone with black replaced.

Figure 10(b). The amount of shifting along the line through black determines the amount of black to print.

5.3 Minimized hue difference

For some very colorful images, even the best duotone will not be able to adequately reproduce all colors. Mapping an unachievable original color to gray may produce a more pleasing duotone than mapping it to another color. To achieve this effect we introduced the idea of an optional second pass over the duotone, performed after the desired pair of inks is selected. During the second pass, we perform a pixel-by-pixel comparison of the duotone and the original image, calculating the perceptual hue difference of each pixel. Hues are measured as *hue angles* in a perceptually uniform color space. In both $L^*a^*b^*$ and $L^*u^*v^*$, a plane perpendicular to the L^* axis and passing through a particular brightness value contains a disk of all the hues present at that brightness, ranging from gray at the center to saturated colors on the edge. Thus the hue angle of a color is defined as $h_{uv} = \tan^{-1}(v^*/u^*)$ or $h_{ab} = \tan^{-1}(b^*/a^*)$ [4, page 65]. The perceptual hue difference between two colors is the difference between the hue angles of the colors.

Our algorithm for reducing hue differences proceeds by desaturating (moving toward gray) each pixel in the original image by an amount parameterized by the magnitude of the hue difference computed for that pixel. The duotone mapping is then applied to the grayed-out image. In addition to reducing offensive hue mappings, graying-out parts of the original may improve the ink-spread transformation by bringing s_{min} and s_{max} closer to \hat{s}_{min} and \hat{s}_{max} . Effectively, minimizing hue difference treats some potentially significant image colors like outliers, sacrificing them so that other colors will be mapped better.

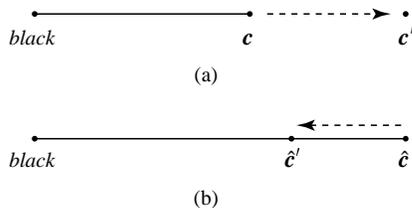


Figure 10 Transformations of black enhancement: (a) reducing black of a color in the original image by moving away from black; (b) replacing black in the duotone by moving toward black.

5.4 Colored paper

The paper used in a duotone need not be white. For an input image without much white, using a colored paper can greatly improve a duotone reproduction by providing an additional color essentially for free. Our implementation allows the user to specify a paper color or leave the choice of paper up to the optimizer. We modified the simulated annealing algorithm to select a paper color by extending the definition of a state to include two inks and a paper.

It is important to note that the colors in a duotone surrounded by an expanse of colored paper (as might occur in a yellow pages directory) appear quite different from the colors in the same duotone printed to cover all of the paper (as in a postcard). The difference is caused by adaptation effects in the visual system, which allow a subtle color to be perceived as white when nothing perceptually closer to white is present [22]. We are currently investigating quantitative models of adaptation that might improve a duotone surrounded by an expanse of colored paper.

6 Results

We have implemented an application that allows a user to create duotone separations. In this section we discuss some details of our implementation, as well as the results we obtained by printing duotones on a two-color press.

6.1 Performance

Our naïve implementation of the optimization loop takes approximately five minutes on a 133 MHz SGI Indy to find three good ink pairs, using the clustered version of an image and an ink set of more than 400 inks. While the optimization process is not interactive, it only needs to be run once for a given image. For a 300×300 image, producing full-size color separations and a preview duotone takes approximately four seconds on the same machine.

6.2 Printing

Testing our method requires that we print duotones on an actual printing press and compare the results to reference prints. While ordinarily a reproduction should look as much as possible like a photograph or an image displayed on a monitor, for the purposes

of this paper we chose to compare duotone prints to four-color process prints. As a consequence, when we provide an input image to our duotone mapping algorithm, its colors should be those that are printed by a four-color press.

All of our original images are stored as *RGB* colors; in order to transform those colors to match the colors obtained from a four-color printing process, we use an *RGB correction* function. We generated data for the *RGB* correction by printing (on a four-color press) a square for each color in a regular three-dimensional grid of colors spanning the *RGB* cube. We scanned each printed square using a spectrophotometer and then converted the spectral data to an *RGB* color. For this set of colors, we constructed a mapping from intended *RGB* values to actual printed *RGB* values. Using piecewise-trilinear interpolation between these discrete colors, we can transform any given *RGB* color to the corresponding *RGB* color that process printing would produce.

There is a second aspect of our color printing experiments that also requires color correction. The Neugebauer model is not entirely accurate in its prediction of colors obtained by halftoning two inks. Therefore, in keeping with the empirical spirit of printing, we developed an empirical correction method for adjusting duotone separations. This *duotone correction* is similar to the *RGB* correction, but we need to measure data for only two dimensions. Using the two selected inks, we print a regular grid of values for α_1 and α_2 . Once again, we measure the *RGB* color of each square in the grid. Locating each measured *RGB* color on the duotone gamut associates intended percentages with printed percentages of inks. Using bilinear interpolation, we construct a mapping from intended amounts to actual printed amounts. Applying the inverse of this function to a pair of separations before printing compensates for the simplicity of the Neugebauer model.

It should be noted that while the *RGB* transformation is valid for all images, the duotone correction is only accurate for the particular pair of inks that were measured. Since one of our goals is to make printing duotones inexpensive and easy, we do not want to require two trips to the printer. We therefore calculated an average correction function based on the duotone correction functions for three different pairs of inks. This average correction was used for most of the results described below.

6.3 Examples

Example 1 depicts a painting by Cézanne printed with process inks and as three duotones. The first two duotones, both printed with black and gold, demonstrate the difference between the traditional approach to printing duotones and our approach. (The traditional duotone was created using Adobe Photoshop™.) The last image in the series, printed with black and two inks selected by optimization, demonstrates black enhancement.

Some very different images reproduce well using a single pair of inks. Conversely, certain images reproduce well using several remarkably different pairs of inks. Example 3 shows a photograph of a koala printed in process color and as a duotone. Example 4 depicts a painting by Renoir in process color and as two duotones. The first duotone of the Renoir painting uses the same pair of user-selected inks as the duotone of the koala photograph. Choosing an unrelated pair of inks for the second duotone of the Renoir painting results in a very different, but still successful reproduction.

Though skin tones are notoriously difficult to reproduce, photographs of people are not beyond the scope of two-color reproduction. Examples 4, 5, and 6 show three portrait photographs printed in process color and as duotones. All three duotones are printed with blue and orange-brown, inks that were selected by the optimizer for the photograph in Example 4.

Example 7 shows a photograph of a sunset in process color and as a black-enhanced duotone printed with hand-picked inks. Viewed on a monitor, the original image contains colors not present in the process gamut. These colors are better achieved in a duotone that has not undergone the correction described in the previous section. Using the same inks, we printed an uncorrected black-enhanced duotone as the final image in the series.

Example 8 shows a painting by Schiele printed in process color and as a duotone on yellow paper with optimizer-selected inks. Using colored paper allows us to reproduce the three dominant hues (yellow, red, and green) present in the original image.

7 Discussion and future work

We have presented an algorithm for reproducing color images with duotones. Our hope is that our approach will be of interest to the desktop publishing community as a high-quality, economical alternative to full-color printing. While the limited number of examples we have sent to a printing press may not be perfect reproductions of the original images, they are certainly vast improvements over traditional duotones.

The core of our method is the duotone mapping: the transformation of a set of scattered image colors onto a surface in three-dimensional color space. Our mapping preserves relationships between image colors at the expense of matching exact colors. One can imagine many other mappings; below are some of the difficulties we found with two alternate approaches:

- **Finding the closest color.** The obvious problem with the approach of mapping an image color to the closest point on the duotone gamut is that out-of-gamut image colors will be clamped. While clamping may be acceptable for certain images and duotone gamuts, in general clamping results in artificial discontinuities and a loss of information.
- **Using orthogonal projection.** Using orthogonal rather than parallel projection would complicate the ink-spread transformation, making it a function of three variables rather than two: $\mathcal{S} = f(y, s, p)$. A more fundamental difficulty with orthogonal projection is the absence of an obvious continuous mapping, since there are colors for which there is no orthogonal projection onto the gamut.

Our research suggests several intriguing directions for future work. Some of the methods presented in this paper might apply to the more general problem of finding the best n inks with which to reproduce an image. Our extension of the optimization algorithm to choose a paper as well as inks suggests that we should adjust our algorithm to take advantage of psychophysical effects such as von Kries adaptation [22] and simultaneous contrast. These effects may allow us to effectively expand the printing gamut by tricking the eye into seeing colors that are not actually achievable.

Another possible extension to our implementation is a system that facilitates the production of two-color brochures by optimizing over images, inks, and papers. The system would take several images as input and choose two inks, one paper, and a specified number of the images that would reproduce well with the selected inks and paper.

Our algorithm has no sense of what parts of images are semantically or aesthetically important. Because the creation of a duotone from an image frequently requires loss of color information, our algorithm would benefit from user input indicating which colors in the original image are most important to preserve. Finally, we are considering combining our duotone mapping algorithm with the artistic screening approach to halftoning presented by Ostromoukhov and Hersch [17].

Acknowledgments

We wish to thank the University of Washington Publications Services for assisting us with our printing experiments. This work was supported by an Alfred P. Sloan Research Fellowship (BR-3495), an NSF Presidential Faculty Fellow award (CCR-9553199), an ONR Young Investigator award (N00014-95-1-0728), a NASA Space Grant, an NSF Graduate Research Fellowship, and industrial gifts from Interval, Microsoft, and Xerox.

Appendix: Parallel projection onto a bilinear surface

Suppose we have a color \mathbf{c} that we want to project in the direction \mathbf{P} onto a duotone gamut defined by the colors \mathbf{g}_0 , \mathbf{g}_1 , \mathbf{g}_2 , and $\mathbf{g}_{1,2}$. The solution, which we will denote by $\hat{\mathbf{c}}$, must lie on the line that passes through \mathbf{c} in the direction \mathbf{P} , so we know $\hat{\mathbf{c}} = \mathbf{c} + t\mathbf{P}$ for some real number t . The solution must also lie on the bilinear surface defining the duotone gamut:

$$\hat{\mathbf{c}} = \mathbf{c} + t\mathbf{P} = \begin{bmatrix} \mathbf{g}_0 & \mathbf{g}_1 & \mathbf{g}_2 & \mathbf{g}_{1,2} \end{bmatrix} \begin{bmatrix} (1 - \alpha_1)(1 - \alpha_2) \\ \alpha_1(1 - \alpha_2) \\ (1 - \alpha_1)\alpha_2 \\ \alpha_1\alpha_2 \end{bmatrix} \quad (1)$$

for some values of α_1 and α_2 . In what follows, we will solve for the unknown ink amounts α_1 and α_2 , rather than the projection point $\hat{\mathbf{c}}$.

First, let's rewrite equation (1) by grouping the terms differently:

$$\mathbf{g}_0 - \mathbf{c} - t\mathbf{P} + (\mathbf{g}_1 - \mathbf{g}_0)\alpha_1 + (\mathbf{g}_2 - \mathbf{g}_0)\alpha_2 + (\mathbf{g}_0 + \mathbf{g}_{1,2} - \mathbf{g}_1 - \mathbf{g}_2)\alpha_1\alpha_2 = 0$$

Notice that if we take the dot product of both sides of this equation with either the \mathbf{S} direction or the \mathbf{Y} direction, we eliminate t because \mathbf{P} , \mathbf{S} , and \mathbf{Y} are defined to be mutually orthogonal. We can write the two equations that result from these dot products as follows:

$$u_1 + u_2\alpha_1 + u_3\alpha_2 + u_4\alpha_1\alpha_2 = 0 \quad (2)$$

$$v_1 + v_2\alpha_1 + v_3\alpha_2 + v_4\alpha_1\alpha_2 = 0 \quad (3)$$

where the constants u_1, \dots, u_4 and v_1, \dots, v_4 are given by

$$\begin{aligned} u_1 &= (\mathbf{g}_0 - \mathbf{c}) \cdot \mathbf{S} & v_1 &= (\mathbf{g}_0 - \mathbf{c}) \cdot \mathbf{Y} \\ u_2 &= (\mathbf{g}_1 - \mathbf{g}_0) \cdot \mathbf{S} & v_2 &= (\mathbf{g}_1 - \mathbf{g}_0) \cdot \mathbf{Y} \\ u_3 &= (\mathbf{g}_2 - \mathbf{g}_0) \cdot \mathbf{S} & v_3 &= (\mathbf{g}_2 - \mathbf{g}_0) \cdot \mathbf{Y} \\ u_4 &= (\mathbf{g}_0 + \mathbf{g}_{1,2} - \mathbf{g}_1 - \mathbf{g}_2) \cdot \mathbf{S} & v_4 &= (\mathbf{g}_0 + \mathbf{g}_{1,2} - \mathbf{g}_1 - \mathbf{g}_2) \cdot \mathbf{Y} \end{aligned}$$

Next, we can solve equation (2) for α_2 in terms of α_1 to get

$$\alpha_2 = -\frac{u_1 + u_2\alpha_1}{u_3 + u_4\alpha_1} \quad (4)$$

When we substitute this expression for α_2 into equation (3) and simplify, we get a quadratic equation for α_1 alone:

$$w_1\alpha_1^2 + w_2\alpha_1 + w_3 = 0$$

where

$$\begin{aligned} w_1 &= u_4v_2 - u_2v_4 \\ w_2 &= u_4v_1 - u_1v_4 + u_3v_2 - u_2v_3 \\ w_3 &= u_3v_1 - u_1v_3 \end{aligned}$$

Therefore, the solution for α_1 is given by

$$\alpha_1 = \begin{cases} \frac{-w_2 \pm \sqrt{w_2^2 - 4w_1w_3}}{2w_1} & \text{if } w_1 \neq 0 \\ -\frac{w_3}{w_2} & \text{if } w_1 = 0 \end{cases}$$

When $w_1 \neq 0$, there are two possible projections of \mathbf{c} onto the bilinear surface; we choose the solution for α_1 that results in a value between zero and one. Once we obtain α_1 , we can compute α_2 from equation (4) and the projection point $\hat{\mathbf{c}}$ from equation (1).

References

- [1] F. R. Clapper and J. A. C. Yule. The effect of multiple internal reflections on the densities of half-tone prints on paper. *Journal of the Optical Society of America*, 43(7):600–603, July 1953.
- [2] James D. Foley, Andries van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics: Principles and Practice*. Addison-Wesley, Reading, MA, second edition, 1990.
- [3] Ronald S. Gentile and Jan P. Allebach. A comparison of techniques for color gamut mismatch compensation. In *Human Vision, Visual Processing, and Digital Display*, volume 1077 of *Proceedings of the SPIE*, pages 342–354. SPIE, Bellingham, WA, 1989.
- [4] Andrew S. Glassner. *Principles of Digital Image Synthesis* volume 1. Morgan Kaufmann, San Francisco, 1995.
- [5] Andrew S. Glassner, Kenneth P. Fishkin, David H. Marimont, and Maureen C. Stone. Device-directed rendering. *ACM Transactions on Graphics*, 14(1):58–76, January 1995.
- [6] Ed Granger. Light Source Computer Images, Inc. Personal communication, December 1995.
- [7] Arthur C. Hardy and F. L. Wurzburg, Jr. Color correction in color printing. *Journal of the Optical Society of America*, 38(4):300–307, April 1948.
- [8] Steven J. Harrington and R. Victor Klassen. Color printing having a highlight color image mapped from a full color image. U.S. Patent 5,237,517, August 17, 1993.
- [9] Ewald Hering. *Outlines of a Theory of the Light Sense*. Harvard University Press, Cambridge, MA, 1964.
- [10] R. W. G. Hunt. *The Reproduction of Colour*. Fountain Press, Kingston-upon-Thames, England, fifth edition, 1995.
- [11] Kansei Iwata and Gabriel Marcu. Computer simulation of printed colors on textile materials. In *Color Hard Copy and Graphic Arts III*, volume 2171 of *Proceedings of the SPIE*, pages 228–238. SPIE, Bellingham, WA, 1994.
- [12] Henry R. Kang. Applications of color mixing models to electronic printing. *Journal of Electronic Imaging*, 3(3):276–287, July 1994.
- [13] Bruce J. Lindbloom. Accurate color reproduction for computer graphics applications. In *Proceedings of SIGGRAPH '89*, pages 117–126. ACM, New York, 1989.
- [14] Gabriel Marcu and Satoshi Abe. Color designing and simulation in non-conventional printing process. In *Applications of Digital Image Processing XVII*, volume 2298 of *Proceedings of the SPIE*, pages 216–223. SPIE, Bellingham, WA, 1994.
- [15] H. E. J. Neugebauer. Die Theoretischen Grundlagen des Mehrfarbendruckes (The theoretical foundation for multicolor printing). *Z. Wiss. Photogr.*, pages 73–89, 1937. Reprinted in Sayanai [20], pages 194–202.
- [16] Victor Ostromoukhov. Chromaticity gamut enhancement by heptatone multi-color printing. In *Device-Independent Color Imaging and Imaging Systems Integration*, volume 1909 of *Proceedings of the SPIE*, pages 139–151. SPIE, Bellingham, WA, 1989.
- [17] Victor Ostromoukhov and Roger D. Hersch. Artistic screening. In *Proceedings of SIGGRAPH 95*, pages 219–228. ACM, New York, 1995.
- [18] R. H. J. M. Otten and L. P. P. van Ginneken. *The Annealing Algorithm*. Kluwer Academic Publishers, Boston, 1989.
- [19] Warren Rhodes. Fifty years of the Neugebauer equations. In Sayanai [20], pages 7–18.
- [20] Kazuo Sayanai, editor. *Neugebauer Memorial Seminar on Color Reproduction*, volume 1184 of *Proceedings of the SPIE*. SPIE, Bellingham, WA, 1990.
- [21] Maureen C. Stone, William B. Cowan, and John C. Beatty. Color gamut mapping and the printing of digital color images. *ACM Transactions on Graphics*, 7(4):249–292, October 1988.
- [22] Johannes von Kries. Chromatic adaptation. In David L. MacAdam, editor, *Sources of Color Science*, pages 109–119. MIT Press, Cambridge, MA, 1970.
- [23] John A. C. Yule. *Principles of Color Reproduction*. Wiley, New York, 1967.