Efficient, Chromaticity-Preserving Midtone Correction for RGB Images

James M. Kasson
IBM Almaden Research Center, San Jose, California

Abstract

RGB Gamma adjustment, the standard method for control of midtone values, causes large changes in chromaticity. This paper presents an image transform algorithm that allows control of midtone values while producing chromatically-correct results. The exact algorithm requires computations only moderately more complex than those required for gamma adjustment. Approximations to the algorithm are simpler to implement than gamma adjustment, yet produce results which are more chromatically correct.

Colors in CRT-based Image Systems

The primaries of a color cathode ray tube (CRT) are the visible emissions of three different mixtures of phosphors, each of which can be independently excited by an electron beam. When an observer views the CRT from a proper distance, the individual phosphor dots cannot be resolved, and the contributions of each of the three phosphors are added together to create a combined spectrum, which the viewer perceives as a single color. To a first approximation, the intensity of light emitted from each phosphor is proportional to the electron beam current raised to a power, usually referred to as *gamma*, thus a CRT with constant gamma for all three phosphors, viewed in a dark room, produces a color which can be described, in the color space of its primaries, as

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} \propto \begin{bmatrix} (I_R)^\gamma \\ (I_G)^\gamma \\ (I_B)^\gamma \end{bmatrix} \tag{1}$$

The beam current of computer monitor usually bears a linear relationship to the values stored in the display buffer. With the RGB values scaled into the range [0,1], if we wish to produce a color with tristimulus values in the color space of the monitor's primaries of $\begin{bmatrix} R \\ G \\ B \end{bmatrix}$, then the beam current should be $\begin{bmatrix} R^{1/\gamma} \\ G^{1/\gamma} \\ B^{1/\gamma} \end{bmatrix}$. This nonlinear RGB encoding is often called *gamma-corrected RGB*.

The jargon of gamma correction has a confusing quirk: an image that has been gamma corrected for a monitor with a gamma of 2.2 is referred to as itself having a gamma of 2.2, even though the gamma correction applied to a linear representation to correct it amounts to raising each component of each pixel in the image to the power 1/2.2, or 0.45. Gamma-corrected RGB is the prevalent color space for storing color images.

Correcting Midtones

One of the most commonly-performed operations in RGB image editing is midtone brightness control. Usually, the operator selects the white and black points, then adjusts the image gamma for the right midtone values using the controls that change the gamma correction for monitors with different nonlinearities. The image gamma adjustment subjects each color plane of pixel in the image to the following operation, for an image scaled into the range [0,1]:

$$\begin{aligned} R_{out} &= R_{in}^{\gamma} \\ G_{out} &= G_{in}^{\gamma} \\ B_{out} &= B_{in}^{\gamma} \end{aligned} \tag{2}$$

Since $0^x=0$ and $1^x=1$, this function does not affect either the white or the black point. Most image editors make it possible to pick different gammas for each color plane and to construct nonlinearities other than power laws, but they provide midtone controls that subject each color plane of the image to a nonlinearity. Performed in a nonlinear RGB color space, for the purpose of modifying midtone values rather than correcting for a specific monitor, this kind of operation causes an unwanted side-effect: the chromaticities of the pixels are altered. The nature of the alteration is neither simple nor easy to predict, and is dependent on the primaries and nonlinearities of the RGB color space selected for image manipulation. Figure 1 illustrates the CIELAB chromaticity changes associated with raising all three color planes of a nonlinear (gamma = 2.2) RGB color space to the 0.667 power, which lightens the midtones. The input image consists of a rectangular grid of CIELAB[1] values, all with an L* of 50.0, and a* and b* values than span the range [-60.0, 60.0] in increments of 10.0.

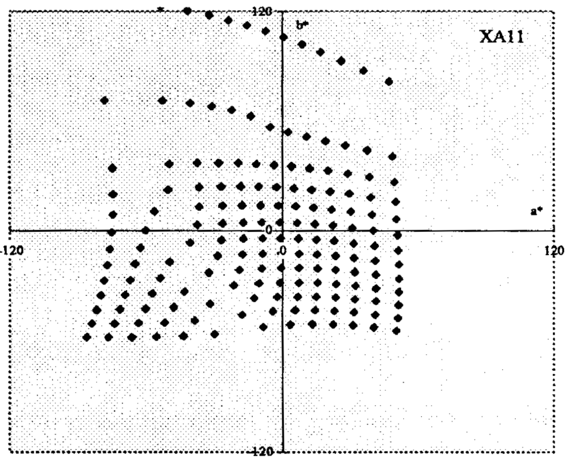

Figure 1. Chromaticity Changes for Gamma Adjustment

Although the CCIR 709 primaries were used to create Figure 1, the effects shown are similar if other common RGB primaries are used: desaturation of the reds, magentas, and blues, overly saturated cyans, and wildly over-saturated greens and yellows, accompanied by some color shifts: saturated greens towards yellow, magentas and some reds towards blue, and saturated blues towards cyan.

Gamma Adjustment in Luminance-Chrominance Space

A conceptually simple, but possibly computationally expensive, solution exists for unwanted chromaticity changes: convert the image to a true luminance-chrominance color space, manipulate only the luminance by subjecting it to a nonlinearity, then convert the result back to RGB. There are several color spaces with pretensions to luminance-chrominance status for which conversions to and from nonlinear RGB are fairly simple. One of these, YCrCb[2], is achieved via the following matrix multiplication (the primes indicate representations of the RGB signals which have been gamma-corrected with a gamma of 2.2):

$$\begin{bmatrix} Y \\ Cr \\ Cb \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.701 & -0.587 & -0.114 \\ -0.299 & -0.587 & 0.886 \end{bmatrix} \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix} \quad (3)$$

The calculation looks even simpler if expressed in the following form:

$$Y = 0.299R' + 0.587G' + 0.114B'$$
$$Cr = R' - Y$$
$$Cb - B' - Y \quad (4)$$

Unfortunately, this putative luminance-chrominance color space produces poor results when used in the previously-described midtone mapping algorithm (the R', G', and B' values upon which the tested YCrCb is based have the XA11 primaries and a gamma of 2.2):

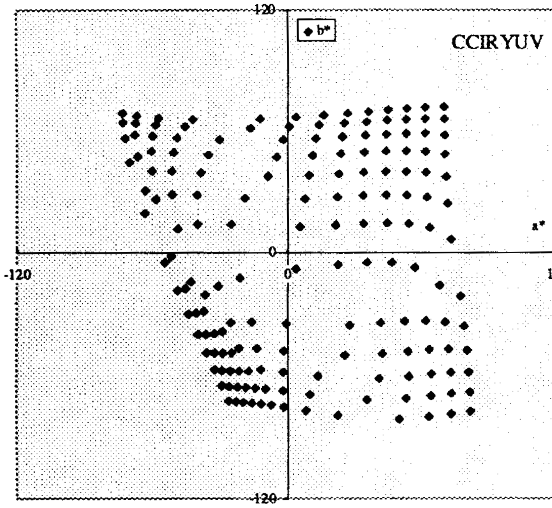

Figure 2. Chromaticity Changes for Gamma Correction in YCrCb

YIQ[3,4,5], a luminance-chrominance space used in National Television Standards Committee (NTSC) television transmission, produces even worse results. Better results could be obtained by converting to CIELAB, CIELUV[6], or possibly ATD[7], but the equations relating these color spaces to nonlinear RGB are complex, and conversion is more practically performed by three-dimensional interpolation, an operation that is computationally expensive.

Goals for Midtone Correction

Before discussing midtone-correction transforms in detail, we must first clearly state their goals. The simplest formulation is that the transforms should afford changes in luminance with no changes in chromaticity. Implied in this formulation is some color space with a chromaticity representation. We take the xy chromaticity as our standard representation. The effect of raising the luminance of part of an image can be likened to shining light into that section. The parallel is not perfect, since the operations described in this disclosure are all point-processes; that is, they operate on each pixel without consideration of any other pixel in the image, but the point is well illustrated if one thinks of an image as consisting of a group of solid-color patches. Raising the luminance of any one patch would be correctly performed if we were able to shine more light on that patch, and lowering the luminance of any one patch would be correctly performed if we were able to shine less light on that patch in the original scene. As shown below, increasing the luminance of the light source does not change the xy chromaticity of a thereby illuminated object.

A Midtone-Mapping Algorithm

Consider a surface color illuminated by an illuminant $I(\lambda)$. If the reflectivity of the surface is $Ref(\lambda)$, the spectrum of the reflected light is $O(\lambda)=I(\lambda)Ref(\lambda)$. To convert the spectrum of the reflected light into a linear RGB color space, we compute the wavelength-by-wavelength product of the reflection spectrum with a set of color-matching functions $r(\lambda)$, $g(\lambda)$, and $b(\lambda)$ as follows:

$$\begin{aligned} R &= \int_{-\infty}^{\infty} O(\lambda)r(\lambda)d\lambda = \int_{-\infty}^{\infty} I(\lambda)Ref(\lambda)r(\lambda)d\lambda \\ G &= \int_{-\infty}^{\infty} O(\lambda)r(\lambda)d\lambda = \int_{-\infty}^{\infty} I(\lambda)Ref(\lambda)g(\lambda)d\lambda \\ B &= \int_{-\infty}^{\infty} O(\lambda)r(\lambda)d\lambda = \int_{-\infty}^{\infty} I(\lambda)Ref(\lambda)b(\lambda)d\lambda \end{aligned} \quad (5)$$

Say an object with is illuminated by a source with spectrum $I_1(\lambda)$. When encoded into an arbitrary RGB color space the results are:

$$\begin{aligned} R_1 &= \int_{-\infty}^{\infty} I_1(\lambda)Ref(\lambda)r(\lambda)d\lambda \\ G_1 &= \int_{-\infty}^{\infty} I_1(\lambda)Ref(\lambda)g(\lambda)d\lambda \\ B_1 &= \int_{-\infty}^{\infty} I_1(\lambda)Ref(\lambda)b(\lambda)d\lambda \end{aligned} \quad (6)$$

Now say the illuminant's intensity is changed so that it is α times as bright as before. The new illuminant $I_2(\lambda)$ has the spectrum

$$I_2(\lambda) = \alpha I_1(\lambda) \quad (7)$$

When encoded in the same RGB color space as above,

$$\begin{aligned} R_2 &= \int_{-\infty}^{\infty} \alpha I_1(\lambda)Ref(\lambda)r(\lambda)d\lambda = \alpha R_1 \\ G_2 &= \int_{-\infty}^{\infty} \alpha I_1(\lambda)Ref(\lambda)g(\lambda)d\lambda = \alpha G_1 \\ B_2 &= \int_{-\infty}^{\infty} \alpha I_1(\lambda)Ref(\lambda)b(\lambda)d\lambda = \alpha B_1 \end{aligned} \quad (8)$$

In words, increasing the illuminant to α times its previous value causes each component of a linear RGB representation to be multiplied by α. Thus, it is not necessary to use a luminance-chrominance color space in the algorithm; all that is required is to linearly increase the value of each component of the linear RGB triplet describing each pixel by an amount that depends on the original luminance of the pixel. This processing will not change the xy chromaticity of the pixel, since multiplying each component of a linear RGB color by a constant α causes the XYZ representation to be multiplied by the same constant. If

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = \mathrm{M} \begin{bmatrix} R_1 \\ G_1 \\ B_1 \end{bmatrix} \quad (9)$$

then

$$\begin{bmatrix} X_2 \\ Y_2 \\ Z_2 \end{bmatrix} = \mathrm{M} \begin{bmatrix} \alpha R_1 \\ \alpha G_1 \\ \alpha B_1 \end{bmatrix} = \alpha \mathrm{M} \begin{bmatrix} R_1 \\ G_1 \\ B_1 \end{bmatrix} = \alpha \begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = \begin{bmatrix} \alpha X_1 \\ \alpha Y_1 \\ \alpha Z_1 \end{bmatrix} \quad (10)$$

and thus

$$\begin{aligned} x_2 &= \frac{X_2}{X_2+Y_2+Z_2} = \frac{\alpha X_1}{\alpha X_1+\alpha Y_1+\alpha Z_1} = \frac{X_1}{X_1+Y_1+Z_1} = x_1 \\ y_2 &= \frac{Y_2}{X_2+Y_2+Z_2} = \frac{\alpha Y_1}{\alpha X_1+\alpha Y_1+\alpha Z_1} = \frac{Y_1}{X_1+Y_1+Z_1} = y_2 \end{aligned} \quad (11)$$

The processing shown below meets the objective.

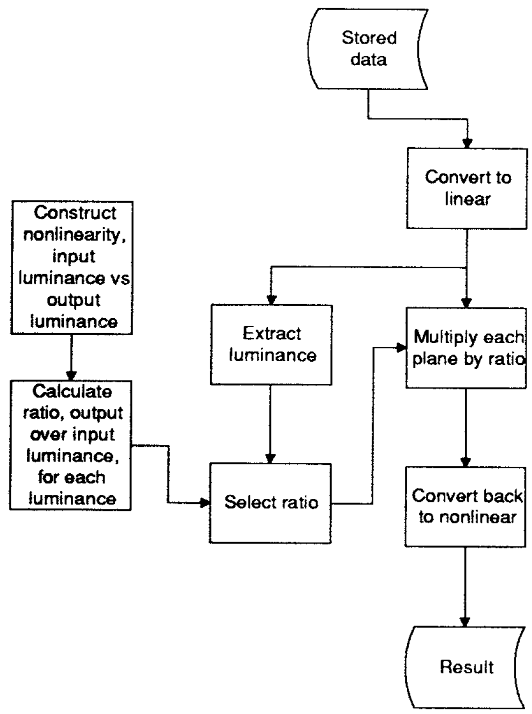

Figure 3. Chromaticity-Preserving Midtone Correction Algorithm

A straightforward implementation of the above algorithm has the following computational costs per pixel:

Operation	Adds	Multiplies	Table look-ups
Linearize			3
Compute Luminance	2	3	
Compute Ratio			1
Multiply by Ratio		3	
Nonlinearize			3
Total	2	6	7

Using the above algorithm, the following results are obtained for the midtone-lightening test, perfectly meeting the objective:

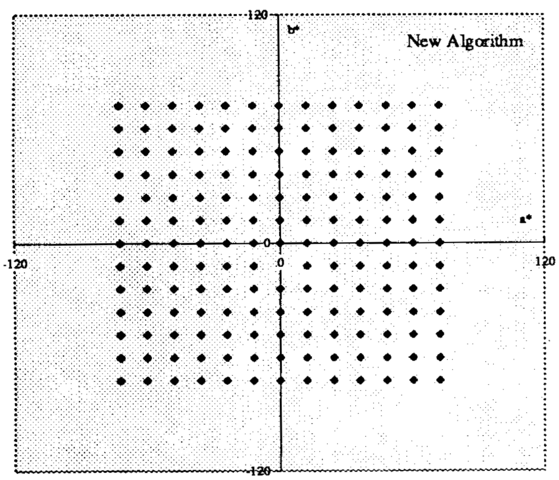

Figure 4. Chromaticity Changes for New Midtone Correction Scheme

Note that the a* and b* values have increased from the input. This occurs because a brighter object of given xy chromaticity has greater perceived chroma than a dimmer one of the same xy chromaticity, and CIELAB takes this occurance into account.

It is possible to improve the computational efficiency without sacrificing accuracy. The first improvement is realized by performing the multiplications directly on the gamma- corrected RGB data, instead of linearizing it first. This can provide equivalent results because:

$$kr' = kr^{\gamma} = \left(k^{\frac{1}{\gamma}} r\right)^{\gamma}$$
$$kg' = kg^{\gamma} = \left(k^{\frac{1}{\gamma}} g\right)^{\gamma} \quad (12)$$
$$kb' = kb^{\gamma} = \left(k^{\frac{1}{\gamma}} b\right)^{\gamma}$$

or, stated another way,

$$k^{\gamma} r' = k^{\gamma} r^{\gamma} = (kr)^{\gamma}$$
$$k^{\gamma} g' = k^{\gamma} g^{\gamma} = (kg)^{\gamma} \quad (13)$$
$$k^{\gamma} b' = k^{\gamma} b^{\gamma} = (kb)^{\gamma}$$

Thus, multiplying a linear representation by a constant, say k, and then raising it to the power γ produces the same results as multiplying a gamma-corrected representation by a different constant, k^{γ}. So the following block diagram can produce identical results to the more-complex one shown in Figure 3, with the proviso that the nonlinearity constructed is different: it is the previous one raised to the γ power.

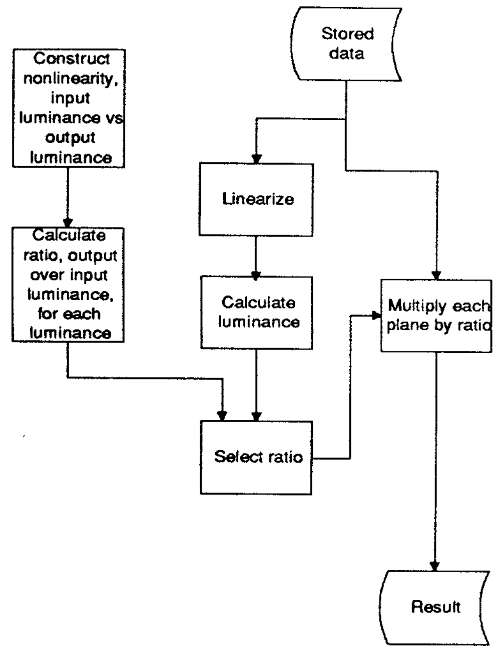

Figure 5. Simplified Midtone Correction Algorithm

This approach saves three table look-ups:

Operation	Adds	Multiplies	Table look-ups
Linearize			3
Compute Luminance	2	3	
Compute Ratio			1
Multiply by Ratio		3	
Total	2	6	4

Approximations

It is reasonable to ask if there are approximations to this approach which consume fewer computational resources yet yield similar results. A promising avenue is simplifying the luminance calculation. The television industry has developed an approximation to luminance which can be computed from gamma-corrected RGB signals with only additions and multiplications. In the NTSC standard, luminance is defined as:

$$Y' = 0.299\, R' + 0.587\, G' + 0.114\, B' \quad (14)$$

where the primes indicate the gamma-corrected values. The prime is usually dropped from the Y' designator.

This approximation to luminance is accurate along the gray axis, and shows increasing errors as colors become more saturated. If we use this approximation instead of computing luminance accurately, we eliminate the three table lookups involved in linearization of the data, and the algorithm becomes:

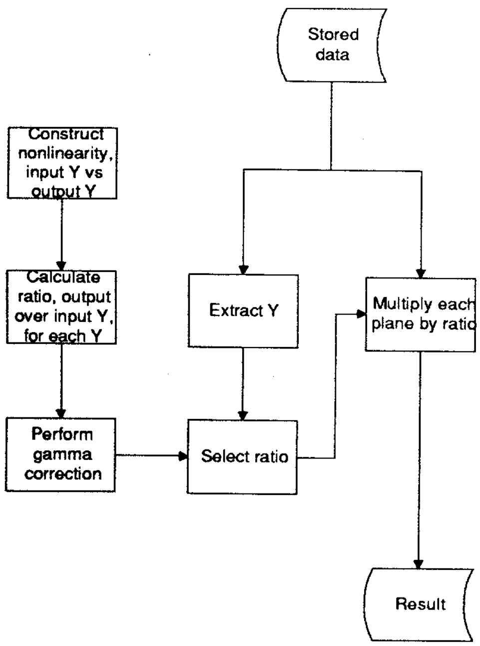

Figure 6. Midtone Correction Algorithm Using NTSC Approximation to Luminance

This produces the following results, which for our set of sample points have an average chrominance error of 4.6 ΔE_{ab}, and a worst-case chrominance error of 9.6 ΔE_{ab}:

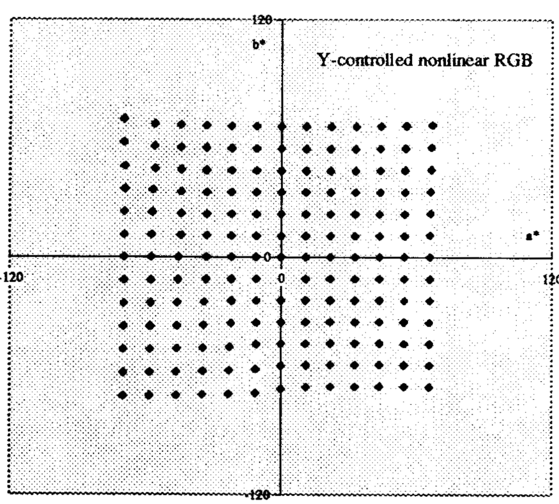

Figure 7. Results: Midtone Correction Using NTSC Approximation to Luminance

This simplification means that the computational cost per pixel is as follows:

Operation	Adds	Multiplies	Table look-ups
Compute Luminance	2	3	
Compute Ratio			1
Multiply by Ratio		3	
Total	2	6	1

Noting that the luminance computation is still a large portion of the total computational cost, we can further approximate Y by employing coefficients that are powers of two and using shifts to avoid actual multiplication.

$$Y' = 0.25r' + 0.625g' + 0.125b'$$
$$Y' = 0.25r' + 0.5g' + 0.125g' + 0.125b' \quad (15)$$

Implementing this costs:

Operation	Adds	Adds	Multiplies	Table look-ups
Compute Luminance	3	4		
Compute Ratio				1
Multiply by Ratio			3	
Total	3	4	3	1

and produces the following results, with an average chrominance error of 4.9 ΔE_{ab}, and a worst-case chrominance error of 9.7 ΔE_{ab}:

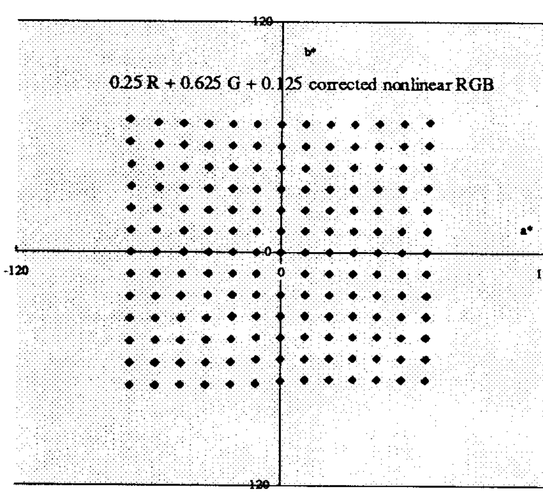

Figure 8. Results: Midtone Correction Using Approximation to NTSC Luminance Approximation

There are other, more approximate simplifications. If we approximate Y by:

$$Y' = 0.25r' + 0.75g'$$
$$Y' = 0.25r' + 0.25g' + 0.5g' \quad (16)$$

the computationoal cost is:

Operation	Adds	Adds	Multiplies	Table look-ups
Compute Luminance	2	3		
Compute Ratio				1
Multiply by Ratio			3	
Total	2	3	3	1

and the results have an average chrominance error of 5.0 ΔE_{ab}, and a worst-case chrominance error of 8.9 ΔE_{ab}:

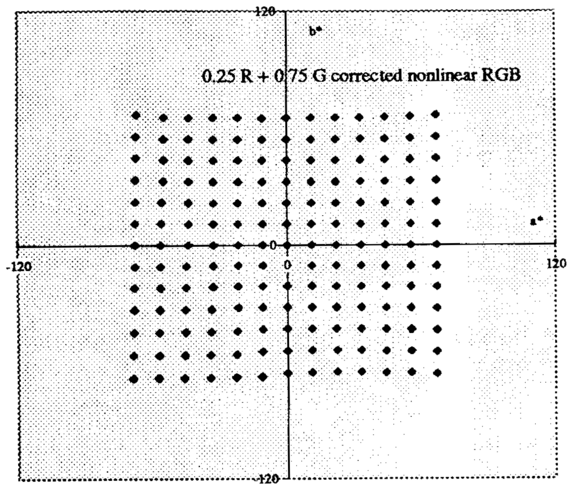

Figure 9. Results: Midtone Correction Using Simpler Approximation to NTSC Luminance Approximation

The simplest approximation to Y is:

$$Y=g' \quad (17)$$

It costs:

Operation	Adds	Adds	Multiplies	Table look-ups
Compute Ratio				1
Multiply by Ratio			3	
Total	0	0	3	1

If, as is the case in most hardware implementations, that multiplies are cheaper than one-dimensional table look-ups, this algorithm is slightly less computationally complex than the traditional one, and yields less color distortion, with an average chrominance error of 5.2 ΔE_{ab}, and a worst-case chrominance error of 12.0 ΔE_{ab}, as shown below.

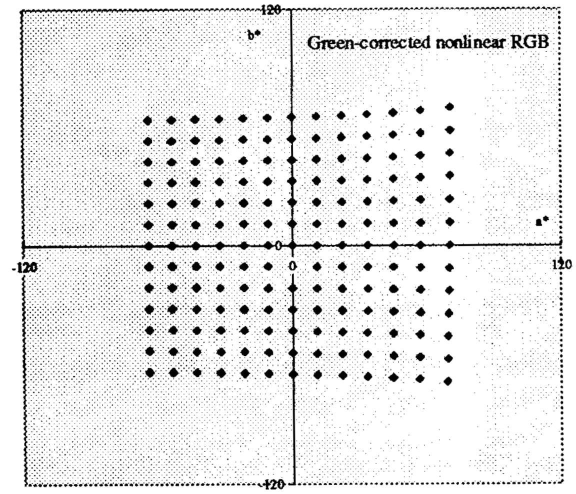

Figure 10. Results: Midtone Correction Using Green as Luminance Approximation

The following table summarizes the CIELAB chromaticity errors, measured in ΔE_{ab}, for the various approximations.

	Average	Maximum
Y-controlled	4.6	9.6
0.25r + 0.625g + 0.125b	4.9	9.7
0.25r + 0.75g	5.0	8.9
Green-controlled	5.2	12.0

Assuming that a shift costs half an add, that a multiply costs four times as much and a table lookup costs eight times as much, the algorithms presented here have the following costs:

	Stan-dard	Lin-ear L	Non-linear L	NTSC Y	0.25r +0.625g +0.125b	0.25r +0.75g	g
Adds		2	2	2	3	2	
Multiplies		6	6	6	3	3	3
Table Lookups	3	7	4	1	1	1	1
Shifts					4	3	
Equivalent Adds	24	82	58	34	25	23.5	20

Thus the improved chromatic accuracy of the approches presented here are purchased at no or little cost over conventional methods.

We have discussed only the chromaticity effects of the various approximations to luminance. There are luminance effects as well, and, for many images, they could be more significant. For example, using just the green plane to moderate midtone correction will cause saturated midtone blues and reds to be less affected than they should be.

References

1. CIE, Colorimetry, *CIE Publication* No. 15.2, 2nd Edition, pp. 29-30, Commission Internationale de L'Eclairage, Vienna (1986).
2. CCIR, "Encoding Parameters of Digital Television for Studios," CCIR Recommendation 601-2, *International Radio Consultative Committee* (ITU) (1990).
3. Foley, J. D., van Dam, A., Feiner, S. K., & Hughes, J. F., *Computer Graphics, Principles and Practice*, Second Edition, Addison-Wesley, Reading, Massachusetts, pp. 589-590 (1990).
4. Hunt, R. W. G., *The Reproduction of Colour in Photography, Printing, and Television*, 4th Edition, pp. 452-458, Fountain Press, England (1987).
5. Benson, K. B., revised by Whittaker, J., *Television Engineering Handbook*, McGraw-Hill, New York, NY, pp 4.34-4.35 (1992).
6. Ibid., pp. 30-31.
7. S. L. Guth, "Model for color vision and light adaptation," *Journal of the Optical Society of America*, vol. **8**, no. 6. pp. 976-993 (June 1991).