

Computer monitors and TVs have light sources made of many groups of three small pixels. One pixel in each group generates visually red light, one green, and one blue. Each group can generate various levels of brightness for each color. Computer monitors thus use a triplet of red, green, and blue (RGB) values to define how much of each primary color is used to generate a desired color. In most monitors, each value of R, G, or B ranges from 0 (no light of that color emitted; the pixel is turned off) to $2^8 - 1 = 255$ (maximum brightness). This is usually called “8 bit/channel” or “24 bit,” i.e. 3×8 , color. These 256 brightness levels for each primary color give a total of $256^3 = 16,777,216$ different colors and levels of brightness that can be displayed, even though the human eye can perceive differences in far fewer colors and brightness levels. Digital cameras likewise store images in terms of RGB values.

As was seen on the Chromaticity page, converting a spectrum $\Lambda(\lambda)$ into CIE (X, Y, Z) and then into (x, y) chromaticity values is easy. Converting between (X, Y, Z) and (R, G, B) values is more complicated.

From CIE XYZ to RGB Values

The personal computer (PC) industry (Microsoft and Hewlett Packard in particular, see Stokes et al., 1996) defined an RGB color model or color space, i.e. a gamut of colors, for use with PC monitors. This color model, called sRGB, is the gamut of colors that can be generated on most PC computer monitors. (Not surprisingly, Apple has its own color model, as does Adobe, and there are at least a dozen others.)

The conversion from CIE XYZ to sRGB coordinates is given by

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 3.2404542 & -1.5371385 & -0.4985314 \\ -0.9692660 & 1.8760108 & 0.0415560 \\ 0.0556434 & -0.2040259 & 1.0572252 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (1)$$

In this equation, the (X, Y, Z) values are assumed to be scaled from 0 to 1. The conversion matrix seen here comes from the excellent web site maintained by Bruce Lindbloom. The Lindbloom web site gives all of the details about color spaces and conversions from one color space to another, including online calculators and downloadable spreadsheets for doing various calculations.

Now consider CIE reference illuminant E, which is pure white, with equal energy at each wavelength. Setting $\Lambda(\lambda) = 1$ for all λ gives a values of $7.30 \cdot 10^4$ for each of X, Y , and Z . Rescaling these values to 1 and using them in Eq. (likesection1) gives $(R, G, B) = (1.205, 0.948, 0.909)$, which can be rescaled to $(255, 201, 192)$. This result is not $(255, 255, 255)$ as might be expected for a pure white color. The computed RGB color is actually a pale red or pink. The reason for this color mismatch is that sRGB is based on the D65 reference illuminant, whereas the CIE 1931 chromaticity diagram uses reference illuminant E to define “white.”

If we let $\Lambda(\lambda)$ be the normalized D65 spectrum seen in the last figure of the Chromaticity page, then $(X, Y, Z) = (58228.0, 61262.5, 66713.0)$. Rescaling these values to a maximum value of 1 and inserting them into Eq. (likesection1) then returns each of R, G , and B as 0.918. Multiplying these values by 255 rescales them to $(234, 234, 234)$, which is a very light

gray. Keep in mind that gray is just white that is not very bright, so these values can be rescaled to (255, 255, 255) without changing the color. Thus the sRGB color model returns the D65 illuminant as the white color, and it returns the equal-energy spectrum E as a pink color.

The conversion from sRGB values to XYZ is given by the inverse of Eq. (likesection1):

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.4124564&0.3575761&0.1804375 \\ 0.2126729&0.7151522&0.0721750 \\ 0.0193339&0.1191920&0.9503041 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (2)$$

The RGB values in Eq. (likesection2) are considered to be between 0 and 1. If we rescale $(R, G, B) = (255, 255, 255)$ to $(1, 1, 1)$ and insert into this equation, it returns $(X, Y, Z) = (0.950, 1.000, 1.089)$, which gives the CIE chromaticity coordinates $(x, y) = (0.313, 0.329)$. This is the slightly bluish color of the D65 spectrum shown by the box symbol in Fig. figure1.

Using Eq. (likesection2) to convert $(R, G, B) = (1, 0, 0)$ gives $(X, Y, Z) = (0.4125, 0.2127, 0.01933)$, which gives $(x, y) = (0.640, 0.330)$. Converting $(0, 1, 0)$ gives $(x, y) = (0.30, 0.60)$, and $(0, 0, 1)$ gives $(x, y) = (0.15, 0.06)$. These three (x, y) values define the vertices of the sRGB gamut triangle seen in Fig. figure1. It is clear that this gamut cannot accurately reproduce many of the colors visible to the eye (nor can any of the other gamuts used in computer monitors and TVs). Indeed, the sRGB color space represents only 35% of the entire CIE (x, y) color space. In other words, only 35% of the colors as seen by the human eye can be accurately reproduced on a computer monitor using the sRGB color model.

It was mentioned on the Color Vision page that “brown is just red that is not very bright.” Figure 4 of that page showed a sequence of color patches that were generated by turning on only the red pixels of a monitor, but with decreasing intensities. The perceived color went from bright red for $(R, G, B) = (255, 0, 0)$ to a dark reddish brown for $(R, G, B) = (125, 0, 0)$. Those two color patches are reproduced in Fig. figure2. If the RGB values for these two color patches are rescaled to $(1, 0, 0)$ and $(0.49, 0, 0)$, respectively, and inserted into Eq. (likesection2), the resulting CIE chromaticities are the same for each: $(x, y) = (0.640, 0.330)$. What is different is the luminance: $Y = 0.2127$ for the bright red patch, and $Y = 0.1043$ for the dark brown patch. It is likely that most people would call these patches different colors, but in the language of color science, they are the same color but different brightnesses (same chromaticity but different luminances).

There are additional complications to mention in the conversion from XYZ to RGB color specifications.

First, the eye does not respond in a linear fashion to differences in brightness. In particular, the eye is more sensitive to differences in darker color tones than it is to differences in brighter tones. Therefore, a linear spacing of RGB values allocates more of the 0-255 range to high values, which the eye cannot distinguish well, than to low values where the eye sees greater differences for a the same difference in RGB values. Therefore, after using Eq. (likesection1) to obtain “linear” RGB values, is it customary to apply a *gamma correction* to the result. A gamma correction is a non-linear “stretching” of the RGB values so as to make better use of the limited range of values (usually 0 to 255) available for storing the

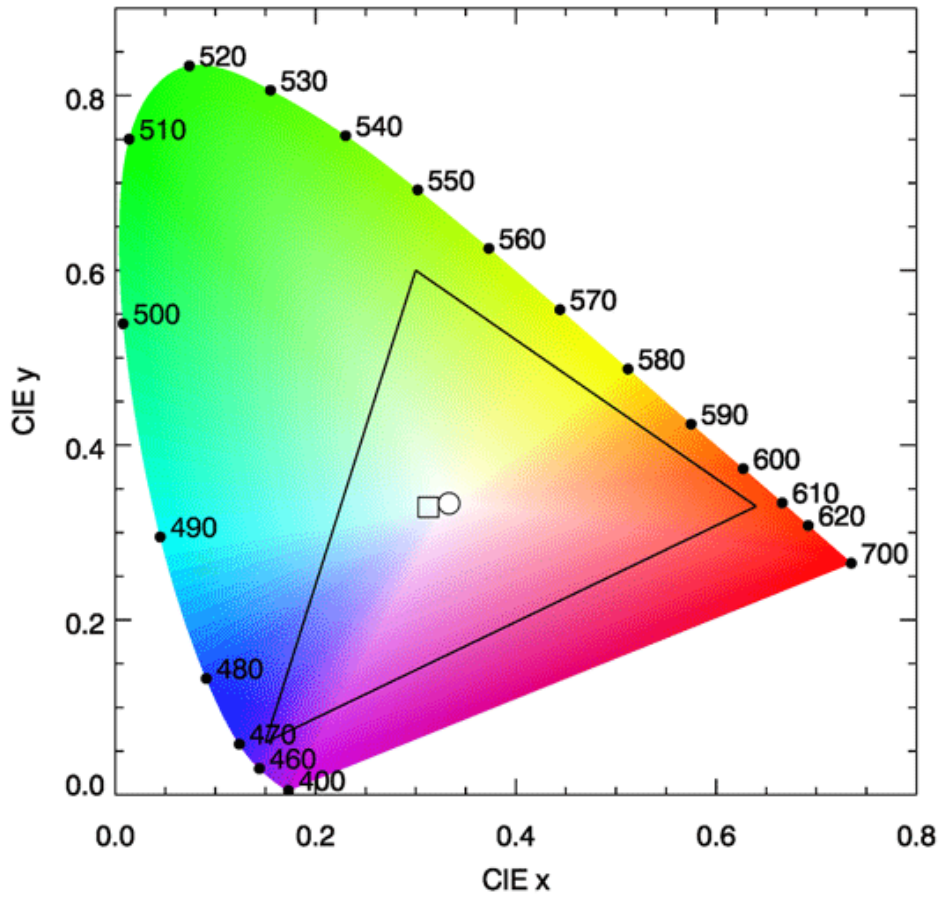


Figure 1: The CIE chromaticity diagram with the sRGB gamut shown by the triangle. The circle is the CIE illuminant E white point, and the box is the sRGB D65 illuminant white point.

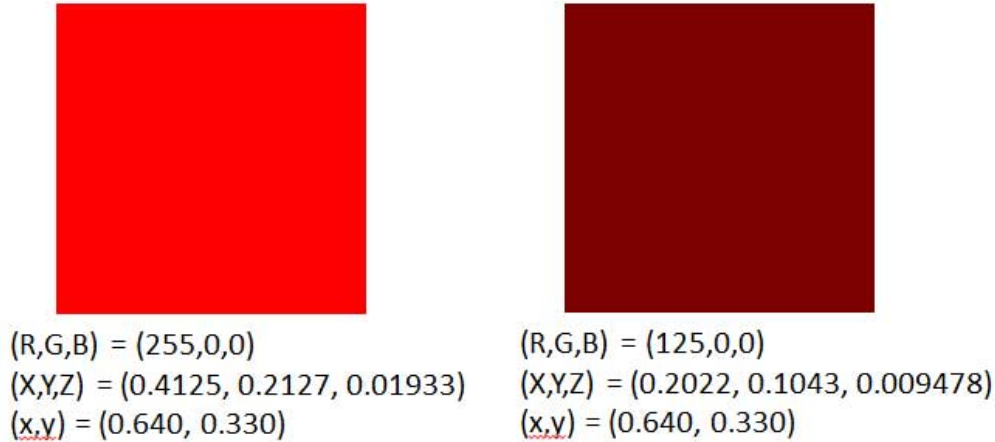


Figure 2: Two color patches with the same chromaticities $(x, y) = (0.64, 0.33)$ but with different luminances $Y = 0.2127$ and $Y = 0.1043$.

color information. The gamma correction for the sRGB color model is given by

$$\begin{aligned} &\text{if } R \leq 0.0031308, \text{ then reset } R \text{ to } 12.92R \\ &\text{if } R > 0.0031308, \text{ then reset } R \text{ to } 1.055R^{1/2.4} - 0.055. \end{aligned}$$

The same transformation is applied to the G and B values. Here the gamma value is $\gamma = 2.4$. After this gamma correction, an input of $(X, Y, Z) = (1, 1, 1)$ results in $(R, G, B) = (255, 230, 225)$ rather than the values of $(255, 201, 192)$ found above using only Eq. (likesection1). Conversely, if gamma-corrected RGB values are to be converted to XYZ values, the gamma correction needs to be undone via

$$\begin{aligned} &\text{if } R \leq 0.04045, \text{ then reset } R \text{ to } R/12.92 \\ &\text{if } R > 0.04045, \text{ then reset } R \text{ to } \frac{(R + 0.055)^{2.4}}{1.055}. \end{aligned}$$

The same transformation is applied to the gamma-corrected G and B values. The resulting “linear” RGB values can then be used in Eq. (likesection2).

Second, you often have to make a subjective decision on how to rescale the RGB values to get the right brightness when displayed. For example, Dierssen et al. (2006) found that scaling (X, Y, Z) so that $Y = 0.4$ gave RGB values in the mid-range of brightness and reasonable screen colors from RGB values computed by Eq. (likesection1). The value of 0.4 was an ad hoc choice determined by trial and error for visual appearance of ocean radiance spectra colors. As was seen in Fig. figure2, the brightness affects the perceived color.

Third, note that the 3x3 transformation matrix in Eq. (likesection2) has positive elements. Thus any (R, G, B) triplet of positive numbers generates an (X, Y, Z) triplet of positive numbers, i.e. a valid point (x, y) in the CIE chromaticity diagram. That is to say, any RGB color can be represented on the CIE diagram. However, some of the matrix elements in Eq. (likesection1) are negative. This means that not every valid (X, Y, Z) or (x, y) can be represented by valid (R, G, B) values; some of the $R, G,$ and B values may be

negative. When that happens, about all you can do is set the negative value to 0 and hope that the displayed color is not too different from what your eye would see in nature.

Fourth, color reproduction gets even more complicated when printing. Computer monitors are emitting light, and various colors are generated by adding together different amounts of red, green, and blue light. Thus yellow is generated by turning on the red and green pixels. The RGB system is an additive color system. When viewing colors on a printed page, you are seeing reflected light. The dyes in the ink absorb, or subtract, various colors from the incident (usually white) light. The printing industry thus deals with subtractive colors, e.g., green is generated by subtracting (absorbing) blue and red from white. Color printers thus use the CMYK (cyan, magenta, yellow, black) subtractive color system, which specifies how much ink of various colors to put onto the page, rather than how much light to generate. Some printers have additional ink colors. Thus the color space or gamut for a printer, when plotted on a CIE diagram, will have more than three vertices if the printer uses more than three colors of ink. This can give better coverage of the human vision color space than can a three-color RGB system. Needless to say, converting a computer screen image from RGB to CMYK values is a messy business, and full account of the reference illuminants and color models for the monitor and the printer must be taken into account. This is what is done in PhotoShop, for example, when you do a “save as CMYK” on an RGB jpeg file. The resulting printed image may or may not be satisfactory, depending on whether or not the underlying color models are consistent with the printer to be used.

The HydroLight radiative transfer model computes the (x, y, Y) values for L_w, L_u, E_d , and E_u whenever the run covers at least the 400-700 nm range. These quantities are uniquely determined for each spectrum. However, HydroLight does not attempt to compute corresponding RGB values because those values are device dependent and may require subjective scaling to get good color reproduction. If you want to turn your spectra into RGB or CMYK values, you can start with the HydroLight-computed (X, Y, Z) values, but beyond that you are on your own. Note that given (x, y, Y) , you can recover (X, Y, Z) from

$$\begin{aligned} X &= \& \frac{xY}{y} \\ Y &= \& Y \\ Z &= \& \frac{(1 - x - y)Y}{y} \end{aligned}$$

where Y is in the range $[0, 1]$.

This is enough to say about colors. This page barely gets you started into the very complicated business of color management and rendering on specific monitors or printers. The previously mentioned web site maintained by Bruce Lindstrom is an excellent place to start if you wish to dig deeper into these topics.