HSL and HSV

From Wikipedia, the free encyclopedia

HSL and **HSV** are the two most common cylindrical-coordinate representations of points in an RGB color model. The two representations rearrange the geometry of RGB in an attempt to be more intuitive and perceptually relevant than the cartesian (cube) representation. Developed in the 1970s for computer graphics applications, HSL and HSV are used today in color pickers, in image editing software, and less commonly in image analysis and computer vision.

HSL stands for *hue*, *saturation*, and *lightness* (or *luminosity*), and is also often called **HLS**. **HSV** stands for *hue*, *saturation*, and *value*, and is also often called **HSB** (*B* for *brightness*). A third model, common in computer vision applications, is **HSI** (*I* for *intensity*). However, while typically consistent, these definitions are not standardized, and any of these abbreviations might be used for any of these three or several other related cylindrical models. (For technical definitions of these terms, see below.)

In each cylinder, the angle around the central vertical axis corresponds to "hue", the distance from the axis corresponds to "saturation", and the distance along the axis corresponds to "lightness", "value" or "brightness". Note that while "hue" in HSL and HSV refers to the same attribute, their definitions of "saturation" differ dramatically.

Because HSL and HSV are simple transformations of device-dependent RGB models, the physical colors they define depend on the colors of the red, green, and blue primaries of the device or of the particular RGB space, and on the gamma correction used to represent the amounts of those primaries. As a result, each unique RGB device has unique HSL and HSV absolute color spaces to accompany it (just as it has unique RGB absolute color space to accompany it), and the same numerical HSL or HSV values (just as numerical RGB values) may be displayed differently by different devices.

Both of these representations are used widely in computer graphics, and one or the other of them is often more convenient than RGB, but both are also criticized for not adequately separating color-making attributes, or for their lack of perceptual uniformity. Other more computationally intensive models, such as CIELAB or CIECAM02, are said to better achieve these goals.
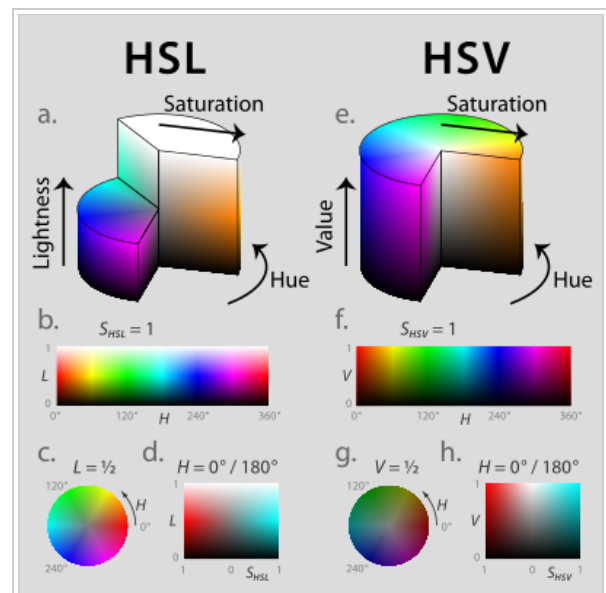

Fig. 1. HSL (a–d) and HSV (e–h). Above (a, e): cut-away 3D models of each. Below: two-dimensional plots showing two of a model's three parameters at once, holding the other constant: cylindrical shells (b, f) of constant saturation, in this case the outside surface of each cylinder; horizontal cross-sections (c, g) of constant HSL lightness or HSV value, in this case the slices halfway down each cylinder; and rectangular vertical cross-sections (d, h) of constant hue, in this case of hues 0° red and its complement 180° cyan.

Contents

Basic principle

HSL and HSV are both cylindrical geometries (fig. 2), with hue, their angular dimension, starting at the red primary at 0°, passing through the green primary at 120° and the blue primary at 240°, and then wrapping back to red at 360°. In each geometry, the central vertical axis comprises the *neutral*, *achromatic*, or *gray* colors, ranging from black at lightness 0 or value 0, the bottom, to white at lightness 1 or value 1, the top.

Fig. 2a. HSL cylinder.   Fig. 2b. HSV cylinder.

In both geometries, the additive primary and secondary colors—red, yellow, green, cyan, blue and magenta—and linear mixtures between adjacent pairs of them, sometimes called *pure colors*, are arranged around the outside edge of the cylinder with saturation 1. These saturated colors have lightness ½ in HSL, while in HSV they have value 1. Mixing these pure colors with black—producing so-called *shades*—leaves saturation unchanged. In HSL, saturation is also unchanged by *tinting* with white, and only mixtures with both black and white—called *tones*—have saturation less than 1. In HSV, tinting alone reduces saturation.

Because these definitions of saturation—in which very dark (in both models) or very light (in HSL) near-neutral colors are considered fully saturated (for instance, ☐ from the bottom right in the sliced HSL cylinder or ☐ from the top right)—conflict with the intuitive notion of color purity, often a conic or bi-conic solid is drawn instead (fig. 3), with what this article calls *chroma* as its radial dimension, instead of saturation. Confusingly, such diagrams usually label this radial dimension "saturation", blurring or erasing the distinction between saturation and chroma.[1] As described below, computing chroma is a helpful step in the derivation of each model. Because such an intermediate model—with dimensions hue, chroma, and HSV value or HSL lightness—takes the shape of a cone or bicone, HSV is often called the "hexcone model" while HSL is often called the "bi-hexcone model" (fig. 8).[2]

Fig. 3a–b. If we plot hue and (a) HSL lightness or (b) HSV value against chroma rather than saturation, the resulting solid is a bicone or cone, respectively, not a cylinder. Such diagrams often claim to represent HSL or HSV directly, with the chroma dimension deceptively labeled "saturation".

Motivation

Most televisions, computer displays, and projectors produce colors by combining red, green, and blue light in varying intensities—the so-called RGB additive primary colors. The resulting mixtures in RGB color space can reproduce a wide variety of colors (called a gamut); however, the relationship between the constituent amounts of red, green, and blue light and the resulting color is unintuitive, especially for inexperienced users, and for users familiar with subtractive color mixing of paints or traditional artists' models based on tints and shades (fig. 4). Furthermore, neither additive nor subtractive color models define color relationships the same way the human eye does.[6]

For example, imagine we have an RGB display whose color is controlled by three sliders ranging from 0–255, one controlling the intensity of each of the red, green, and blue primaries. If we begin with a relatively colorful orange ☐, with sRGB values $R = 217$, $G = 118$, $B = 33$, and want to reduce its colorfulness by half to a less saturated orange ☐, we would need to drag the sliders to decrease R by 31, increase G by 24, and increase B by 59, as pictured below.
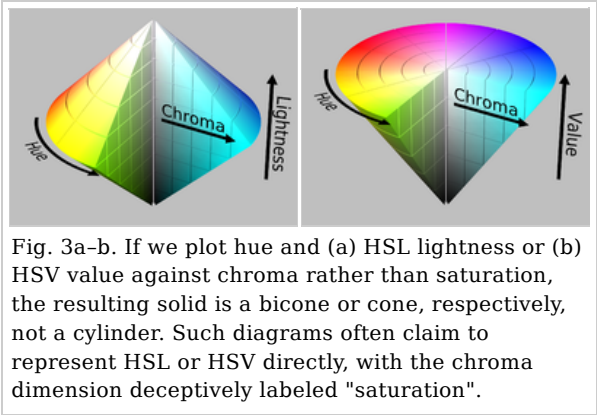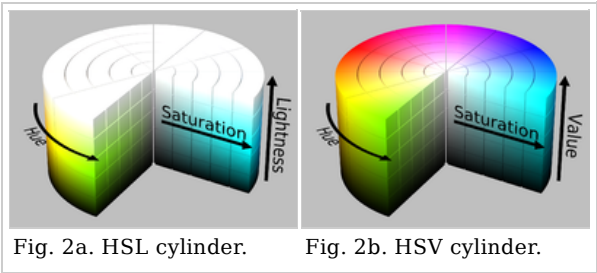
In an attempt to accommodate more traditional and intuitive color mixing models, computer graphics pioneers at PARC and NYIT developed the HSV model in the mid-1970s, formally described by Alvy Ray Smith in the August 1978 issue of *Computer Graphics*. In the same issue, Joblove and Greenberg described the HSL model—whose dimensions they labeled *hue*, *relative chroma*, and *intensity*—and compared it to HSV (fig. 1). Their model was based more upon how colors are organized and conceptualized in human vision in terms of other color-making attributes, such as hue, lightness, and chroma; as well as upon traditional color mixing methods—e.g., in painting—that involve mixing brightly colored pigments with black or white to achieve lighter, darker, or less colorful colors.

The following year, 1979, at SIGGRAPH, Tektronix introduced graphics terminals using HSL for color designation, and the Computer Graphics Standards Committee recommended it in their annual status report (fig. 7). These models were useful not only because they were more intuitive than raw RGB values, but also because the conversions to and from RGB were extremely fast to compute: they could run in real time on the hardware of the 1970s. Consequently, these models and similar ones have become ubiquitous throughout image editing and graphics software since then. Some of their uses are described below.[7][8][9][10][11][12]
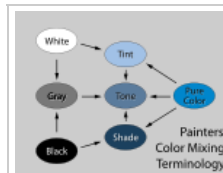
Fig. 4. Painters long mixed colors by combining relatively bright pigments with black and white. Mixtures with white are called *tints*, mixtures with black are called *shades*, and mixtures with both are called *tones*. See Tints and shades.[3]
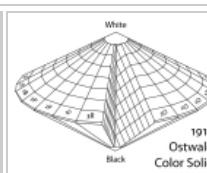
Fig. 5. This 1916 color model by German chemist Wilhelm Ostwald exemplifies the "mixtures with white and black" approach, organizing 24 "pure" colors into a hue circle, and colors of each hue into a triangle. The model thus takes the shape of a bicone.[4]
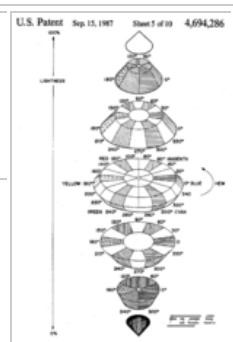
Fig. 7. Tektronix graphics terminals used the earliest commercial implementation of HSL, in 1979. This diagram, from a patent filed in 1983, shows the bicone geometry underlying the model.[5]
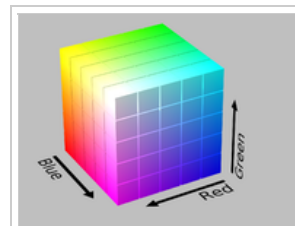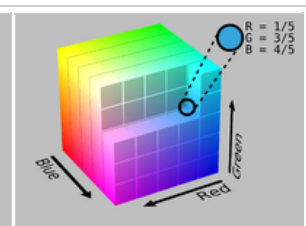
Fig. 6a. The RGB gamut can be arranged in a cube.

Fig. 6b. The same image, with a portion removed for clarity.

Formal derivation

Color-making attributes

The dimensions of the HSL and HSV geometries—simple transformations of the not-perceptually-based RGB model—are not directly related to the photometric color-making attributes of the same names, as defined by scientists such as the CIE or ASTM. Nonetheless, it is worth reviewing those definitions before leaping into the derivation of our models.[13][14][15]

Hue
> The "attribute of a visual sensation according to which an area appears to be similar to one of the perceived colors: red, yellow, green, and blue, or to a combination of two of them".

Radiance ($L_{e,\Omega}$)
> The radiant power of light passing through a particular surface per unit solid angle per unit projected area, measured in SI units in watt per steradian per square metre ($W \cdot sr^{-1} \cdot m^{-2}$).

Luminance (Y or $L_{v,\Omega}$)
> The radiance weighted by the effect of each wavelength on a typical human observer, measured in SI units in candela per square meter (cd/m^2). Often the term *luminance* is used for the relative luminance, Y/Y_n, where Y_n is the luminance of the reference white point.

Luma (Y')
> The weighted sum of gamma-corrected R', G', and B' values, and used in Y′CbCr, for JPEG compression and video transmission.

Brightness
> The "attribute of a visual sensation according to which an area appears to emit more or less light".

Lightness, value
> The "brightness relative to the brightness of a similarly illuminated white".

Colorfulness
> The "attribute of a visual sensation according to which the perceived color of an area appears to be more or less chromatic".

Chroma
> The "colorfulness relative to the brightness of a similarly illuminated white".

Saturation
> The "colorfulness of a stimulus relative to its own brightness".

Brightness and *colorfulness* are absolute measures, which usually describe the spectral distribution of light entering the eye, while *lightness* and *chroma* are measured relative to some white point, and are thus often used for descriptions of surface colors, remaining roughly constant even as brightness and colorfulness change with different illumination. *Saturation* can be defined as either the ratio of colorfulness to brightness or of chroma to lightness.

General approach

HSL, HSV, and related models can be derived via geometric strategies, or can be thought of as specific instances of a "generalized LHS model". The HSL and HSV model-builders took an RGB cube—with constituent amounts of red, green, and blue light in a color denoted $R, G, B \in [0, 1]$[16]—and tilted it on its corner, so that black rested at the origin with white directly above it along the vertical axis, then measured the hue of the colors in the cube by their angle around that axis, starting with red at 0°. Then they came up with a characterization of brightness/value/lightness, and defined saturation to range from 0 along the axis to 1 at the most colorful point for each pair of other parameters.[3][8][9]

Fig. 8. The geometric derivation of the cylindrical HSL and HSV representations of an RGB "colorcube".

Hue and chroma

In each of our models, we calculate both *hue* and what this article will call *chroma*, after Joblove and Greenberg, in the same way—that is, the hue of a color has the same numerical values in all of these models, as does its chroma. If we take our tilted RGB cube, and project it onto the "chromaticity plane" perpendicular to the neutral axis, our projection takes the shape of a hexagon, with red, yellow, green, cyan, blue, and magenta at its corners (fig. 9). *Hue* is roughly the angle of the vector to a point in the projection, with red at 0°, while *chroma* is roughly the distance of the point from the origin.[17][18]

More precisely, both hue and chroma in this model are defined with respect to the hexagonal shape of the projection. The *chroma* is the proportion of the distance from the origin to the edge of the hexagon. In the lower part of the adjacent diagram, this is the ratio of lengths *OP/OP′*, or alternately the ratio of the radii of the two hexagons. This ratio is the difference between the largest and smallest values among *R*, *G*, or *B* in a color. To make our definitions easier to write, we'll define these maximum, minimum, and chroma component values as *M*, *m*, and *C*, respectively.[19]

$$M = \max(R, G, B)$$
$$m = \min(R, G, B)$$
$$C = M - m$$

To understand why chroma can be written as $M - m$, notice that any neutral color, with $R = G = B$, projects onto the origin and so has 0 chroma. Thus if we add or subtract the same amount from all three of *R*, *G*, and *B*, we move vertically within our tilted cube, and do not change the projection. Therefore, any two colors (R, G, B) and $(R - m, G - m, B - m)$ project on the same point, and have the same chroma. The chroma of a color with one of its components equal to zero ($m = 0$) is simply the maximum of the other two components. This chroma is *M* in the particular case of a color with a zero component, and $M - m$ in general.

The *hue* is the proportion of the distance around the edge of the hexagon which passes through the projected point, originally measured on the range [0, 1] but now typically measured in degrees [0°, 360°). For points which project onto the origin in the chromaticity plane (i.e., grays), hue is undefined. Mathematically, this definition of hue is written piecewise:[20]
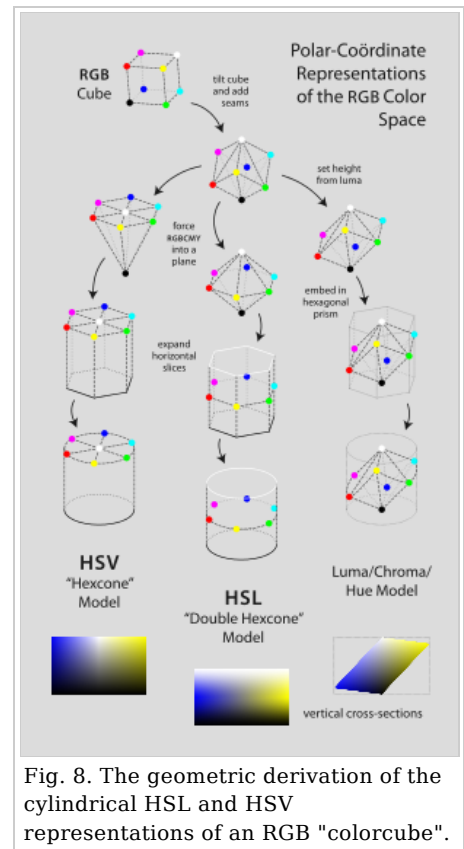
$$H' = \begin{cases} \text{undefined,} & \text{if } C = 0 \\ \frac{G-B}{C} \bmod 6, & \text{if } M = R \\ \frac{B-R}{C} + 2, & \text{if } M = G \\ \frac{R-G}{C} + 4, & \text{if } M = B \end{cases}$$

$$H = 60° \times H'$$

Sometimes, neutral colors (i.e. with $C = 0$) are assigned a hue of 0° for convenience of representation.

These definitions amount to a geometric warping of hexagons into circles: each side of the hexagon is mapped linearly onto a 60° arc of the circle (fig. 10). After such a transformation, hue is precisely the angle around the origin and chroma the distance from the origin: the angle and magnitude of the vector pointing to a color.

Sometimes for image analysis applications, this hexagon-to-circle transformation is skipped, and *hue* and *chroma* (we'll denote these H_2 and C_2) are defined by the usual cartesian-to-polar coordinate transformations (fig. 11). The easiest way to derive those is via a pair of cartesian chromaticity coordinates which we'll call α and β:[21][22][23]

$$\alpha = \tfrac{1}{2}(2R - G - B)$$
$$\beta = \tfrac{\sqrt{3}}{2}(G - B)$$
$$H_2 = \text{atan2}(\beta, \alpha)$$
$$C_2 = \sqrt{\alpha^2 + \beta^2}$$

(The atan2 function, a "two-argument arctangent", computes the angle from a cartesian coordinate pair.)

Notice that these two definitions of hue (H and H_2) nearly coincide, with a maximum difference between them for any color of about 1.12°—which occurs at twelve particular hues, for instance $H = 13.38°$, $H_2 = 12.26°$—and with $H = H_2$ for every multiple of 30°. The two definitions of chroma (C and C_2) differ more substantially: they are equal at the corners of our hexagon, but at points halfway between two corners, such as $H = H_2 = 30°$, we have $C = 1$, but $C_2 = \sqrt{3/4} \approx 0.866$, a difference of about 13.4%.

$$C = \frac{OP}{OP'} = B - R = \tfrac{4}{5} - \tfrac{1}{5} = \tfrac{3}{5} = .6$$

$R = \tfrac{1}{5}$
$G = \tfrac{3}{5}$
$B = \tfrac{4}{5}$

$$H = 60° \times \left(4 + \frac{R - G}{C}\right) = 60° \times (4 - \tfrac{2}{3}) = 200°$$
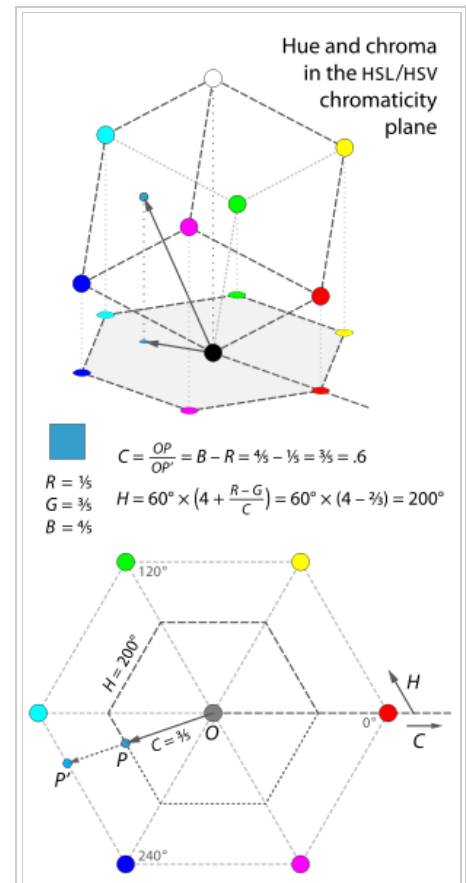
Fig. 9. Both hue and chroma are defined based on the projection of the RGB cube onto a hexagon in the "chromaticity plane". Chroma is the relative size of the hexagon passing through a point, and hue is how far around that hexagon's edge the point lies.
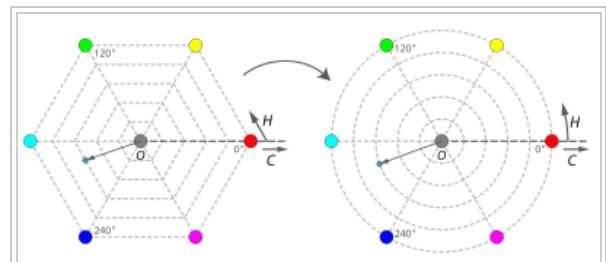

Fig. 10. The definitions of *hue* and *chroma* in HSL and HSV have the effect of warping hexagons into circles.
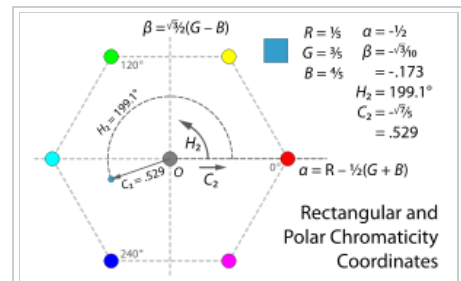
Fig. 11. Constructing rectangular chromaticity coordinates α and β, and then transforming those into hue H_2 and chroma C_2 yields slightly different values than computing hexagonal hue H and chroma C: compare the numbers in this diagram to those earlier in this section.

Lightness

While the definition of *hue* is relatively uncontroversial—it roughly satisfies the criterion that colors of the same perceived hue should have the same numerical hue—the definition of a *lightness* or *value* dimension is less obvious: there are several possibilities depending on the purpose and goals of the representation. Here are four of the most common (fig. 12; three of these are also shown in fig. 8):

Fig. 12a–d. Four different possible "lightness" dimensions, plotted against chroma, for a pair of complementary hues. Each plot is a vertical cross-section of its three-dimensional color solid.

- The simplest definition is just the average of the three components, in the HSI model called *intensity* (fig. 12a). This is simply the projection of a point onto the neutral axis—the vertical height of a point in our tilted cube. The advantage is that, together with Euclidean-distance calculations of hue and chroma, this representation preserves distances and angles from the geometry of the RGB cube.[22][24]
  $$I = \tfrac{1}{3}(R + G + B)$$
- In the HSV "hexcone" model, *value* is defined as the largest component of a color, our M above (fig. 12b). This places all three primaries, and also all of the "secondary colors"—cyan, yellow, and magenta—into a plane with white, forming a hexagonal pyramid out of the RGB cube.[8]
  $$V = M$$
- In the HSL "bi-hexcone" model, *lightness* is defined as the average of the largest and smallest color components (fig. 12c). This definition also puts the primary and secondary colors into a plane, but a plane passing halfway between white and black. The resulting color solid is a double-cone similar to Ostwald's, shown above.[9]
  $$L = \tfrac{1}{2}(M + m)$$
- A more perceptually relevant alternative is to use *luma*, Y', as a lightness dimension (fig. 12d). Luma is the weighted average of gamma-corrected R, G, and B, based on their contribution to perceived luminance, long used as the monochromatic dimension in color television broadcast. For the Rec. 709 primaries used in sRGB, $Y'_{709} = 0.21R + 0.72G + 0.07B$; for the Rec. 601 NTSC primaries, $Y'_{601} \approx 0.30R + 0.59G + 0.11B$; for other primaries different coefficients should be used.[14][25][26]
  $$Y'_{601} = 0.299R + 0.587G + 0.114B$$

All four of these leave the neutral axis alone. That is, for colors with $R = G = B$, any of the four formulations yields a lightness equal to the value of R, G, or B.
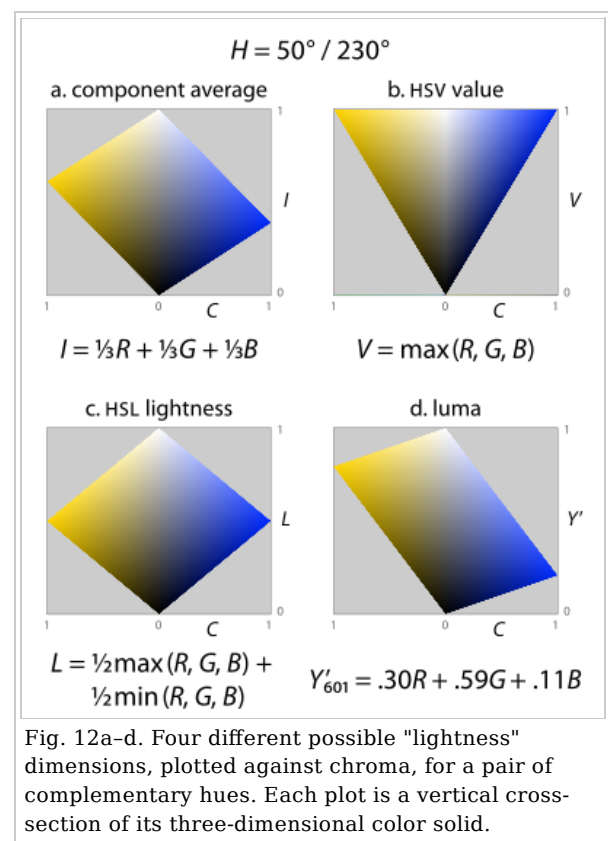
For a graphical comparison, see fig. 13 below.

Saturation

If we encode colors in a hue/lightness/chroma or hue/value/chroma model (using the definitions from the previous two sections), not all combinations of lightness (or value) and chroma are meaningful: that is, half of the colors we can describe using $H \in [0°, 360°)$, $C \in [0, 1]$, and $V \in [0, 1]$ fall outside the RGB gamut (the gray parts of the slices in figure 14). The creators of these models considered this a problem for some uses. For example, in a color selection interface with two of the dimensions in a rectangle and the third on a slider, half of that rectangle is made of unused space. Now imagine we have a slider for lightness: the user's intent when adjusting this slider is potentially ambiguous: how should the software deal with out-of-gamut colors? Or conversely, If the user has selected as colorful as possible a dark purple ☐, and then shifts the lightness slider upward, what should be done: would the user prefer to see a lighter purple still as colorful as possible for the given hue and lightness ☐, or a lighter purple of exactly the same chroma as the original color ☐?[9]

To solve problems such as these, the HSL and HSV models scale the chroma so that it always fits into the range [0, 1] for every combination of hue and lightness or value, calling the new attribute *saturation* in both cases (fig. 14). To calculate either, simply divide the chroma by the maximum chroma for that value or lightness.
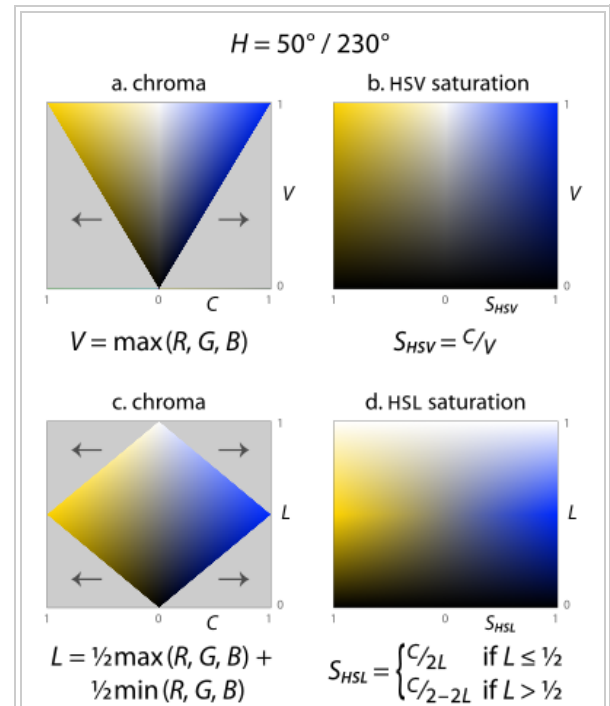

Fig. 14a–d. In both HSL and HSV, *saturation* is simply the chroma scaled to fill the interval [0, 1] for every combination of hue and lightness or value.

$$S_{HSV} = \begin{cases} 0, & \text{if } V = 0 \\ \frac{C}{V}, & \text{otherwise} \end{cases}$$

$$S_{HSL} = \begin{cases} 0, & \text{if } L = 1 \\ \frac{C}{1 - |2L - 1|}, & \text{otherwise} \end{cases}$$

The HSI model commonly used for computer vision, which takes H_2 as a hue dimension and the component average I ("intensity") as a lightness dimension, does not attempt to "fill" a cylinder by its definition of saturation. Instead of presenting color choice or modification interfaces to end users, the goal of HSI is to facilitate separation of shapes in an image. Saturation is therefore defined in line with the psychometric definition: chroma relative to lightness (fig. 15). See the Use in image analysis section of this article.[27]

$$S_{HSI} = \begin{cases} 0, & \text{if } I = 0 \\ 1 - \frac{m}{I}, & \text{otherwise} \end{cases}$$
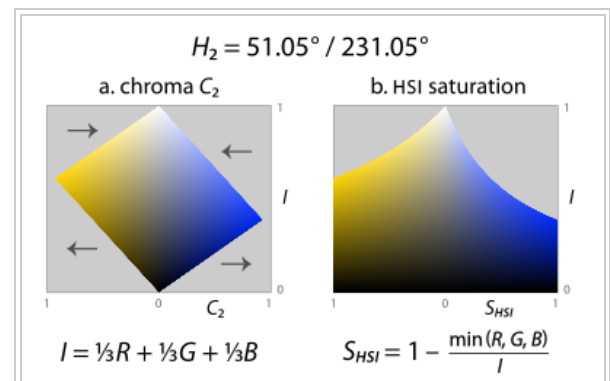

Fig. 15a–b. In HSI, *saturation*, shown in the slice on the right, is roughly the chroma relative to lightness. Also common is a model with dimensions I, H_2, C_2, shown in the slice on the left. Notice that the hue in these slices is the same as the hue above, but H differs slightly from H_2.

Using the same name for these three different definitions of saturation leads to some confusion, as the three attributes describe substantially different color relationships; in HSV and HSI, the term roughly matches the psychometric definition, of a chroma of a color relative to its own lightness, but in HSL it does not come close. Even worse, the word *saturation* is also often used for one of the measurements we call chroma above (C or C_2).

Examples

All parameter values shown below are in the interval [0, 1], except those for H and H_2 which are in the interval [0°, 360°).[28]

Color	R	G	B	H	H_2	C	C_2	V	L	I	Y_{601}	S_{HSV}	S_{HSL}	S_{HSI}
#FFFFFF	1.000	1.000	1.000	n/a	n/a	0.000	0.000	1.000	1.000	1.000	1.000	0.000	0.000	0.000
#808080	0.500	0.500	0.500	n/a	n/a	0.000	0.000	0.500	0.500	0.500	0.500	0.000	0.000	0.000
#000000	0.000	0.000	0.000	n/a	n/a	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
#FF0000	1.000	0.000	0.000	0.0°	0.0°	1.000	1.000	1.000	0.500	0.333	0.299	1.000	1.000	1.000
#BFBF00	0.750	0.750	0.000	60.0°	60.0°	0.750	0.750	0.750	0.375	0.500	0.664	1.000	1.000	1.000
#008000	0.000	0.500	0.000	120.0°	120.0°	0.500	0.500	0.500	0.250	0.167	0.293	1.000	1.000	1.000
#80FFFF	0.500	1.000	1.000	180.0°	180.0°	0.500	0.500	1.000	0.750	0.833	0.850	0.500	1.000	0.400
#8080FF	0.500	0.500	1.000	240.0°	240.0°	0.500	0.500	1.000	0.750	0.667	0.557	0.500	1.000	0.250
#BF40BF	0.750	0.250	0.750	300.0°	300.0°	0.500	0.500	0.750	0.500	0.583	0.457	0.667	0.500	0.571
#A0A424	0.628	0.643	0.142	61.8°	61.5°	0.501	0.494	0.643	0.393	0.471	0.581	0.779	0.638	0.699
#411BEA	0.255	0.104	0.918	251.1°	250.0°	0.814	0.750	0.918	0.511	0.426	0.242	0.887	0.832	0.756
#1EAC41	0.116	0.675	0.255	134.9°	133.8°	0.559	0.504	0.675	0.396	0.349	0.460	0.828	0.707	0.667
#F0C80E	0.941	0.785	0.053	49.5°	50.5°	0.888	0.821	0.941	0.497	0.593	0.748	0.944	0.893	0.911
#B430E5	0.704	0.187	0.897	283.7°	284.8°	0.710	0.636	0.897	0.542	0.596	0.423	0.792	0.775	0.686
#ED7651	0.931	0.463	0.316	14.3°	13.2°	0.615	0.556	0.931	0.624	0.570	0.586	0.661	0.817	0.446
#FEF888	0.998	0.974	0.532	56.9°	57.4°	0.466	0.454	0.998	0.765	0.835	0.931	0.467	0.991	0.363
#19CB97	0.099	0.795	0.591	162.4°	163.4°	0.696	0.620	0.795	0.447	0.495	0.564	0.875	0.779	0.800
#362698	0.211	0.149	0.597	248.3°	247.3°	0.448	0.420	0.597	0.373	0.319	0.219	0.750	0.601	0.533
#7E7EB8	0.495	0.493	0.721	240.5°	240.4°	0.228	0.227	0.721	0.607	0.570	0.520	0.316	0.290	0.135

Use in end-user software

The original purpose of HSL and HSV and similar models, and their most common current application, is in color selection tools. At their simplest, some such color pickers provide three sliders, one for each attribute. Most, however, show a two-dimensional slice through the model, along with a slider controlling which particular slice is shown. The latter type of GUI exhibits great variety, because of the choice of cylinders, hexagonal prisms, or cones/bicones that the models suggest (see the diagram near the top of the page). Several color choosers from the 1990s are shown to the right, most of which have remained nearly unchanged in the intervening time: today, nearly every computer color chooser uses HSL or HSV, at least as an option. Some more sophisticated variants are designed for choosing whole sets of colors, basing their suggestions of compatible colors on the HSL or HSV relationships between them.[30]

Most web applications needing color selection also base their tools on HSL or HSV, and pre-packaged open source color choosers exist for most major web front-end frameworks. The CSS 3 specification allows web authors to specify colors for their pages directly with HSL coordinates.[31]

HSL and HSV are sometimes used to define gradients for data visualization, as in maps or medical images. For example, the popular GIS program ArcGIS historically applied customizable HSV-based gradients to numerical geographical data.[32]
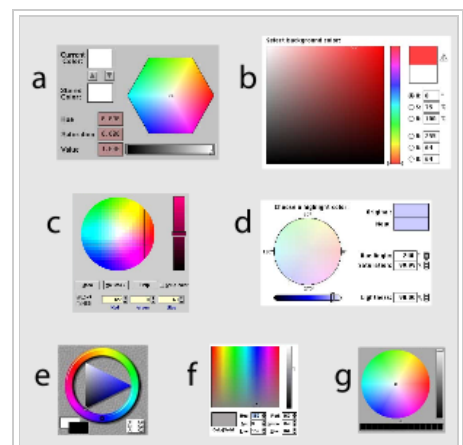

Fig 16a–g. By the 1990s, HSL and HSV color selection tools were ubiquitous. The screenshots above are taken from:
· (a) SGI IRIX 5, c. 1995;
· (b) Adobe Photoshop, c. 1990;
· (c) IBM OS/2 Warp 3, c. 1994;
· (d) Apple Macintosh System 7, c. 1996;
· (e) Fractal Design Painter, c. 1993;
· (f) Microsoft Windows 3.1, c. 1992;
· (g) NeXTSTEP, c. 1995. These are undoubtedly based on earlier examples, stretching back to PARC and NYIT in the mid-1970s.[29]

Image editing software also commonly includes tools for adjusting colors with reference to HSL or HSV coordinates, or to coordinates in a model based on the "intensity" or luma defined above. In particular, tools with a pair of "hue" and "saturation" sliders are commonplace, dating to at least the late-1980s, but various more complicated color tools have also been implemented. For instance, the Unix image viewer and color editor xv allowed six user-definable hue (H) ranges to be rotated and resized, included a dial-like control for saturation (S_{HSV}), and a curves-like interface for controlling value (V)—see fig. 17. The image editor Picture Window Pro includes a "color correction" tool which affords complex remapping of points in a hue/saturation plane relative to either HSL or HSV space.[33]

Video editors also use these models. For example, both Avid and Final Cut Pro include color tools based on HSL or a similar geometry for use adjusting the color in video. With the Avid tool, users pick a vector by clicking a point within the hue/saturation circle to shift all the colors at some lightness level (shadows, mid-tones, highlights) by that vector.

Fig. 18. The hue/saturation tool in Photoshop 2.5, ca. 1992.

Fig. 17. xv's HSV-based color modifier.

Fig. 19. Avid's video color adjustment tool, based on HSL or a similar model.

Since version 4.0, Adobe Photoshop's "Luminosity", "Hue", "Saturation", and "Color" blend modes composite layers using a luma/chroma/hue color geometry. These have been copied widely, but several imitators use the HSL (e.g. PhotoImpact, Paint Shop Pro) or HSV (e.g. GIMP) geometries instead.[34]

Use in image analysis

HSL, HSV, HSI, or related models are often used in computer vision and image analysis for feature detection or image segmentation. The applications of such tools include object detection, for instance in robot vision; object recognition, for instance of faces, text, or license plates; content-based image retrieval; and analysis of medical images.[27]

For the most part, computer vision algorithms used on color images are straightforward extensions to algorithms designed for grayscale images, for instance k-means or fuzzy clustering of pixel colors, or canny edge detection. At the simplest, each color component is separately passed through the same algorithm. It is important, therefore, that the features of interest can be distinguished in the color dimensions used. Because the R, G, and B components of an object's color in a digital image are all correlated with the amount of light hitting the object, and therefore with each other, image descriptions in terms of those components make object discrimination difficult. Descriptions in terms of hue/lightness/chroma or hue/lightness/saturation are often more relevant.[27]

Starting in the late 1970s, transformations like HSV or HSI were used as a compromise between effectiveness for segmentation and computational complexity. They can be thought of as similar in approach and intent to the neural processing used by human color vision, without agreeing in particulars: if the goal is object detection, roughly separating hue, lightness, and chroma or saturation is effective, but there is no particular reason to strictly mimic human color response. John Kender's 1976 master's thesis proposed the HSI model. Ohta et al. (1980) instead used a model made up of dimensions similar to those we have called I, α, and β. In recent years, such models have continued to see wide use, as their performance compares favorably with more complex models, and their computational simplicity remains compelling.[35][27][36][37][38]

Disadvantages

While HSL, HSV, and related spaces serve well enough to, for instance, choose a single color, they ignore much of the complexity of color appearance. Essentially, they trade off perceptual relevance for computation speed, from a time in computing history (high-end 1970s graphics workstations, or mid-1990s consumer desktops) when more sophisticated models would have been too computationally expensive.[39]

HSL and HSV are simple transformations of RGB which preserve symmetries in the RGB cube unrelated to human perception, such that its R, G, and B corners are equidistant from the neutral axis, and equally spaced around it. If we plot the RGB gamut in a more perceptually-uniform space, such as CIELAB (see below), it becomes immediately clear that the red, green, and blue primaries do not have the same lightness or chroma, or evenly spaced hues. Furthermore, different RGB displays use different primaries, and so have different gamuts. Because HSL and HSV are defined purely with reference to some RGB space, they are not absolute color spaces: to specify a color precisely requires reporting not only HSL or HSV values, but also the characteristics of the RGB space they are based on, including the gamma correction in use.
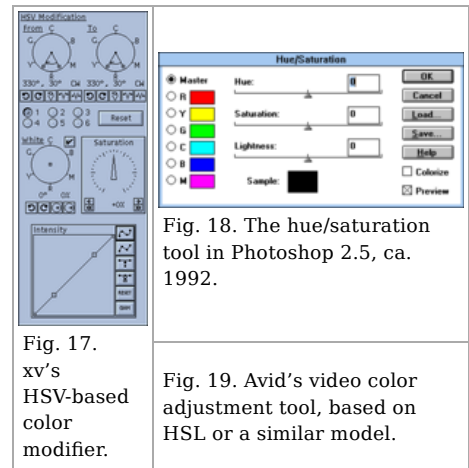
If we take an image and extract the hue, saturation, and lightness or value components, and then compare these to the components of the same name as defined by color scientists, we can quickly see the difference, perceptually. For example, examine the following images of a fire breather (fig. 13). CIELAB $L*$ is a CIE-defined quantity intended to match perceptual lightness response, and it is plain that $L*$ appears similar in lightness to the original color image. Luma is roughly similar, but differs somewhat at high chroma. HSL L and HSV V, by contrast, diverge substantially from perceptual lightness.
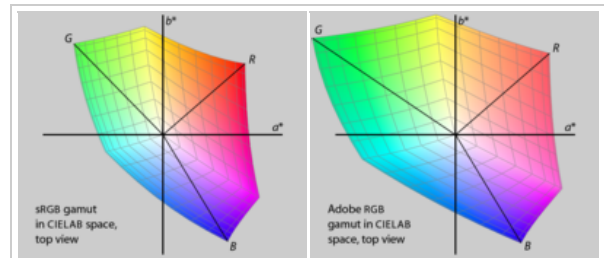
Fig 20a. The sRGB gamut mapped in CIELAB space. Notice that the lines pointing to the red, green, and blue primaries are not evenly spaced by hue angle, and are of unequal length. The primaries also have different $L*$ values.

Fig 20b. The Adobe RGB gamut mapped in CIELAB space. Also notice that these two RGB spaces have different gamuts, and thus will have different HSL and HSV representations.

Fig. 13a. Color photograph.

Fig. 13b. CIELAB $L*$.

Fig. 13c. Rec. 601 luma Y'.

Fig. 13d. Component average: "intensity" I.

Fig. 13e. HSV value V.

Fig. 13f. HSL lightness L.

Though none of the dimensions in these spaces match their perceptual analogs, the *value* of HSV and the *saturation* of HSL are particular offenders. In HSV, the blue primary □ and white □ are held to have the same value, even though perceptually the blue primary has somewhere around 10% of the luminance of white (the exact fraction depends on the particular RGB primaries in use). In HSL, a mix of 100% red, 100% green, 90% blue—that is, a very light yellow □—is held to have the same saturation as the green primary □, even though the former color has almost no chroma or saturation by the conventional psychometric definitions. Such perversities led Cynthia Brewer, expert in color scheme choices for maps and information displays, to tell the American Statistical Association:

> Computer science offers a few poorer cousins to these perceptual spaces that may also turn up in your software interface, such as HSV and HLS. They are easy mathematical transformations of RGB, and they seem to be perceptual systems because they make use of the hue–lightness/value–saturation terminology. But take a close look; don't be fooled. Perceptual color dimensions are poorly scaled by the color specifications that are provided in these and some other systems. For example, saturation and lightness are confounded, so a saturation scale may also contain a wide range of lightnesses (for example, it may progress from white to green which is a combination of both lightness and saturation). Likewise, hue and lightness are confounded so, for example, a saturated yellow and saturated blue may be designated as the same 'lightness' but have wide differences in perceived lightness. These flaws make the systems difficult to use to control the look of a color scheme in a systematic manner. If much tweaking is required to achieve the desired effect, the system offers little benefit over grappling with raw specifications in RGB or CMY.[40]

If these problems make HSL and HSV problematic for choosing colors or color schemes, they make them much worse for image adjustment. HSL and HSV, as Brewer mentioned, confound perceptual color-making attributes, so that changing any dimension results in non-uniform changes to all three perceptual dimensions, and distorts all of the color relationships in the image. For instance, rotating the hue of a pure dark blue ▢ toward green ▢ will also reduce its perceived chroma, and increase its perceived lightness (the latter is grayer and lighter), but the same hue rotation will have the opposite impact on lightness and chroma of a lighter bluish-green—▢ to ▢ (the latter is more colorful and slightly darker). In the example below (fig. 21), the image on the left (a) is the original photograph of a green turtle. In the middle image (b), we have rotated the hue (H) of each color by −30°, while keeping HSV value and saturation or HSL lightness and saturation constant. In the image on the right (c), we make the same rotation to the HSL/HSV hue of each color, but then we force the CIELAB lightness ($L*$, a decent approximation of perceived lightness) to remain constant. Notice how the hue-shifted middle version without such a correction dramatically changes the perceived lightness relationships between colors in the image. In particular, the turtle's shell is much darker and has less contrast, and the background water is much lighter.

Fig. 21a. Color photograph.

Fig. 21b. HSL/HSV hue of each color shifted by −30°.

Fig. 21c. Hue shifted but CIELAB lightness ($L*$) kept as in the original.

Because hue is a circular quantity, represented numerically with a discontinuity at 360°, it is difficult to use in statistical computations or quantitative comparisons: analysis requires the use of circular statistics. Furthermore, hue is defined piecewise, in 60° chunks, where the relationship of lightness, value, and chroma to R, G, and B depends on the hue chunk in question. This definition introduces discontinuities, corners which can plainly be seen in horizontal slices of HSL or HSV.[41][42]

Charles Poynton, digital video expert, lists the above problems with HSL and HSV in his *Color FAQ*, and concludes that:

> HSB and HLS were developed to specify numerical Hue, Saturation and Brightness (or Hue, Lightness and Saturation) in an age when users had to specify colors numerically. The usual formulations of HSB and HLS are flawed with respect to the properties of color vision. Now that users can choose colors visually, or choose colors related to other media (such as PANTONE), or use perceptually-based systems like L*u*v* and L*a*b*, HSB and HLS should be abandoned.[43]

Other cylindrical-coordinate color models

The creators of HSL and HSV were far from the first to imagine colors fitting into conic or spherical shapes, with neutrals running from black to white in a central axis, and hues corresponding to angles around that axis. Similar arrangements date back to the 18th century, and continue to be developed in the most modern and scientific models. A pair of the most influential older models are Philipp Otto Runge's 1810 *Farbenkugel* (color sphere), and the early-20th-century Munsell color system. Albert Munsell began with a spherical arrangement in his 1905 book *A Color Notation*, but he wished to properly separate color-making attributes into separate dimensions, which he called *hue*, *value*, and *chroma*, and after taking careful measurements of perceptual responses, he realized that no symmetrical shape would do, so he reorganized his system into a lumpy blob.[44]
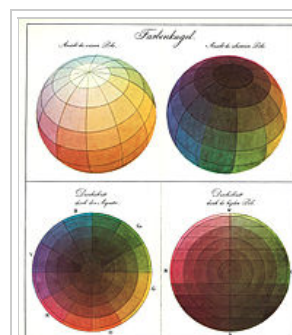
Fig. 22. Runge's *Farbenkugel*, 1810.
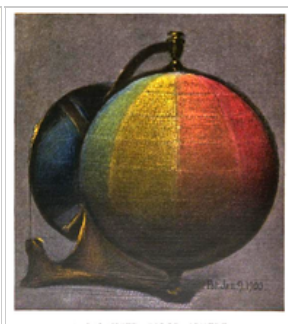
Fig. 23. Munsell's balanced color sphere, 1900, from *A Color Notation*, 1905.

Munsell's system became extremely popular, the de facto reference for American color standards—used not only for specifying the color of paints and crayons, but also, e.g., electrical wire, beer, and soil color—

because it was organized based on perceptual measurements, specified colors via an easily learned and systematic triple of numbers, because the color chips sold in the *Munsell Book of Color* covered a wide gamut and remained stable over time (rather than fading), and because it was effectively marketed by Munsell's Company. In the 1940s, the Optical Society of America made extensive measurements, and adjusted the arrangement of Munsell colors, issuing a set of "renotations". The trouble with the Munsell system for computer graphics applications is that its colors are not specified via any set of simple equations, but only via its foundational measurements: effectively a lookup table. Converting from RGB ↔ Munsell requires interpolating between that table's entries, and is extremely computationally expensive in comparison with converting from RGB ↔ HSL or RGB ↔ HSV which only requires a few simple arithmetic operations.[45][46][47][48]

In densitometry, a model quite similar to the hue defined above is used for describing colors of CMYK process inks. In 1953, Frank Preucil developed two geometric arrangements of hue, the "Preucil hue circle" and the "Preucil hue hexagon", analogous to our H and H_2, respectively, but defined relative to idealized cyan, yellow, and magenta ink colors. The Preucil *hue error* of an ink indicates the difference in the "hue circle" between its color and the hue of the corresponding idealized ink color. The *grayness* of an ink is m/M, where m and M are the minimum and maximum among the amounts of idealized cyan, magenta, and yellow in a density measurement.[49]

The Swedish Natural Color System (NCS), widely used in Europe, takes a similar approach to the Ostwald bicone shown earlier. Because it attempts to fit color into a familiarly shaped solid based on "phenomenological" instead of photometric or psychological characteristics, it suffers from some of the same disadvantages as HSL and HSV: in particular, its lightness dimension differs from perceived lightness, because it forces colorful yellow, red, green, and blue into a plane.[50]

The International Commission on Illumination (CIE) developed the XYZ model for describing the colors of light spectra in 1931, but its goal was to match human visual metamerism, rather than to be perceptually uniform, geometrically. In the 1960s and 70s, attempts were made to transform XYZ colors into a more relevant geometry, influenced by the Munsell system. These efforts culminated in the 1976 CIELUV and CIELAB models. The dimensions of these models—($L*$, $u*$, $v*$) and ($L*$, $a*$, $b*$), respectively—are cartesian, based on the opponent process theory of color, but both are also often described using polar coordinates—($L*$, $C*_{uv}$, $h*_{uv}$) or ($L*$, $C*_{ab}$, $h*_{ab}$) and also HCL, where $L*$ is lightness, $C*$ is chroma, and $h*$ is hue angle. Officially, both CIELAB and CIELUV were created for their color difference metrics $\Delta E*_{ab}$ and $\Delta E*_{uv}$, particularly for use defining color tolerances, but both have become widely used as color order systems and color appearance models, including in computer graphics and computer vision. For example, gamut mapping in ICC color management is usually performed in CIELAB space, and Adobe Photoshop includes a CIELAB mode for editing images. CIELAB and CIELUV geometries are much more perceptually relevant than many others such as RGB, HSL, HSV, YUV/YIQ/YCbCr or XYZ, but are not perceptually perfect, and in particular have trouble adapting to unusual lighting conditions.[45][51][52][53]

The CIE's most recent model, CIECAM02 (CAM stands for "color appearance model"), is more theoretically sophisticated and computationally complex than earlier models. Its aims are to fix several of the problems with models such as CIELAB and CIELUV, and to explain not only responses in carefully controlled experimental environments, but also to model the color appearance of real-world scenes. Its dimensions J (lightness), C (chroma), and h (hue) define a polar-coordinate geometry.[45][50]

Converting to RGB

To convert from HSL or HSV to RGB, we essentially invert the steps listed above (as before, R, G, $B \in [0, 1]$). First, we compute chroma, by multiplying saturation by the maximum chroma for a given lightness or value. Next, we find the point on one of the bottom three faces of the RGB cube which has the same hue and chroma as our color (and therefore projects onto the same point in the chromaticity plane). Finally, we add equal amounts of R, G, and B to reach the proper lightness or value.[18]

From HSL

Given a color with hue $H \in [0°, 360°]$, saturation $S_{HSL} \in [0, 1]$, and lightness $L \in [0, 1]$, we first find chroma:

$$C = (1 - |2L - 1|) \times S_{HSL}$$

Then we can find a point (R_1, G_1, B_1) along the bottom three faces of the RGB cube, with the same hue and chroma as our color (using the intermediate value X for the second largest component of this color):

$$H' = \frac{H}{60°}$$
$$X = C \times (1 - |H' \mod 2 - 1|)$$
$$(R_1, G_1, B_1) = \begin{cases} (0, 0, 0) & \text{if } H \text{ is undefined} \\ (C, X, 0) & \text{if } 0 \le H' \le 1 \\ (X, C, 0) & \text{if } 1 \le H' \le 2 \\ (0, C, X) & \text{if } 2 \le H' \le 3 \\ (0, X, C) & \text{if } 3 \le H' \le 4 \\ (X, 0, C) & \text{if } 4 \le H' \le 5 \\ (C, 0, X) & \text{if } 5 \le H' < 6 \end{cases}$$

Fig. 24. A graphical representation of RGB coordinates given values for HSV.

Overlap (when H' is an integer) occurs because two ways to calculate the value are equivalent: $X = 0$ or $X = C$, as appropriate.

Finally, we can find R, G, and B by adding the same amount to each component, to match lightness:

$$m = L - \tfrac{1}{2}C$$
$$(R, G, B) = (R_1 + m, G_1 + m, B_1 + m)$$

From HSV

Given an HSV color with hue $H \in [0°, 360°]$, saturation $S_{HSV} \in [0, 1]$, and value $V \in [0, 1]$, we can use the same strategy. First, we find chroma:

$$C = V \times S_{HSV}$$

Then we can, again, find a point (R_1, G_1, B_1) along the bottom three faces of the RGB cube, with the same hue and chroma as our color (using the intermediate value X for the second largest component of this color):

$$H' = \frac{H}{60°}$$
$$X = C \times (1 - |H' \mod 2 - 1|)$$
$$(R_1, G_1, B_1) = \begin{cases} (0, 0, 0) & \text{if } H \text{ is undefined} \\ (C, X, 0) & \text{if } 0 \le H' \le 1 \\ (X, C, 0) & \text{if } 1 \le H' \le 2 \\ (0, C, X) & \text{if } 2 \le H' \le 3 \\ (0, X, C) & \text{if } 3 \le H' \le 4 \\ (X, 0, C) & \text{if } 4 \le H' \le 5 \\ (C, 0, X) & \text{if } 5 \le H' < 6 \end{cases}$$

Overlap (when H' is an integer) occurs because two ways to calculate the value are equivalent: $X = 0$ or $X = C$, as appropriate.

Finally, we can find R, G, and B by adding the same amount to each component, to match value:

$$m = V - C$$
$$(R, G, B) = (R_1 + m, G_1 + m, B_1 + m)$$

From HSI

Given an HSI color with hue $H \in [0°, 360°]$, saturation $S_{HSI} \in [0, 1]$, and intensity $I \in [0, 1]$, we can use the same strategy, in a slightly different order:

$$H' = \frac{H}{60°}$$
$$Z = 1 - |H' \mod 2 - 1|$$
$$C = \frac{3 \times I \times S_{HSI}}{1 + Z}$$
$$X = C \times Z$$

Where C is the chroma.

Then we can, again, find a point (R_1, G_1, B_1) along the bottom three faces of the RGB cube, with the same hue and chroma as our color (using the intermediate value X for the second largest component of this color):

$$(R_1, G_1, B_1) = \begin{cases} (0,0,0) & \text{if } H \text{ is undefined} \\ (C,X,0) & \text{if } 0 \leq H' \leq 1 \\ (X,C,0) & \text{if } 1 \leq H' \leq 2 \\ (0,C,X) & \text{if } 2 \leq H' \leq 3 \\ (0,X,C) & \text{if } 3 \leq H' \leq 4 \\ (X,0,C) & \text{if } 4 \leq H' \leq 5 \\ (C,0,X) & \text{if } 5 \leq H' < 6 \end{cases}$$

Overlap (when H' is an integer) occurs because two ways to calculate the value are equivalent: $X = 0$ or $X = C$, as appropriate.

Finally, we can find R, G, and B by adding the same amount to each component, to match lightness:

$$m = I \times (1 - S_{HSI})$$
$$(R, G, B) = (R_1 + m, G_1 + m, B_1 + m)$$

From luma/chroma/hue

Given a color with hue $H \in [0°, 360°]$, chroma $C \in [0, 1]$, and luma $Y_{601} \in [0, 1]$,[54] we can again use the same strategy. Since we already have H and C, we can straightaway find our point (R_1, G_1, B_1) along the bottom three faces of the RGB cube:

$$H' = \frac{H}{60°}$$
$$X = C \times (1 - |H' \mod 2 - 1|)$$
$$(R_1, G_1, B_1) = \begin{cases} (0,0,0) & \text{if } H \text{ is undefined} \\ (C,X,0) & \text{if } 0 \leq H' \leq 1 \\ (X,C,0) & \text{if } 1 \leq H' \leq 2 \\ (0,C,X) & \text{if } 2 \leq H' \leq 3 \\ (0,X,C) & \text{if } 3 \leq H' \leq 4 \\ (X,0,C) & \text{if } 4 \leq H' \leq 5 \\ (C,0,X) & \text{if } 5 \leq H' < 6 \end{cases}$$

Overlap (when H' is an integer) occurs because two ways to calculate the value are equivalent: $X = 0$ or $X = C$, as appropriate.

Then we can find R, G, and B by adding the same amount to each component, to match luma:

$$m = Y'_{601} - (.30R_1 + .59G_1 + .11B_1)$$
$$(R, G, B) = (R_1 + m, G_1 + m, B_1 + m)$$

Swatches

Mouse over the swatches below to see the R, G, and B values for each swatch in a tooltip.

HSL

| | | **H = 180°** (Cyan) | | | | | | | | **H = 210°** (Blue-Cyan) | | | | | | |
| **H = 0°** (Red) | | | **H = 30°** (Yellow-Red) | | | | | | | | | | | | | |

$L \setminus S$	1	¾	½	¼	0	¼	½	¾	1	$L \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1										1									
⅞										⅞									
¾										¾									

⅝									
½									
⅜									
¼									
⅛									
0									

⅝									
½									
⅜									
¼									
⅛									
0									

	$H = 240°$ (Blue)					$H = 60°$ (Yellow)			
$L \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									
⅞									
¾									
⅝									
½									
⅜									
¼									
⅛									
0									

	$H = 270°$ (Magenta-Blue)					$H = 90°$ (Green-Yellow)			
$L \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									
⅞									
¾									
⅝									
½									
⅜									
¼									
⅛									
0									

	$H = 300°$ (Magenta)					$H = 120°$ (Green)			
$L \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									
⅞									
¾									
⅝									
½									
⅜									
¼									
⅛									
0									

	$H = 330°$ (Red-Magenta)					$H = 150°$ (Cyan-Green)			
$L \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									
⅞									
¾									
⅝									
½									
⅜									
¼									
⅛									
0									

HSV

	$H = 180°$ (Cyan)					$H = 0°$ (Red)			
$V \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									
⅞									
¾									
⅝									
½									
⅜									
¼									
⅛									
0									

	$H = 210°$ (Blue-Cyan)					$H = 30°$ (Yellow-Red)			
$V \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									
⅞									
¾									
⅝									
½									
⅜									
¼									
⅛									
0									

	$H = 240°$ (Blue)					$H = 60°$ (Yellow)			
$V \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									

	$H = 270°$ (Magenta-Blue)					$H = 90°$ (Green-Yellow)			
$V \setminus S$	1	¾	½	¼	0	¼	½	¾	1
1									

⅞										⅞							
¾										¾							
⅝										⅝							
½										½							
⅜										⅜							
¼										¼							
⅛										⅛							
0										0							

***H* = 300°** (Magenta)					***H* = 120°** (Green)					***H* = 330°** (Red-Magenta)					***H* = 150°** (Cyan-Green)			
V* \ *S	1	¾	½	¼	0	¼	½	¾	1	***V* \ *S***	1	¾	½	¼	0	¼	½	¾	1
1										1									
⅞										⅞									
¾										¾									
⅝										⅝									
½										½									
⅜										⅜									
¼										¼									
⅛										⅛									
0										0									

Notes and references

1. In Joblove and Greenberg's (1978) paper first introducing HSL, they called HSL lightness "intensity", called HSL saturation "relative chroma", called HSV saturation "saturation" and called HSV value "value". They carefully and unambiguously described and compared three models: hue/chroma/intensity, hue/relative chroma/intensity, and hue/value/saturation. Unfortunately, later authors were less fastidious, and current usage of these terms is inconsistent and often misleading.
2. The name *hexcone* for hexagonal pyramid was coined in Smith, and stuck.
3. Levkowitz and Herman
4. Wilhelm Ostwald (1916). *Die Farbenfibel*. Leipzig.
   Wilhelm Ostwald (1918). *Die Harmonie der Farben*. Leipzig.
5. Gar A. Bergstedt (April 1983). US patent 4694286, "Apparatus and method for modifying displayed color images" (http://www.google.com/patents/about?id=WA8xAAAAEBAJ&dq). Filed 1983-04-08. Issued 1987-09-15. Assigned to Tektronix, Inc.
6. For instance, a 1982 study by Berk, et al., found that users were better at describing colors in terms of HSL than RGB coordinates, after being taught both systems, but were much better still at describing them in terms of the natural-language CNS model (which uses names such as "very dark grayish yellow-green" or "medium strong bluish purple"). This shouldn't be taken as gospel however: a 1987 study by Schwarz, et al., found that users could match colors using RGB controls faster than with HSL controls; a 1999 study by Douglas and Kirkpatrick found that the visual feedback in the user interface mattered more than the particular color model in use, for user matching speed.
   Toby Berk; Arie Kaufman; Lee Brownston (August 1982). "A human factors study of color notation systems for computer graphics". *Communications of the ACM*. **25** (8): 547–550. doi:10.1145/358589.358606.
   Michael W. Schwarz; William B. Cowan; John C. Beatty (April 1987). "An experimental comparison of RGB, YIQ, LAB, HSV, and opponent color models". *ACM Transactions on Graphics*. **6** (2): 123–158. doi:10.1145/31336.31338.
   Sarah A. Douglas; Arthur E. Kirkpatrick (April 1999). "Model and representation: the effect of visual feedback on human performance in a color picker interface". *ACM Transactions on Graphics*. **18** (2): 96–127. doi:10.1145/318009.318011.
7. Maureen C. Stone (August 2001). "A Survey of Color for Computer Graphics" (http://graphics.stanford.edu/courses/cs448b-02-spring/04cdrom.pdf). Course at SIGGRAPH 2001.
8. Smith
9. Joblove and Greenberg
10. Ware Myers (July 1979). "Interactive Computer Graphics: Flying High-Part I". *Computer*. **12** (7): 8–17. doi:10.1109/MC.1979.1658808.
11. N. Magnetat-Thalmann; N. Chourot; D. Thalmann (March 1984). "Colour Gradation, Shading and Texture Using a Limited Terminal". *Computer Graphics Forum*. **3**: 83. doi:10.1111/j.1467-8659.1984.tb00092.x.
12. Computer Graphics Staff (August 1979). "Status Report of the Graphics Standards Planning Committee". *ACM SIGGRAPH Computer Graphics*. **13** (3): 1. doi:10.1145/988497.988498.

13. Mark Fairchild: "Clearly, if color appearance is to be described in a systematic, mathematical way, definitions of the phenomena being described need to be precise and universally agreed upon." For the definitions of color-making attributes which follow, see:
Fairchild 2005, pp. 83–93 (https://books.google.com/books?id=8_TxzK2B-5MC&pg=PT106) (this is the source of the direct quotations).
Kuehni
ASTM (2009). *Standard Terminology of Appearance* (http://www.astm.org/Standards/E284.htm). E284-09a.
CIE (1987). *International Lighting Vocabulary* (http://www.cie.co.at/publ/abst/17-4-89.html). Number 17.4. CIE, 4th edition. ISBN 978-3-900734-07-7.
14. Poynton
15. Sharma, G. (2003). *Digital Color Imaging Handbook*. Boca Raton, FL: CRC Press. ISBN 0-8493-0900-X.
16. In Levkowitz and Herman's formulation, R, G, and B stand for the voltages on the guns of a CRT display, which might have different maxima, and so their cartesian gamut could be a box of any unequal dimensions. Other definitions commonly use integer values in the range [0, 255], storing the value for each component in one byte. We define the RGB gamut to be a unit cube for convenience, because it simplifies and clarifies the math. Also, in general, HSL and HSV are today computed directly from gamma-corrected R', G', and B'—for instance in sRGB space—but, when the models were developed, might have been transformations of a linear RGB space. Early authors don't address gamma correction at all. We will drop the primes, and the labels R, G, and B should be taken to stand for the three attributes of the origin RGB space, whether or not it is gamma corrected.
17. Using the *chroma* here not only agrees with the original Joblove and Greenberg paper, but is also in the proper spirit of the psychometric definition of the term. Some models call this attribute *saturation*—for instance Adobe Photoshop's "Saturation" blend mode—but such use is even more confusing than the use of the term in HSL or HSV, especially when two substantially different definitions are used side by side.
18. Most of the computer graphics papers and books discussing HSL or HSV have a formula or algorithm describing them formally. Our formulas which follow are some mix of those. See, for instance, Agoston or Foley
19. Hanbury and Serra put a great deal of effort into explaining why what we call *chroma* here can be written as $\max(R, G, B) - \min(R, G, B)$, and showing that this value is a seminorm. They reserve the name *chroma* for the Euclidean norm in the chromaticity plane (our C_2), and call this hexagonal distance *saturation* instead, as part of their IHLS model
20. In the following, the multiplication of hue by 60°—that is, 360°/6—can be seen as the hexagonal-geometry analogue of the conversion from radians to degrees, a multiplication by $360°/2\pi$: the circumference of a unit circle is 2π; the circumference of a unit hexagon is 6.
21. Hanbury and Serra (2002)
22. Allan Hanbury (2008). "Constructing cylindrical coordinate colour spaces" (PDF). *Pattern Recognition Letters*. **29** (4): 494. doi:10.1016/j.patrec.2007.11.002.
23. Patrick Lambert; Thierry Carron (1999). "Symbolic fusion of luminance-hue-chroma features for region segmentation". *Pattern Recognition*. **32** (11): 1857. doi:10.1016/S0031-3203(99)00010-2.
24. Rafael C. Gonzalez and Richard Eugene Woods (2008). *Digital Image Processing*, 3rd ed. Upper Saddle River, NJ: Prentice Hall. ISBN 0-13-168728-X. pp. 407–413 (https://books.google.com/books?id=8uGOnjRGEzoC&pg=PA410&lpg=PA410).
25. For a more specific discussion of the term *luma*, see Charles Poynton (1999). "YUV and *luminance* considered harmful: A plea for precise terminology in video" (http://poynton.com/papers/YUV_and_luminance_harmful.html). *www.poynton.com*. Accessed January 2010.
26. See also RGB color space#Specifications. Photoshop exclusively uses the NTSC coefficients for its "Luminosity" blend mode regardless of the RGB color space involved. See:
Bruce Lindbloom (2001-09-25). "Re: Luminosity channel..." (http://lists.apple.com/archives/colorsync-users/2001/Sep/msg00488.html). *colorsync-users mailing list*.
27. Heng-Da Cheng; Xihua Jiang; Angela Sun; Jingli Wang (2001). "Color image segmentation: Advances and prospects". *Pattern Recognition*. **34** (12): 2259. doi:10.1016/S0031-3203(00)00149-7. This computer vision literature review briefly summarizes research in color image segmentation, including that using HSV and HSI representations.
28. The first nine colors in this table were chosen by hand, and the last ten colors were chosen at random.
29. See Smith. Many of these screenshots were taken from the GUIdebook (http://www.guidebookgallery.org/), and the rest were gathered from image search results.
30. For instance, a tool in Illustrator CS4, and Adobe's related web tool, Kuler, both allow users to define color schemes based on HSV relationships, but with a hue circle modified to better match the RYB model used traditionally by painters. The web tools ColorJack (http://www.colorjack.com/sphere/), Color Wizard (http://www.colorsontheweb.com/colorwizard.asp), and ColorBlender (http://www.colorblender.com/) all pick color schemes with reference to HSL or HSV.
31. Try a web search for "«framework name» color picker" for examples for a given framework, or "JavaScript color picker" for general results.
Tantek Çelik, Chris Lilley, and L. David Baron (July 2008). "CSS3 Color Module" (http://www.w3.org/TR/2008/WD-css3-color-20080721/#hsl-color). W3C Working Draft 21 July 2008.
32. ArcGIS calls its map-symbol gradients "color ramps". Current versions of ArcGIS can use CIELAB instead for defining them.
ESRI (January 2008). "Working with color ramps" (http://webhelp.esri.com/arcgisdesktop/9.2/index.cfm?TopicName=Working_with_color_ramps). *ArcGIS 9.2 Help*. Accessed February 2010.
33. For instance, the first version of Photoshop had an HSL-based tool; see "Photoshop hue/saturation" (http://www.guidebookgallery.org/apps/photoshop/huesaturation) in the GUIdebook for screenshots.
John Bradley (1994). *XV 3.10a Manual*. "The HSV Modification Tools" (http://www.trilon.com/xv/manual/xv-3.10a/color-editor-2.html)
Kiril Sinkel (January 2010). *User Guide for Picture Window and Picture Window Pro* (http://www.dl-c.com/Temp/downloads/PW%20Doc/PW70.pdf). Digital Light & Color.

34. Photoshop's documentation explains that, e.g., "Luminosity: Creates a result color with the hue and saturation of the base color and the luminance of the blend color."
"List of Blending Modes" (http://livedocs.adobe.com/en_US/Photoshop/10.0/help.html?content=WSfd1234e1c4b69f30ea53e41001031ab64-77e9.html). *Adobe Help Resource Center*.

35. The Ohta et al. model has parameters $I_1 = (R + G + B)/3$, $I_2 = (R − B)/2$, $I_3 = (2G − R − B)/4$. I_1 is the same as our I, and I_2 and I_3 are similar to our β and α, respectively, except that (a) where α points in the direction of R in the "chromaticity plane", I_3 points in the direction of G, and (b) the parameters have a different linear scaling which avoids the $\sqrt{3}$ of our β.

36. John Kender (1976). "Saturation, hue and normalized color". Carnegie Mellon University, Computer Science Dept. Pittsburgh, PA.

37. Yu-Ichi Ohta; Takeo Kanade; Toshiyuki Sakai (1980). "Color information for region segmentation". *Computer Graphics and Image Processing*. **13** (3): 222. doi:10.1016/0146-664X(80)90047-7.

38. Ffrank Perez; Christof Koch (1994). "Toward color image segmentation in analog VLSI: Algorithm and hardware". *International Journal of Computer Vision*. **12**: 17. doi:10.1007/BF01420983.

39. Most of the disadvantages below are listed in Poynton, though as mere statements, without examples.

40. The text of Brewer's presentation to the ASA is online, "Color Use Guidelines for Data Representation" (http://www.personal.psu.edu/cab38/ColorSch/ASApaper.html). The published paper is Cynthia A. Brewer (1999). "Color Use Guidelines for Data Representation", *Proceedings of the Section on Statistical Graphics*. Alexandria, VA: American Statistical Association. pp. 55–60.

41. Hanbury, Allan (2003). *Circular Statistics Applied to Colour Images*. 8th Computer Vision Winter Workshop. CiteSeerX 10.1.1.4.1381 .

42. Circular statistics are described in the textbook Fisher, Nicholas (1996). *Statistical Analysis of Circular Data*. Cambridge, England: Cambridge University Press.

43. Charles Poynton (1997). "What are HSB and HLS?" (http://www.poynton.com/notes/colour_and_gamma/ColorFAQ.html#RTFToC36) poynton.com

44. Philipp Otto Runge (1810). *Die Farben-Kugel, oder Construction des Verhaeltnisses aller Farben zueinander*. Hamburg, Germany: Perthes.
Albert Henry Munsell (1905). *A Color Notation*. Boston, MA: Munsell Color Company.
See also Fairchild, and Munsell Color System and its references.

45. Fairchild

46. Edward Landa and Mark Fairchild (September–October 2005). "Charting Color from the Eye of the Beholder" (http://www.americanscientist.org/issues/feature/charting-color-from-the-eye-of-the-beholder/5). *American Scientist* **93**(5): 436

47. Dorothy Nickerson (1976). "History of the Munsell Color System". *Color Research and Application*. **1**: 121–130.

48. Sidney Newhall; Dorothy Nickerson; Deane Judd (1943). "Final Report of the OSA Subcommittee on the Spacing of the Munsell Colors". *Journal of the Optical Society of America*. **33** (7): 385. doi:10.1364/JOSA.33.000385.

49. Frank Preucil (1953). "Color Hue and Ink Transfer—Their Relation to Perfect Reproduction". *Proceedings of the 5th Annual Technical Meeting of TAGA*. pp. 102–110.
See also Hue.

50. MacEvoy

51. Kuehni

52. Robert Hunt (2004). *The Reproduction of Colour*. 6th ed. MN: Voyageur Press. ISBN 0-86343-368-5.

53. MacEvoy 2010. "Modern Color Models" (http://www.handprint.com/HP/WCL/color7.html#CIELUV)
Adobe (January 2007). "The Lab Color Mode in Photoshop" (http://web.archive.org/web/20081207061220/http://kb.adobe.com/selfservice/viewContent.do?externalId=310838). Technical Note 310838.
Steven K. Shevell (2003) *The Science of Color*. 2nd ed. Elsevier Science & Technology. ISBN 0-444-51251-9. pp. 202–206 (https://books.google.com/books?id=G1TC1uXb7awC&pg=PA201)
See also CIELAB, CIELUV, Color difference, Color management, and their references.

54. Some points in this cylinder fall out of gamut.

Bibliography

- Agoston, Max K. (2005). *Computer Graphics and Geometric Modeling: Implementation and Algorithms*. London: Springer. pp. 300–306. ISBN 1-85233-818-0. Agoston's book contains a description of HSV and HSL, and algorithms in pseudocode for converting to each from RGB, and back again.

- Fairchild, Mark D. (2005). *Color Appearance Models* (2nd ed.). Addison-Wesley. This book doesn't discuss HSL or HSV specifically, but is one of the most readable and precise resources about current color science.

- Foley, J. D.; et al. (1995). *Computer Graphics: Principles and Practice* (2nd ed.). Redwood City, CA: Addison-Wesley. ISBN 0-201-84840-6. The standard computer graphics textbook of the 1990s, this tome has a chapter full of algorithms for converting between color models, in C.

- Allan Hanbury; Jean Serra (December 2002). *A 3D-polar Coordinate Colour Representation Suitable for Image Analysis*. *Pattern Recognition and Image Processing Group Technical Report 77*. Vienna, Austria: Vienna University of Technology.

- Joblove, George H.; Greenberg, Donald (August 1978). "Color spaces for computer graphics". *Computer Graphics*. **12** (3): 20–25. doi:10.1145/965139.807362. Joblove and Greenberg's paper was the first describing the HSL model, which it compares to HSV.

- Kuehni, Rolf G. (2003). *Color Space and Its Divisions: Color Order from Antiquity to the present*. New York: Wiley. ISBN 978-0-471-32670-0. This book only briefly mentions HSL and HSV, but is a comprehensive description of color order systems through history.

- Levkowitz, Haim; Herman, Gabor T. (1993). "GLHS: A generalised lightness, hue and saturation color model". *CVGIP: Graphical Models and Image Processing*. **55** (4): 271–285. doi:10.1006/cgip.1993.1019.

This paper explains how both HSL and HSV, as well as other similar models, can be thought of as specific variants of a more general "GLHS" model. Levkowitz and Herman provide pseudocode for converting from RGB to GLHS and back.

- MacEvoy, Bruce (January 2010). "Color Vision". *handprint.com*.. Especially the sections about "Modern Color Models" (http://www.handprint.com/HP/WCL/color7.html) and "Modern Color Theory" (http://www.handprint.com/HP/WCL/color18a.html). MacEvoy's extensive site about color science and paint mixing is one of the best resources on the web. On this page, he explains the color-making attributes, and the general goals and history of color order systems—including HSL and HSV—and their practical relevance to painters.
- Poynton, Charles (1997). "Frequently Asked Questions About Color". This self-published frequently asked questions page, by digital video expert Charles Poynton, explains, among other things, why in his opinion these models "are useless for the specification of accurate color", and should be abandoned in favor of more psychometrically relevant models.
- Smith, Alvy Ray (August 1978). "Color gamut transform pairs". *Computer Graphics*. **12** (3): 12–19. doi:10.1145/965139.807361. This is the original paper describing the "hexcone" model, HSV. Smith was a researcher at NYIT's Computer Graphics Lab. He describes HSV's use in an early digital painting program.

External links

- Demonstrative color conversion applet (http://www.cs.rit.edu/~ncs/color/a_spaces.html)
- HSV Colors (http://demonstrations.wolfram.com/HSVColors/) by Hector Zenil, The Wolfram Demonstrations Project.

Retrieved from "https://en.wikipedia.org/w/index.php?title=HSL_and_HSV&oldid=778532498"

Categories: Color space