Color Spaces

Raul Rojas

Institut für Informatik
Freie Universität Berlin
Takustr. 9, 14195 Berlin, Germany

Abstract. This paper discusses the differences and advantages of some color spaces from the viewpoint of robotics applications. Starting from the canonical RGB space, we look at the various definitions of alternative spaces such as YUV, HSI, HSV, etc. We provide graphical representations for the magnitudes used in color spaces, which should give an intuitive understanding of their usefulness and relationship.

1 Color spaces

Digital cameras usually detect color using filters in front of each pixel in the imaging chip. This arrangement mimics the way the human eye detects light of different wavelengths using cells shaped like cones, and which are preferentially sensitive to light around red, green, or blue monochromatic light. Whatever the method used to detect light in the color camera, the most common encoding of color information in the computer is the RGB color space, in which a color is represented by the amount of primary red (R), green (G), and blue (B) it "contains". In this section we review some alternative color spaces and their relationship to the canonical RGB space.

A color space is a way of encoding color using several color components. The purpose of color encoding is to allow the decoder (that is, a TV, computer monitor, or printer) to reproduce the appearance of the original color so as to meet the expectations of the human eye. That is, a printed photograph should look as real as the scene it captures. Originally, color spaces were defined for TV broadcasting since this decoder is much older than computers. This is a difficulty we will encounter with almost all color spaces: they were not originally defined for robots or pattern recognition, but for human entertainment.

1.1 RGB color space

Colors in RGB space have three coordinates. RGB space can be represented by a cube with an axis for each primary color, normalized between 0 and 1. A usual representation in computers is 24-bit RGB, in which each primary color is encoded using 8 bits (the integer coordinate runs between 0 and 255, which can be interpreted as a value between 0/255 and 255/255). Fig. 1 shows the RGB color space cube.

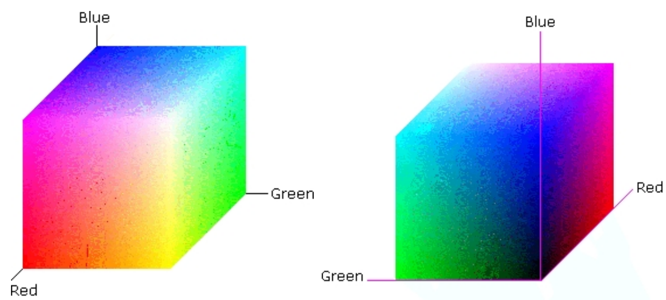


Fig. 1. The front and back of RGB color space

Pure red has the coordinates $(1, 0, 0)$ in the cube, pure green $(0, 1, 0)$, and pure blue $(0, 0, 1)$. Other corners of the RGB cube are magenta, cyan, and yellow, which correspond to the color mixtures (1,0,1), (0,1,1), and (1,1,0). These colors can be also used as the basic colors and their mixture can produce any of the other colors. However, there is an important difference. A given color (x, y, z) in the RGB color cube can be reproduced as

$$(x, y, z) = x(1, 0, 0) + y(0, 1, 0) + z(0, 0, 1)$$

which represents an additive mixture of red, green, and blue (since x, y, and z are positive and less than 1). This is the kind of color mixing we obtain from LCD projectors. In color TVs, additive color mixes are used, since light of one color emitted by the screen does not cancel light of another color emitted also by the screen.

In subtractive color mixing, such as the one used for printing, ink reflects a specific color mixture and the color reflected by two layers of ink is the componentwise product of their reflectivities (in a very abstract model). Therefore, red can be represented as

$$(1, 0, 0) = (1, 0, 1) \ominus (1, 1, 0)$$

that is, as the subtractive combination of magenta and yellow. Magenta $(1, 0, 1)$ absorbs the green component, while yellow $(1, 1, 0)$ absorbs the blue component, so that only the red component remains when we mix both colors.

In additive color mixtures red, green, and blue together produce white, since $(1,1,1) = (1,0,0) + (0,1,0) + (0,0,1)$. In subtractive color mixtures magenta, cyan, and yellow produce black. The figure below shows the difference between additive and subtractive color mixtures.

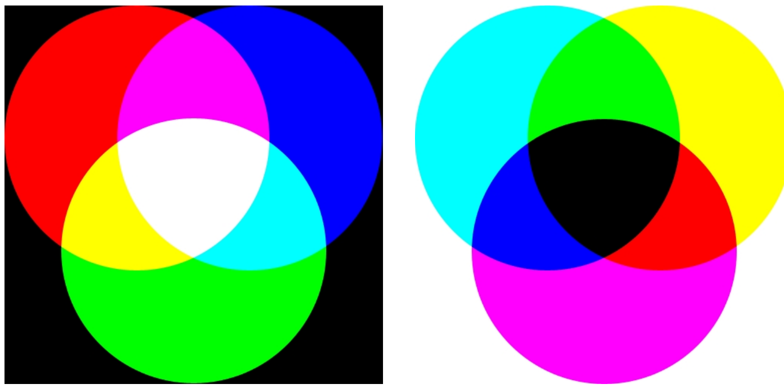


Fig. 2.

2 Linear transformations of the RGB color space

Color processing in the brain actually combines the individual measurements at each rod and cone in the retina in a complex manner. Color spaces are transformations of the RGB cube designed with the purpose of best capturing the kind of color classification done by humans. There are many different color spaces, each with its own rationale.

The first kind of alternative color spaces are linear transformations of the RGB cube. One important example is the YUV color space. In this space Y refers to the perceived intensity of a color. Separating the intensity component from the chromatic components allows us to broadcast video images with each pixel encoded as three numbers, the YUV coordinates. A black and white TV, for example, just decodes the Y component and produces a picture. Psychophysical measurements of the perceived intensity of colors has led to many proposals for the perceived intensity of colors (which is also very variable between humans). The formula used is

$$Y = 0.299R + 0.587G + 0.114B.$$

Green has a higher weight than red, and almost five time higher than blue. The human eye contains just 2% of blue cones, but they are more sensitive than the green cones, so that final weighting must be obtained from actual experiments. The U and V components are obtained from the following expressions

$$U = c_1(B - Y) = -0.169R - 0.332G + 0.500B + 128$$

and

$$V = c_2(R - Y) = 0.500R - 0.419G - 0.0813B + 128$$

whwre c_1 and c_2 are normalizations constants adjusted to obtain the coefficient 0.5 in front of the blue component (in the first equation), and of the red component (in the second ecuation), and where the variables R, G, and B run from 0 to 255. The constant 128 is added to avoid negative U and V values.

Since the human eye is more sensitive to changes in illumination than color, many video cameras provide a Y value for every pixel, and U and V values for every two or every four pixels. For example, YUV 4:2:2 refers to a video signal with factor two undersampling in each row of the video image. This saves bandwidth in the transmission of the image from the video camera to the computer.

Fig. **??** shows the transformed RGB cube of colors. The Y axis is the new direction of the black to white colors.

3 Nonlinear transformations

Nonlinear transformations of the RGB cube are based on the same idea: to separate the measurement of the light intensity from the chromatic information. The main difference is the definition of the intensity and the distance from the intensity axis.

3.1 Intensity – Projection on a sphere, a plane, a cube, other surfaces

The color receptors in the human eye record the relative mixtures of wavelengths present in light meeting the human eye. The mixture is up to certain point independent of the intensity of the light. That is, doubling the light registered by the red, green, and blue receptors modifies our perception of the luminosity of an object, but not of its hue. We do not expect objects to change color just because we dim the lights.

If we assume that a color (r, g, b) is perceptually equivalent to a color $(\alpha r, \alpha g, \alpha b)$, where α is a positive constant, then any ray $(\alpha r, \alpha g, \alpha b)$ represents the same color. If for any such ray we represent a color by the intersection of this ray with a sphere of radius 1, then all colors can be mapped to the surface of this sphere in the octant of positive values for r,g, and b, as shown in Fig. 3. There is a one-to-one correspondence between this 2D representation and each color. In this case intensity is defined as

$$I_1 = \sqrt{R^2 + G^2 + B^2)}$$

I_1 can be also divided by $sqrt3$ if white is normalized to intensity 1. If we use the Manhattan metric for measuring the length of the RGB vector, we can map each ray to the plane with coordinates $R + G + B = 1$ (Fig. 3, middle). The definition of intensity, in this case, is

$$I_2 = (R + G + B)$$

I_2 can be divided by 3 if white is normalized to intensity 1. Another possibility is to map each ray (each color) to the intersection of the ray with the sides of the RGB cube. In this case the definition of intensity changes to

$$I_3 = \max(R, G, B)$$

Fig. 3 (right) shows this alternative projection. Any of these definitions of intensity is possible and mathematically equivalent from the point of view of obtaining a reversible transformation (as we see below).

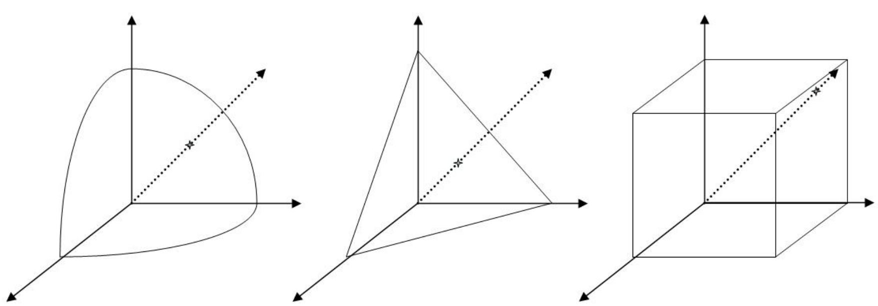


Fig. 3. Projection of a color ray to the surface of a sphere (radius 1), the plane $R + G + B = 1$, and the surface of the RGB cube.

An interesting alternative for defining the intensity is the formula:

$$I_4 = \frac{\max(R, G, B) + \min(R, G, B)}{2}$$

We can graph the surfaces of iso-luminosity when this formula is used. Fig. 4 shows the result. The surfaces for luminosity 0.8, 0.4, and 0.2 are represented. Each isoluminosity surface is parallel to each other. The surface is a patch of six planes which meet at the black-white diagonal. The result is similar to that obtained with the Manhattan metric but the spacing of the surfaces of isointensity is more regular.

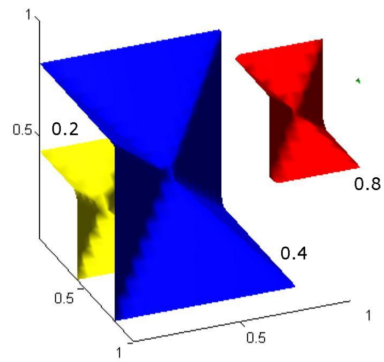


Fig. 4. Surfaces of equal luminosity (0.8, 0.4, and 0.2) for $I = (\max(R, G, B) + \min(R, G, B))/2$

Given the surface used for the projection of a color ray, we can now define a 2D system of coordinates on this surface. One of the coordinates is very simple to define: the angle of the projection around the diagonal joining black with white (the line from $(0, 0, 0)$ to $(1, 1, 1)$). Many color spaces use this definition of "hue. If we look at any of the projections explained above, aligning the center of our eye or camera with the vector $(1, 1, 1)$, the different colors change when we turn around the center of the image. From the center of the image to the sides, the colors become more vivid. The diagonal corresponds to colors of the type (α, α, α), that is colors with equal contributions of red, green, and blue, that is shades of white (gray values). The farther away a color is from the diagonal, the more vivid it looks. This is the 'saturation of the color, which can be measured in different but related ways, as we see below.

3.2 HSI and HSV color space

In HSI color space, H stands for *hue*, S for *saturation*, and I for *intensity*. V stands for *value* in HSV color space. HSI and HSV are "families" of color spaces rather than well-defined monolithic spaces. A review of the literature immediately shows that under HSI many alternative definitions of hue, saturation, and intensity are used. The same goes for HSV, so that we should really understand the alternatives and think of HSI and HSV as clusters of related color spaces.

In the HSI and HSV families several definitions of saturation are used. The general idea is to assign saturation 1 to the primaries red, green, and blue, and saturation zero to gray levels, interpolating for intermediate colors. There are at least the following alternative definitions of saturation:

$$S_1 = \frac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B)}$$

or

$$S_2 = \frac{\max(R,G,B) - \min(R,G,B)}{\max(R,G,B) + \min(R,G,B)}$$

or

$$S_3 = 1 - 3\frac{\min(R,G,B)}{R+G+B}$$

or

$$S_4 = d((R,G,B),(1,1,1))$$

where the function d computes the normalized Euclidean distance between the point (R,G,B) and the line from the origin to (1,1,1) (the distances are normalized so that pure red, green, and blue have saturation 1).

In order to implement this kind of computation we need to rotate first the system of coordinates using the transformation

$$\begin{pmatrix} m_1 \\ m_2 \\ i_1 \end{pmatrix} = \begin{pmatrix} \frac{2}{\sqrt{6}} & -\frac{1}{\sqrt{6}} & -\frac{1}{\sqrt{6}} \\ 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1}{\sqrt{3}} \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} \tag{1}$$

We then compute

$$S_4 = \sqrt{m_1^2 + m_2^2}$$

The intensity can be computed also directly from the above transformation as $I_2 = \sqrt{3} \cdot i_1$

In the first three cases S_1, S_2, and S_3 the saturation for pure red, green, and blue is 1. With the definition of S_4, however, it is $sqrt2/sqrt3$, a little less than 1. For any shade of gray $R = G = B$, and therefore $S_1 = S_2 = S_3 = S_4 = 0$. The different definitions of saturation yield sometimes different values for cyan, magenta, and yellow. For cyan, which corresponds to $(0,1,1)$, for example, $S_1 = S_2 = S_3 = 1$, but $S_4 = sqrt2/sqrt3$

Fig. 5 show the surfaces of isosaturation for S_1 and S_2. The surfaces of isosaturation for S_4 are cylinders with axis parallel to the black-white diagonal. The isosaturation curves for S_1 and S_2 look like "cones" with planar boundaries (six planes). The isosaturation surfaces for S_2 are not as symmetrical as for S_1. They assign, in general, less saturation to colors along the diagonals from the black-white line to cyan, magenta, and yellow.

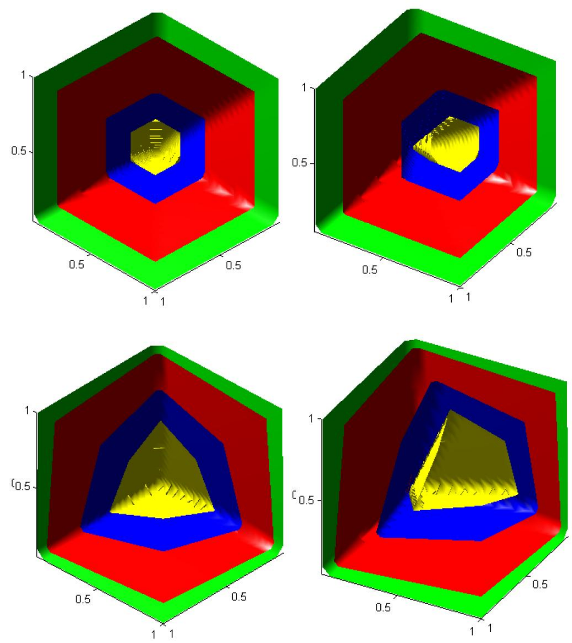
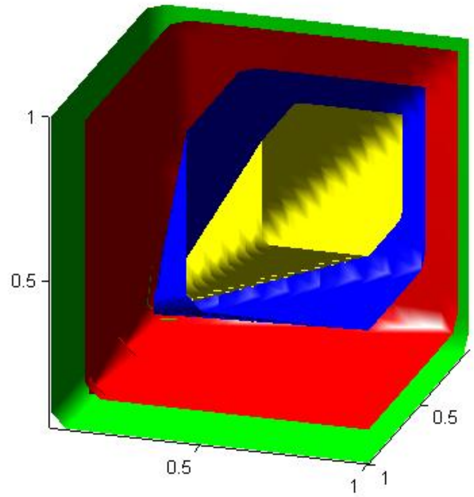


Fig. 5. Regions of isosaturation in the RGB cube, according to two different definitions of the saturation. The upper diagrams show the isosaturation surfaces for S_1, the two lower diagrams correspond to S_2.

Fig. 7 shows the isosaturation surfaces for $S-2$. They are very similar to the surfaces for S_1 but are more unformly spaced.

3.3 The hue

Whatever the definition of the intensity I and the saturation S, the hue is measured using the angle around the black-white diagonal. One of the axis has



to be taken as the reference for zero degrees, for example, the red axis.

$$H_1 = \begin{cases} (0 + \frac{(G-B)}{max-min}) \cdot 60 & \text{if } R = max \\ (2 + \frac{(B-R)}{max-min}) \cdot 60 & \text{if } G = max \\ (4 + \frac{(R-G)}{max-min}) \cdot 60 & \text{if } B = max \end{cases}$$

Fig. 7 shows the kind of computation which results from the formula above. In the figure we are looking to the RGB cube along the black-white diagonal. There are six regions, for each of them the maximum and minimum coordinate has been written. Outside the hexagon (projection of the RGB cube on the plane) we see the computation performed for finding the hue. The computation, when R is the maximum coordinate, yields values from -1 to 1. The same occurs when B or G are the maximum. But when G is the maximum we add 2, and when B is the maximum we add 4. So the computations yield values which run from -1 at angle -60 degrees up to 1 at 60 degrees, 2 at 120 degrees, 3 at 180 degrees, 4 at 240 degrees, and 5 at 300 degrees. Multiplying by 60 we obtain angles running from -60 to 300 degrees, that is, the whole circumference.

The computation of the angle between the axis for pure red and the vector (r, g, b) for a color is the same for all rays $(r, g, b)+\alpha(1, 1, 1)$, since maximum and minimum do not change, and the subtractions in the numerator and denominator of the expression for H_1 eliminate all α. This shows that the computation is consistent when we look at the RGB cube along the black-white diagonal.

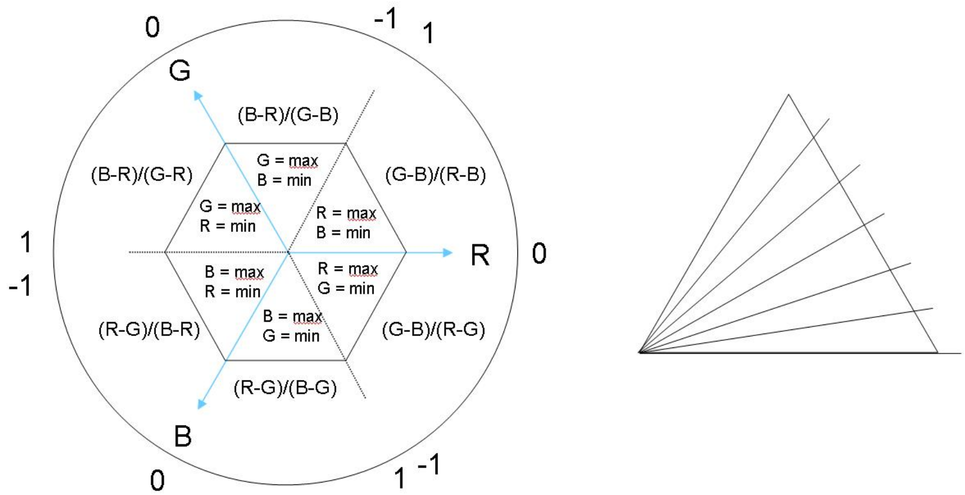


Fig. 7.

An exact computation of the angle would require the use of the arccos, or another trigonometric function. Instead, in the computation for H_1, the angle is interpolated using the six axes visible in Fig. 7. The computation is very simple for the periphery of the cube (max= 1, min= 0). The triangle to the right of Fig. 7 shows how the hues corresponding to $0, 10, 20, \ldots, 60$ degrees are found, just dividing the side of the equilateral triangle in six equal segments. Of course, the lines do not correspond to the exact position of $0, 10, 20, \ldots, 60$ degrees, but are very near. This kind of computation avoids using an expensive trigonometric function.

Another option is to measure exactly the angle around the black-white line. In this case we use the transformation given by Eq. **??**, and compute the hue as:

$$H_2 = \arctan\left(\frac{m_1}{m_2}\right)$$

where the arctan function returns the appropriate value between 0 and 360 degrees, according to the signs of m_1 and m_2.

In this definition of hue (based on the cylindrical cooordinates of the transformed points after applying the transformation **??**, a point (r, g, b) has the same hue, and the same saturation S_4 as any other point $(r, g, b) + \alpha(1, 1, 1)$ (since the transformation of $\alpha(1, 1, 1)$ yields $m_1 = m_2 = 0$, and we have a linear transformation). This definition of hue combined with S_4 and $I_2 = (R + G + B)/3$ corresponds to just a transformation from Cartesian to cylindrical coordinates.

Still a third alternative for the definition of the hue is the formula

$$H_3 = cos^{-1}\left(\frac{(R-G)+(R-B)}{2\sqrt{(R-G)^2+(R-B)(G-B))}}\right)$$

For pure red we obtain $H_3 = 0$ degrees, for pure green $H_3 = 120$ degrees, and for pure blue $H_3 = 240$ degrees. This is, in principle, the same computation as H_2 but for a rotated system of coordinates.

3.4 Combinations of I, S, and H

As we saw in the previous sections, there are many alternative definitions for intensity and saturation. Even the hue can be measured differently if we use the arccos function to compute the polar angle for a system of coordinates, or if we interpolate linearly using predefined angles for the red, green, and blue axis.

Any combination of I, S, and H qualifies as a color space. However, why should we prefer one color space over another? If the color mapping from RGB to a new color space is reversible, all such color spaces are equivalent, if our only goal is to encode colors.

Frequent combinations found in the literature are:

- HSI, as I_2, S_3, and H_2.
- HSI, as I_2, S_4, and H_2.
- HSV as I_3, S_1, and H_1.

There is an important aspect which must be considered when comparing color spaces, which is the metric we can use to compare two colors. Given the coordinates of two colors in a three-dimensional color space, such as RGB or HSI, we would like to be able to compare them using their Euclidean distance in the given color space. The question is if "similar" colors in the selected color space also look similar for humans. Or if the computer finds that two colors are near, according to their coordinates, are they indistinguishable for a human? If so, they could be classified as the same color for the purpose of segmenting visual scenes.

All these questions boil down to the question of finding a color space in which color comparisons can be done with an Euclidean metric, which is convenient and easy to implement. However, it is known that the human eye has different color resolutions in different parts of the RGB cube. This kind of questions have been studied by the CIE (Commision Internationale de l'Eclaraige) who has proposed several color spaces over the years.

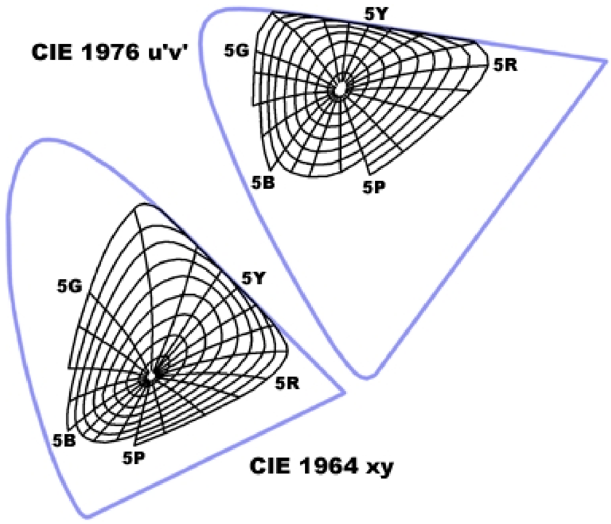


Fig. 8.

References

1. D. Forsythe, and J. Ponce, *Computer Vision: A Modern Approach*, Prentice-Hall, 2003.