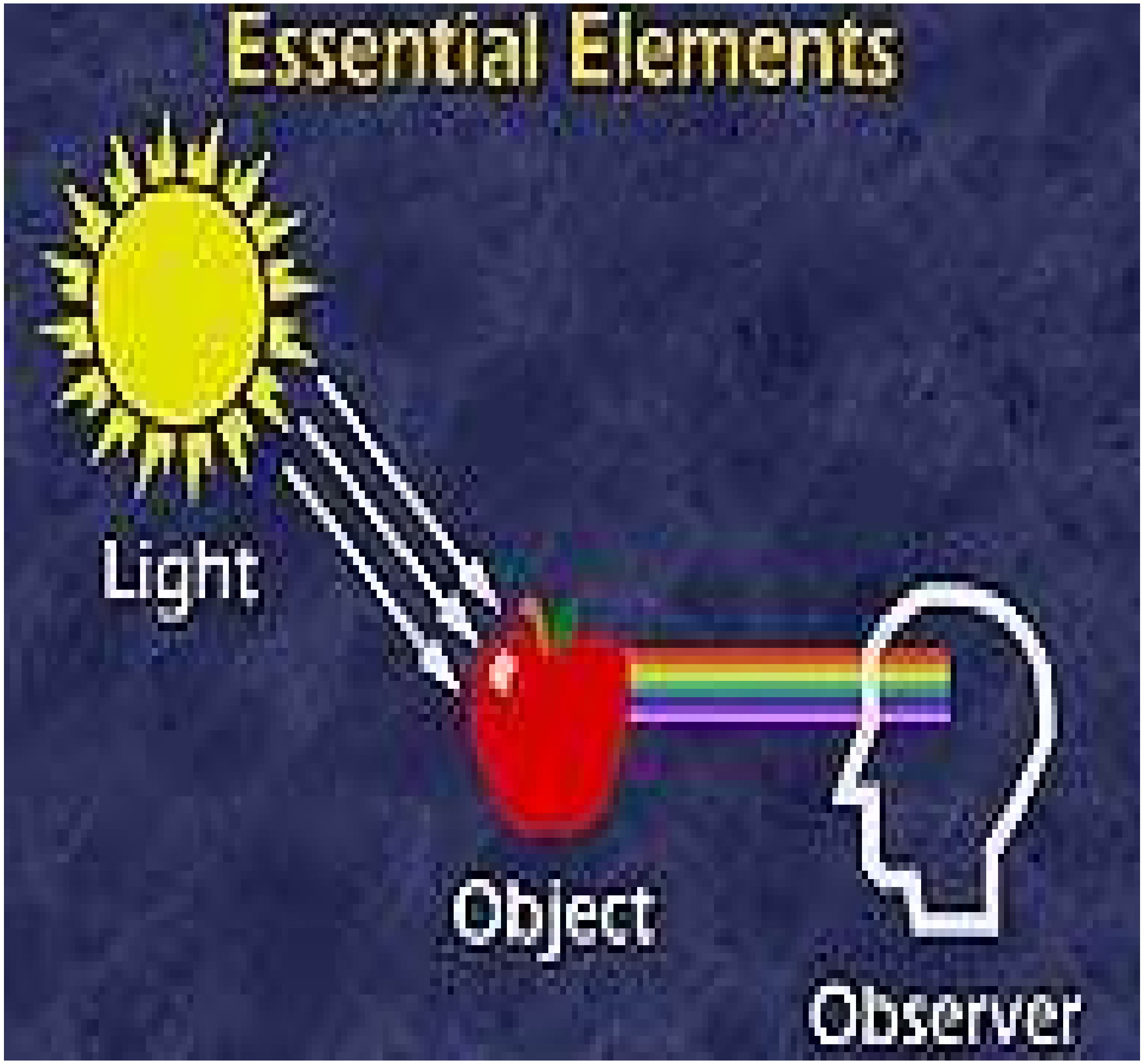
Color Models and Color Applications

Color Fundamentals

Color Models

What is Color ?



To see color, three essential elements must be present:

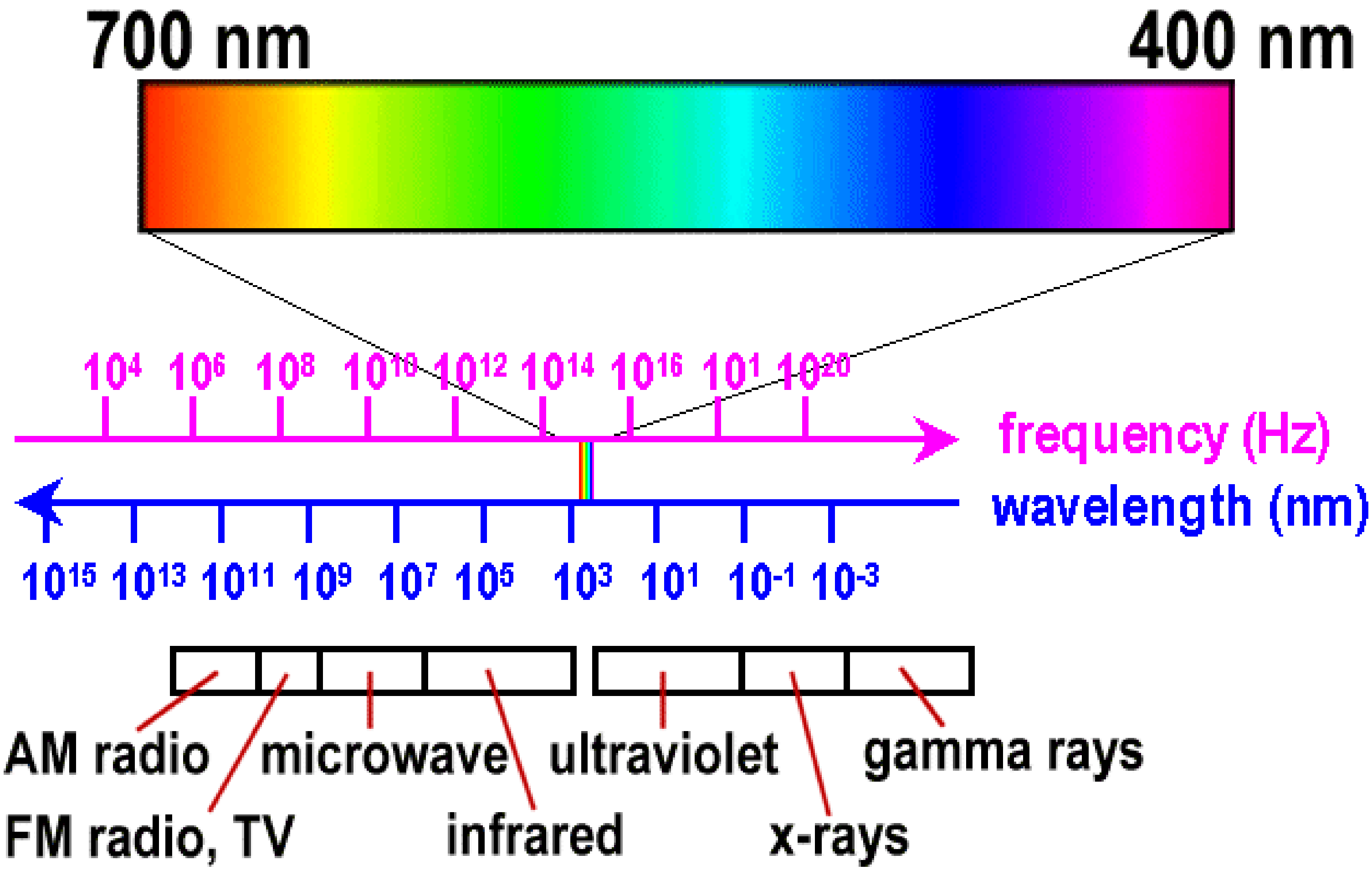
light,

an illuminated object,

and an observer.

Visible Spectrum

We perceive electromagnetic energy having wavelengths in the range 400-700 nm as *visible light*.



Light and Color

The frequency (or mix of frequencies) of the light determines the color.

The amount of light(sheer quantity of photons) is the intensity.

Three independent quantities are used to describe any particular color. : hue, saturation, and lightness or brightness or intensity.

The ***hue*** is determined by the dominant wavelength.(the apparent color of the light)

When we call an object "red," we are referring to its hue. Hue is determined by the dominant wavelength.

Light and Color

The saturation of a color ranges from neutral to brilliant. The circle on the right is a more vivid red than the circle on the left although both have the same hue.

The ***saturation*** is determined by the excitation purity , and depends on the amount of white light mixed with the hue. A pure hue is fully saturated, i.e. no white light mixed in. Hue and saturation together determine the *chromaticity* for a given color.

Light and Color

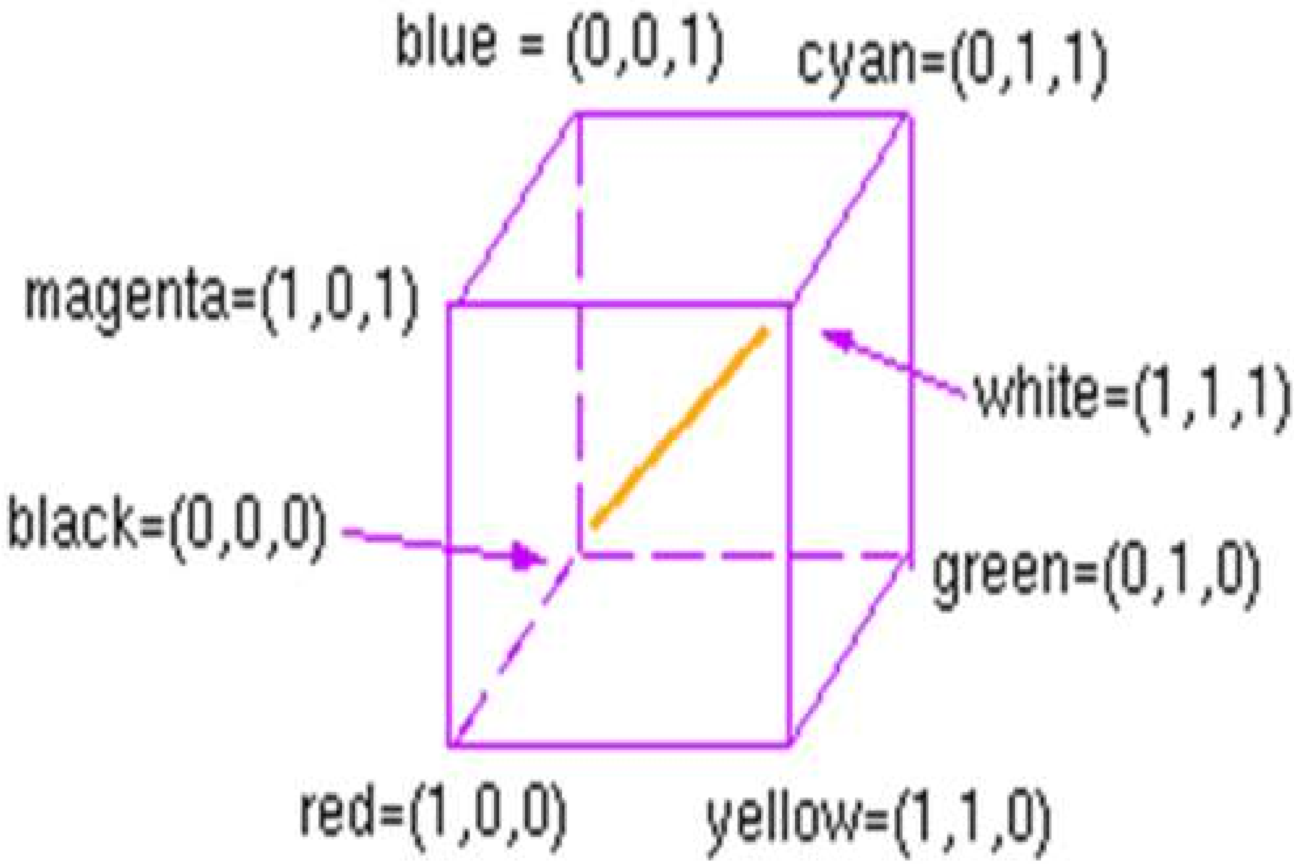
Lightness or brightness refers to the amount of light the color reflects or transmits.

Color Models

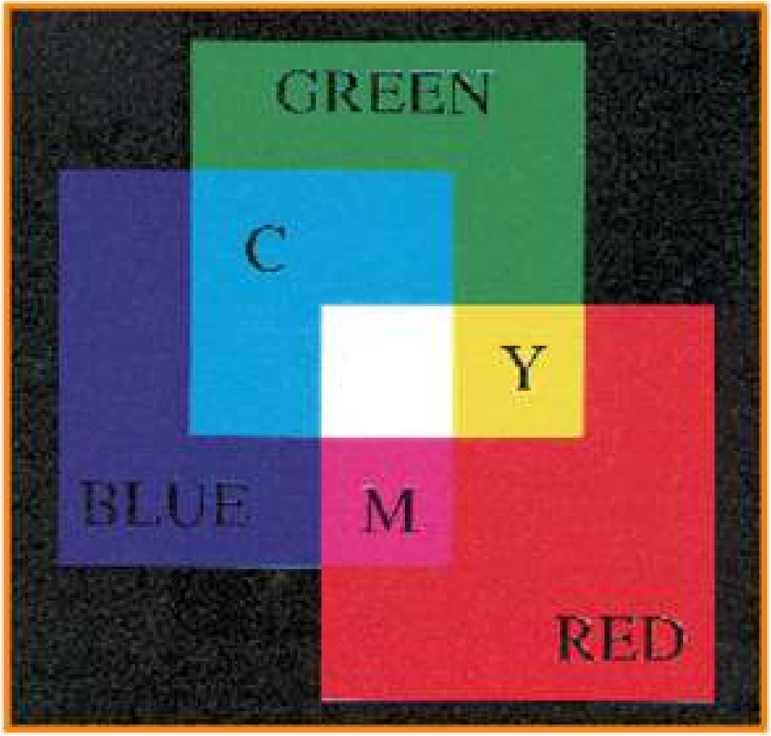
Color models provide a standard way to specify a particular color, by defining a 3D coordinate system, and a subspace that contains all constructible colors within a particular model. Any color that can be specified using a model will correspond to a single point within the subspace it defines. Each color model is oriented towards either specific hardware (RGB,CMY,YIQ), or image processing applications (HSI).

The RGB Color Cube

The additive color model used for computer graphics is represented by the RGB color cube, where R, G, and B represent the colors produced by red, green and blue phosphorus, respectively.



RGB Color Model

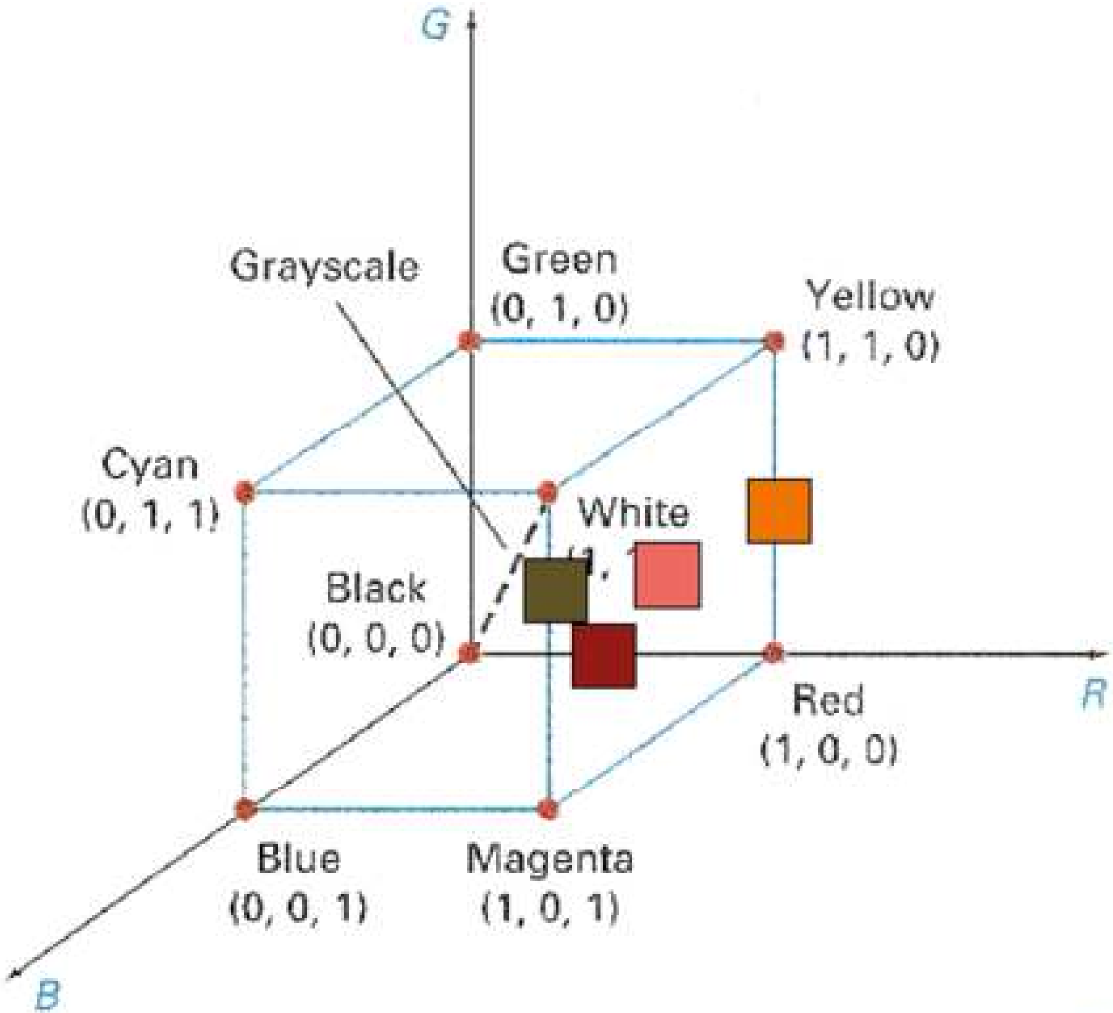


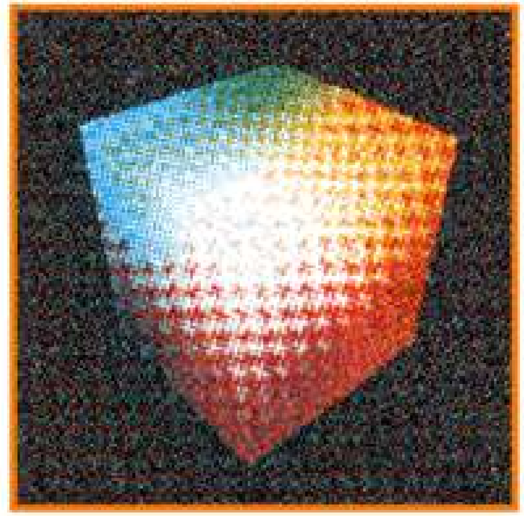
Colors are additive

R	G	B	Color
0.0	0.0	0.0	Black
1.0	0.0	0.0	Red
0.0	1.0	0.0	Green
0.0	0.0	1.0	Blue
1.0	1.0	0.0	Yellow
1.0	0.0	1.0	Magenta
0.0	1.0	1.0	Cyan
1.0	1.0	1.0	White
0.5	0.0	0.0	?
1.0	0.5	0.5	?
1.0	0.5	0.0	?
0.5	0.3	0.1	?

Plate II.3 from FvDFH

RGB Color Cube



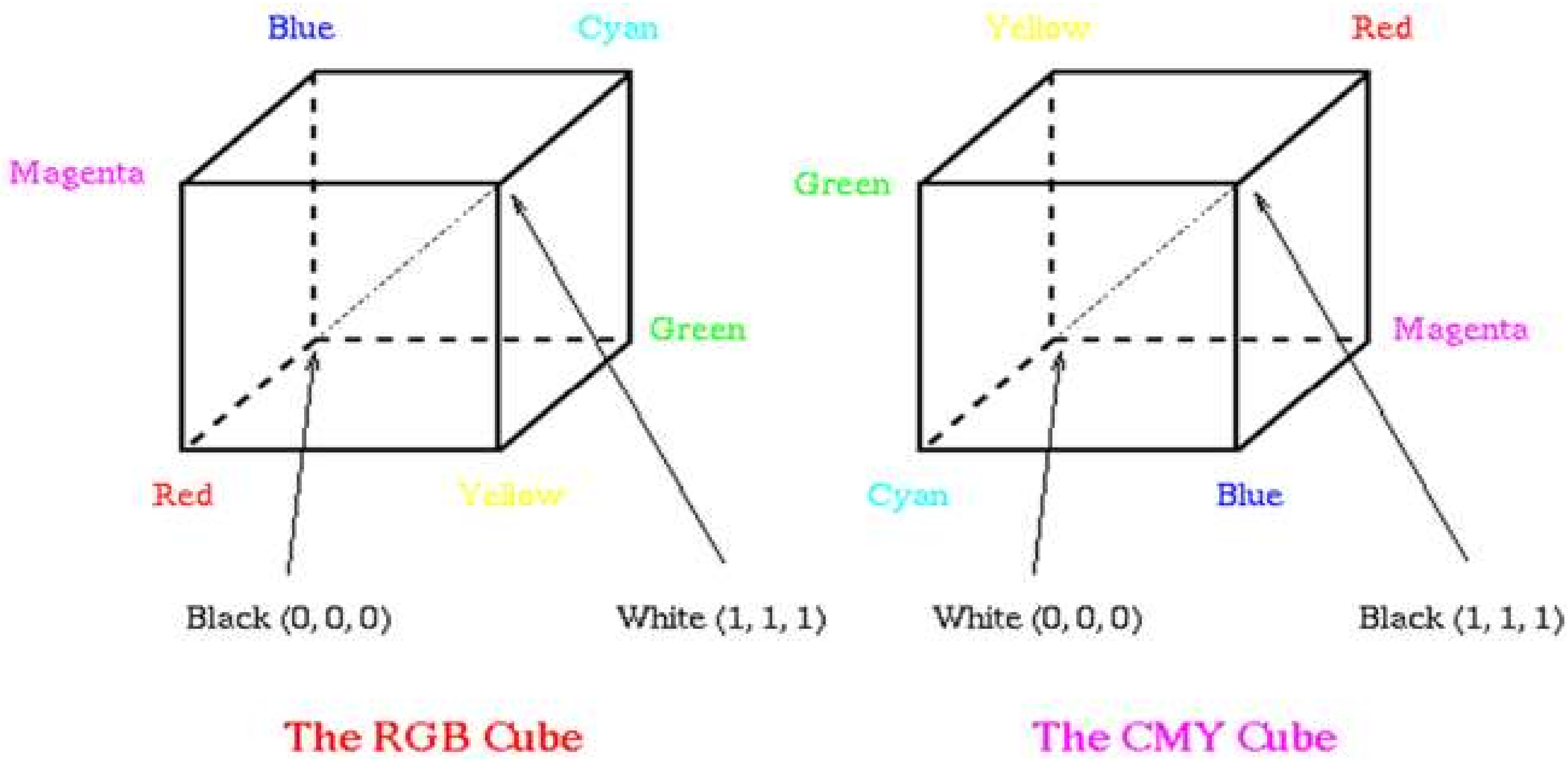


Figures 15.11&15.12 from H&B

CMY Color Model

•Cyan, Magenta, and Yellow (CMY) are complementary colors of RGB. They can be used as *Subtractive Primaries*.

•CMY model is mostly used in printing devices where the color pigments on the paper absorb certain colors (e.g., no red light reflected from cyan ink).



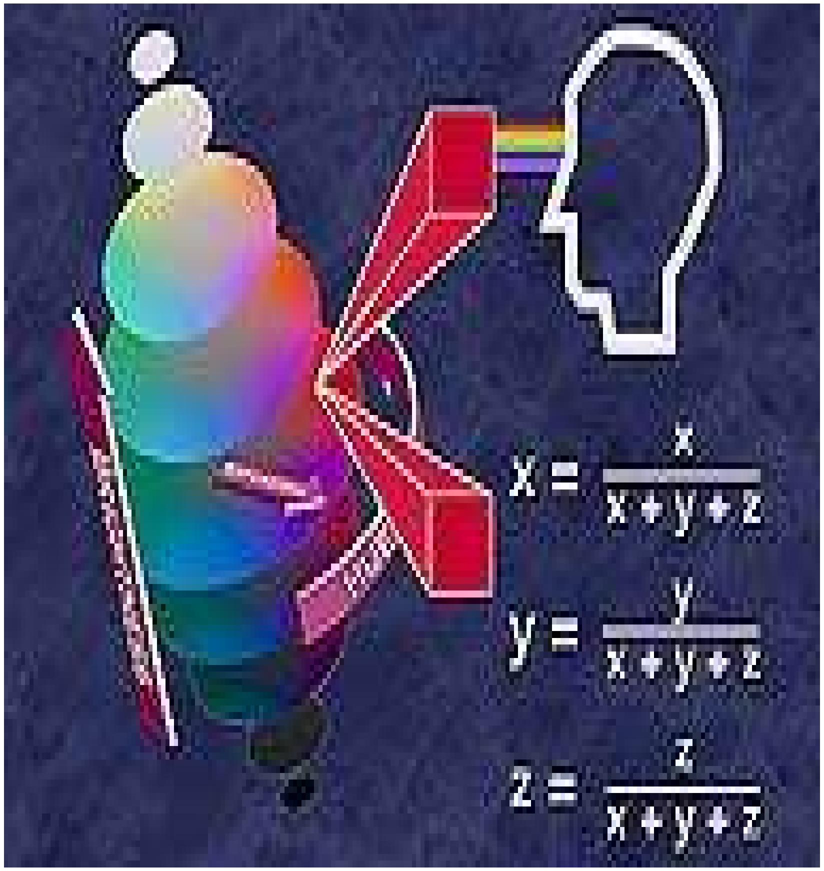
Conversion between RGB and CMY

- Convert White from (1, 1, 1) in RGB to (0, 0, 0) in CMY:

- Sometimes, an alternative CMYK model (K stands for *Black*) is used in color printing (e.g., to produce darker black than simply mixing CMY).

K := min (C, M, Y), C := C - K, M := M - K, Y := Y - K.

CIE Chromaticity Diagram

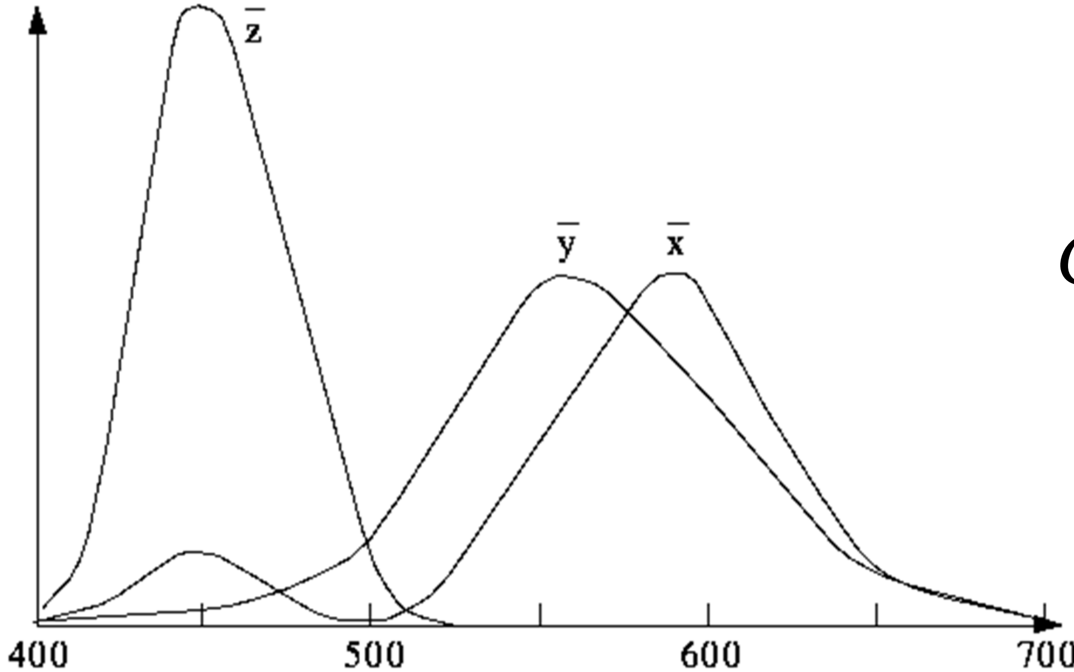


To measure and predict the appearance of a particular color, we need a way to link perception to numbers and formulas.

Scientific color values were established earlier this century by the CIE group. CIE models for defining color space all rely on the same basic numbers.

CIE Chromaticity Diagram

In 1931, the CIE defined three standard primaries **(X, Y, Z)**. The **Y** primary was intentionally chosen to be identical to the luminous-efficiency function of human eyes.

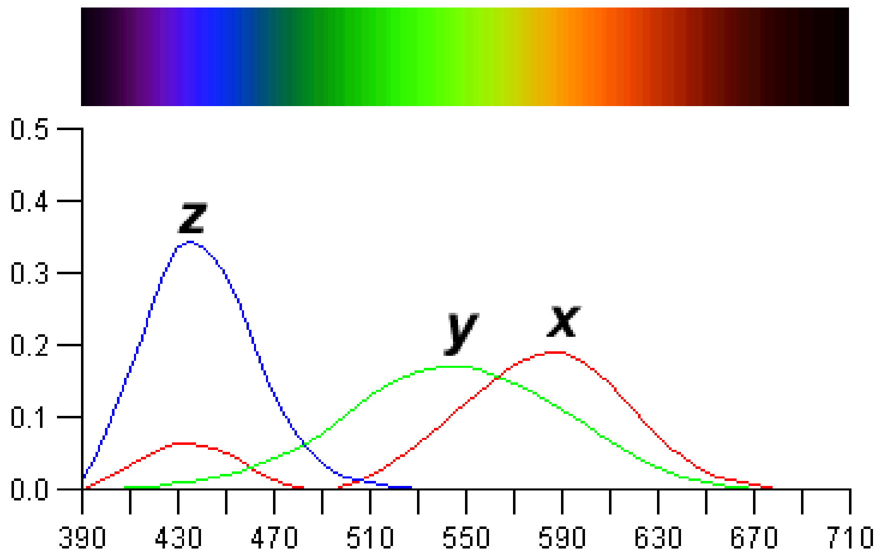


$$C_\lambda = X\mathbf{X} + Y\mathbf{Y} + Z\mathbf{Z}$$

•The above figure shows the amounts of X, Y, Z needed to exactly reproduce any visible color.

CIE Color Space

In order to achieve a representation which uses only positive mixing coefficients, the CIE ("Commission Internationale d'Eclairage") defined three new hypothetical light sources, x, y, and z, which yield positive matching curves:



If we are given a spectrum and wish to find the corresponding X, Y, and Z quantities, we can do so by integrating the product of the spectral power and each of the three matching curves over all wavelengths. The weights X,Y,Z form the three-dimensional CIE XYZ space, as shown below.

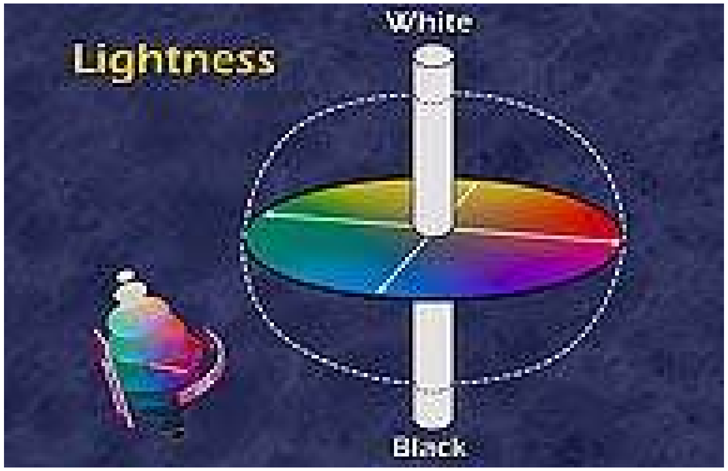
The formulas for converting from the tristimulus values (*X*,*Y*,*Z*) to the well-known CRT colors (*R*,*G*,*B*) and back are given by:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1.9107 & -0.5326 & -0.2883 \\ -0.9843 & 1.9984 & -0.0283 \\ 0.0583 & -0.1185 & 0.8986 \end{bmatrix} \bullet \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

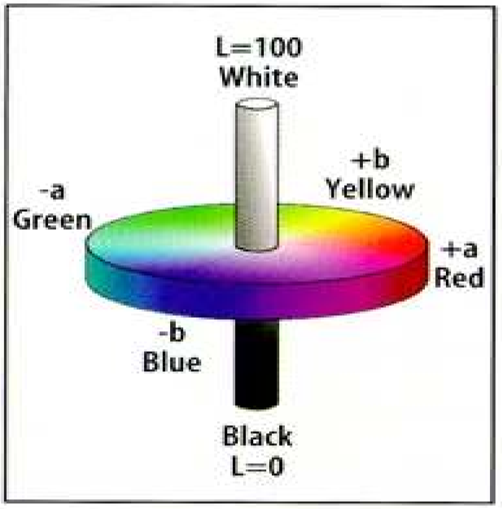
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.6067 & 0.1736 & 0.2001 \\ 0.2988 & 0.5868 & 0.1143 \\ 0.0000 & 0.0661 & 1.1149 \end{bmatrix} \bullet \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

As long as the position of a desired color (*X*,*Y*,*Z*) is inside the phosphor triangle in Figure , the values of *R*, *G*, and *B* as computed by eq. will be positive and can therefore be used to drive a CRT monitor.

L*a*b Color Model



Lightness is the third dimension that is not shown in color wheels often used in image editing software



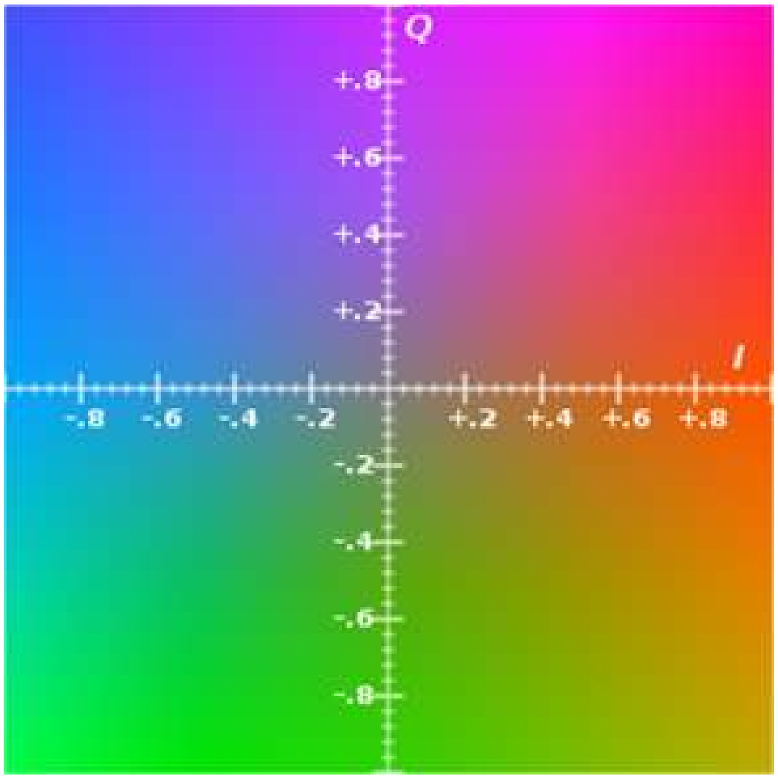
Lab model

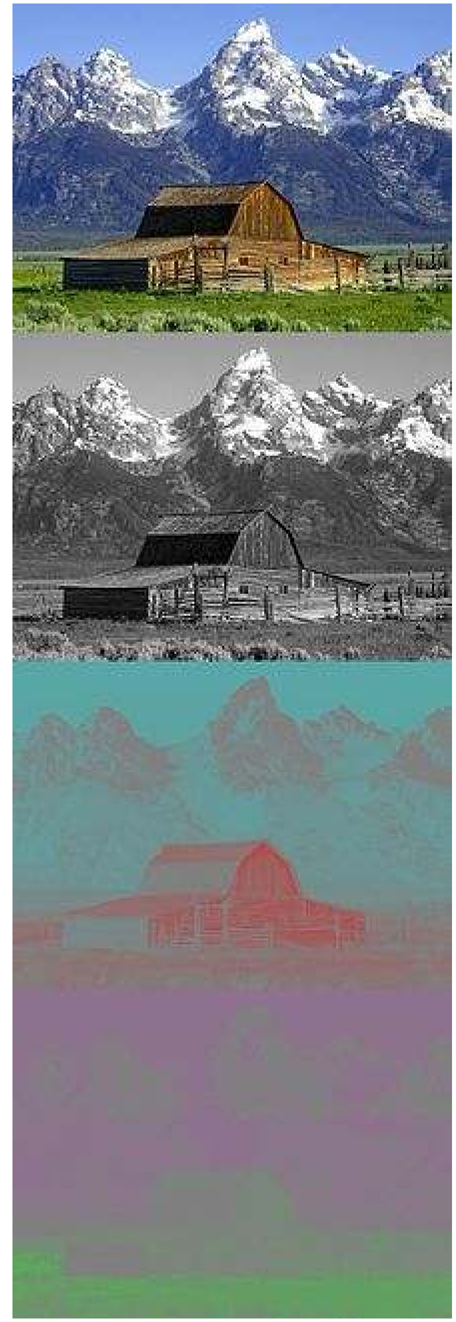
•A refined CIE model, named CIE L*a*b in 1976

•Luminance: L
Chrominance: a -- ranges from green to red, b -- ranges from blue to yellow

The **YIQ (luminance-inphase-quadrature)** model is a recoding of RGB for color television, and is a very important model for color image processing.

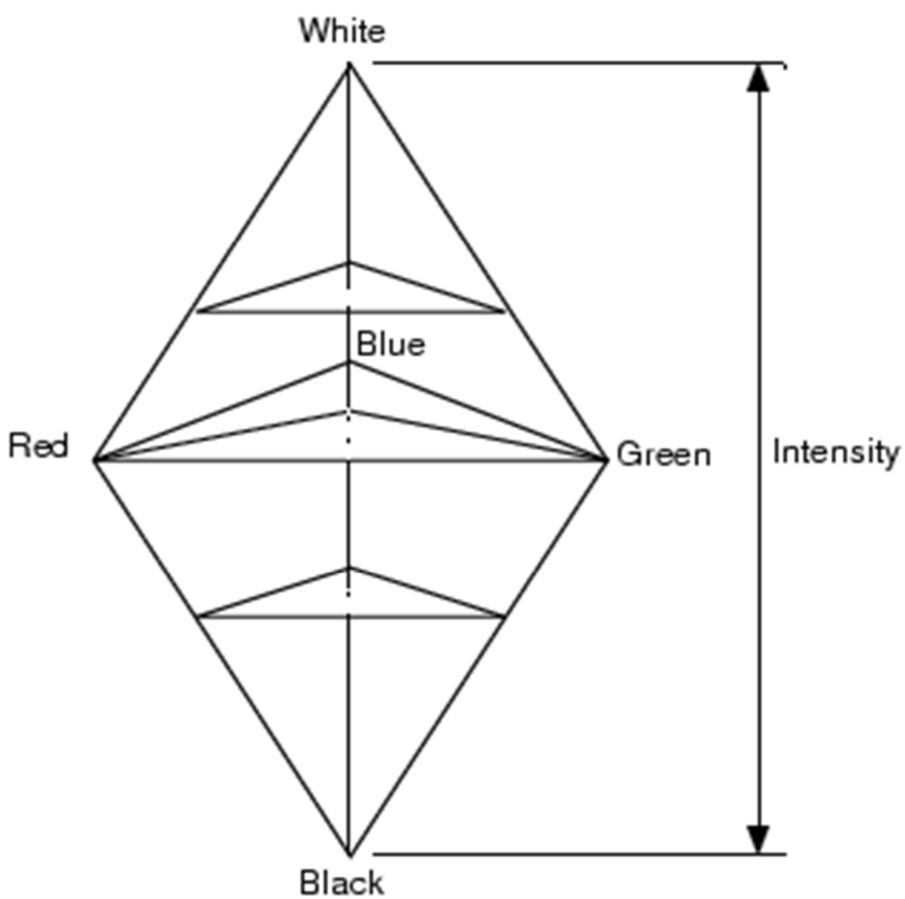
YIQ

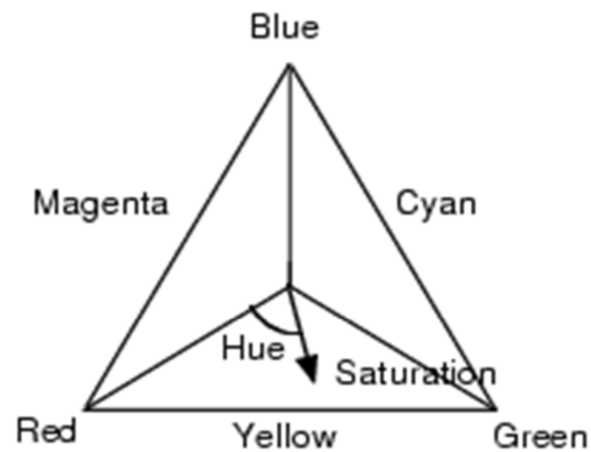




The HSI Model

As mentioned above, color may be specified by the three quantities ***hue, saturation and intensity.*** This is the HSI model, and the entire space of colors that may be specified in this way is shown in figure .





The HSI Model

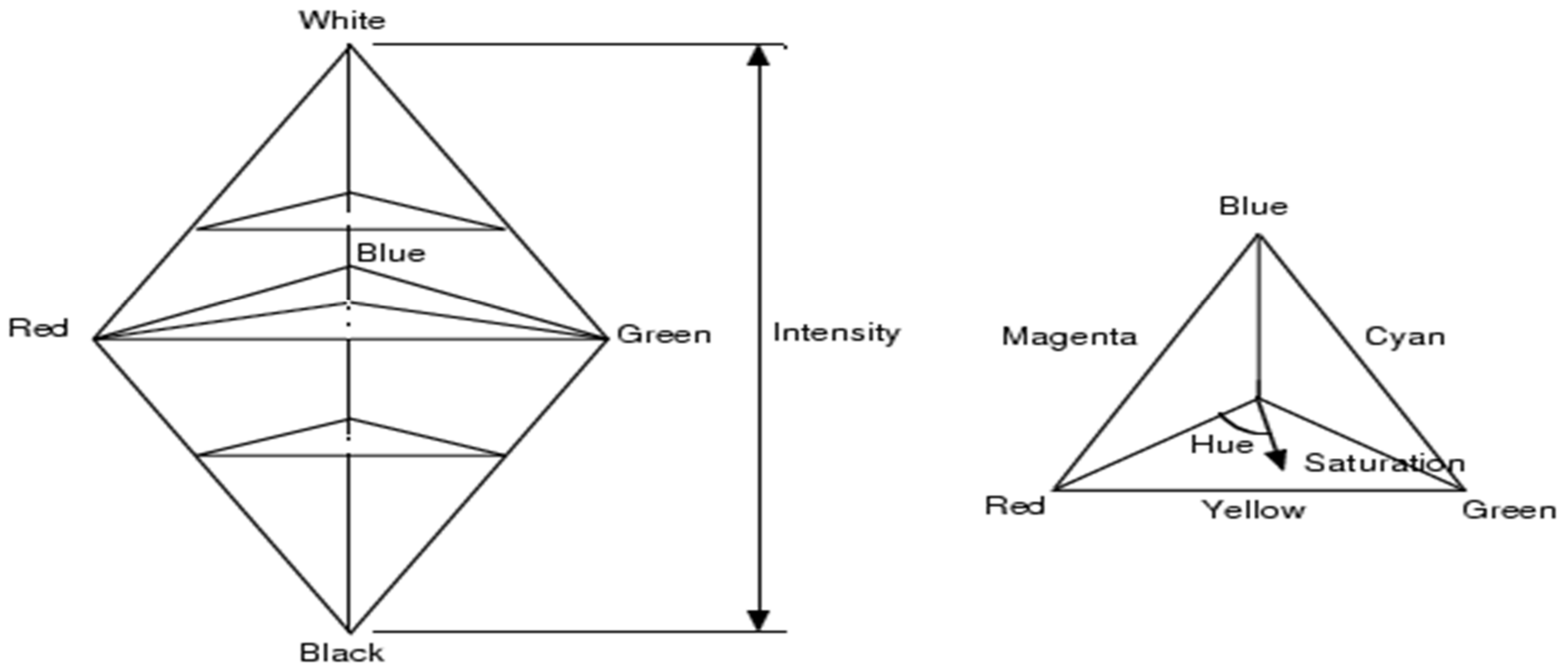
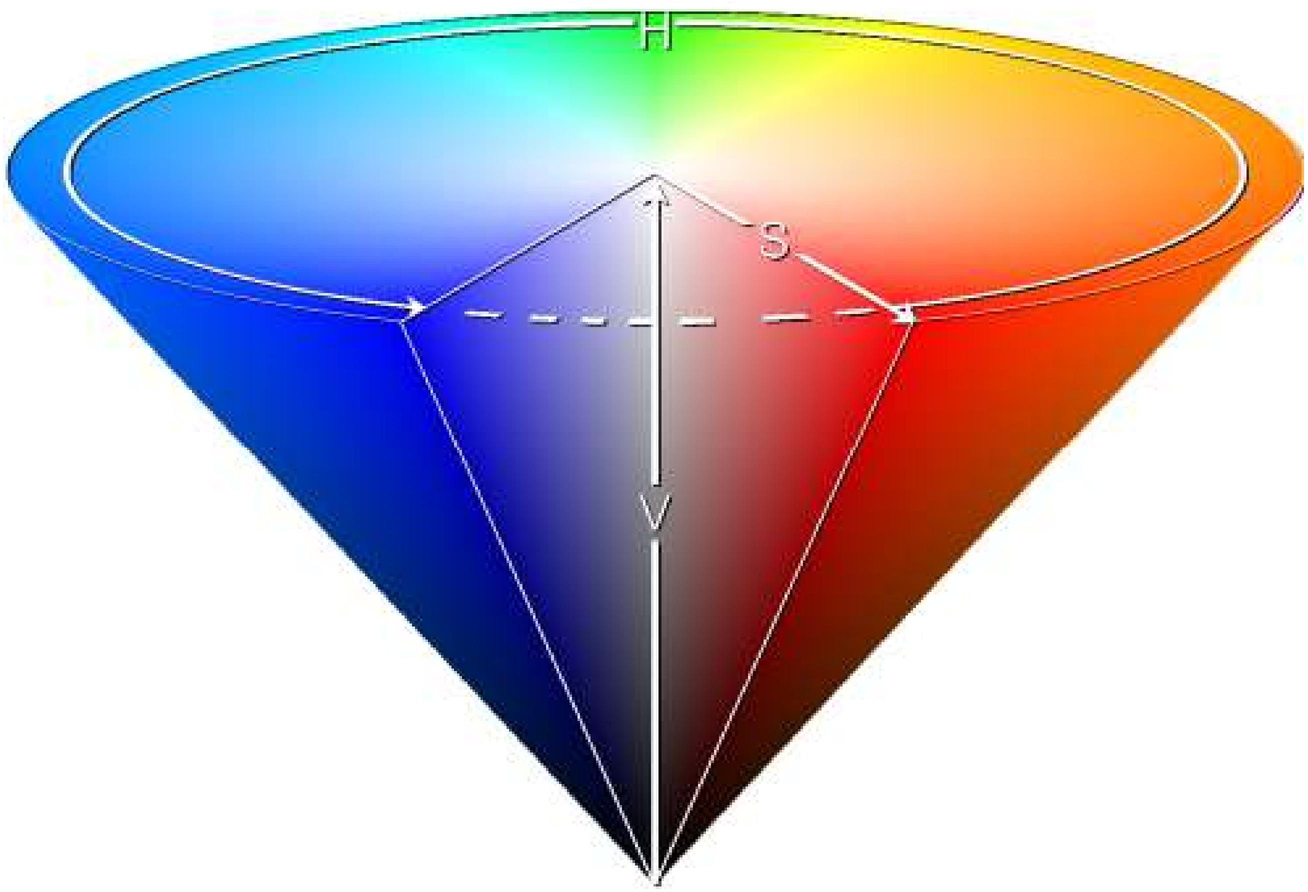


Figure : The HSI model, showing the HSI solid on the left, and the HSI triangle on the right, formed by taking a horizontal slice through the HSI solid at a particular intensity. Hue is measured from red, and saturation is given by distance from the axis. Colors on the surface of the solid are fully saturated, i.e. pure colors, and the greyscale spectrum is on the axis of the solid. For these colors, hue is undefined.

HSV Color Model

- **H**ue: true color attribute
- **S**aturation: amount that the color is diluted by white
  - pure red → high saturation
  - light red → low saturation
- **V**alue: (another) degree of brightness

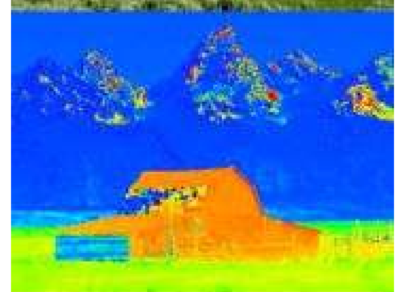
HSV Color Space



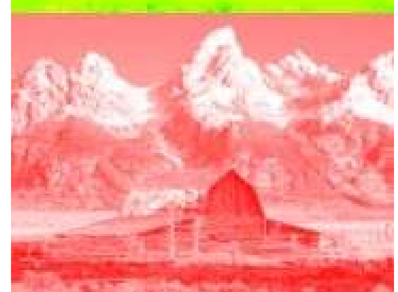
RGB Image VS HSV Image

RGB Image

Hue Image



Saturation Image
(white : low)



Value Image

Setting Colors in OpenGL

On a color computer screen, the hardware cause each pixel on the screen to emit different amounts of red, green, and blue light. OpenCL support two colors models , **RGB, or RGBA** mode and **color-index mode**. In RGB mode , each color is a triplet of red, green , and blue values. The eye blends these primary colors , forming the color that we see. In **RGBA** model we use fourth color component , **A, or alpha**, which is an **opacity**. An opacity of 1.0 means that the color is opaque and cannot be "seen through", whereas a value of 0.0 means that a color is transparent.

Specifying a Color in RGBA Mode

glColor3f(TYPE r, TYPE g, TYPE b)

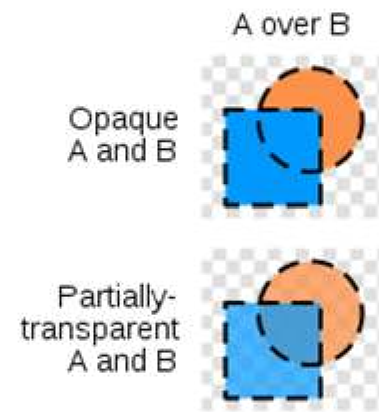
glColor3f(0.0, 0.0, 0.0) - black

http://fly.cc.fer.hr/~unreal/theredbook/chapter05.html

Specifying a Color in Color-Index Mode

glIndex{siffix}(TYPE c)

Compositing / Blending



- OpenGL composites images using alpha blenc
- The alpha value of an opaque object is 1. The transparency of an object is 1-alpha.
- A completely opaque object completely blocks from passing through it. A completely transparent object should not be visible.
- The ideas of source and destination bits from earlier apply to pixels here. The color and alpha value of a buffer pixel can depend on multiple objects in depth.

Blending in OpenGl

- Blending in enabled in OpenGL by

```
glEnable(GL_BLEND)
```

- Source and destination blending factors are set using

```
glBlendFunc(source_factor,
    destination_factor)
```

- Often, the source alpha is used for the source factor and 1-alpha is used for the destination factor giving a resulting pixel value

  $(R_{d'}, G_{d'}, B_{d'}, \alpha_{d'}) = (\alpha_s R_s + (1-\alpha_s) R_d, \alpha_s G_s + (1-\alpha_s) G_d, \alpha_s B_s + (1-\alpha_s) B_d, \alpha_s \alpha_d + (1-\alpha_s) \alpha_d)$
- The result of blending depends on order that the objects are rasterized.