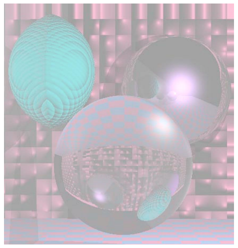
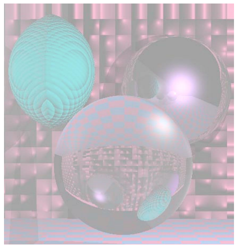


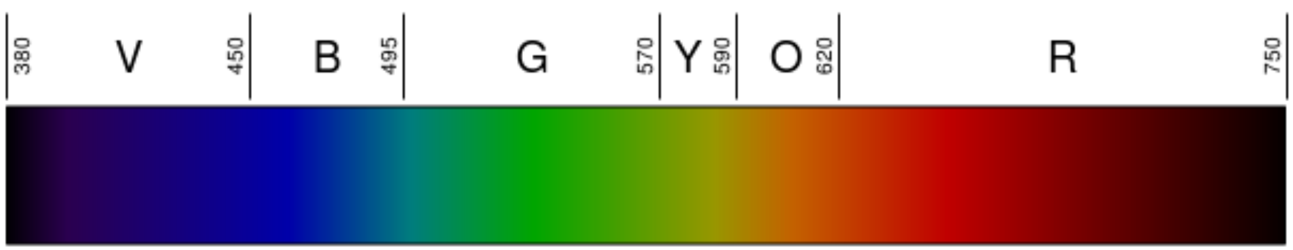
CS 4300
Computer Graphics

Prof. Harriet Fell
Fall 2012
Lecture 4 – September 12, 2012



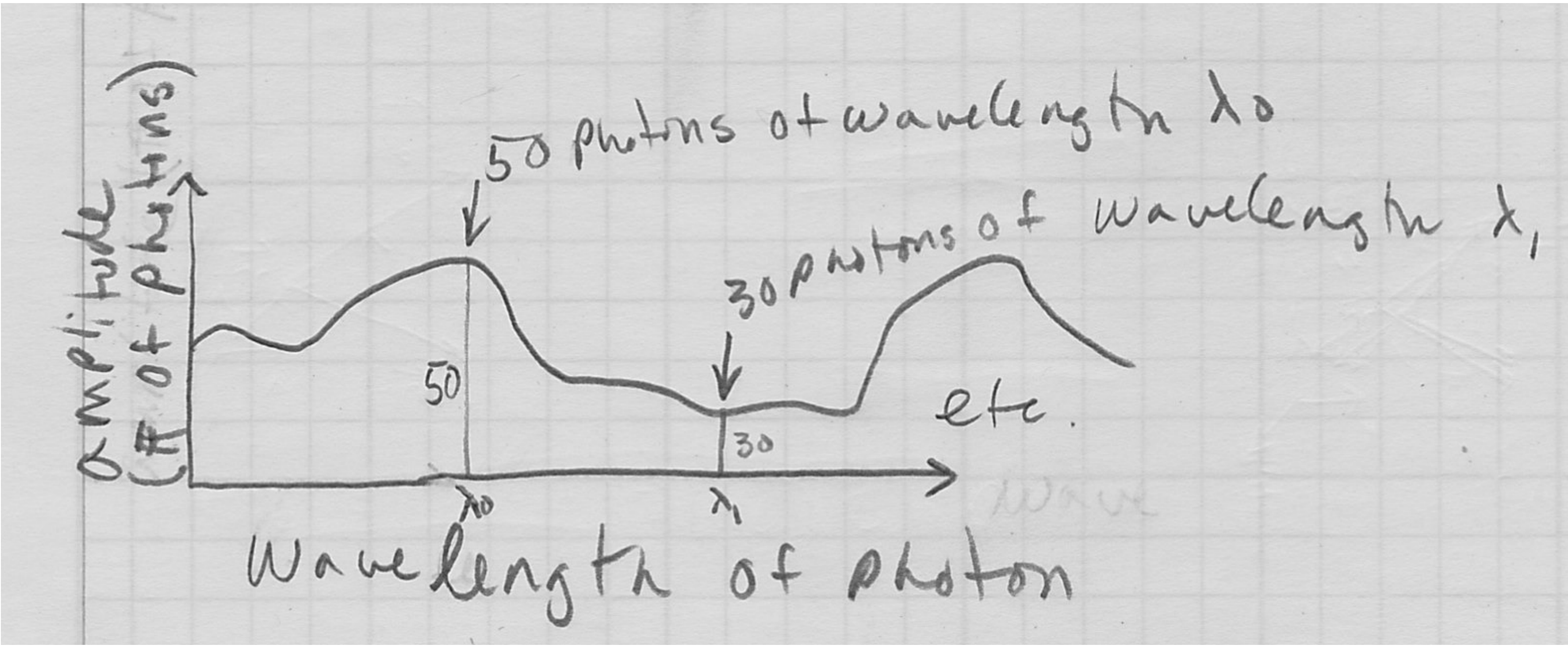
What is color?

- from physics, we know that the *wavelength of a photon (typically measured in nanometers, or billionths of a meter) determines its apparent color*
- *we cannot see all wavelengths, but only the visible spectrum from around 380 to 750 nm*



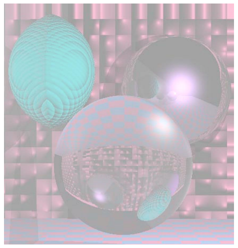
Where are the other colors?

- but where are the following colors: “brown”, “pink”, “white”, …?
- clearly, the *color spectrum does not actually contain all colors; some colors are non-spectral*
- *generally, a large number of photons with different wavelengths are simultaneously impinging on any given location of your retina*



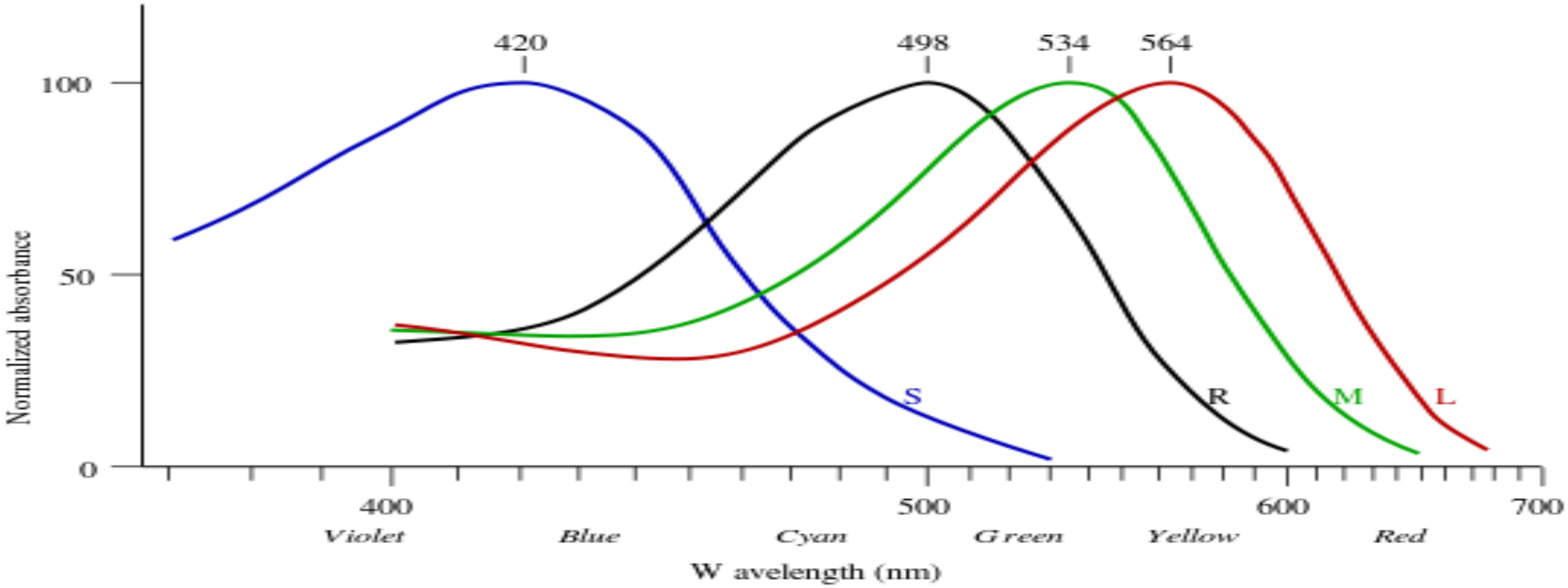
Marty Vona's sketch

- the actual incident light is not of a single wavelength, but can be described by a spectral histogram
- the histogram represents the relative quantity of photons of each wavelength



Human Perception of Color

- the human eye cannot determine the exact histogram
- in fact just representing a complete spectral histogram exactly would require an infinite amount of space because it's a continuous quantity
- the biological solution is another form of sampling
- three types of cone cells respond (with the equivalent of a single number each) to the degree to which the actual incident histogram is similar to response histograms with peaks near red, green, and blue



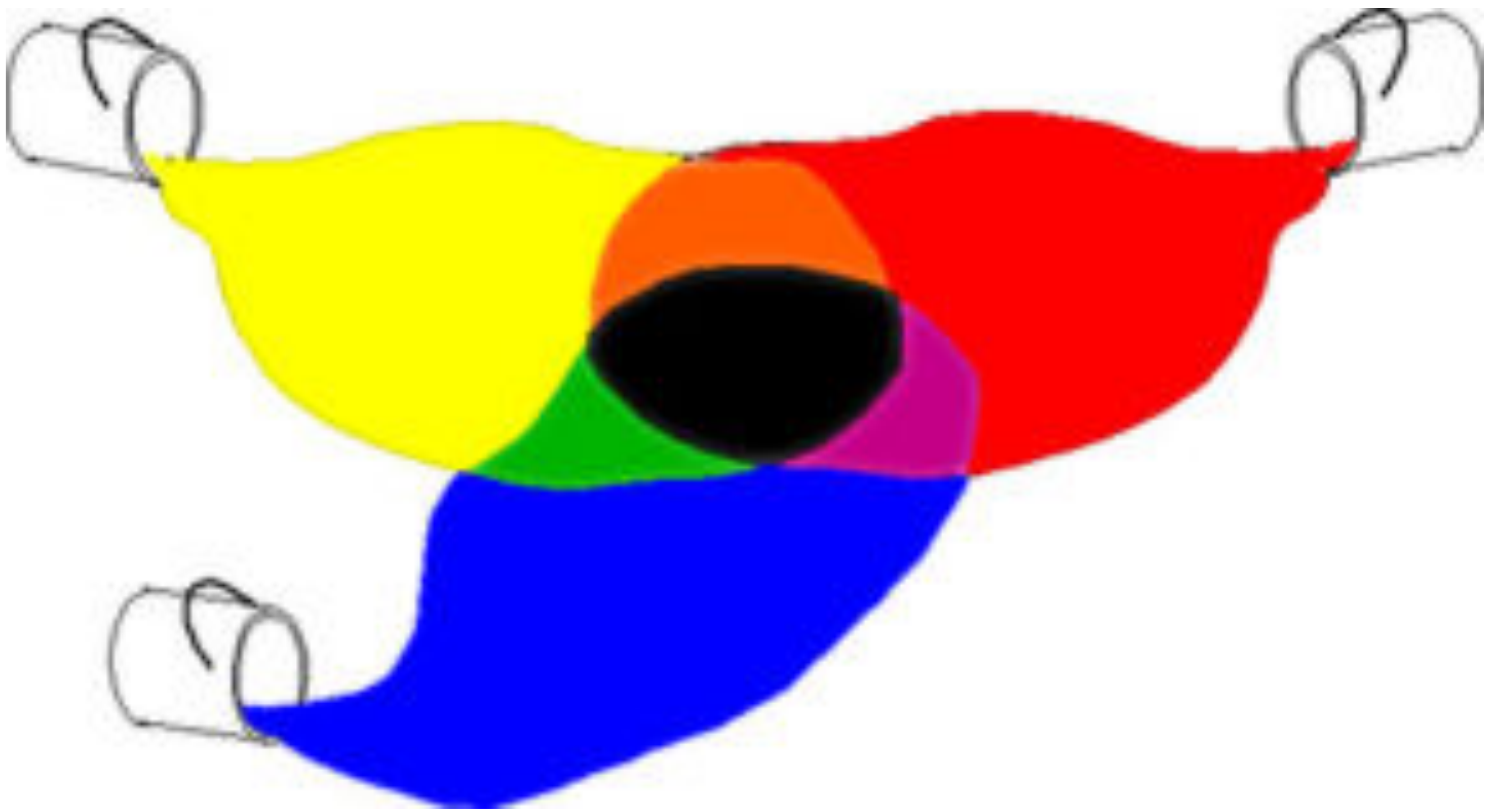
- so the original continous histogram impinging on one location of your retina is reduced to three measurements
- (actually, there is a fourth *rod cell type, which is mainly active in low light conditions)*
- *color blindness is typically caused by anomalies in the types of cone cells*
- *other animals also have different cone cells*

- because we have converted a continuous object into a set of discrete samples, we have to consider *aliasing*
  - *different incident histograms, called metamers, may be mapped to the same set of cone cell responses*
  - *how many distinct colors can be seen?*
  - *one way to think about it is to know that each cone cell type can distinguish between about 100 intensity levels of the associated response curve, and then to take a constructive approach*
  - *there are ~1M ways to combine cone cell responses, so an average human can distinguish roughly that many colors*
- the biology of human cone cells is the not only the reason we often use RGB to represent color; in fact, it defines color. Color is not an intrinsic property of light, but rather a result of the interaction between human cone cells and histograms of incident light.

From the Hubble

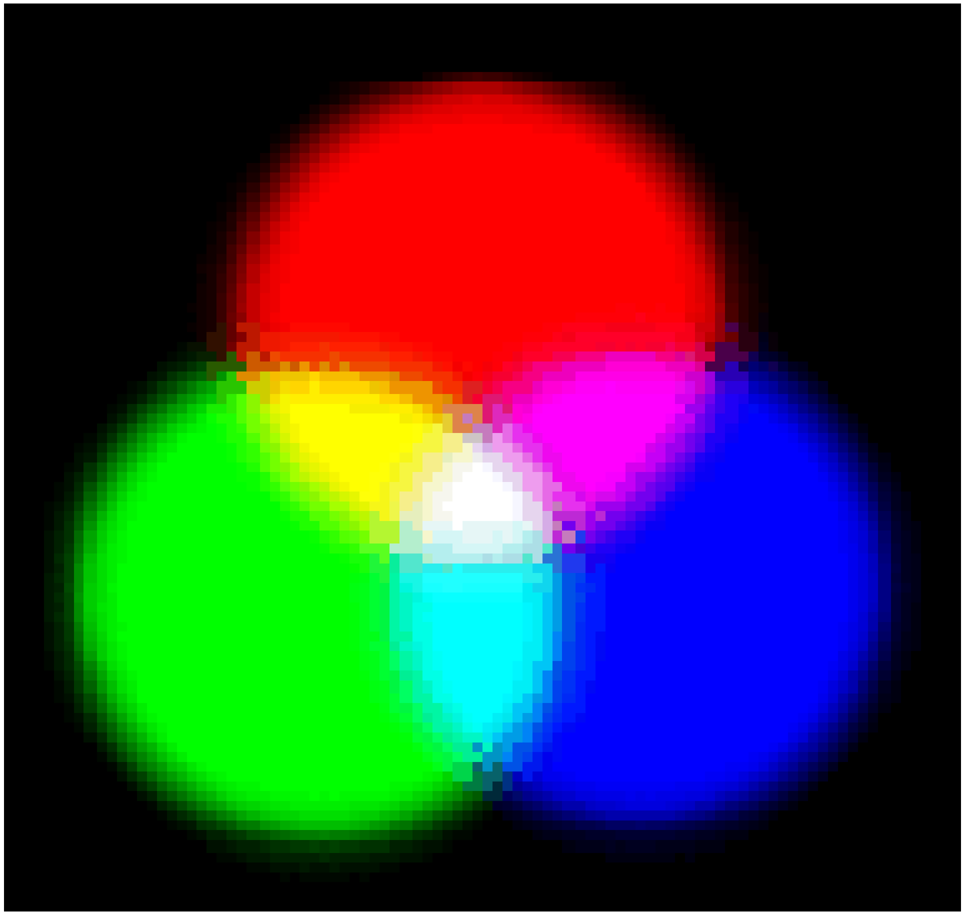
Hubble Site Link

Color

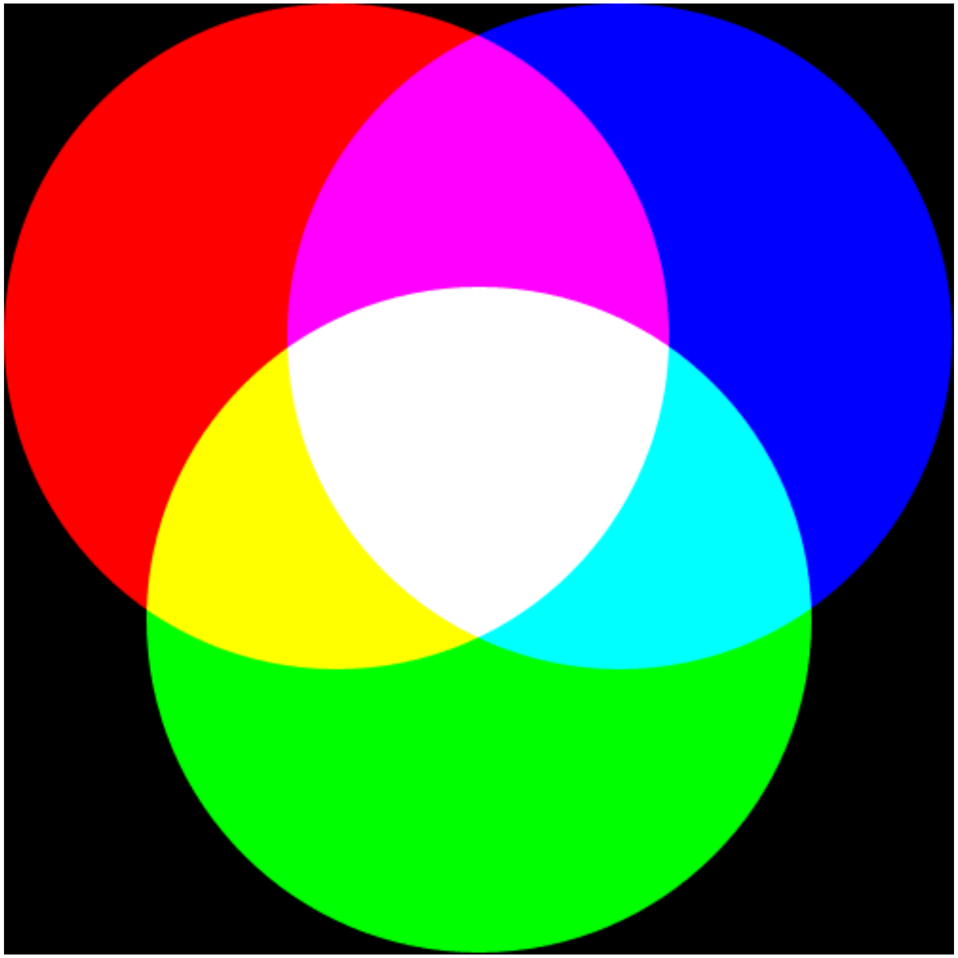


www.thestagecrew.com

Red, Green, and Blue Light

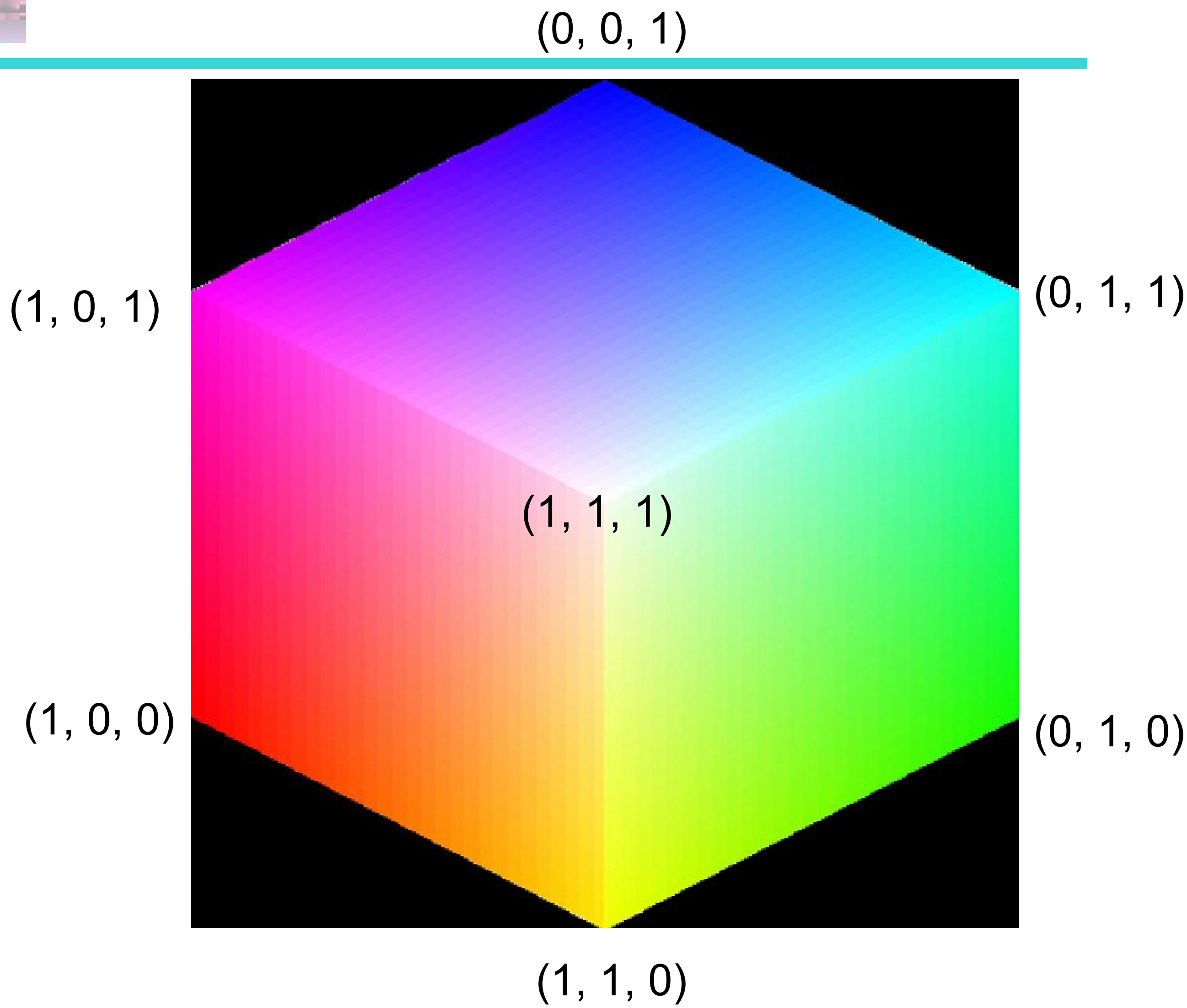


Adding R, G, and B Values

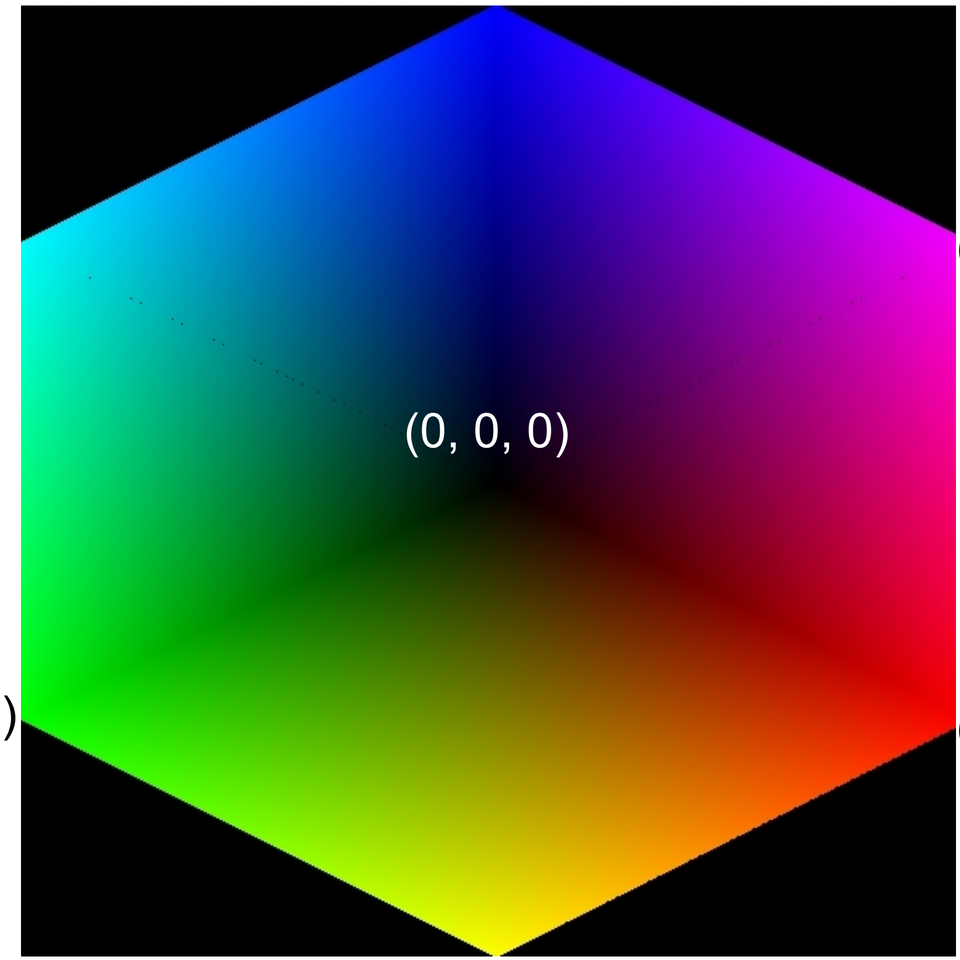


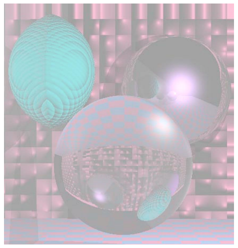
http://en.wikipedia.org/wiki/RGB

RGB Color Cube

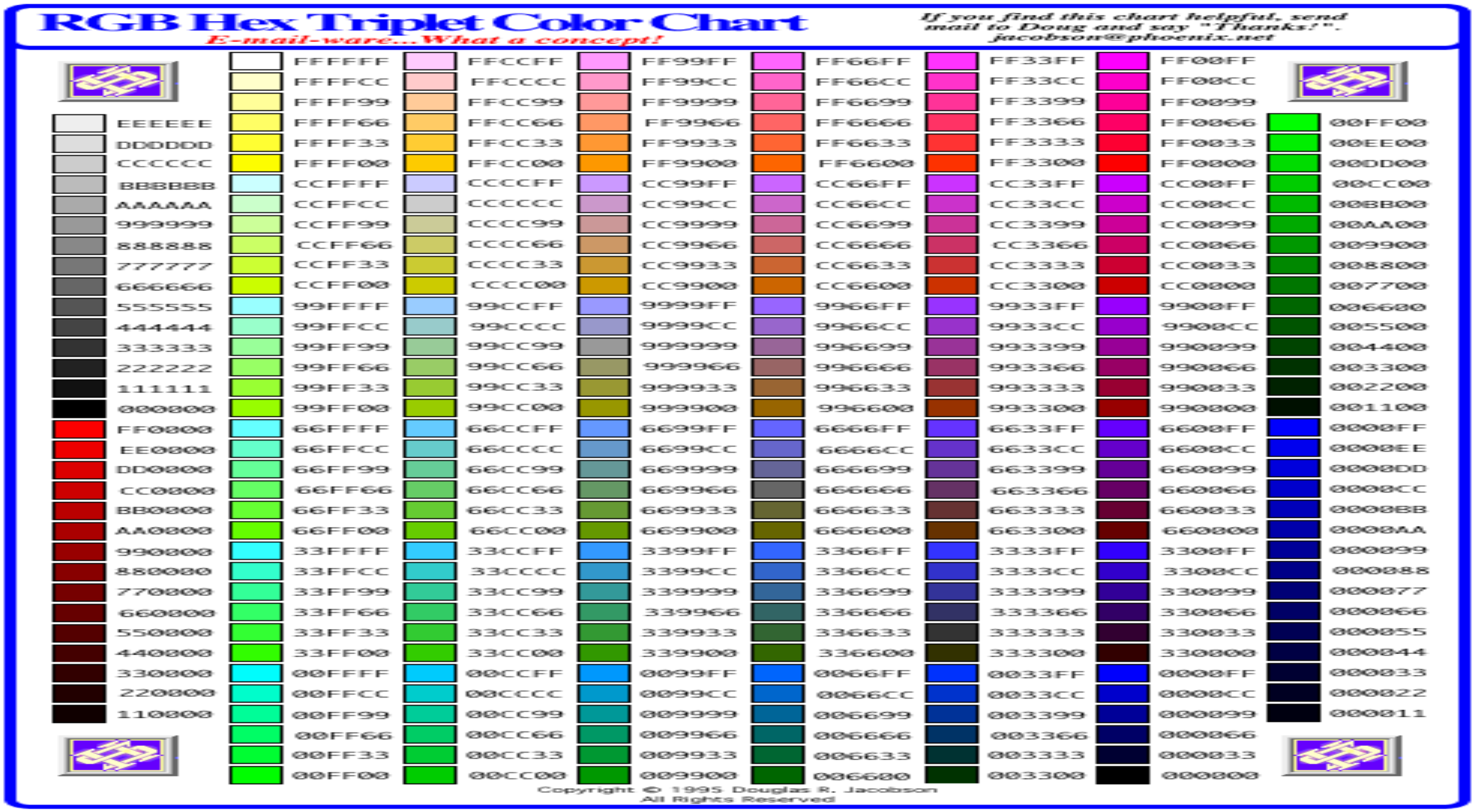


RGB Color Cube The Dark Side

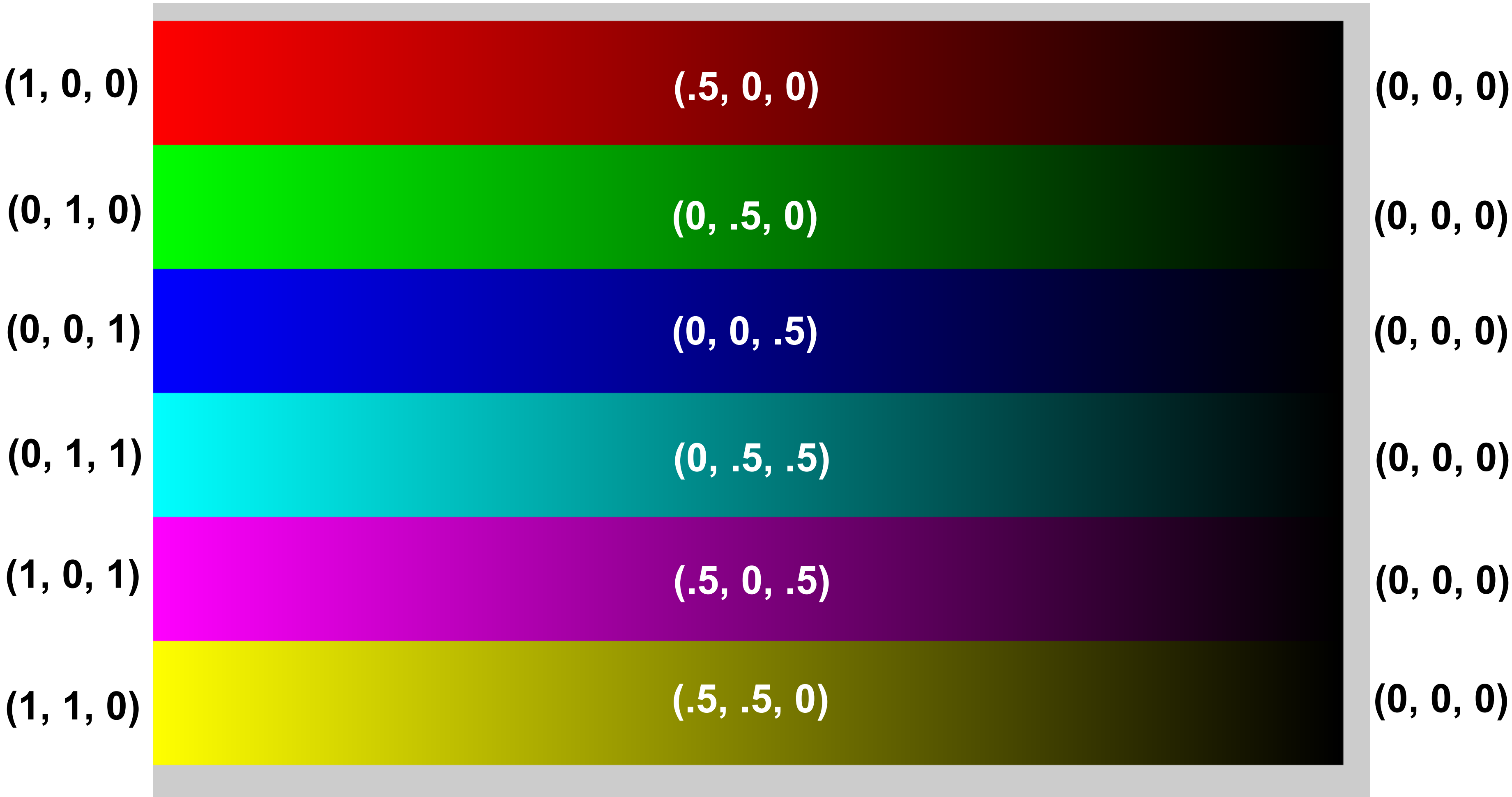




Doug Jacobson's RGB Hex Triplet Color Chart

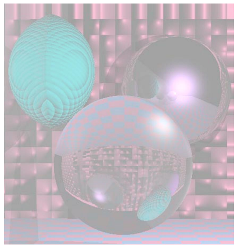


Making Colors Darker

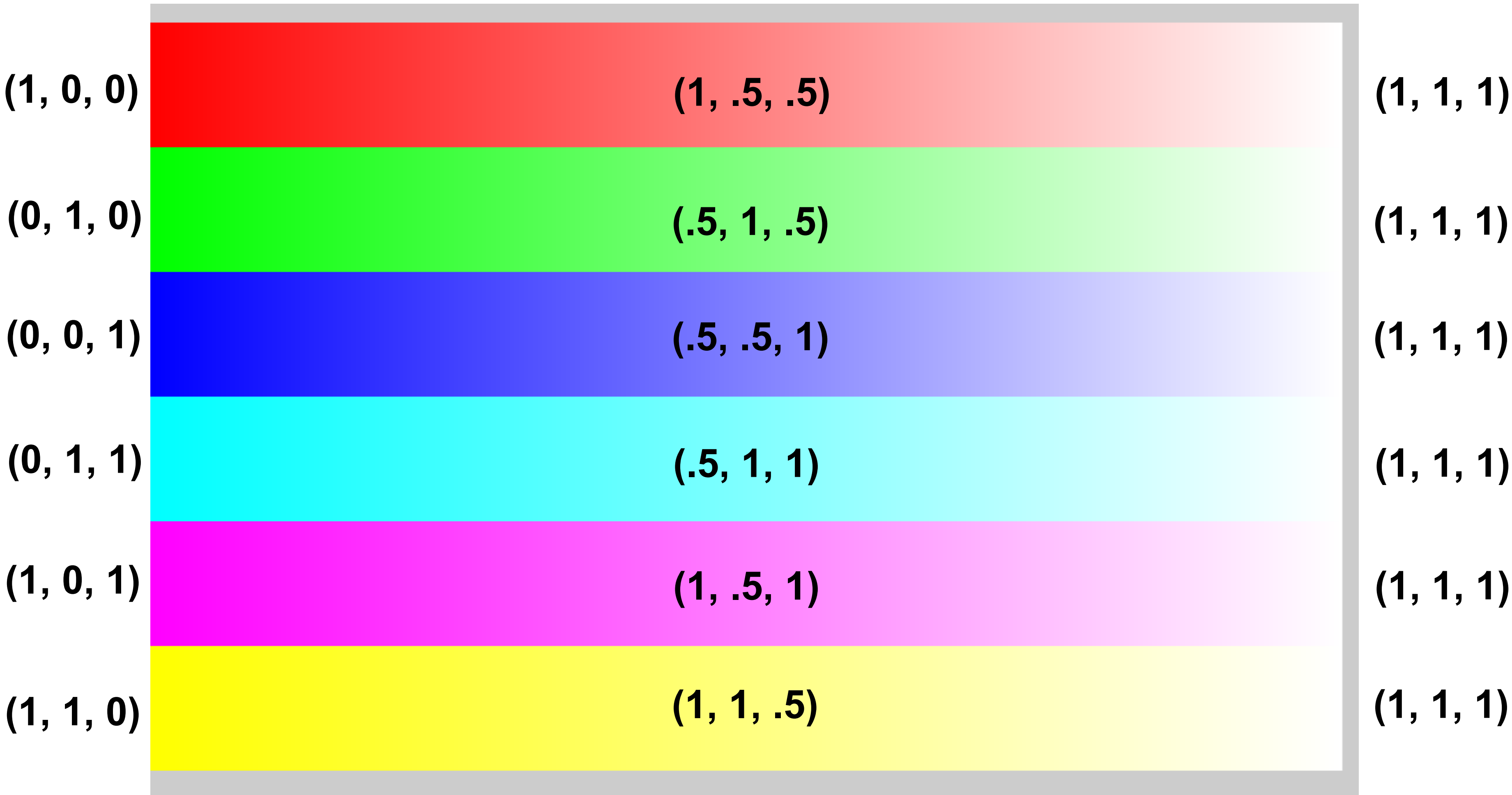


Getting Darker, Left to Right

```
for (int b = 255; b >= 0; b--){
    c = new Color(b, 0, 0); g.setPaint(c);
    g.fillRect(800+3*(255-b), 50, 3, 150);
    c = new Color(0, b, 0); g.setPaint(c);
    g.fillRect(800+3*(255-b), 200, 3, 150);
    c = new Color(0, 0, b); g.setPaint(c);
    g.fillRect(800+3*(255-b), 350, 3, 150);
    c = new Color(0, b, b); g.setPaint(c);
    g.fillRect(800+3*(255-b), 500, 3, 150);
    c = new Color(b, 0, b); g.setPaint(c);
    g.fillRect(800+3*(255-b), 650, 3, 150);
    c = new Color(b, b, 0); g.setPaint(c);
    g.fillRect(800+3*(255-b), 800, 3, 150);
}
```



Making Pale Colors



Getting Paler, Left to Right

```
for (int w = 0; w < 256; w++){
        c = new Color(255, w, w); g.setPaint(c);
        g.fillRect(3*w, 50, 3, 150);
        c = new Color(w, 255, w); g.setPaint(c);
        g.fillRect(3*w, 200, 3, 150);
        c = new Color(w, w, 255); g.setPaint(c);
        g.fillRect(3*w, 350, 3, 150);
        c = new Color(w, 255, 255); g.setPaint(c);
        g.fillRect(3*w, 500, 3, 150);
        c = new Color(255,w, 255); g.setPaint(c);
        g.fillRect(3*w, 650, 3, 150);
        c = new Color(255, 255, w); g.setPaint(c);
        g.fillRect(3*w, 800, 3, 150);
}
```

Additive and Subtractive Color Space

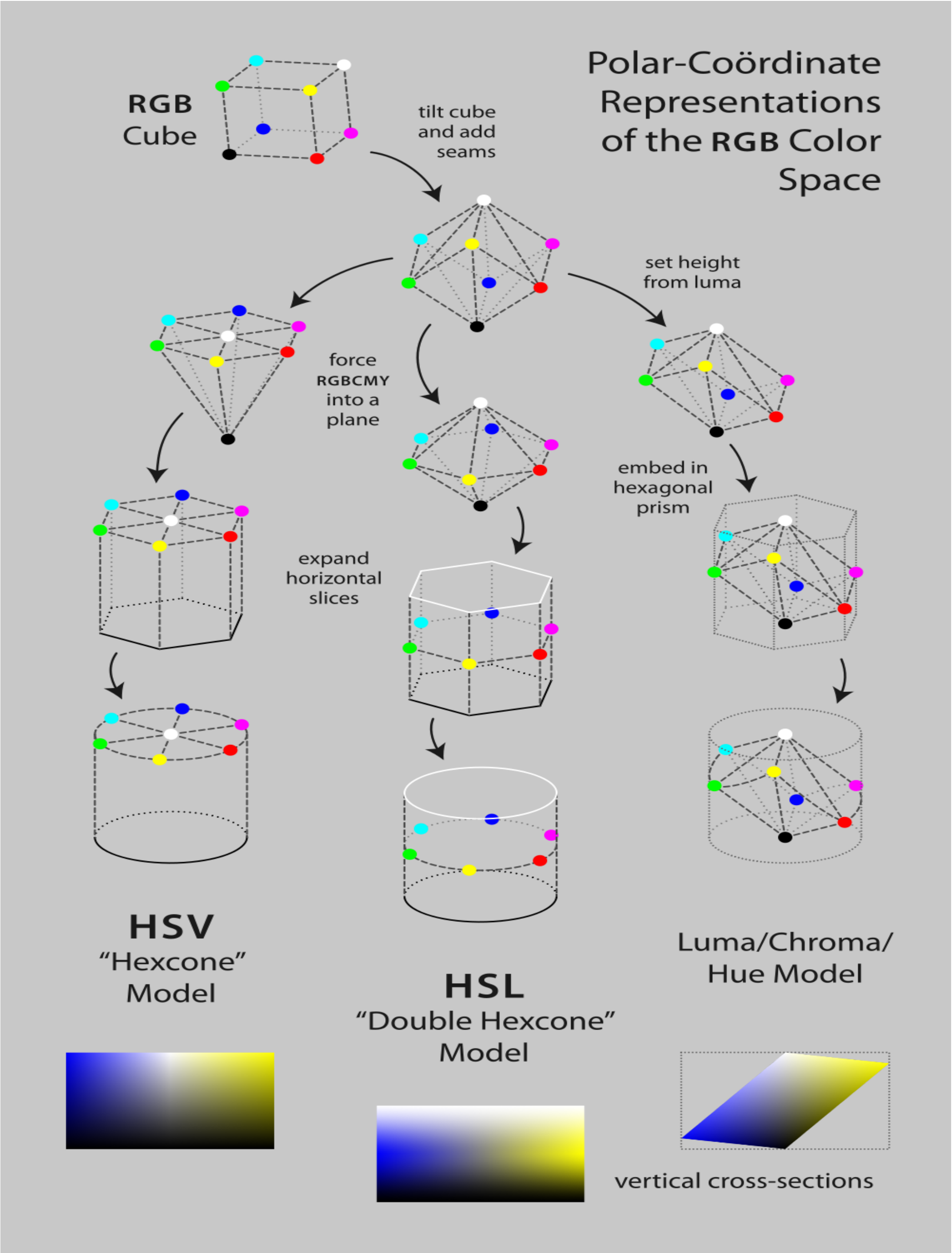
- sometimes RGB are considered “additive” colors because they form a basis for the color space relative to black
- CMY can similarly be considered “subtractive” colors because, effectively
  - cyan+red = white
  - magenta+green = white
  - yellow+blue = white

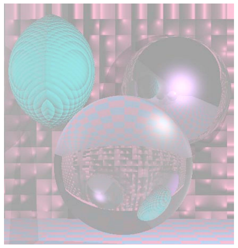
Display vs. Print

- additive colors typically used when light is generated by an output device (e.g. CRT, LCD)
- subtractive colors typically used when printing on white paper
- sometimes RGB and CMY are considered distinct color spaces

HSV Color Space

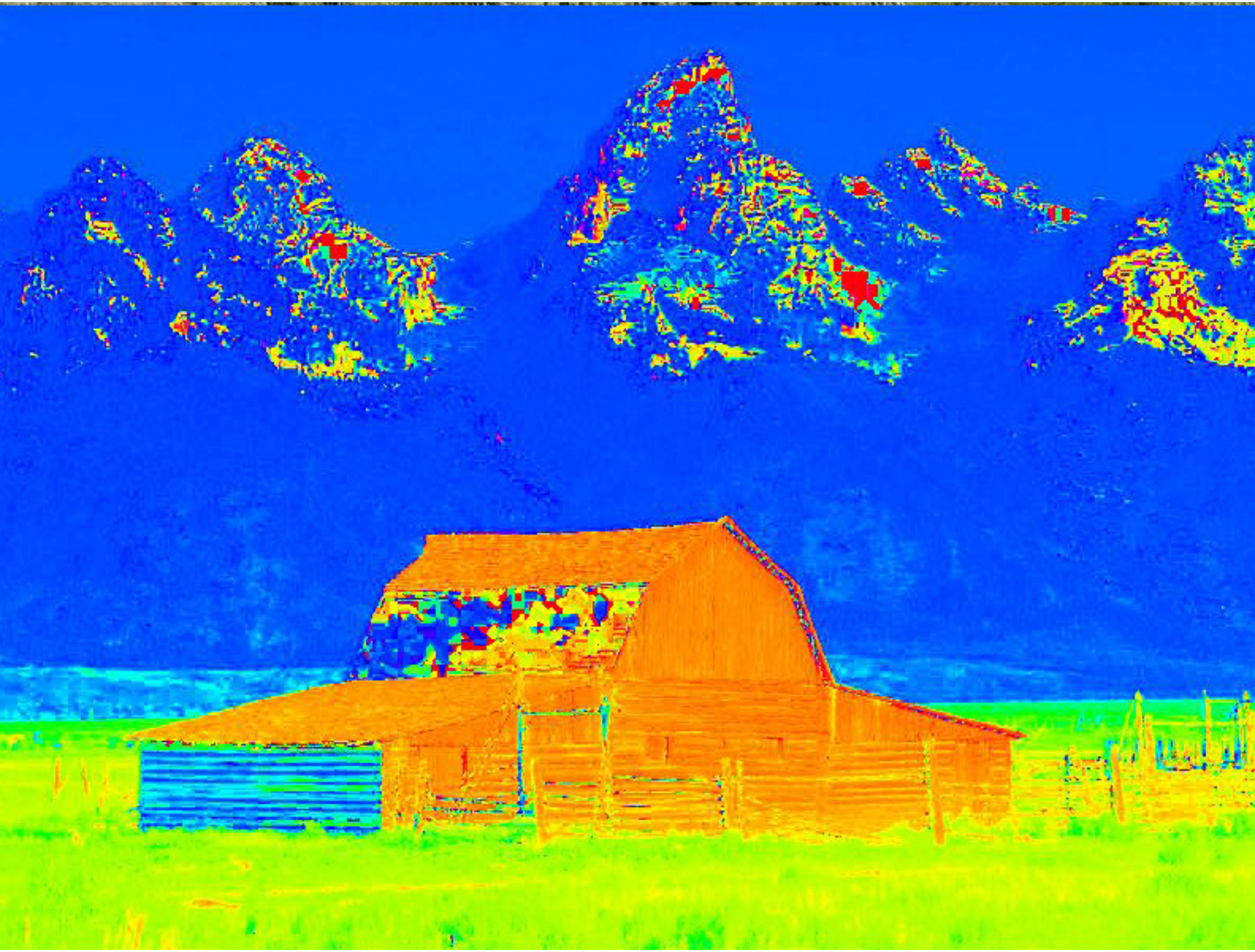
- ***hue:*** *the basic color, or chromaticity*
- ***saturation****: how “deep” the color is (vs “pastel”)*
- ***value:*** *the brightness of the color*


Polar-Coördinate Representations of the RGB Color Space
RGB Cube
tilt cube and add seams
set height from luma
force RGBCMY into a plane
embed in hexagonal prism
expand horizontal slices
HSV "Hexcone" Model
HSL "Double Hexcone" Model
Luma/Chroma/ Hue Model
vertical cross-sections



RGB to HSV

- HSV is again a 3 dimensional space, but it is typically considered to use *cylindrical coordinates*
  - *this is mainly a construction to decompose the three dimensional color space in a way that is more useful to human designers*
  - *also often useful in machine vision algorithms, which simulate our theories of (aspects of) human vision*
  - *can visualize HSV space as a “morph” of RGB space*
    - *“stretch” the white and black vertices up and down*
    - *“line up” the remaining six vertices along a common horizontal plane*
    - *for HSV, put the white vertex back onto plane*
  - *(a variation, HSL, keeps white and black symmetrically above and below)*



Try the color picker