Color and Images

Lecture 20

Colors in CSS

- Use: fonts, borders, backgrounds
- Provides semantic signal:

  Green – go, success, complete, solution

  Red – stop, failure, incomplete, problem

  Yellow – yield, warning, attention
- Helps to set mood/emotion/tone:

  Bright – cheerful, playful, positive

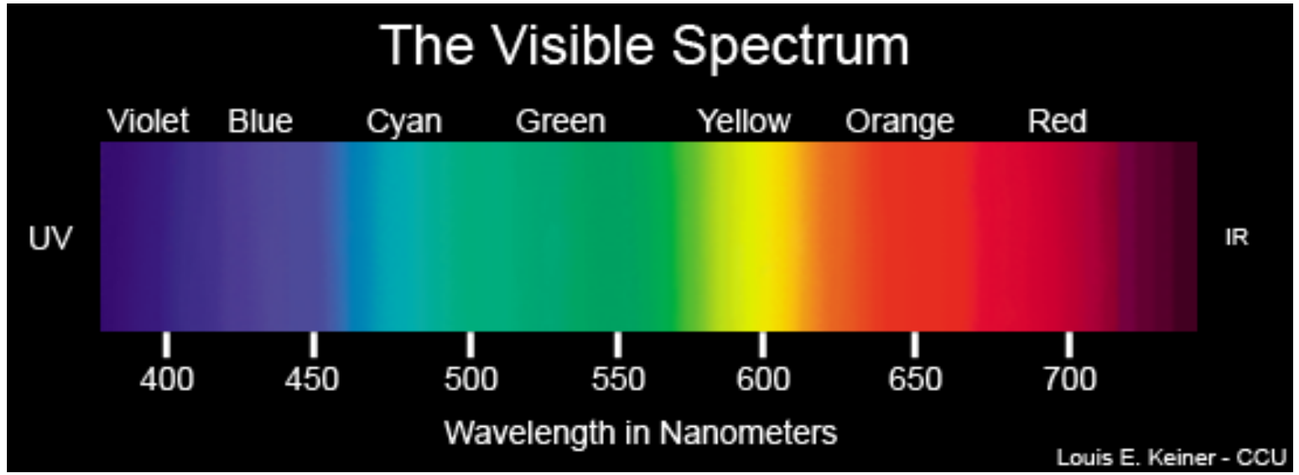
  Dark – somber, serious, negative

  Warm – energetic, alert, active

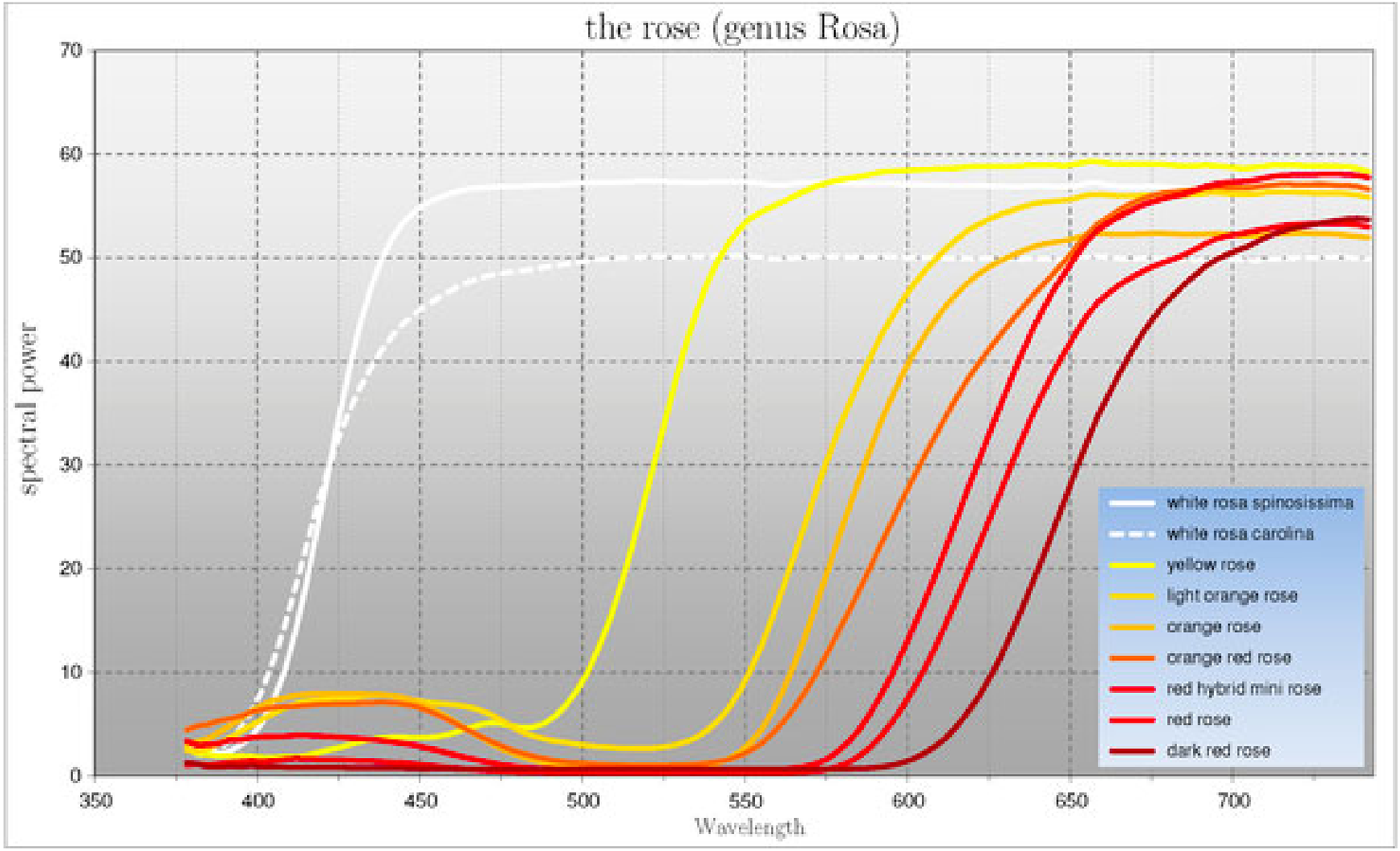
  Cool – calm, tranquil, peaceful

Elementary Color Theory

- Combination of
  - Physics: wavelengths in nm
  - Biology: perception of “red” vs “yellow” vs…
- Visible spectrum: 390-700nm
  - Spectral colors: rainbow, single wavelength
  - Nonspectral colors (pink, brown, white…) result from presence of *multiple* wavelengths

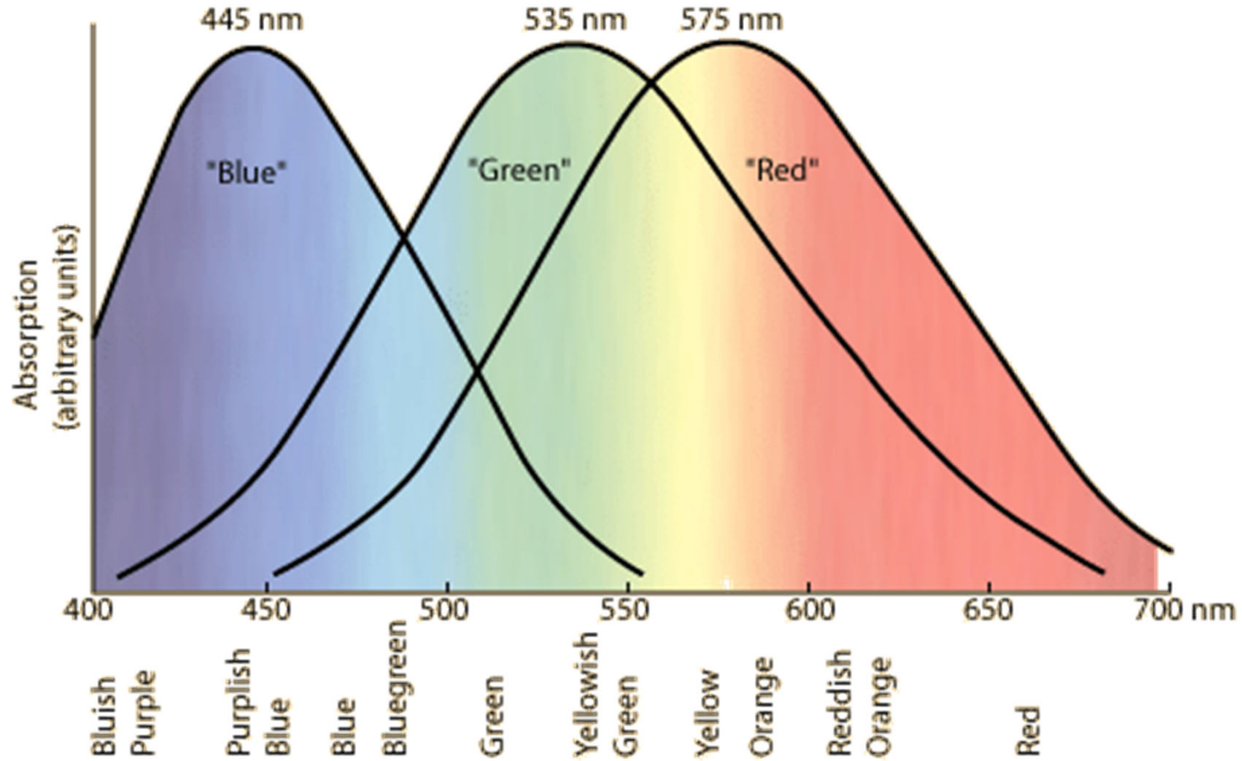


Power Spectrum = Color



Color Perception

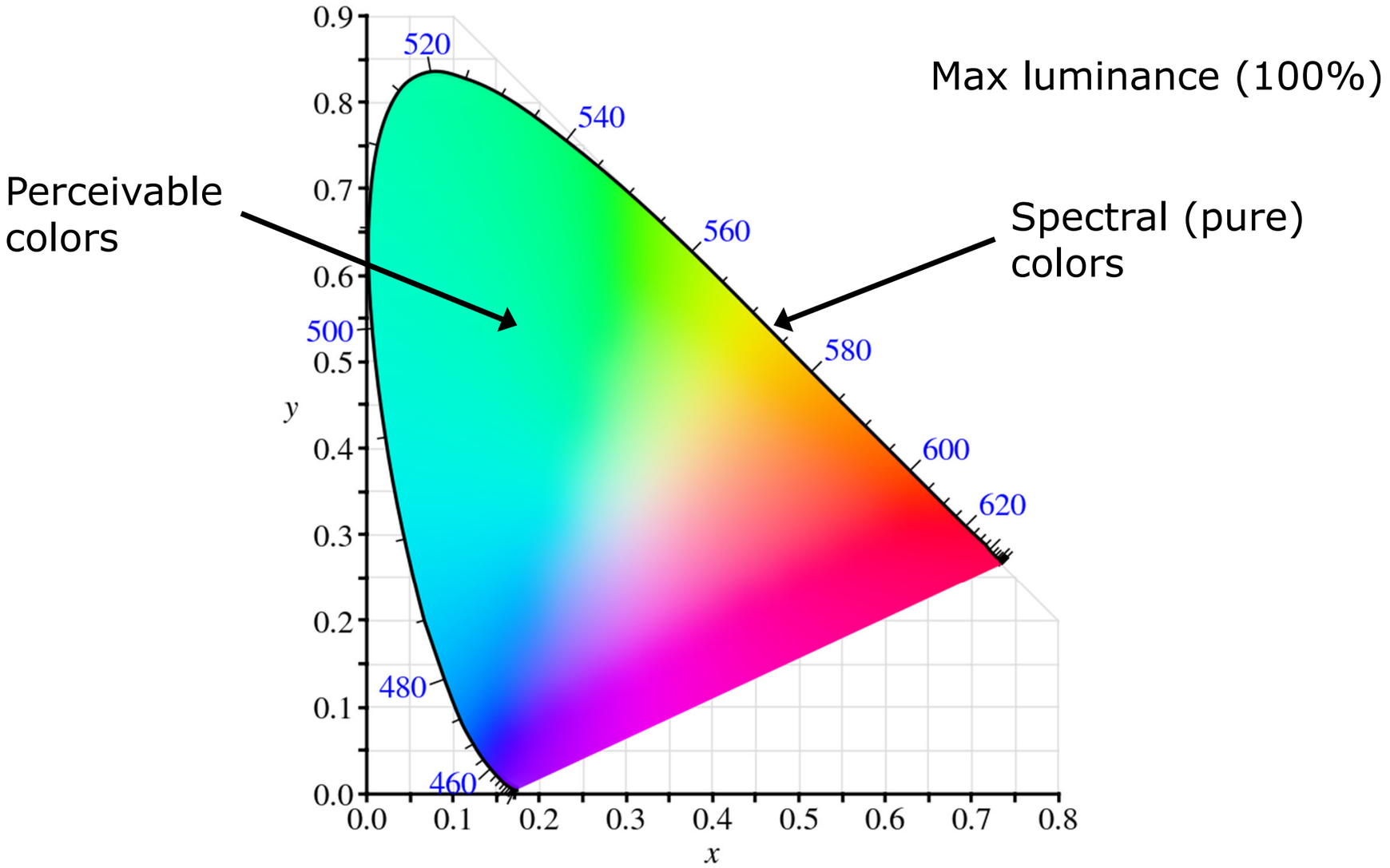
- Human eyes have 3 types of cones
  - Respond to different wavelengths (LMS)
- Response from each type of cone → perceived color



Metamerism

- Different (continuous) spectra can stimulate our eyes in identical ways
  - Consequence: Different spectra with indistinguishable (to humans) color
- Example: white
  - Spectrum 1: all wavelengths equally present
  - Spectrum 2: a few wavelengths present, stimulating LMS cones equally
- Consequence: Any continuous spectrum can be projected down to 3 components (as far as human eyes are concerned)
  - XYZ “tristimulus values”
  - Not truly independent (overlap of response), so any 2 give the 3rd; ie a 2D space (xy)…

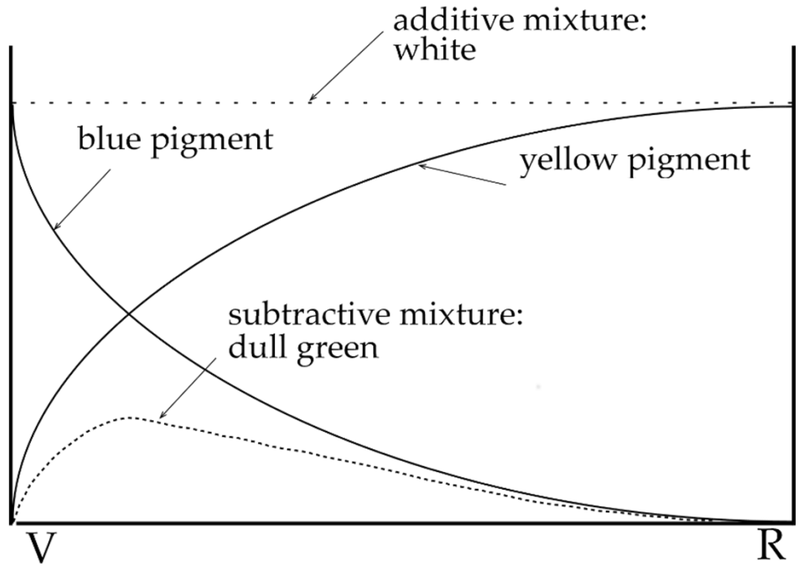
CIE 1931 xy Chromaticity



Color Mixing

- There are two ways to combine colors

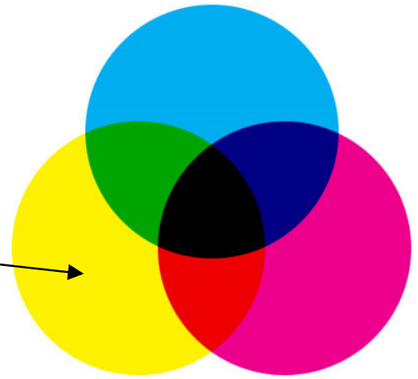
1. Subtractive: Color is a *filter*
   - Mixing = filter out both
   - Used for printing (& dyes, paints, gels)
2. Additive: Color is a *light source*
   - Mixing = sum both
   - Used for monitors



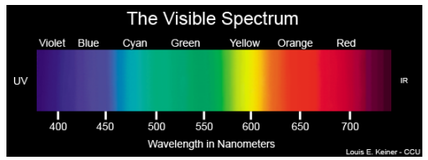
Subtractive Color Mixing: CMYK

- Filters transmit different *spectra*
  - Mixture transmits the *product* of both
  - Mix all three primaries = black
- Primary colors: cyan, magenta, yellow
  - Black (K) added for quality and cost
  - Traditional set (RYB) popular for painting

Primary yellow
(transmits R & G)
(absorbs B)

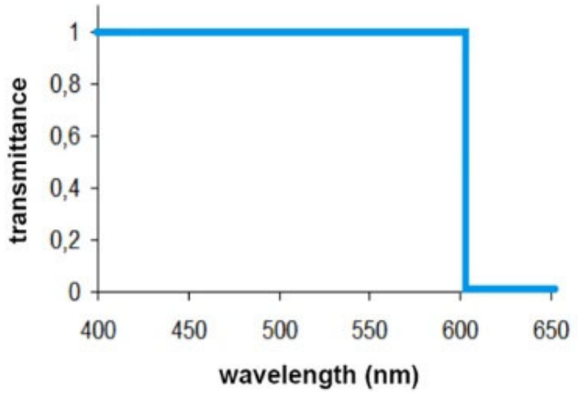


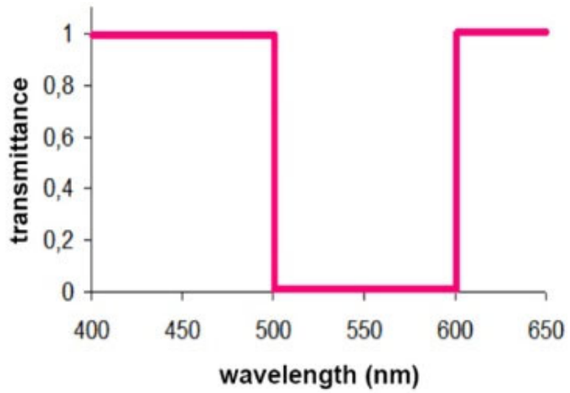
Colors as Filters

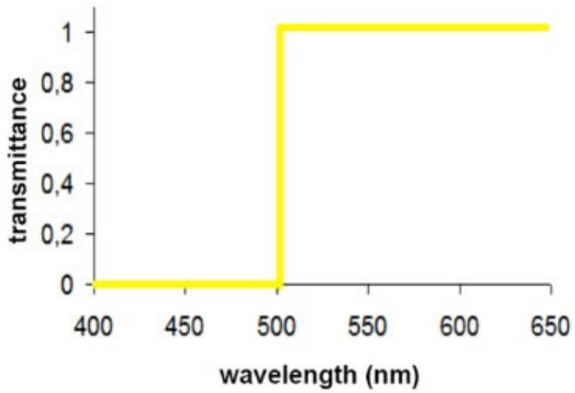


Yellow:
Filters out (only) blue

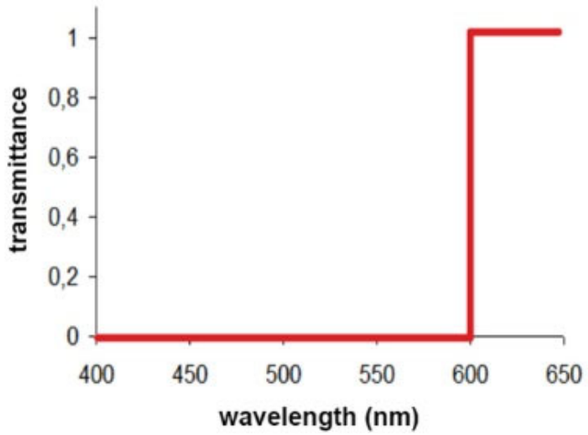
Primary

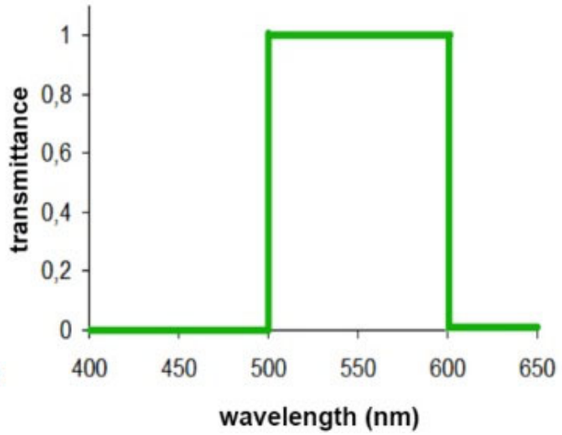


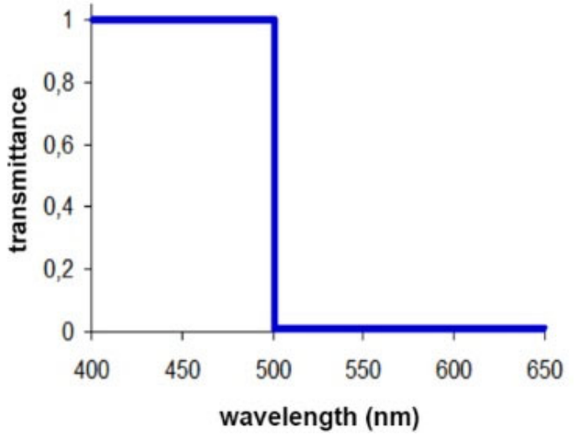




Secondary
(mixes)

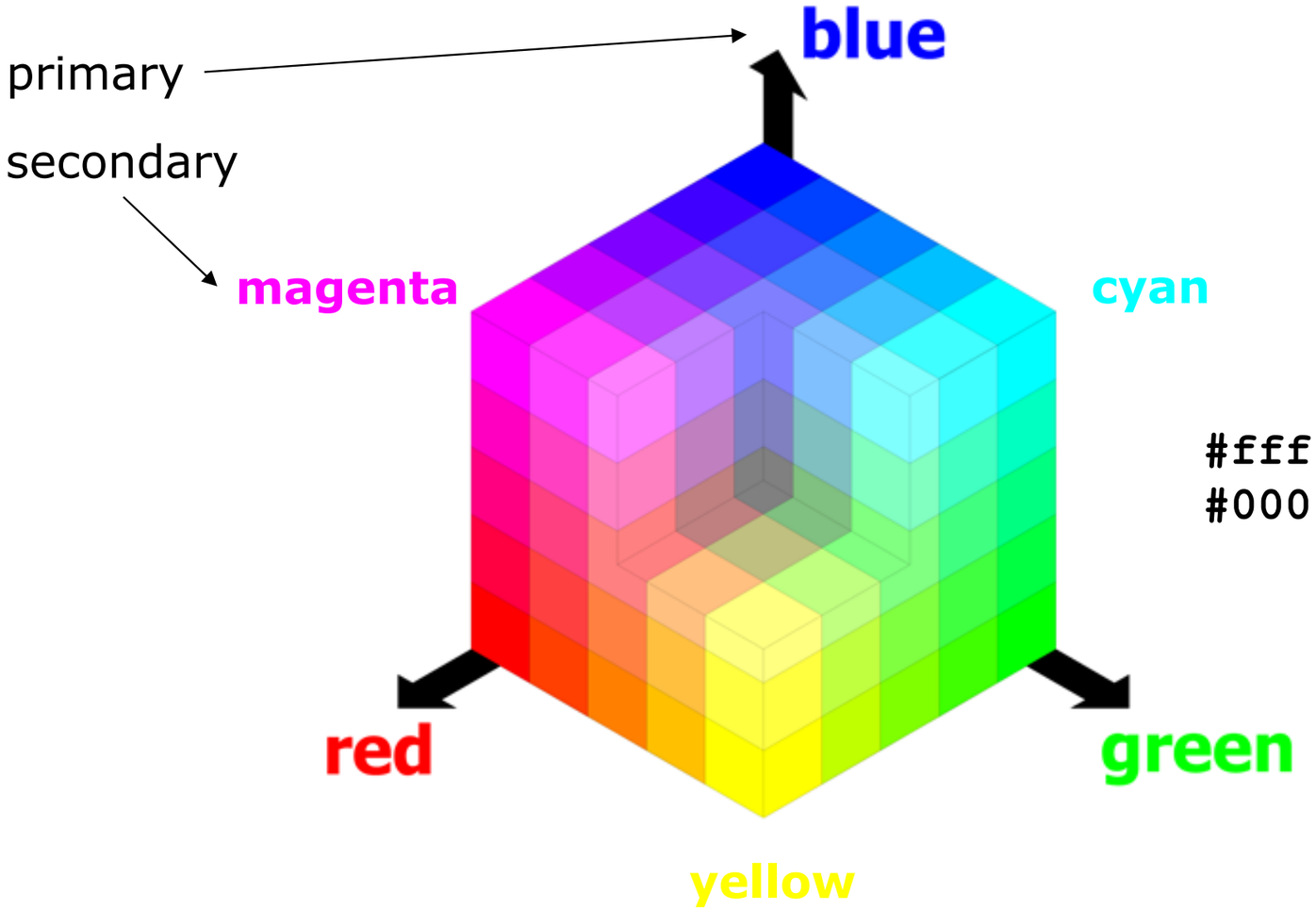






Rosi et al., Euro. J. of Physics, 37(6), 2016

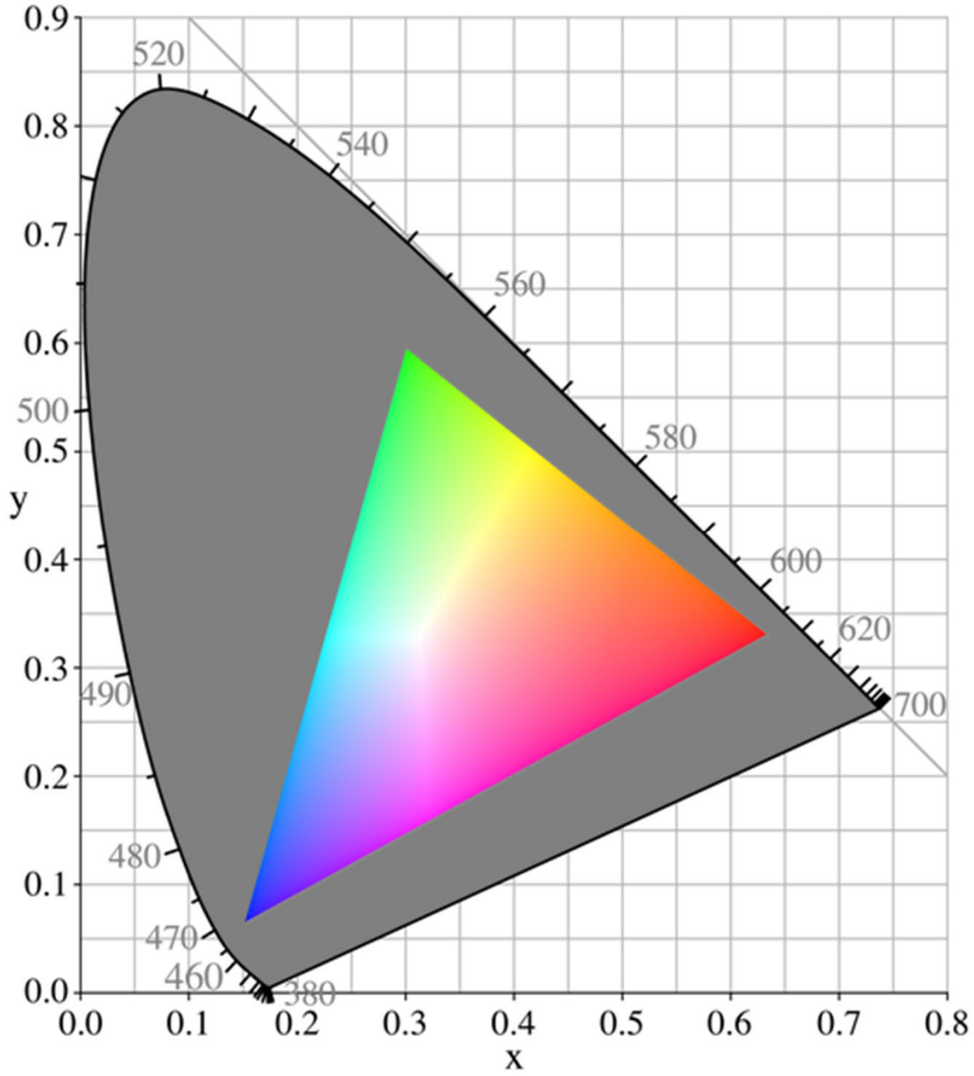
Additive Color Mixing: RGB Cube



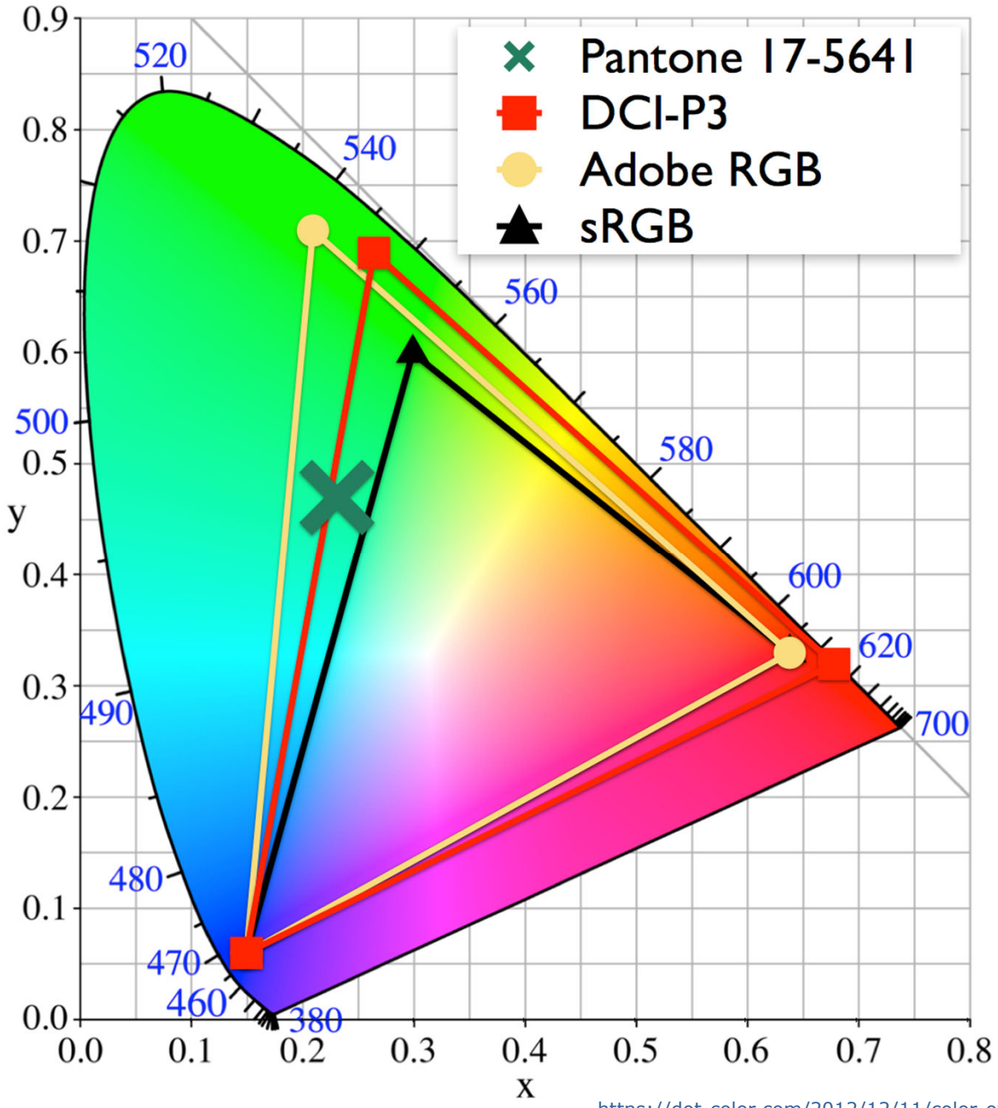

```
#fff /* white */
#000 /* black */
```

http://www.flickr.com/photos/ethanhein/3103830956/

Color Mixing: sRGB Gamut

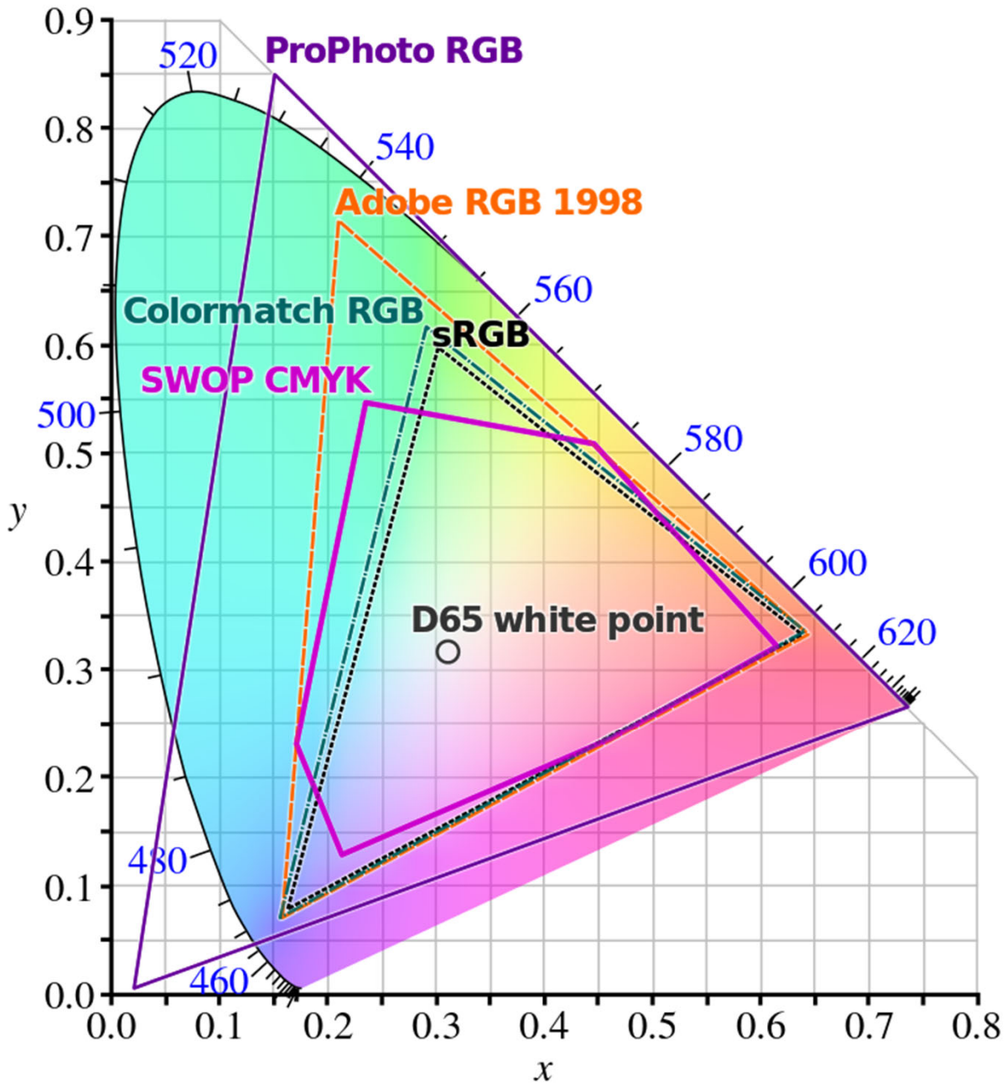


Gamuts for Monitors

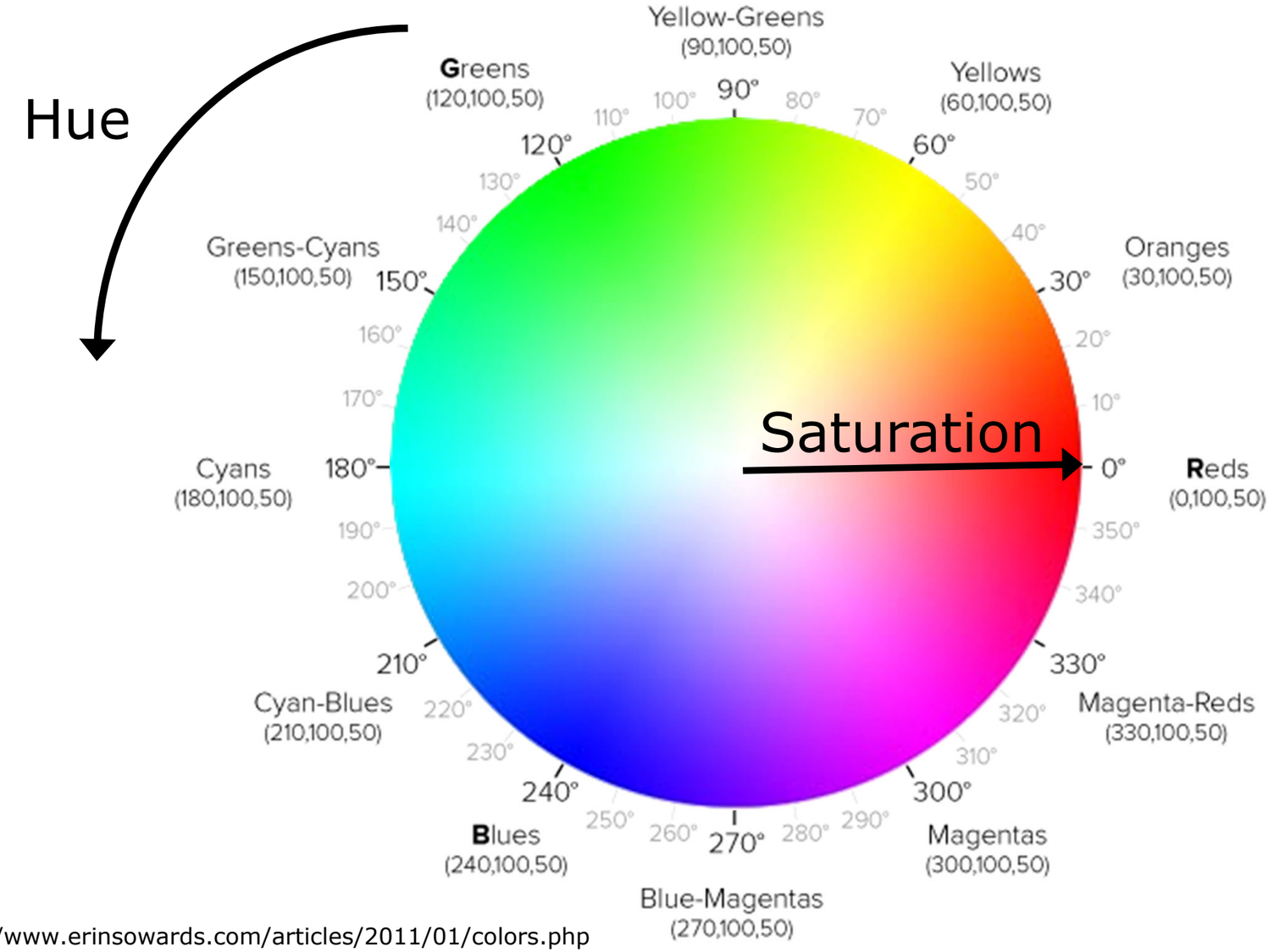


https://dot-color.com/2012/12/11/color-of-the-year-for-2013-falls-outside-srgb-gamut/

And Many More Gamuts…

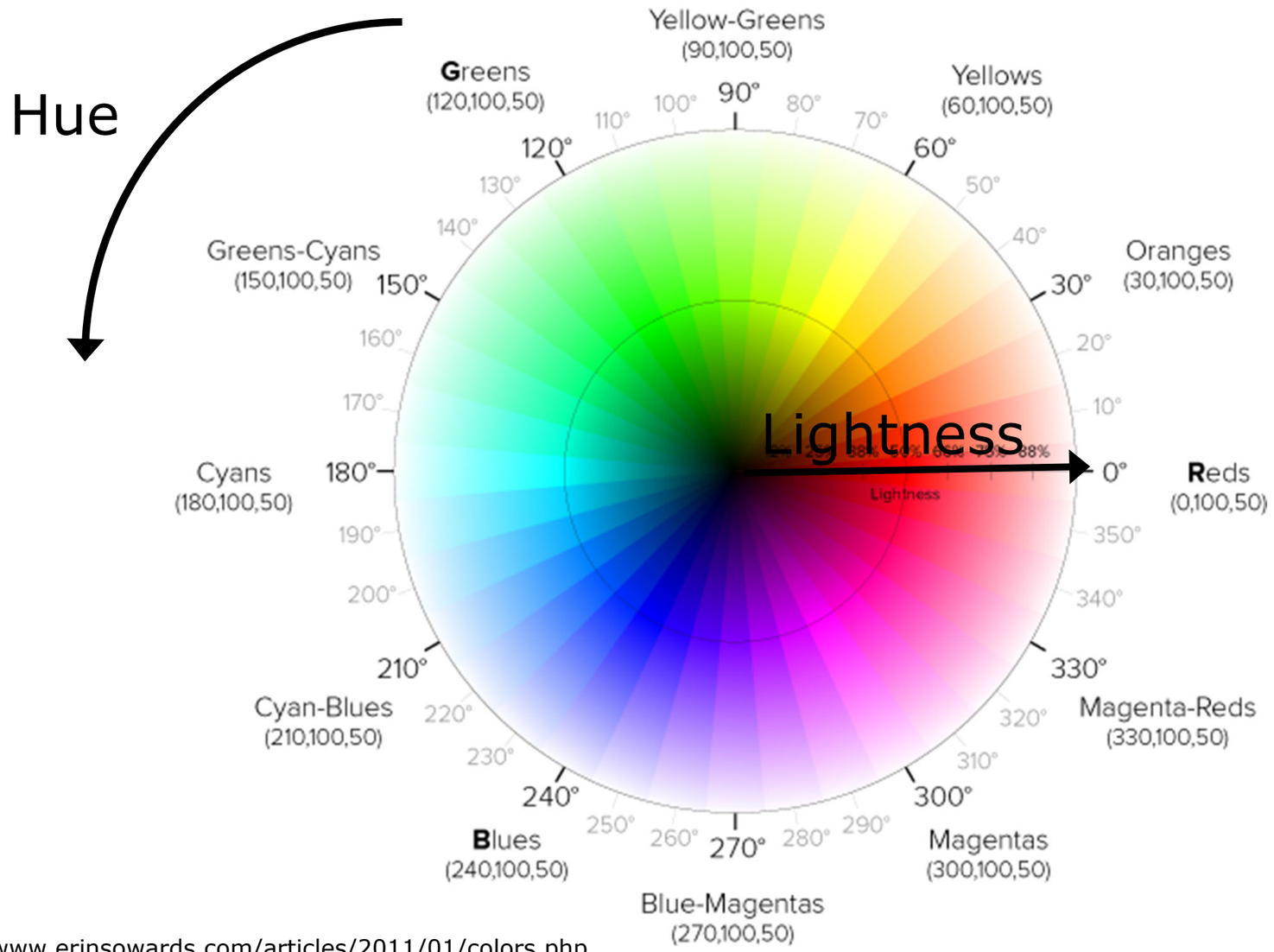


HSL Color Wheel (50% Lightns)



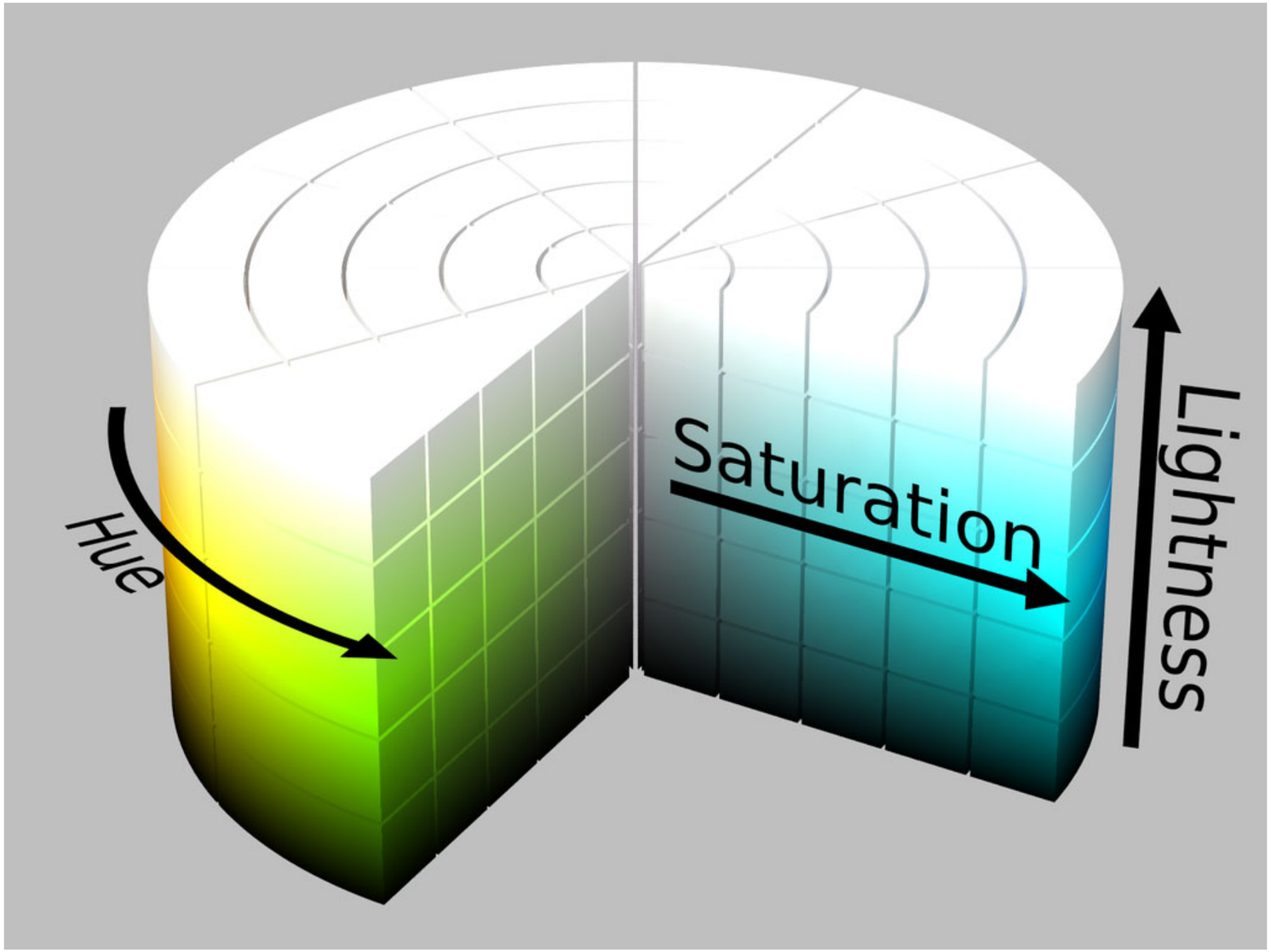
http://www.erinsowards.com/articles/2011/01/colors.php

HSL Color Wheel (100% Sat)

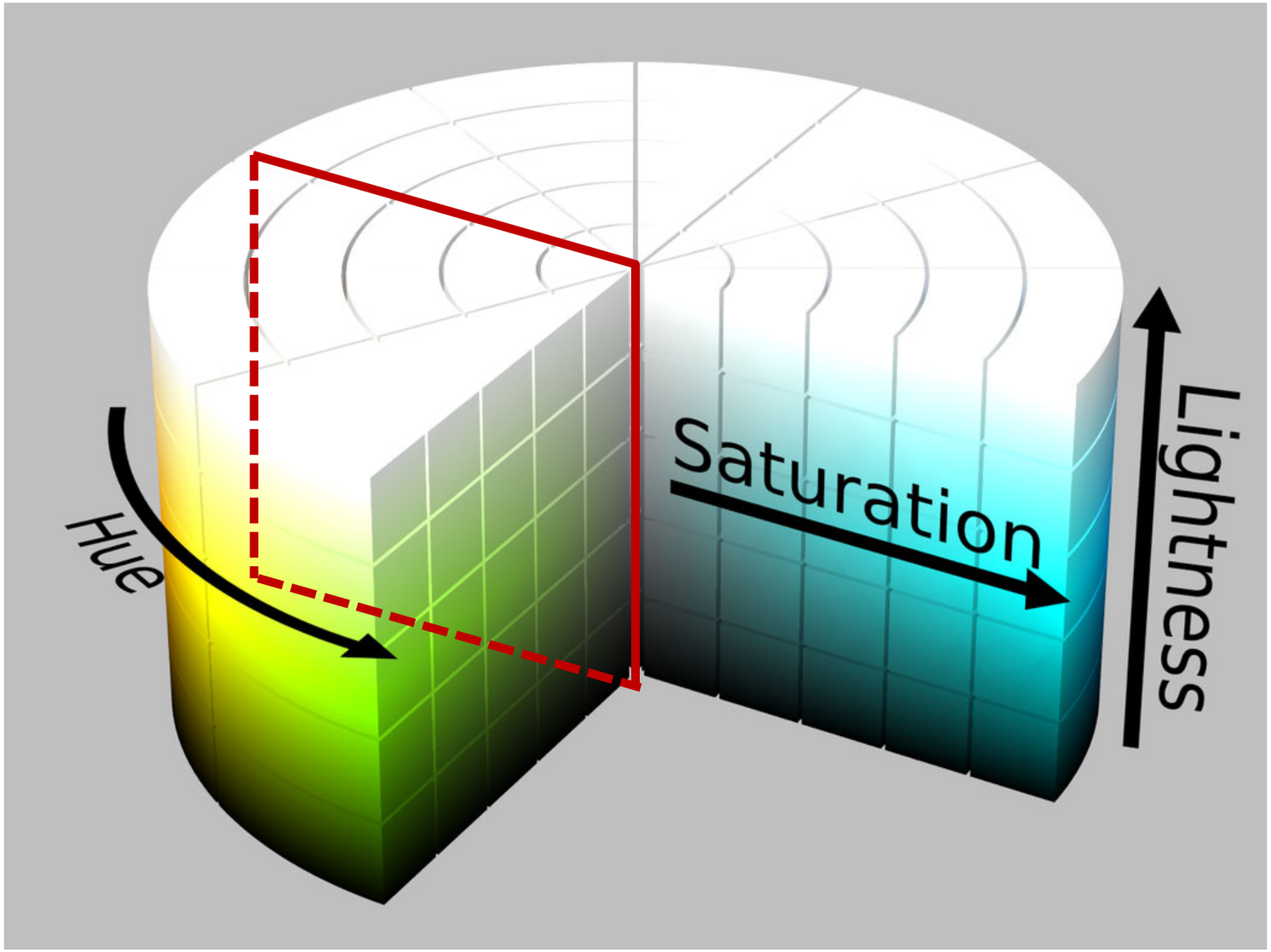


http://www.erinsowards.com/articles/2011/01/colors.php

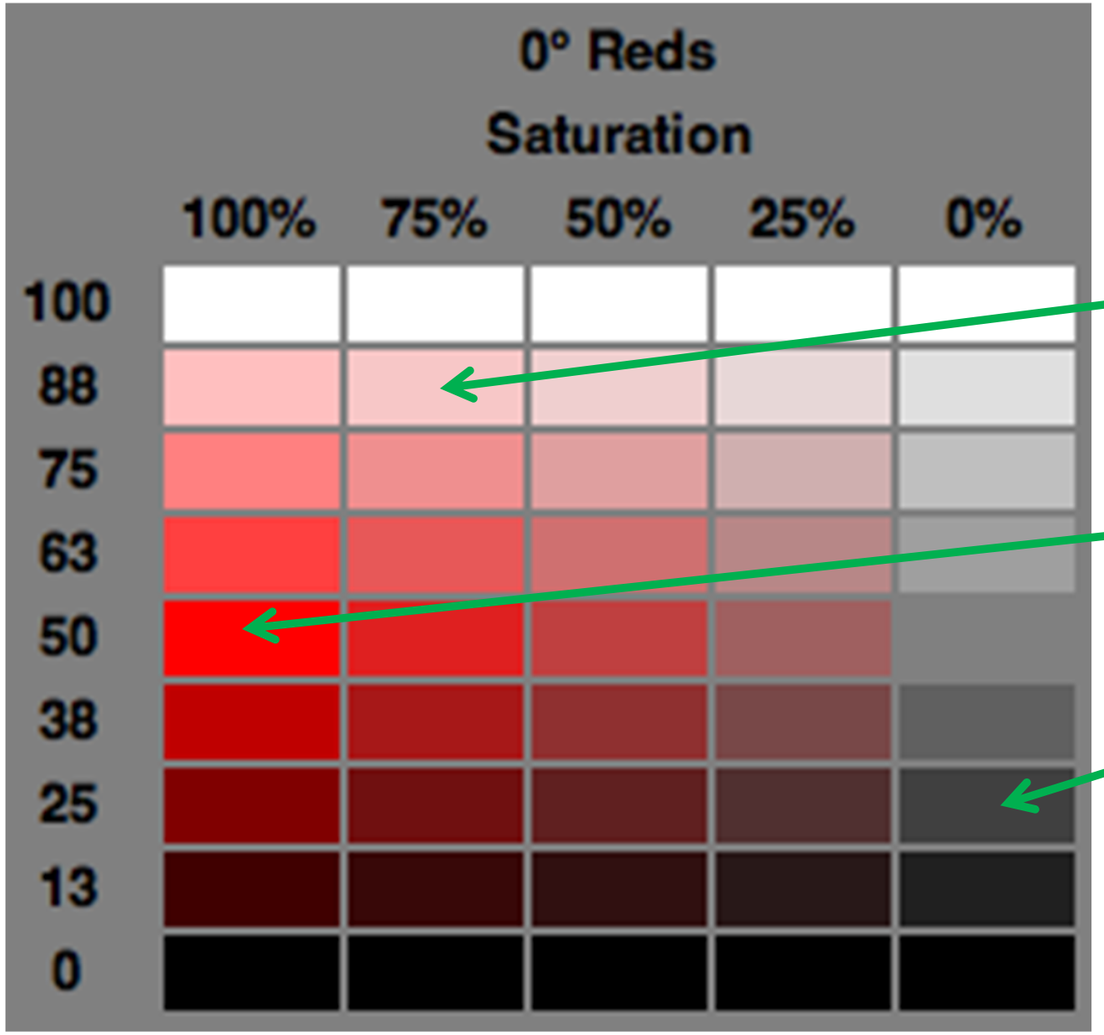
HSL Color Space: 3D Cylinder



HSL Color Space: 3D Cylinder (Sliced)



HSL Grid for Red (ie 0, S, L)



(0, 75%, 88%)

(0, 100%, 50%)

(0, 0%, 25%)

CSS Color Values

- Keywords: case-insensitive identifiers

```
red, navy, firebrick, chocolate
```

- RGB as decimal (0-255), percentage, or hex

```
rgb (255, 0, 0)  /* pure red */
rgb (100%, 0%, 0%)
#ff0000
#f00 /* expand by doubling each digit */
```

- HSL (Hue, Saturation, Light)
  - Hue (0-360) is angle on color wheel: 0 is red, 120 green, 240 blue
  - Saturation & light are both %'s

```
hsl (0, 100%, 50%) /* full bright red */
```

- Alpha channel (aka transparency): Note that 1 is opaque!

```
rgba (255, 0, 0, 0.5)
hsla (0, 100%, 50%, 1)
```

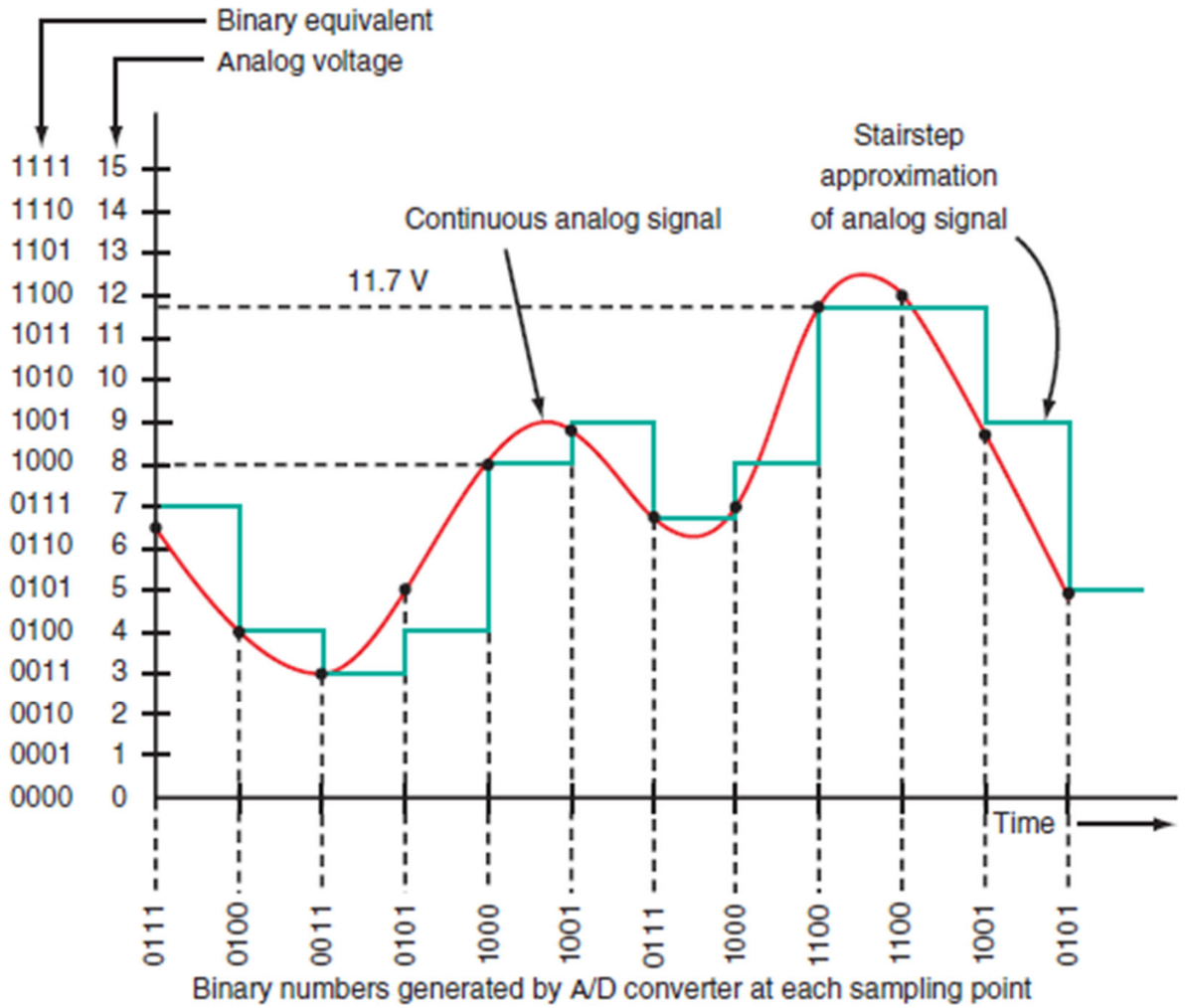
Color Keywords: 147 (141 unique colors)

aliceblue	antiquewhite	aqua	aquamarine	azure	beige
bisque	black	blanchedalmond	blue	blueviolet	brown
burlywood	cadetblue	chartreuse	chocolate	coral	cornflowerblue
cornsilk	crimson	cyan	darkblue	darkcyan	darkgoldenrod
darkgray	darkgreen	darkkhaki	darkmagenta	darkolivegreen	darkorange
darkorchid	darkred	darksalmon	darkseagreen	darkslateblue	darkslategray
darkturquoise	darkviolet	deeppink	deepskyblue	dimgray	dodgerblue
firebrick	floralwhite	forestgreen	fuchsia	gainsboro	ghostwhite
gold	goldenrod	gray	green	greenyellow	honeydew
hotpink	indianred	indigo	ivory	khaki	lavender
lavenderblush	lawngreen	lemonchiffon	lightblue	lightcoral	lightcyan
lightgoldenrodyellow	lightgray	lightgreen	lightpink	lightsalmon	lightseagreen
lightskyblue	lightslategray	lightsteelblue	lightyellow	lime	limegreen
linen	magenta	maroon	mediumaquamarine	mediumblue	mediumorchid
mediumpurple	mediumseagreen	mediumslateblue	mediumspringgreen	mediumturquoise	mediumvioletred
midnightblue	mintcream	mistyrose	moccasin	navajowhite	navy
oldlace	olive	olivedrab	orange	orangered	orchid
palegoldenrod	palegreen	paleturquoise	palevioletred	papayawhip	peachpuff
peru	pink	plum	powderblue	purple	rebeccapurple
red	rosybrown	royalblue	saddlebrown	salmon	sandybrown
seagreen	seashell	sienna	silver	skyblue	slateblue
slategray	snow	springgreen	steelblue	tan	teal
thistle	tomato	turquoise	violet	wheat	white
whitesmoke	yellow	yellowgreen			

Color Depth

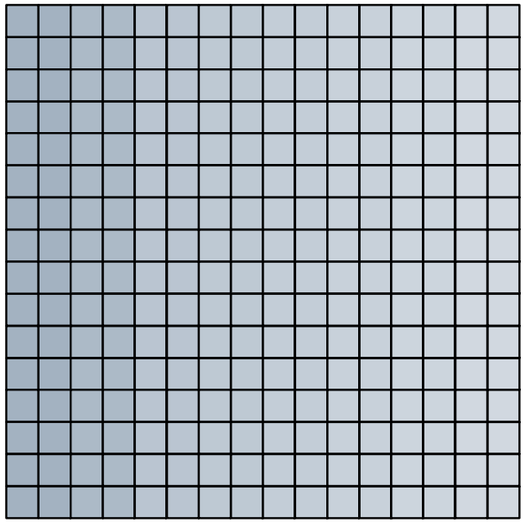
- "Depth" = # of bits in representation
  - 8 bits ➔ 256 different colors (eg GIF format)
  - 24 bits ➔ 16,777,216 different colors: 8 bits each for r,g,b
- Alpha sometimes (incorrectly) included
  - rgba is a point in 4-dimensional space
- Problem: image color depth > display color depth
  - Quantization: each pixel gets closest available color (leads to banding)
  - Dithering: add noise, which looks better!

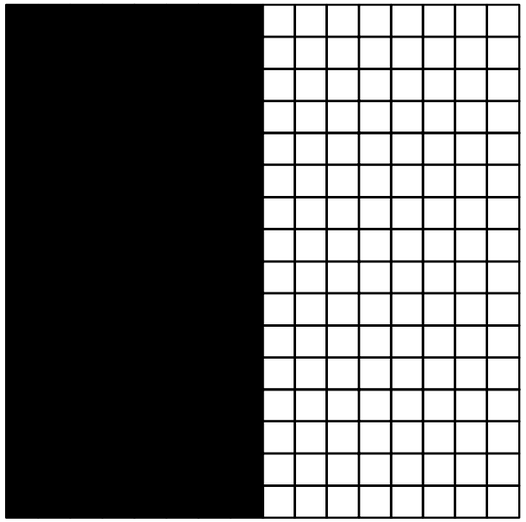
Quantization of Continuous Func



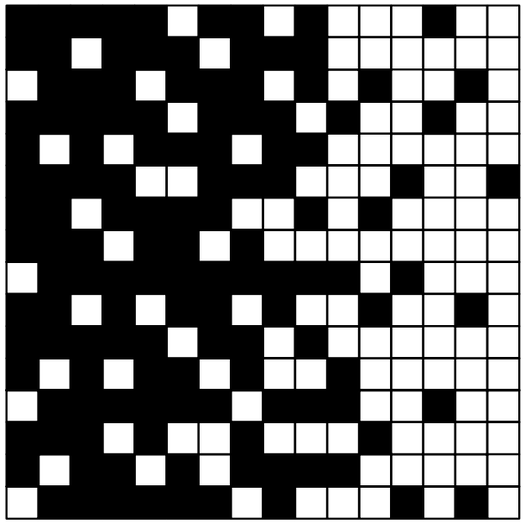
Quantization vs Dithering

original



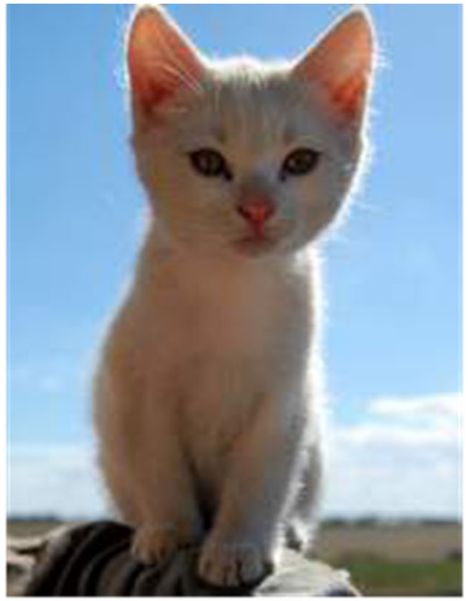


quantized



dithered

Quantization vs Dithering: Impact



Original Image

GIF without dithering

GIF with dithering

www.coffeecup.com/help/articles/dither-modes-in-animation-studio/

HTML `<img>` Tag Attributes

- **`src`**: location (URL) of image file
- **`width`**, **`height`**:
  - Area in *window* to reserve for image
  - Image is *scaled* to those dimensions
  - These attributes affect browser flow, regardless of when/if image is displayed
- **`alt`**: text to show if graphic can not be displayed or seen (ie alternative)
  - Necessary for accessibility
- **`title`**: text to *augment* displayed graphic (eg tooltip)
  - Optional, doesn't help accessibility

Image Representation

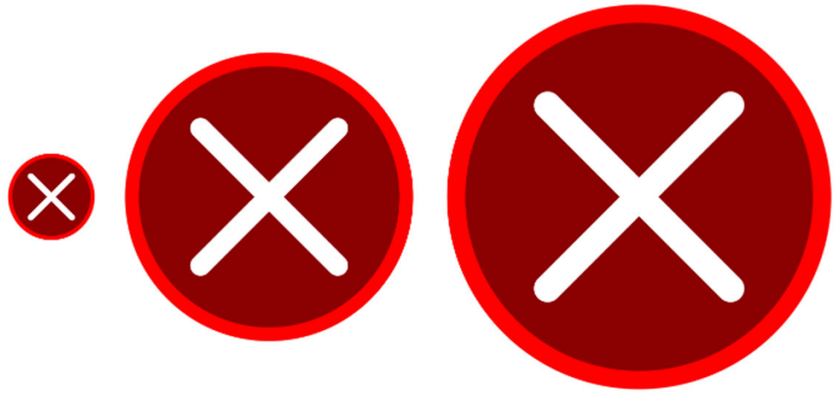
- Raster vs vector graphics
  - Raster: stored pixel-by-pixel
  - Vector: mathematical description
- Compression of raster images
  - Lossy: better compression, lower quality image
  - Lossless: largest file size, best quality

Major Formats

- GIF
  - Raster graphics, lossy compression (oldest)
  - 8 bit, basic transparency (on/off)
  - Frame-based animation (groan)
  - Good for small file size, crisp lines, logos
- JPEG
  - Raster, lossy compression
  - 24 bit, no transparency
  - Good for photos, gradual gradients
- PNG
  - Raster, lossless (but still often good) compression
  - Variable depth, full alpha transparency
  - Good replacement for GIF (but no animation)
- SVG
  - vector graphics
  - Good for crisp lines, simple logos, graphs

Scaling Images

- Vector graphics scale perfectly



- Raster images should be *pre-scaled*
  - Width (height) attributes of image tag should match actual width (height) of image
  - Why?
  - Cloud services can help (eg cloudinary.com)

Alternative: CSS


```
.button {
  display: inline-block;
  padding: 0.3em 1.2em;
  margin: 0 0.3em 0.3em 0;
  border-radius: 2em;
  box-sizing: border-box;
  text-decoration: none;
  font-weight: 300;
  color: #FFFFFF;
  background-color: #4eb5f1;
  text-align: center;
  transition: all 0.2s;
}
```

Summary: Color and Images

- Color theory
  - Perception, metamerism
  - Mixing: subtractive, additive
  - RGB, HSL, keywords
- Images
  - Quantization and dithering
  - Raster graphics vs vector graphics
  - Formats jpeg, png, gif, svg
  - Tradeoffs of size, quality, features