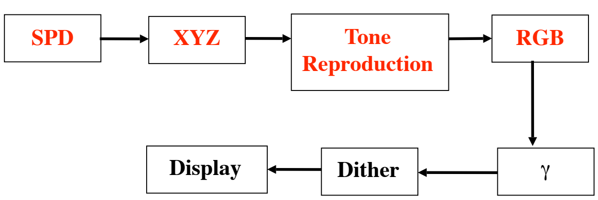
Fundametals of Rendering - Image Pipeline

Chapter 8 of “Physically Based Rendering” by Pharr&Humphreys

Chapter 8 – Film and the Imaging Pipeline

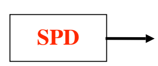
8.1-8.2	PBRT interface to film and image
8.3	Image pipeline – 2 paragraphs: read
8.4	Perceptual issues – we’ll cover this in class Except for 8.4.2 (Bloom): read this yourself
8.5	Final image pipeline stages - read

Image Pipeline

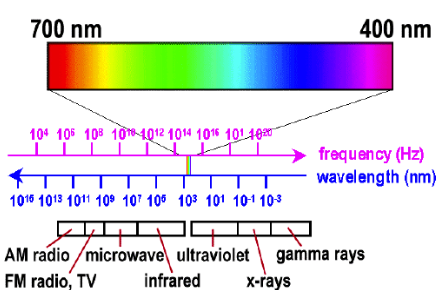


SPD - spectral power distribution
XYZ – Computed color from samples
Tone Reproduction – perceptual mapping
RGB – display color values
gamma correction – compensate for display non-linearities
Dithering – trade-off spatial resolution for color resolution
Display

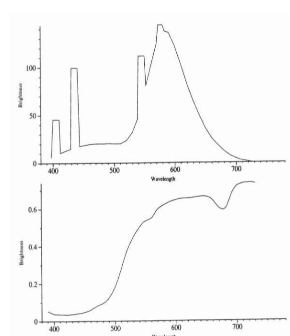
Image Pipeline



Visible Light

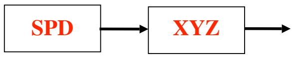


SPD



- Light not a single wavelength
- Combination of wavelengths
- A spectrum, or spectral power distribution (SPD).
- Tristimulus theory, metamers

Image Pipeline



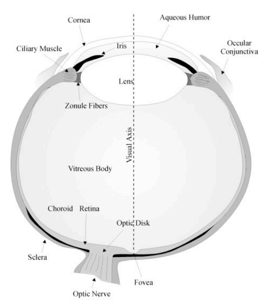
3-Component Color

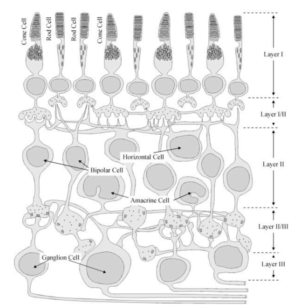
- The de facto representation of color on screen display is RGB. (additive color)
- Most printers use CMY(K), (subtractive color)
- Color spectrum can be represented by 3 basis functions
- Compute floating point values of color intensities from shading model

Perception: human eye and vision

- Eye is an amazing device !
  - Vision is even more so
- Yet, can trick it rather easily
- Need to understand what is important
- CG has to be tuned to perception
  - Already used three receptor fact – got RGB
  - Where does the eye stop and the brain begin?

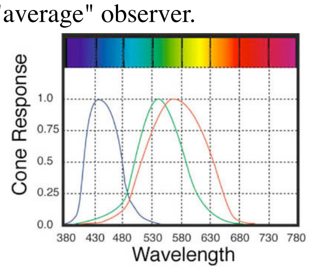
The eye and the retina





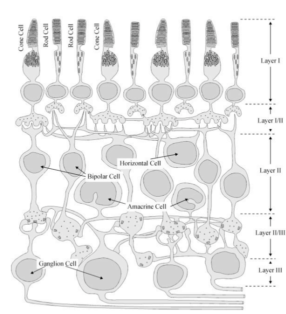
Retina detectors

- 3 types of color sensors - S, M, L (cones)
  - Works for bright light (photopic)
  - Peak sensitivities located at approx. 430nm, 560nm, and 610nm for "average" observer.
  - Roughly equivalent to blue, green, and red sensors



Retina detectors

- 1 type of monochrome sensor (rods)
  - Important at low light (scotopic)
- Next level: lots of specialized cells
  - Detect edges, corners, etc.
- Sensitive to contrast
  - Weber's law: $\frac{\Delta I}{I} = K$



Radiometry vs. Photometry

Luminance – how bright an SPD is to a human observer

Physics	Radiometry	Radiometric Units
	Radiant energy	joules $[J = kg\,m^2/s^2]$
Flux	Radiant power	watts $[W = joules/s]$
Angular flux density	Radiance	$[W/m^2\,sr]$
Flux density	Irradiance	$[W/m^2]$
Flux density	Radiosity	$[W/m^2]$
	Radiant intensity	$[W/sr]$

Physics	Photometry	Photometric Units
	Luminous energy	talbot
Flux	Luminous power	lumens $[talbots/second]$
Angular flux density	Luminance	Nit $[lumens/m^2\,sr]$
Flux density	Illuminance	Lux $[lumens/m^2\,sr]$
Flux density	Luminosity	Lux $[lumens/m^2\,sr]$
	Luminous intensity	Candela $[lumens/sr]$

Radiometry vs. Photometry

Each spectral quantity can be converted to its corresponding photometric quantity by integrating the product of its spectral distribution and the spectral response curve that describes the relative sensitivity of the human eye to various wavelengths. – under normally illuminated indoor environments

CIE XYZ color - all visible SPDs can be accurately represented for human observers with 3 values - computed by integrating with the 3 matching curves.

$$x_\lambda = \int_\lambda S(\lambda)X(\lambda)d\lambda$$

$$y_\lambda = \int_\lambda S(\lambda)Y(\lambda)d\lambda$$

$$z_\lambda = \int_\lambda S(\lambda)Z(\lambda)d\lambda$$

Luminance,Y, related to spectral radiance by spectral response

$$Y = \int_\lambda L(\lambda)V(\lambda)d\lambda$$

CIE Y curve proportional to V so that

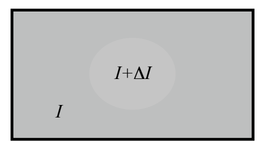
$$Y = 683 \int_\lambda L(\lambda)Y(\lambda)d\lambda$$

Human Vision

- What does the human observer really notice in the real world?
- How does the human vision change under different lighting conditions?
- What does the human observer notice in an image?
- What is the best way to represent an image on a digital display?

Just Noticeable Differences

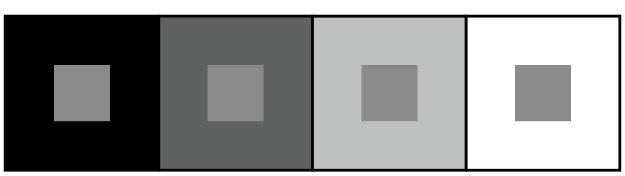
- Contrast: $\frac{\Delta I}{I}$



- For most intensities, contrast of .02 is just noticeable
- We're sensitive to contrasts, not intensity!

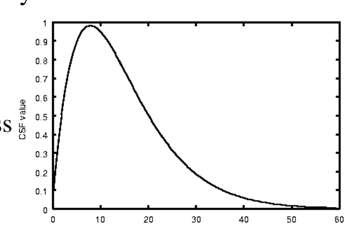
Contrast

- Inner gray boxes are the same intensity



Contrast sensitivity

- In reality, different sensitivity for different (spatial) frequencies
  - Max at ~8 cycles/degree



- Lose sensitivity in darkness
- More sensitive to achromatic changes
  - Try the same but red on green pattern
  - Practical consequence: color needs fewer bits
    - Used in video coding

Constancies

- Ability to extract the same information under different conditions
  - approximately the same info, in fact
- Size constancy: object at 10m vs. 100m
- Lightness constancy: dusk vs. noon
- Color constancy: tungsten vs. sunlight
- Not completely clear how this happens

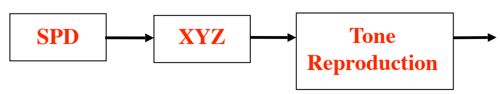
Adaptation

- Partially discard "average" signal
  - If everything is yellowish – ignore this
- Receptors "getting tired" of the same input
- Need some time to adapt when condition change
  - Stepping into sunlit outside from inside
- Model "adaptation" to look more realistic
  - Viewing conditions for monitors might be very different

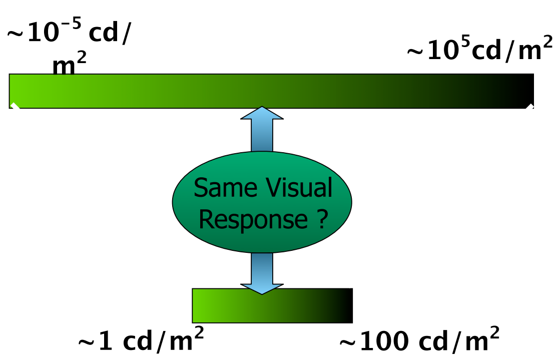
Tone mapping

- Real world range (physical light energy units)
- Monitors cover very small part of it
- Sensible conversion is needed
  - Tone mapping procedure
  - Book describes a few methods
- Often ignored in many applications
  - Might calibrate Light = (1,1,1), surface = (0.5, 0.5, 0.5)
  - No "right" basis for light
  - Works because of real-world adaptation process

Image Pipeline



Tone Reproduction



Ranges

Luminance (cd/m^2, or nits)	
600,000	Sun at horizon
120,000	60 Watt light bulb
8,000	Clear sky
100–1000	Typical office
1–100	Typical computer display
1–10	Street lighting
0.25	Cloudy moonlight

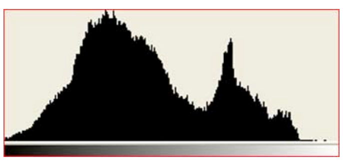
High Dynamic Range (HDR)

- The range of light in the real world spans 10 orders of magnitude!
- A single scene's luminance values may have as much as 4 orders of magnitude difference
- A typical CRT can only display 2 orders of magnitude
- Tone-mapping is the process of producing a good image of HDR data

Approaches

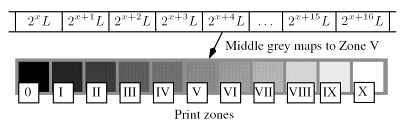
- Tone Reproduction or Mapping
- Mapping from image luminance range to display luminance range
- Use a scale factor to map pixel values
- Spatially uniform vs spatially varying?
  - Spatially uniform – monotonic, single factor
  - Non-uniform – scale varies
- Histogram

Histogram



Zone System

- Used by Ansel Adams. Utilizes measured luminance to produce a good final print
- Zone: an approximate luminance level. There are 11 print zones
- Middle-grey: Subjective middle brightness region of the scene, typically map to zone V
- Key: Subjective lightness or darkness of a scene

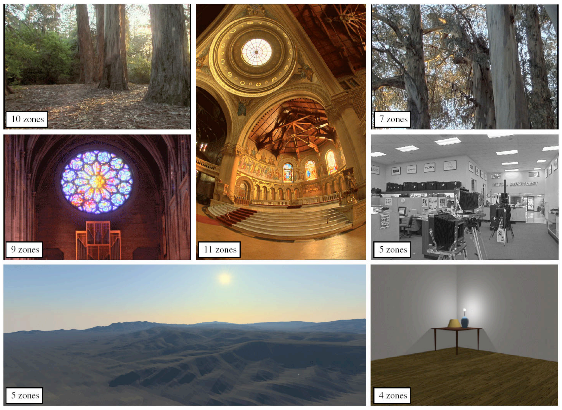


Zone System

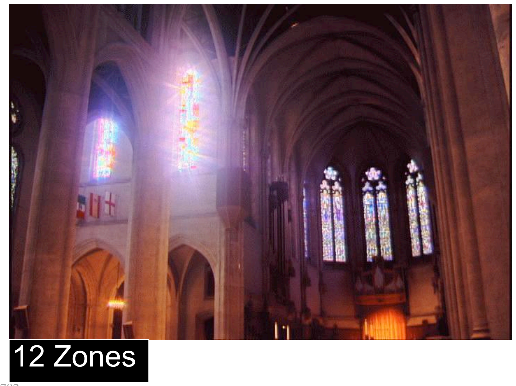
- Measure the luminance on a surface perceived as middle-gray - map to zone V
- Measure dynamic range from both light and dark areas.
- If dynamic range < 9 zones then full range can be captured in print
- Otherwise withhold or add light in development to lighten or darken the final print

Results

Results



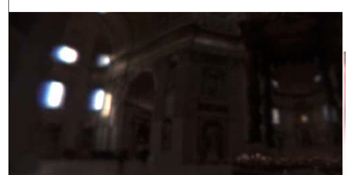
Running Example

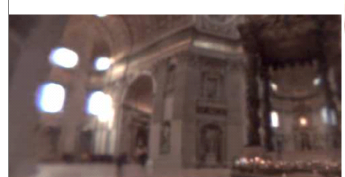


Maximum to White Operator

- Map brightest pixel to max luminance of display
- Problems for very well lit scenes
- Nothing about the visual system

Typical Tone Maps





Map Value to White & Scale
Fixed radiance value to map to the brightest displayable color.
3 values differ by factor of 10

Contrast Based

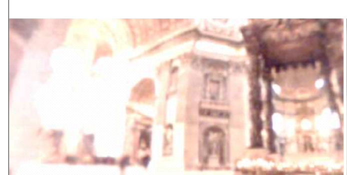
- JND – Just notice difference
- Objective: set luminances so that one JND in displayed image corresponds to one JND in actual environment

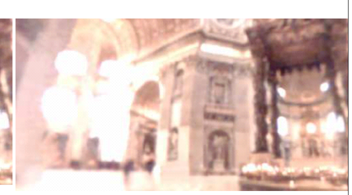
$\Delta Y(Y^a)$ is the JND for adaptation luminance Y^a

$$\Delta Y(Y^a) = 0.0594(1.219 + (Y^a)^{0.4})^{2.5}$$

$$\Delta Y(Y^a_d) = s\Delta Y(Y^a_w)$$

Contrast Based





Solve for s:

$$s = \left(\frac{1.219 + (Y^a_d)^{0.4}}{1.219 + (Y^a_w)^{0.4}}\right)^{2.5}$$

Apply constant scale factor s

World adaptation luminance – log average of all luminance values in image

Adaptation Luminance

- How to compute adaptation luminance?
  - Average
  - Log average
  - Spatially varying: uniform radius
  - Spatially varying: varying radius

Luminance Scaling

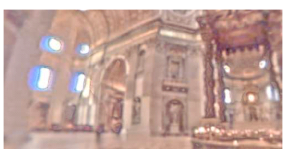
- Use log-average luminance to approximate the key of the scene

$$\bar{Y} = exp\left(\frac{1}{N}\sum_{x,y} log(\delta + Y_w(x,y))\right)$$

- Use log since small bright areas do not influence unduly

Varying Adaptive Luminance

- Compute local adaptation luminance that varies smoothly over the image.
- Need care at boundaries of light & dark
- Halo artifact if dark local adapt. luminance Includes bright pixels (mapped to black)



High Contrast Operator

Need to detect boundaries – only use neighboring dark pixels around dark pixels

TVI(): threshold versus intensity - gives just noticeable luminance difference for given adaptation level.

Number of JNDs in range: $$\frac{Y_a - Y_b}{TVI(Y^a)}$$

High Contrast Operator

Auxiliary capacity function: $$C(Y) = \int_0^Y \frac{dY'}{TVI(Y')}$$

Used to determine JNDs in a range $$C(Y_a) - C(Y_b)$$

Better definition that can be integrated easily

$$C(Y) = \begin{cases} Y/0.0014 & Y < 0.0034 \\ 2.4483 + log(L/0.0034)/0.4027 & 0.0034 \le Y < 1 \\ 16.563 + (Y-1)/0.4027 & 1 \le Y < 7.2444 \\ 32.0693 + log(Y/7.2444)/0.0556 & \text{otherwise} \end{cases}$$

Tone Mapping Operator

$$T(Y) = Y_d^{max} \frac{C(Y) - C(Y_{min})}{C(Y_{max}) - C(Y_{min})}$$

$$s(x,y) = \frac{T(Y^a(x,y))}{Y^a(x,y)}$$

High Contrast Method

Find minimum and maximum image luminance

Build luminance image pyramid

Apply high contrast tone mapping

Local Contrast

Consider area around pixel:

as large as possible

as small as necessary to exclude high contrast

use blurred versions of image

pixel (x,y) value in blurred image: $B_s(x,y)$

s: filter width

Local Contrast

$$lc(s,x,y) = \frac{B_s(x,y) - B_{2s}(x,y)}{B_s(x,y)}$$

find largest s such that: $|lc(s,x,y)| < c$

$$Y^a(x,y) = B_s(x,y)$$

Local Contrast

$$C(Y) = \begin{cases} Y/0.0014 & Y < 0.0034 \\ 2.4483 + log(L/0.0034)/0.4027 & 0.0034 \le Y < 1 \\ 16.563 + (Y-1)/0.4027 & 1 \le Y < 7.2444 \\ 32.0693 + log(Y/7.2444)/0.0556 & \text{otherwise} \end{cases}$$

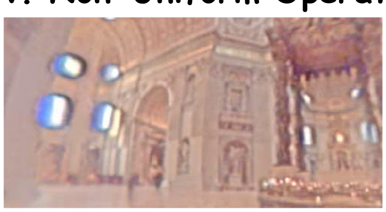
$$T(Y) = Y_d^{max} \frac{C(Y) - C(Y_{min})}{C(Y_{max}) - C(Y_{min})}$$

$$Y^a(x,y) = B_s(x,y)$$

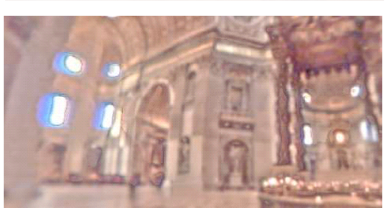
$$s(x,y) = \frac{T(Y^a(x,y))}{Y^a(x,y)}$$

Uniform v. Non-Uniform Operators

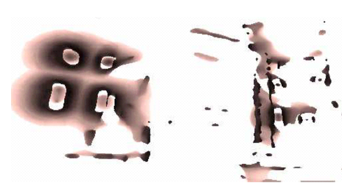
uniform



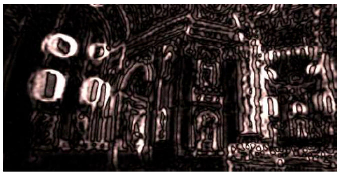
non-uniform



Neighborhood Sizes



Determining Neighborhoods

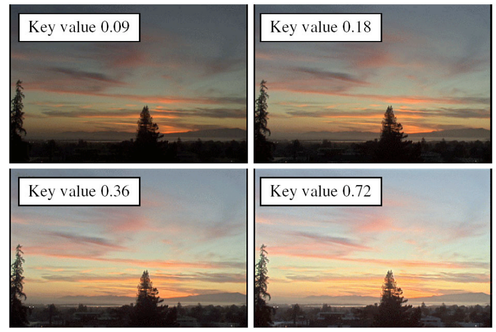


local contrast computed with blur radius of 1.5 and 3.0

Photographic Tone Reproduction for Digital Images

Erik Reinhard, Michael Stark, Peter Shirley, James Ferwerda
SIGGRAPH 2002

key of a scene: subjective value indicating scene lit normal, light (high key), or dark (low key)
used to map zone V of scene to key-percent-reflectivity of print

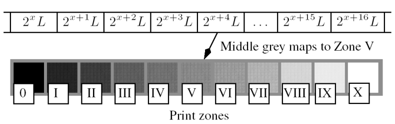


Dodging and Burning

Printing technique in which some light is added (burning) or withheld (dodging) from a portion of the print during development

Developed by Ansel Adams and his Zone System

In a normal-key image middle-gray maps to a key value a = .18



Luminance mapping

$$Y = \frac{aY_w(x,y)}{\bar{Y}}$$

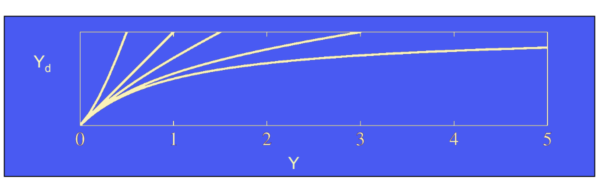
$$Y_d(x,y) = \frac{Y(x,y)}{1+Y(x,y)}$$

Control burn out of high luminance – global operator

$$Y_{display}(x,y) = \frac{Y(x,y)\left(1+\frac{Y(x,y)}{Y_{white}^2}\right)}{1+Y(x,y)}$$

Luminance mapping

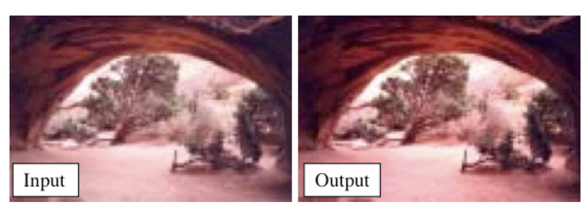
$$Y_{display}(x,y) = \frac{Y(x,y)\left(1+\frac{Y(x,y)}{Y_{white}^2}\right)}{1+Y(x,y)}$$



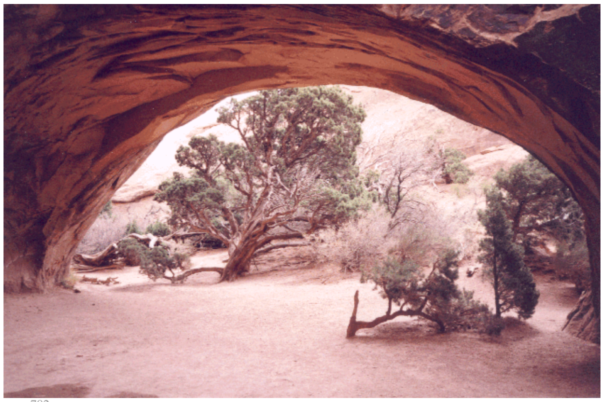
showing curve for various values of Y_{white}

Luminance mapping

Images from a pdf of the paper

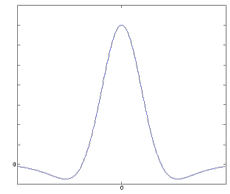


From Reinhard's web site



Local Adaptation

- Need a properly chosen neighborhood
- Dodging-and-burning is applied to regions bounded by large contrasts
- Use center-surround functions to measure local contrast at different scales
- E.g, use difference of Gaussians



Local Adaptation

at scale, s and for pixel (x,y)

$$R_i(x,y,s) = \frac{1}{\pi(\alpha_i s)^2} exp\left(-\frac{x^2+y^2}{(\alpha_i s)^2}\right)$$

convolve image with Gaussians to get response function

$$V_i(x,y,s) = L(x,y) \otimes R_i(x,y,s)$$

or multiply in the frequency domain

Local Adaptation

center-surround function

$$V(x,y,s) = \frac{V_1(x,y,s) - V_2(x,y,s)}{2^{\phi}a/s^2 + V_1(x,y,s)}$$

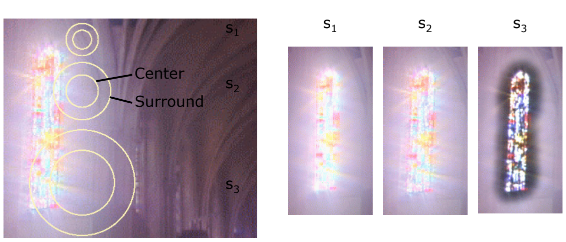
normalized by

$$2^{\phi}a/s^2 + V_1$$

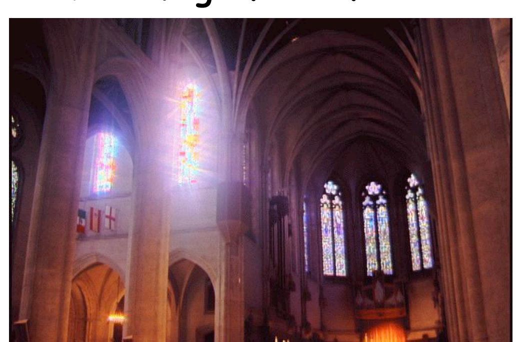
sharpening parameter (ϕ), key value (a)

Varying Scales

- The effects of using different scales



Full image for reference



12 Zones

Automatic Dodging-and-Burning

- Choose largest neighborhood around a pixel with fairly even luminance
- Take the largest scale that doesn't exceed a contrast threshold:

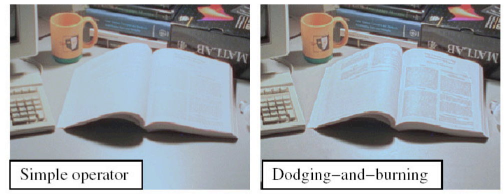
$$|V(x,y,s_m)| < \varepsilon$$

- Final local operator

$$Y_d(x,y) = \frac{Y(x,y)}{1 + V_l(x,y,s_m)}$$

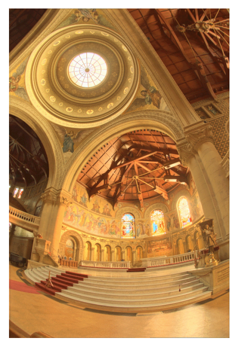
Automatic Dodging-and-Burning

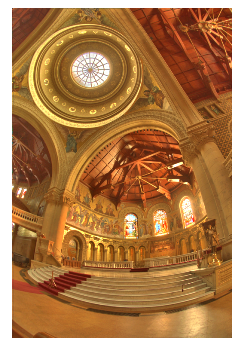
- Details recovered by using dodging-and-burning



Results

Comparison


Reinhard et al.


Durand et al.

Comparison


Durand et al.

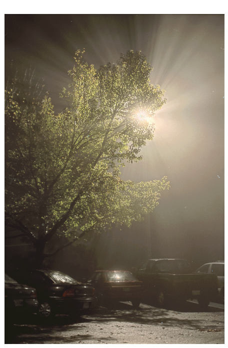
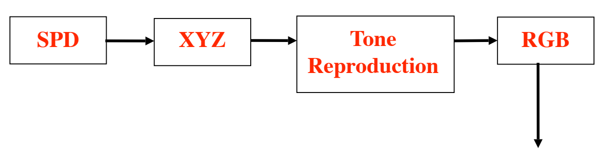

Reinhard et al.

Image Pipeline

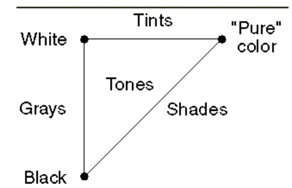


Color Systems

- Response: $R = \int w(\lambda) L(\lambda) d\lambda$
- Detector response is linear
  - Scaled input -> scaled response
  - response(L_1+L_2) = response(L_1)+response(L_2)
- Choose three basis lights L_1, L_2, L_3
  - Record responses to them
  - Can compute response to any linear combination
  - Tristimulus theory of light
- Most color systems are just different choice of basis lights
  - Could have "RBG" lights as a basis

Color Systems

- Our perception registers:
  - Hue
  - Saturation
  - Lightness or brightness
- Artists often specify colors in terms of
  - Tint
  - Shade
  - Tone



Tristimulus Response

- Given spectral power distribution S(λ)

$$X = \int x(\lambda) S(\lambda) d\lambda$$

$$Y = \int y(\lambda) S(\lambda) d\lambda$$

$$Z = \int z(\lambda) S(\lambda) d\lambda$$

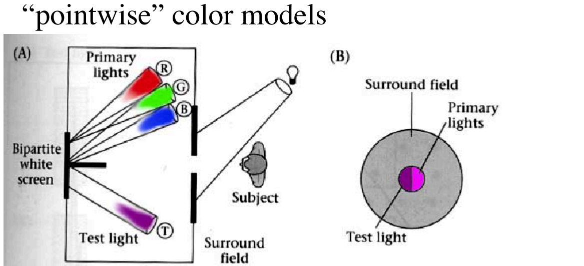
- Given $S_1(\lambda)$, $S_2(\lambda)$, if the X, Y, and Z responses are same then they are metamers wrt to the sensor
- Used to show that three sensor types are same

CIE Standard

- CIE: International Commission on Illumination (Comission Internationale de l'Eclairage).
- Human perception based standard (1931), established with color matching experiment
- Standard observer: a composite of a group of 15 to 20 people

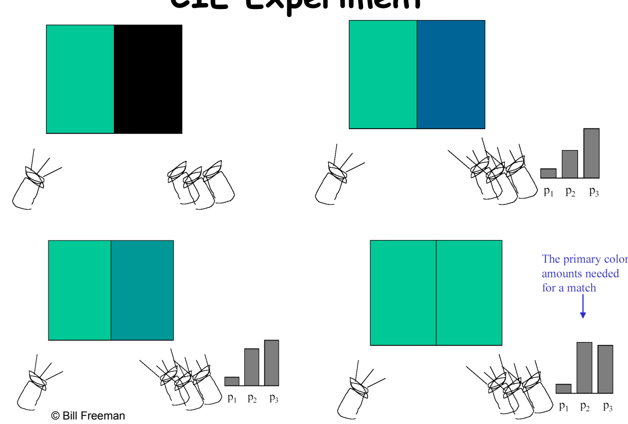
CIE Color Matching Experiment

- Basis for industrial color standards and "pointwise" color models



4.10 THE COLOR-MATCHING EXPERIMENT. The observer views a bipartite field and adjusts the intensities of the three primary lights to match the appearance of the test light. (A) A top view of the experimental apparatus. (B) The appearance of the stimuli to the observer. After Judd and Wyszecki, 1975.

CIE Experiment



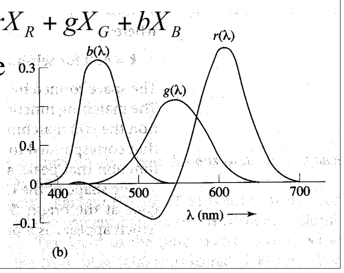
CIE Experiment Result

- Three pure light sources:
  R = 700 nm, G = 546 nm, B = 436 nm.

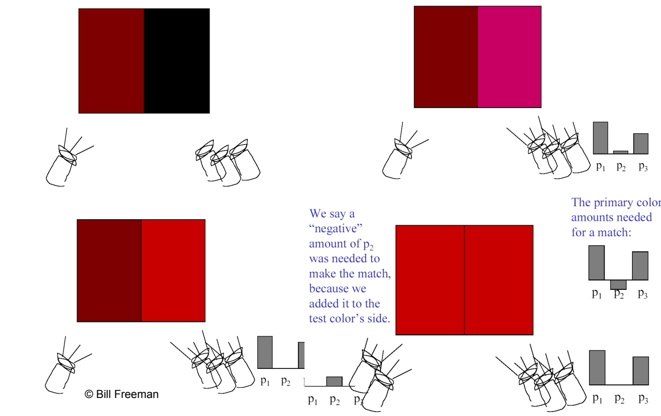
$$S(\lambda) = rR(\lambda) + gG(\lambda) + bB(\lambda)$$

$$X_S = \int x(\lambda)S(\lambda)d\lambda = rX_R + gX_G + bX_B$$

- r, g, b can be negative

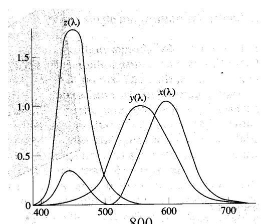


CIE Experiment



CIE Color Space

- 3 hypothetical light sources, X, Y, and Z, which yield positive matching curves
- Use linear combinations of real lights –R, G-2R,B+R
  - One of the lights is grey and has no hue
  - Two of the lights have zero luminance and provide hue
- Y: roughly corresponds to luminous efficiency characteristic of human eye



$$X_S = 683\int_{380}^{800} x(\lambda)S(\lambda)d\lambda$$

$$Y_S = 683\int_{380}^{800} y(\lambda)S(\lambda)d\lambda$$

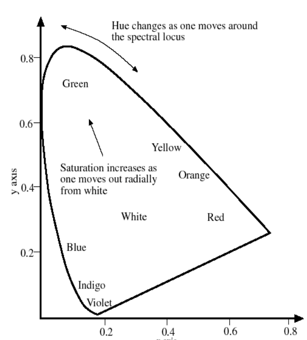
$$Z_S = 683\int_{380}^{800} z(\lambda)S(\lambda)d\lambda$$

CIE tristimulus values

- Particular way of choosing basis lights
  - Gives rise to a standard !!!
- Gives X, Y, Z color values
  - Y corresponds to achromatic (no color) channel
- Chromaticity values:
  - x=X/(X+Y+Z); y=Y/(X+Y+Z)
  - Typically use x,y,Y

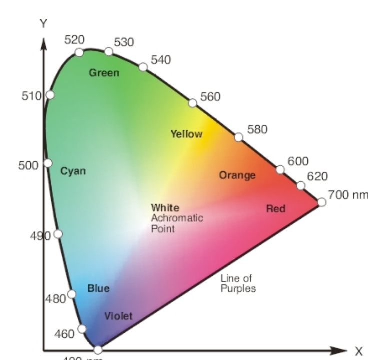
Chromaticity

- Normalize XYZ by dividing by luminance
- Project onto X+Y +Z=1
- Doesn't represent all visible colors, since luminous energy is not represented

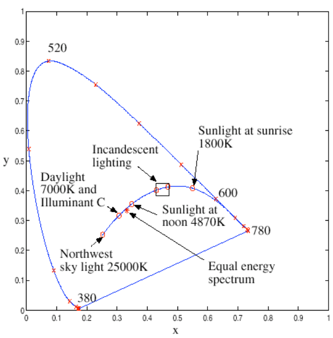


x, y: hue or chromatic part

Chromaticity

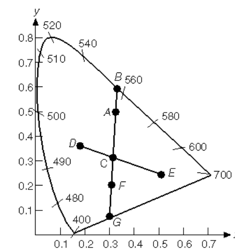


Chromaticity



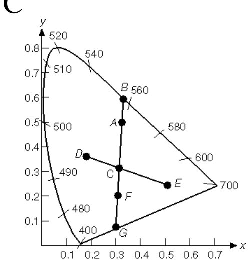
Chromaticity

- When 2 colors are added together, the new color lies along the straight line between the original colors
  - E.g. A is mixture of B (spectrally pure) and C (white light)
  - B - dominant wavelength
  - AC/BC (as a percentage) is excitation purity of A
  - The closer A is to C, the whiter and less pure it is.



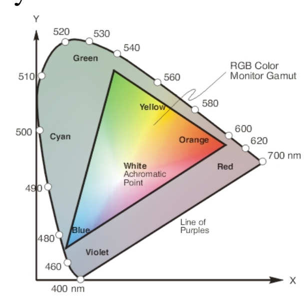
Chromaticity

- D and E are complementary colors
- can be mixed to produce white light
- color F is a mix of G and C
- F is non-spectral
  its dominant wavelength
  is the complement of B

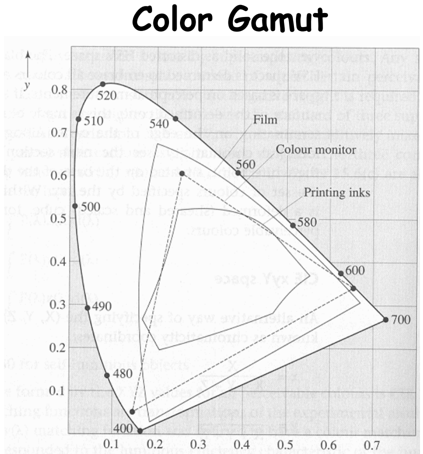


Color Gamut

- area of colors that a physical device can represent
- hence - some colors can't be represented on an RGB screen

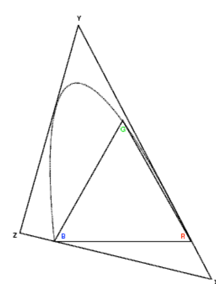


Color Gamut



Color Gamut

no triangle can lie within the horseshoe and cover the whole area

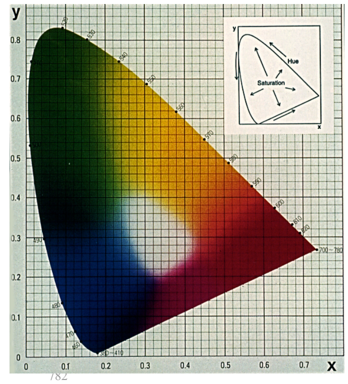


RGB <-> XYZ

- Just a change of basis
- Need detailed monitor information to do this right
  - Used in high quality settings (movie industry, lighting design, publishing)
- Normalized (lazy) way:
  - (1,1,1) in RGB <-> (1,1,1) in XYZ
  - matrices exist

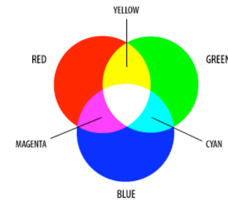
$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 0.5149 & .3244 & .1607 \\ 0.2654 & .6704 & .0642 \\ 0.0248 & .1248 & .8504 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Chromaticity Diagram



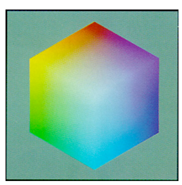
$$x = \frac{X}{X+Y+Z}$$

$$y = \frac{Y}{X+Y+Z}$$



The RGB Cube

- RGB color space is perceptually non-linear
- Dealing with > 1.0 and < 0 !
- RGB space is a subset of the colors human can perceive
- Con: what is 'bloody red' in RGB?



Other color spaces

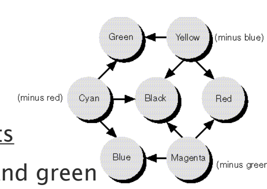
- CMY(K) – used in printing
- LMS – sensor response
- HSV – popular for artists
- Lab, UVW, YUV, YCrCb, Luv,
- Opponent color space – relates to brain input:
  - R+G+B(achromatic); R+G-B(yellow-blue); R-G(red-green)
- All can be converted to/from each other
  - There are whole reference books on the subject

Differences in Color Spaces

- What is the use? For display, editing, computation, compression, …?
- Several key (very often conflicting) features may be sought after:
  - Additive (RGB) or subtractive (CMYK)
  - Separation of luminance and chromaticity
  - Equal distance between colors are equally perceivable (Lab)

CMY(K): printing

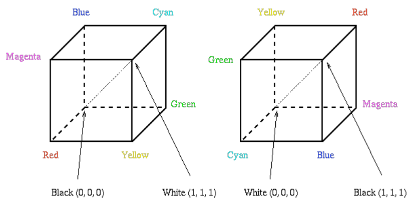
- Cyan, Magenta, Yellow (Black) – CMY(K)
- A subtractive color model



dye color	absorbs	reflects
Cyan	red	blue and green
Magenta	green	blue and red
yellow	blue	red and green
Black	all	none

RGB and CMY

- Converting between RGB and CMY



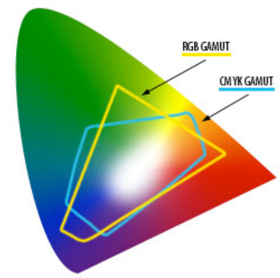
The RGB Cube

The CMY Cube

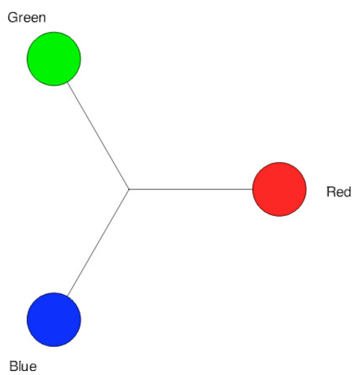
$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

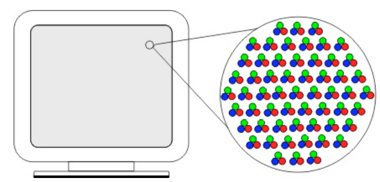
$$\begin{bmatrix} C \\ M \\ Y \\ K \end{bmatrix} = \begin{bmatrix} \max(R,G,B) \\ \max(R,G,B) \\ \max(R,G,B) \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \\ \max(R,G,B) \end{bmatrix}$$

RGB and CMY

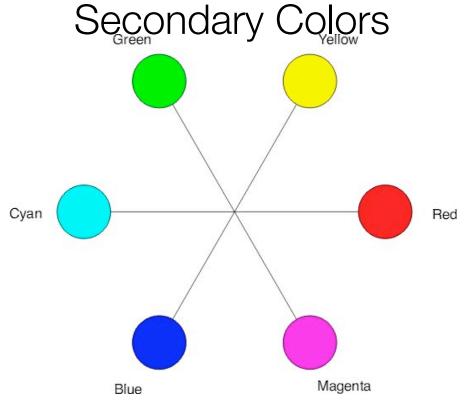


Primary Colors

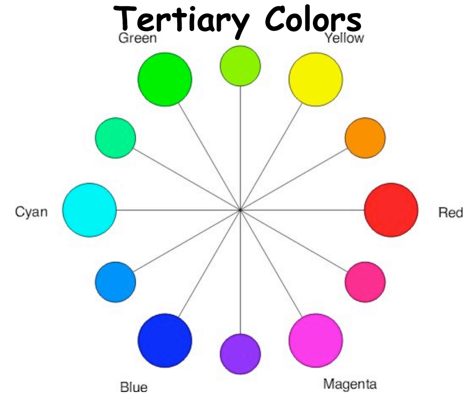




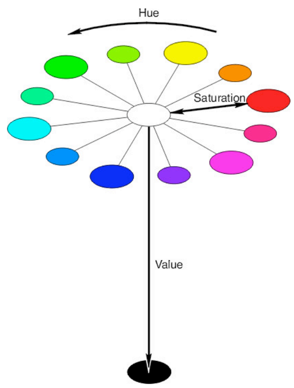
Secondary Colors



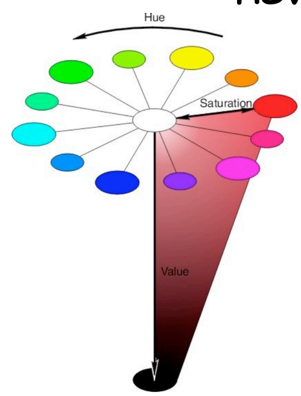
Tertiary Colors



HSV



HSV

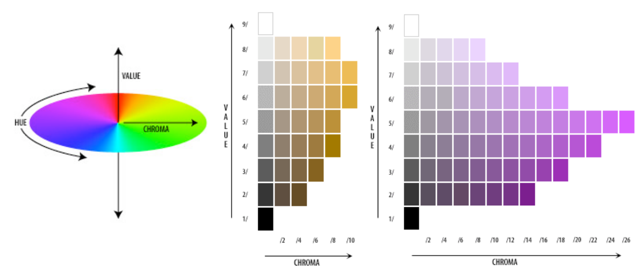


HSV

- This color model is based on polar coordinates, not Cartesian coordinates.
- HSV is a non-linearly transformed (skewed) version of RGB cube
  - Hue: quantity that distinguishes color family, say red from yellow, green from blue
  - Saturation (Chroma): color intensity (strong to weak). Intensity of distinctive hue, or degree of color sensation from that of white or grey
  - Value (luminance): light color or dark color

HSV Hexcone

- Intuitive interface to color



Luv and UVW

- A color model for which, a unit change in luminance and chrominance are uniformly perceptible
- U = 13 W* (u - uo); V = 13 W* (v - vo); W = 25 (100 Y) 1/3 - 17
  - where Y , u and v can be calculated from :
  - $X = 0.607\ Rn + 0.174\ Gn + 0.200Bn$
  - $Y = 0.299\ Rn + 0.587\ Gn + 0.114Bn$
  - $Z = 0.066\ Gn + 1.116\ Bn$
  - $x = X / (X + Y + Z)$
  - $y = Y / (X + Y + Z)$
  - $z = Z / (X + Y + Z)$
  - $u = 4x / (-2x + 12y + 3)$
  - $v = 6y / (-2x + 12y + 3)$

Luv and UVW

- Chrominance is defined as the difference between a color and a reference white at the same luminance.
- Luv is derived from UVW and Lab, with all components guaranteed to be positive

Yuv and YCrCb: digital video

- Initially, for PAL analog video, it is now also used in CCIR 601 standard for digital video
- Y (luminance) is the CIE Y primary.
  Y = 0.299R + 0.587G + 0.114B
- It can be represented by U and V -- the color differences.
  U = B – Y; V = R - Y
- YCrCb is a scaled and shifted version of YUV and used in JPEG and MPEG (all components are positive)

Cb = (B - Y) / 1.772 + 0.5; Cr = (R - Y) / 1.402 + 0.5

Examples (RGB, HSV, Luv)

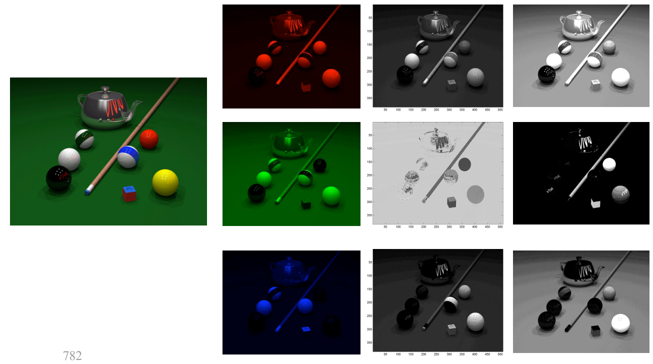
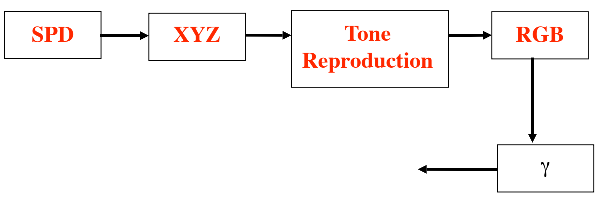


Image Pipeline



Color Matching on Monitors

- Use CIE XYZ space as the standard

$$\begin{bmatrix} A' \\ B' \\ C' \end{bmatrix} = \begin{bmatrix} X_R X_G X_B \\ Y_R Y_G Y_B \\ Z_R Z_G Z_B \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix}$$

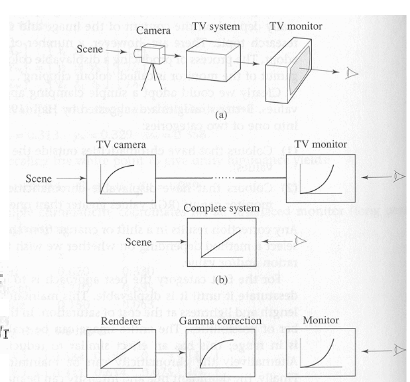
- Use a simple linear $C_2 = M_2^{-1} M_1 C_1$
- Color matching on printer is more difficult, approximation is needed (CMYK)

Gamma Correction

- The phosphor dots are not a linear system (voltage vs. intensity)

$$R_m = K(R'_i)^{\gamma_r}$$

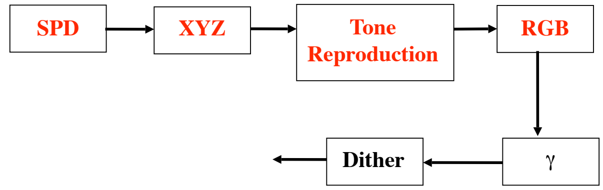
$$R'_i = k(R_i)^{1/\gamma_r}$$



No gamma correction

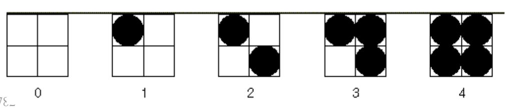
Gamma corrected to 1.7

Image Pipeline



Half-toning

- If we cannot display enough intensities? reduce spatial resolution and increase intensity resolution by allowing our eyes to perform spatial integration
- example is halftoning
  - approximate 5 intensity levels with the following 2x2 patterns.



Dithering

- maintain the same spatial resolution
- diffuse the error between the ideal intensity and the closest available intensity to neighbouring pixels below and to the right
- try different scan orders to "better" diffuse the errors
- e.g. Floyed-Steinberg:

		$\frac{7}{16}$
$\frac{3}{16}$	$\frac{5}{16}$	$\frac{1}{16}$

Image Pipeline