Lecture 11

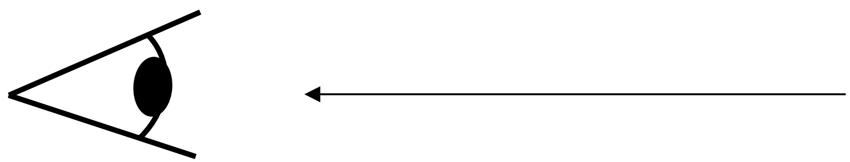
Color

Don't worry Sir, being colour-blind
is not much of a problem around here...

Starting Point: What is light?

Electromagnetic radiation (EMR) moving along rays in space

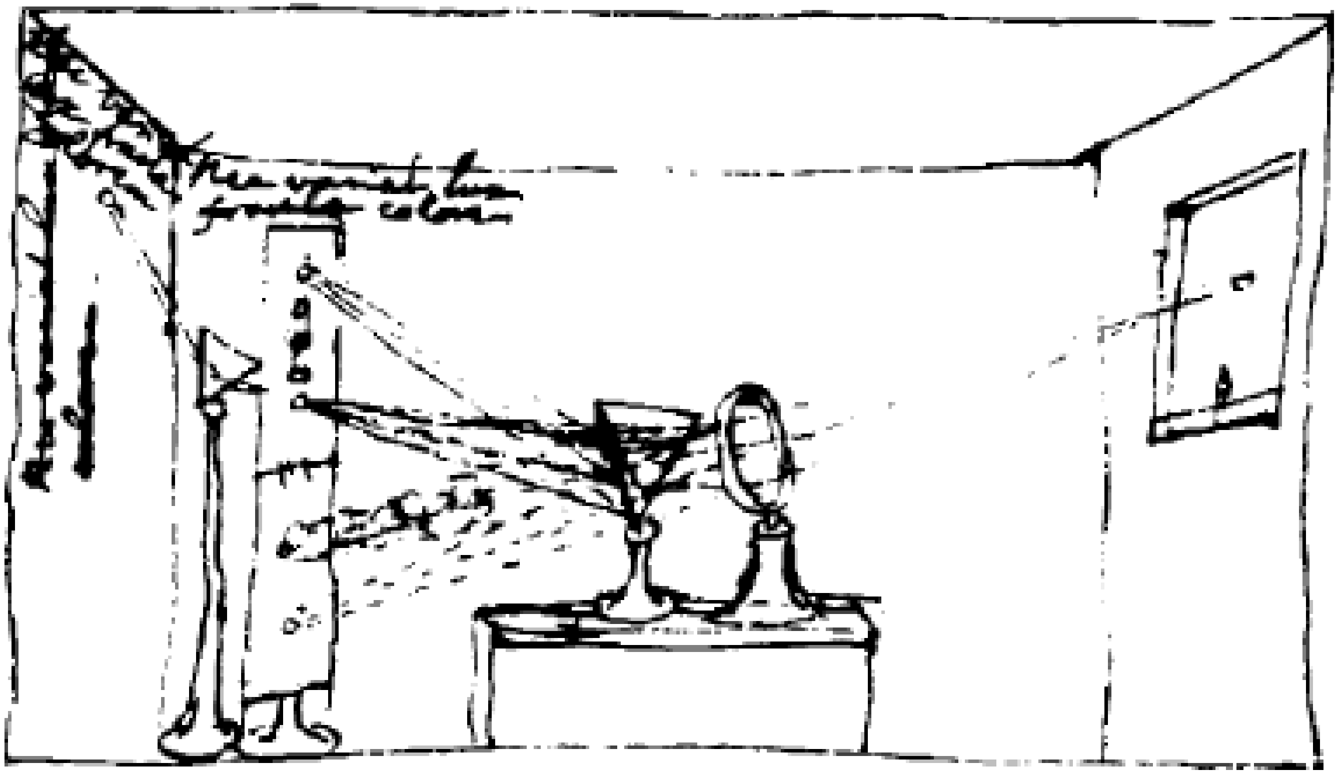
- $R(\lambda)$ is EMR, measured in units of power (watts)
  - λ is wavelength



Perceiving light

- How do we convert radiation into “color”?
- What part of the spectrum do we see?

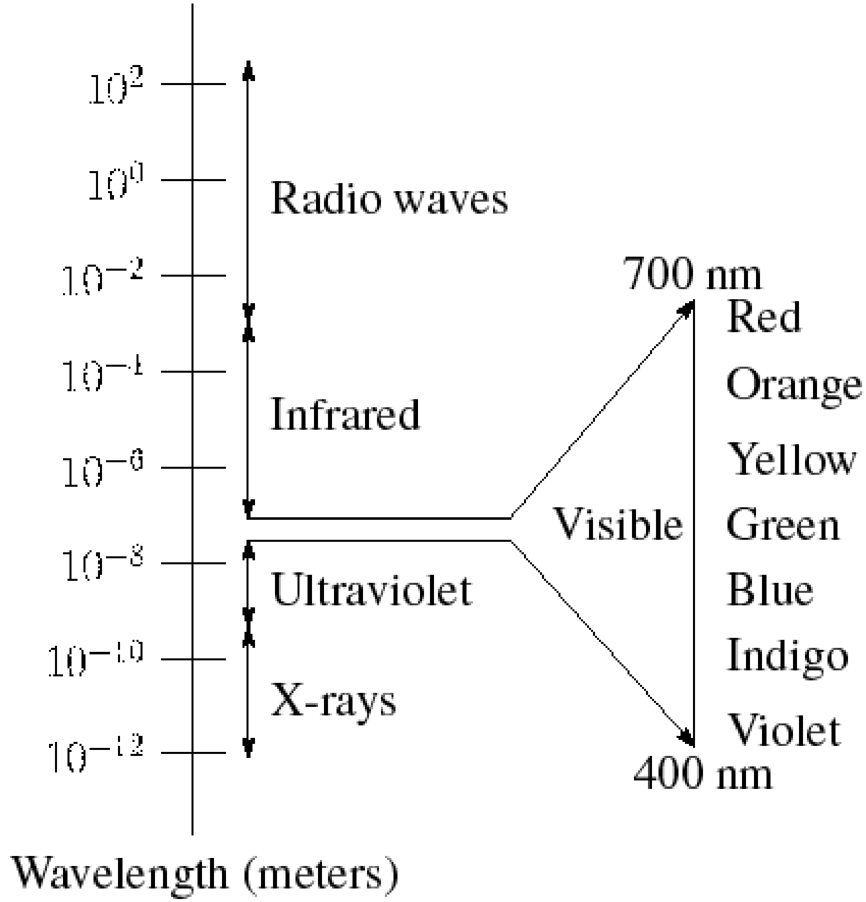
Newton’s prism experiment



Newton’s own drawing of his experiment showing decomposition of white light

The light spectrum

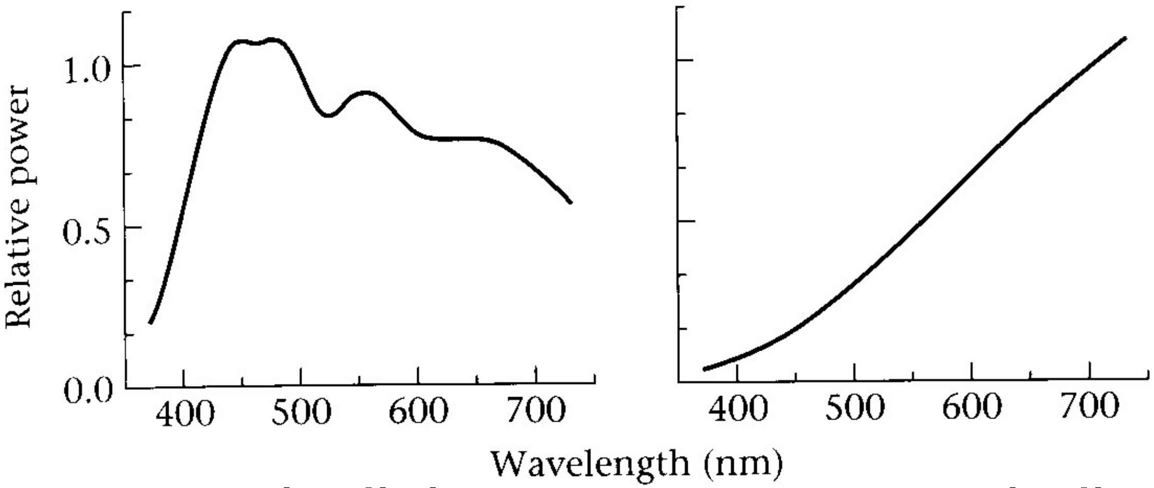
We “see” electromagnetic radiation in a range of wavelengths



Light spectrum

The appearance of light depends on its power **spectrum**

- How much power (or energy) at each wavelength

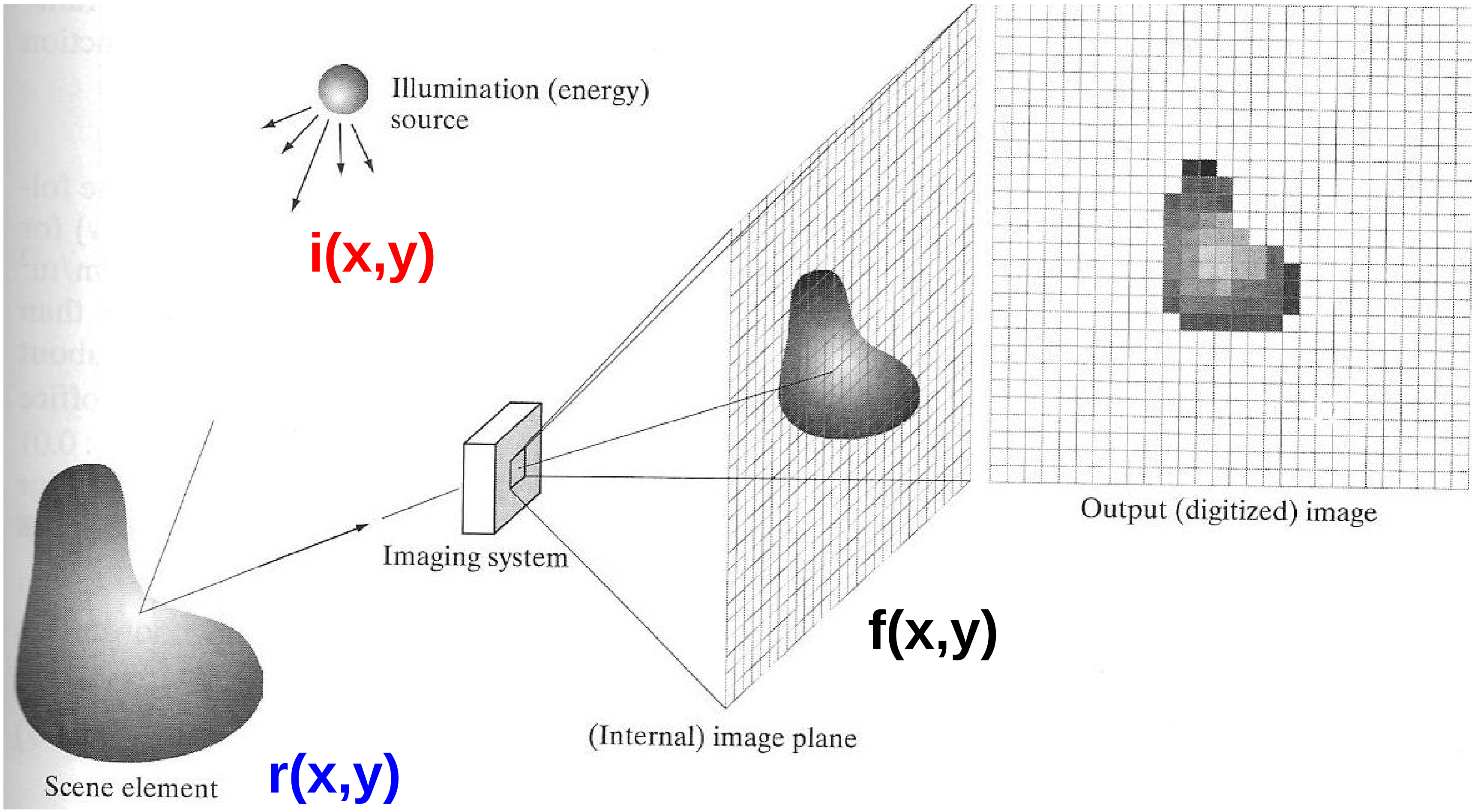


daylight tungsten bulb

Our visual system converts a light spectrum into “color”

- This is a rather complex transformation

Recall: Image Formation Basics



(from Gonzalez & Woods, 2008)

Image Formation: Basics

Image f(x,y) is characterized by 2 components

1. **Illumination i(x,y)** = Amount of source illumination incident on scene
2. **Reflectance r(x,y)** = Amount of illumination reflected by objects in the scene

$$f(x,y) = i(x,y)r(x,y)$$

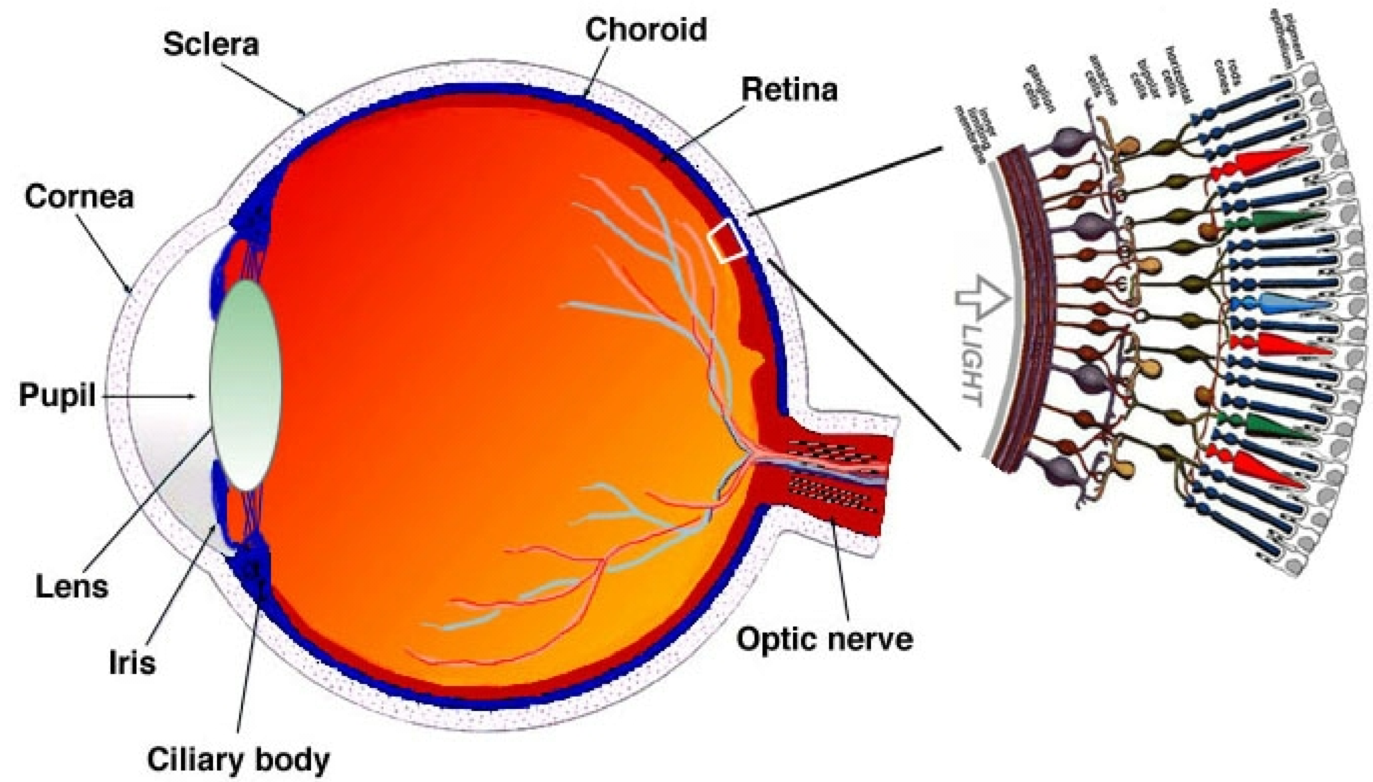
where

$$0 < i(x,y) < \infty \text{ and } 0 < r(x,y) < 1$$

r(x,y) depends on object properties

r = 0 means total absorption and 1 means total reflectance

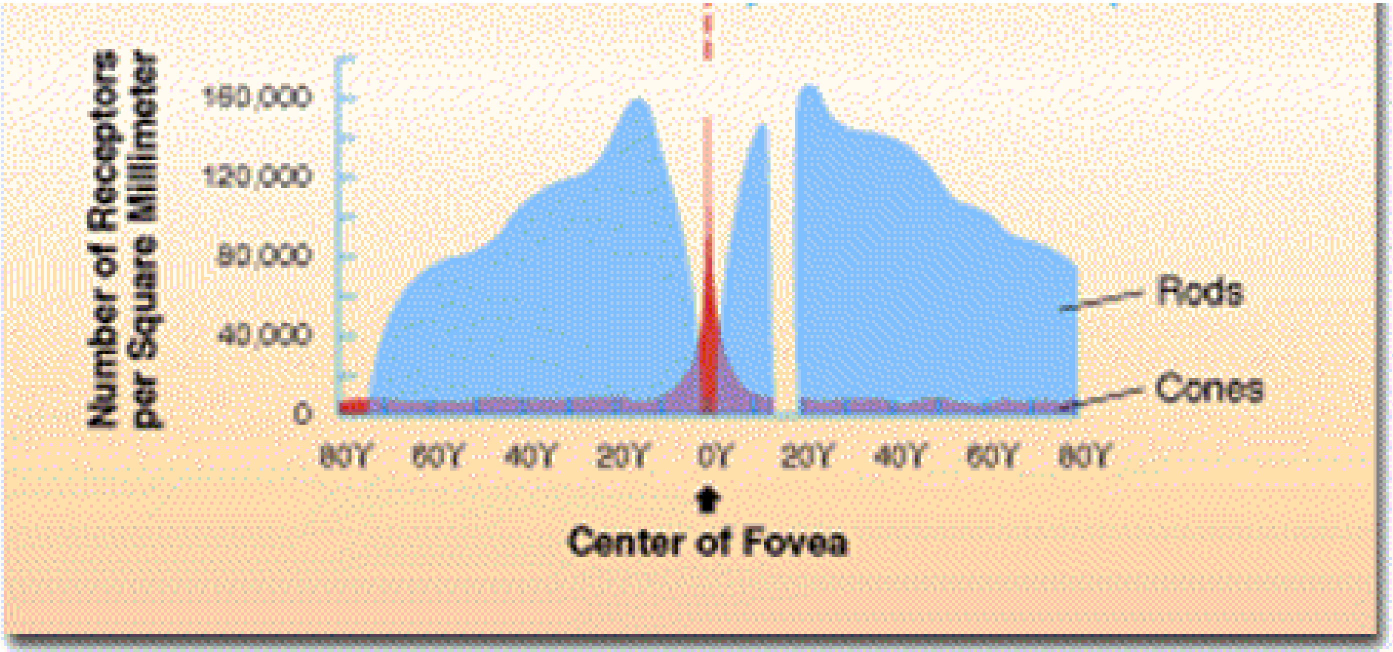
The Human Eye and Retina



Color perception

- Light hits the retina, which contains photosensitive cells
  - rods and cones
- Rods responsible for intensity, cones responsible for color

Density of rods and cones



Rods and cones are *non-uniformly* distributed on the retina

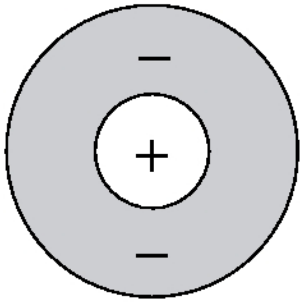
- **Fovea** - Small central region (1 or 2°) containing the highest density of cones (and no rods)
- Less visual acuity in the periphery—many rods wired to the same neuron

Demonstration of Blind Spot

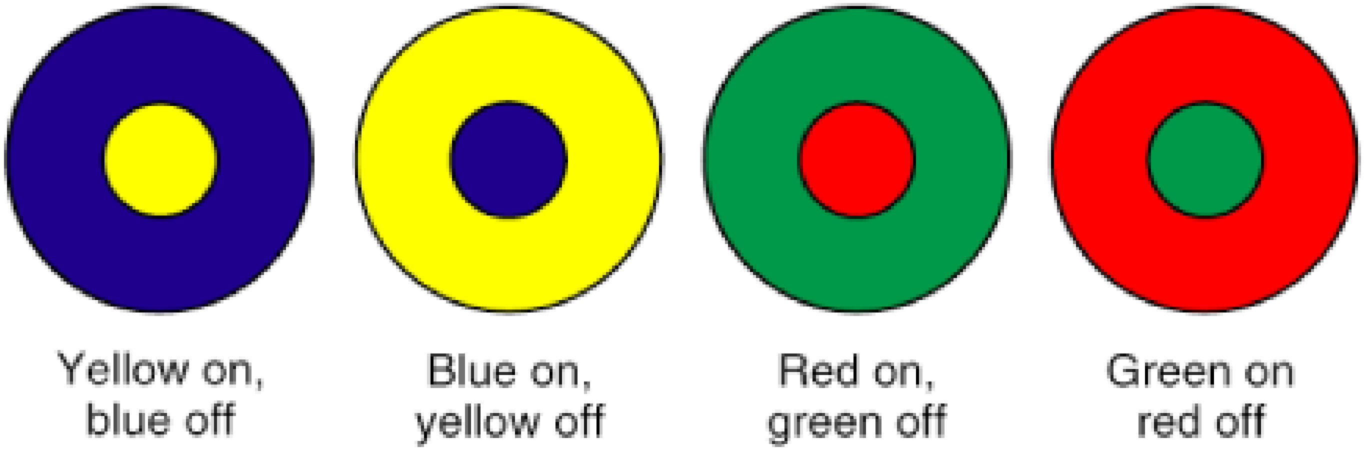
With left eye shut, look at the cross on the left. At the right distance, the circle on the right should disappear (Glassner, 1.8).

Retinal cells compare center with surrounding pixels

On-center Off-surround cell



"Color-opponent" processing



Brightness Contrast

Is the leftmost central gray square lighter or darker than the rightmost central gray square?

Brightness Contrast and Color Constancy

All gray squares actually have same color

Brain perceives color based on surrounding context

Such a mechanism useful for color constancy
- An apple appears red despite varying lighting conditions

Light response is nonlinear

Our visual system has a large *dynamic range*

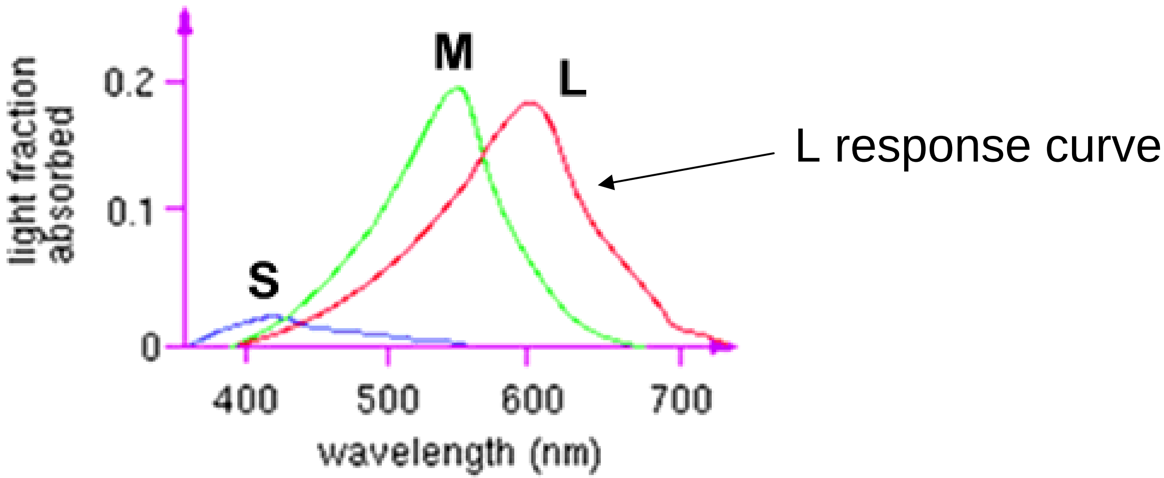
- We can resolve both light and dark things at the same time
- One mechanism for achieving this is that we sense light intensity on a *logarithmic scale*
  - an exponential intensity ramp will be seen as a linear ramp
- Another mechanism is *adaptation*
  - rods and cones adapt to be more sensitive in low light, less sensitive in bright light.

Visual dynamic range

Background	Luminance (candelas per square meter)
Horizon sky	
Moonless overcast night	0.00003
Moonless clear night	0.0003
Moonlit overcast night	0.003
Moonlit clear night	0.03
Deep twilight	0.3
Twilight	3
Very dark day	30
Overcast day	300
Clear day	3,000
Day with sunlit clouds	30,000
Daylight fog	
Dull	300–1,000
Typical	1,000–3,000
Bright	3,000–16,000
Ground	
Overcast day	30–100
Sunny day	300
Snow in full sunlight	16,000

FIGURE 1.13
Luminance of everyday backgrounds. *Source:* Data from Rea, ed., *Lighting Handbook 1984 Reference and Application*, fig. 3-44, p. 3-24.

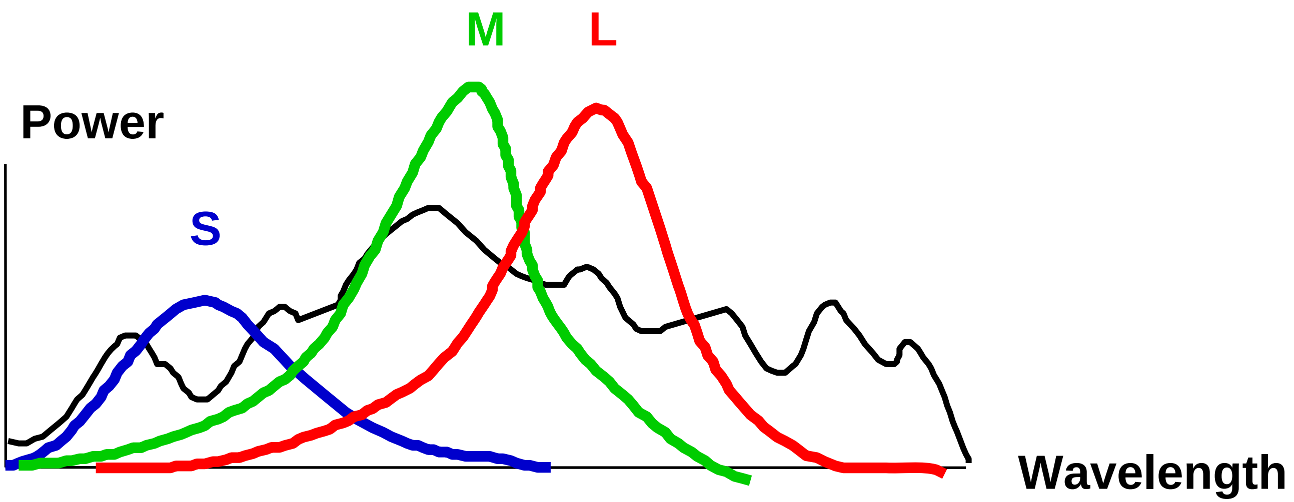
Color perception



Three types of cones

- Each is sensitive to a different region of the spectrum but regions overlap
  - Short (S) corresponds to blue
  - Medium (M) corresponds to green
  - Long (L) corresponds to red
- Different sensitivities: we are more sensitive to green than red
  - varies from person to person (and with age)
- Colorblindness—deficiency in at least one type of cone

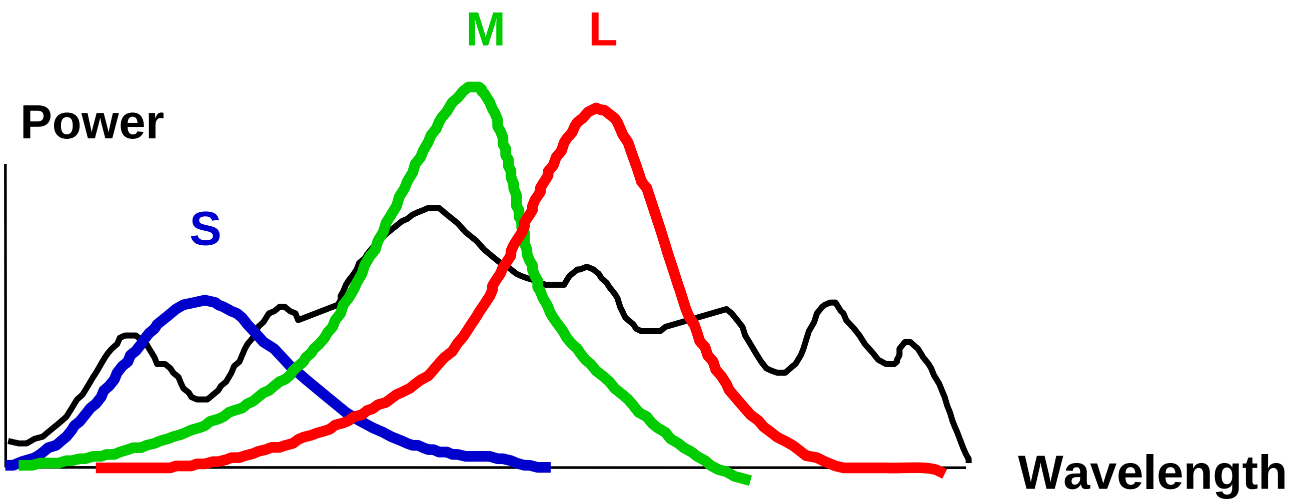
Color perception



Rods and cones act as filters on the spectrum

- To get the output of a filter, multiply its response curve by the spectrum, integrate over all wavelengths
  - Each cone yields one number

Color perception



Q: How can we represent an entire spectrum with 3 numbers?

A: We can't! Most of the information is lost.

– As a result, two different spectra may appear indistinguishable

» such spectra are known as **metamers**

» Online demo

Color coding methods

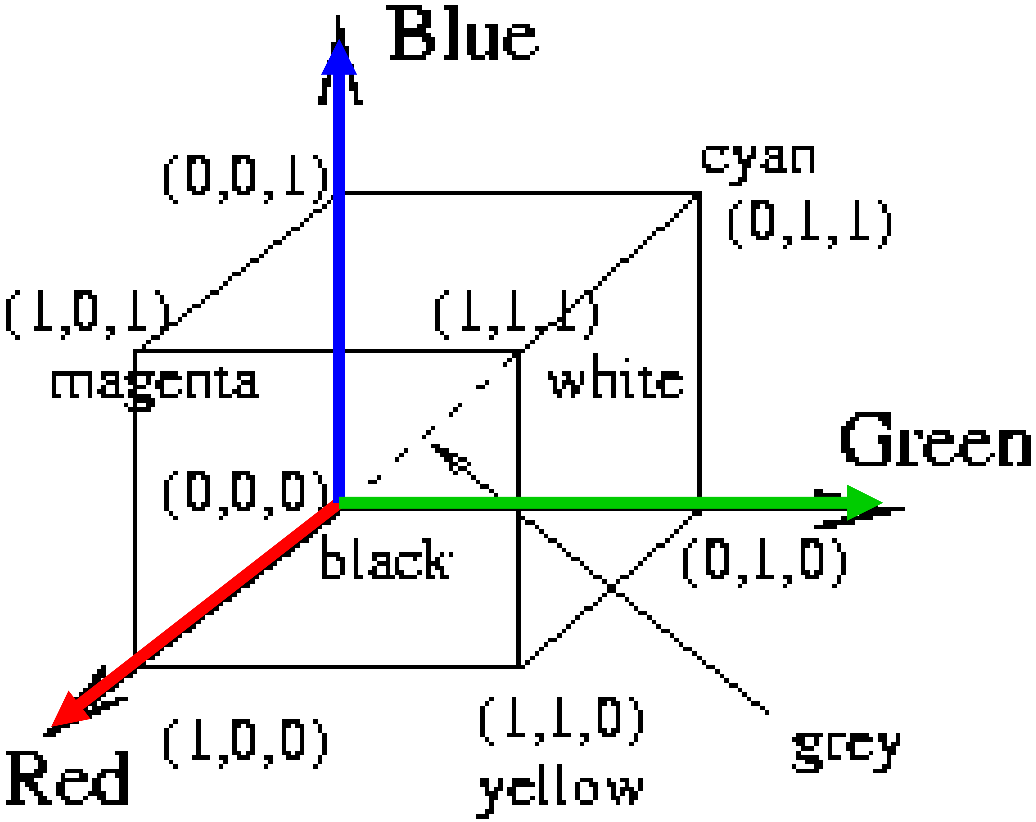
- **RGB** is an additive system (add colors to black) used for displays (CRTs, LCDs).

- **CMY** is a subtractive system for printing.

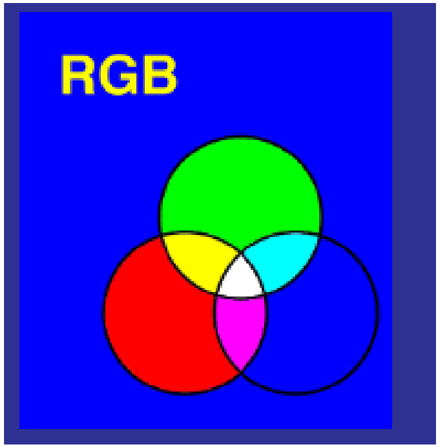
- **HSI** is a good perceptual space for art, psychology, and recognition.

- **YIQ** used for TV is good for compression.

RGB color cube (additive color model)

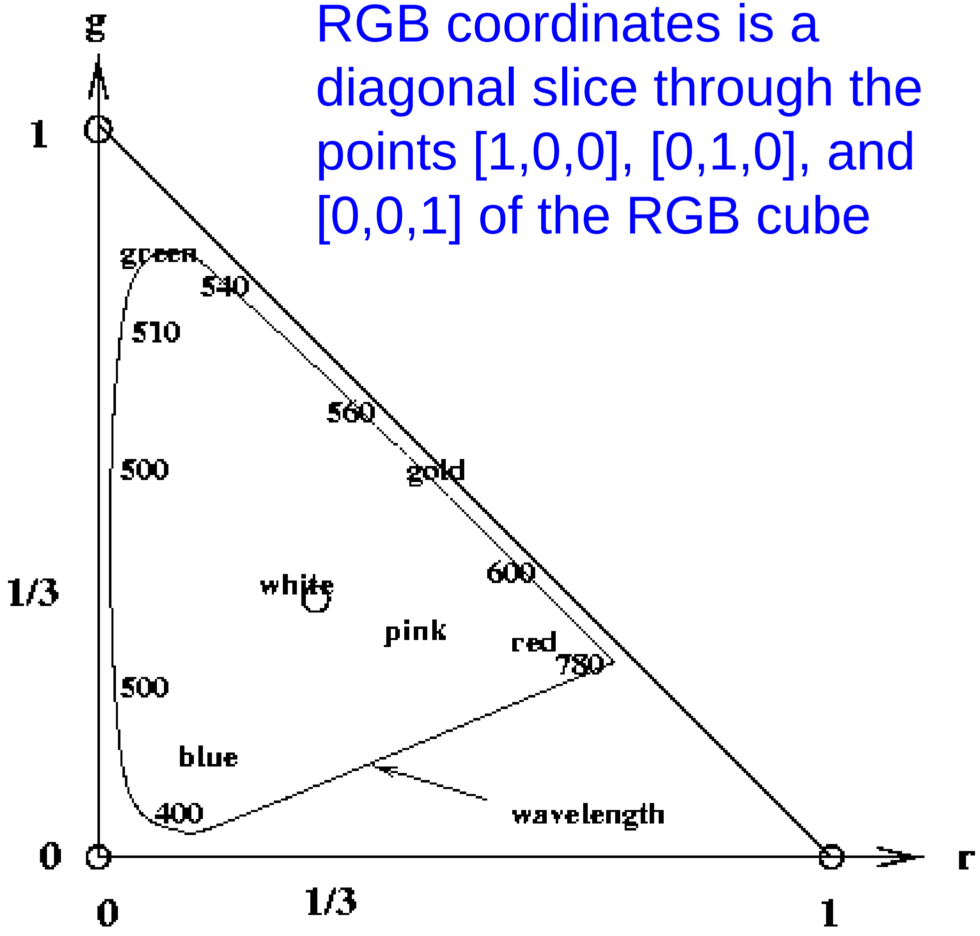


- R, G, B values normalized to (0, 1) interval
- Human perceives gray for triples along the diagonal; origin=black
- Additive: Mix RGB to get colors



Color triangle (CIE system) and normalized RGB

Color triangle for normalized RGB coordinates is a diagonal slice through the points [1,0,0], [0,1,0], and [0,0,1] of the RGB cube

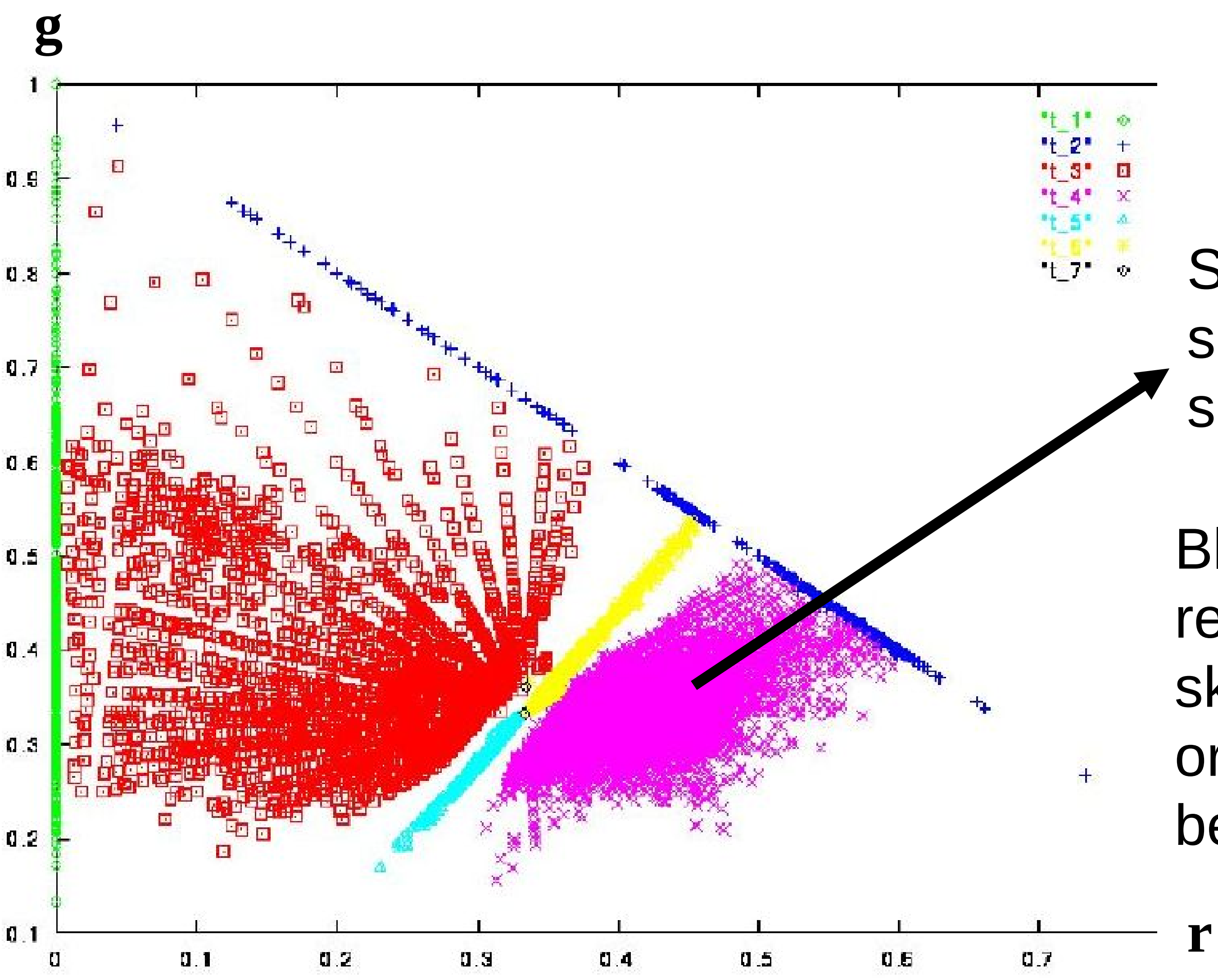


Intensity	**I = (R+G+B) / 3**
Normalized red	**r = R/(R+G+B)**
Normalized green	**g = G/(R+G+B)**
Normalized blue	**b = B/(R+G+B)**

In this normalized representation, **b = 1 – r –g**, so r and g values automatically imply b value

Note: To represent the full gamut of colors (e.g., black), you need to include brightness and therefore you are back in a 3D space (like the RGB cube)

Skin color in normalized RGB space

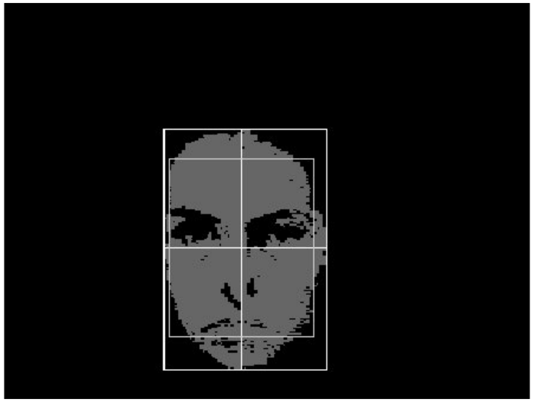


Skin color samples from several people

Blue and yellow regions show skin in shadow or behind a beard

Finding a face in video frame





(left) input video frame

(center) pixels classified according to RGB space

(right) largest connected component with aspect similar to a face

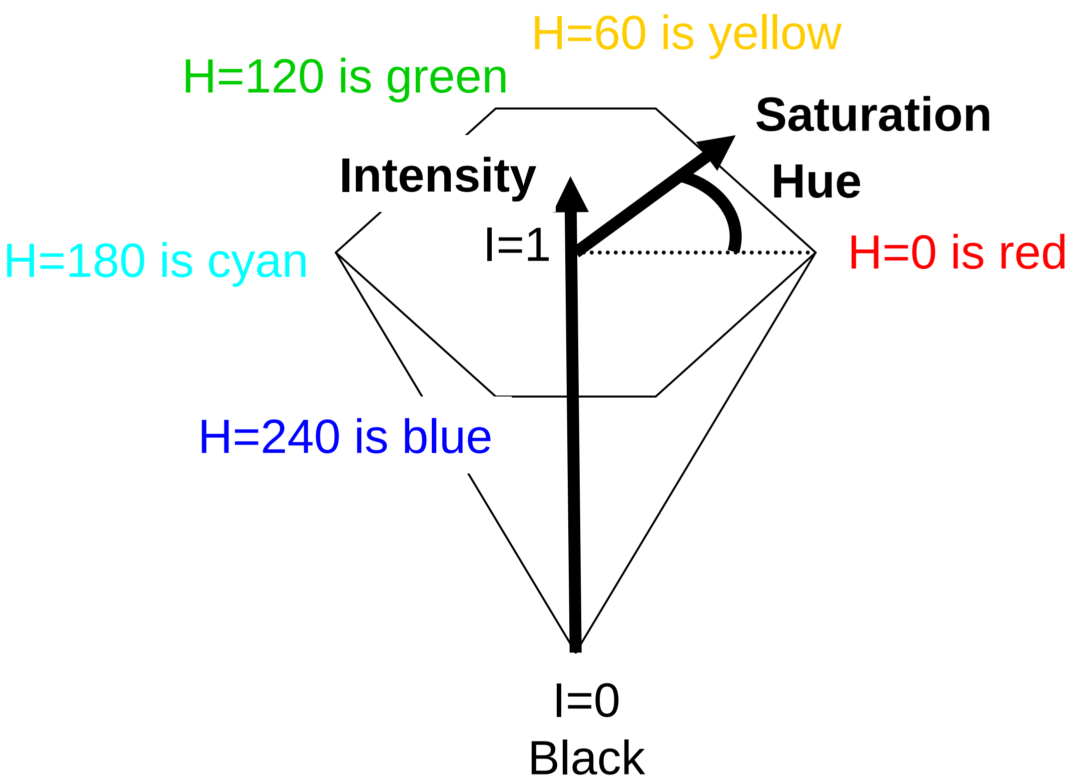
(work by Vera Bakic)

HSI (or HSV) Model (Color hexagon)

Hue: Distinguishes between colors (angle between 0 and 2π).

Saturation: Purity of color (distance on vertical axis (0 to 1)).

Intensity: Light versus dark shades of a color (height along the vertical axis (0 to 1)



Saturation example

(Left) Finger food image from a digital camera;

(Center) Saturation value of each pixel decreased 20%;

(Right) Saturation value of each pixel increased 40%.

YIQ and YUV for TV signals

Encodes luminance Y separately from chrominance I, Q (see next slide)

Has better compression properties: Luminance Y encoded using more bits than chrominance values I and Q; humans more sensitive to Y than I,Q

Luminance used by black/white TVs

All 3 values used by color TVs

YUV encoding used in some digital video and JPEG and MPEG compression

Conversion from RGB to YIQ

An approximate linear transformation from RGB to YIQ:

$$\begin{aligned} luminance \;\; Y &= 0.30R + 0.59G + 0.11B \\ R - cyan \;\; I &= 0.60R - 0.28G - 0.32B \\ magenta - green \;\; Q &= 0.21R - 0.52G + 0.31B \end{aligned}$$

We often use this for color to gray-tone conversion

CIE

CIE stands for Commission Internationale de l'Eclairage

- This commission determines standards for color and lighting. It developed the Normalized Color system (color triangle) or (X,Y,Z) system and the Lab Color System (also called the CIELAB Color System).

CIELAB

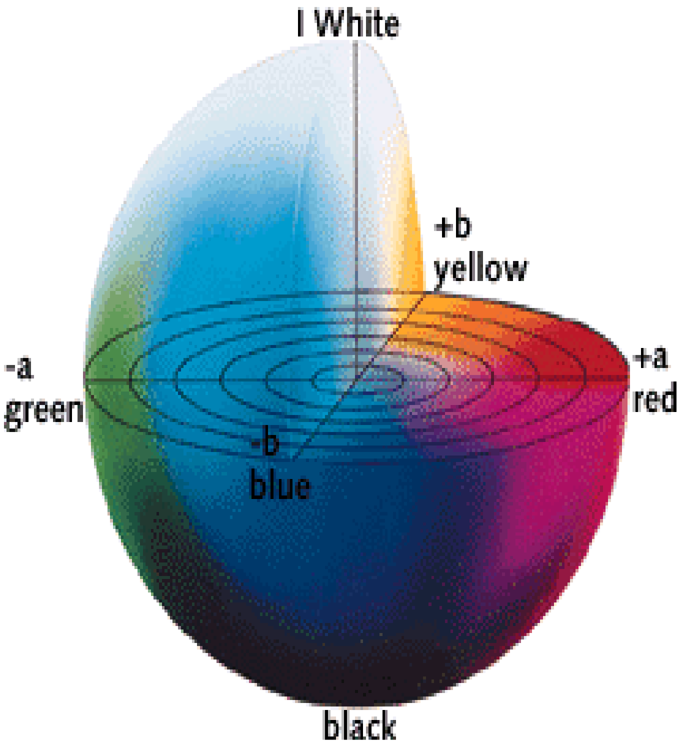
Designed to approximate human vision

One luminance channel (L) and two color-opponent channels (a and b).

In this model, the color differences which you perceive correspond to Euclidian distances in CIELab.

The a axis extends from green (-a) to re (+a) and the b axis from blue (-b) to yellow (+b). The brightness (L) increases from the bottom to the top of the three-dimensional model.

More info for the curious [here]



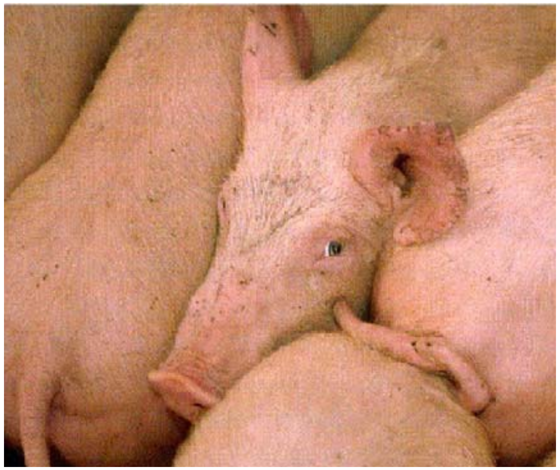
What can color be used for in computer vision?

Color histograms

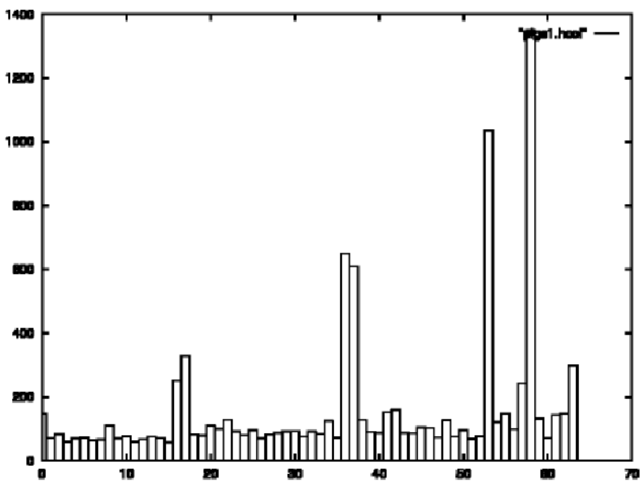
- A useful color feature descriptor for images
- Definition: Given color space defined by some axes (e.g., RGB):
  - discretize colors into *n* bins and
  - count the number of pixels having each of the *n* colors in the image

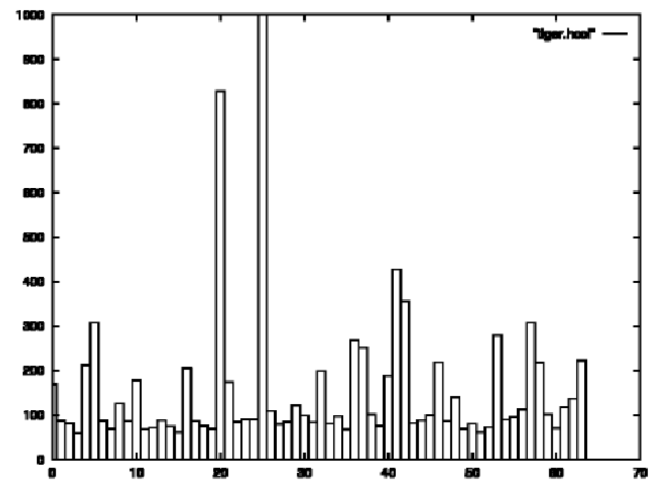
- Note: If 8 bits/channel, number of possible colors is 256^3 but *n* is usually chosen to be much smaller

Histograms of two color images



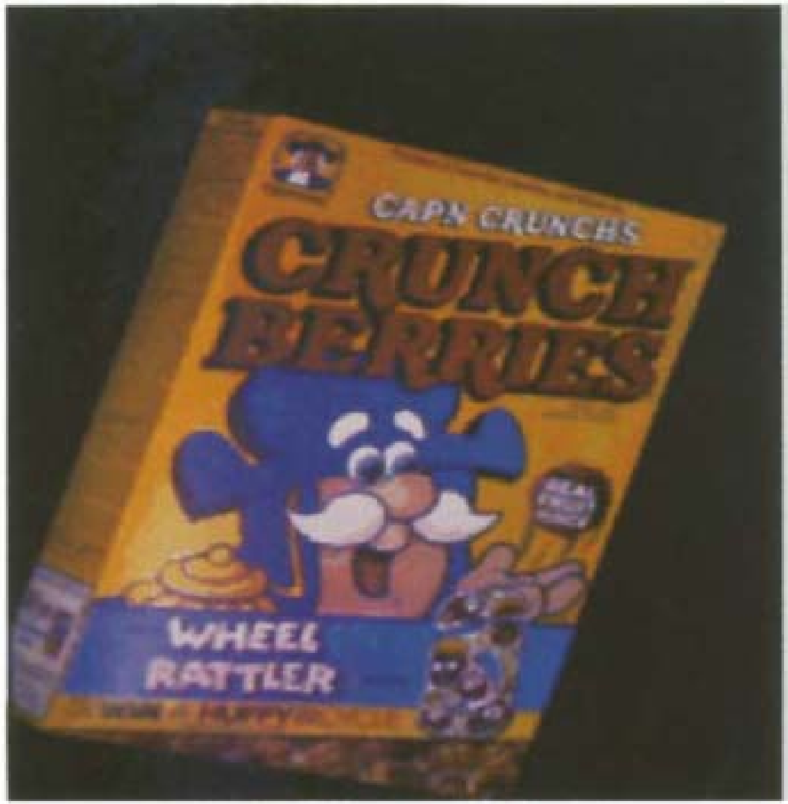
No. of pixels

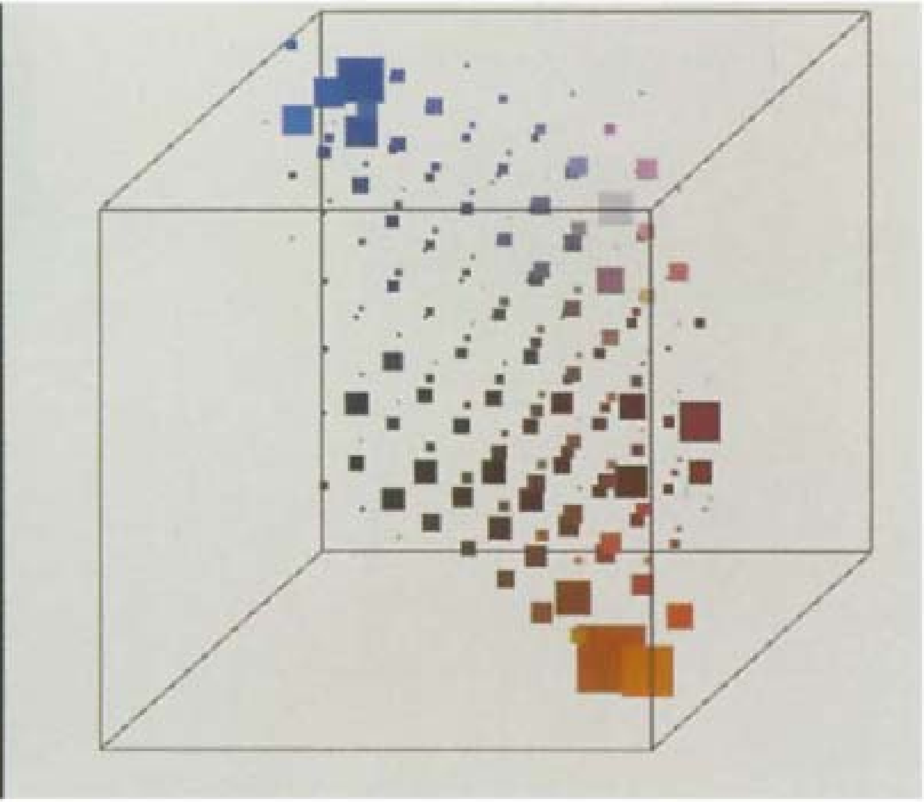




Color bins

Another Example





cereal box image

3D color histogram

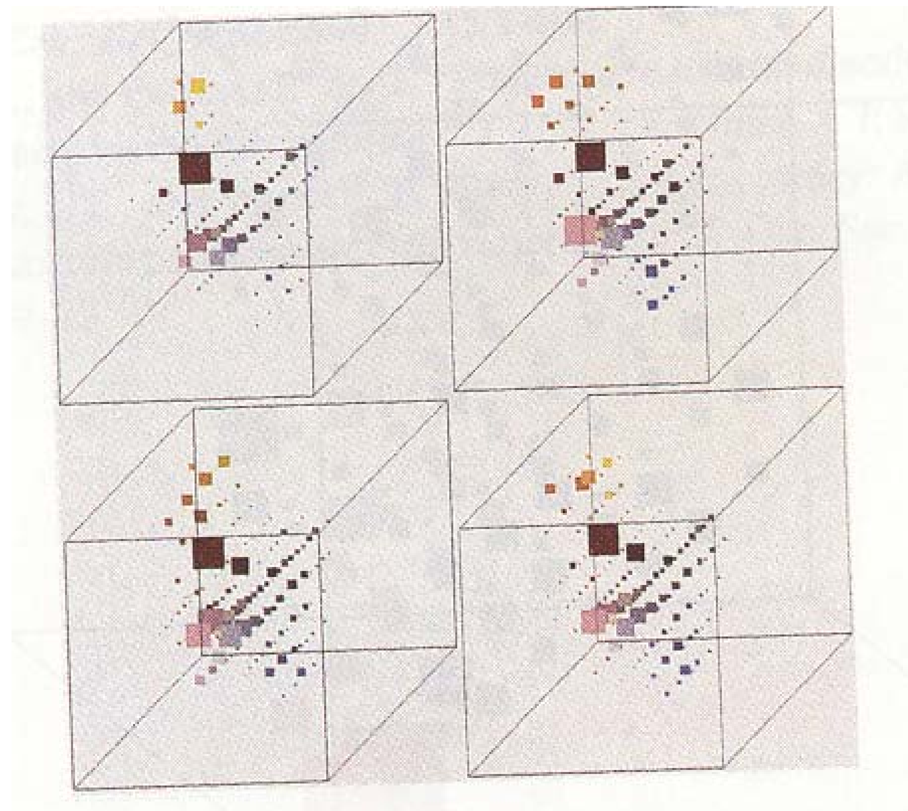
(Swain & Ballard, 1991)

Advantages of Color histograms

- Fast and easy to compute
- Invariant to:
  - Translation
  - Rotation about the viewing axis
- Changes slowly with
  - Rotation about other axes (viewpoint changes)
  - Distance to object (scaling)
  - Occlusion
- Can be used for image matching in content-based image retrieval, object identification, and object location
- NOT invariant to changes in lighting conditions

Invariance to Minor Viewpoint Changes

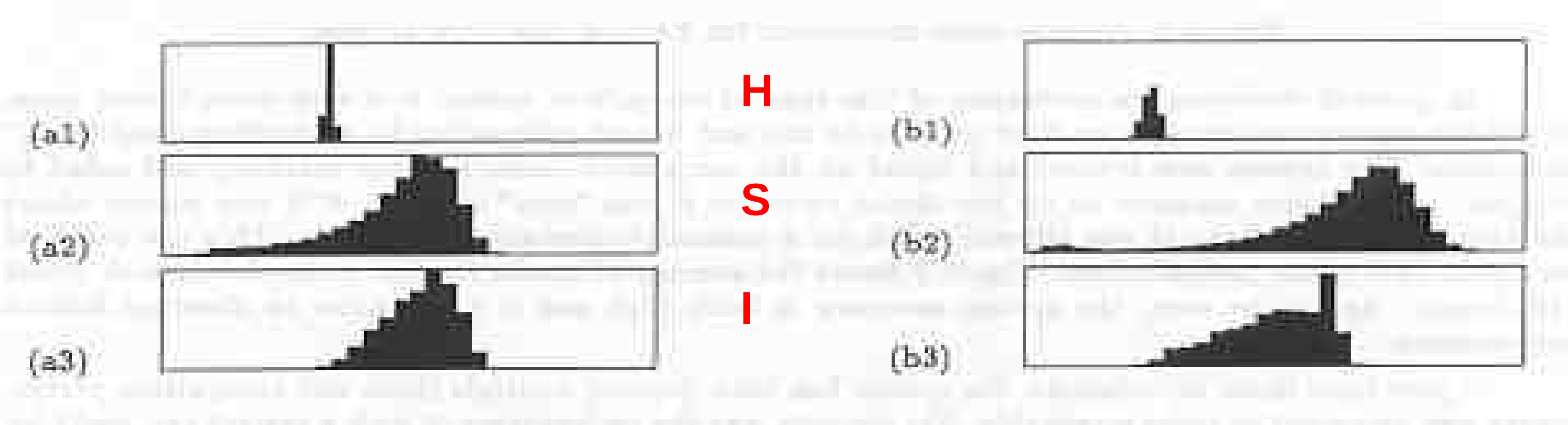
Four views of Snoopy



Histograms

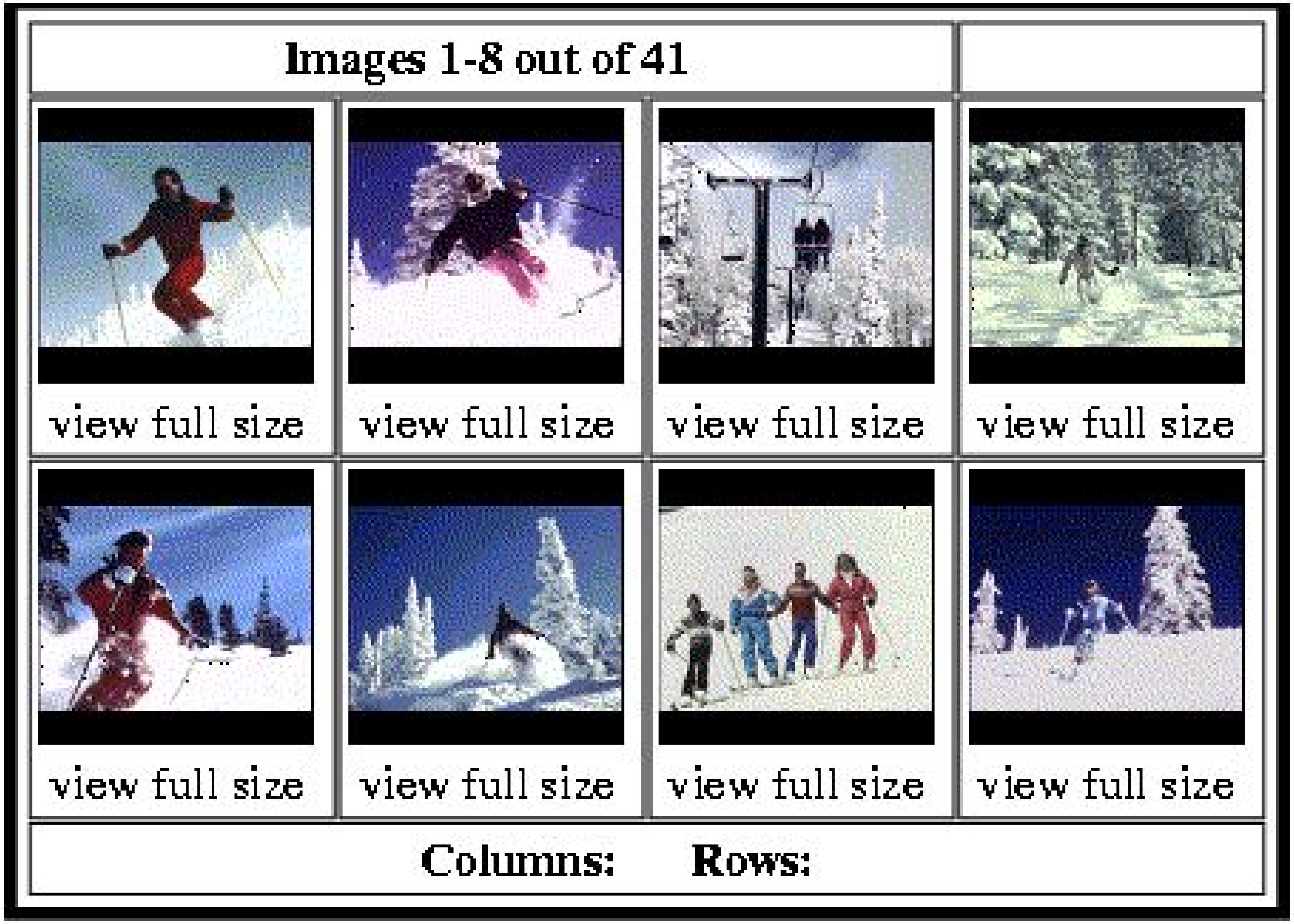
(Swain & Ballard, 1991)

Example Application: Apples versus Oranges



Separate HSI histograms for apples (left) and oranges (right) used by IBM's VeggieVision for recognizing produce at the grocery store checkout station (see Ch 16).

Example Application: Content-Based Image Retrieval



Top left image is query image. The others are retrieved by having similar color histogram (See Ch 8).

Color Histograms for Image Matching (Swain & Ballard, 1991)

Opponent Encoding:

- wb = R + G + B
- rg = R - G
- by = 2B - R - G

Histogram used: 8 wb x 16 rg x 16 by = 2048 bins (instead of 256^3 possible)

8 wb bins only because wb (intensity) is more sensitive to lighting variations

Given a color histogram for a new image and color histogram for a known model image, how do we compute similarity?

Color Histograms for Image Matching (Swain & Ballard, 1991)

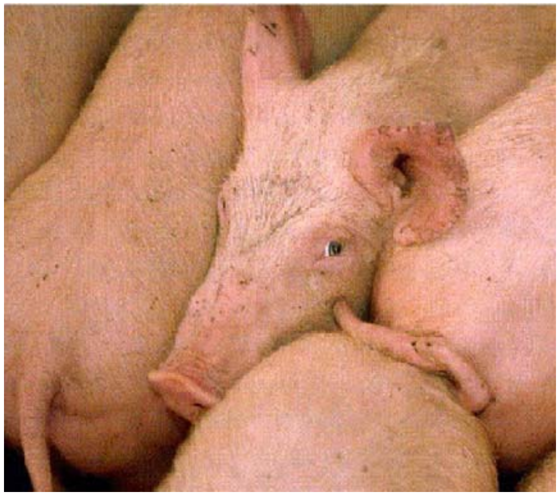
Histogram Intersection of image histogram I and model histogram M (both with n bins):

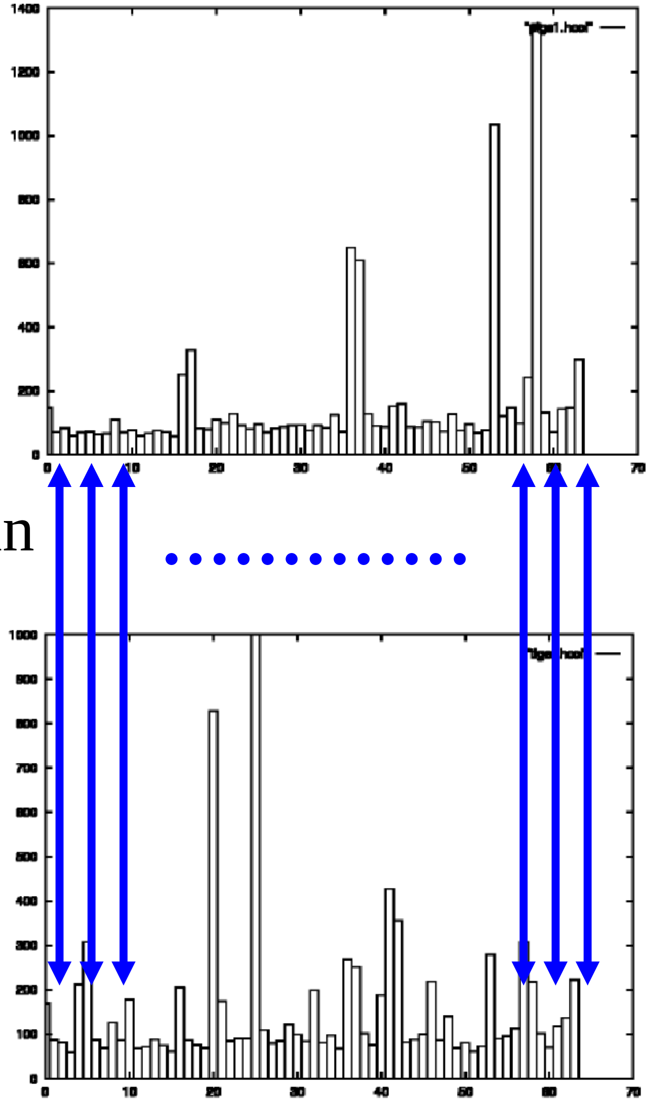
$$\mathrm{Int}(I,M) = \sum_{j=1}^{n} \min(I_j, M_j)$$

Normalized Histogram Intersection (gives a value between 0 and 1):

$$\mathbf{H(I,M) = Int(I,M) / \sum_{j=1}^{n} M_j}$$

Histogram Intersection Example

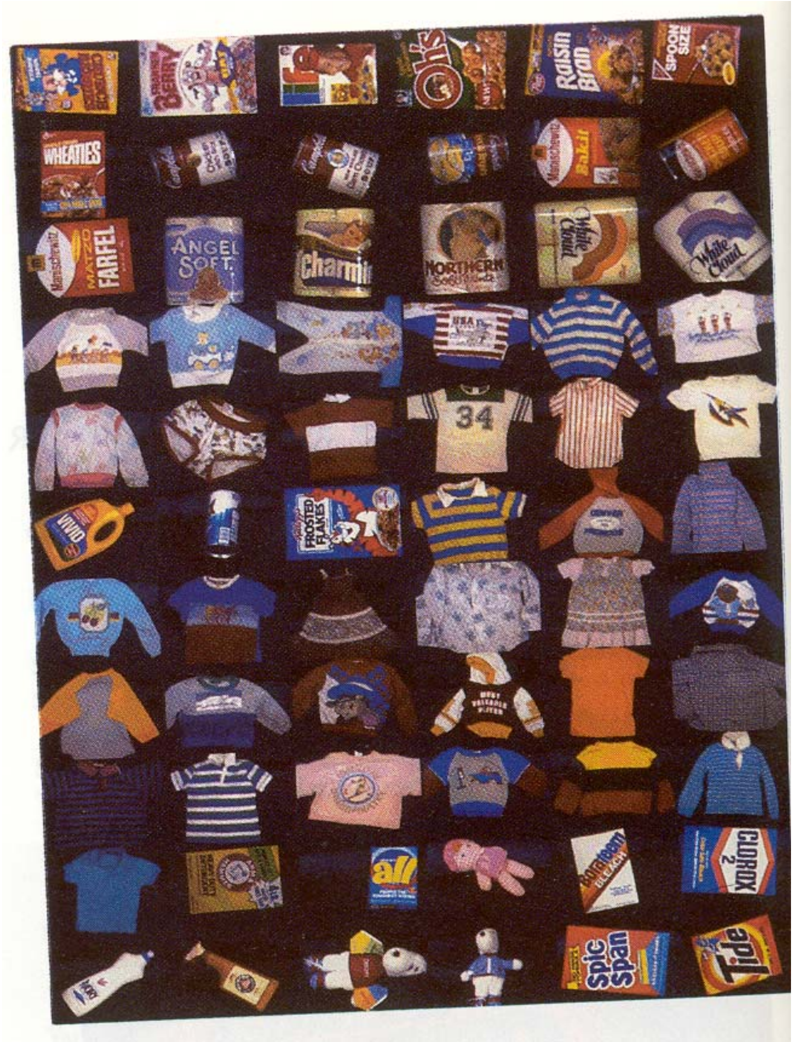




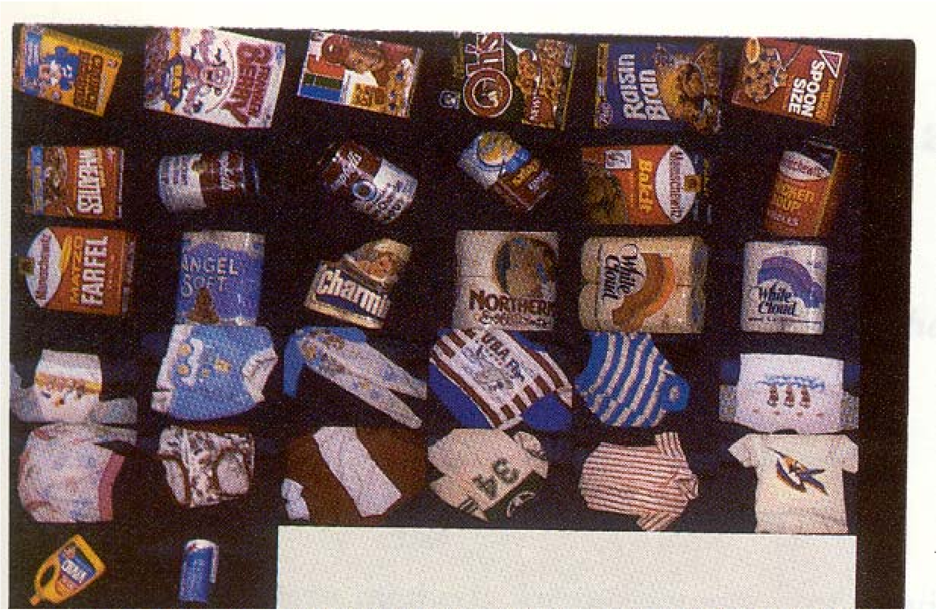
$\sum \min$

Objects for Training and Testing

(Swain & Ballard, 1991)



66 model objects

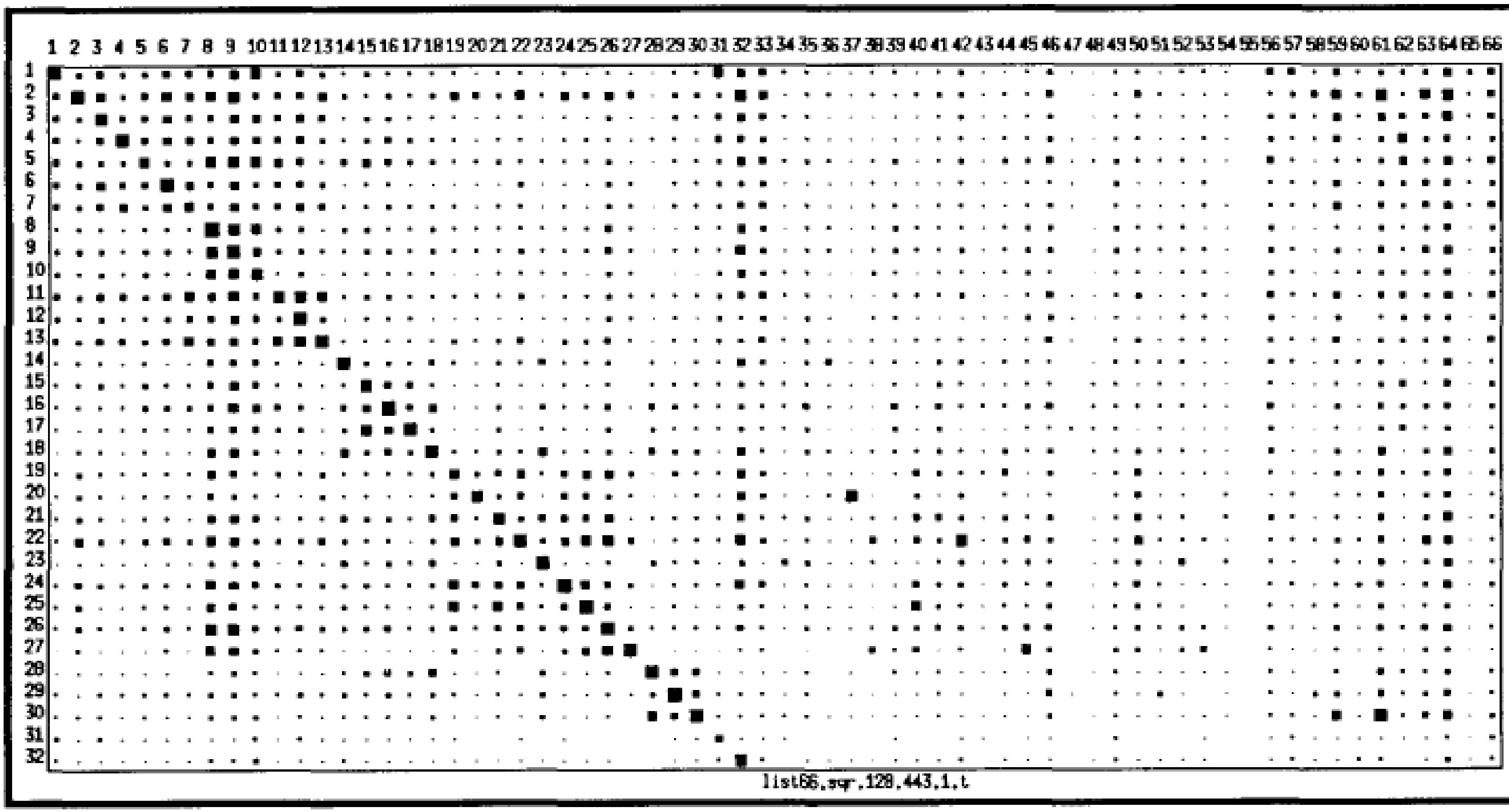


Some test objects
(rotated, scaled, occluded, deformed, etc.)

Object Identification (Image Retrieval)

Models

Unknown Objects

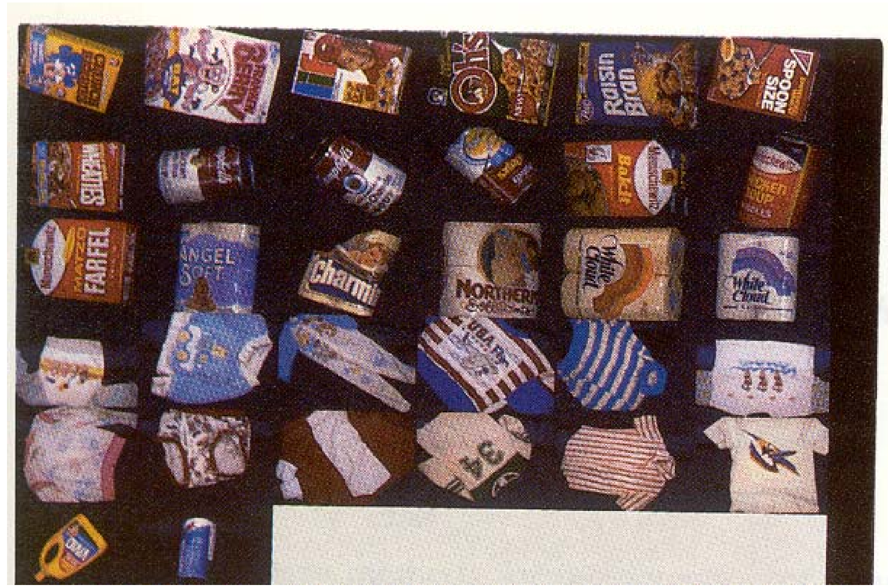


29 out of 32 correct; in 3 cases, correct model had second highest score

(Swain & Ballard, 1991)

Object Location Example

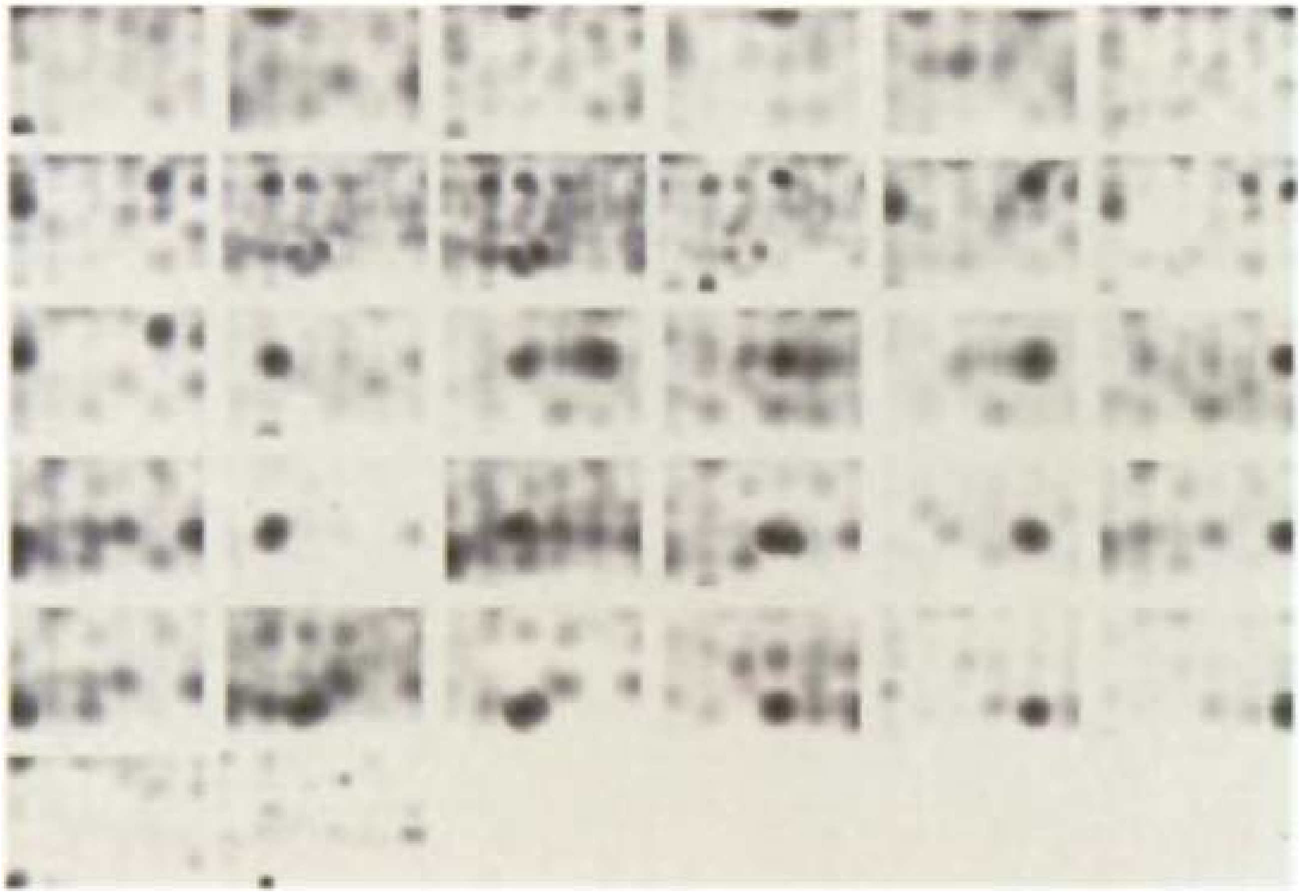
Input Image



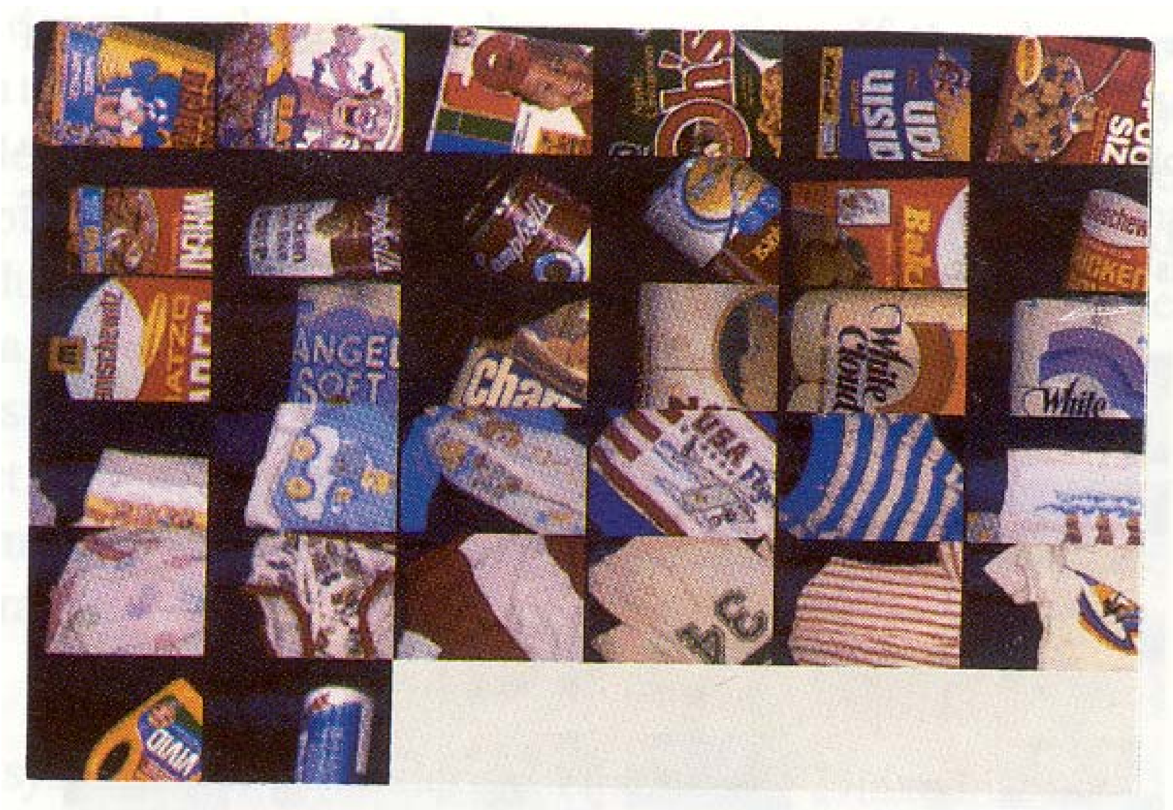
Result of histogram “backprojection” for blue striped shirt

Object Location: Results for All Objects

Result for obj 1   Result for obj 2   Result for obj 3 …



Occlusion Testing



Test objects used in occlusion experiments

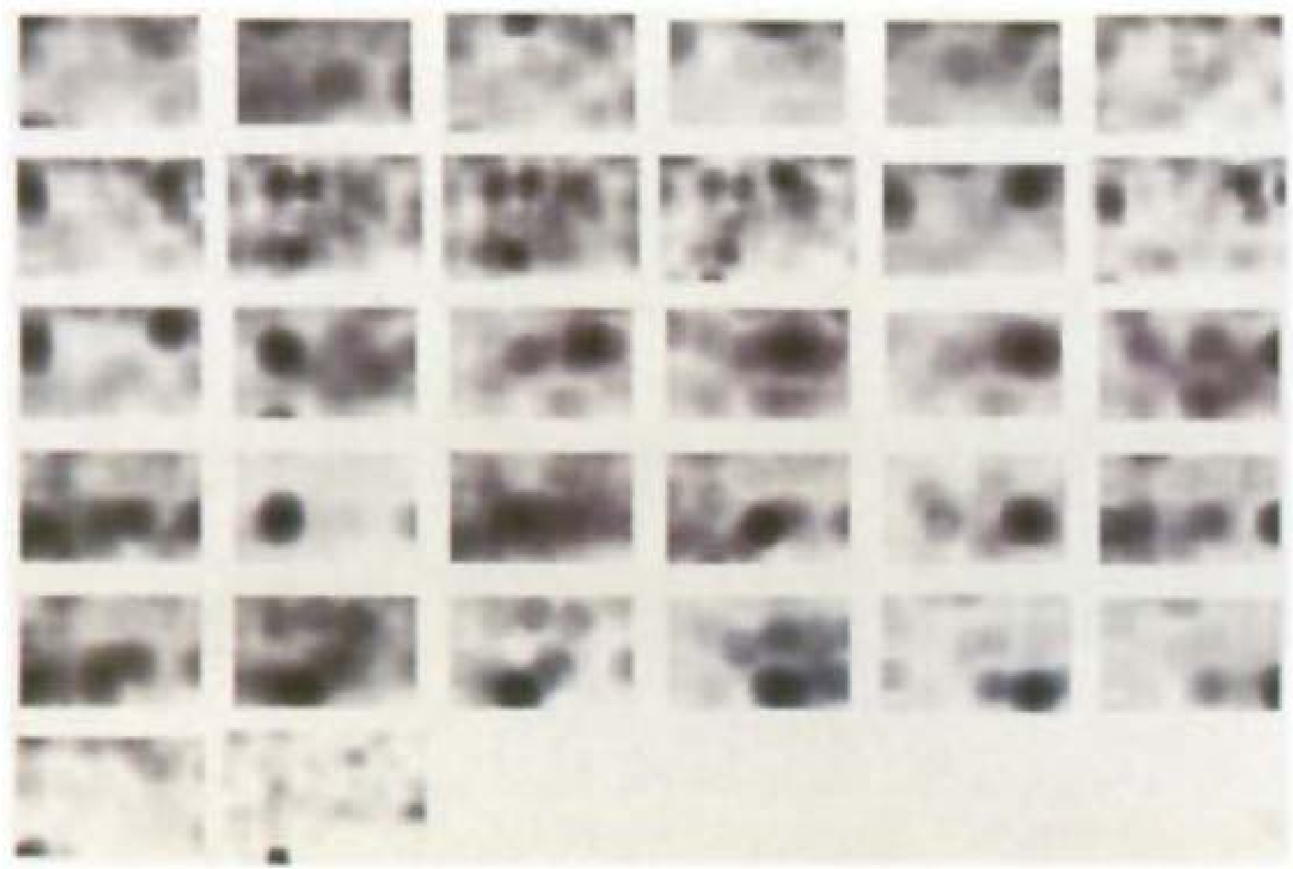
Invariance to Occlusion

Input Image

Result of histogram backprojection for blue striped shirt

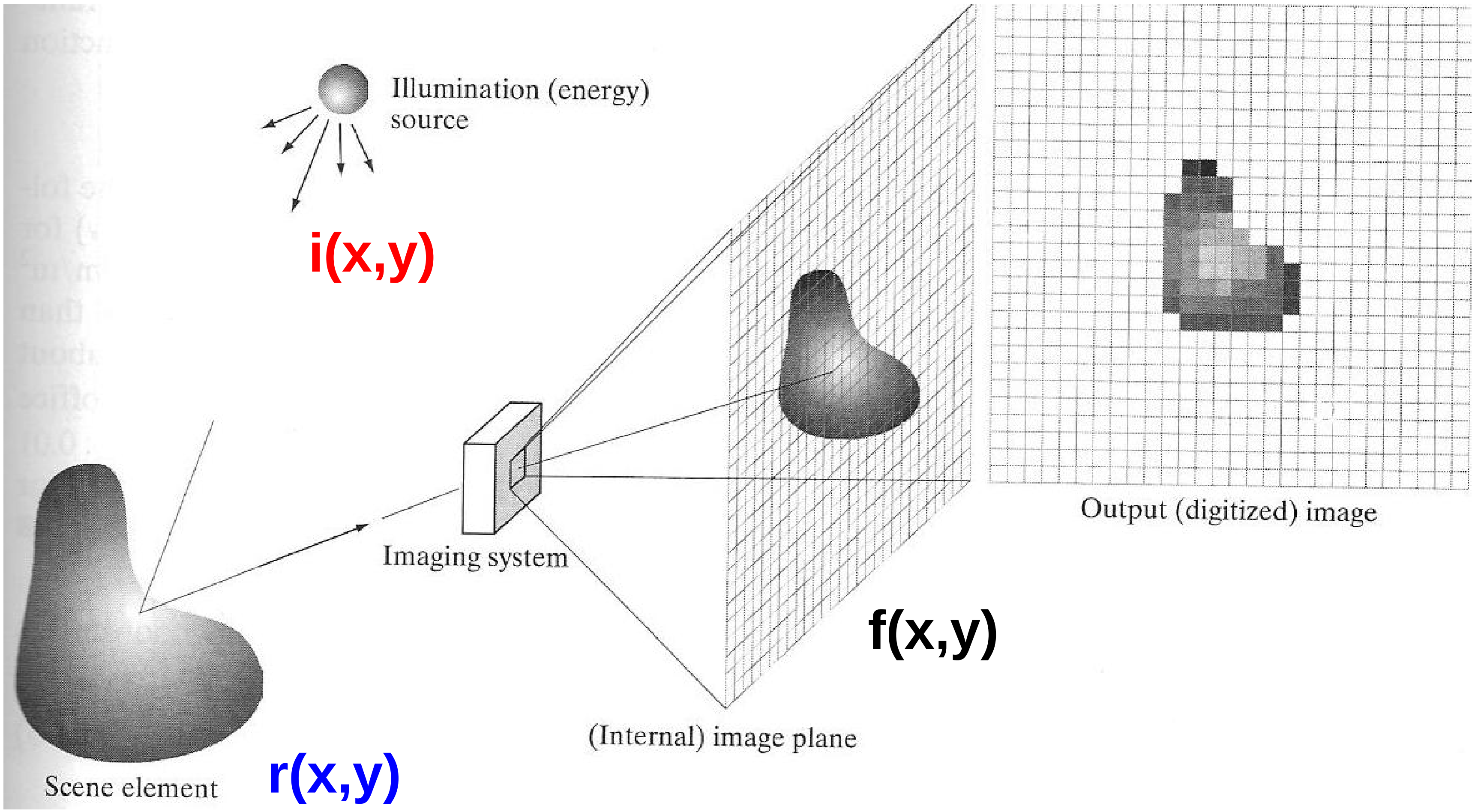
Finding Occluded Objects: Results for All Objects

Result for obj 1   Result for obj 2   Result for obj 3 …



How can we achieve invariance to illumination/lighting changes?

Recall: Image Formation Basics



(from Gonzalez & Woods, 2008)

Image Formation: Basics

Image f(x,y) is characterized by 2 components

1. **Illumination i(x,y)** = Amount of source illumination incident on scene
2. **Reflectance r(x,y)** = Amount of illumination reflected by objects in the scene

$$\boxed{f(x,y) = i(x,y)r(x,y)}$$

where

$$0 < i(x,y) < \infty \text{ and } 0 < r(x,y) < 1$$

r(x,y) depends on object properties
r = 0 means total absorption and 1 means total reflectance

What about ratios of sensor responses?

For slowly varying illumination, *i(x,y)* will be the same at neighboring locations

Rato of neighboring locations is independent of illumination:

$$\frac{f(x,y)}{f(x+1,y)} = \frac{i(x,y)r(x,y)}{i(x,y)r(x+1,y)} = \frac{r(x,y)}{r(x+1,y)}$$

Taking logarithms:

$$\ln f(x,y) - \ln f(x+1,y) = \ln r(x,y) - \ln r(x+1,y)$$

This is just the derivative of the logarithm of image!

Color Constant Color Indexing (Funt & Finlayson, 1995)

- Steps:
  - Take pixel-wise logarithm of R, G, B images
  - Convolve images with derivative filters (directional derivative or Laplacian)
  - Compute histogram as before but based on derivative images

COMPARATIVE PERFORMANCE: SYNTHETIC IMAGES WITH ILLUMINATION VARYING SPATIALLY IN INTENSITY AND SPECTRAL COMPOSITION

Indexing Algorithm	*Rank: 1*	*Failures*	*Av. Percentile*	*Av. Tolerance*
Swain's	7 out of 30	12	N/A	N/A
Laplacian of Gaussian ($\sigma = 1$)	30 out of 30	0	1.000	.567207

Images retrieved

Input

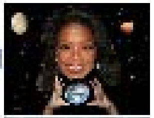
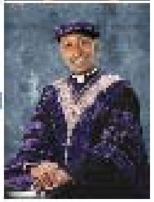
Color Indexing

Color constant indexing

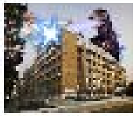
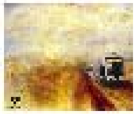
Project 1 Intelligent Scissors
Artifact Voting Results

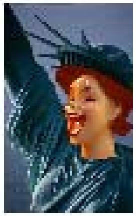
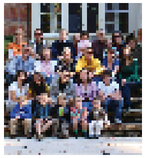
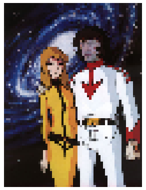
And the nominees are…

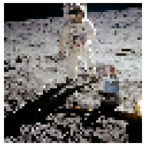
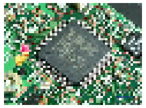


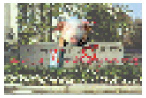




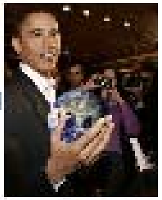


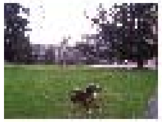
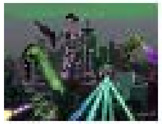




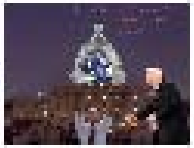




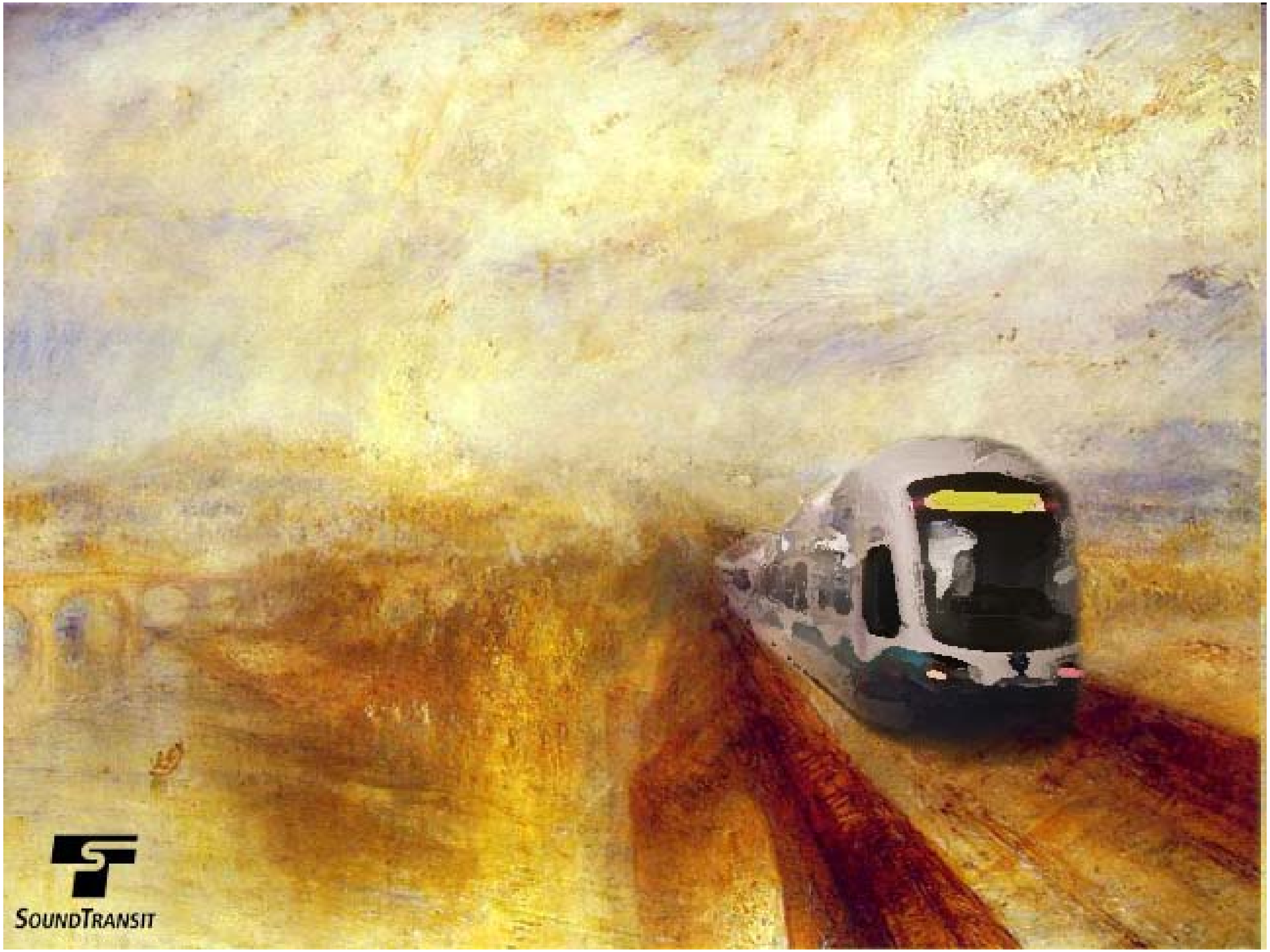




And the winner is…

Sorry, not eligible – done using real and not intelligent scissors…

1st place: James R. George (81 points)



2nd place: Lars O. Bull (77 points)

3rd place: Johnathan G. Lyon (76 points)

Honorable mention: Mark D. Agoncillo (74 points)

Honorable mention: Elizabeth G. Muhm (74 points)

Next Time: Texture

Things to do:

- Finish Project 2
- Read Chap. 7