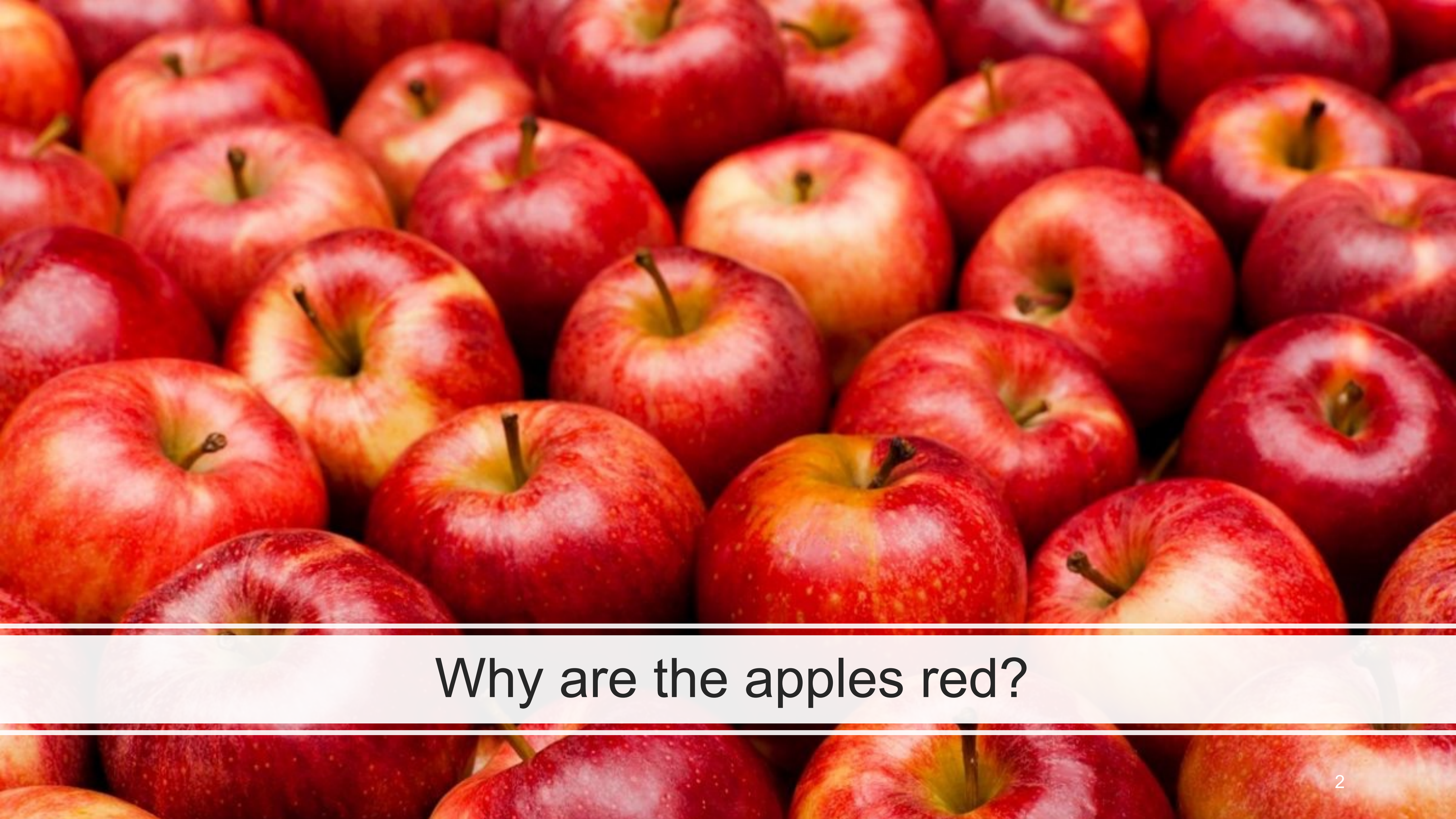
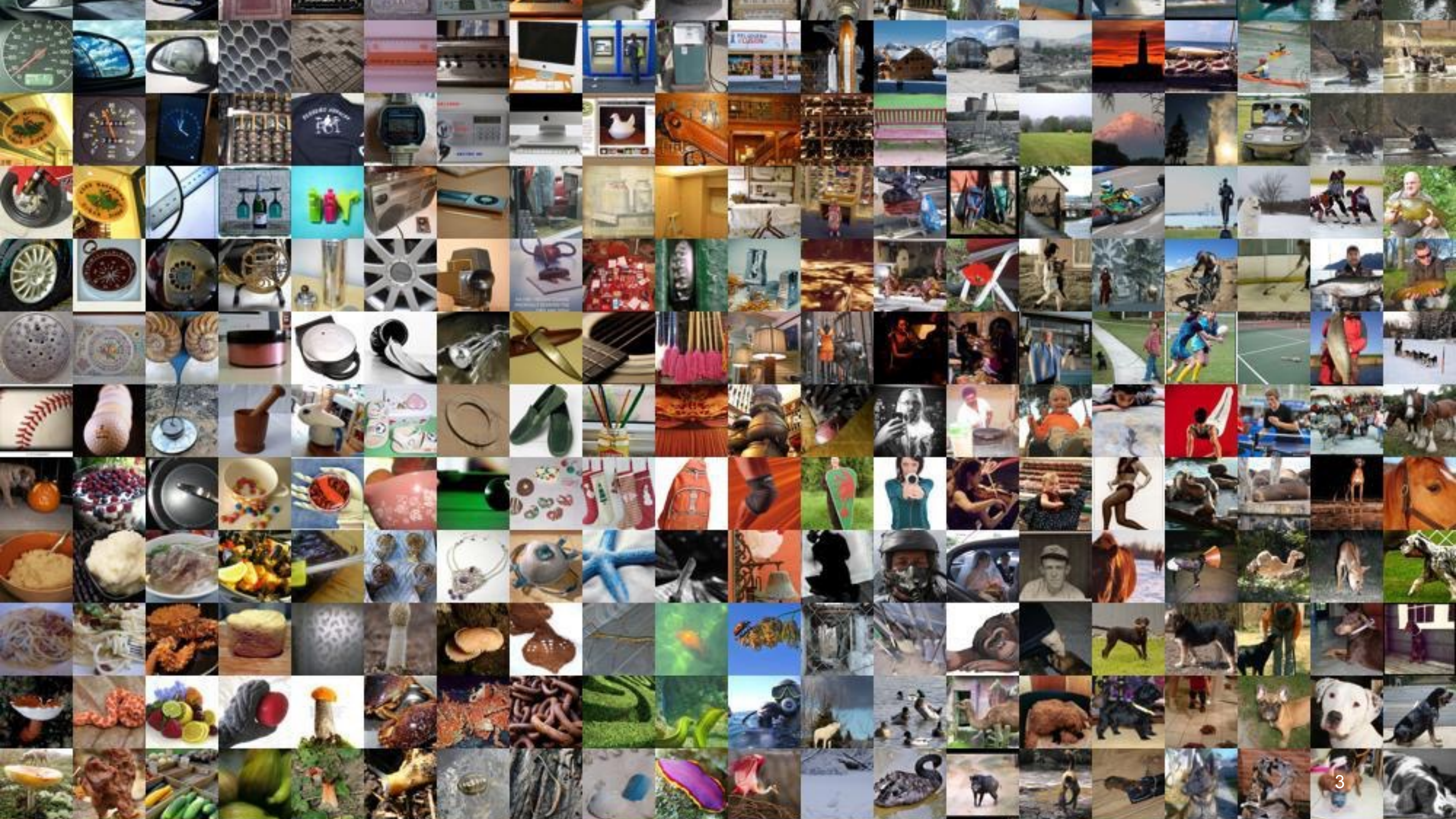
THE UNIVERSITY OF TEXAS AT DALLAS

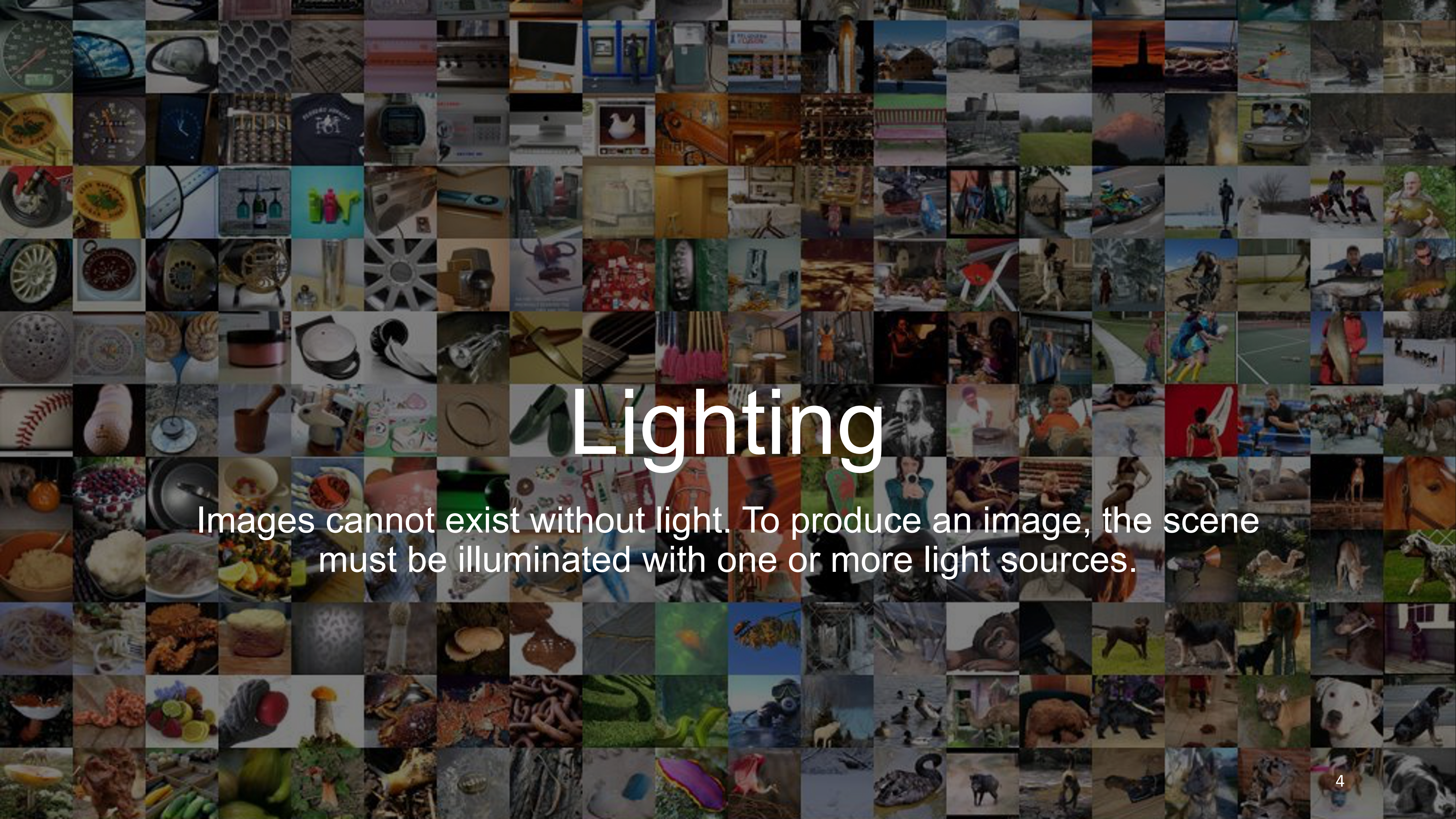
Image Formulation: Lighting and Color

CS 6384 Computer Vision
Professor Yapeng Tian
Department of Computer Science



Why are the apples red?



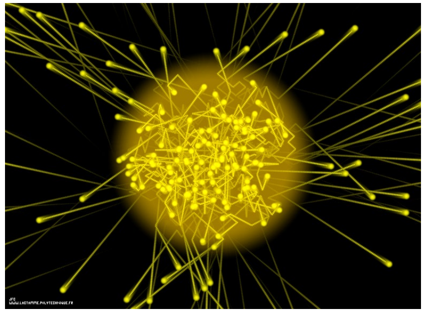


Lighting

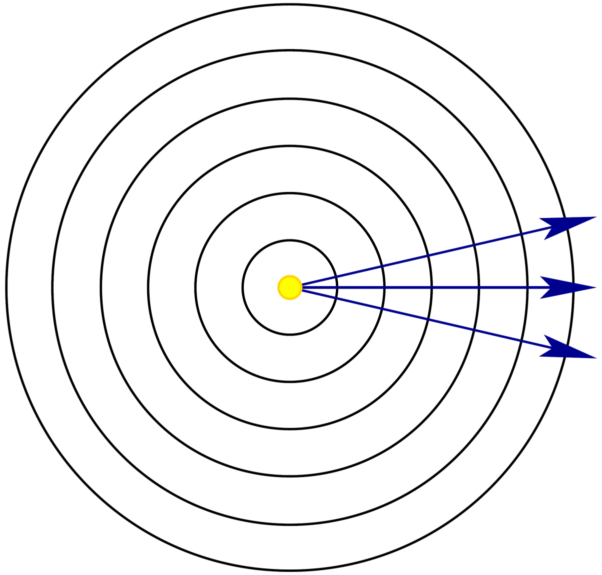
Images cannot exist without light. To produce an image, the scene must be illuminated with one or more light sources.

Basic Behavior of Light

Light can be described in three ways

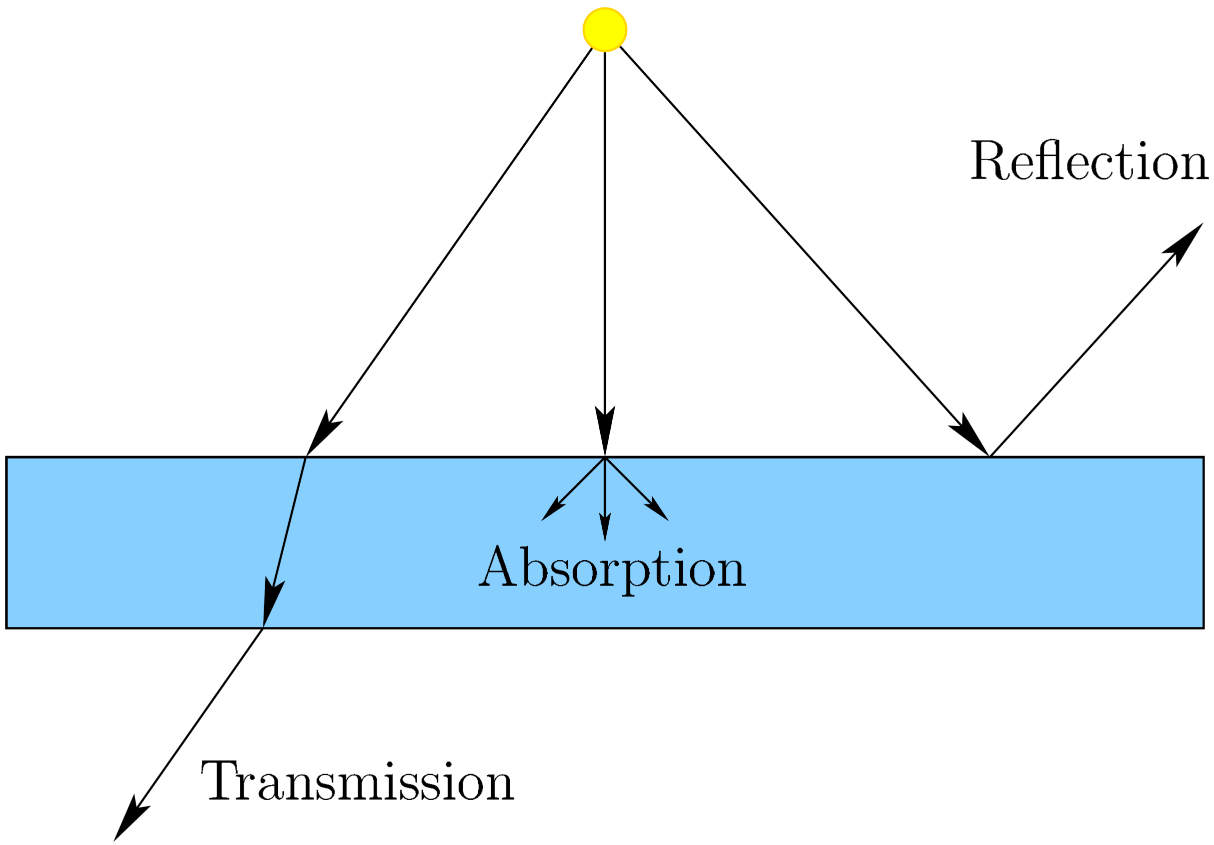


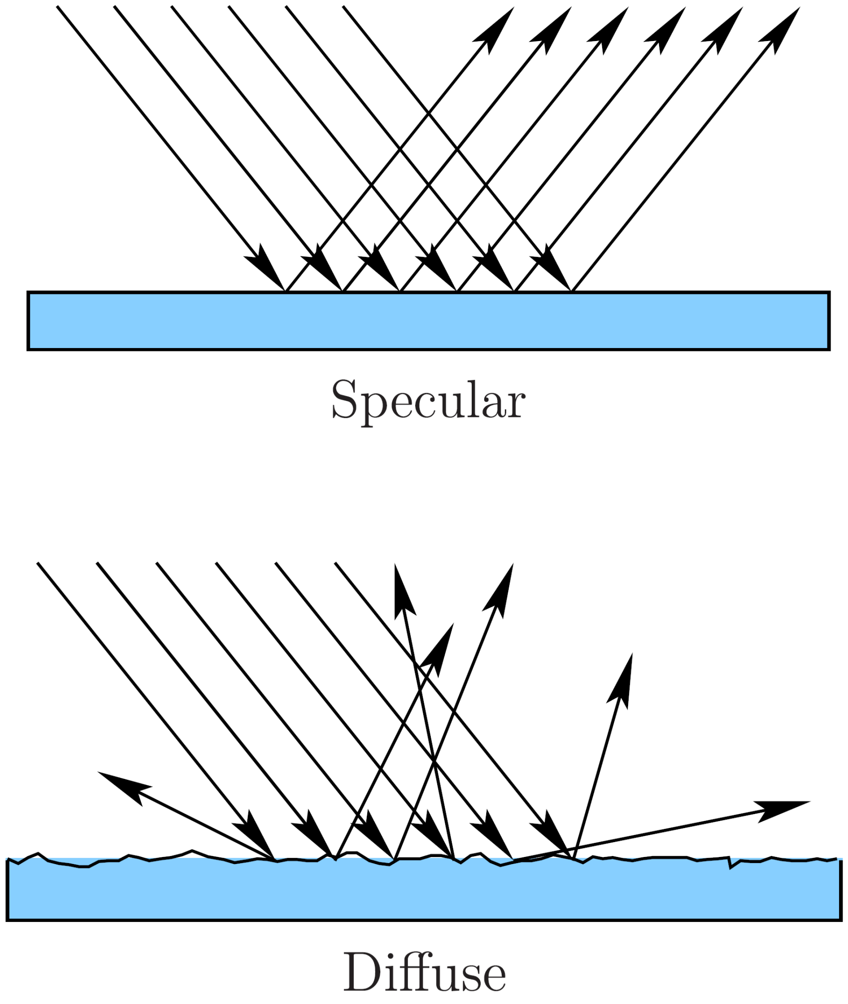
- Photons: tiny particles of energy moving through space at high speed



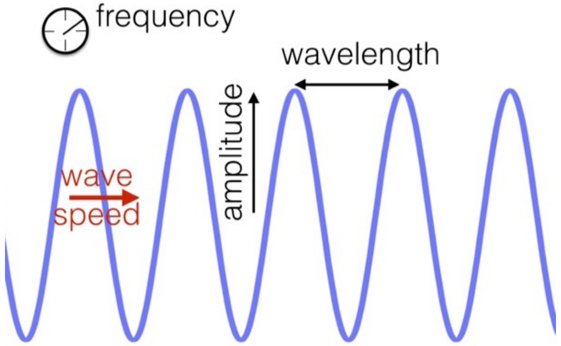
- Waves: ripples through space
- Rays: a ray traces the motion of a single hypothetical photon

Interactions with Materials

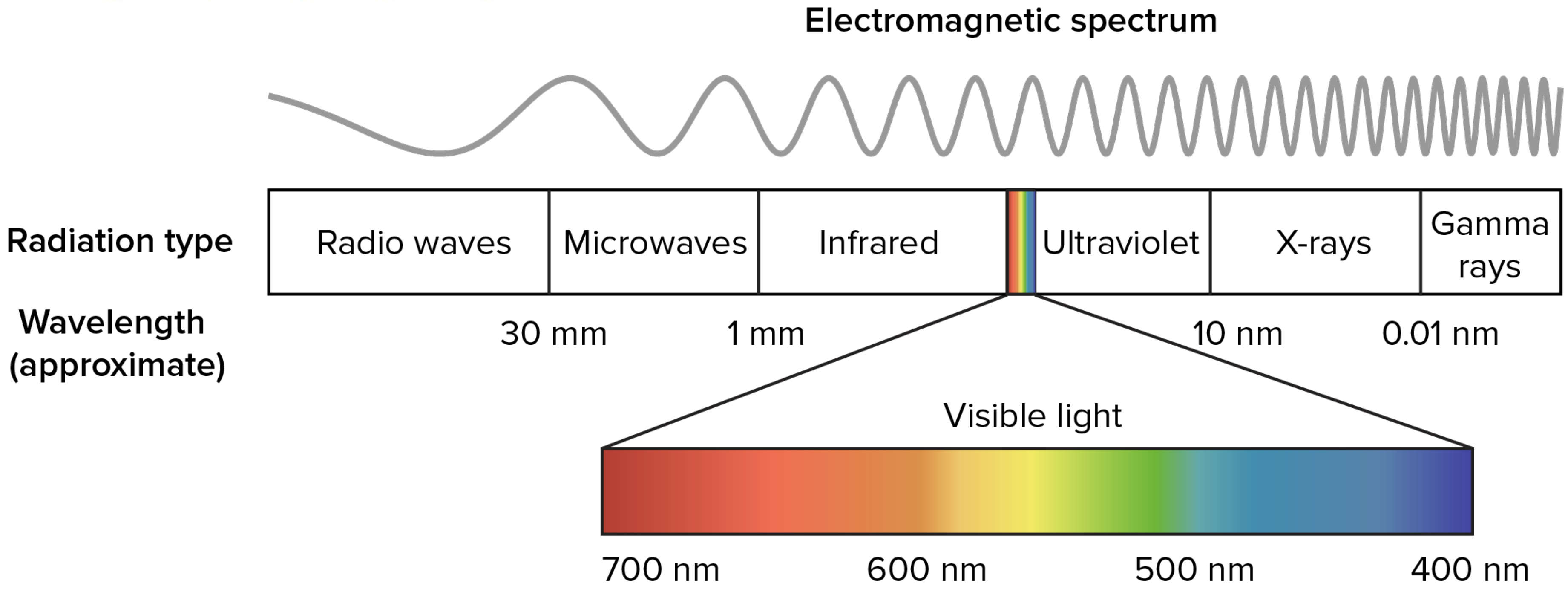




Wavelengths and Colors

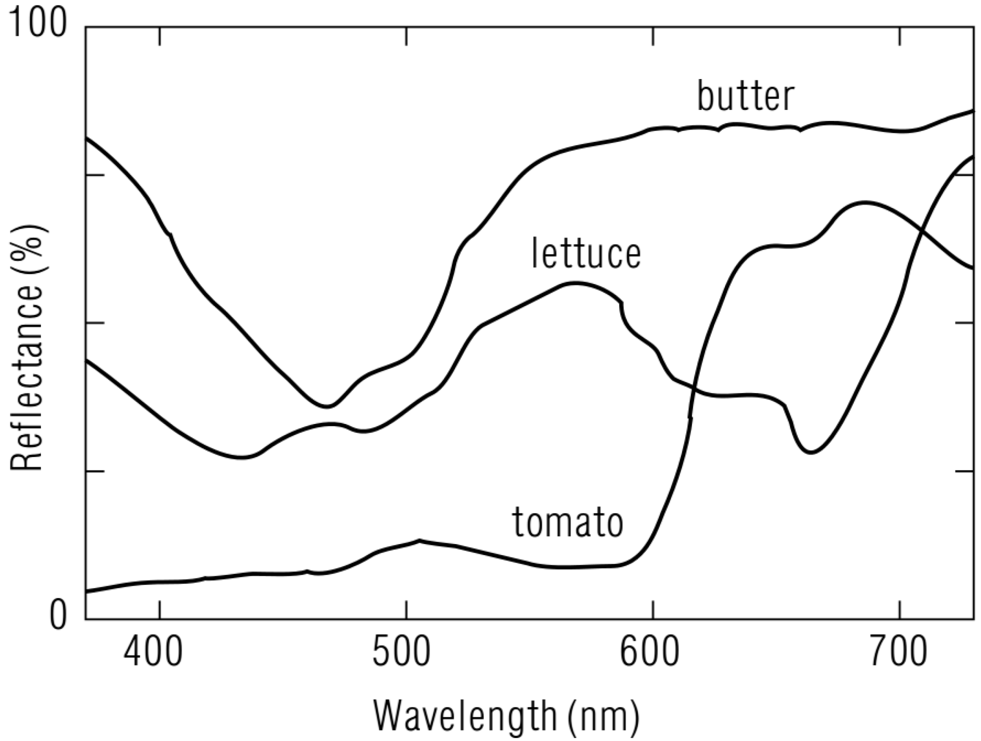


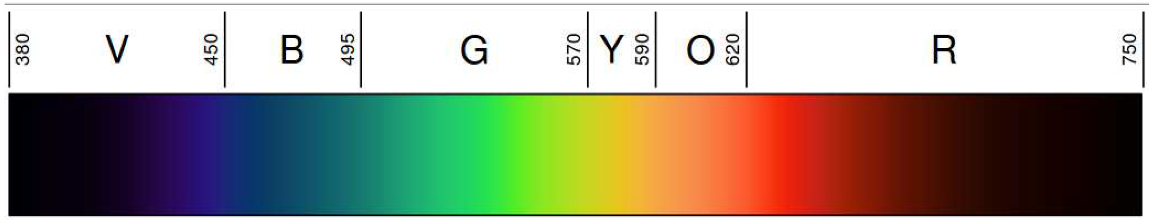
Wavelength $\lambda = \frac{v}{f}$ Speed / Frequency



Reflection of Materials

We see objects with different colors because the materials reflect specific colors differently



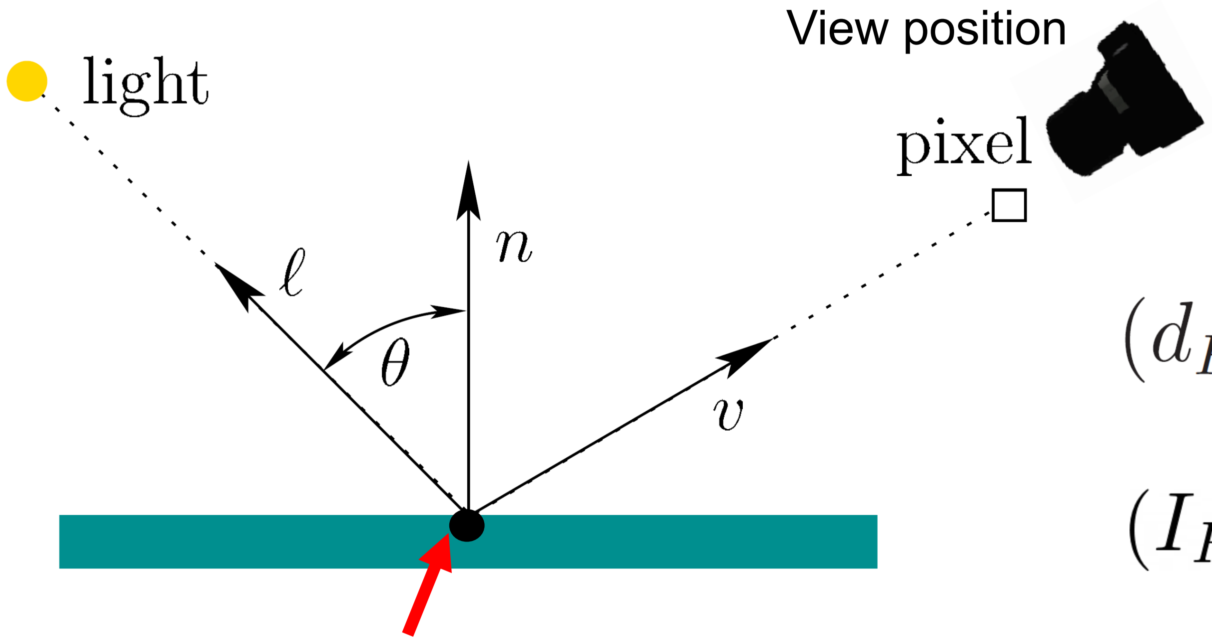


The Color of an Object Depends Upon the Light Source

Selective Reflection

https://www.youtube.com/watch?v=xA8MT6yhP4w

Lambertian Lighting



Given a 3D point, we want to compute its color on the image

Diffuse reflection

$$R = d_R I_R \max(0, n \cdot \ell)$$
$$G = d_G I_G \max(0, n \cdot \ell)$$
$$B = d_B I_B \max(0, n \cdot \ell)$$
$$n \cdot \ell = \cos\theta$$

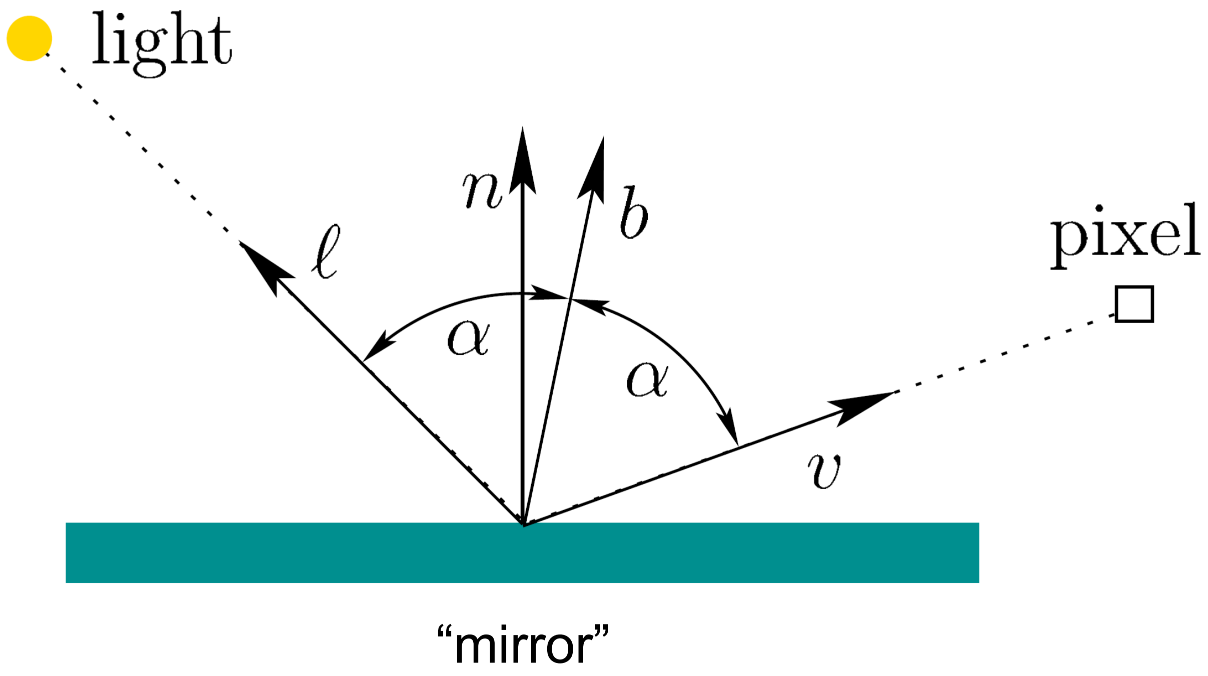
(d_R, d_G, d_B) Reflectance property of the material

(I_R, I_G, I_B) Spectral power distribution of the light source

$$L = dI \max(0, n \cdot \ell) \quad n \cdot \ell < 0$$

Light behind surface

Blinn-Phong Lighting



Related to specular reflection

$$b = \frac{\ell + v}{\|\ell + v\|}$$

x Material property that expresses the amount of surface shininess

x=100, mild amount of shininess

x=10000, almost like a mirror $0.99^{10000} = 2.24^{-44}$

s Specular reflectance property of the material

$$L = dI \max(0, n \cdot \ell) + sI \max(0, n \cdot b)^x$$

Ambient Lighting

Ambient lighting provides the general illumination of an environment

Independent of light/surface position, viewer, normal

Adding some background color

Ambient light

$$L = dI \max(0, n \cdot \ell) + sI \max(0, n \cdot b)^x + L_a$$

Multiple Light Sources and Attenuation

N light sources

$$L = L_a + \sum_{i=1}^{N} dI_i \max(0, n \cdot l_i) + sI_i \max(0, n \cdot b_i)^x$$

Attenuation: the greater the distance, the low the intensity

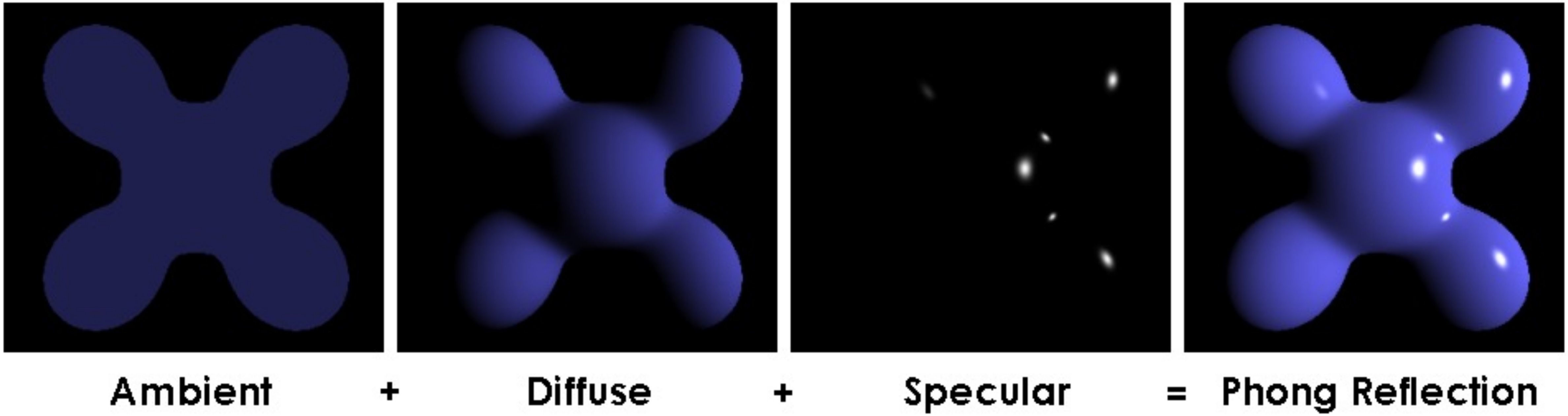
$$L = L_a + \sum_{i=1}^{N} \frac{1}{k_c + k_l c + k_q c^2} \left(dI_i \max(0, n \cdot l_i) + sI_i \max(0, n \cdot b_i)^x \right)$$

c Light source distance to surface

constant linear quadratic attenuation

Used by OpenGL for ~25 years

Phong Reflection Model



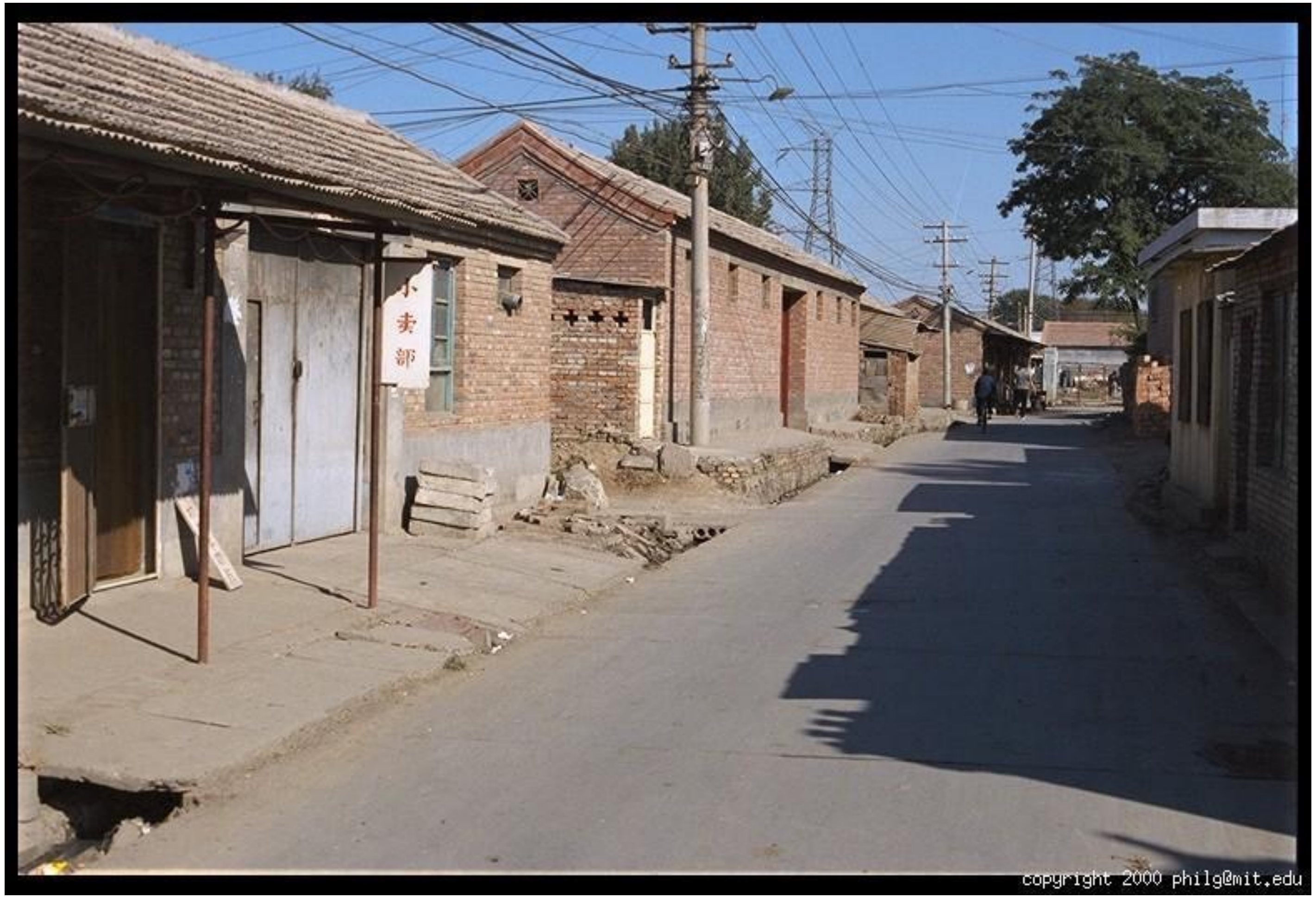
Color Formulation

When the incoming light hits the imaging sensor, light from different parts of the spectrum is integrated into the discrete red, green, and blue (RGB) color values that we see in a digital image.

Mixing different colors can obtain a new one

- Red+green makes yellow
- Red+blue+green makes white

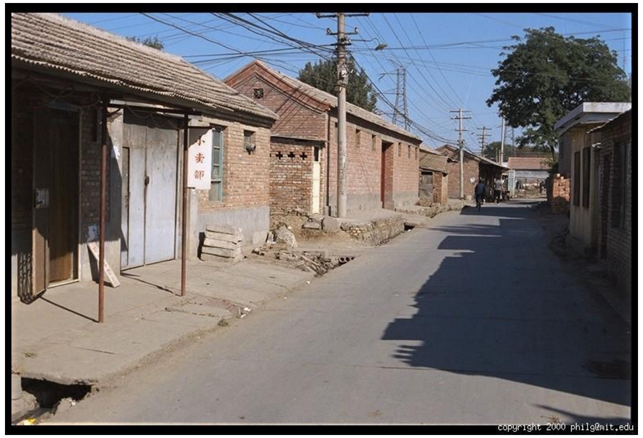
Color Images



Slide Credit: J. Hays

Color Images

Combined



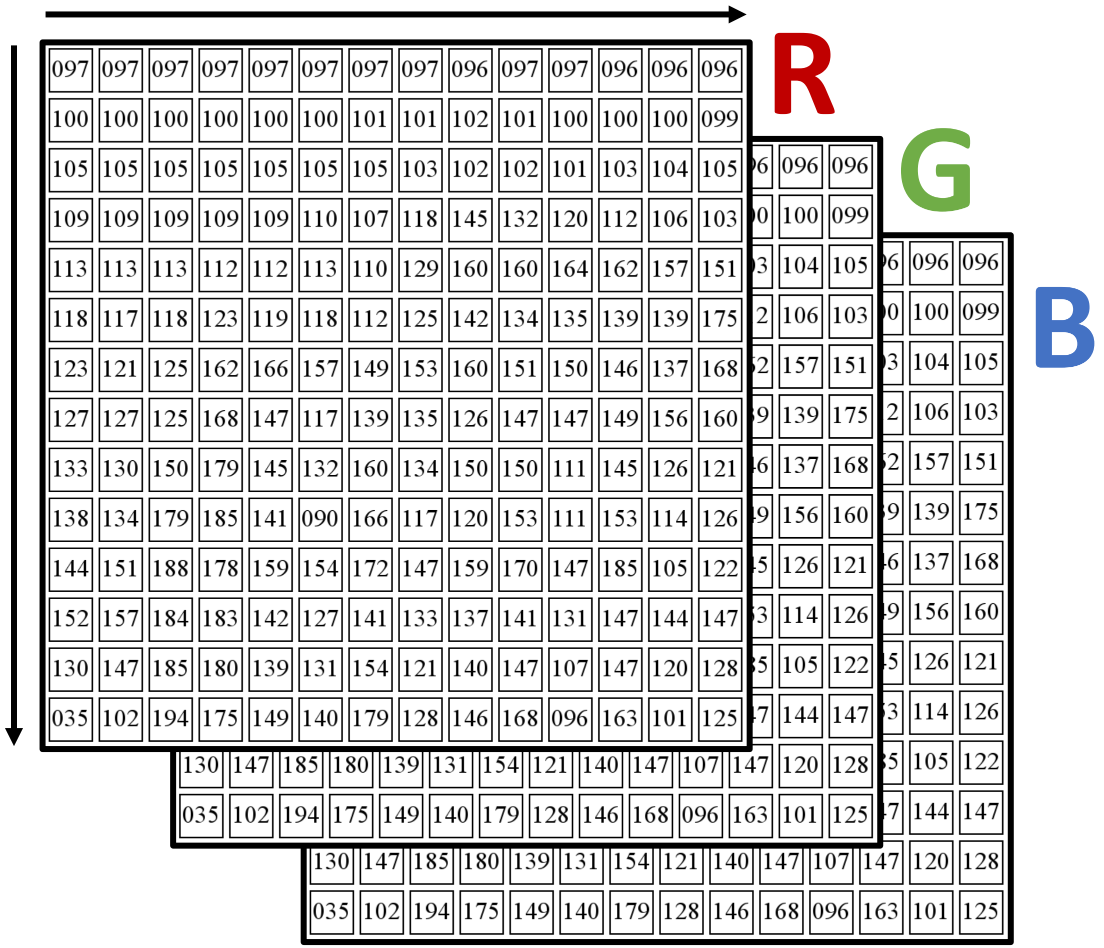
Red

Green

Blue

Slide Credit: J. Hays

Images in Python



Slide Credit: D. Fouhey, J. Johnson

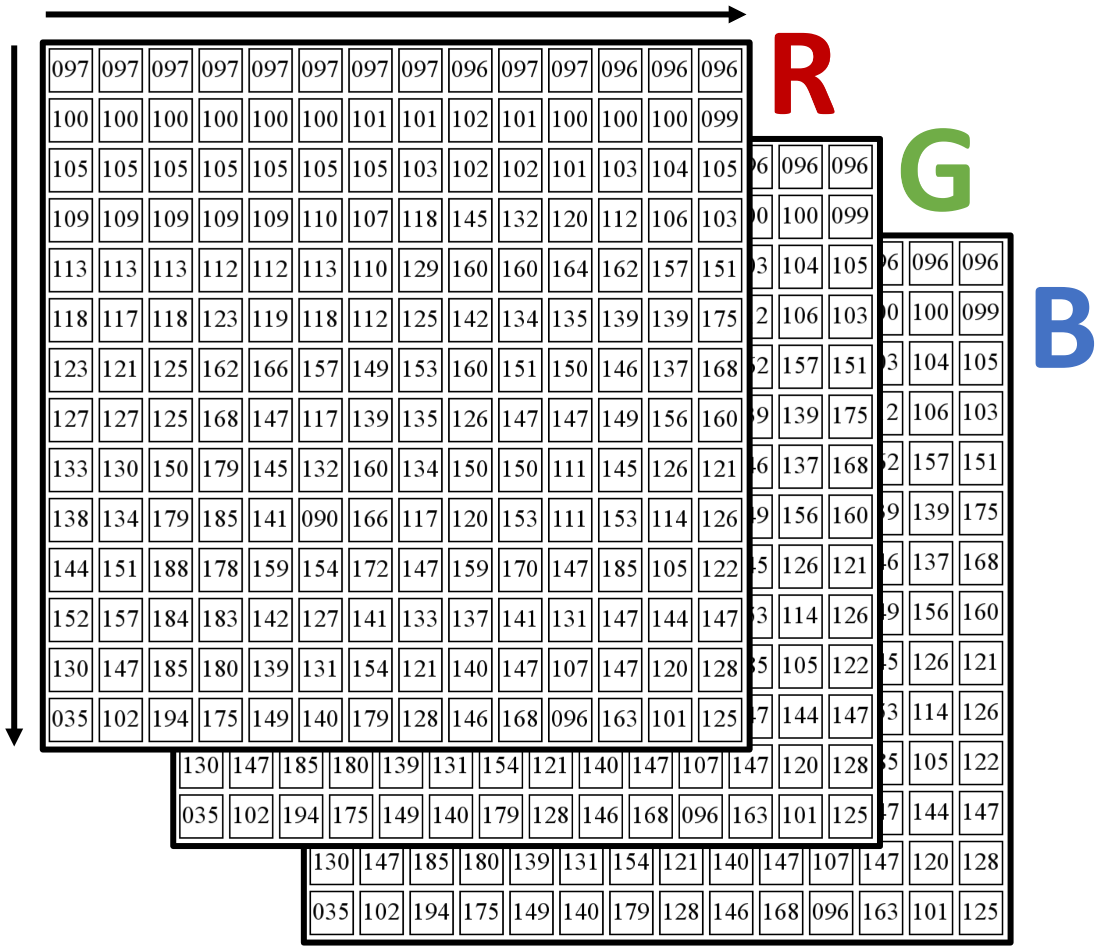
Images in Python

Images are matrix / tensor `im`

`im[0,0,0]`
top, left, red

`im[y,x,c]`
row y, column x, channel c

`im[H-1,W-1,2]`
bottom right blue



what is the index for bottom right red?

Slide Credit: D. Fouhey, J. Johnson

Few Things to Remember

- Origin is top left

- Rows are first

- Usually referred to as HWC (Height x Width x Channel). But you'll sometimes see CHW (especially with neural networks)

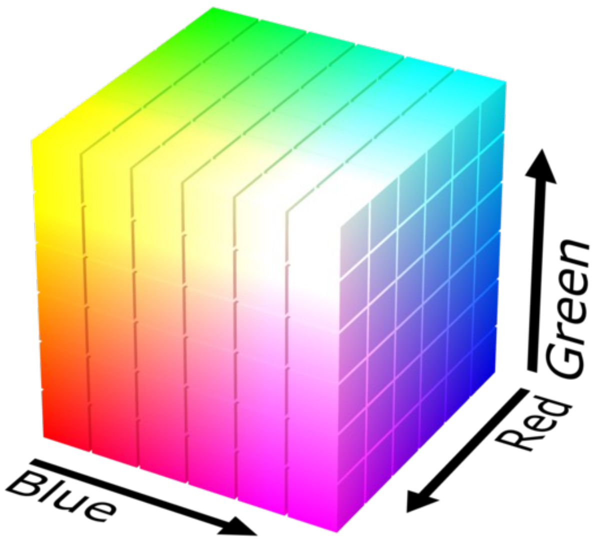
- Typically stored as uint8 [0,255]

Slide Credit: D. Fouhey, J. Johnson

RGB Color Space

Pros
1. Simple
2. Common

Cons
1. Distances don't make sense
2. Correlated

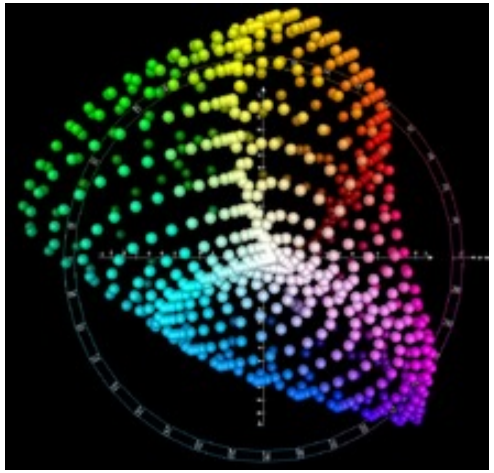


Slide Credit: J. Hays, RGB cube: https://en.wikipedia.org/wiki/RGB_color_model

LAB Color Space

Pros

1. Distances correspond with human judgment
2. Useful for color correction



Cons

1. Complex to calculate (don't write it yourself, lots of calculations)

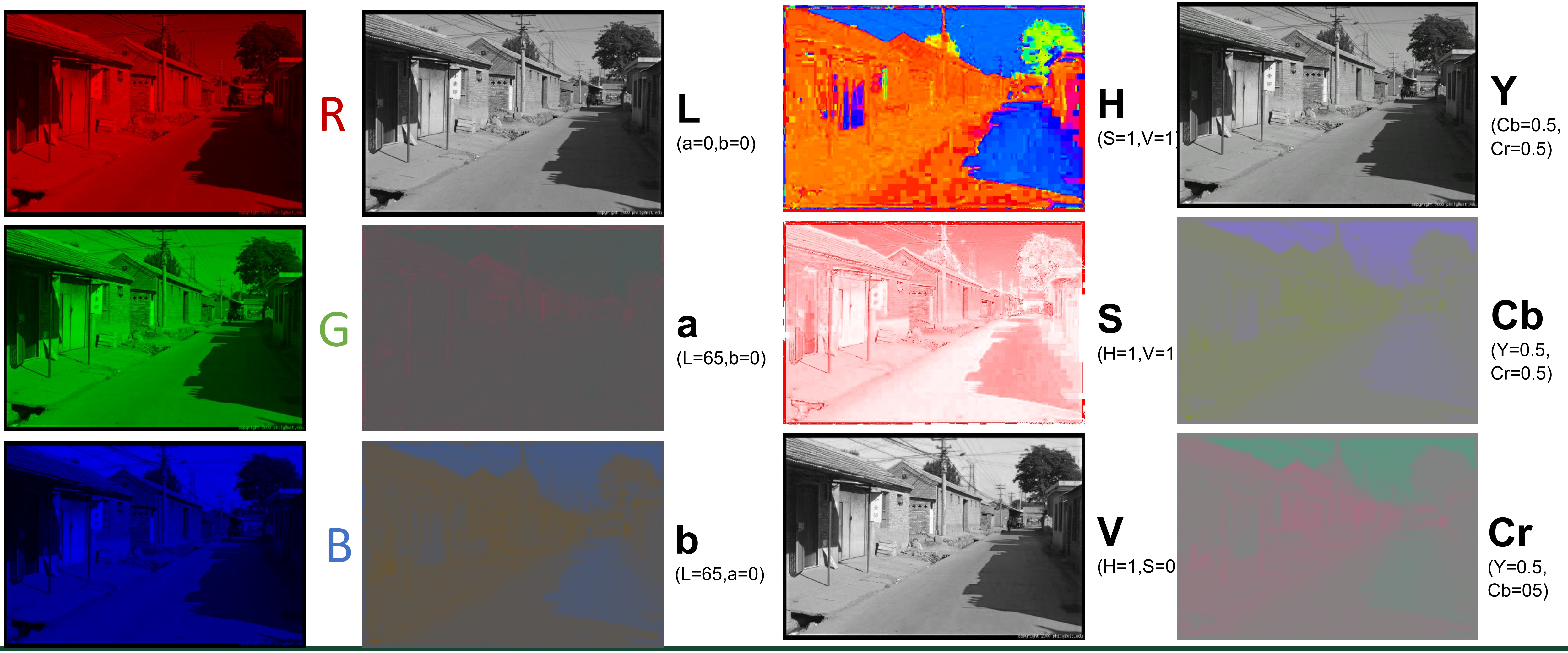
L
(a=0,b=0)

a
(L=65,b=0)

b
(L=65,a=0)

Slide Credit: J. Hays, Lab diagram cube: https://en.wikipedia.org/wiki/CIELAB_color_space

Different Color Spaces



Different Color Spaces

- RGB: sort of intuitive, standard, everywhere
- HSV: good for picking specific colors, fast to compute from RGB
- YCbCr/YUV: fast to compute, great for compression
- Lab: the right(?) thing to do, but “slow” to compute

RGB space is commonly used to represent colorful images in most of our applications

Color Conversion: One Example

Question: how to convert a RGB image to a Grayscale image?

`im[y,x,c]`

`im[y,x]`

RGB Color to Gray Conversion

RGB2Gray function: $I = 0.2989 * R + 0.5870 * G + 0.1140 * B$

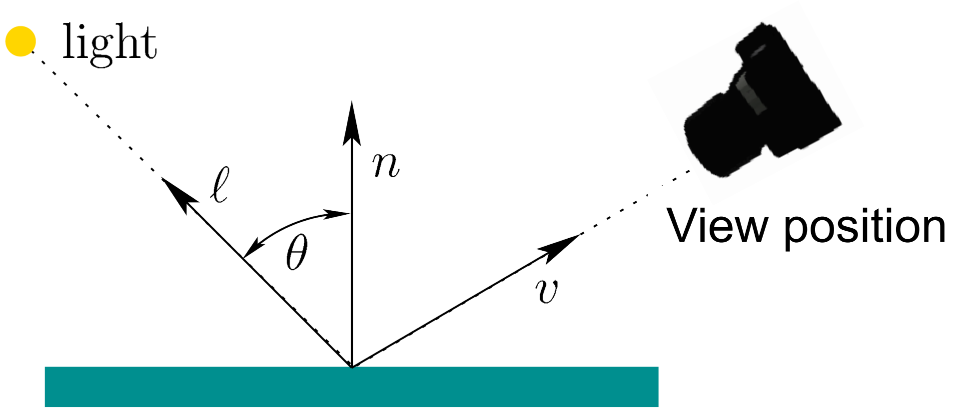
Based on research on human vision, we know that our eyes react to each color in a different manner.

Specifically, our eyes are more sensitive to green, then to red, and finally to blue.

Summary

Lighting Computation:

- compute color given material properties, light source color and position, normal position, view position



Color Space:

- a color can be represented by three primaries, such as RGB
- there are different color spaces, and they can be converted to each other
- im[y,x,c] – row, col, channel

Further Reading

Chapters 2.2.1, 2.2.2, and 2.3.2, Computer Vision: Algorithms and Applications, Richard Szeliski

Chapter 7.1, Virtual Reality, Steven LaValle