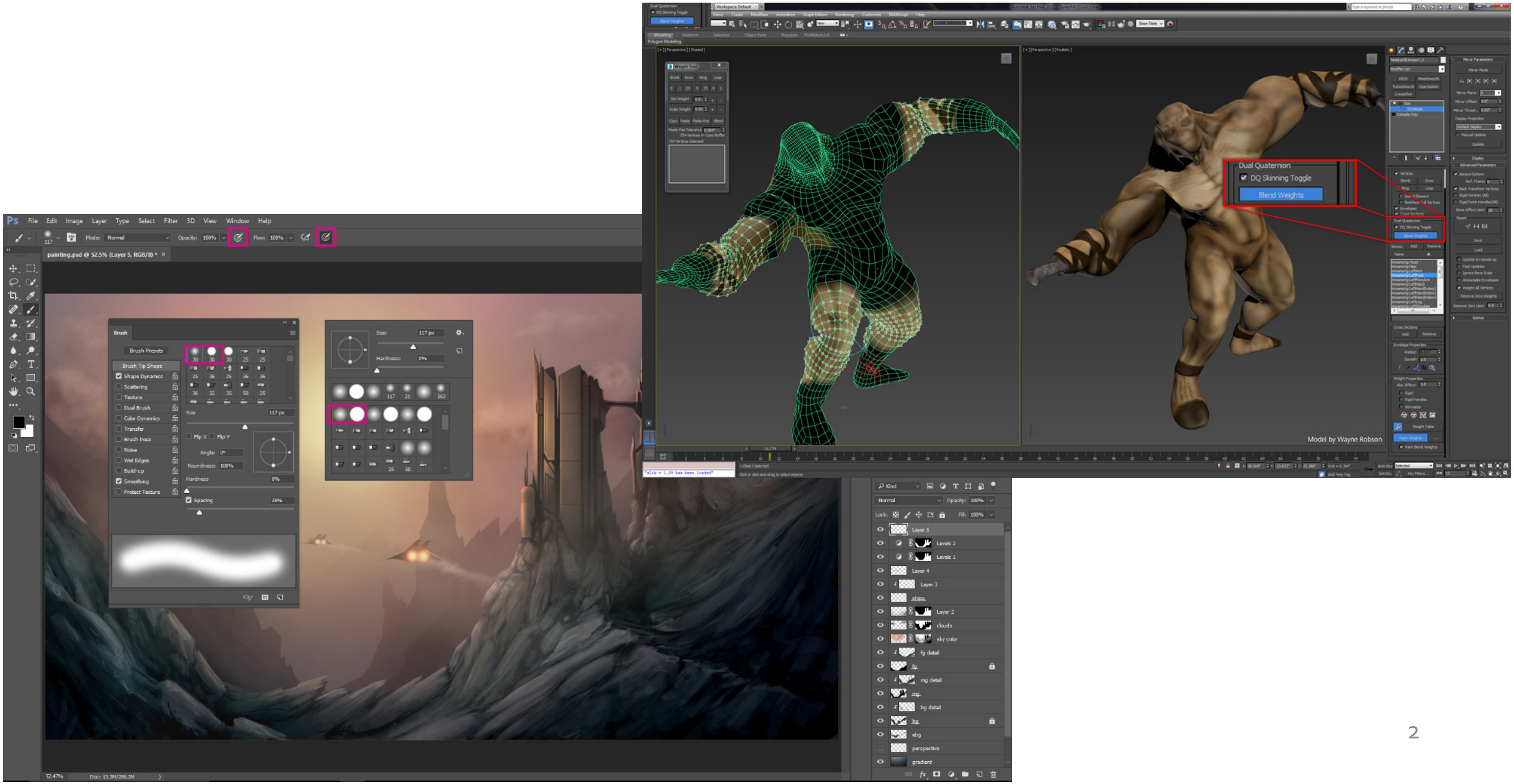
COMP3271 Computer Graphics

Introduction

2019-20

What is Computer Graphics?

It’s NOT about using commercial graphics software



What is Computer Graphics?

From Wikipedia:

- **Computer graphics** is a sub-field of Computer Science which studies methods for digitally synthesizing and manipulating visual content.
- **Computer graphics** studies the manipulation of visual and geometric information using computational techniques. It focuses on the mathematical and computational foundations of image generation and processing rather than purely aesthetic issues.

What is Computer Graphics?

Algorithms for visual simulation

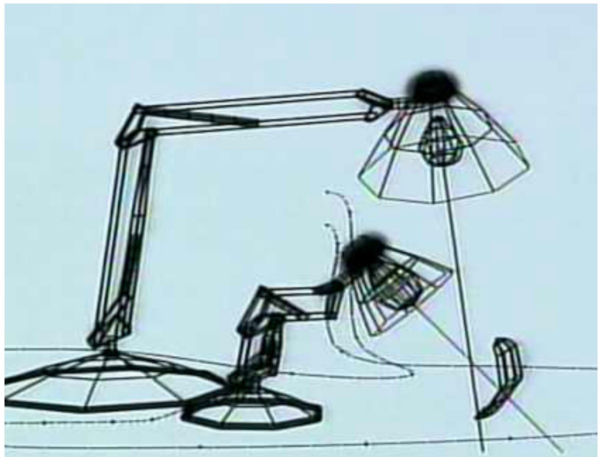
How to come up with a CG product?

Pixar Animation – Luxo Jr. 1986 (https://youtu.be/D4NPQ8mfKUo)

Three Main Themes of Computer Graphics

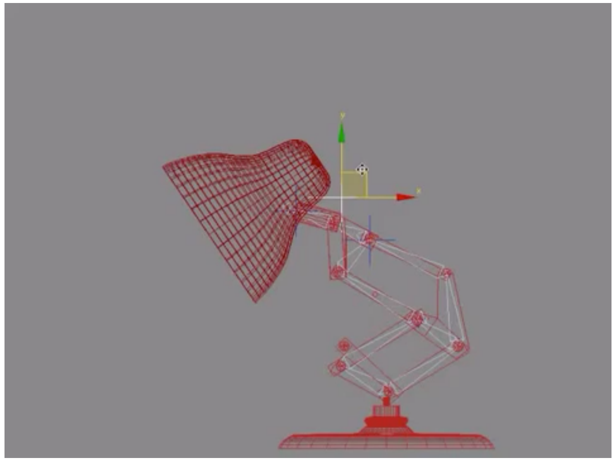
Modeling

- How do we represent (or model) 3-D objects?
- How do we construct models for specific objects?



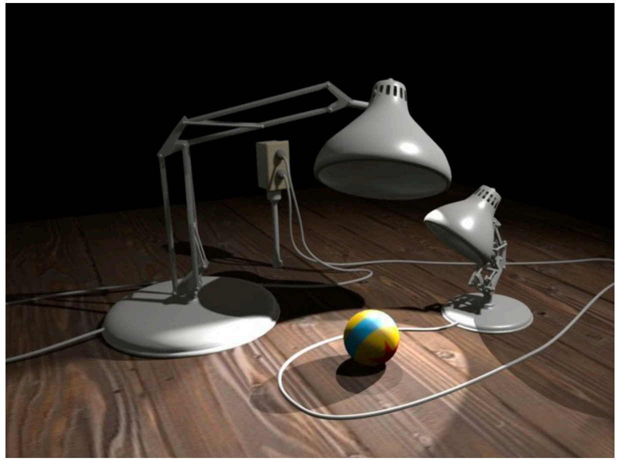
Animation

- How do we generate the motion of objects?
- How do we give animators control of this motion?



Rendering

- How do we simulate the formation of images?
- How do we simulate real-world light-material interaction?



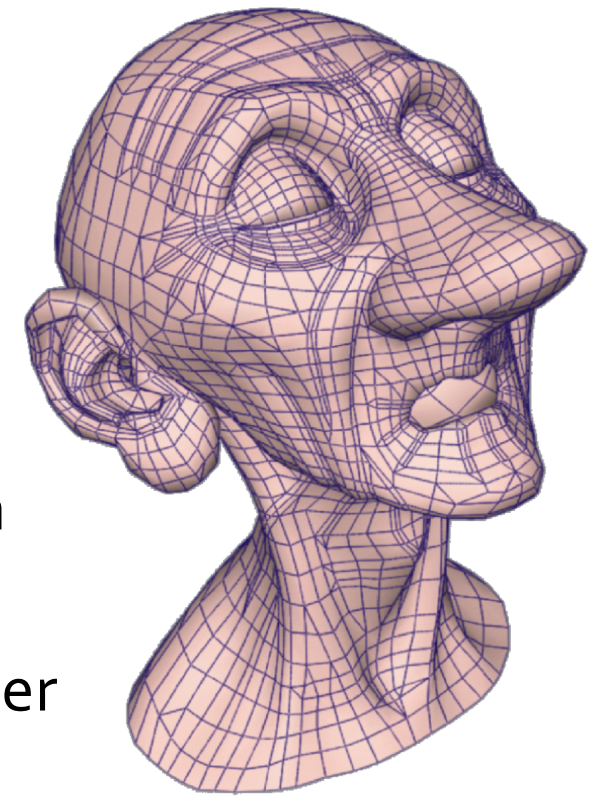
Modeling

How do we represent objects/environments?

- shape : the geometry of the object
- appearance : emission, reflection, and transmission of light

How do we construct these models?

- manual description (e.g., write down a formula)
- interactive manipulation
- procedurally : write a generating program (e.g., fractals)
- scan a real object (laser scanners, computer vision, ...)



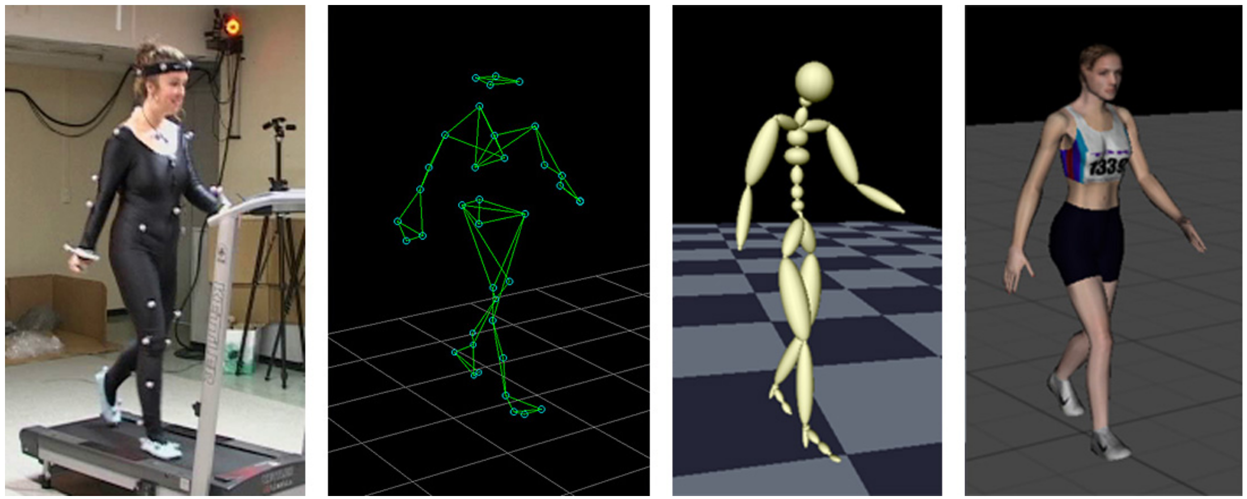
Animation

How do we represent the motion of objects?

- positions, angles, etc. as functions of time

How do we control/specify this motion?

- generate poses by hand, fill in with keyframing
- behavioral simulation (program little “brains” for objects)
- physical simulation
- motion capture



Rendering

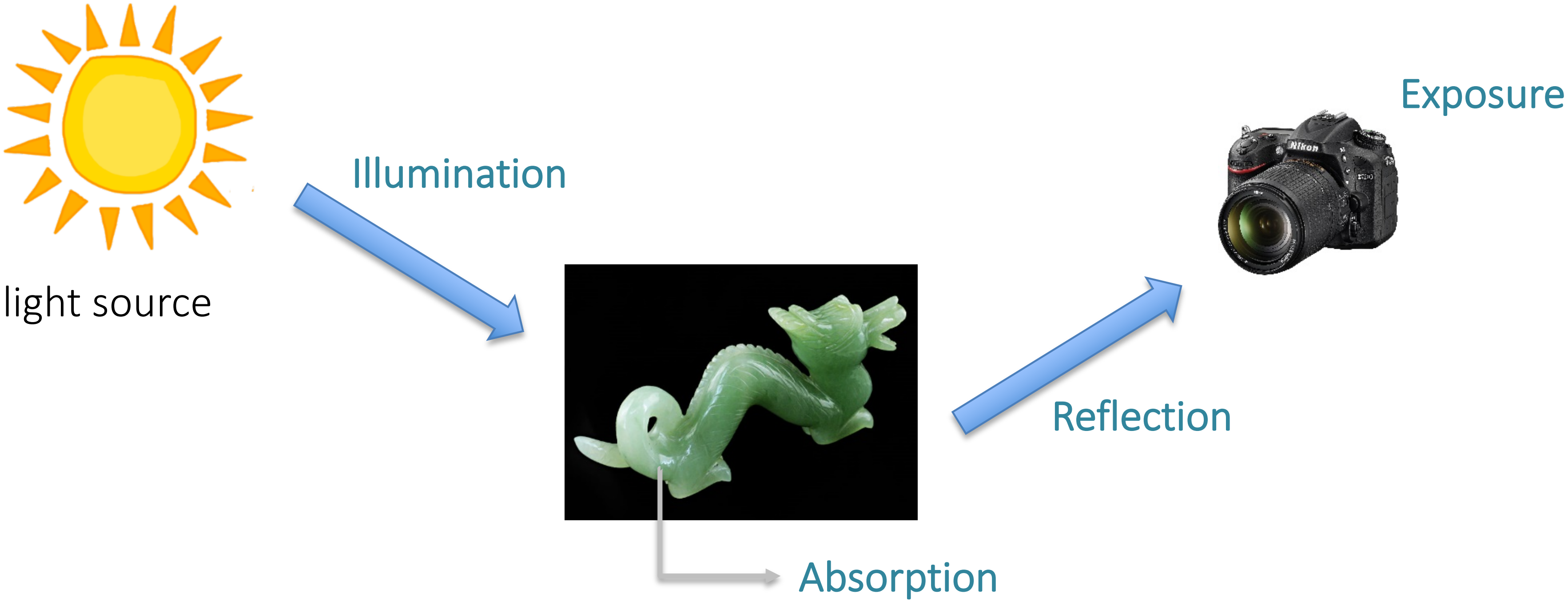
How do we simulate the formation of images?

- incoming light is focused by a lens
- light energy "exposes" a light-sensitive "film"
- represent images as discrete 2-D arrays of pixels

How do we simulate the behavior of light?

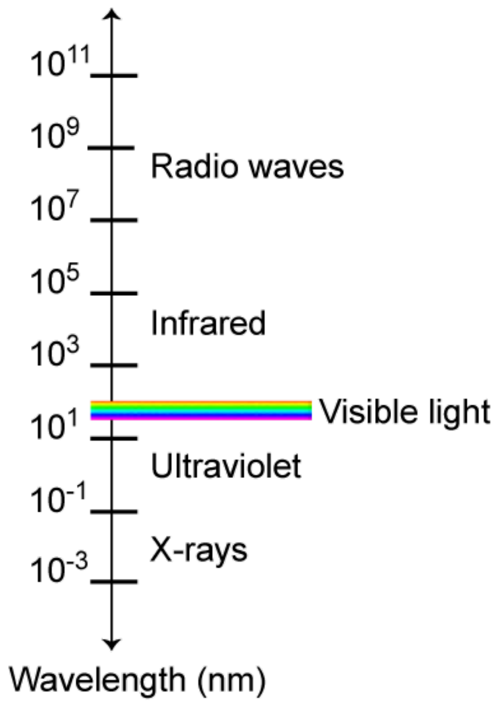
- consider light as photons (light particles)
- trace straight-line motion of photons
- must model interactions when light hits surfaces

Image Formation at a Glance



This is **light transport**.
Illumination is generated at light sources, propagates through the world, and interacts with objects in scene.

What is Light?



A form of electromagnetic (EM) radiation

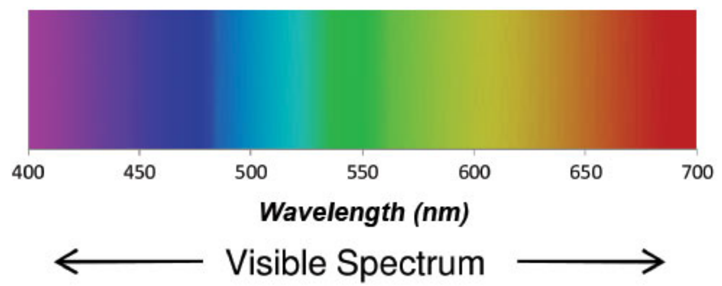
- like x-rays, microwaves, radio waves, …
- characterized by wavelength λ
- amplitude of wave determines intensity

We perceive limited section of the spectrum (visible light spectrum)

- each wavelength is a specific color

higher frequency
shorter wavelength

UV



IR

lower frequency
longer wavelength

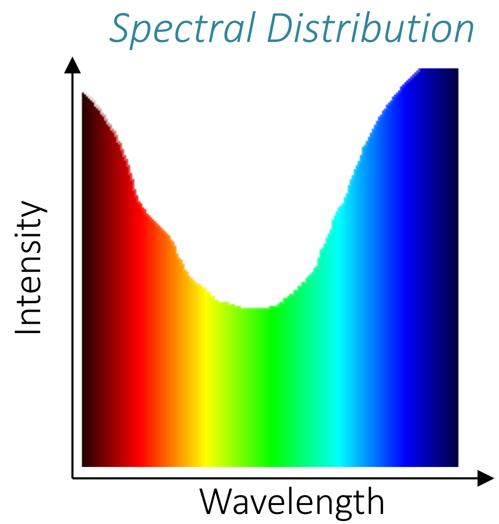
Composition of Illumination

Generally a mixture of many wavelengths

- each as some intensity
- spectral distribution: intensity as a function of wavelength over the entire spectrum

We perceive these distributions as colors

Sample Color



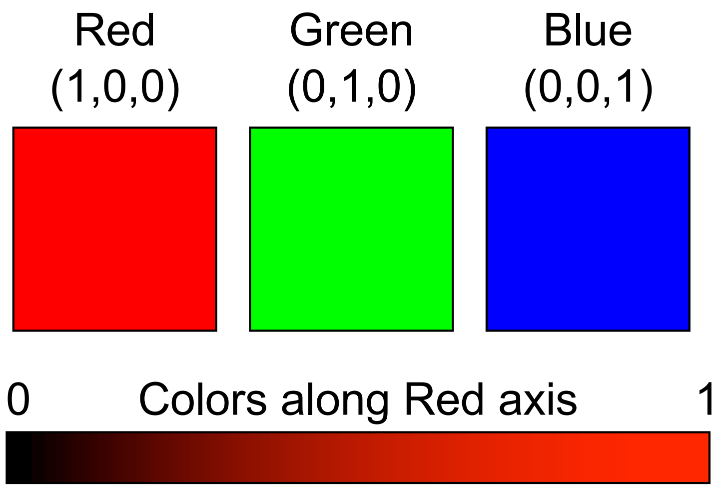
Basic Color Representation in Graphics

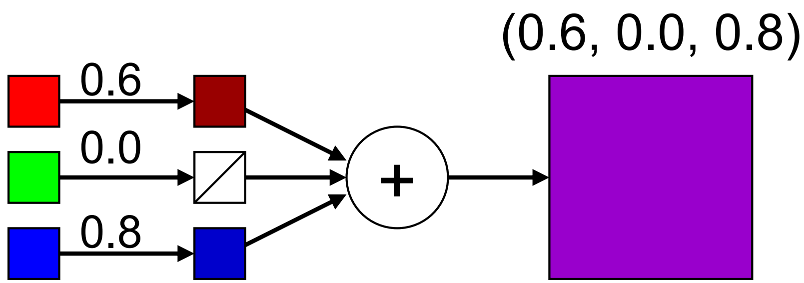
A spectral distribution $I(\lambda)$ is a continuous representation

- inconvenient for use on computers

We treat colors as a 3-D space of (r, g, b) triples

- all colors will be composed from three primary colors: red, green, blue
- the value of each (r, g, b) is between 0 and 1
- coefficients represent relative contribution of each primary

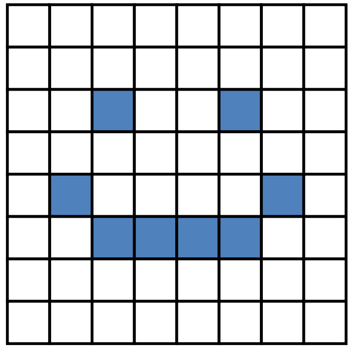


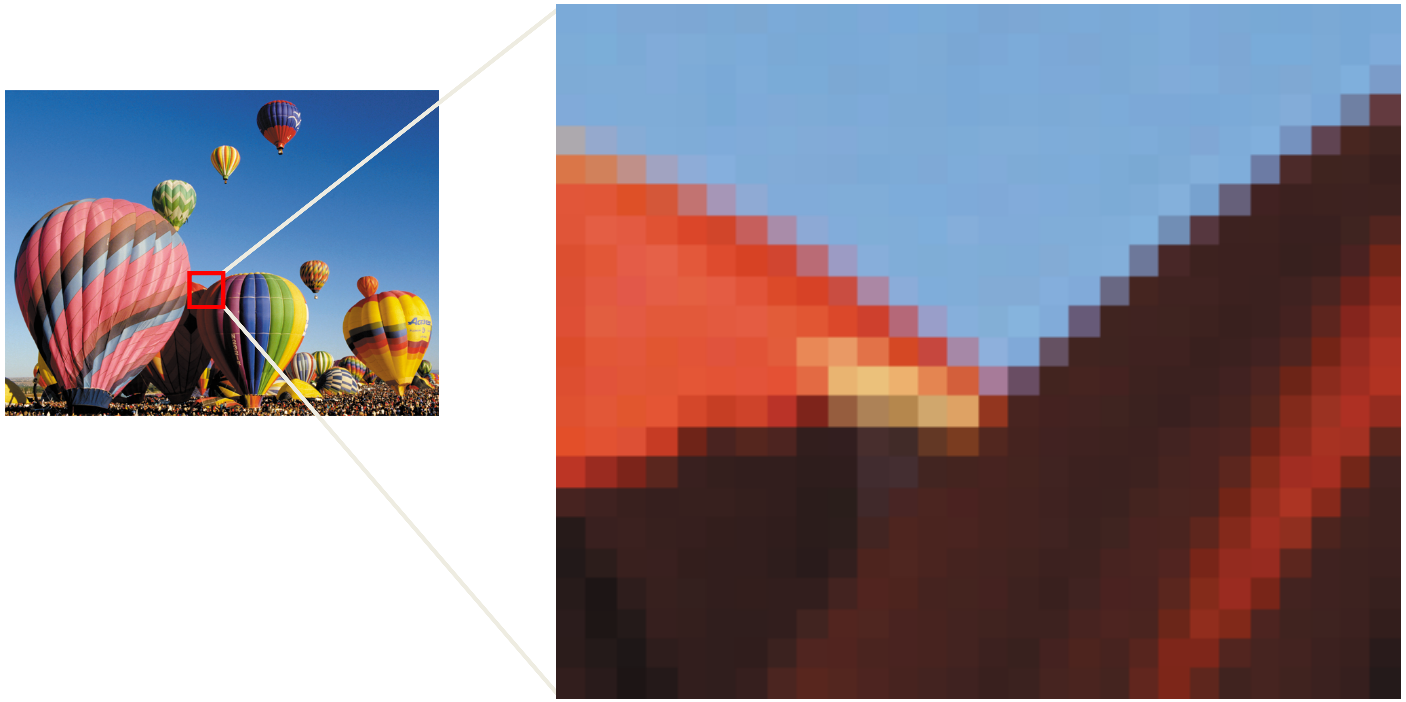


Raster Image Representation

We want a convenient discrete representation of images

- as with color, distribution of light energy over film plane is continuous
- instead, we'll carve up image into a rectangular grid of pixels ***P*[*x*,*y*]**
- each pixel ***p*** will store a color value
  - RGB triple for color images
  - single value for grayscale (or monochrome) images





Raster Image Representation

We want a convenient discrete representation of images

- as with color, distribution of light energy over film plane is continuous
- instead, we'll carve up image into a rectangular grid of pixels ***P*[*x*,*y*]**
- each pixel ***p*** will store a color value
  - RGB triple for color images
  - single value for grayscale (or monochrome) images

Can separate an RGB color image into 3 distinct color channels

- each by itself is a monochrome image

Important Application Areas

Entertainment

- Movie production (special effects & full-length features)
- Computer games

Industrial design

- CAD
- automated machining

Training

Visualization

- scientific datasets
- medical scans (e.g., x-rays, MRI)
- architectural prototyping

Online distribution

State-of-the-Art Research

Conferences

- ACM SIGGRAPH & SIGGRAPH Asia
- Eurographics
- ACM CHI (Conference on Human Factors in Computing Systems)
- IEEE VIS
- ...

Journals

- ACM Transactions on Graphics
- IEEE Transactions on Visualization and Computer Graphics
- ...