# Computer Graphics

Szymon Rusinkiewicz

Princeton University

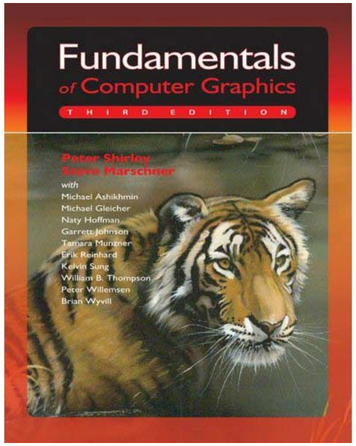
COS 426, Spring 2011

# Overview

- Administrivia
  - People, times, places, etc.
- Syllabus
  - What will I learn in this course?
- Raster Graphics
  - Getting started …

# Administrative Matters

- Instructors
  - Szymon Rusinkiewicz
  - TA #1: Vladimir (Vova) Kim
  - TA #2: TBA
- Book
  - *Fundamentals of Computer Graphics*
    *Peter Shirley and Steve Marschner,*
    *Third Edition, A.K. Peters, July 2009,*
    *ISBN: 978-1568814698*




- Web page
  - http://www.cs.princeton.edu/cos426

# Questions / Discussion

- Experimenting with Piazzza (www.piazzza.com) to handle question/answer and general help
- Somewhere between Wiki and newsgroups…
- Use this instead of email to instructors/TAs
- Will set it up for everyone enrolled as of today

# Coursework

- Programming Assignments (50%)
  - Assignment #0: C++ programming / HTML / dropbox
  - Assignment #1: Image Processing
  - Assignment #2: Mesh Processing
  - Assignment #3: Ray Tracing
  - Assignment #4: Particle System Animation
- Exams (25%)
  - In class (Mar 10 and Apr 28)
- Final Project (25%)
  - Video game!
  - Completed in groups of 2-4

# Programming Assignments

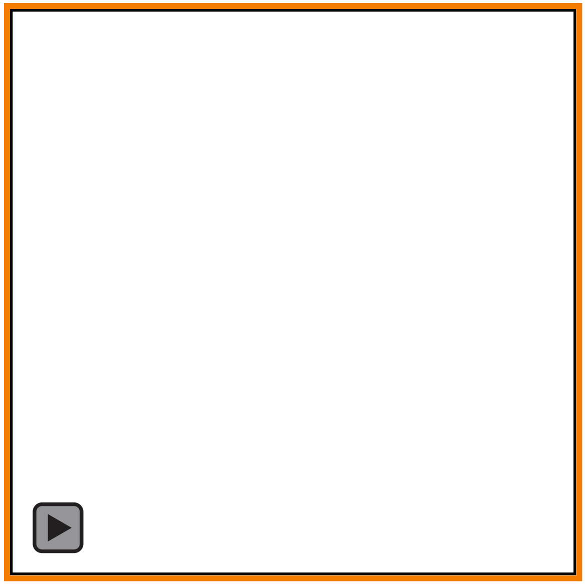
- When?
  - Roughly every 2-3 weeks
- Where?
  - Anywhere you want, e.g. home or Friend 017 lab
- How?
  - C++ (Precept this week; install compiler now!)
  - Interactive rendering APIs: OpenGL, GLUT
- What?
  - Basic feature lists
  - Optional features
  - Art contest

# Art Contest

- Everybody should submit entries!
  - 1 point for submitting
  - 2 points for winning

Cool Images

*(James Percy, CS 426, Fall99)*



Videos

*(Phil Wei, CS 426, Spr04)*

Bloopers

*(Alex Combs, CS 426, Spr05)*

# Collaboration Policy

- Overview:
  - You must write your own code (no credit for other code)
  - You must reference your sources of any ideas/code
- It's OK to …
  - Talk with other students about ideas, approaches, etc.
  - Get ideas from information in books, web sites, etc.
  - Get "support" code from example programs
    - » But, you must reference your sources
- It's NOT OK to …
  - Share code with another student
  - Use ideas or code acquired from another sources without attribution

# Precepts

- Schedule?
  - Friday 1:30 – 2:30
  - Friday 3:00 – 4:00
  - Other?

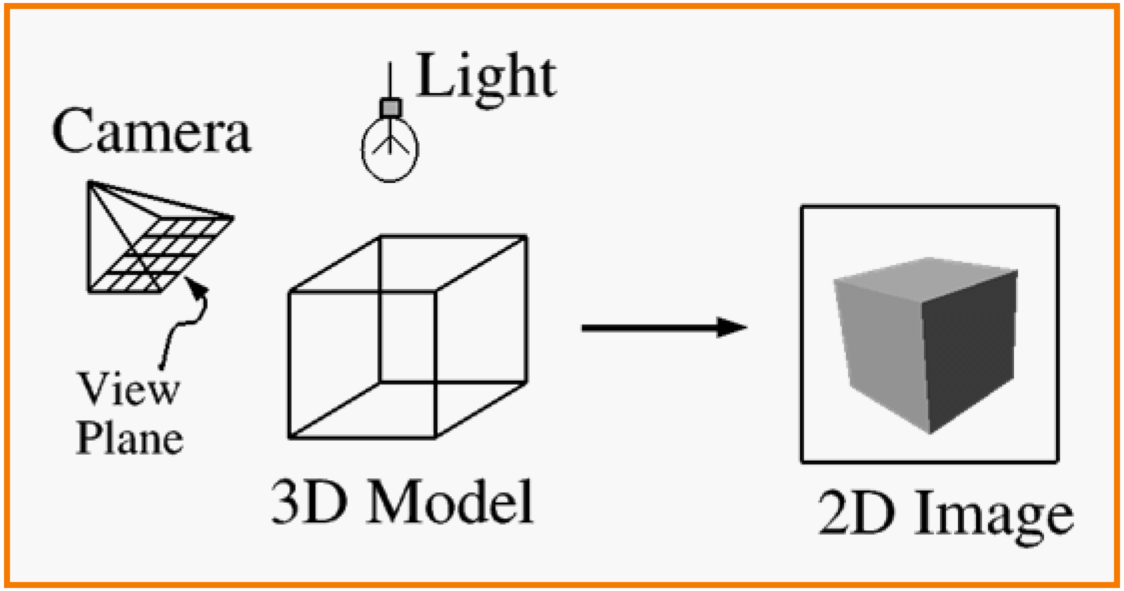
- Place?
  - TBA

# Overview

- Administrivia
  - People, times, places, etc.
- ➢ Syllabus
  - What will I learn in this course?
- Raster Graphics
  - Getting started …

# Introduction

- What is computer graphics?
  - Imaging = *representing 2D images*
  - Modeling = *representing 3D objects*
  - Rendering = *constructing 2D images from 3D models*
  - Animation = *simulating changes over time*



# Syllabus

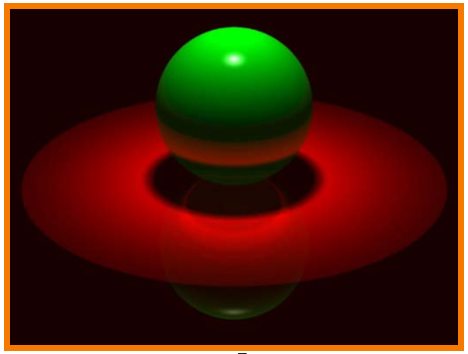
I. Image processing

II. Modeling

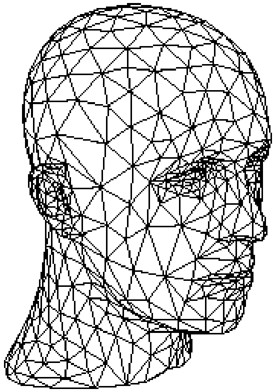
III. Rendering

IV. Animation

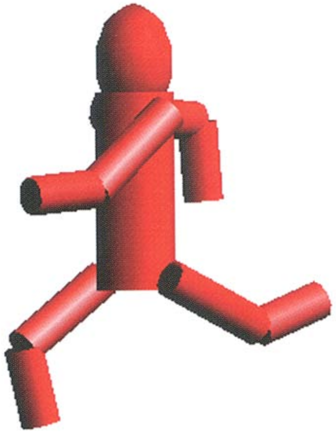
Image Processing
*(Rusty Coleman, CS426, Fall99)*



Rendering
*(Michael Bostock, CS426, Fall99)*



Modeling
*(Dennis Zorin, CalTech)*



Animation
*(Angel, Plate 1)*

# Part I: Image Processing

- Raster Graphics
  - Display devices
  - Color models
- Image Representation
  - Sampling
  - Reconstruction
  - Quantization & Aliasing
- Image Processing
  - Filtering
  - Warping
  - Composition
  - Morphing

Image Composition
*(Michael Bostock, CS426, Fall99)*

Image Morphing
*(All students in CS 426, Fall98)*

# Part II: Modeling

- Representations of geometry
  - Curves: splines
  - Surfaces: meshes, splines, subdivision
  - Solids: voxels
- Procedural modeling
  - Sweeps
  - Fractals
  - Grammars

Scenery Designer
*(Dirk Balfanz, Igor Guskov, Sanjeev Kumar, & Rudro Samanta, CS426, Fall95)*

Shell
*(Douglas Turnbull, CS 426, Fall99)*

# Part III: Rendering

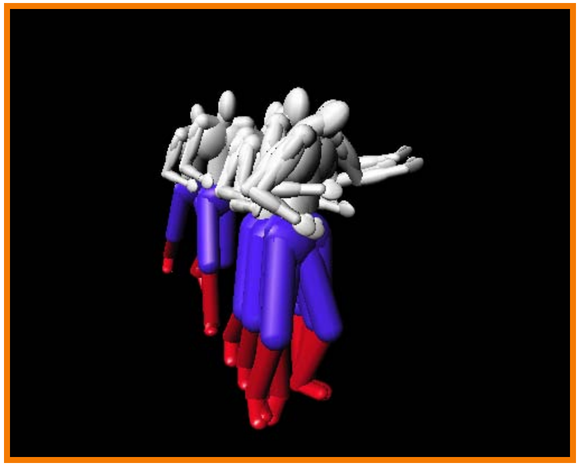
- Interactive 3D Pipeline
  - Modeling transformations
  - Viewing transformations
  - Hidden surface removal
  - Illumination, shading, and textures
  - Scan conversion, clipping
  - Hierarchical scene graphics
  - OpenGL
- Global illumination
  - Ray tracing
  - Radiosity

Pixel Shading
*(Final Fantasy, Square Pictures)*

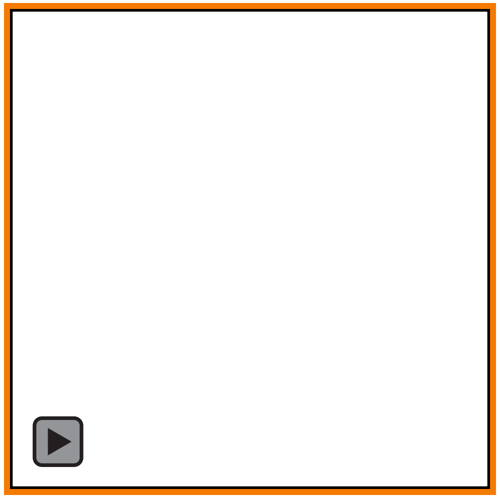
Ray Tracing
*(Sid Kapur, CS 426, Spr04)*

# Part IV: Animation

- Keyframing
  - Kinematics
  - Articulated figures
- Motion capture
  - Capture
  - Warping
- Dynamics
  - Physically-based simulations
  - Particle systems
- Behaviors
  - Planning, learning, etc.



Dancing Guy
*(Jon Beyer, CS426, Spr05)*



Ice Queen
*(Mao Chen, Zaijin Guan, Zhiyan Liu, & Xiaohu Qie, CS426, Fall98)*

# Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- Computer art

# Applications

- **Entertainment**
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- Computer art

Up
*(Pixar Animation Studios)*

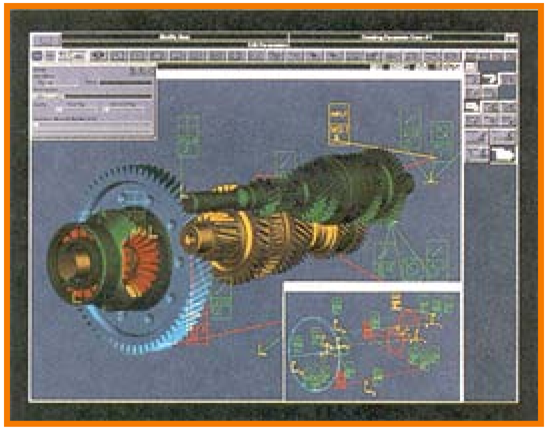
Jurassic Park
*(Industrial, Light, & Magic)*

Halo
*(Bungie)*

# Applications

- Entertainment
- **Computer-aided design**
- Scientific visualization
- Training
- Education
- E-commerce
- Computer art

Los Angeles Airport
*(Bill Jepson, UCLA)*

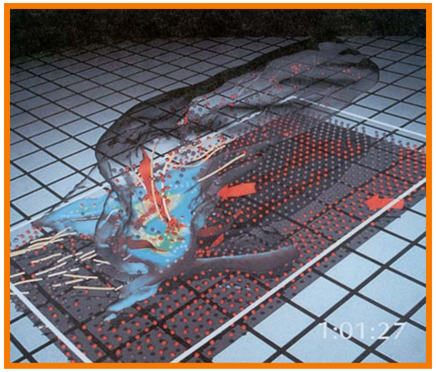


Gear Shaft Design
*(Intergraph Corporation)*

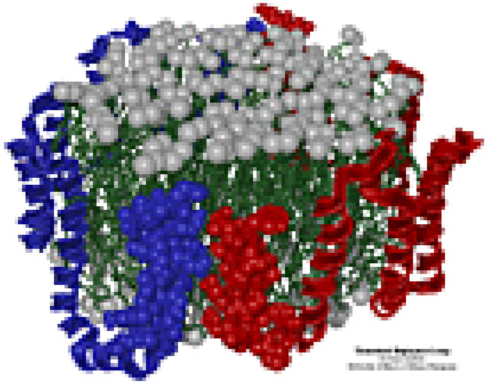
Boeing 777 Airplane
*(Boeing Corporation)*

# Applications

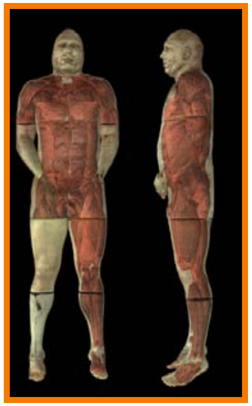
- Entertainment
- Computer-aided design
- **Scientific visualization**
- Training
- Education
- E-commerce
- Computer art




Airflow Inside a Thunderstorm
*(Bob Wilhelmson,*
*University of Illinois at Urbana-Champaign)*



Apo A-1
*(Theoretical Biophysics Group,*
*University of Illinois at Urbana-Champaign)*

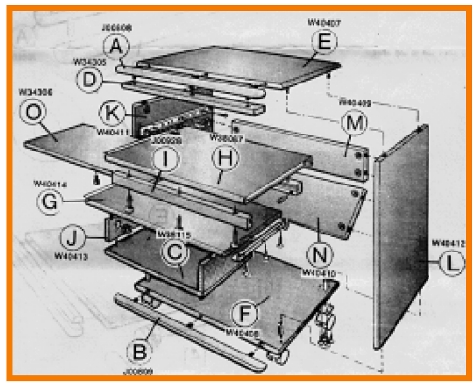


Visible Human
*(National Library of Medicine)*

# Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- **Training**
- Education
- E-commerce
- Computer art

Driving Simulation
*(Evans & Sutherland)*




Desk Assembly
*(Silicon Graphics, Inc.)*

Flight Simulation
*(NASA)*

# Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- → **Education**
- E-commerce
- Computer art

Forum of Trajan
*(Bill Jepson, UCLA)*

Human Skeleton
*(SGI)*

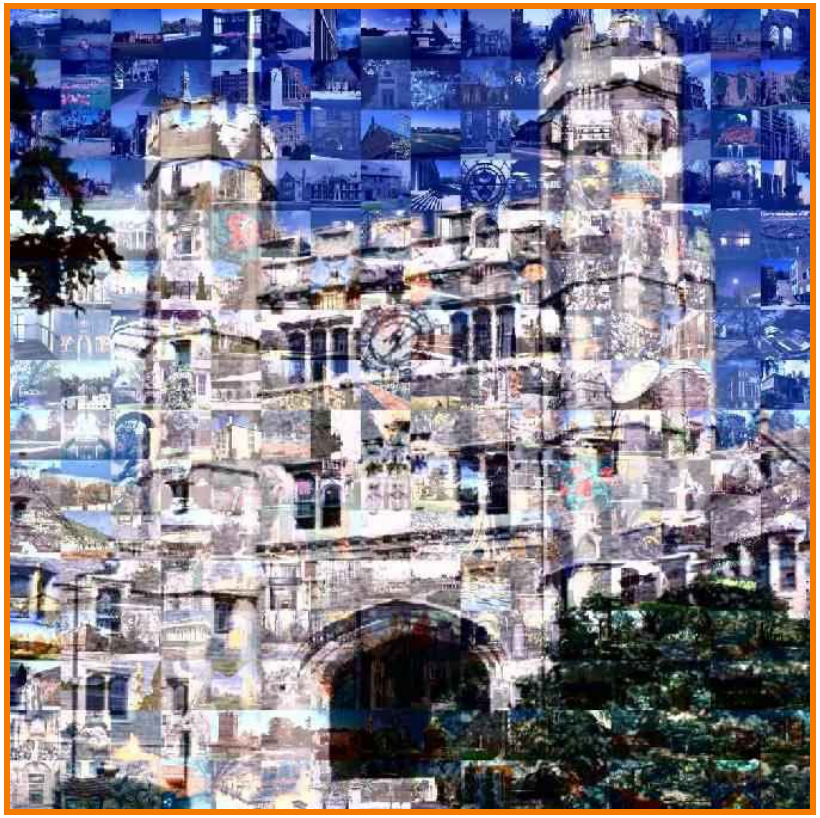
# Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- → **E-commerce**
- Computer art


Virtual Footwear Wall
*(Intel)*

# Applications

- Entertainment
- Computer-aided design
- Scientific visualization
- Training
- Education
- E-commerce
- ➔ **Computer art**



Blair Arch

*(Marissa Range '98)*

# Overview

- Administrivia
  - People, times, places, etc.
- Syllabus
  - What will I learn in this course?
- Raster Graphics
  - Let's get started …

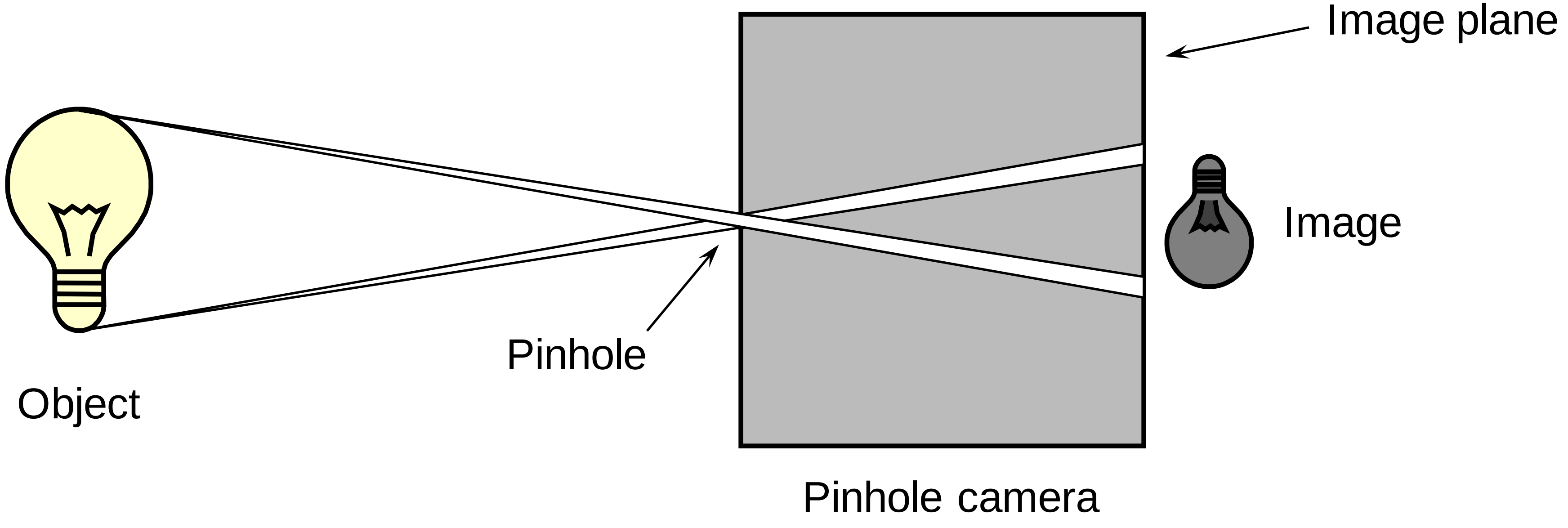
# Raster Graphics

- Images
  - What is an image?
  - How are images displayed?
- Colors
  - What is a color?
  - How do we perceive colors?
  - How do we represent colors in a computer?

# What is an Image?

# What is an Image?

- Amount of light as a function of direction, flowing through an ideal camera




Points on image plane
↔ directions of light

# What is a Digital Image?

# What is a Digital Image?

- Sampled representation of a continuous image…
- Stored as a 2D rectilinear array of ***pixels***

Continuous image

Digital image

# What is a Digital Image?

A Pixel is a Sample, not a Little Square!

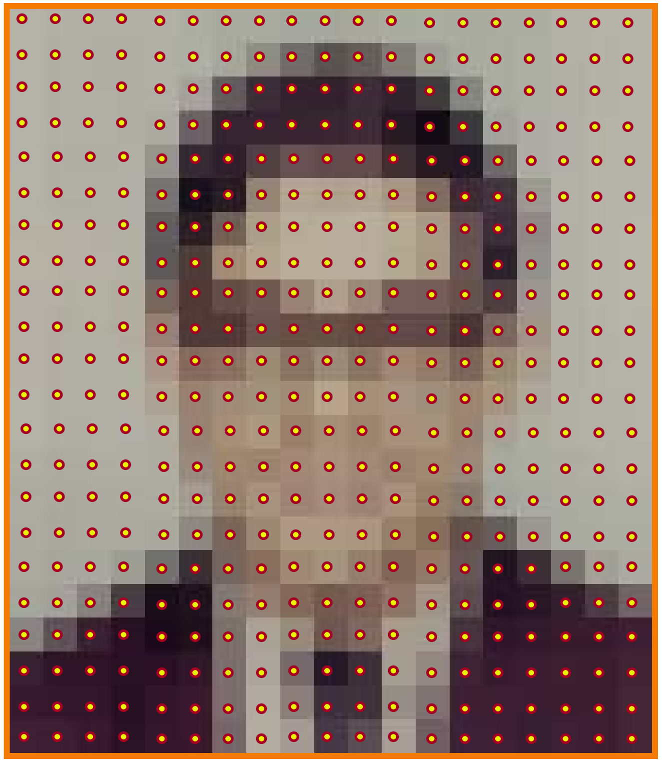
Continuous image

Digital image

# What is a Digital Image?

A Pixel is a Sample, not a Little Square!

Continuous image



Digital image

# What is a Digital Image?

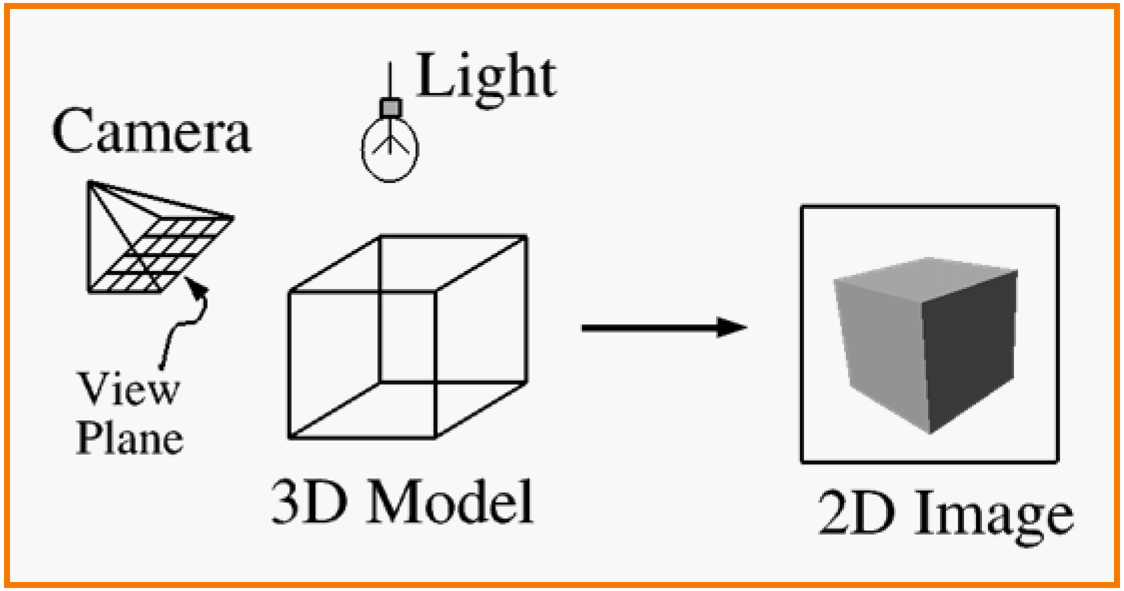
A Pixel is a Sample, not a Little Square!

Continuous image

Digital image

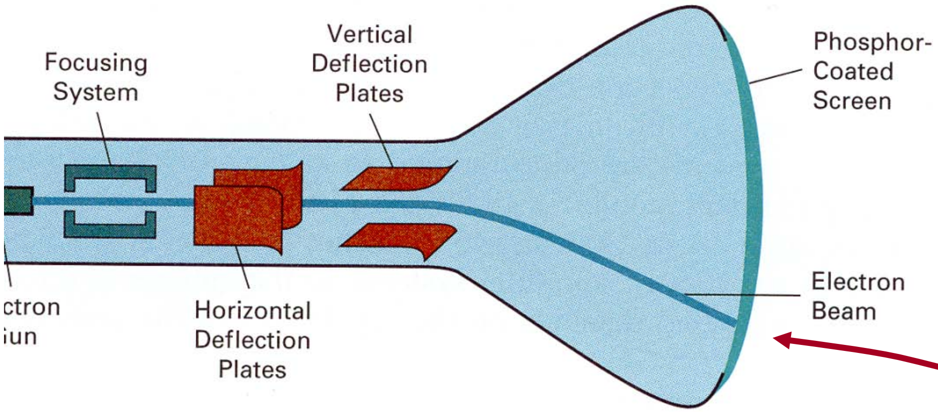
# Image Acquisition

- Pixels are samples from continuous function
  - Photoreceptors in eye
  - CCD cells in digital camera
  - Rays in virtual camera



# Image Display

- Re-create continuous function from samples
  - Example: cathode ray tube



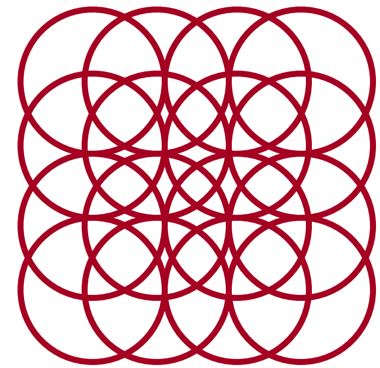



Image is reconstructed by displaying pixels with finite area (Gaussian)

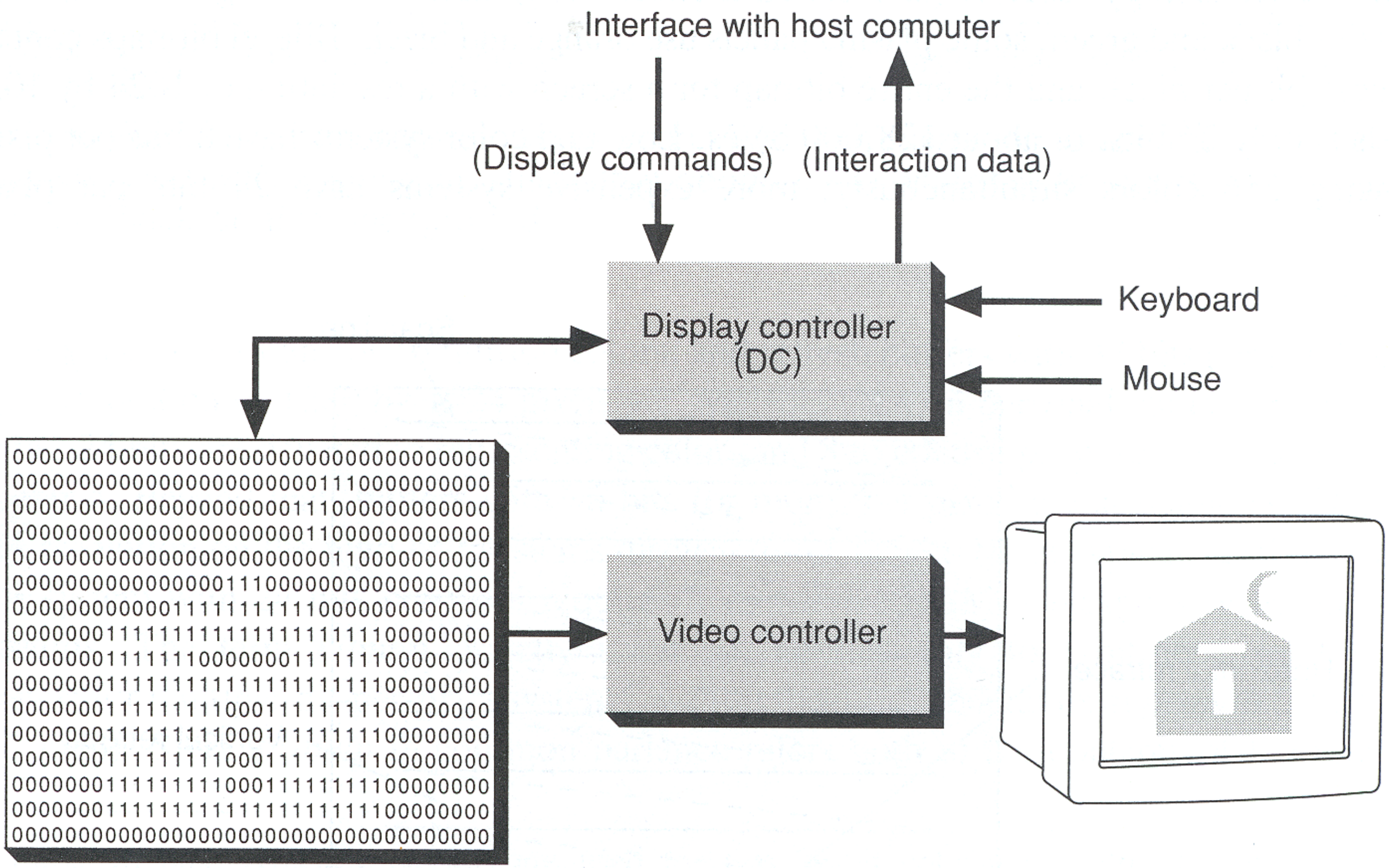
# Image Resolution

- Intensity resolution
  - Each pixel has only “Depth” bits for colors/intensities
- Spatial resolution
  - Image has only “Width” x “Height” pixels
- Temporal resolution
  - Monitor refreshes images at only “Rate” Hz

Typical Resolutions

| | Width x Height | Depth | Rate |
|---|---|---|---|
| NTSC | 640 x 480 | 8 | 30 |
| Workstation | 1280 x 1024 | 24 | 75 |
| Film | 3000 x 2000 | 12 | 24 |
| Laser Printer | 6600 x 5100 | 1 | - |

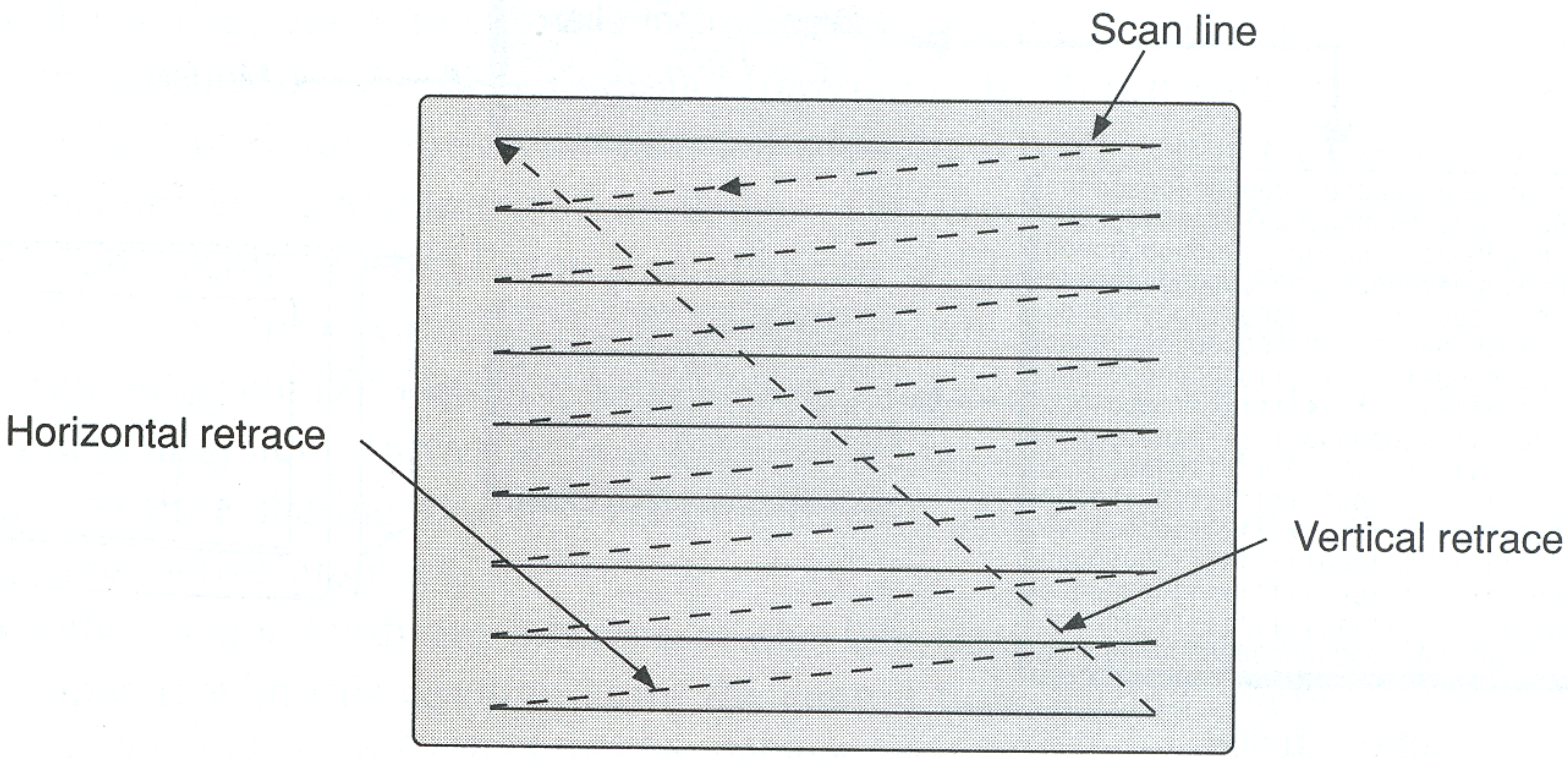
# Frame Buffer




Frame Buffer

Figure 1.2 from FvDFH

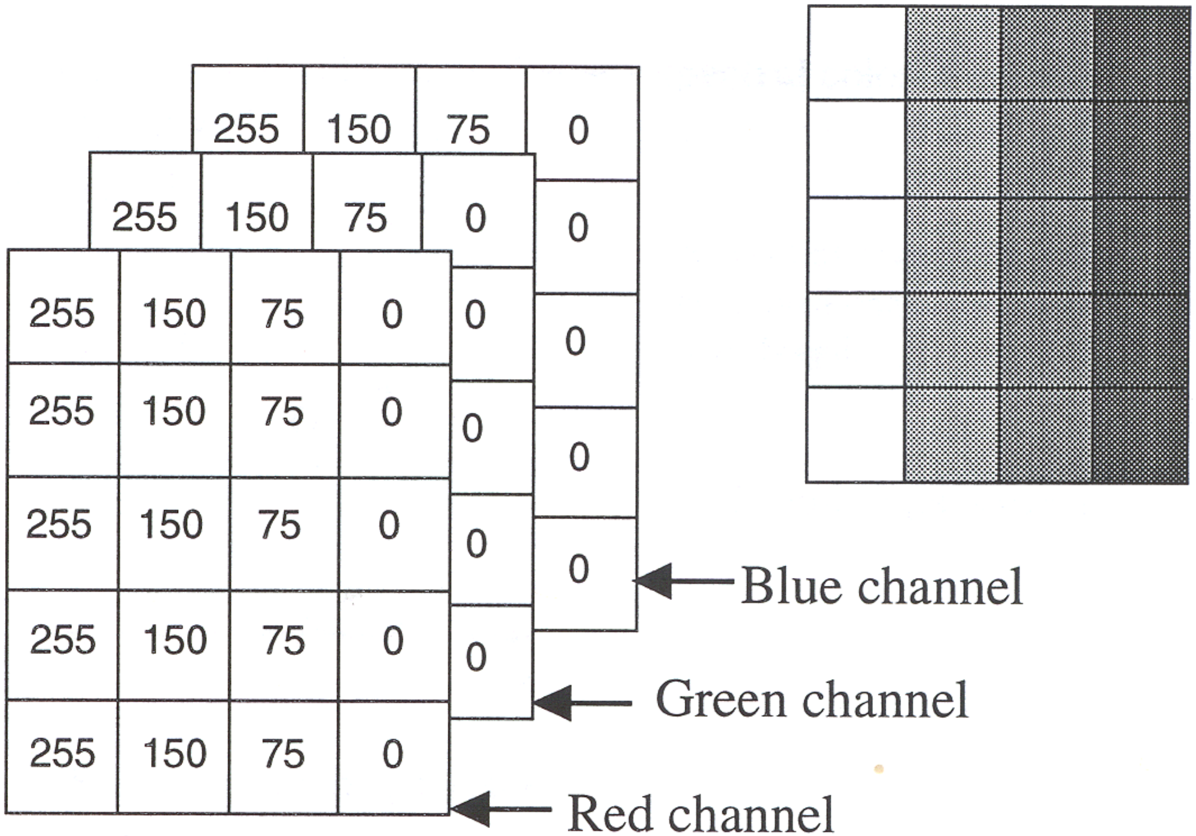
# Frame Buffer Refresh




Refresh rate is usually 60-75Hz

Figure 1.3 from FvDFH

# Color Frame Buffer



# Color CRT

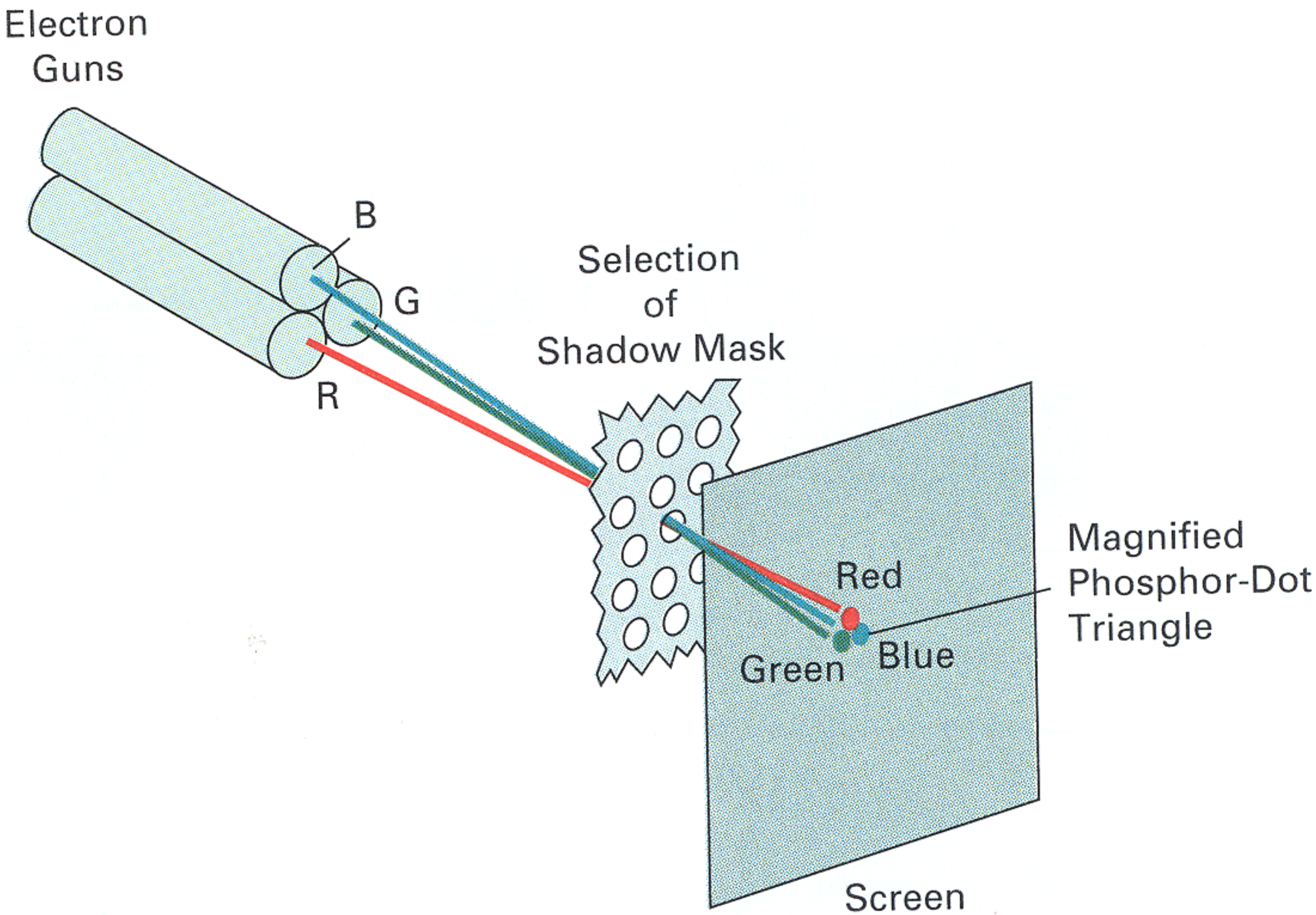
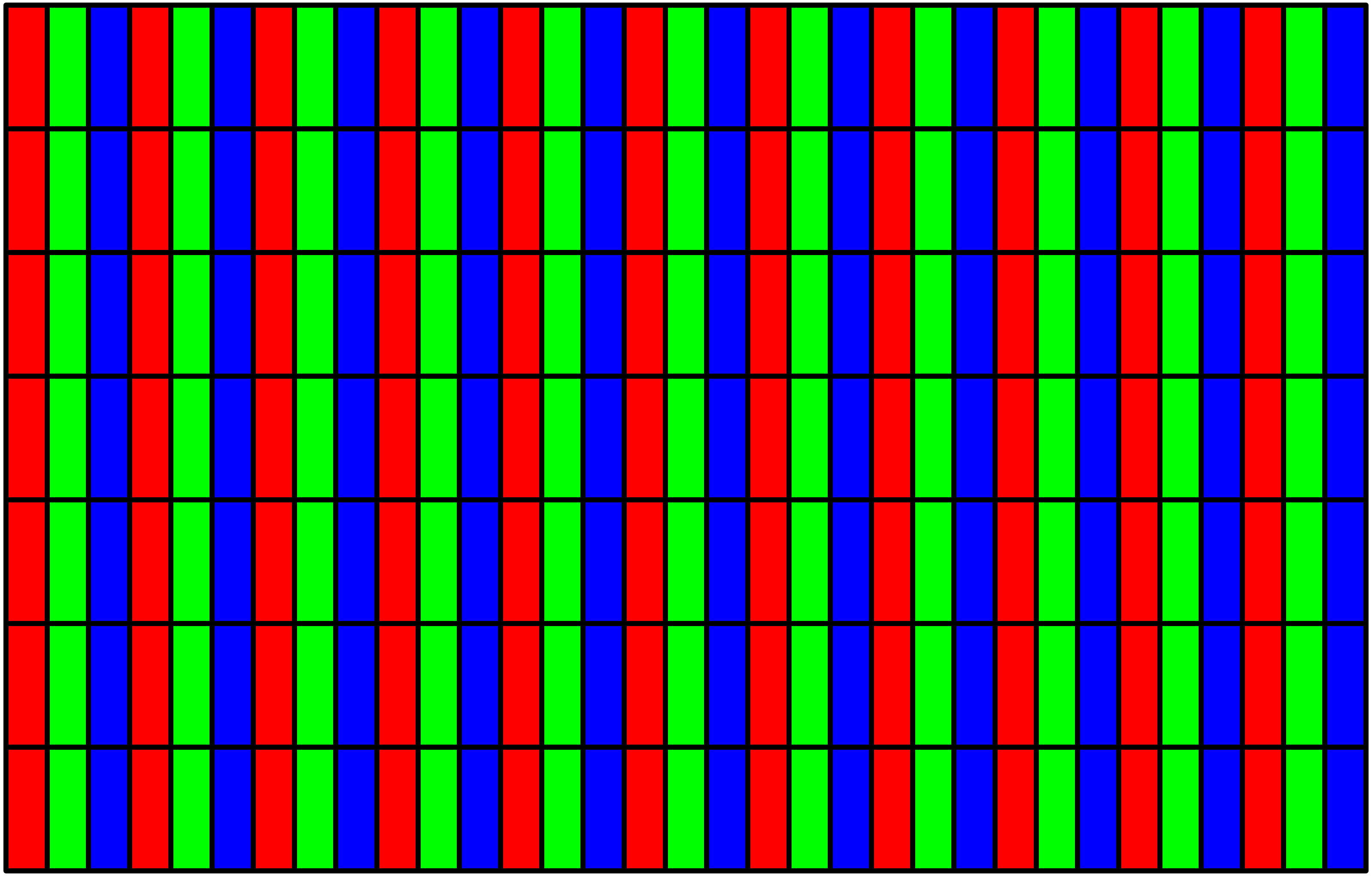



Figure 2.8 from H&B

# Color LCD



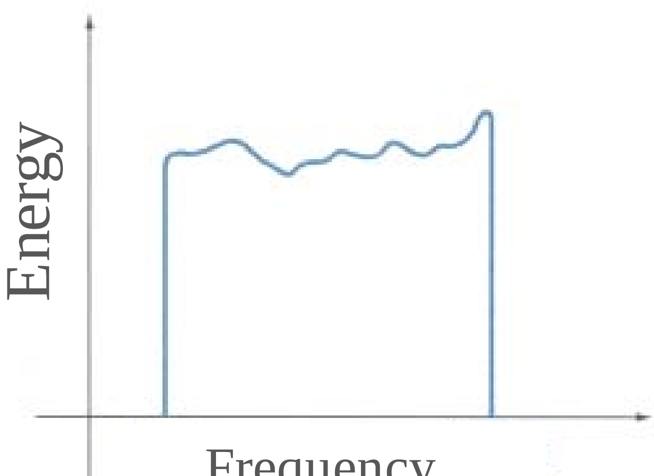
# Raster Graphics

- Images
  - What is an image?
  - How are images displayed?
- Colors
  - What is a color?
  - How do we perceive colors?
  - How do we represent colors in a computer?

# What is a Color?

# What is a Color?

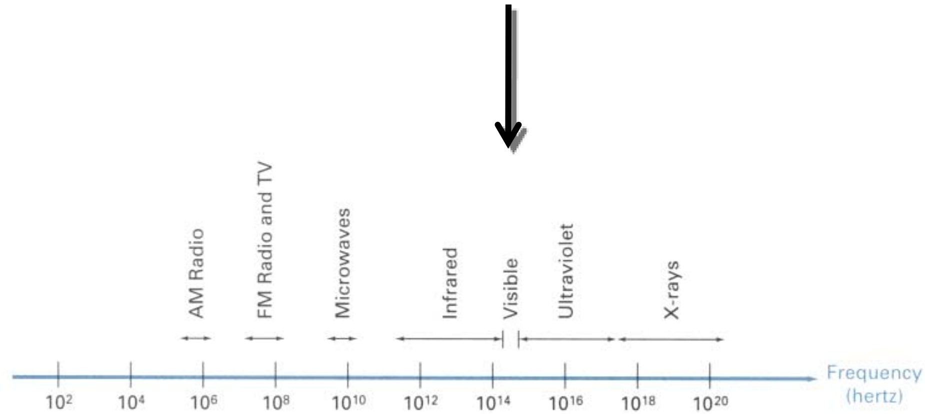
- One definition is a distribution of energies amongst frequencies in the visible light range




Red
$4.3 \times 10^{14}$ Hz
(700nm)

Violet
$7.5 \times 10^{14}$ Hz
(400nm)

Visible Light

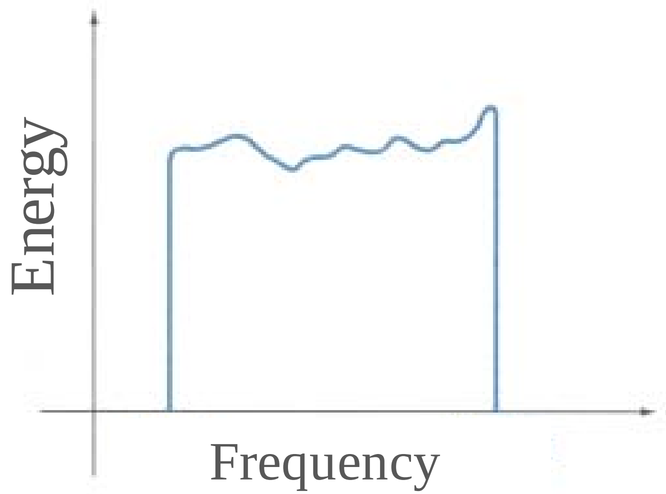



Frequency

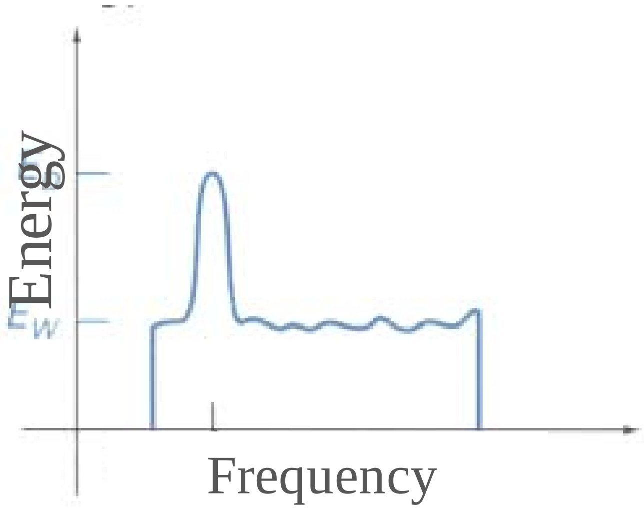
Figures 15.1,3 from H&B

# Visible Light

- The color of light is characterized by …
  - Hue = dominant frequency (highest peak)
  - Lightness = luminance (area under curve)
  - Saturation = excitation purity (ratio of highest to rest)




White Light



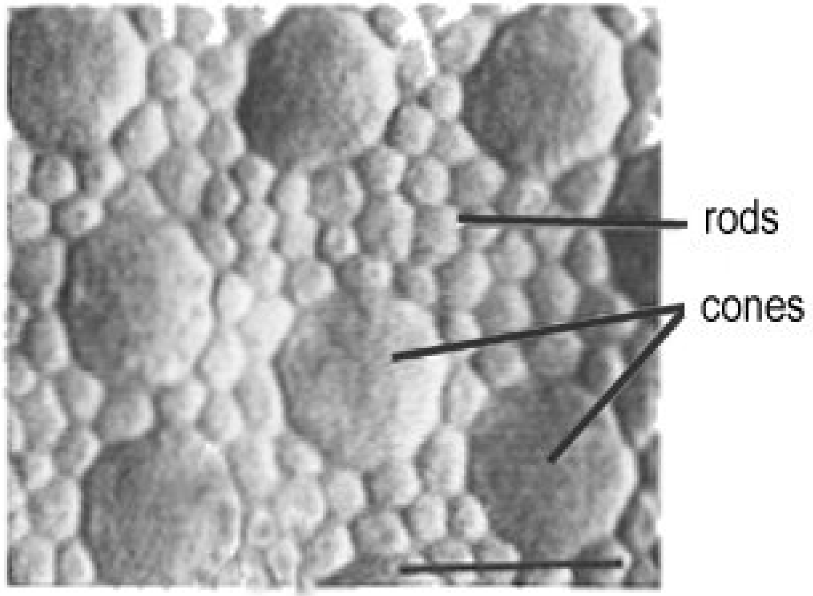

Orange Light

Figures 15.3-4 from H&B

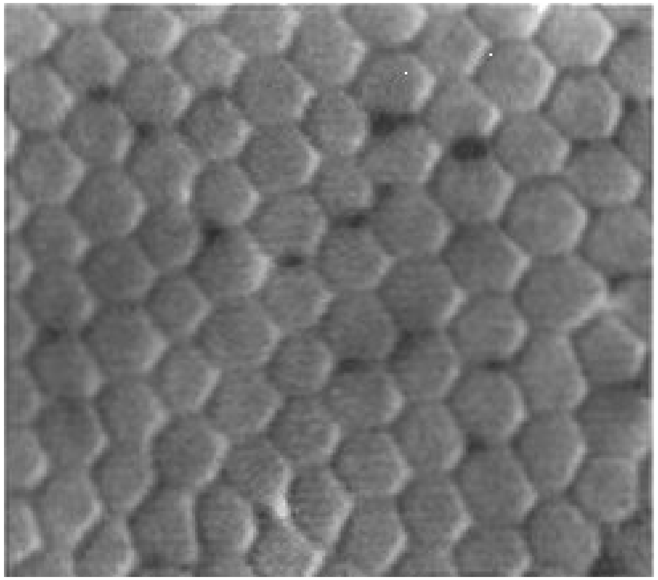
# How Do We Perceive Color?

# Modern Understanding of Color

- Two types of receptors: rods and cones

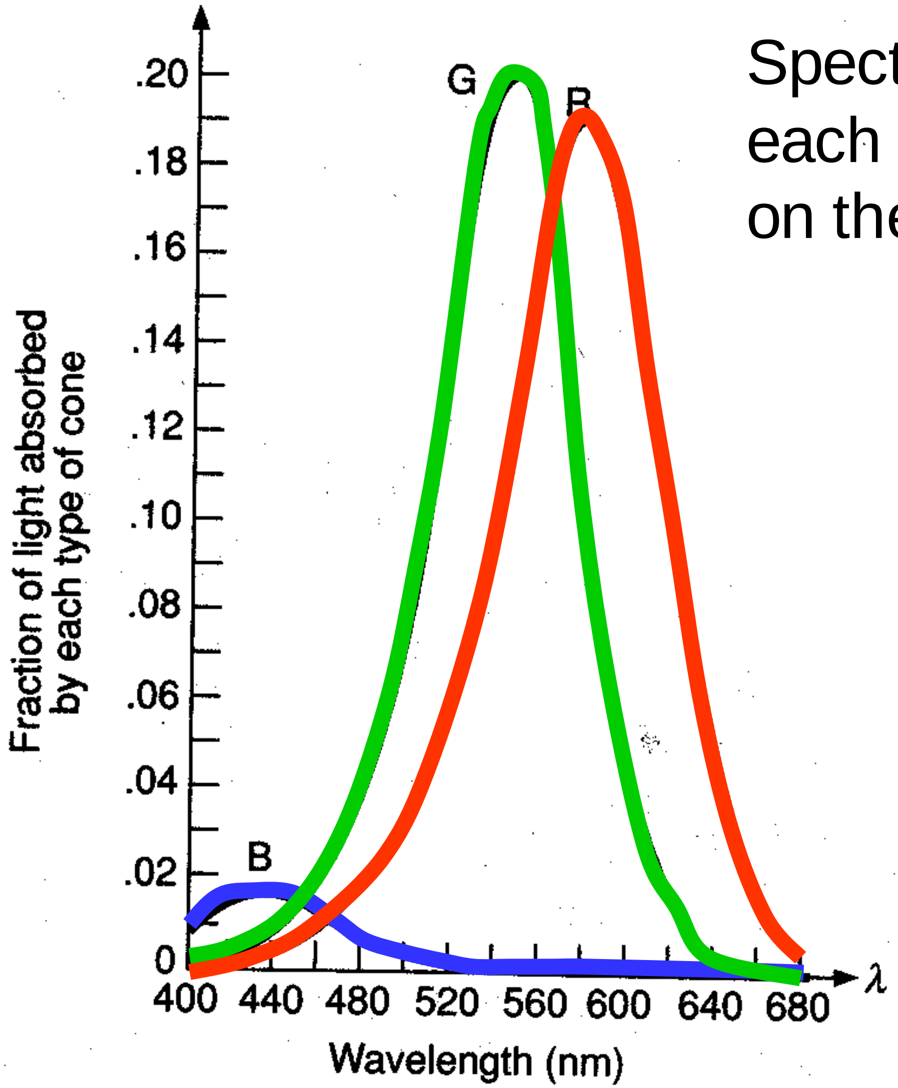



Rods and cones



Cones in fovea

# Color Perception




Spectral-response functions of each of the three types of cones on the human retina.

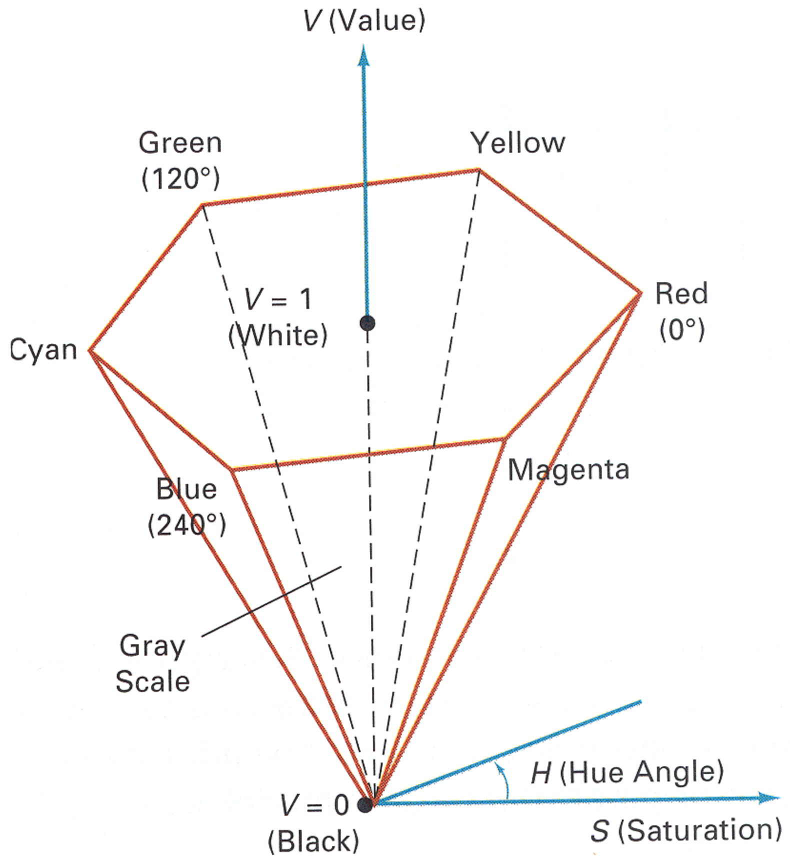
Figure 13.18 from FvDFH

# Representing Colors in a Computer

- Common color models
  - HLS
  - HSV
  - RGB
  - XYZ
  - CMY
  - Others

Tristimulus theory of color

# HLS & HSV Color Models

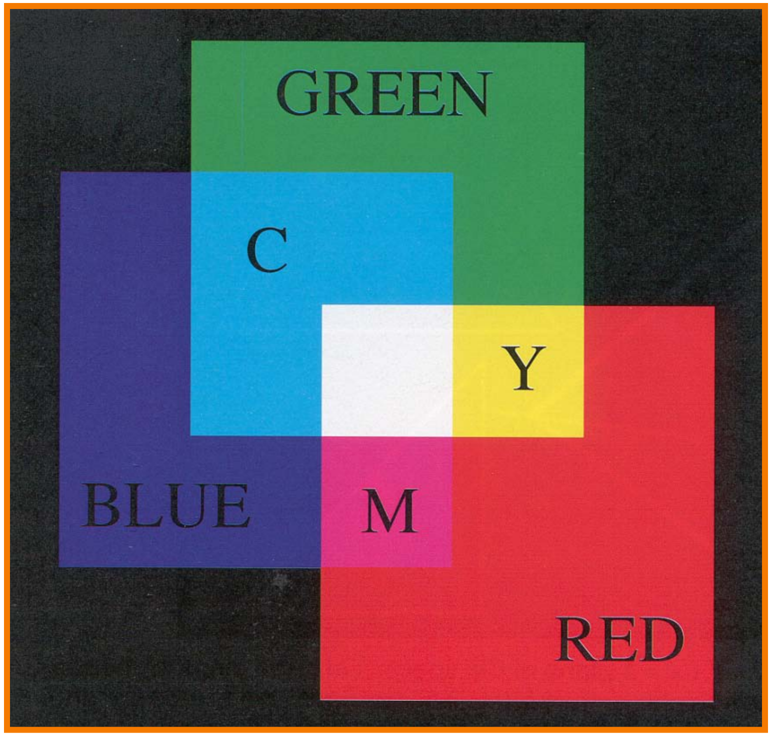



HSV Color Cone

| H | S | V | Color |
|---|---|---|---|
| 0 | 1.0 | 1.0 | Red |
| 120 | 1.0 | 1.0 | Green |
| 240 | 1.0 | 1.0 | Blue |
| * | 0.0 | 1.0 | White |
| * | 0.0 | 0.5 | Gray |
| * | * | 0.0 | Black |
| 60 | 1.0 | 1.0 | |
| 270 | 0.5 | 1.0 | |
| 270 | 0.0 | 0.7 | |

Figure 15.16&15.17 from H&B

# RGB Color Model

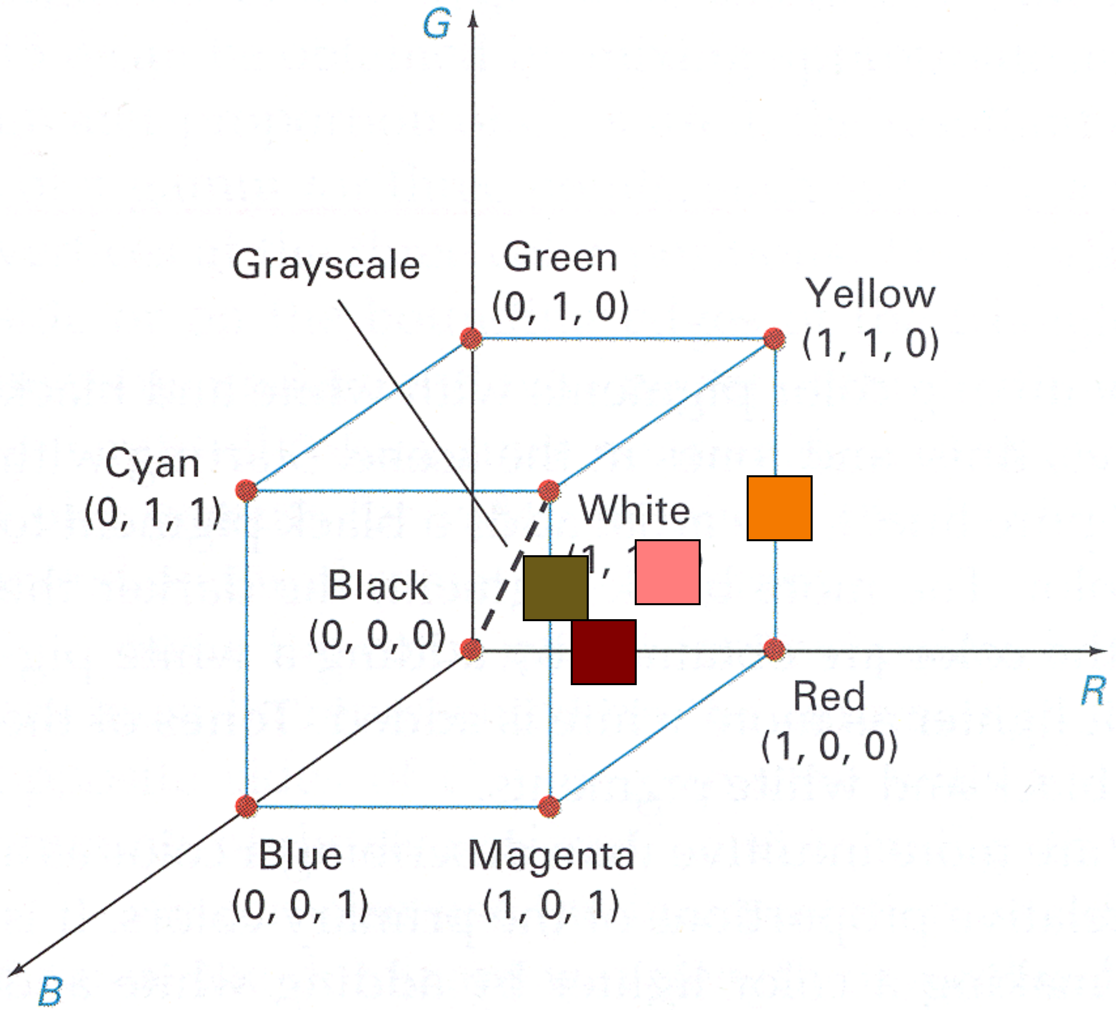


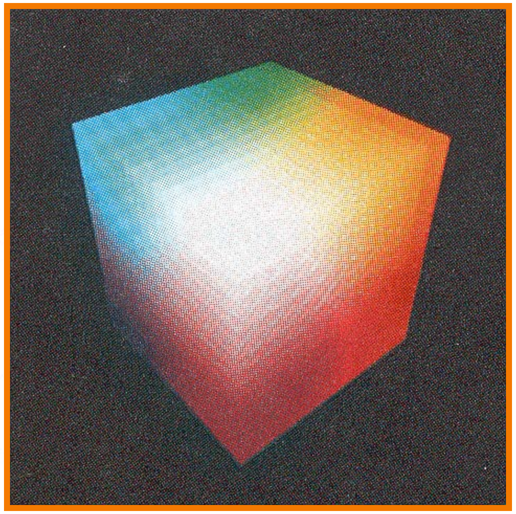

Colors are additive

| R | G | B | Color |
|---|---|---|---|
| 0.0 | 0.0 | 0.0 | Black |
| 1.0 | 0.0 | 0.0 | Red |
| 0.0 | 1.0 | 0.0 | Green |
| 0.0 | 0.0 | 1.0 | Blue |
| 1.0 | 1.0 | 0.0 | Yellow |
| 1.0 | 0.0 | 1.0 | Magenta |
| 0.0 | 1.0 | 1.0 | Cyan |
| 1.0 | 1.0 | 1.0 | White |
| 0.5 | 0.0 | 0.0 | ? |
| 1.0 | 0.5 | 0.5 | ? |
| 1.0 | 0.5 | 0.0 | ? |
| 0.5 | 0.3 | 0.1 | ? |

Plate II.3 from FvDFH

# RGB Color Cube



Figures 15.11&15.12 from H&B

# RGB Spectral Colors

Amounts of RGB primaries needed to display spectral colors

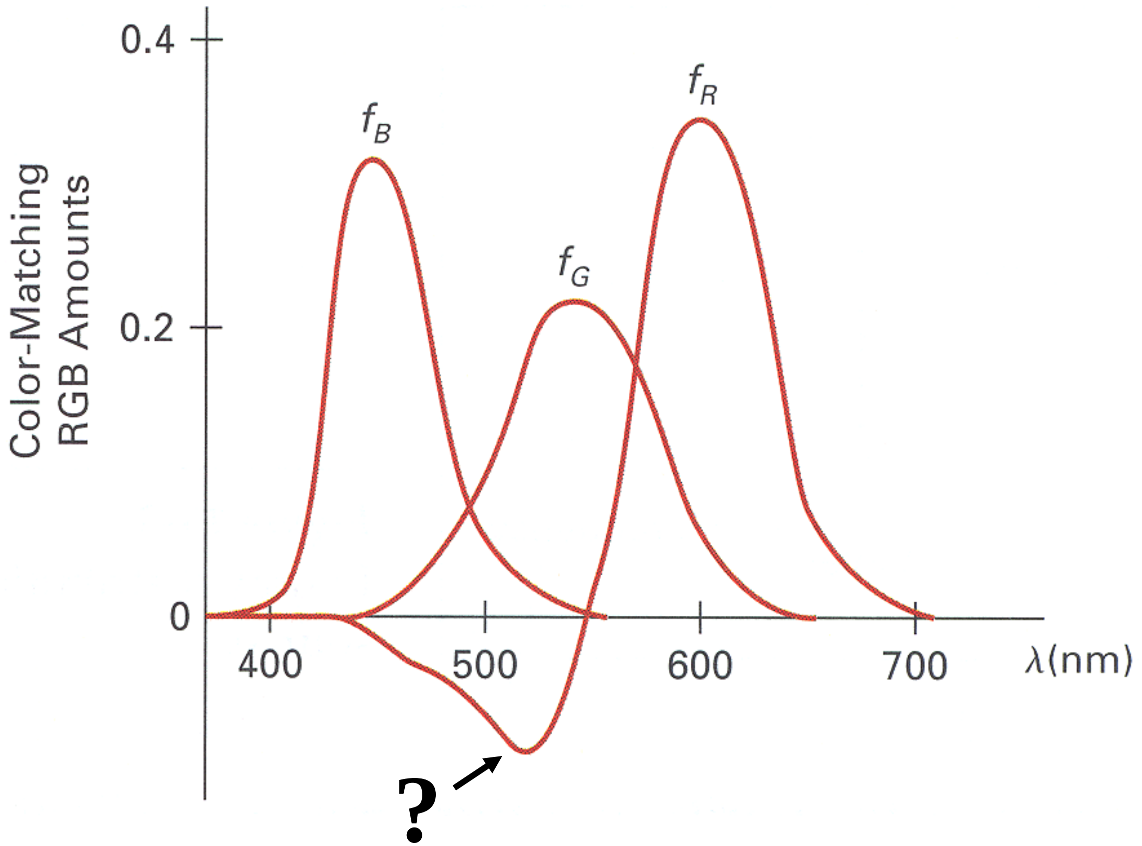



Figure 15.5 from H&B

# XYZ Color Model (CIE)

Amounts of CIE primaries needed to display spectral colors

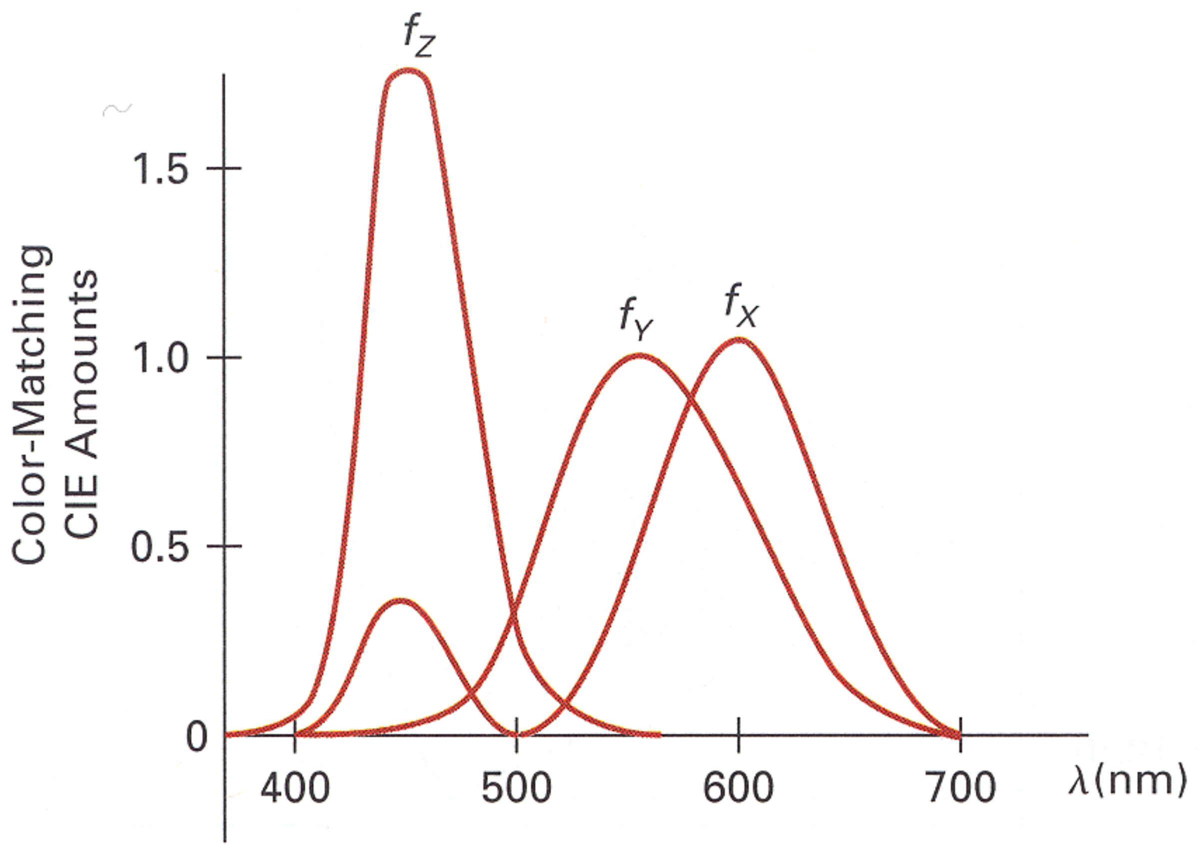



Figure 15.6 from H&B

# CIE Chromaticity Diagram

Normalized amounts of X and Y for colors in visible spectrum

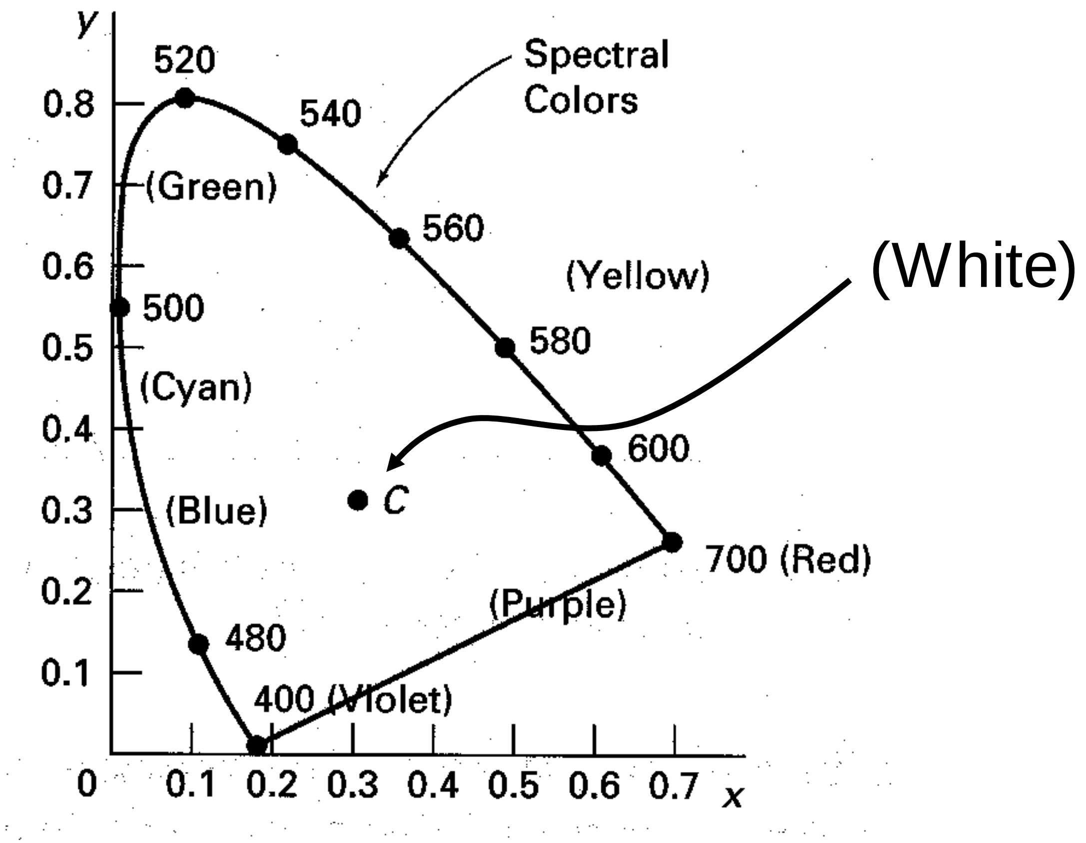
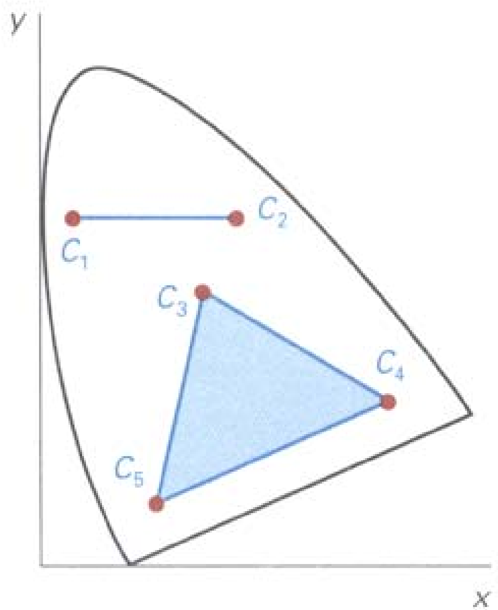
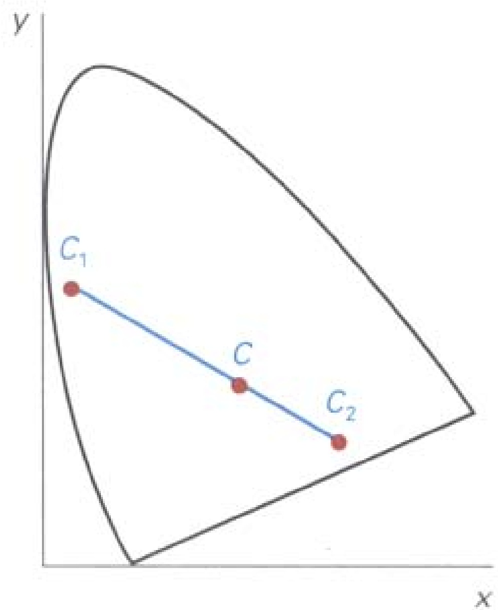



Figure 15.7 from H&B

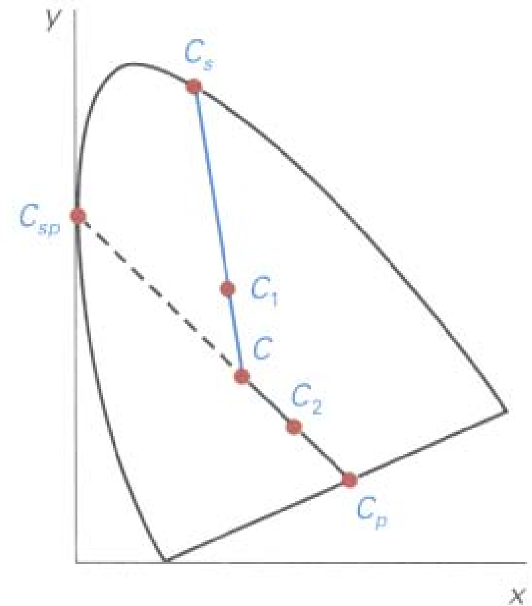
# CIE Chromaticity Diagram




Compare
Color
Gamuts




Identify
Complementary
Colors




Determine
Dominant Wavelength
and Purity

Figures 15.8-10 from H&B

# RGB Color Gamut

Color gamut for a typical RGB computer monitor

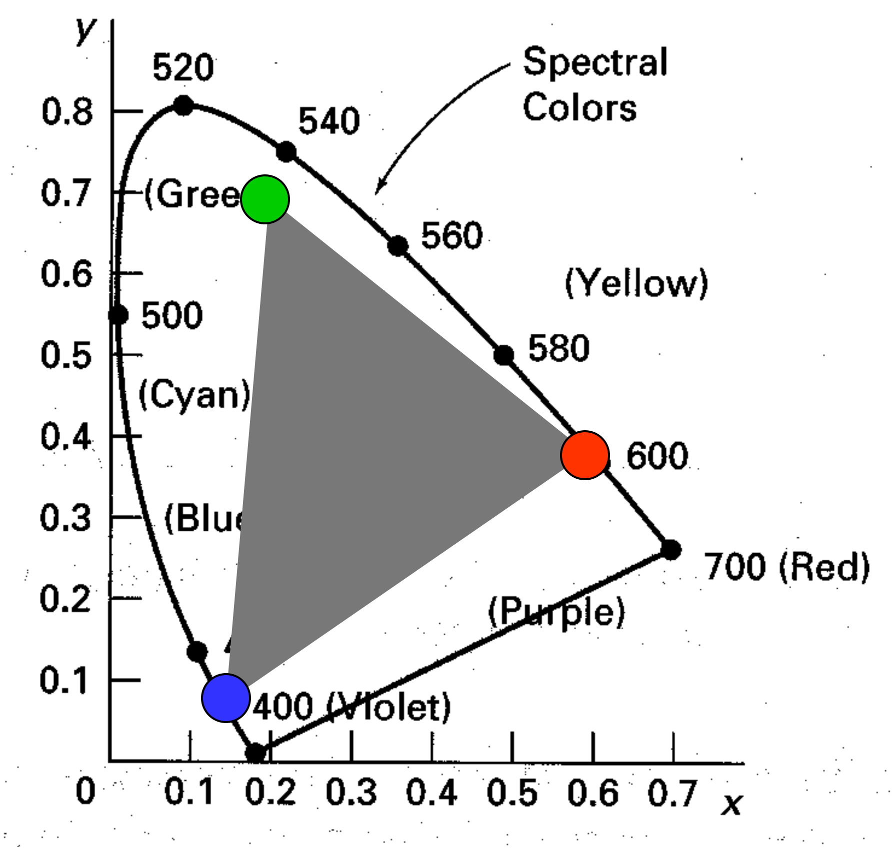



Figure 15.13 from H&B

# CMY Color Model


Colors are subtractive

| C | M | Y | Color |
|---|---|---|---|
| 0.0 | 0.0 | 0.0 | White |
| 1.0 | 0.0 | 0.0 | Cyan |
| 0.0 | 1.0 | 0.0 | Magenta |
| 0.0 | 0.0 | 1.0 | Yellow |
| 1.0 | 1.0 | 0.0 | Blue |
| 1.0 | 0.0 | 1.0 | Green |
| 0.0 | 1.0 | 1.0 | Red |
| 1.0 | 1.0 | 1.0 | Black |
| 0.5 | 0.0 | 0.0 | |
| 1.0 | 0.5 | 0.5 | |
| 1.0 | 0.5 | 0.0 | |

Plate II.7 from FvDFH

# CMY Color Cube

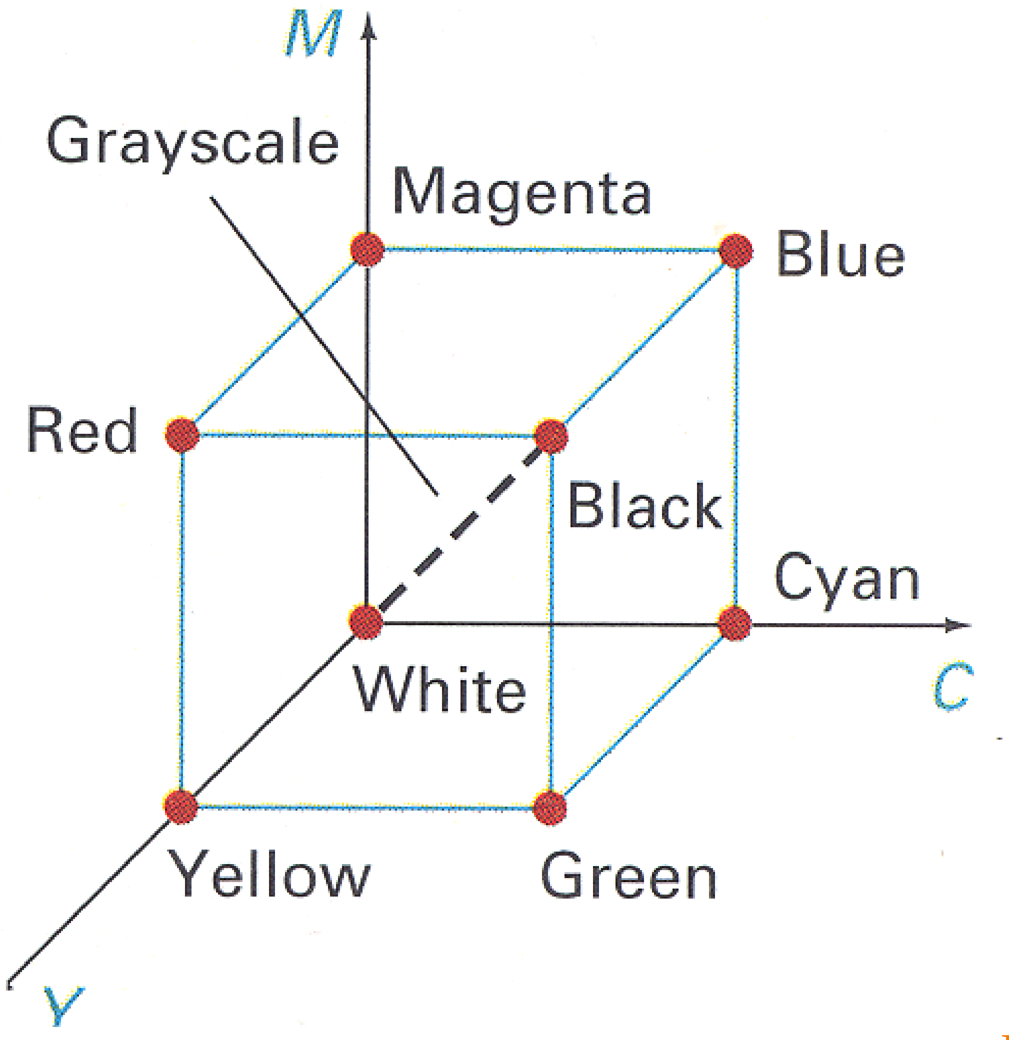



Figure 15.14 from H&B

# Summary

- Images
  - Pixels are samples
  - Frame buffers
  - Display hardware (CRTs, LCDs, printers, etc.)
  - Devices have limited resolution
- Colors
  - Spectrum across visible light frequencies
  - Tristimulus theory of color
  - CIE Chromaticity Diagram
  - Different color models for different devices, uses, etc.