Computer Graphics - Week 1

Bengt-Olaf Schneider
IBM T.J. Watson Research Center

Class Objectives

- **Overview of important topics in computer graphics**
- **Detailed understanding of fundamental 3D computer graphics algorithms and techniques**
- **Ability to design and implement graphics applications**
- **Starting point for understanding technical literature and foundation for independent research**

Class Rules: Grading

- **Requirements**
  - Midterm exam (20%)
  - Final exam (40%)
  - 4 assignments (40%)

Class Rules: Assignments

- **Programming problems about topics covered in class**
- **No cheating and no collaboration !**
- **Overall grade is the average of all assignments**

- **Each assignment worth max. 100 points**
  - Assignments due before class on assigned date
  - 10 point penalty for each day after deadline
  - 0 points if not submitted by Monday 5:30 pm

- **Programming solutions must …**
  - work with (at least) the provided data sets
  - be well structured and use only specified libraries or API calls
  - be clearly documented (comments, Readme file, etc.)

Literature

J. Foley, A. van Dam, S. Feiner, J. Hughes, *Computer Graphics Principles and Practice*, 2nd edition (C edition), Addison Wesley, 1990.

M. Woo, J. Neider, T. Davis, *OpenGL Programming Guide*, 2nd edition, Addison Wesley, 1997.

OpenGL ARB, *OpenGL Reference Manual*, 2nd edition, Addison Wesley, 1997.

A. Glassner, *Principles of Digital Image Synthesis*, Morgan Kaufman,1995.

C. Hoffmann, *Geometric and Solid Modeling*, Morgan Kaufman, 1989.

D. Rogers, *Mathematical Elements of Computer Graphics*, 2nd edition, McGraw Hill, 1989.

Graphics Gems, Vol 1-5, Academic Press.

Various journals, e.g.
- IEEE Computer Graphics and Applications
- ACM Transactions on Graphics
- IEEE Transactions on Visualization and Computer Graphics
- Computer Graphics Forum
- Computers & Graphics
- The Visual Computer

Proceedings from annual events, e.g.
- SIGGRAPH conference
- Eurographics conference
- Symposium on Interactive 3D Graphics
- Eurographics Rendering Workshop
- Siggraph/Eurographics Hardware Workshop
- IEEE Symposium on Parallel Rendering
- IEEE Symposium on Volume Rendering
- ... and many, many others

Where to Get More Information

- **Class Web-page**
  - www.cs.columbia.edu/graphics/courses/csw4160/
- **Class Newsgroup**
  - columbia.spring.cs4160
- **Office hours**
  - Wednesdays before class, adjunct office MUDD 460
  - Elias: Thursdays 12:00-1:00 pm, CEPSR 603
  - By appointment
- **E-mail + Phone**
  - bosch@us.ibm.com 914-945-1585
  - gagman@cs.columbia.edu 212-939-7077

Course Overview: Introduction

- **Introduction**
  - Historic overview, graphics application domains, 2D vs 3D graphics, graphics vs. image processing, human visual system

Course Overview: Raster Graphics

- **Raster Graphics Pipeline**
  - Overview, Coordinate systems, modeling transformations, hierarchical modeling, basic animation techniques
  - Viewing transformations, camera model, orthographic and perspective projection.
  - Lighting models. Rendering primitives
  - Clipping for lines and polygons, scissoring, capping, non-convex clip regions
- **Raster Graphics Pipeline**
  - Scan Conversion for lines and triangles, attribute interpolation, perspective correction
  - Fragment processing, z-buffer, texture mapping, stipple pattern, anti-aliasing, double-buffering
  - Event handling
  - Graphics APIs and description languages

Course Overview: Advanced Topics

- **Advanced Topics**
  - Global Illumination
    - Ray tracing, spatial data structures and advanced lighting models (refraction, transparency, optical simulation)
    - Radiosity and two-pass rendering
  - Modeling
    - CSG, free-form curves and surfaces
  - Graphics Hardware
  - Color
    - color theory, color gamuts, gamut matching, gamma correction, color maps
- **Advanced Topics**
  - Color
    - color theory, color gamuts, gamut matching, gamma correction, color maps
  - Volume Rendering
    - particle rendering, gaseous media, special effects
  - Applications
    - SciVis, games, CAD, user-interfaces

Overview of Week 1

- **What is Computer Graphics ?**
- **Applications of computer graphics**
- **Historic overview**
- **System view of computer graphics**
- **Optics for Dummies**
- **Human visual system**

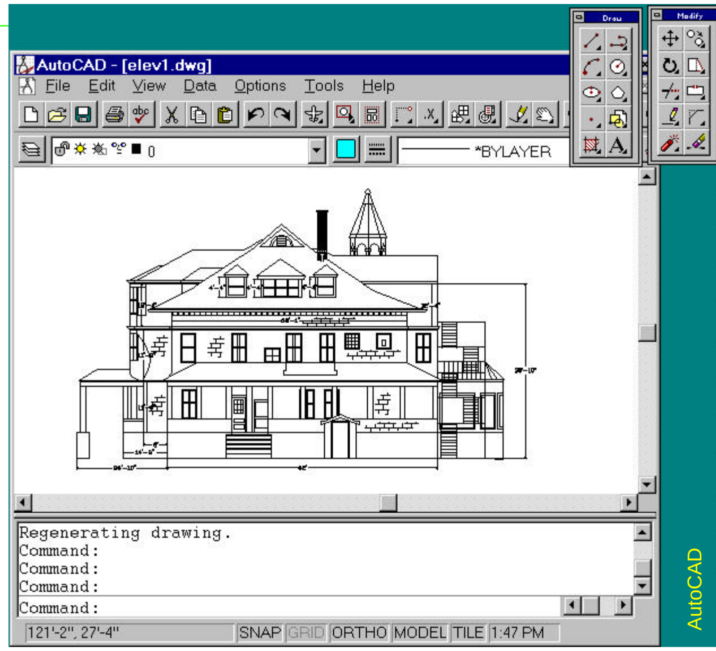
Computer Graphics

- **Also known as**
  - image synthesis
  - computer generated imagery (CGI)
  - rendering
- **Creation, storage, display, and manipulation of models and images of objects**
- **Design of software and hardware to support the display of images**
- **Interactive manipulation and editing of models**

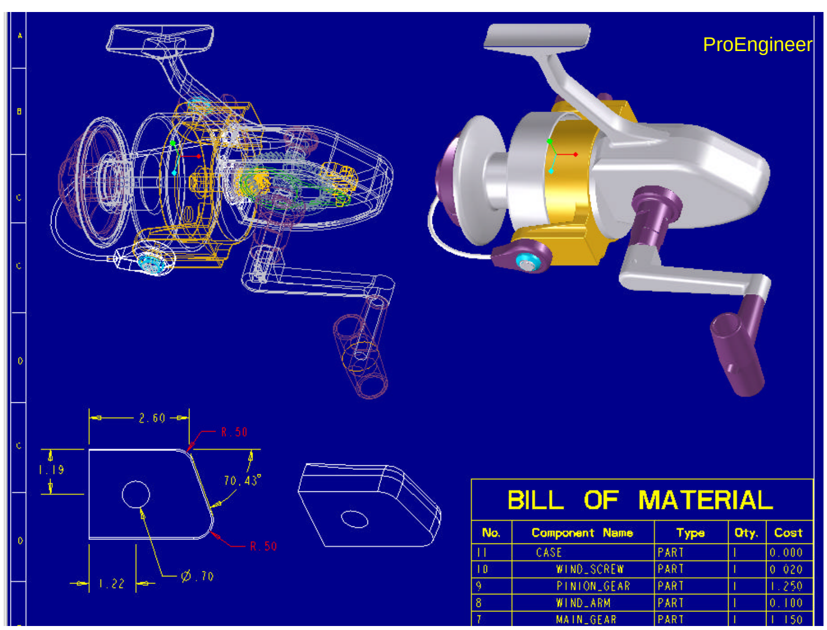
Applications of Computer Graphics

- **CAD/CAM**
  - Mechanical and architectural design
- **Entertainment**
  - Games
  - TV Animations
- **Scientific Visualization**
  - Display of multi-dimensional data, e.g. weather, petroleum, medical
- **Business Presentations**
- **Virtual Reality (VR)**
  - Modeling of a “virtual” 3D worlds (Contrast: augmented reality, a.k.a. AR)
  - Manipulation and interaction
  - Immerse or fish-tank VR
- **User interface design**
  - WIMP interface
  - Text and font technology
  - Direct Manipulation
- **Internet / WWW**

Applications: CAD / CAM

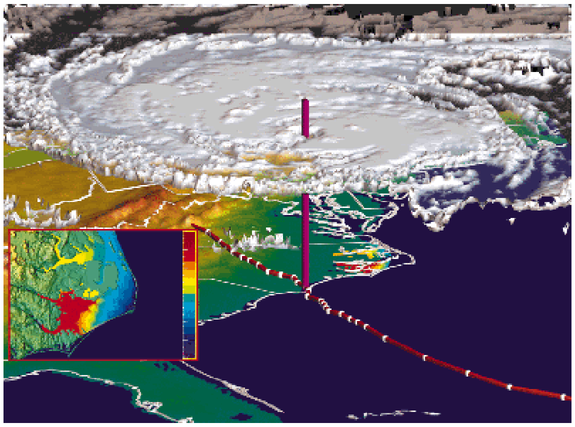


Applications: CAD / CAM



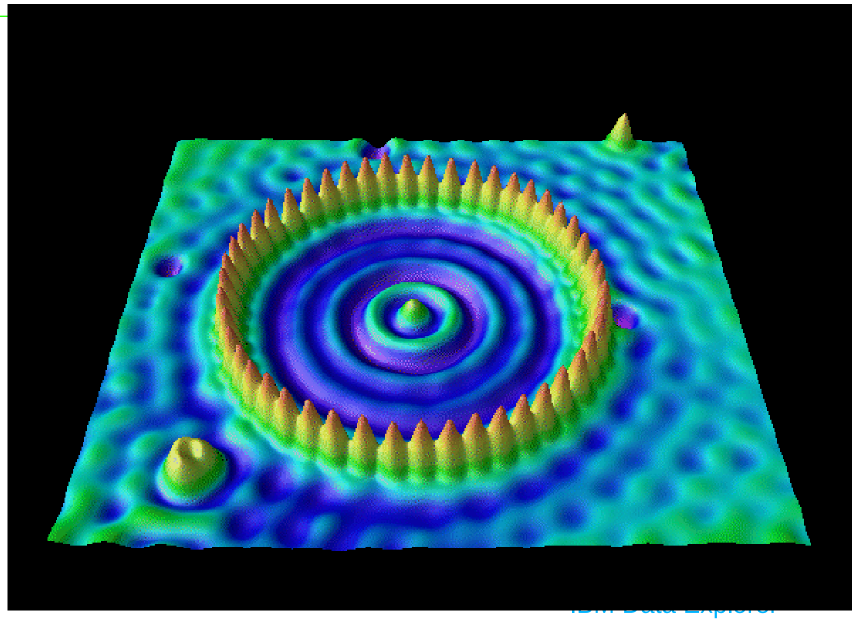
Applications: Games

Applications: Weather Visualization



IBM Data Explorer

Applications: Scanning Tunneling Microscope

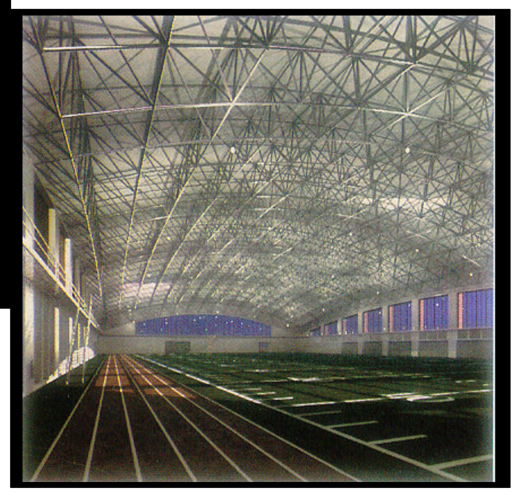


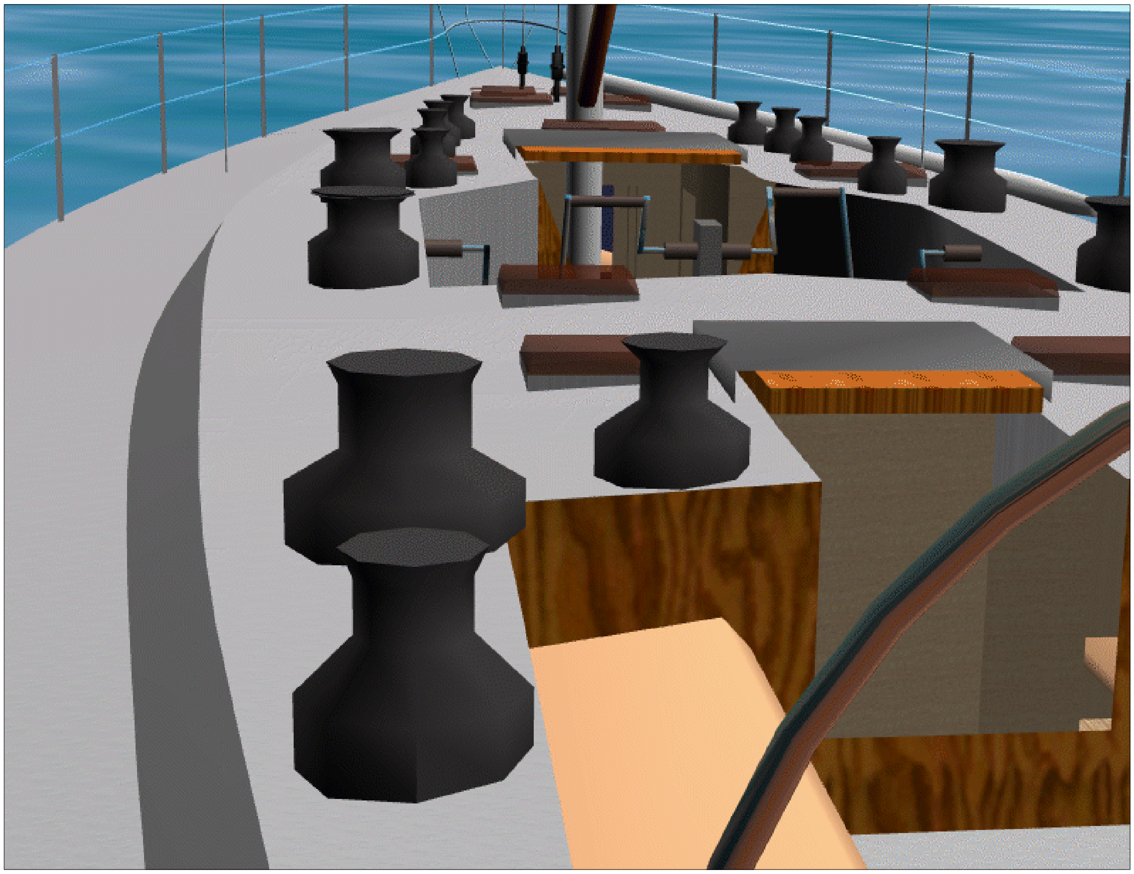
Applications: Visualization

DX Dynamic Brittle-Ductile Transition.mpg

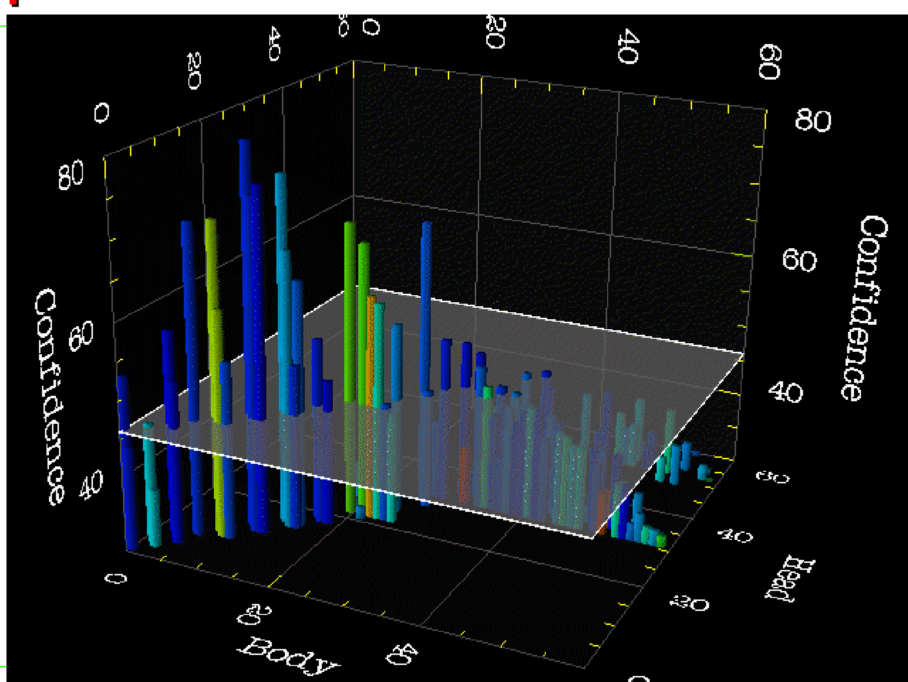
Applications: Architectural Design

(C) Scott Routen + Reuben McFarland





Applications: Business Visualization



Computer Graphics: Related Fields

- **Simulation**
  - Various fields generate data to be visualized, e.g. engineering, science, art, sociology/psychology, architecture
- **Modeling**
  - Exact (mathematical) description of models
  - Representations optimized for storage, portability, editing, queries, robustness, efficient display, ...
- **Image Processing**
  - Manipulation, storage and display of raster images
- **Physics / Optics**
- **Electrical and Computer Engineering**
  - Design of computers and devices to display graphics

2D vs. 3D Graphics

- **2D Graphics**
  - 2D primitives (duhh !)
    - Lines, Polygons, Text, Patterns
  - Rendering
    - Typically directly controlled by the application
    - Low-level specification, e.g. `DrawLine (p1, p2, color)`
  - Applications
    - User interfaces
    - Desktop graphics (word processing, presentations, drawing packages)
    - Drafting
    - ...
- **3D Graphics**
  - 2D and 3D primitives
    - Lines, polygons, polyhedra, ...
  - Rendering
    - Specified by the application and controlled by graphics subsystem
    - Low-level specification, e.g. `DrawTri (p1, p2, p3, color)`
    - Higher-level specificiation, e.g. `DrawTri (p1, p2, p3, light)`
    - High-level specification, e.g. `DrawObject (o, xform, mat)`
  - Applications
    - CAD / CAM
    - Virtual Reality
    - Simulation + Games
    - ...

Raster vs. Vector Graphics

- **Raster Graphics**
  - The display is divided into small dot, the picture elements (a.k.a. pixels)
  - Allows the display of filled and shaded areas
  - In the end, anything to be displayed is converted into pixels.
  - Hardware must provide storage for every pixel on the screen: the Frame Buffer
- **Vector graphics**
  - The basic display primitive are line segments (a.k.a. vectors or strokes)
  - Allows for display of wireframes
  - All objects have to be converted to a collection of lines
  - Hardware is responsible for cycling through all vectors in no more 1/30s second.

Graphics vs. Image Processing

- **Graphics**
  - Basic primitives are geometric shapes
    - Simplest primitive is a 2D/3D point
    - Higher-level primitives include other geometric shapes
      - Polygons, conic sections, splines, solids
    - Primitives describes in arbitrary coordinate system that gets mapped onto the display device
  - Operations
    - Affine transformations
    - Lighting and Shading (3D)
    - Texturing and pattern
- **Image Processing**
  - Basic primitives are raster images
    - Simplest primitive is a single pixel
    - Higher-level primitives
      - rectangular bitmaps / pixmaps
      - objects (arbitrary collection of pixels)
      - layers and channels
  - Operations defined to modify and manipulate pixel values
    - Manipulation
      - Rotation, scale, shear, ...
    - Filtering
      - Brightness, contrast, edges, blurring, ...
    - Transformations
      - Fourier, DCT, Wavelet, ...
    - Compression

A Brief History of Computer Graphics

- **Pre-1960**
  Special-purpose graphics solutions
  - MIT Whirlwind
    - Modified oscilloscope for visualization and analysis of aircraft stability
  - SAGE air-defense system
    - Vector screen for display of radar targets
    - Light pen input

- **1960s**
  Graphics as a discipline
  - Basic graphics algorithms
    - Modeling and viewing
    - Line and polygon clipping
  - Satellite displays
    - Attached to mainframe computers over low-bandwidth connections
    - High-level commands controlled the graphics terminal, requiring "intelligence" in the terminal
    - High cost, restriction to defense and industrial applications
  - 1963: Ivan Sutherland's PhD
    - Introduced many fundamental concepts still in use today

A Brief History of Computer Graphics

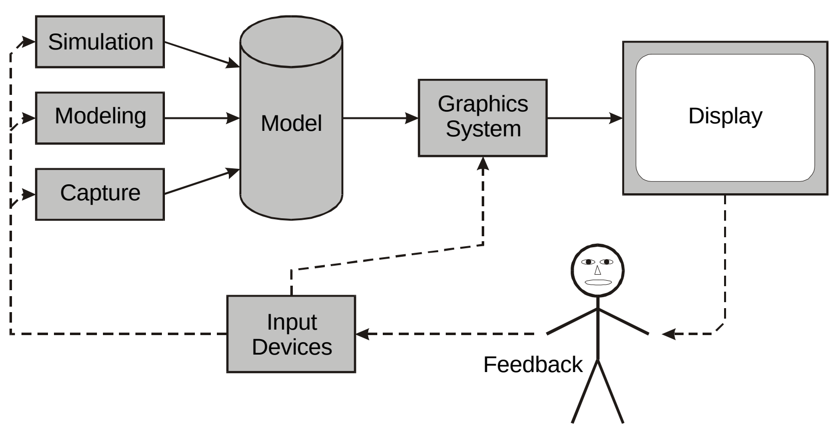
- **1970s**
  Commercial graphics & Raster graphics
  - First raster displays
  - All fundamental (raster) graphics algorithms
    - Hidden surface removal
    - Clipping
    - Lighting and shading
    - Curve and surface modeling
  - First textbooks
    - Newman & Sproull, *Principles of Interactive Computer Graphics,* 1973.
    - Rogers & Adams, *Mathematical Elements for Computer Graphics, 1976.*

- **1980s**
  Mainstream, Modeling, Photorealism
  - Personal computers
    - Apple II, Lisa, Macintosh
    - Later IBM PCs with CGA, EGA, VGA, SVGA, XGA, ... graphics
    - Window-based operating systems
  - Solid Modeling
    - Binary Space Partitioning (BSP)
    - Constructive Solid Geometry (CSG)
  - Global Illumination
    - Ray-tracing (1968, late 70s, early 80s)
    - Radiosity (1984)
    - Improved illumination models

A Brief History of Computer Graphics

- **1990s Ubiquitous+cheap, convergence**
  - Pervasive graphics hardware
    - Ubiquitous use of graphics in business and consumer applications
    - 3D graphics hardware is standard on home PCs
  - Image-based rendering
    - combines graphics and imaging
    - visualization of (really) large models
  - Digital content creation
    - Special effects and post-processing
    - Toy Story
- **2000 and beyond ???**
  - Improved user interfaces (3D, direct manipulation, intuitive and productive)
  - Distributed (networked) graphics
  - 3D hardware will be truly universal and cheap
  - Converged graphics, video and imaging
  - Large and small high-resolution displays

System View of Computer Graphics

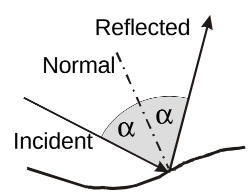


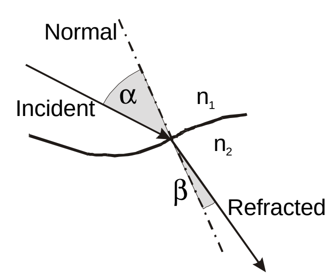
Key Components of a Computer Graphics System

- **Application**
  - Generates graphics data and interprets user input
- **Model**
  - Interface between application and graphics components
  - Representation must support application and graphics
- **Graphics Subsystem**
  - Interprets model and converts it into pixels (or vectors)
- **Output (Display)**
  - CRT, plotter, printer, film …
- **User**
  - Receives and interprets visual signals
  - Interacts with the application to control model and/or graphics
  - Closes the feedback from display to application

Optics for Dummies

- **Ray optics**
  - In a homogeneous medium light proceeds in a straight line
- **Reflection**
  - Incident and reflected ray subtend same angles with the surface normal
  - Incident ray, reflected ray and surface normal lie in a plane
- **Refraction**
  - Ray is refracted towards surface normal when entering denser material: $\sin a / \sin b = n_2 / n_1$
  - Wavelength dependent (prism !)
  - Incidentray, refracted ray and surface normal lie in a plane





Optics for Dummies (cont'd)

- **Wave optics**
  - Light is electromagnetic energy, characterized by
    - Amplitude *I*, perceived as intensity or brightness
    - Frequency or wavelength *f* = c / k, perceived as spectral color
      - Visual light: ? = 380 … 780 nm (red … violet)
    - Polarization
- **Waves can be linearly superimposed**
- **A spectrum is the linear combinations of light with different wavelengths**
- **Wave optics are important when explaining**
  - diffuse illumination phenomena
  - interaction of light and thin layers, e.g. soap bubbles
  - surface properties (metal vs. plastic, color filters, etc.)

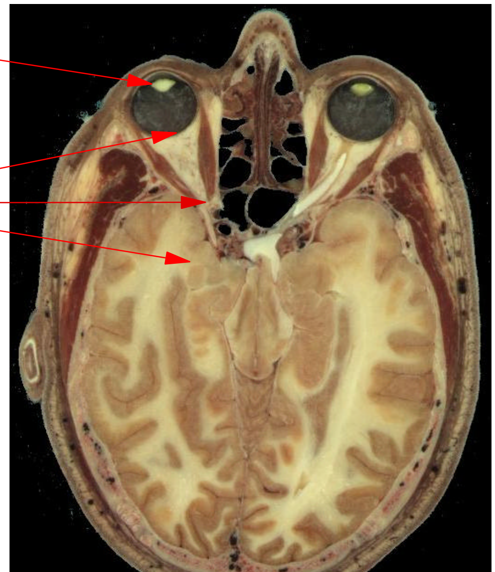
Human Visual System

- **Why study the visual system ?**
  - It is part of the graphics *system*.
  - It helps to understand various "optical illusions".
  - Good user interface design account for its properties.
- Further reading
  - A. Glassner, *Principles of Digital Image Synthesis*, Vol. 1
  - G. Wyszecki, W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, 2nd edition, Wiley, 1982.

Human Visual System: Overview

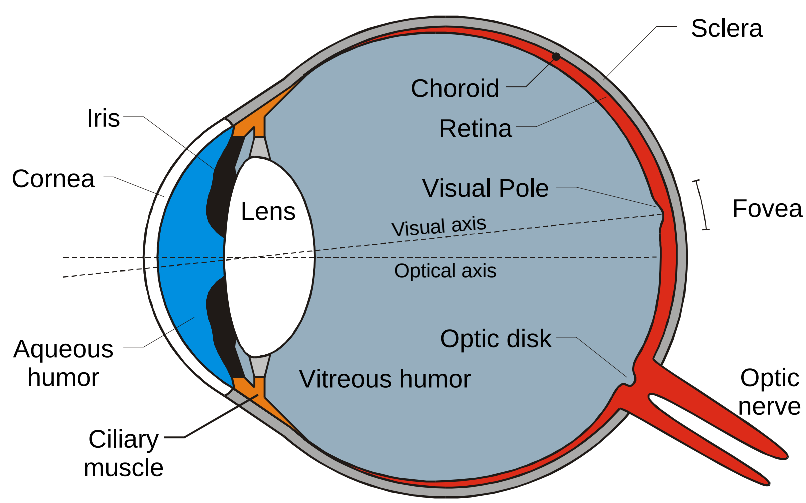
- **Optical path**
  - Eyes
  - Relatively well understood
- **Processing part**
  - Retina (low-level processing)
  - Optic nerve
  - Brain (high-level processing)

  - Processing at increasing levels of abstraction
  - Decreasing levels of knowledge about the details



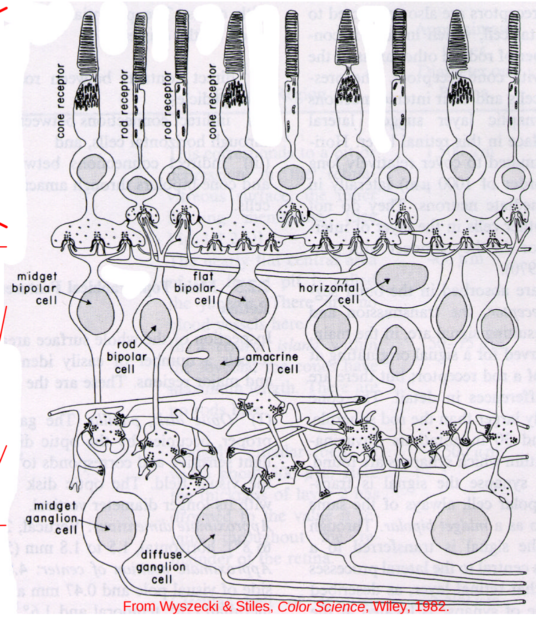
Visual Human Data Set, Axial Slice #109

Cross-section of the Eye



Cross-section of the Retina

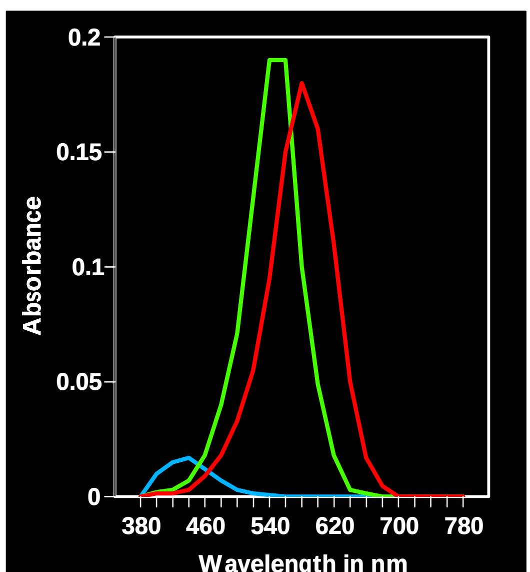
- **Photoreceptors**
  - Rods
    - Black/white perception
    - Very sensitive
    - Denser around perimeter, none in the fovea
  - Cones
    - Color perception
    - Concentrated around fovea
    - 3 types for different colors
- **Signal Processing Layer**
  - First Synaptic Layer
  - Intermediate Neurons
  - Ganglion Cells
  - Second Synaptic Layer



From Wyszecki & Stiles, *Color Science*, Wiley, 1982.

Spectral Absorbance

- **3 types of cones for different wavelengths**
  - S (short): 430 nm (blue)
  - M (medium): 550 nm (green)
  - L (long): 580 nm (red)
  - Different sensitivity
  - Mostly used during daytime
- **Rods have maximum sensitivity at 500 nm**
  - Useful for night vision

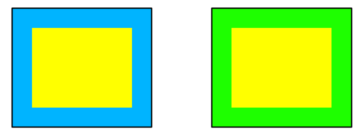


Redrawn from Foley et al. *Computer Graphics - Principles and Practice*

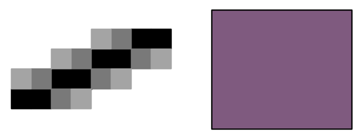
Retina: Signal Processing Functions

- **Edge detection and enhancement**
  - Mach-band effect
  - Shading of smooth surfaces

- **Simple color transformations**
  - Color opponency, e.g. contrasting colors are mutually amplifying
  - User interface design



- **Noise tolerance**
  - Distribution of cones across the retina
  - Anti-aliasing and dithering



Depth Perception

- **Combination of different mechanisms**
- **Oculomotor Depth**
  - Accomodation of the lens to bring object into focus
- **Binocular Depth**
  - Stereopsis
  - Near field depth perception
  - Requires matching of left and right images
    - Note: Random dot stereograms indicate that pattern matching is not the first step
- **Monocular Depth**
  - Interposition (visibility, occlusion)
  - Perspective
    - Size; larger objects are closer than smaller objects.
    - Convergence of straight lines
    - Texture gradient (pattern denser in the distance)
    - Athmospheric effects, e.g. fog, haze
  - Motion parallax
    - Detection of visual flow
    - Close objects move faster across the visual field than distant objects.

Input Devices

- Closes the feedback loop between user and graphics system
- Classification of Input Devices
  - Type
  - Absolute or relative
  - 2D or 3D
  - Haptic (force feedback)

Input Devices Types

- **Locator**
  - Position or orientation
  - Tablet, Mouse, Trackball, Joystick, Lightpen, Touchscreen
- **Pick**
  - Selection of a graphical entity
  - Many locator devices coupled with trigger button(s)
- **Valuator**
  - Input of a single real number
  - Dials, sliders
- **Keyboard**
- **Choice**
  - Selection from a limited number of choice
  - Function keys, menus, soft (on-screen) buttons

Absolute vs. Relative Locator Devices

- **Absolute locators report coordinates with reference to a fixed coordinate system**
  - Examples: Tablet, Lightpen, Touchscreen
  - Advantage: Accurate input of space coordinates
  - Disadvantage: Cost, Footprint
- **Relative locators report coordinates with respect to previous coordinates**
  - Examples: Mouse, Trackball, Continuous dials (potentiometers)
  - Advantage: Cost, Footprint
  - Disadvantage: Requires software to compute absolute coordinates

Input Mechanisms

- **Optical**
  - Light barrier
    - Optical touch screens to detect finger position
  - Bending of fibers
    - E.g. VPL data glove
  - Pattern recognition
    - Optical mice detect a pattern of red and green lines in the mouse pad
    - Tracking of features for motion capture
- **Mechanical**
  - Mouse: Rotation of ball is sensed using rollers that actuate potentiometers or slotted disks interrupting a beam of light
  - Accelerometers
- **Electrical**
  - Resistance
    - Potentiometers
    - Resistive foam
    - Strain gauges (Thinkpad)
  - Capacitance
    - Sense the distortion in an electric field
- **Magnetic**
  - Distortion / strength of magnetic field
  - Emitter stationary, sensor mobile

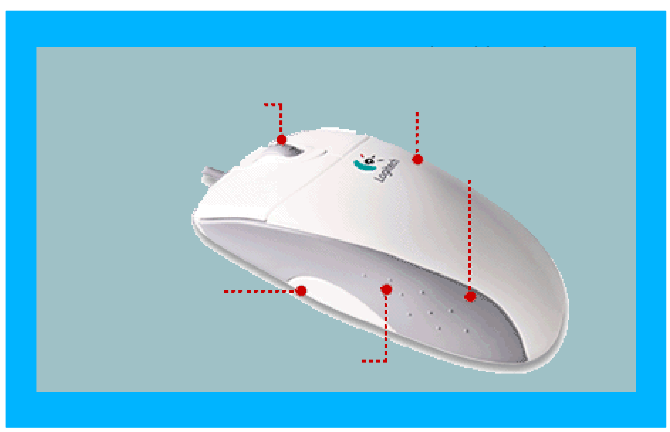
Interfacing to Input Devices

- **Polling**
  - Application-controlled query of the device
    - Request of state and position information
    - Often requires detailed knowledge about how to talk to the device
  - Might introduce delays in application due to wait for slow device
    - May require application to estimate delay between request and delivery of data
  - Often the only way to use non-standard devices

- **Events**
  - OS provides application with events to signal device activity
    - Requires integration of the device (type) into the OS or windowing system
    - Abstracts from device specific interface
    - Event data structure contains all or partial data about device
  - Application may get inundated with events
    - Event manager and/or application may have to discard intermediate events

Input Devices: Mice

Copyright IBM Corp.



Copyright Logitech Inc.

Input Devices: 3D Mouse

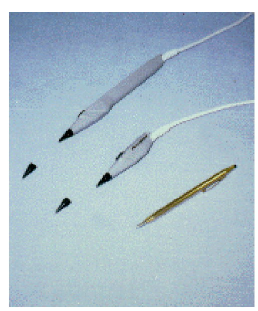
Copyright Logitech Inc.

Input Devices: Spaceball

Copyright Sun Microsystems Inc.

Input Devices: Puck

Input Devices: Polhemus Trackers



Input Devices: Data Glove

Input Devices: Tablet

Summary

- **Objectives of Computer Graphics**
  - Differences to neighboring disciplines
- **Overview of the historical development of the field**
- **Overall structure of a Graphics System**
- **Working of the human visual system**
- **Characteristics and classification of input devices**

Homework

- **Review material and read background texts**
  - Foley et al.: Chapters 1, 8, 8.1,
  - Glassner: Chapter 1
- **Prepare**
  - Foley et al.: Chapters 5 + 7 (Geometrical Transformations and Object Hierarchy)
  - Foley et al.: Appendix (Mathematics for Computer Graphics)

Next week ...

- **Raster Graphics Pipeline**
- **Geometric Transformations and Hierarchical Modeling**