

# Graphics and Visualization

Holger Kenn

International University Bremen

Spring Semester 2006

Introduction

Colorful pictures

Course Organization

# Graphics and Visualization

- ▶ Course Website:  
<http://www.cubeos.org/lectures/cg-iub-spring-2006/>
- ▶ Textbook:F.S. Hill: Computer Graphics using OpenGL

# What to expect:

This course is about...

- ▶ Programming CG
- ▶ CG Algorithms
- ▶ OpenGL and GLUT
- ▶ Data Structures for CG
- ▶ File Formats for CG
- ▶ Image and Video Codecs
- ▶ Applications of CG

## What to expect:

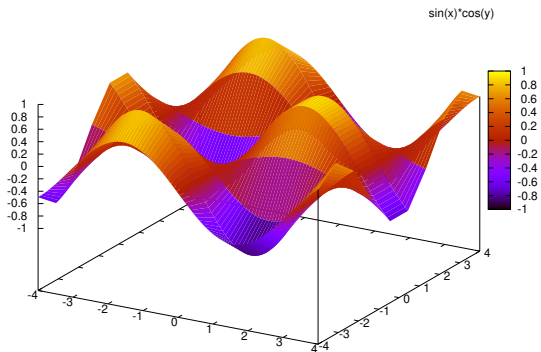
This course is not about...

- ▶ Photoshop / Corel Draw
- ▶ AutoCAD / OrCAD
- ▶ IDL
- ▶ Blender
- ▶ ripping DVDs and making divx files
- ▶ ... some knowledge can help ...
- ▶ ... and these are CG applications ...

# What to expect:

- ▶ Colorful pictures
- ▶ Movies
- ▶ Games
- ▶ Programming (C/C++)
- ▶ Mathematics (Matrix algebra, Trigonometry)

# Computer Graphics



# OpenGL Graphics

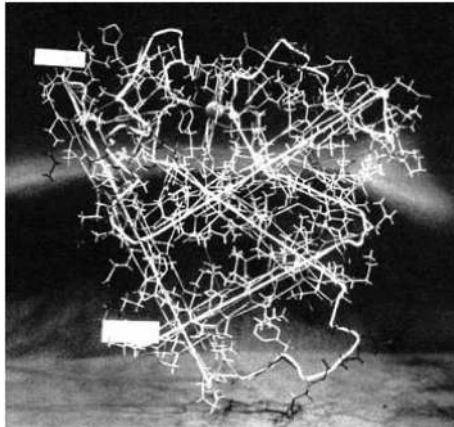




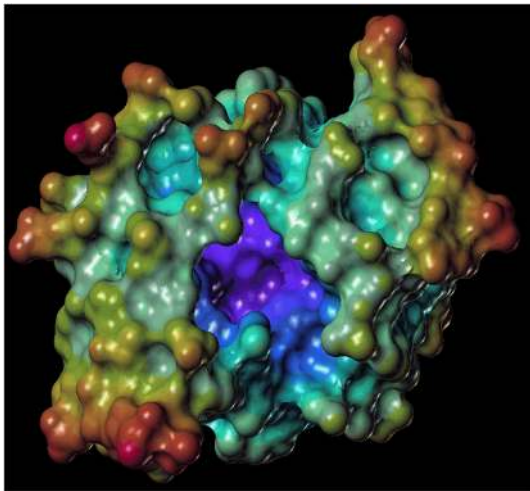
# Graphics in Science



# Graphics in Science



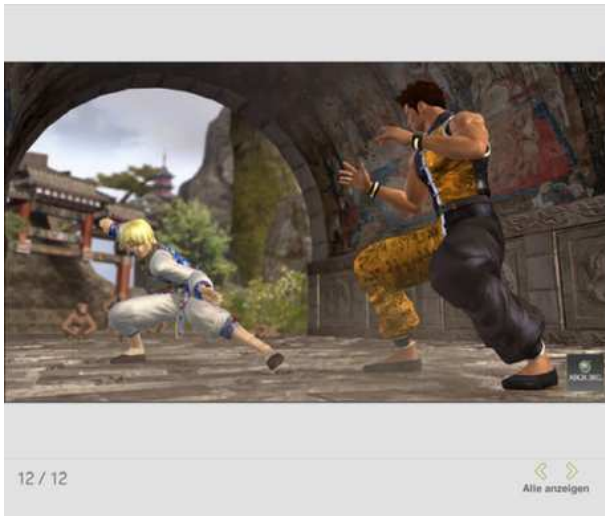
# Graphics in Science



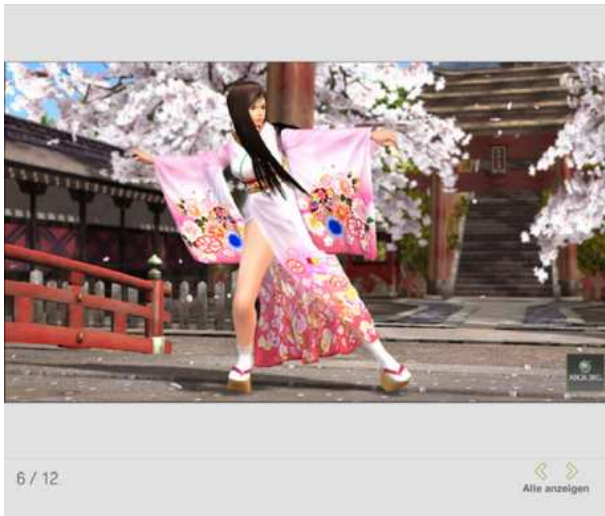
# Graphics in Games



# Graphics in Games



# Graphics in Games



# Graphics in Games



8 / 19

Alle anzeigen

# Graphics in Games



1 / 6

Alle anzeigen



# Graphics in Movies



# Course Overview

- ▶ Pixels, Bitmaps, Aliasing, Color, Gamma Correction, Alpha Blending
- ▶ OpenGL Basics
- ▶ Interactive graphics, i.e. GLUT
- ▶ Coordinate Systems, representation, transformations
- ▶ Drawing lines, affine transformations, clipping, fractals, textures
- ▶ Drawing 3D objects, 3D affine transformations
- ▶ Surfaces, Light and Cameras
- ▶ 3D Texture mapping
- ▶ Ray Tracing, Reflection, Refraction

# The Basics

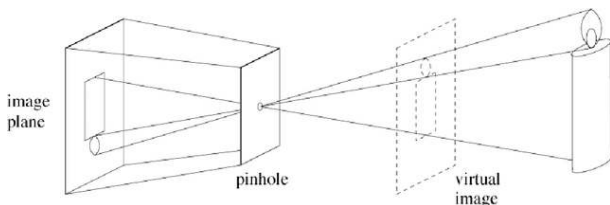
- ▶ Toy Optics:
  - ▶ Light is emitted by a light source and travels in a straight line through space
  - ▶ Light is absorbed or reflected by objects or passes through objects
  - ▶ An observer can detect light, i.e. “see” Observers are the eye or a camera.
- ▶ Holding up a photographic plate (or a CCD sensor) does not give an image.
- ▶ An optical system is needed to “see”.

# Camera Obscura



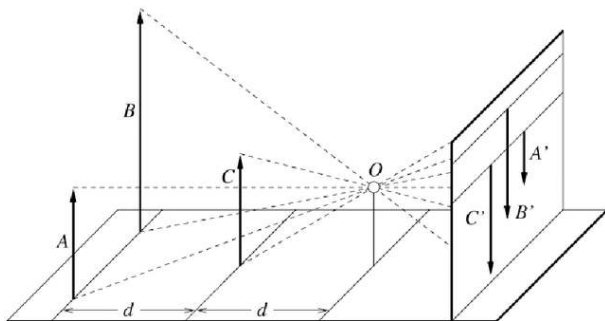
- ▶ Dark room with small hole and projection surface
- ▶ Known since 5th century BC (China)

# Pinhole Camera



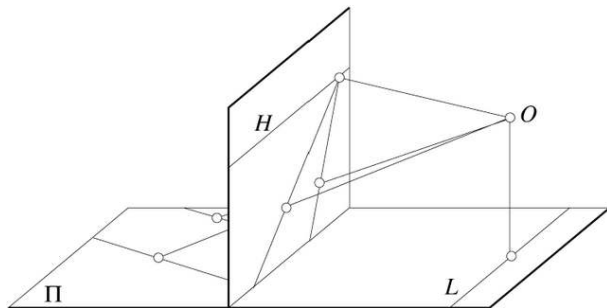
- ▶ Simple abstract camera model
- ▶ Used in many simple CG systems

# Pinhole Camera



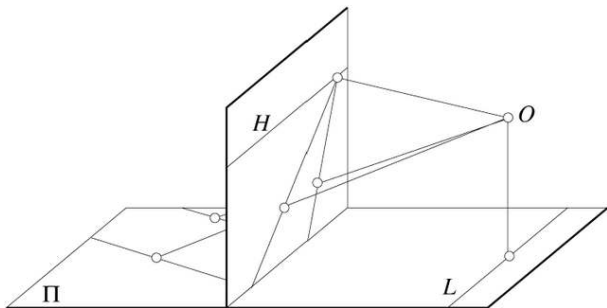
- Perspective: distant objects are smaller

# Pinhole Camera



- ▶ Drawing convention: Draw focal plane “in front” of the focal point

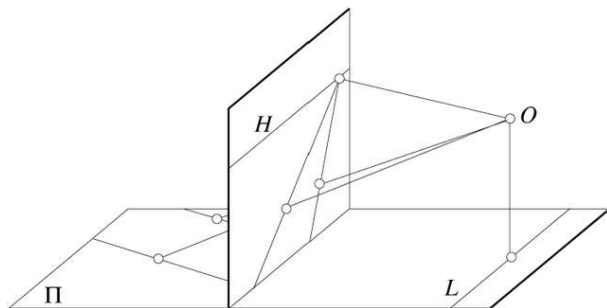
# Pinhole Camera



- ▶ Observation: Moving the focal plane from or towards the focal point scales the image

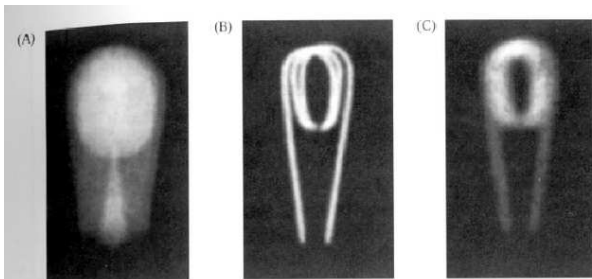


# Pinhole Camera



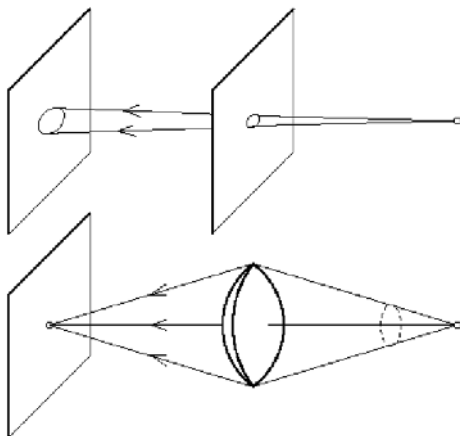
- ▶ Observation: Parallel lines in the scene meet in the image

# The Real Pinhole Camera

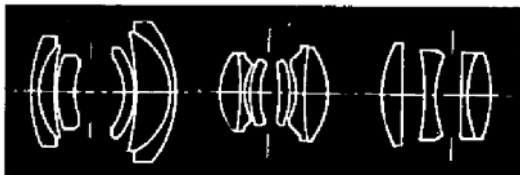


**2.18 DIFFRACTION LIMITS THE QUALITY OF PINHOLE OPTICS.** These three images of a bulb filament were made using pinholes with decreasing size. (A) When the pinhole is relatively large, the image rays are not properly converged, and the image is blurred. (B) Reducing the size of the pinhole improves the focus. (C) Reducing the size of the pinhole further worsens the focus, due to diffraction. From Ruechardt, 1958.

# Pinhole vs. Lens



# Lens Systems



- ▶ A lens system breaks the tradeoff between “sharp” images and “bright” images
- ▶ Lens systems are characterized by focal distance and aperture
- ▶ Complex lens systems are used to compensate undesirable lens effects: Pincushion and barrel distortions, Spherical and chromatic aberration, Astigmatic distortions

# Camera Optics in CG

- ▶ Complexity of lens-based camera model used to be the reason to use the pinhole camera model in CG
- ▶ Pinhole optics is a simple mathematical model for perspective projection
- ▶ Some lens effects cannot be modeled. (Which?)
- ▶ Modern CG systems model lens effects to create more realistic images

# INSERT FILM HERE

- ▶ Film: Geri's game, Pixar
- ▶ Task: Watch for lens effects

# Course Organization

- ▶ Tuesday, 8.15-11 (if Robotics lecture can be moved)
- ▶ Lectures: West Hall 2
- ▶ Tutorial sessions: CLAMV teaching lab
- ▶ Prerequisites:
  - ▶ Formal: Fundamental CS 1 & 2, CS 2nd year lab course
  - ▶ Informal: C++ programming, Unix-based software engineering
- ▶ Course work:
  - ▶ Reading and understanding OpenGL documentation
  - ▶ Independent OpenGL C++ programming
- ▶ Grading: To be decided . . .