University of California, Los Angeles

UCLA CS174A (Fall 2022)
Introduction to Computer Graphics

Prof. Asish Law

TAs: Chenda, Noor, Wuyue, Junlan

Instructor & TAs

Name	Asish Law	Chenda Duan (A)	Noor Nakhaei (B)	Wuyue Lu (C)	Junlan Lu (D)
Role	Instructor	TA	TA	TA	TA
Email	alaw@cs.ucla.edu	chenda@ucla.edu	noornk@ucla.edu	luwuyue@ucla.edu	jlu21@cs.ucla.edu
Phone	626.379.3377	310.254.5864	310.254.0439	424.431.4929	626.715.3847
Class Location	BUNCHE 2209A	BROAD 2160E	PERLOFF 1102	KAPLAN A65	DODD 146
Class Hours	TR 6 - 8 PM	F 4 - 6 PM	F Noon - 2 PM	F 2 - 4 PM	W 11 AM - 1 PM
Office Location	Bunche 2209A	Zoom	Zoom	Zoom	Zoom
Office Hours	TR 8:00 - 8:30 PM	W 10 AM - Noon	R 3 - 5 PM	M 2 - 4 PM	W 4 - 6 PM

Make sure you’re ready for class:

BruinLearn: https://bruinlearn.ucla.edu/courses/140100

Email: ???? ~~http://www.ursa.ucla.edu~~

My Background

- 1996: PhD in CS from The OSU, specialized in 3D Volume Rendering
- 1996-1999: moved to LA in 3D gaming/animation industry
- 1998: joined UCLA to initiate CS174A
- 1998-2004: taught 3D CG at UCLA and USC
- 2019: returned to UCLA to teach CS174A
- Other Educational Areas:
  - High school math/science tutor
  - Course design for using AI in Education

Teaching Protocol

- ~~Fully online using Zoom, including office hours~~
  - ~~Do not share zoom link~~
  - ~~Keep yourself muted at all times unless you want to ask a question~~
  - ~~No need to turn on your camera~~
  - ~~Follow all other zoom class protocols~~
- ~~Hybrid media, switching between:~~
  - ~~**Slides**: using Powerpoint on my Windows laptop~~
  - ~~**Whiteboard**: using Notability on my iPad~~
  - ~~**Zoom Chat**: students are free to chat and post questions in chat box~~

Teaching Protocol (Contd.)

- ~~All **zoom** lectures will be recorded and posted in BruinLearn~~
  - ~~**IMPORTANT**: Any time you realize that I forgot to start recording the lecture, PLEASE remind me~~
- ~~Interact during class: raise hand, use chat, no in-class polling~~
- ~~Post-lecture hangout: 10-15 mins~~

Contact Protocol

- We expect you to use the course forum (BruinLearn)
- Good for questions and answers
  - Dissemination of information, assignments, etc.
- Try to keep emails for class related personal matters
  - If you ask about assignments, etc. we will likely redirect you to forum

Exam Protocol (~~if ZOOM~~)

- ~~Fully online on Zoom~~
- ~~All cameras must be turned on~~
- ~~Scan and upload in Gradescope~~
- ~~We will mockup an end-to-end quiz during TA session~~

Purpose of Class

- Learn basic computer graphics and interaction
- Opportunity to use all of your CS skills
- Help each other, learn how to work in a team
- HAVE FUN!
- More detail in syllabus

Textbook (Optional)

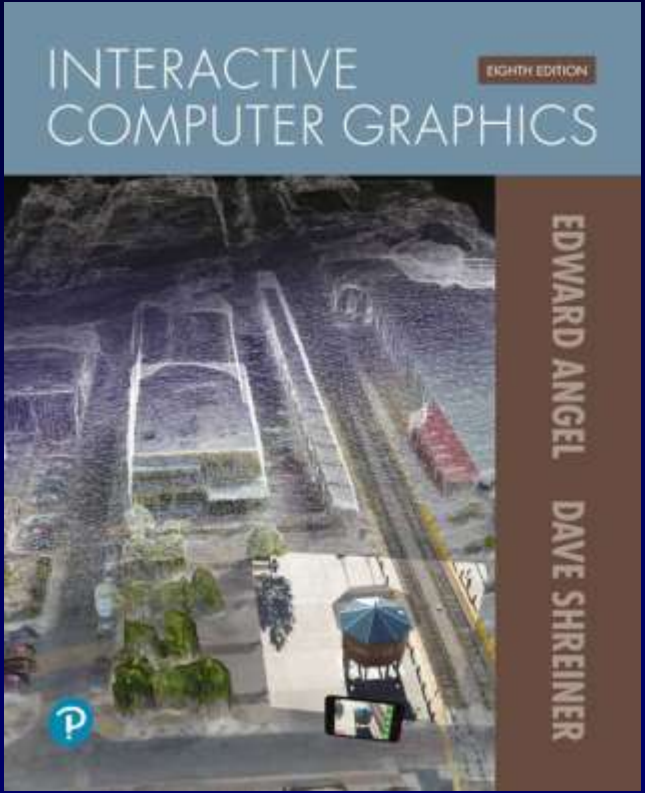
Interactive Computer Graphics – A Top-Down Approach with WebGL

E-Text only

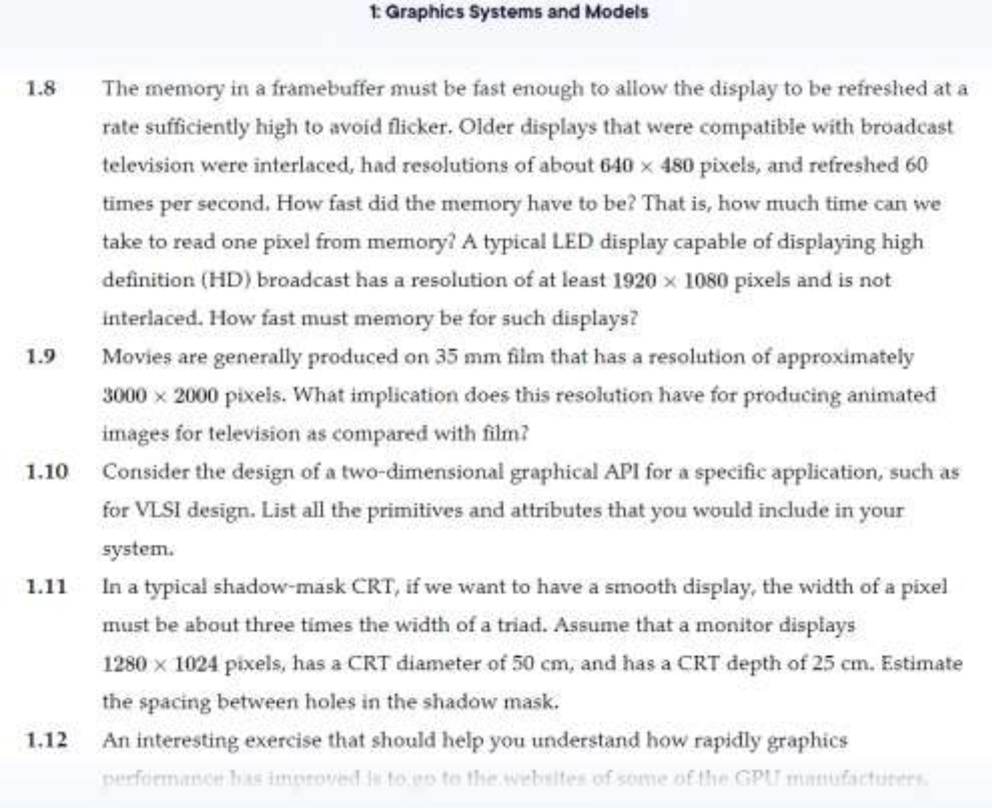
Edition 8e, Angel & Shreiner

ISBN-13: 978-0135258262

Pearson E-Book



Textbook: Sample Exercises



1.8 The memory in a framebuffer must be fast enough to allow the display to be refreshed at a rate sufficiently high to avoid flicker. Older displays that were compatible with broadcast television were interlaced, had resolutions of about 640 × 480 pixels, and refreshed 60 times per second. How fast did the memory have to be? That is, how much time can we take to read one pixel from memory? A typical LED display capable of displaying high definition (HD) broadcast has a resolution of at least 1920 × 1080 pixels and is not interlaced. How fast must memory be for such displays?

1.9 Movies are generally produced on 35 mm film that has a resolution of approximately 3000 × 2000 pixels. What implication does this resolution have for producing animated images for television as compared with film?

1.10 Consider the design of a two-dimensional graphical API for a specific application, such as for VLSI design. List all the primitives and attributes that you would include in your system.

1.11 In a typical shadow-mask CRT, if we want to have a smooth display, the width of a pixel must be about three times the width of a triad. Assume that a monitor displays 1280 × 1024 pixels, has a CRT diameter of 50 cm, and has a CRT depth of 25 cm. Estimate the spacing between holes in the shadow mask.

1.12 An interesting exercise that should help you understand how rapidly graphics performance has improved is to go to the websites of some of the GPU manufacturers.

Textbook

- **Textbooks are expensive**
  - Ask on Canvas
- **Only loosely going to follow this book**
  - Our methodology will diverge quickly
- **Otherwise this class will mostly follow the usual progression**
  - Many lecture slides have stood the test of time
  - Some we’ll skip to make room for hands-on demos

Course Administration

- **Five project assignments**
  - Last one is an open ended group project
  - Others are to build skills for that one
  - 25% penalty for each day late submission
- **Grading breakdown (500 pts):**
  - 4 projects (0+25+25+25): 75 pts
  - Team project: 150 pts
  - Midterm: 100 pts
  - Final: 175 pts

Course Administration (contd.)

- Exam format: freeform, not multiple choice, math-heavy
- Curving final grades up or down is not ruled out, if needed to move the distribution so that grades are not too uniform or too low
- Besides that, final grades will be awarded as follows:
  - D-: 60%+, D: 63%+, D+: 67%+, C-: 70%+, C: 73%+, C+: 77%+,
    B-: 80%+, B: 83%+, B+: 87%+, A-: 90%+, A: 93%+, A+: 97%+
- PNP option for S22: ***https://www.seasoasa.ucla.edu/academic-updates/***

Programming Assignments

- JavaScript, HTML/CSS, Git are used in this class
- If you don’t know these, you will by the end
- You’ll even see TAs use them on the screen
- Learn to use debuggers (we will not debug for you)

Programming Assignments

- Production software off limits - let’s make our own
- Libraries - only our small helper code is permitted for most assignments
- For team project, outside code won't count towards grade or impressiveness

Team Project (150 points)

- Team sizes 3-4
- Expectations scale with size
- Preliminary proposal: 25 points
- Final proposal & midway evaluation: 25 points
- Final demo & report: 100 points

Team Project

- **You get to make something!**
- **Show off!**
- **Be proud of it**
- **Put it on your resume**
  - But it was a team effort?
  - So far that's been the effective part!
- **Congratulatory Video Spring 2020**

Individual Assignment Examples

- **Done by one person**
- **Done using only:**
    - Plain JavaScript
    - Low level math
    - Pixel operations
    - Some 2D image files
    - An understanding of 3D techniques

TA Sessions

- Primarily to discuss project assignments: JavaScript, project, tiny-graphics, etc.
- Occasionally for more detail (linear algebra)
- Deeper dives into specific topics from class
- Additional math examples
- Q&A, homework help

TA Session This Week

Following topics are on TA Session agenda:

- ~~***Online sessions & exams***~~
- ***Walkthrough of Discussion and how the class intends to use it***
- ***Discuss assignment #1***
- ***Introduction to JavaScript, WebGL & tiny-graphics library***
- ***Grading criteria for projects: individual and team***
- ***Intro to linear algebra and matrices***

Mathematics of Computer Graphics

- **Linear (vector/matrix) algebra**
- **Coordinate systems**
- **Geometry**
  - Points, lines, planes
- **Affine transformations**
- **Projection transformations**
- **More geometry**
  - Curves, surfaces

Typical Comments From Prior Quarters

- Lots of math!
- A lot of material
- A lot of programming
- A lot of fun!!!!
- Not for modelers or artists

Remember:

- Come to class and Friday discussions (in-person)
- Manage your course load
- Do individual work
- Start assignments early
- Get help early (on Canvas and in TA sessions)

Remember

- *If you:*
  - Do the Assignments
  - Take the Exams
  - Complete the Term Project
  - …you should be able to do just fine

- *Extra stuff is for those of you who*
  - Want to challenge yourself (or show off)
  - Have some extra fun
  - Something cool to show during interviews…

Detailed Class Schedule

See Syllabus posted in Canvas

Online Class Etiquette

~~An Etiquette Guide to Remote Classroom~~

- ~~Raise your hand while asking questions, lower when done~~
- ~~Behave as if you’re in a physical classroom: dress and act accordingly~~
- ~~Academic integrity is academic integrity whether you are on campus or learning remotely. Take your exams following the rules set out for particular classes. If you are taking a test that asks you not to use notes or outside resources, don’t. Others will be following the rules and your decision is violating campus and community norms.~~
- ~~Quick Guide to Online Etiquette~~

Computer Graphics

The Art and Science of creating imagery by computer

Art

Mathematics

Vision

Optics

Biomechanics

Physics

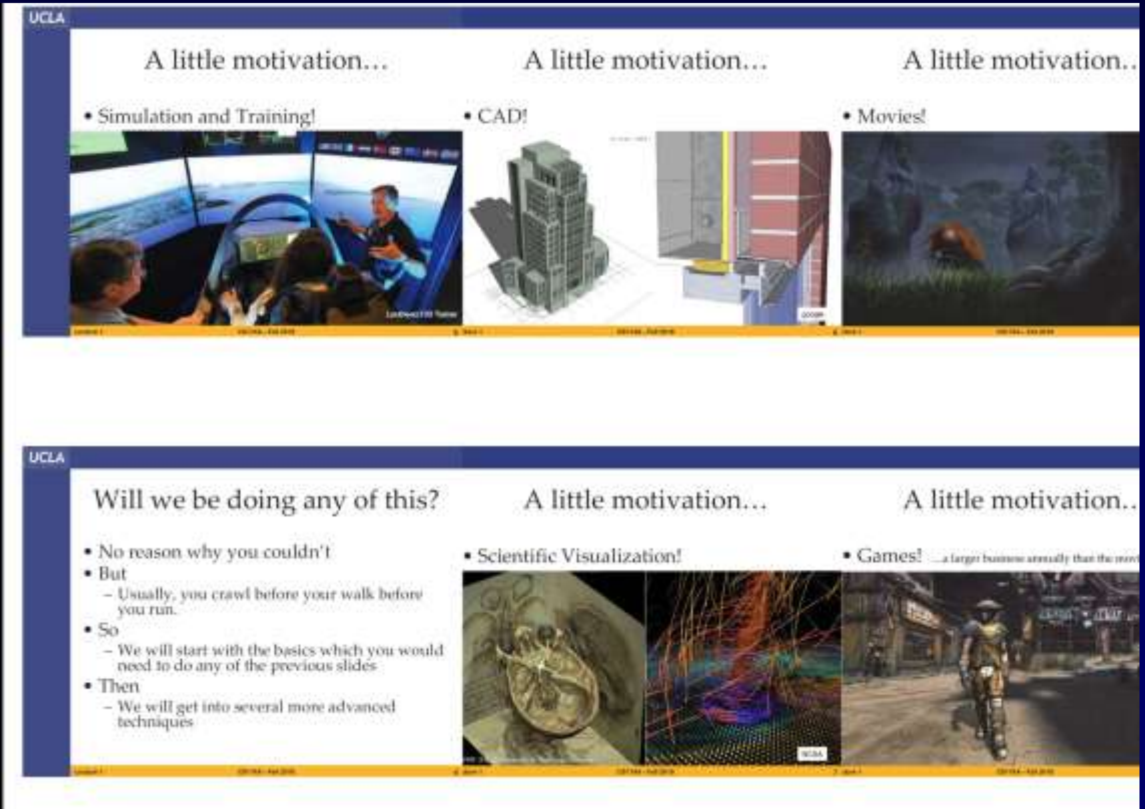
Engineering

Artificial Intelligence

Applications of CG

- ***Entertainment***
  - Films
  - Computer games
  - Virtual reality
- ***Visualization***
  - Scientific visualization
  - Medical visualization
  - Flight simulation
  - Architecture
- ***Education, etc.***

Motivation/local history



UCLA

A little motivation…

- Simulation and Training!

A little motivation…

- CAD!

A little motivation…

- Movies!

UCLA

Will we be doing any of this?

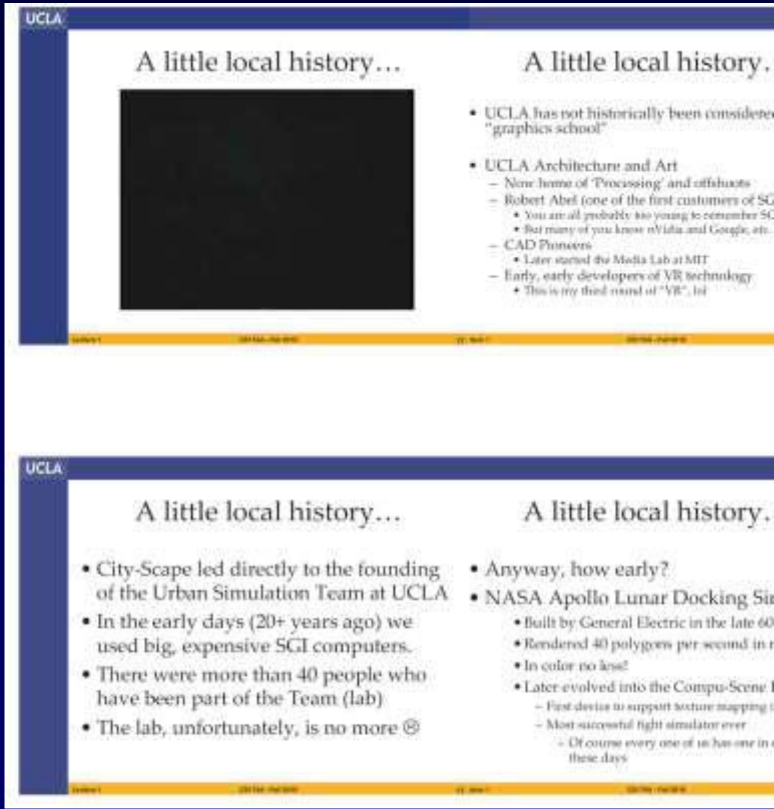
- No reason why you couldn't
- But
  - Usually, you crawl before your walk before you run.
- So
  - We will start with the basics which you would need to do any of the previous slides
- Then
  - We will get into several more advanced techniques

A little motivation…

- Scientific Visualization!

A little motivation…

- Games! …a larger business annually than the movi



UCLA

A little local history…

A little local history.

- UCLA has not historically been considere "graphics school"
- UCLA Architecture and Art
  - Now home of 'Processing' and offshoots
  - Robert Abel (one of the first customers of SG
    - You are all probably too young to remember SG
    - But many of you know nVidia and Google, etc.
  - CAD Pioneers
    - Later started the Media Lab at MIT
  - Early, early developers of VR technology
    - This is my third round of "VR", lol

UCLA

A little local history…

- City-Scape led directly to the founding of the Urban Simulation Team at UCLA
- In the early days (20+ years ago) we used big, expensive SGI computers.
- There were more than 40 people who have been part of the Team (lab)
- The lab, unfortunately, is no more ☹

A little local history.

- Anyway, how early?
- NASA Apollo Lunar Docking Si
  - Built by General Electric in the late 60
  - Rendered 40 polygons per second in r
  - In color no less!
  - Later evolved into the Compu-Scene I
    - First device to support texture mapping
    - Most successful flight simulator ever
      - Of course every one of us has one in these days

History

- 2000 B.C.
  - *orthographic projection*
- 1400s
  - *Perspective: Italian Renaissance*
- 1600s
  - *coordinate systems: Descartes*
  - *optics: Huygens*
  - *optics, calculus, physics: Newton*

History

- ***1897 oscilloscope: Braun***
- ***1950-1970***
  - computers with vector displays
- ***1966***
  - first true raster display
- ***1993***
  - 1200x1200, 500k triangles/sec, 36-bit color, stereo, texture mapping… all at 60Hz
- ***1995***
  - feature-length CG films
- ***Today…still rapidly evolving***

Genesis of Computer Graphics and Interactive Techniques

- ***A PhD project at MIT in the early 1960s***
- ***Ivan E. Sutherland, 1963***
  - “Sketchpad, a man-machine graphical communication system”

Trivia

https://design.osu.edu/carlson/history/timeline.html

When was the term “Computer Graphics” first stated?

William Fetter of Boeing coins the term "computer graphics" for his human factors cockpit drawings 1960

1. When was the Graphical User Interface developed?

   GUI developed by Xerox (Alan Kay) 1969

2. When was Tron released?

   Disney contracts Abel, III, MAGI and DE to create computer graphics for the movie Tron released in 1981

Trivia

4. Which is the first **animated** movie to employ CG?

   “The Great Mouse Detective” (1986) was the first animated film to be aided by CG

5. When was the game “Doom” released?

   1993

6. Which are the best selling games of all time?

   Minecraft, Grand Theft, Tetris

Trivia

7. Which are some of latest CG animated movies?

   Minion 3D, Encanto, Onward, Frozen 2, Lion King, Raya and the Last Dragon, Vivo, Luca

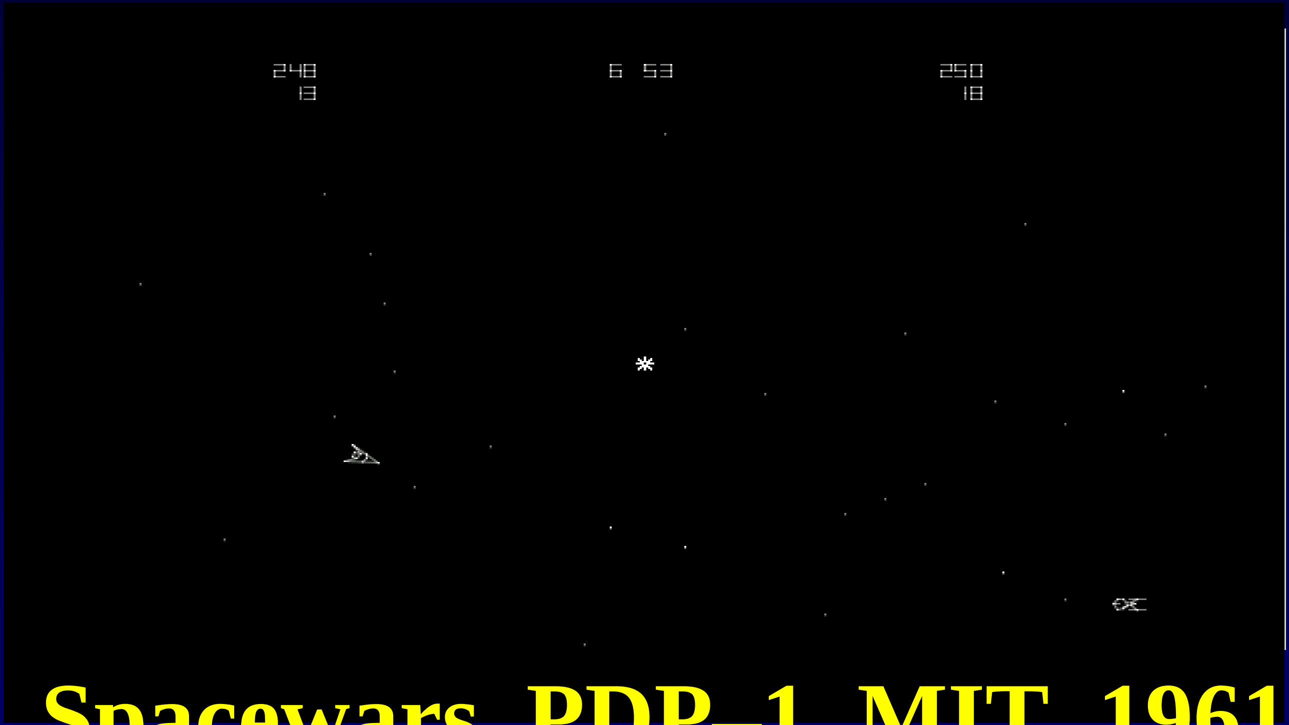
8. Which is bigger in terms of gross revenue, the game industry or the (Hollywood) movie industry?

   The game industry

9. Which is your favorite animated movie?

   Tangled, Moana, Frozen, Incredibles, Lion King

The First Computer Game?



Spacewars, PDP–1, MIT, 1961

1972

The First “Computer” Game – 1958 !!

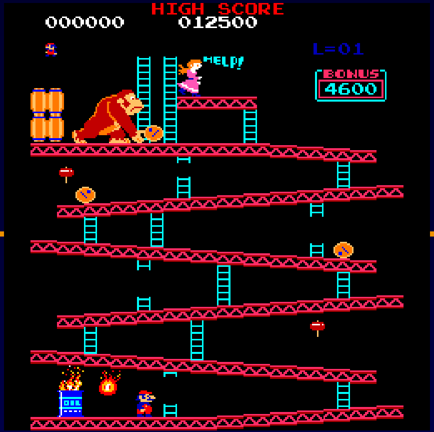
“Tennis for Two”

https://www.youtube.com/watch?v=s2E9iSQfGdg

Metal Gear Solid 4

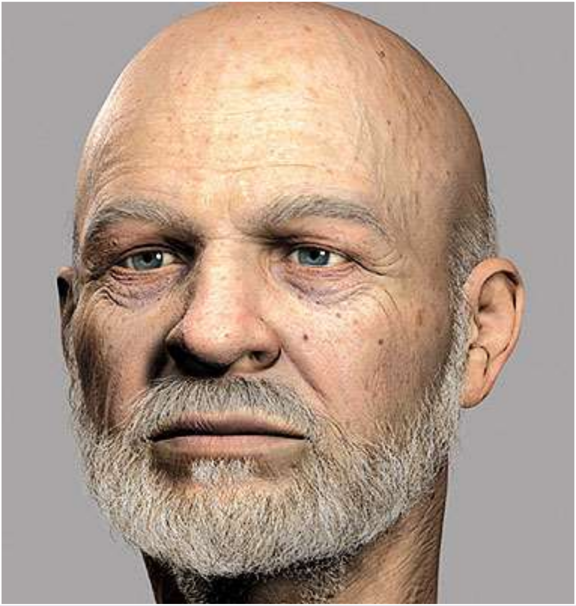
Games

Focus on interactivity



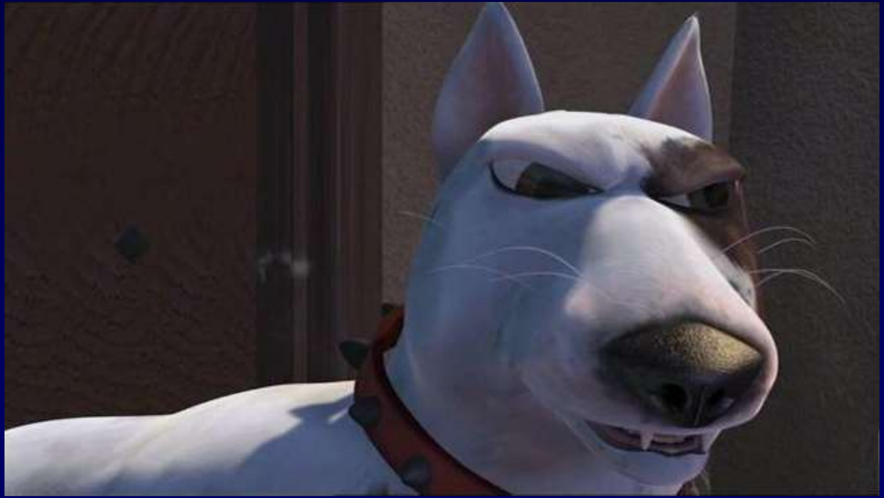
Movies

To reality and beyond !

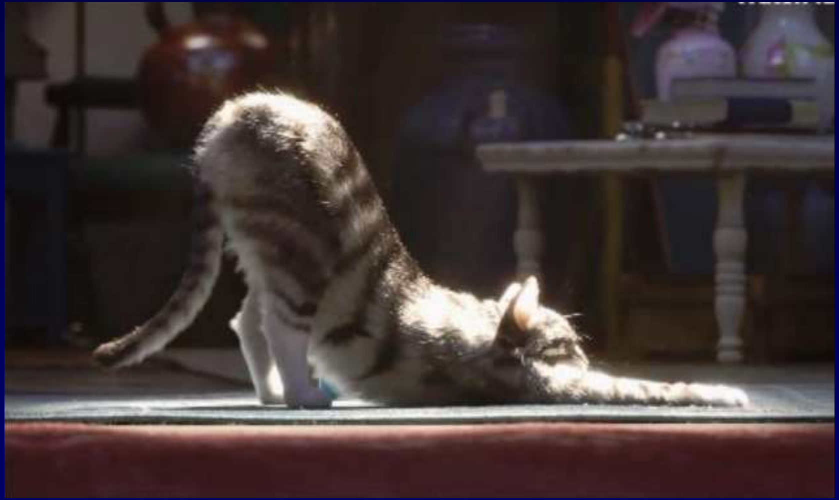


Movies

Toy Story 1 (1995)

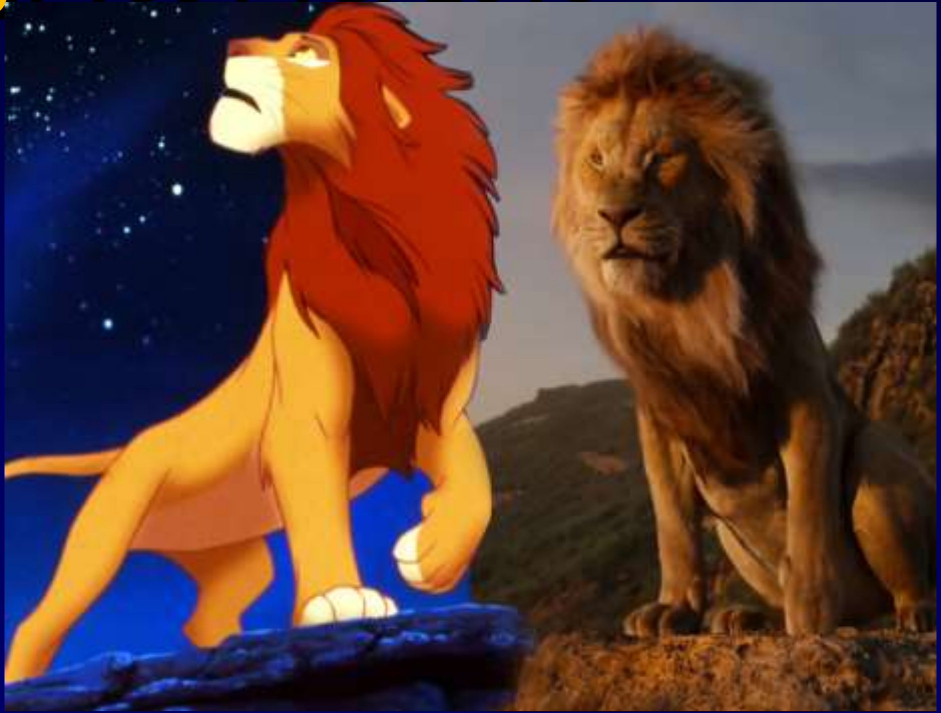


Toy Story 4 (2019)



Movies

Lion King: 1994 vs. 2019



Movies

Special effects

Movies

Digital Compositing

CG Axe
Base Color

Digital Compositing

Cartoons

Computer-Aided Design

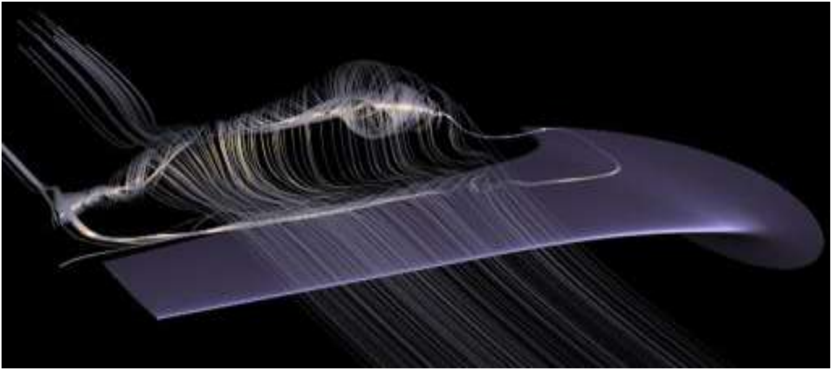
Precision modeling

Engineering

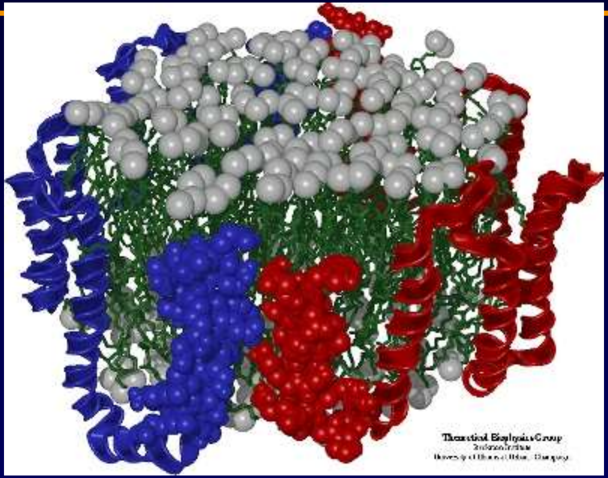
Modeling Clay

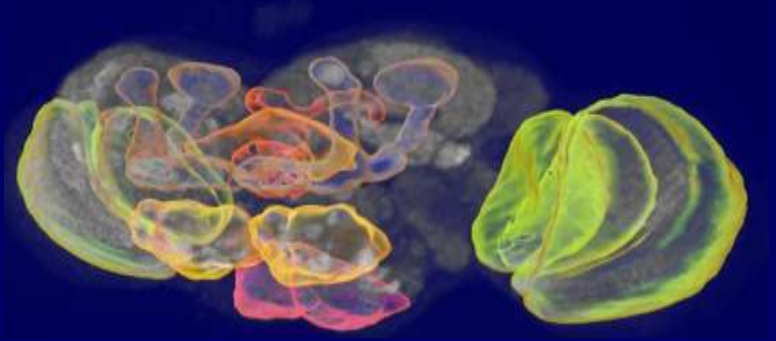
Computer-Aided Design

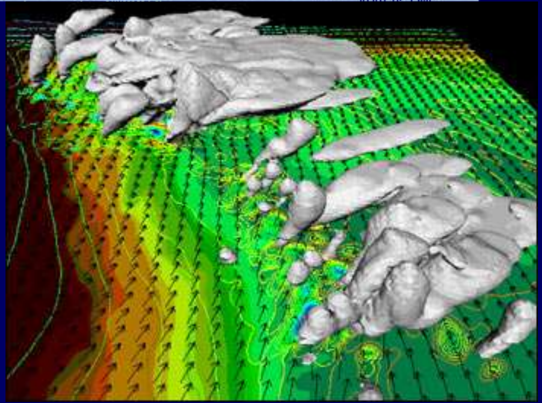
- ***It’s not just about visualization***
  - Simulation is also useful



Visualization: Scientific


Theoretical Biophysics Group
Beckman Institute
University of Illinois at Urbana-Champaign





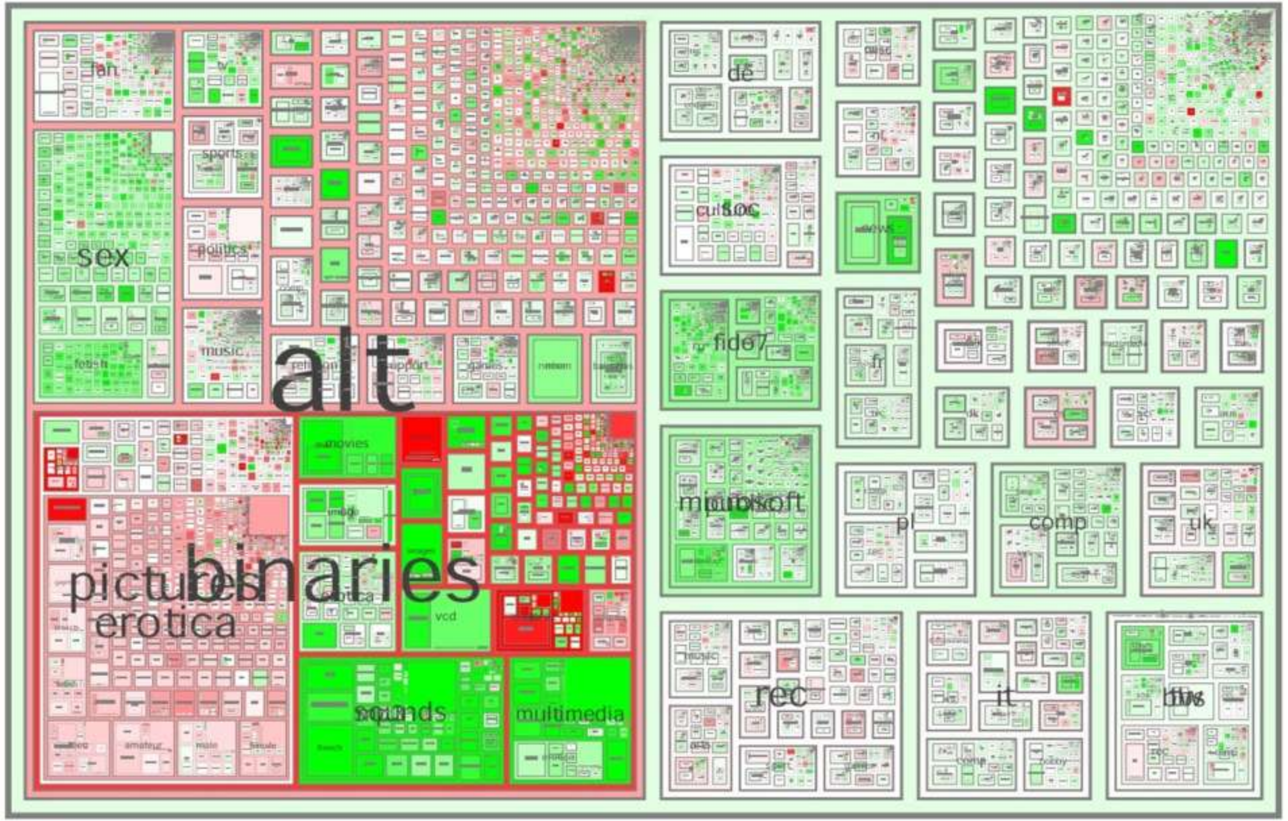
Visualization: Architectural

[redacted]

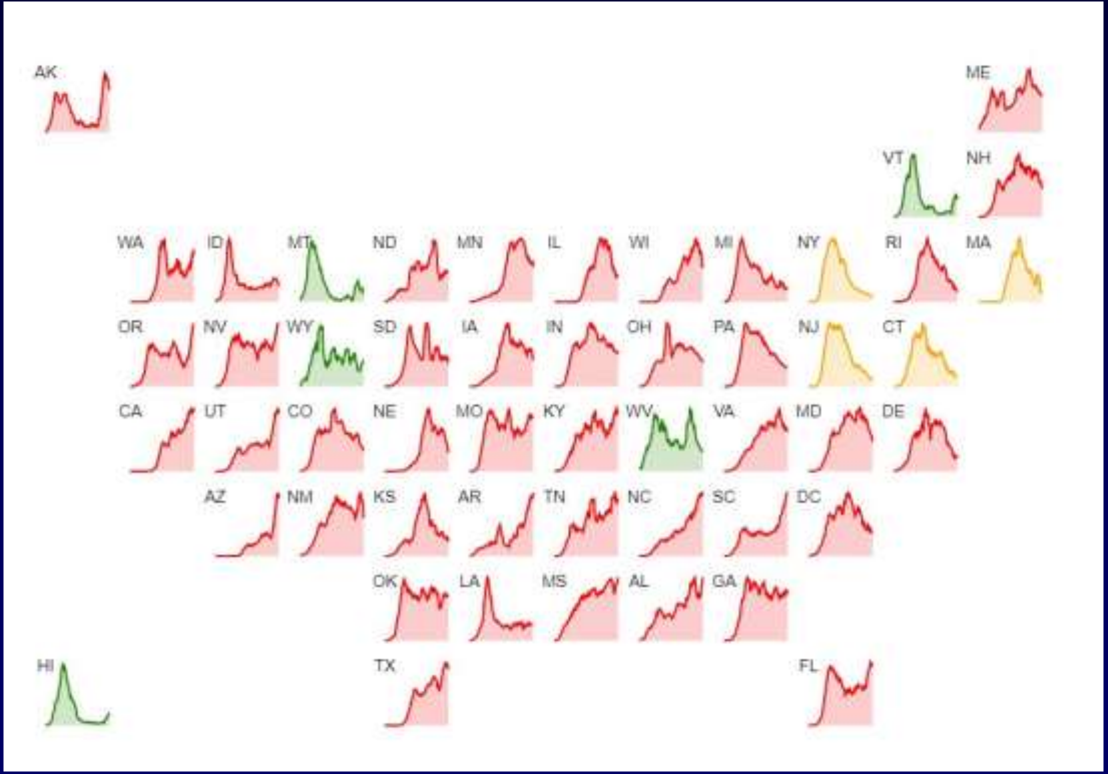
http://www.diamondschmitt.com/

Visualization: Info

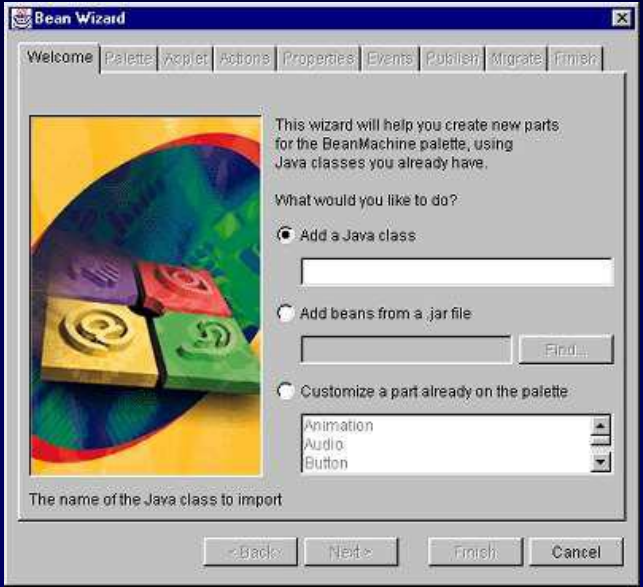
Smith and Fiore



Visualization: Info (Covid Tracker)



Graphical User Interfaces



This wizard will help you create new parts
for the BeanMachine palette, using
Java classes you already have.

What would you like to do?

● Add a Java class

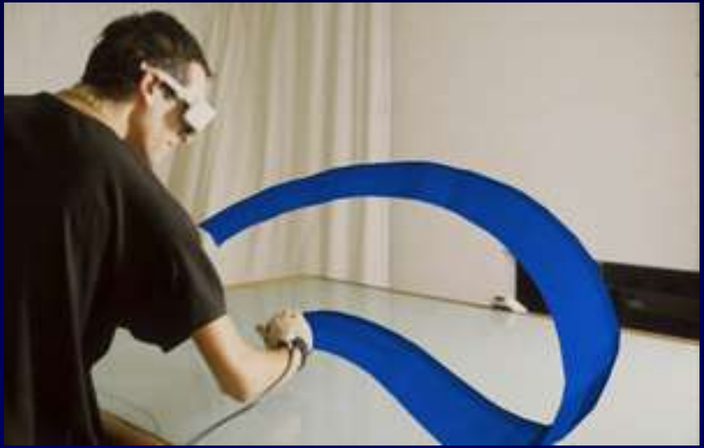
○ Add beans from a .jar file

○ Customize a part already on the palette

Animation
Audio
Button

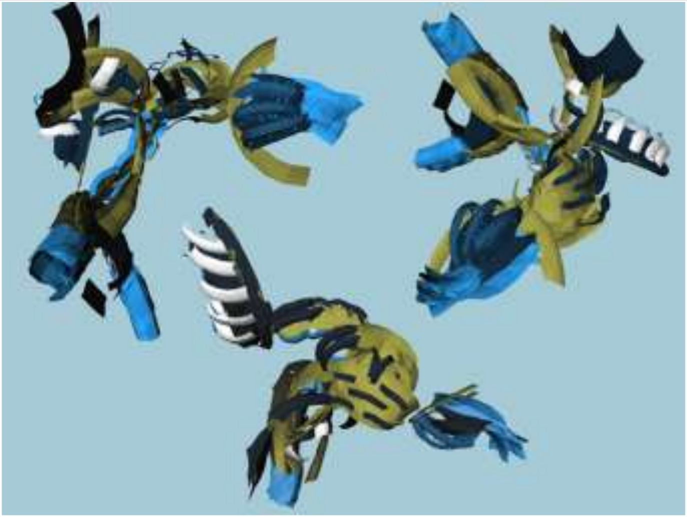
The name of the Java class to import

WIMP



Steven Schkolne

Art

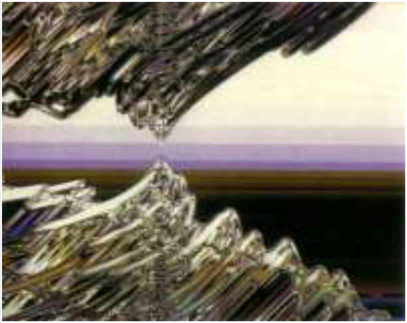


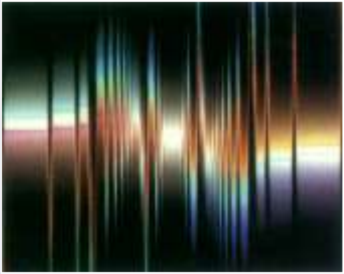
Steven Schkolne

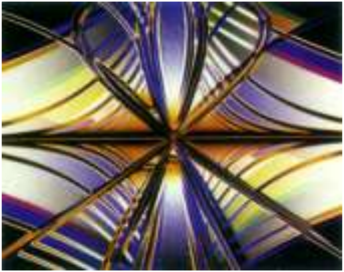
Digital Art

Genetically evolved

Carl Sims







Digital Art

By Jason Salavon

The Top-Grossing Film of All Time

^
2nd

“Titanic”

Basic Elements

Modeling

Animation

Rendering

Interaction

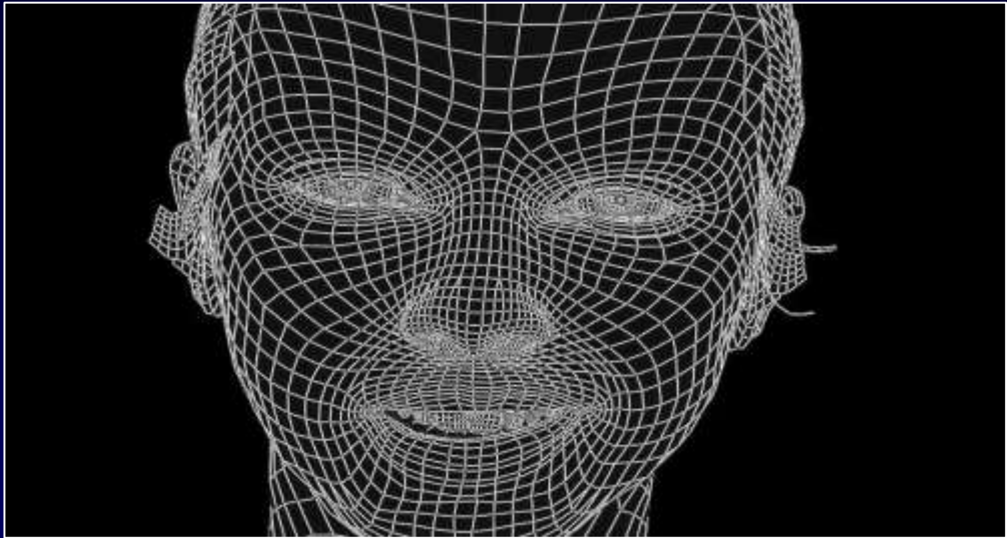
Basic Elements

- ***Modeling***
  - How do we model (mathematically represent) objects?
  - How do we construct models of specific objects?
- ***Animation***
  - How do we represent the motions of objects?
  - How do we give animators control of this motion?
- ***Rendering***
  - How do we simulate the real-world behavior of light?
  - How do we simulate the formation of images?
- ***Interaction***
  - How do we enable humans and computers to interact?
  - How do we design human-computer interfaces?

Modeling

Primitives

- 3D points
- 3D lines and curves
- Surfaces (BREPs): polygons, patches
- Volumetric representations
- Image-based representations



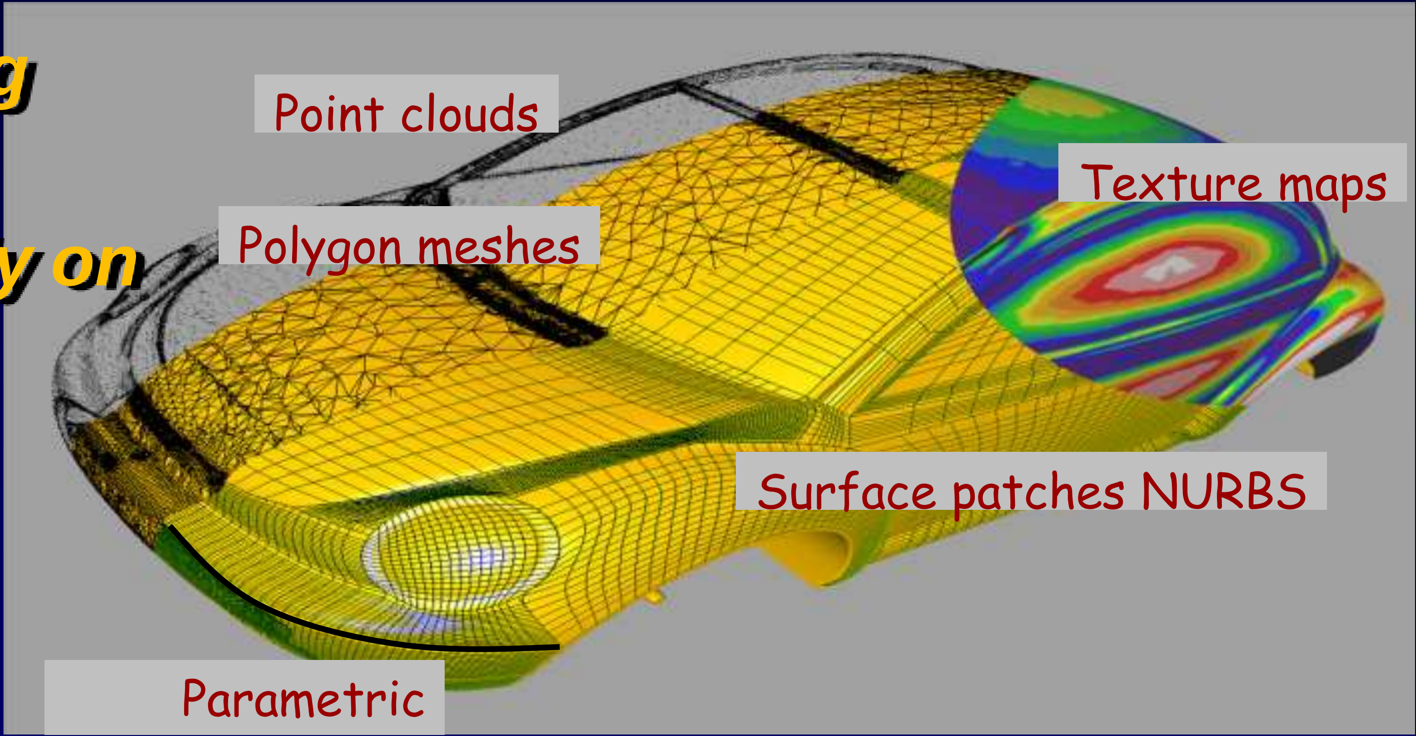
Attributes

- Color, texture maps
- Lighting properties

Geometric transformations

Modeling

Representing objects geometrically on a computer



Point clouds

Polygon meshes

Texture maps

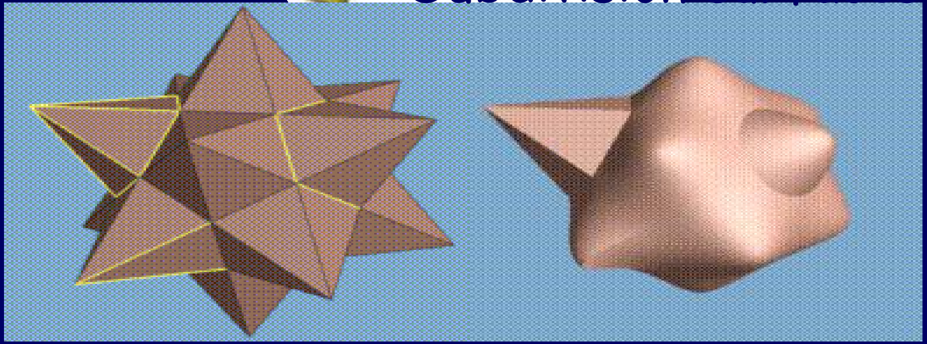
Surface patches NURBS

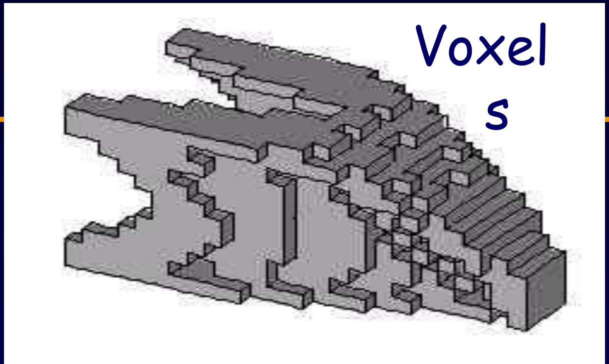
Parametric curves

Modeling

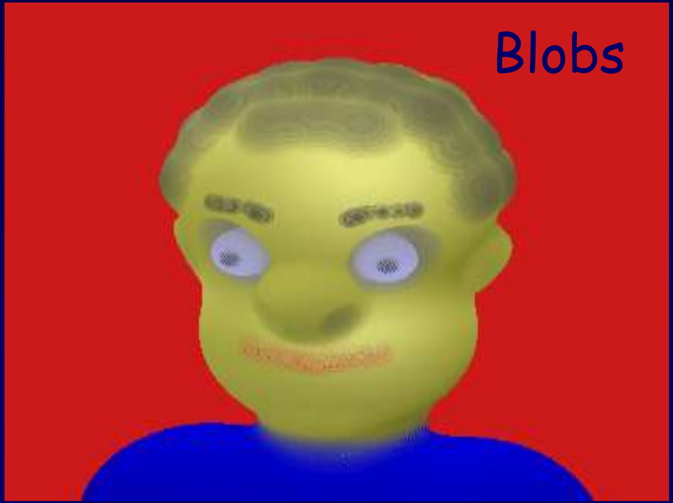
Alternative representations

Subdivision surfaces





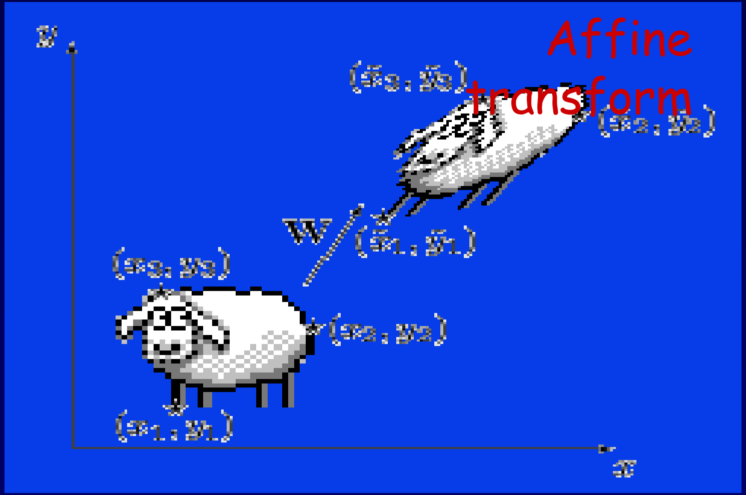
Voxels



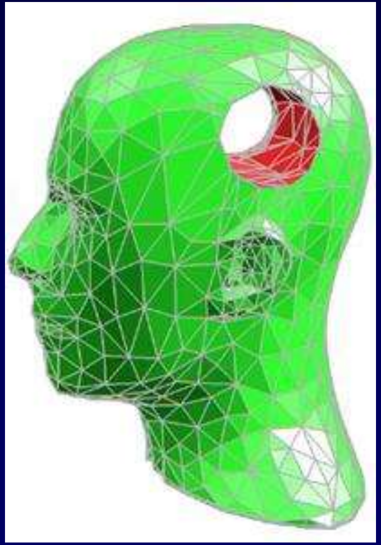
Blobs

Modeling

Altering geometric models

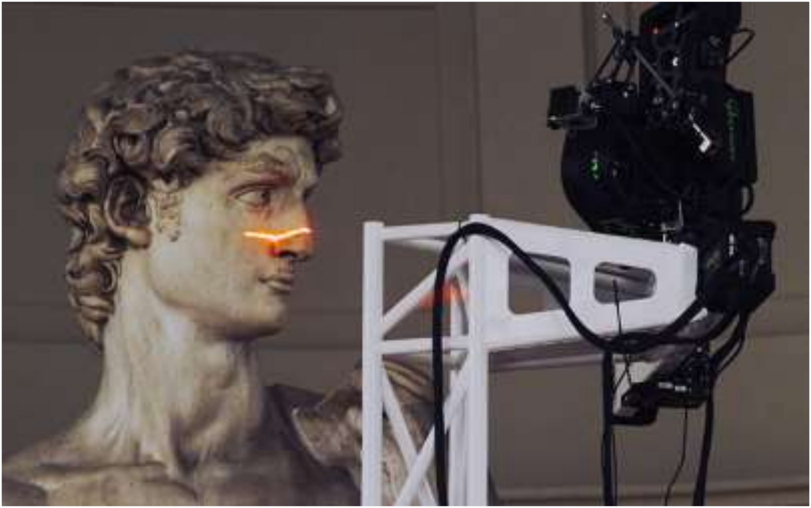


Affine transform



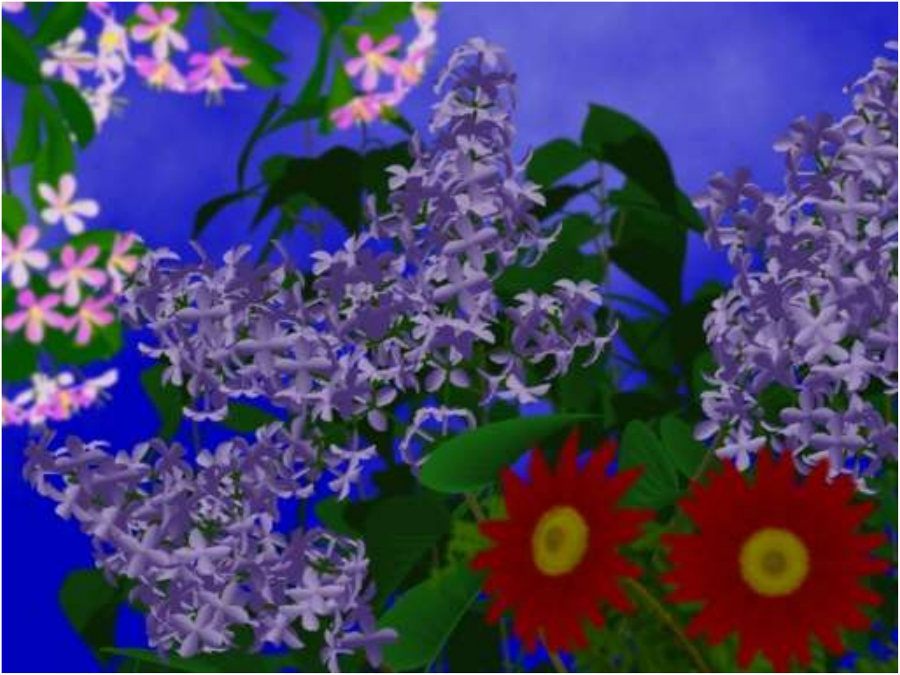
Ying, Kristjansson, Biermann, Zorin

Scanning Shapes



Digital Michaelangelo Project

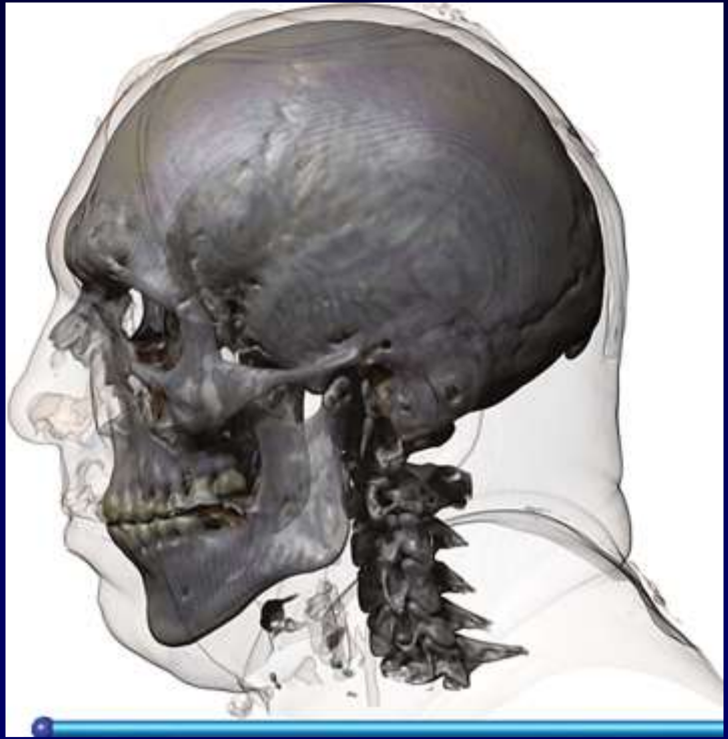
Plant Modeling



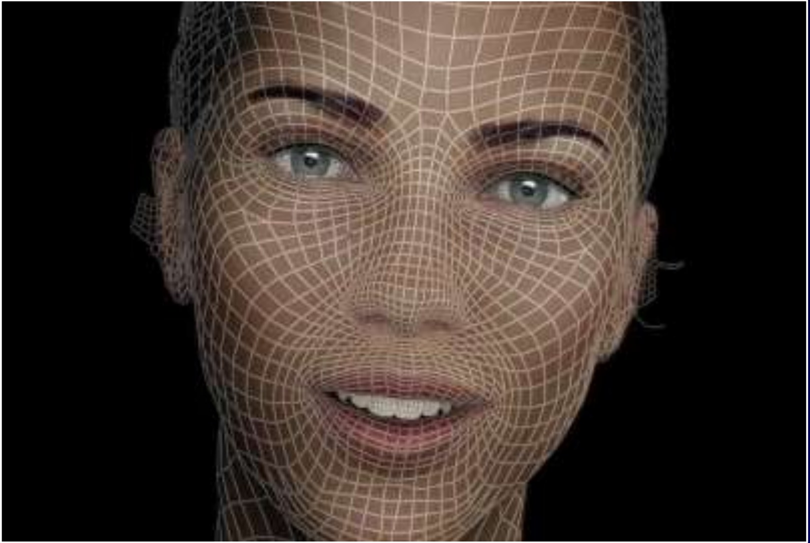
Plant Modeling

Volume Modeling

- ***CT Scans***
- ***MRI***
- ***Ultrasound***
- ***PET Scans***



Rendering



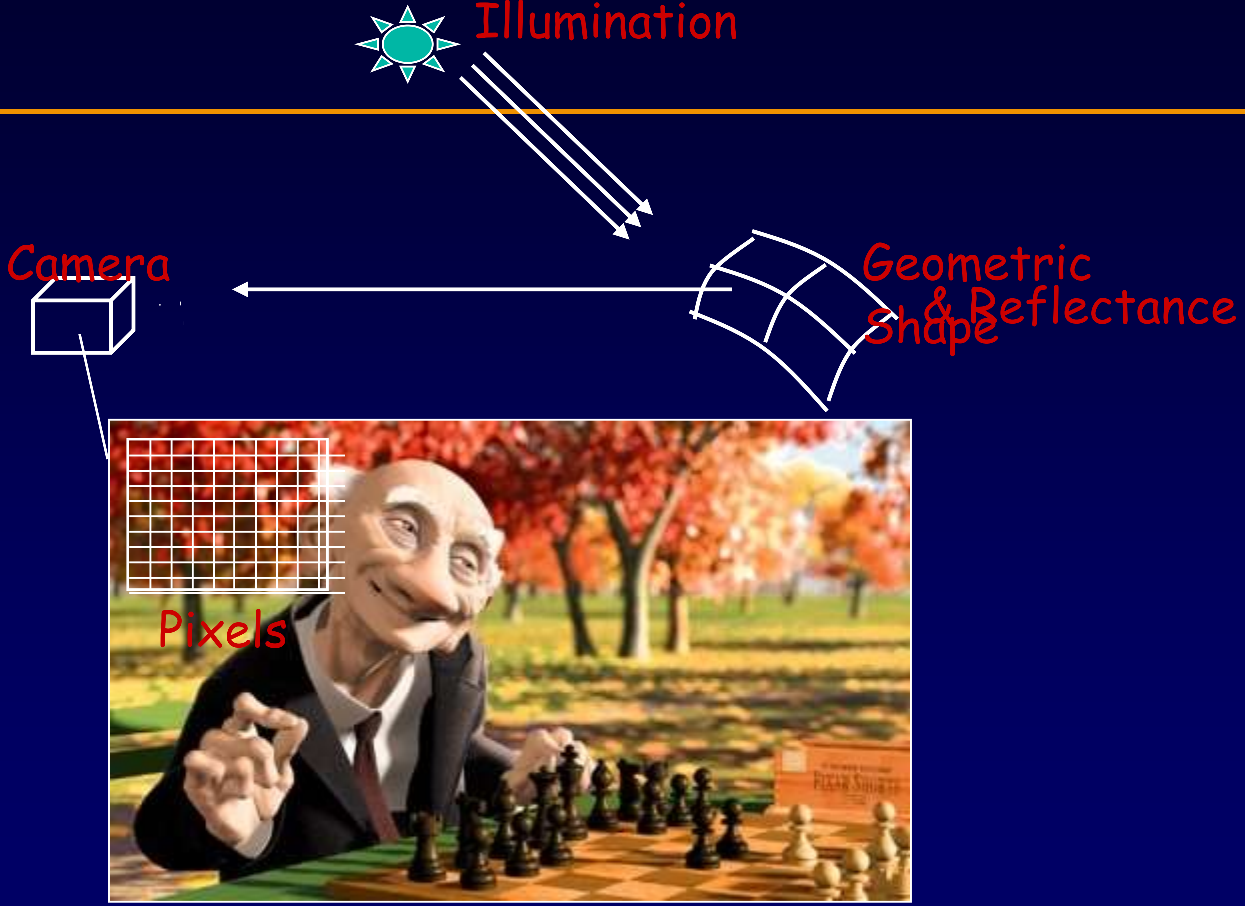
Visibility

Simulating light propagation

- Reflection
- Absorption
- Scattering
- Emission
- Interference

Rendering

Key elements



Illumination

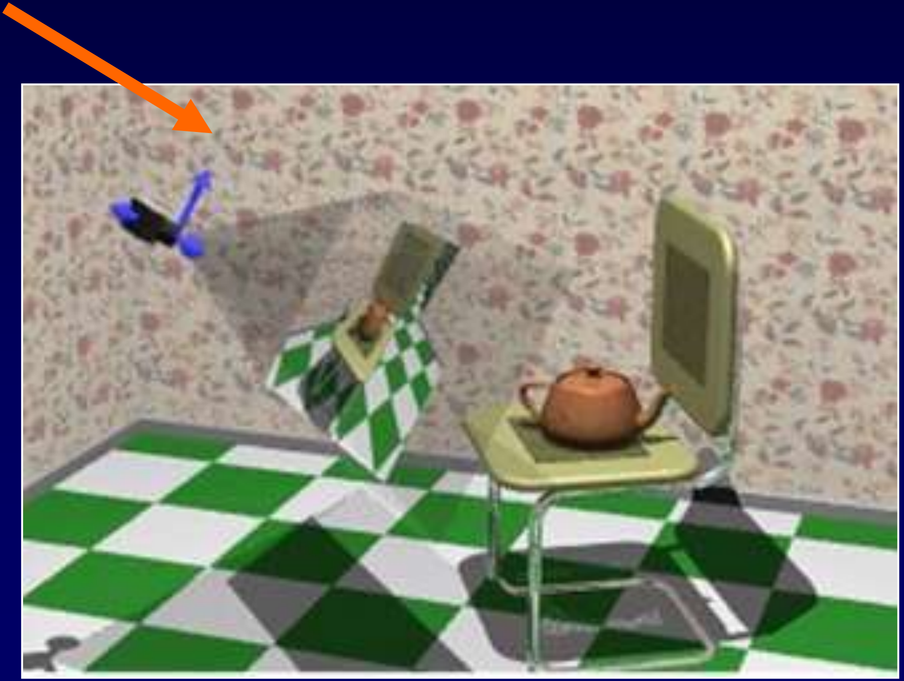
Camera

Geometric Shape

& Reflectance

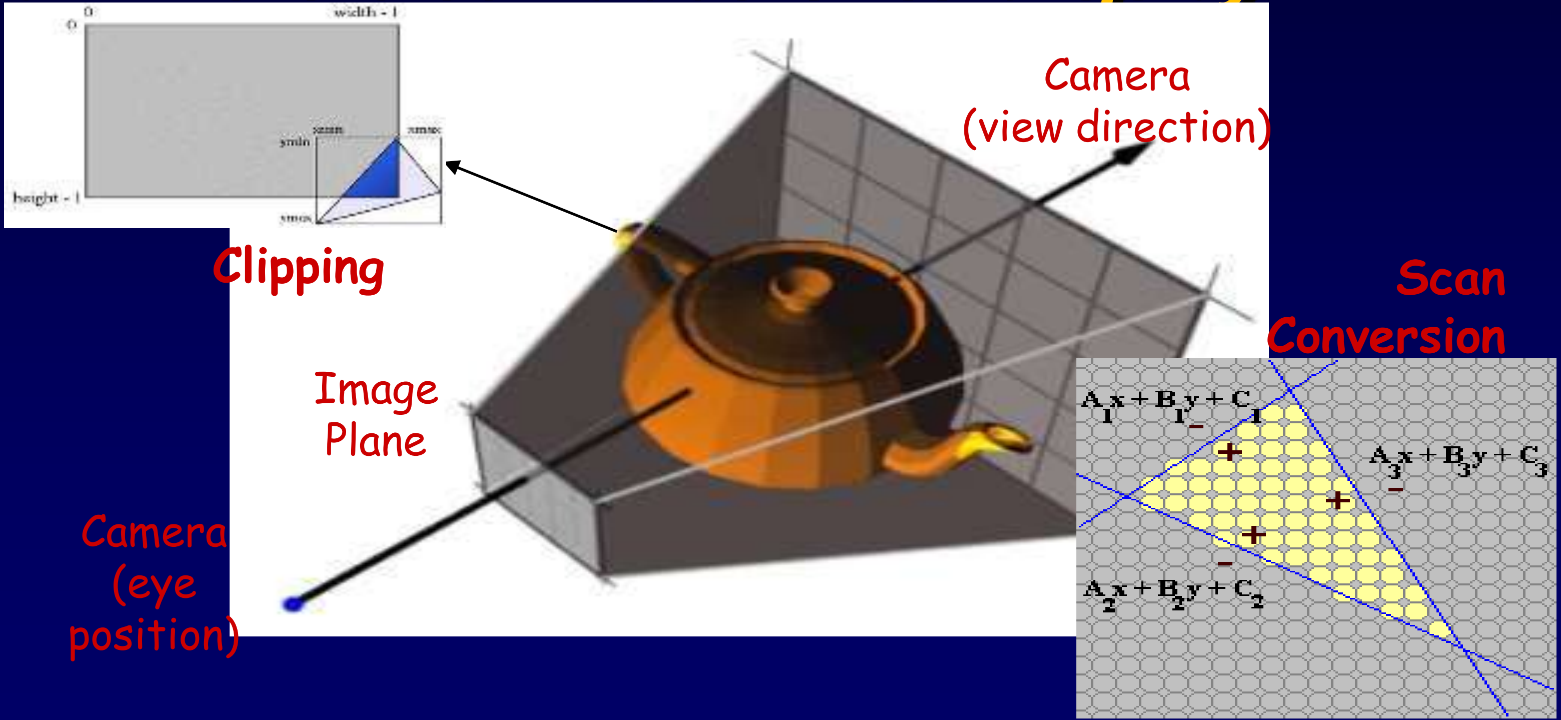
Pixels

Camera



Rendering

Draw visible surfaces onto display



Camera (view direction)

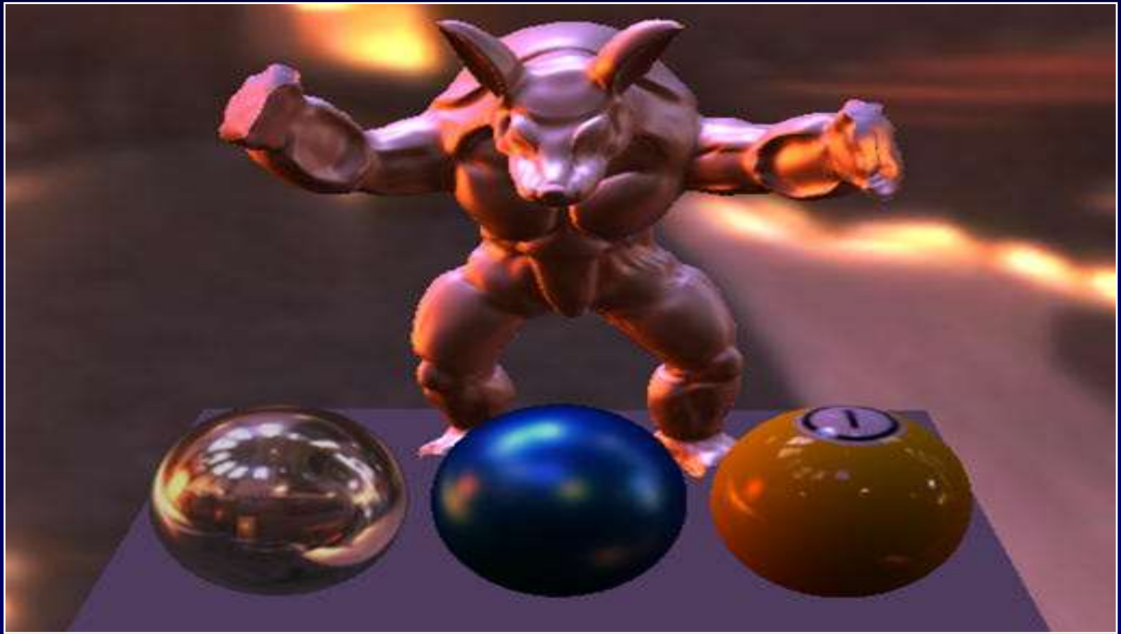
Clipping

Scan Conversion

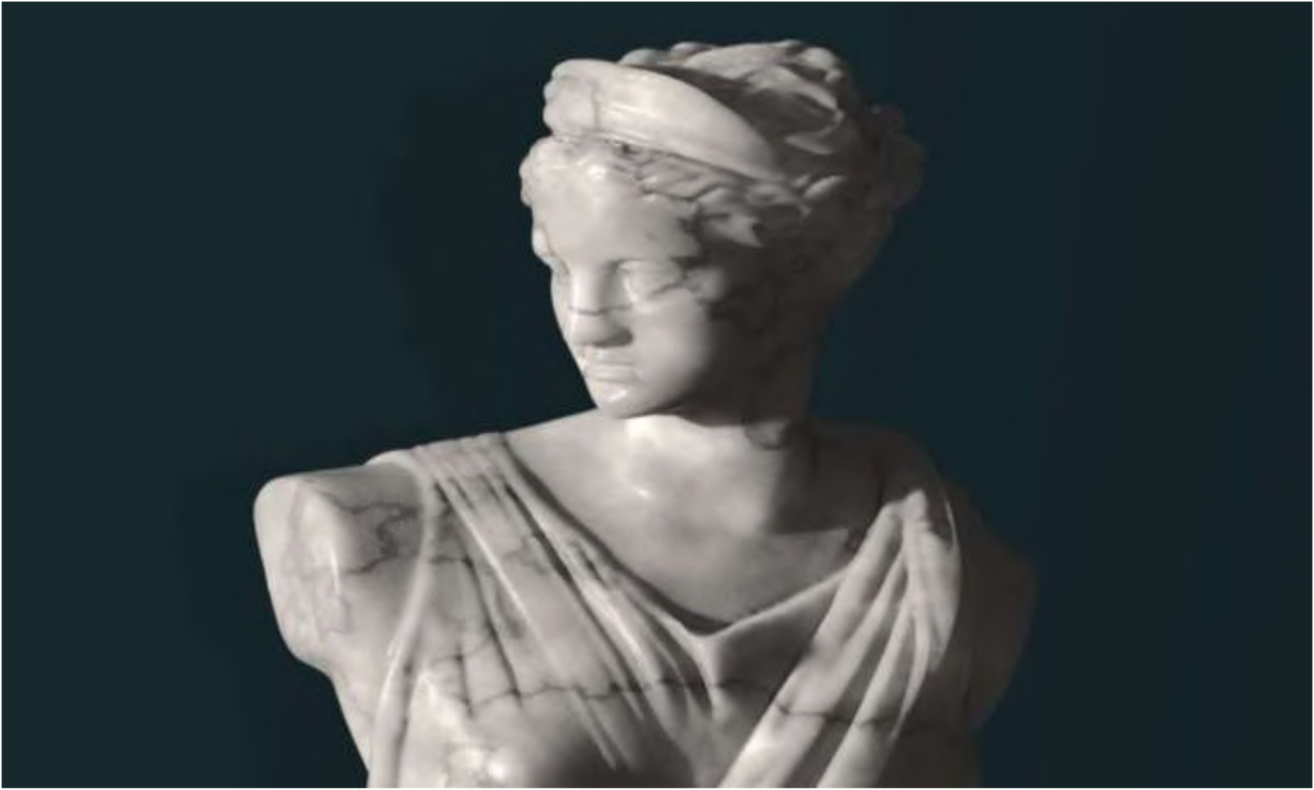
Image Plane

Camera (eye position)

Reflectance Modeling

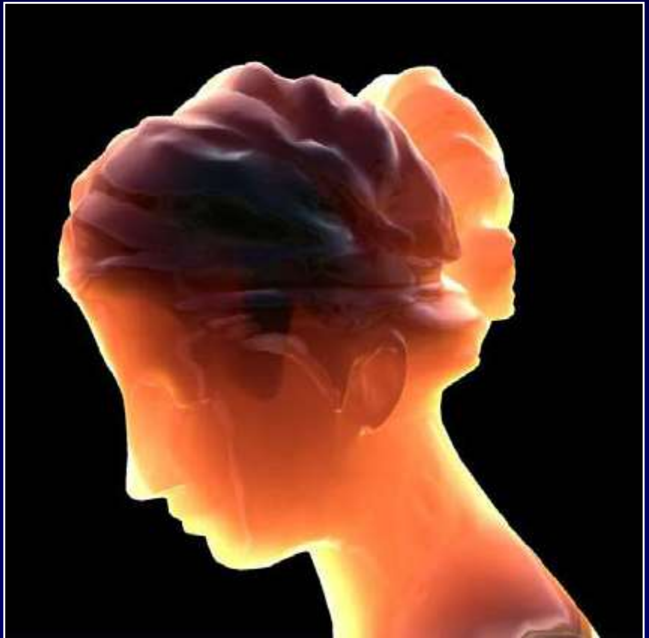


Complex Reflectance



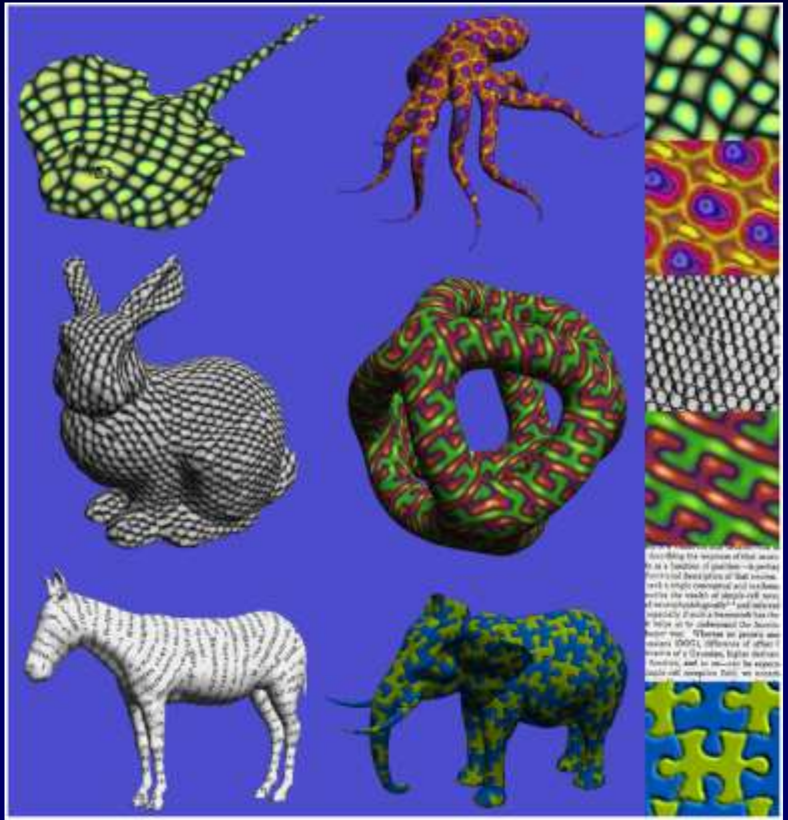
Subsurface Scattering

Translucency and varied levels of light penetration can be created using subsurface scattering effects



Texture

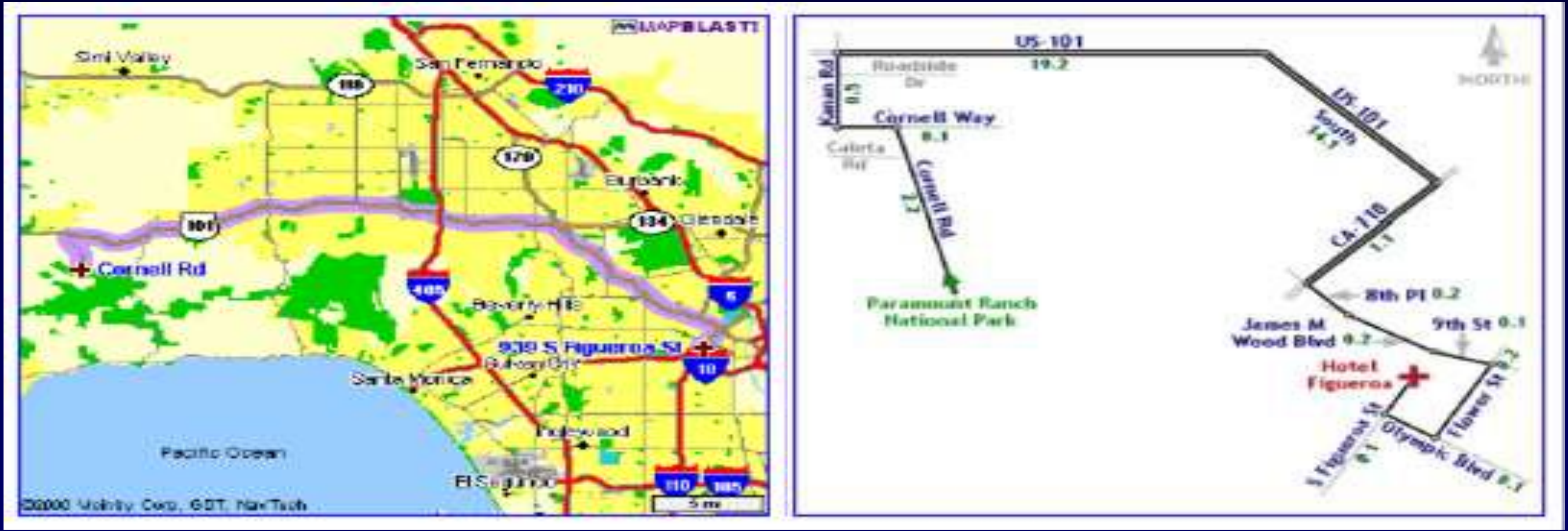
Multilevel texture synthesis



Non-Photorealistic Rendering

Aaron Hertzmann

Rendering: Information



Animation

Keyframe animation

- Flipbook animation

Motion capture

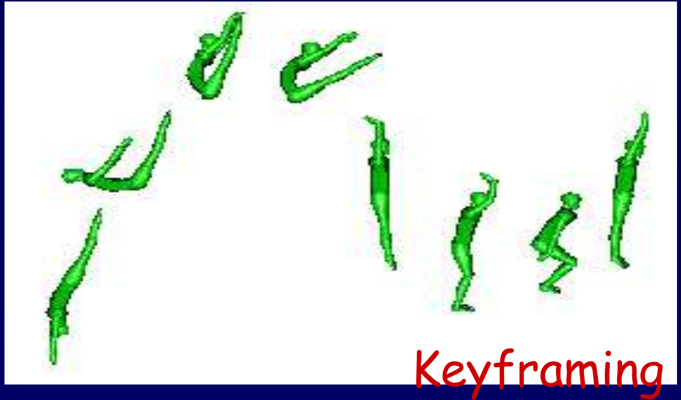
Procedural animation

- Physics-based animation
- Behavioral animation
- Emotion-based animation

Animation

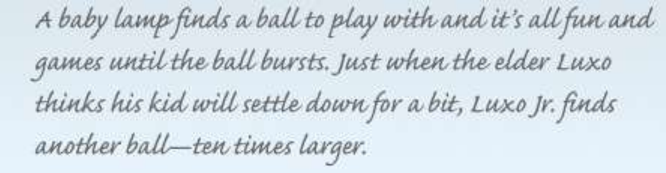
Keyframe animation

<u>Flipbook Animation</u>

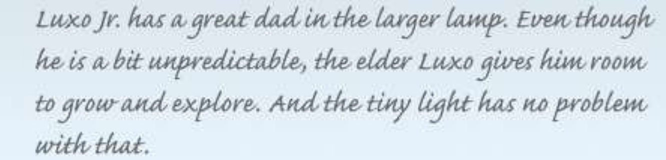


Animation

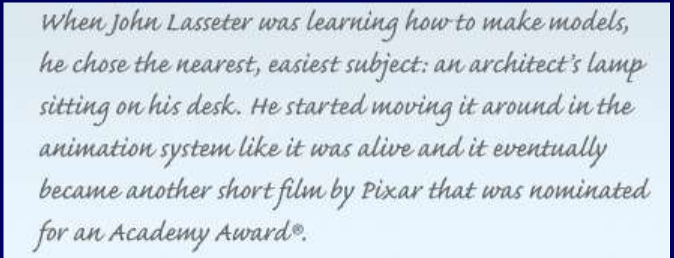
Pixar: "Luxo Jr." (1986)



A baby lamp finds a ball to play with and it's all fun and games until the ball bursts. Just when the elder Luxo thinks his kid will settle down for a bit, Luxo Jr. finds another ball—ten times larger.



Luxo Jr. has a great dad in the larger lamp. Even though he is a bit unpredictable, the elder Luxo gives him room to grow and explore. And the tiny light has no problem with that.

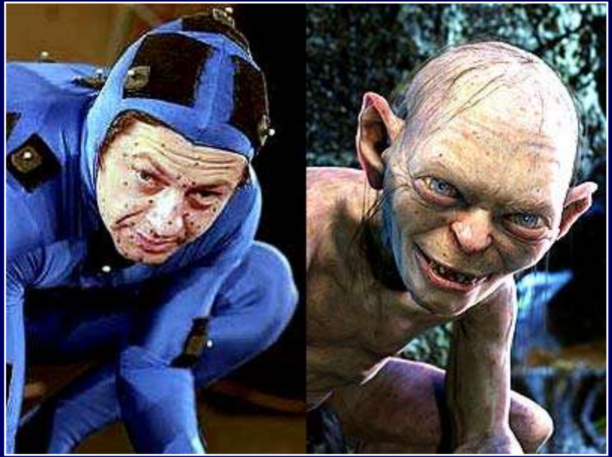


When John Lasseter was learning how to make models, he chose the nearest, easiest subject: an architect's lamp sitting on his desk. He started moving it around in the animation system like it was alive and it eventually became another short film by Pixar that was nominated for an Academy Award®.

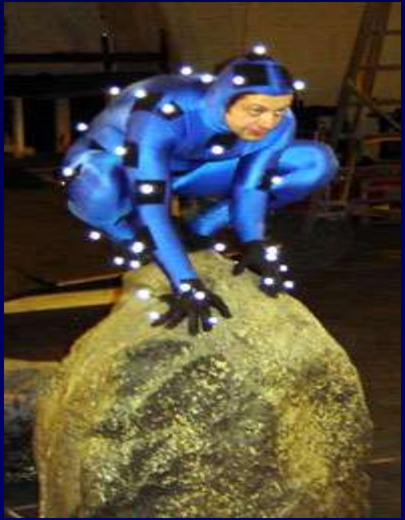
Animation

Motion capture

(c) MMIII New Line Productions, Inc. All Rights Reserved.



Animating Golem in LOTR



Animation

The Animatrix –

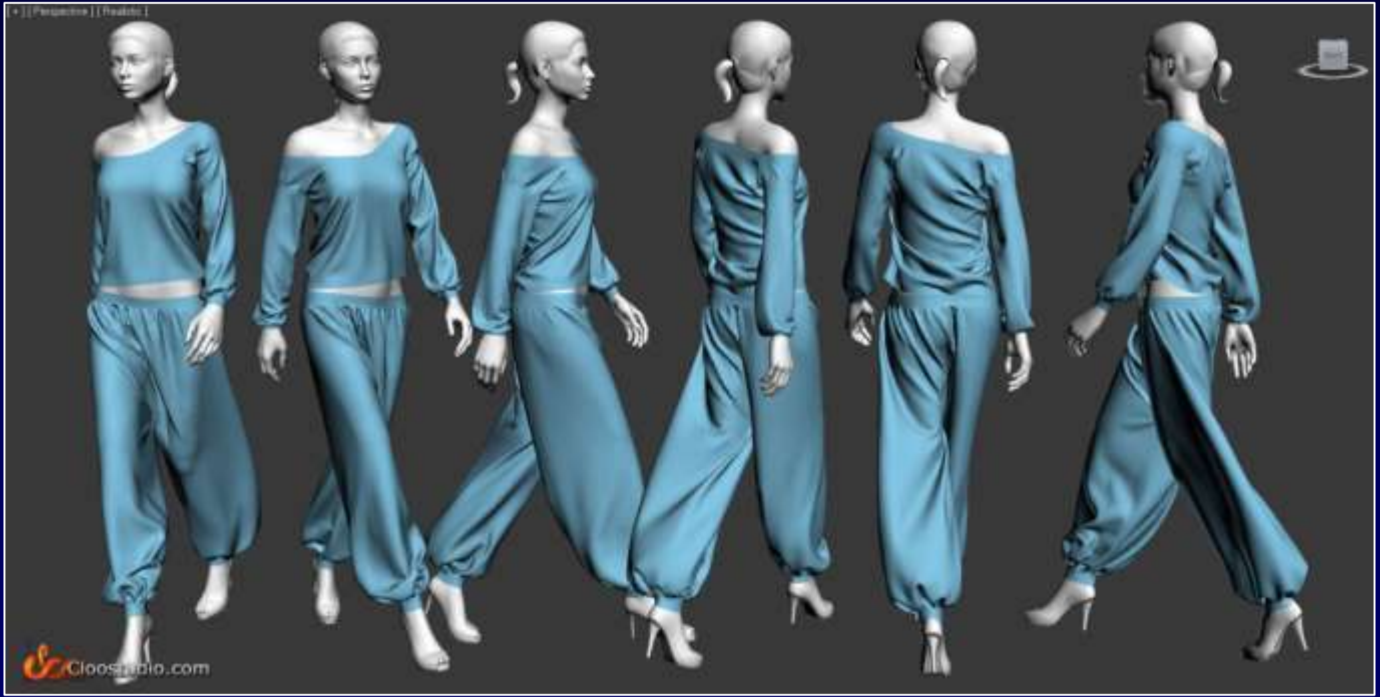
“Final Flight of the Osiris”

Animation

Example: “Geri’s Game” - Pixar

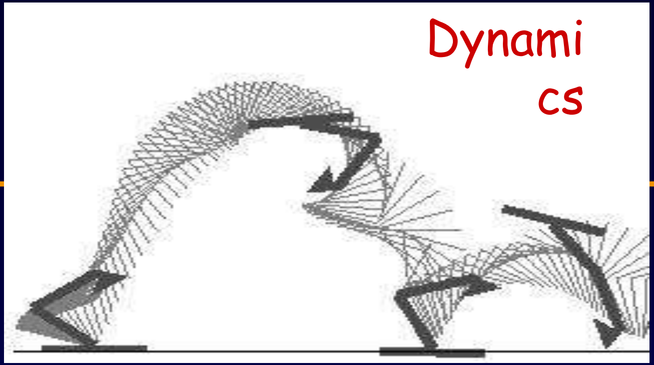
PIXAR SHORTS

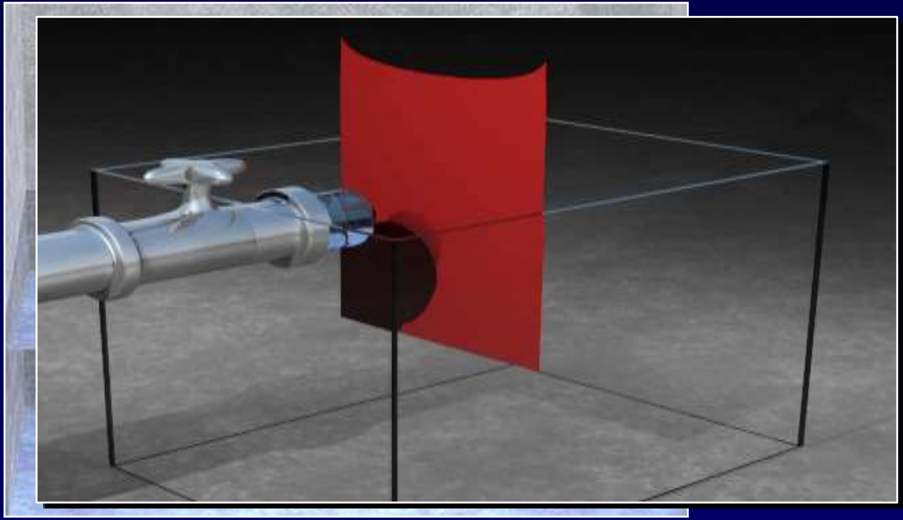
Cloth Simulation



Animation

Physics-based animation





Fluid Simulation

Modeling

- Incompressibility
- Viscocity

Navier-Stokes Equations

$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \mathbf{u}}{\partial t} = \nu \nabla \cdot (\nabla \mathbf{u}) - (\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho} \nabla p + \mathbf{g}$$

$\mathbf{u}$: fluid velocity field

$\mathbf{g}$: gravity

p: pressure

ν: viscosity

ρ: density

Level Sets

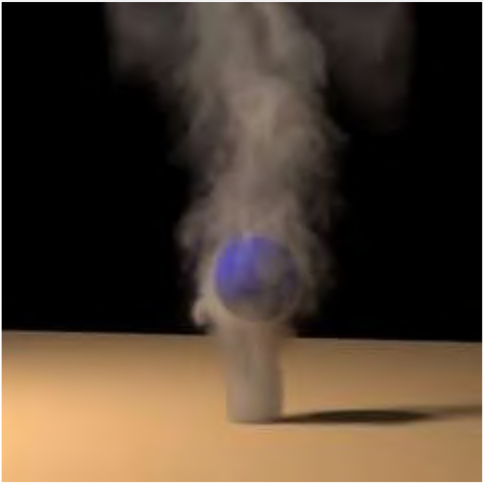
Smoke Simulation

Assumptions

- No viscosity

Rendering

- Photon maps
- Multiple scattering



$$\nabla \cdot \mathbf{u} = 0$$

$$\frac{\partial \mathbf{u}}{\partial t} = (\mathbf{u} \cdot \nabla)\mathbf{u} - \frac{1}{\rho}\nabla p + \mathbf{f}$$

$\mathbf{u}$: smoke velocity field

$\mathbf{f}$: external forces

p: pressure

ρ : density

Animation

Behavioral animation

Ethology

NEW LINE CINEMA

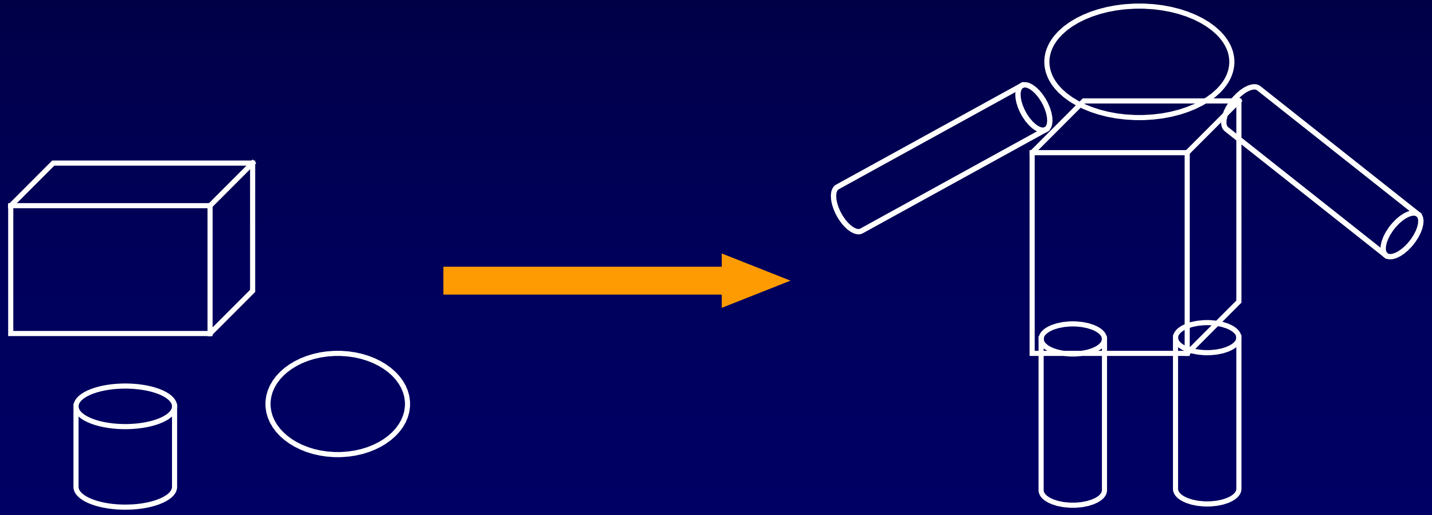
Reality is *Very* Complex

Reality is *Very* Complex

Reality is *Very* Complex

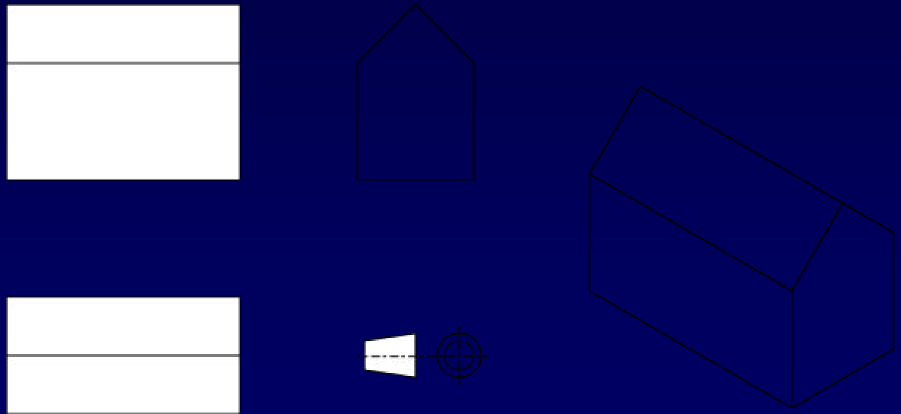
Modeling Transformations

Assembly

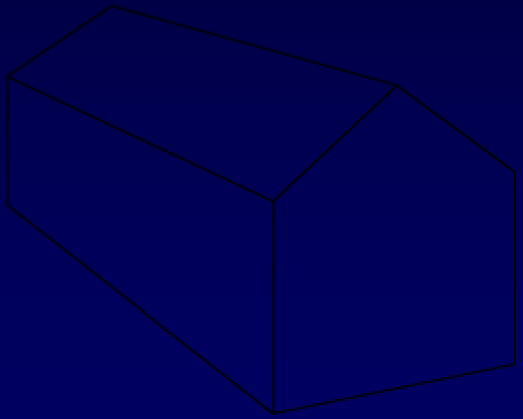


Viewing

Orthographic

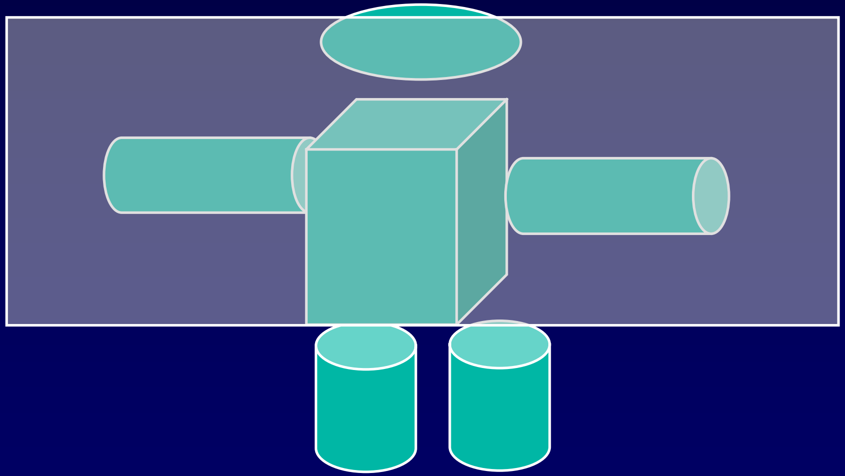


Perspective



Clipping

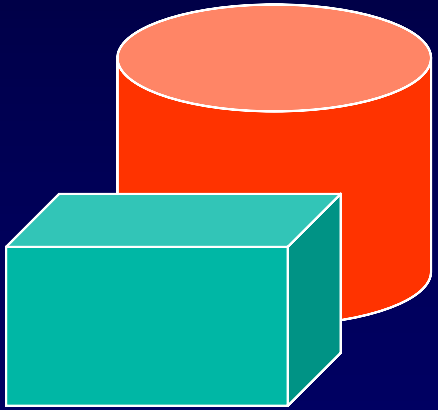
Remove what is not visible



Visibility

Resolve occlusions

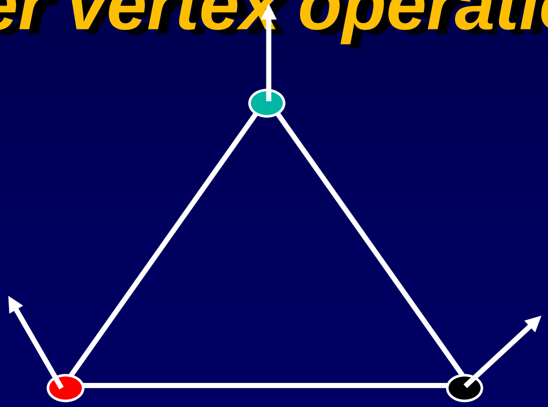
(efficiently)

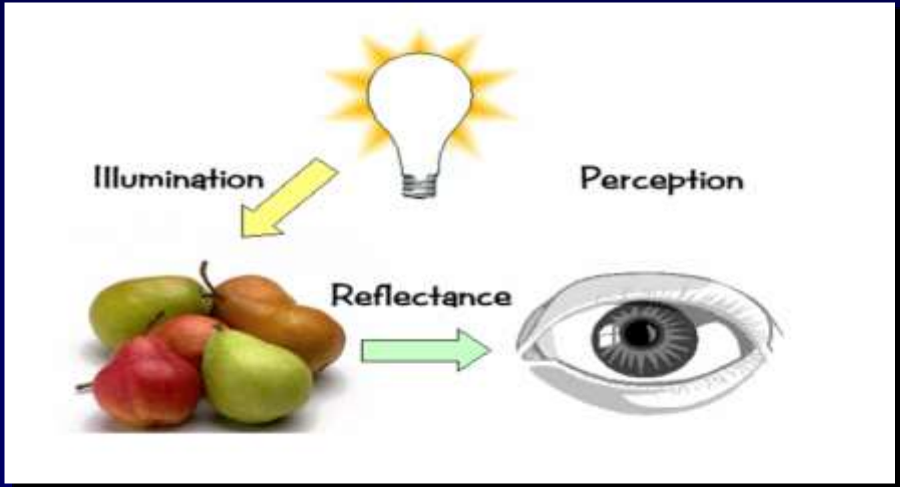


Illumination

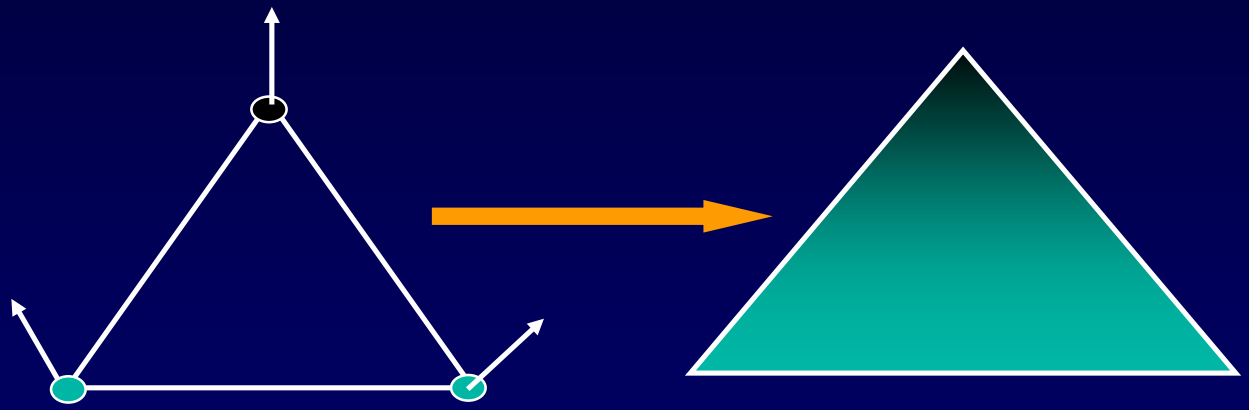
Compute normals and color at vertices

Per vertex operations



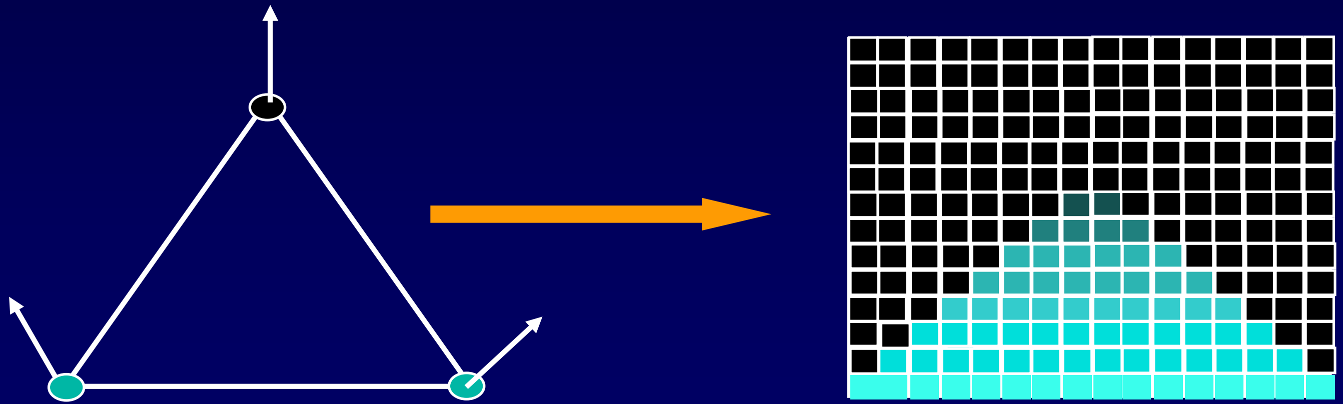


Shading

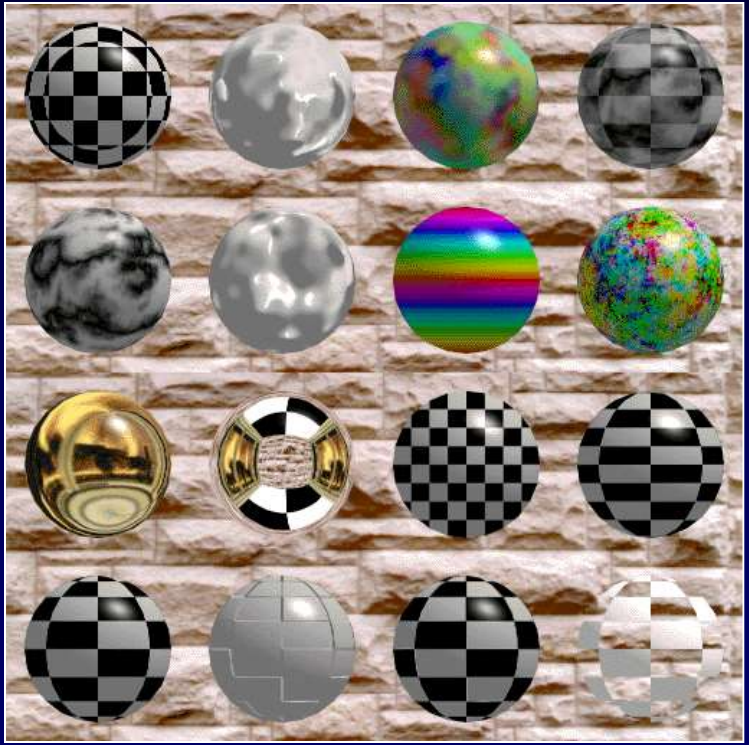


Rasterization

Convert to colored pixels



Texture Mapping



Other Issues

Shadows

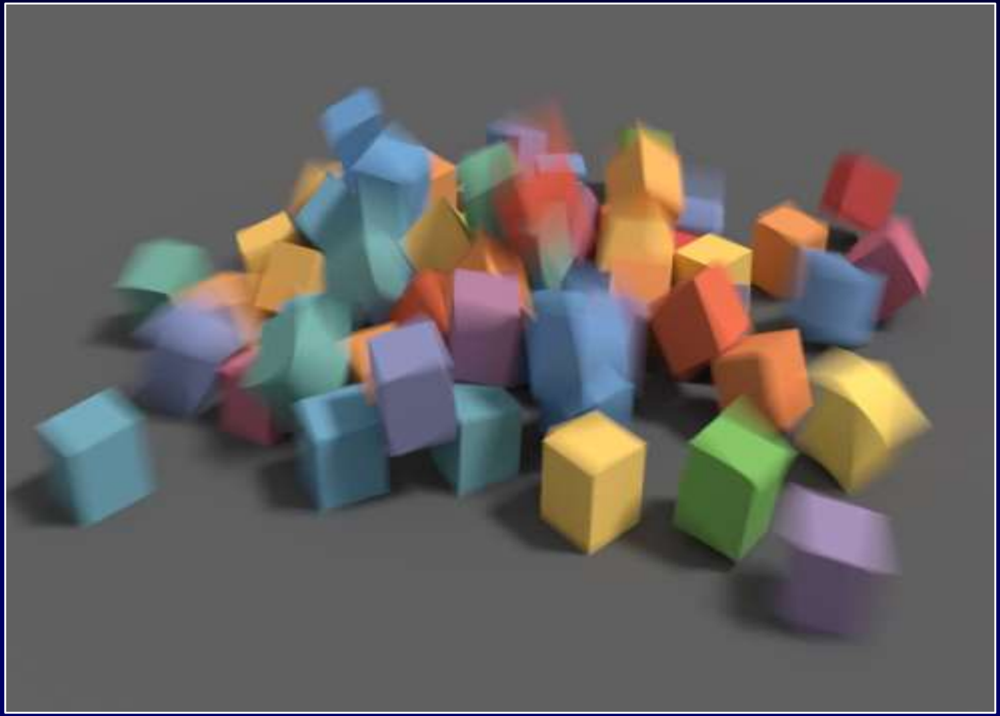
Participating media

Subsurface scattering

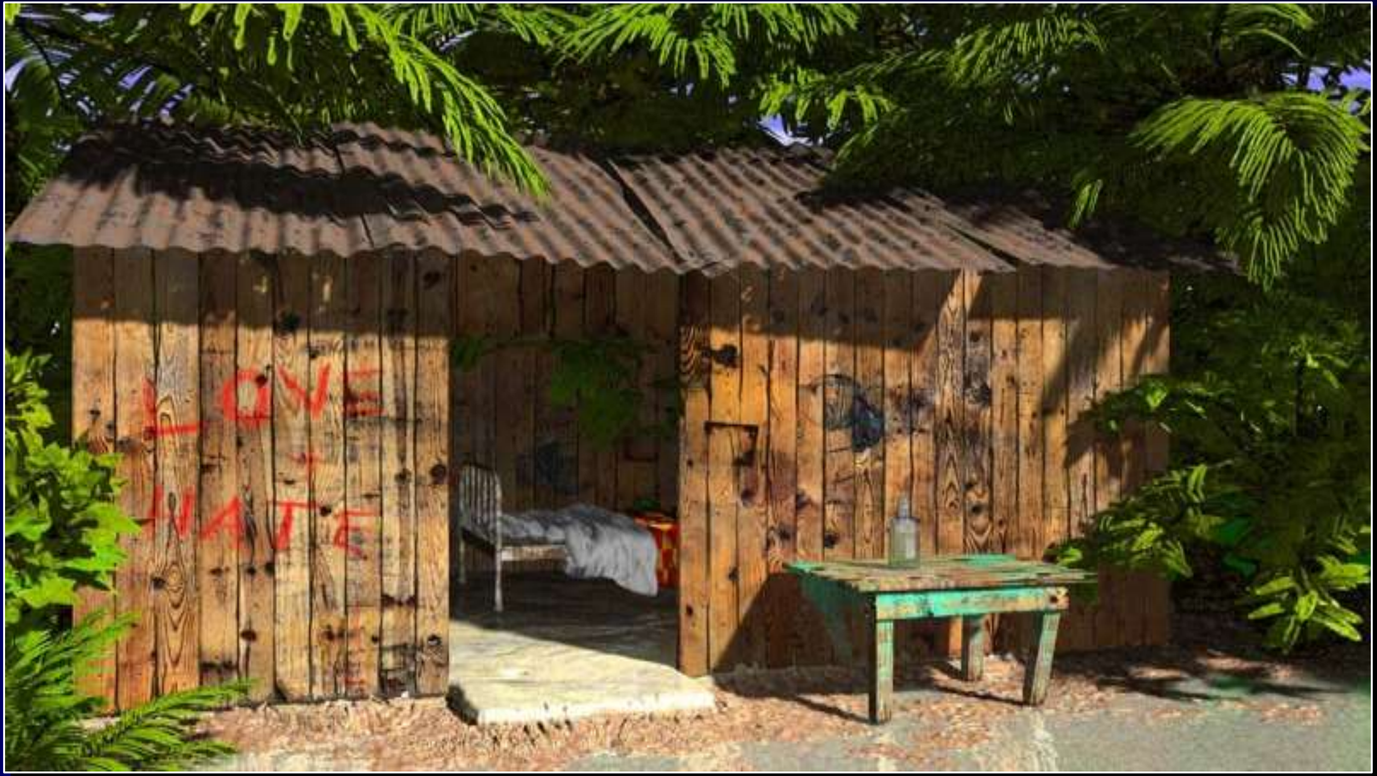
Motion blur

Camera models

Etc.



A Final Result



What is Academic about all this?

- *The full collection of techniques and “hacks” known to the industry? No*
- *The history of the techniques? No*
- *The skill of getting code working? No*

- *Well-understood math concepts that separate graphics from other programming? That’s better*
- *Tying into graphics research sometimes*