A breadth-first approach to Computer Graphics

Serena Morigi

Computer Graphics =

pretty pictures of possibly moving, possibly interactive, solid or fluid, artificial or living things for people to see on displays

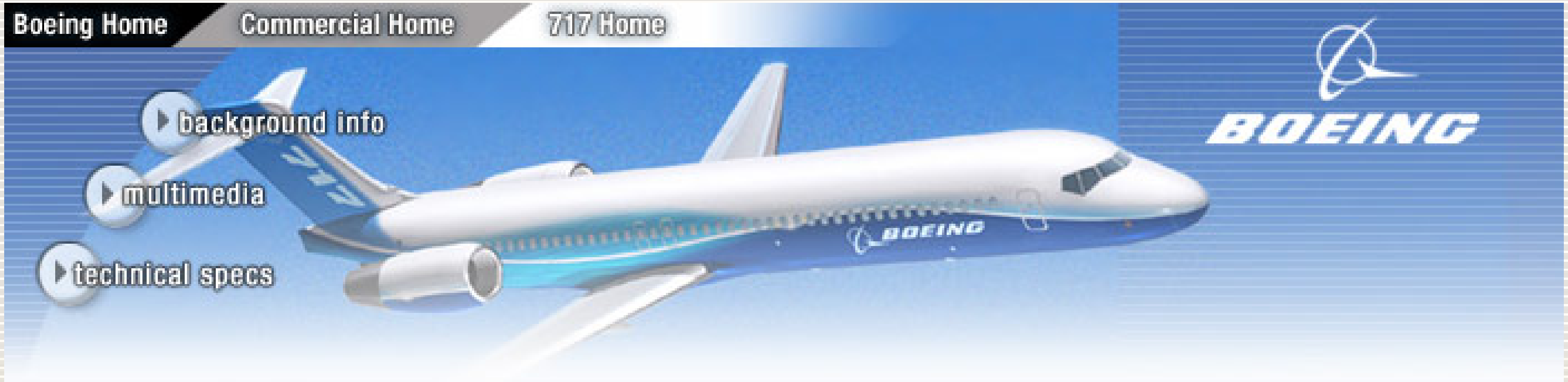
Who needs Computer Graphics?

- Computer-Aided Design/Manufacturing
- Medical Imaging
- Simulation
- Architecture
- Electronic publishing
- Computer Animation / Film Production
- Art
- Games
- ...

Industrial Design

The final product is 3D

- Aeroplane
- Cars
- Boat
- Toys
- Tools
- ….

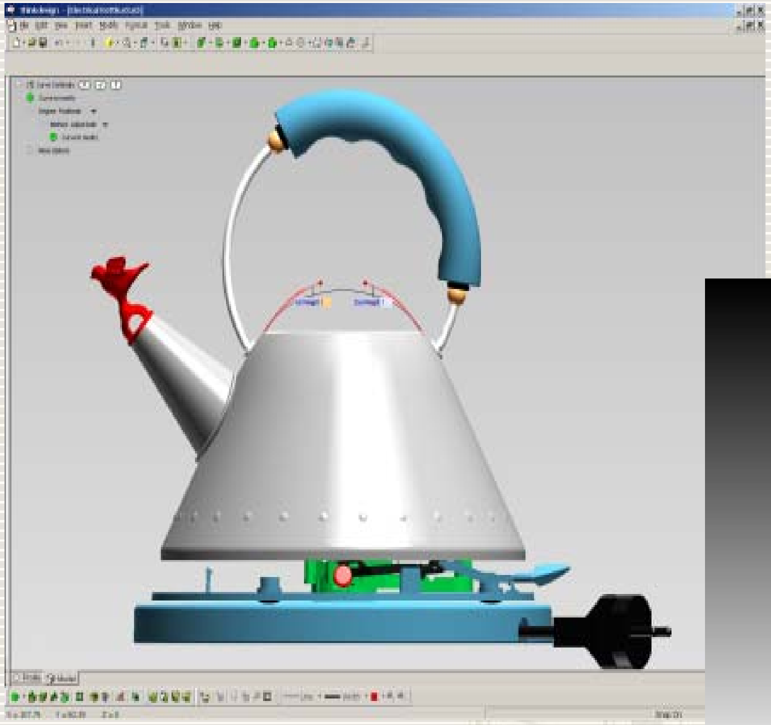


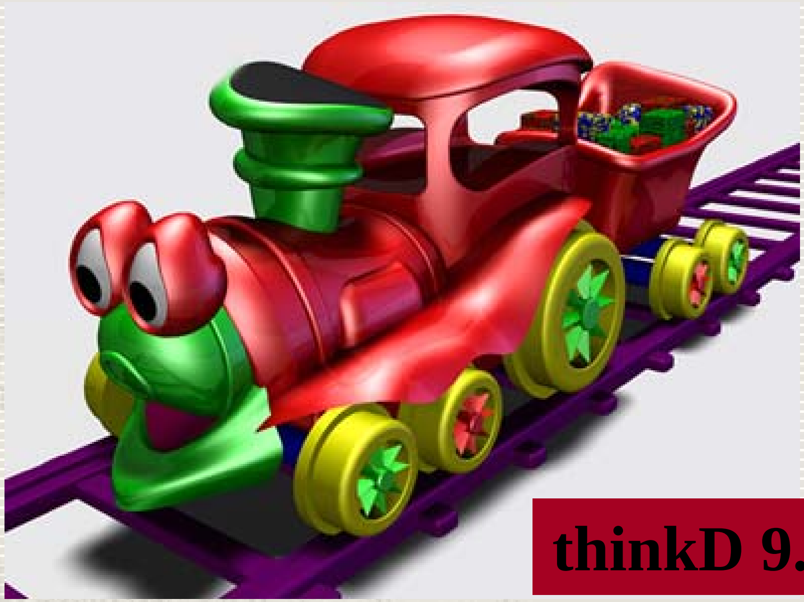
CAD Systems

William Fetter coined term "**computer graphics**" in 1960 to describe new design methods he was pursuing at Boeing

think3®

Higher Quality Models are achieved through thinkD’s extensive modeling, editing, and continuity analysis toolsets. The dynamic, real-time capabilities provide immediate feedback for making critical aesthetic, functional and engineering decisions





thinkD 9.0

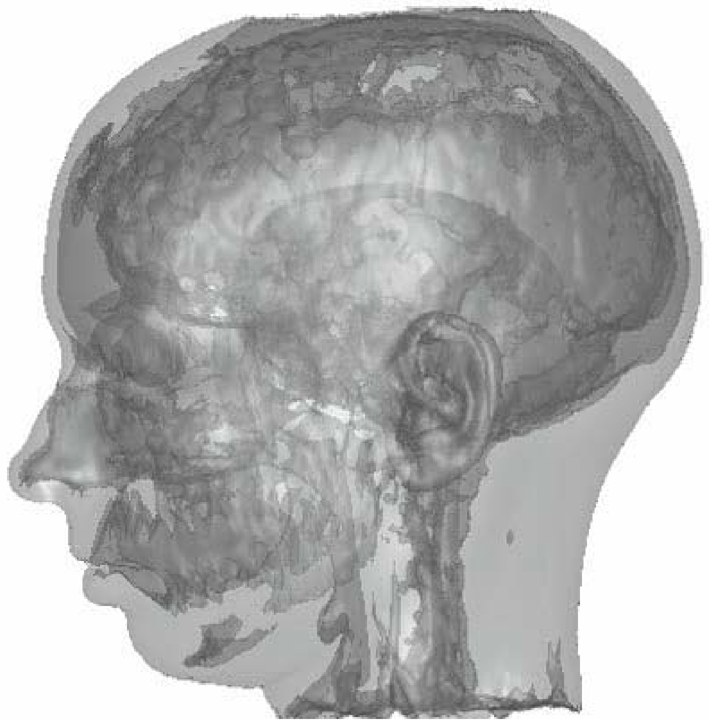
Medical Imaging

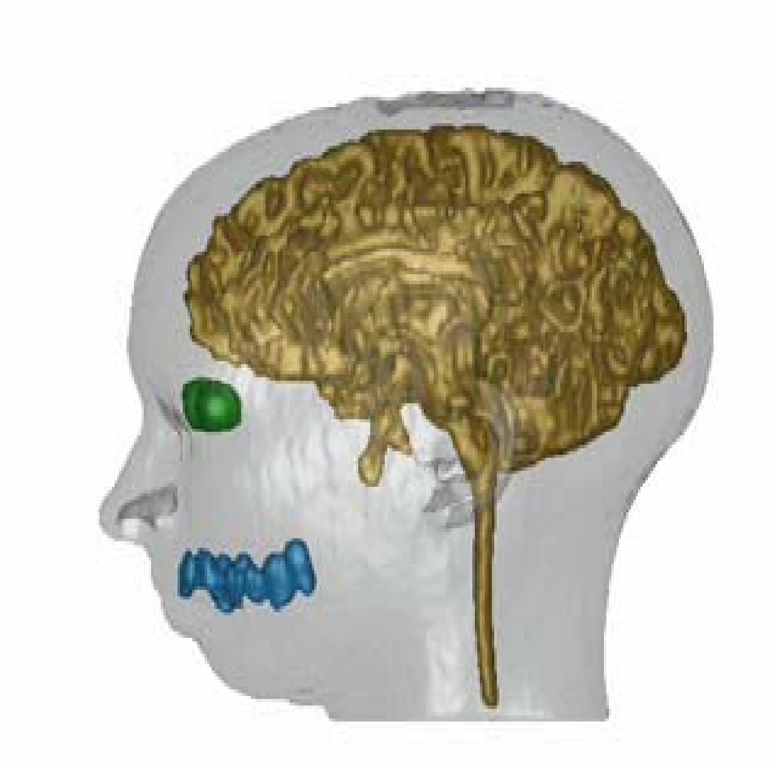
How to represent volume data?

-volume rendering

-isosurfacing

Magnetic resonance MRI (3D volume 257^3)

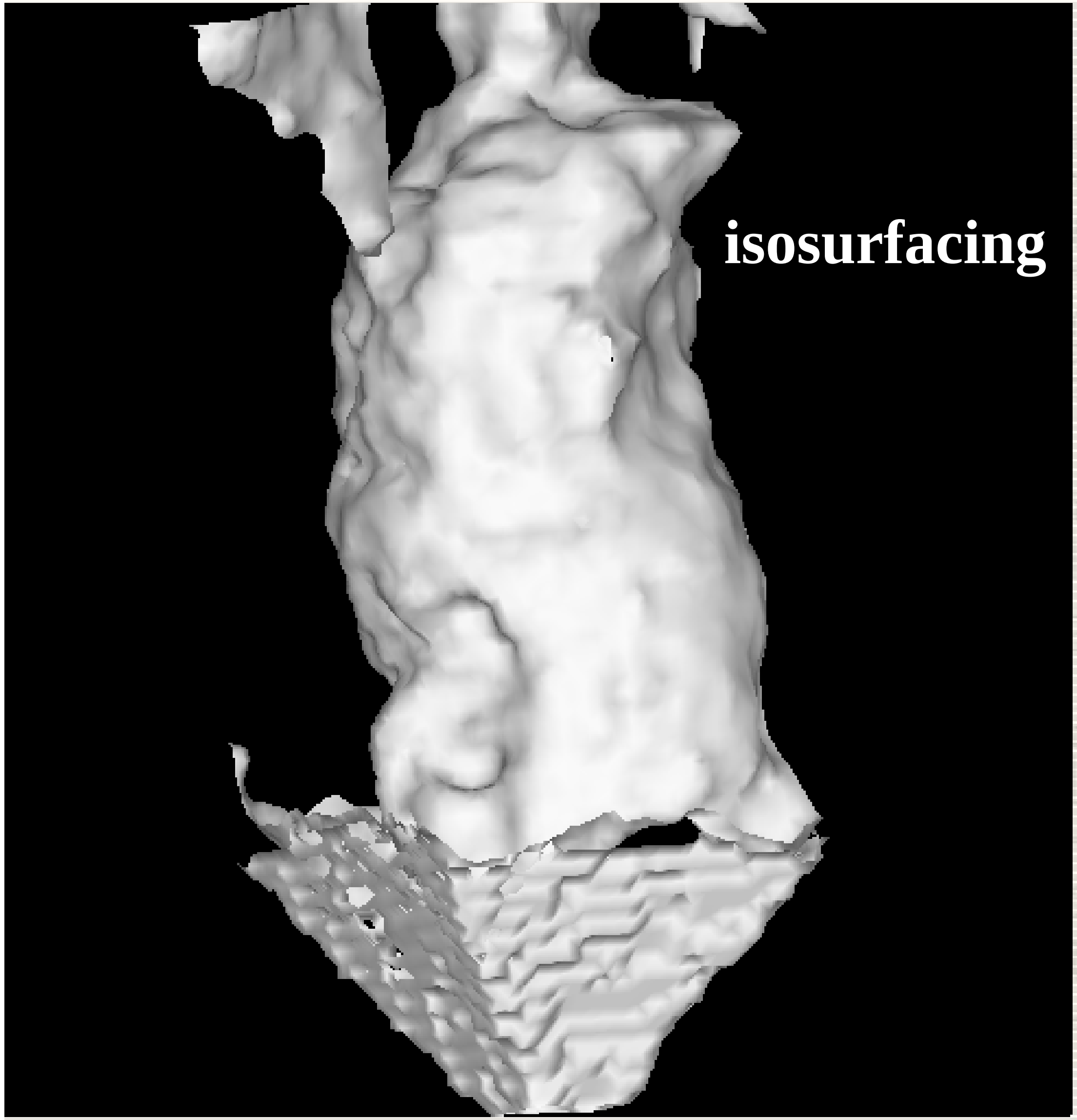




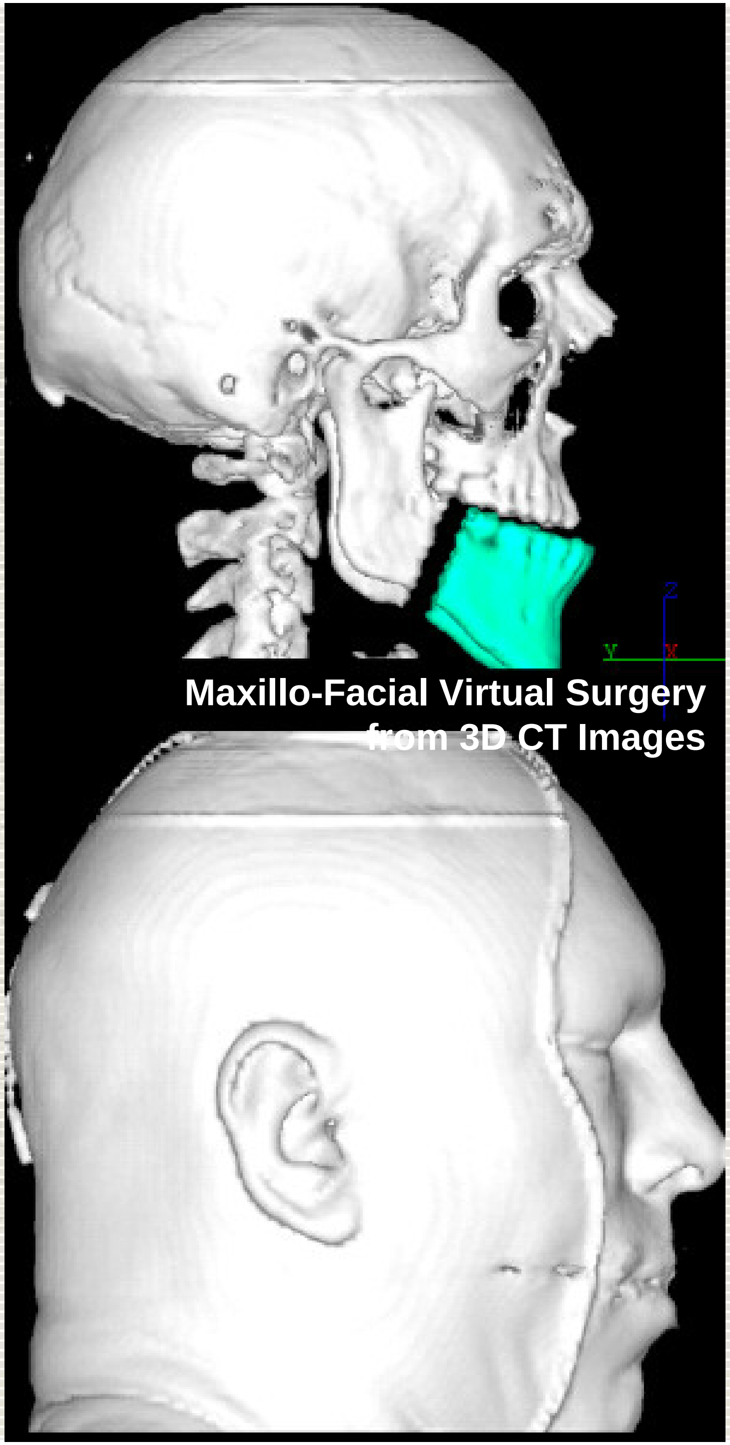
segmentation

Collaboration with Prof. Martin Rumpf – Bonn Germany

Medical Imaging



isosurfacing



Maxillo-Facial Virtual Surgery
from 3D CT Images

Collaboration Prof. Sgallari –ESAOTE

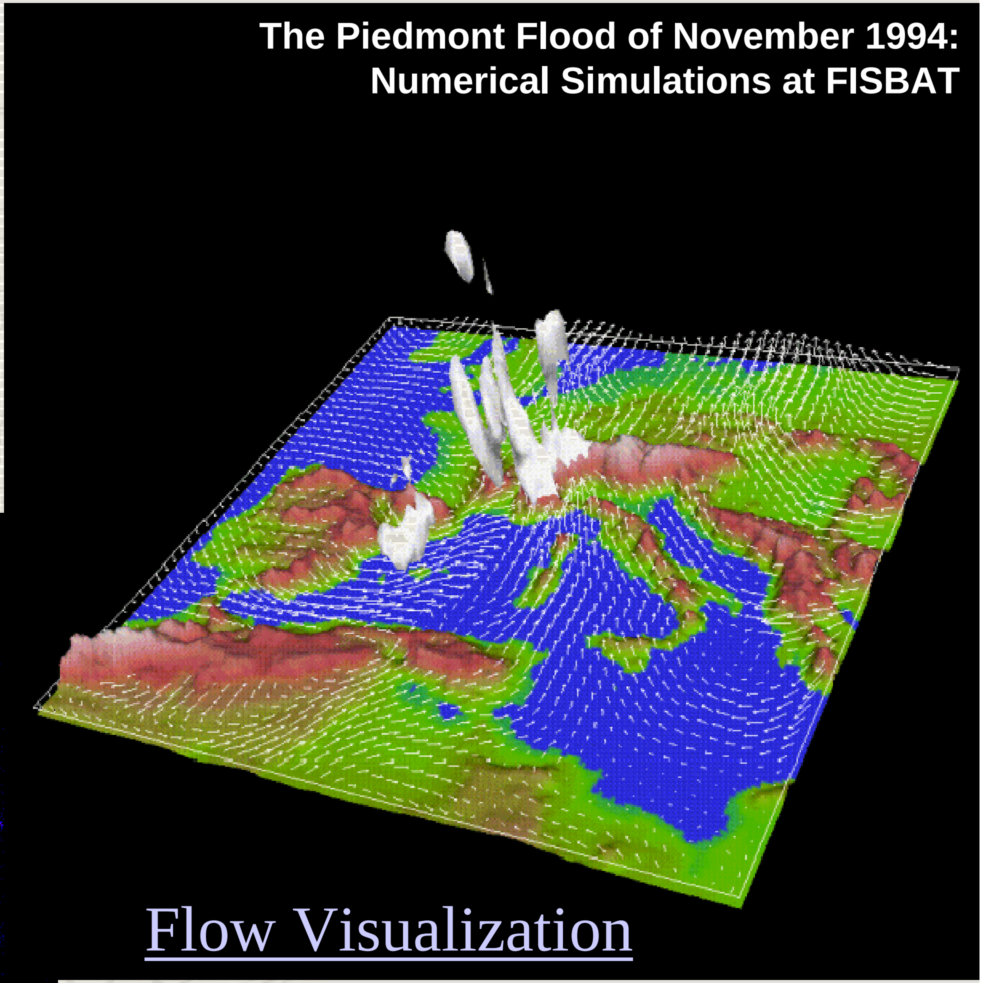
Scientific Visualization

“The merging of data with the display of geometric objects through computer graphics”

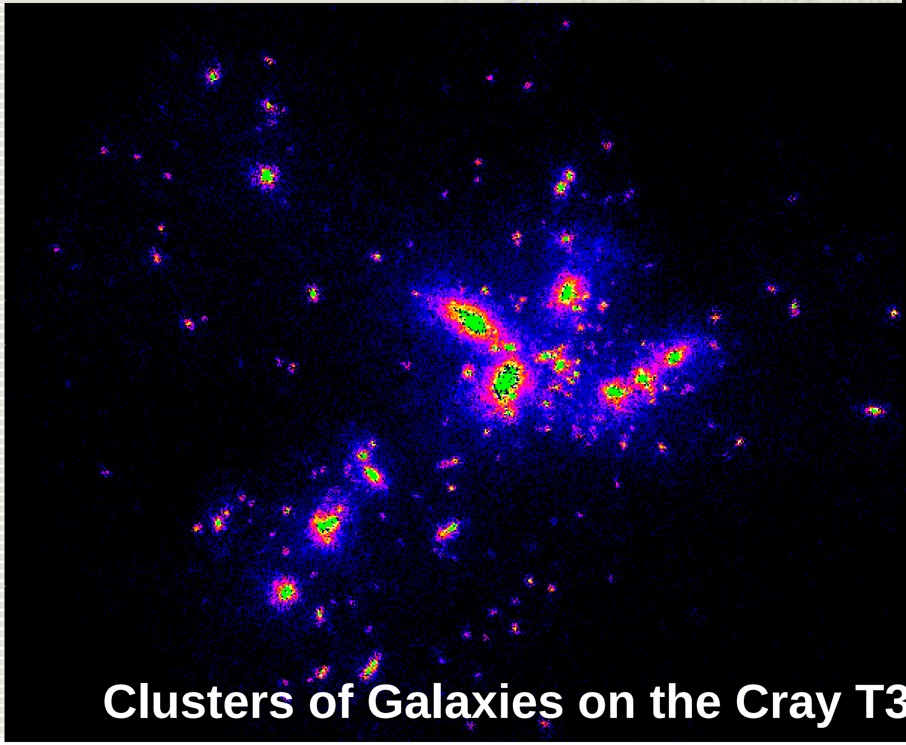
DATA + GEOMETRY:

- Understanding of data
- Insight into information
- Presentation and sharing of insights.

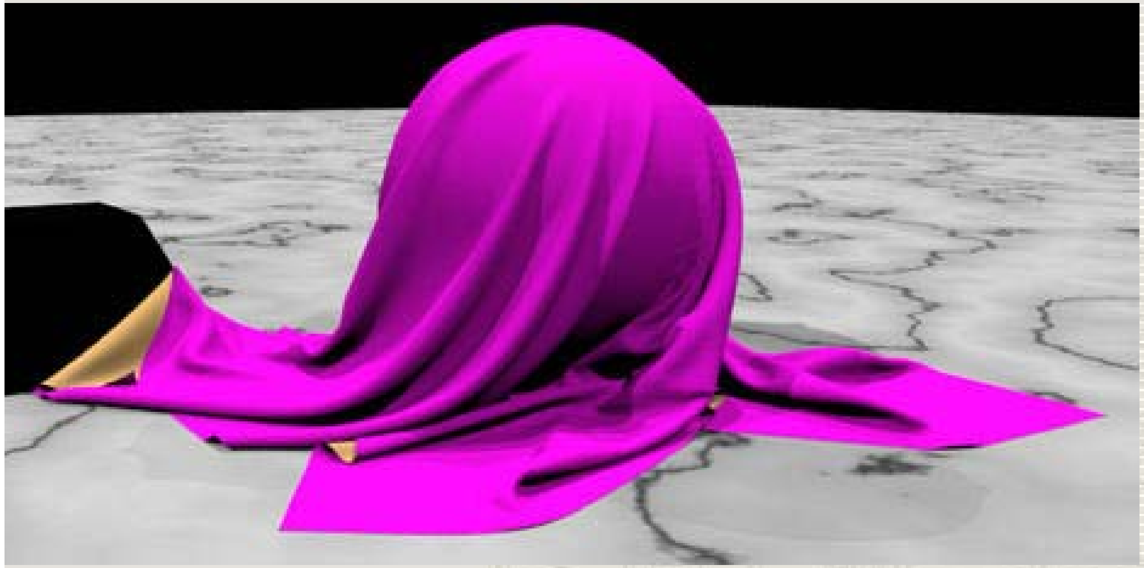
Scientific Visualization

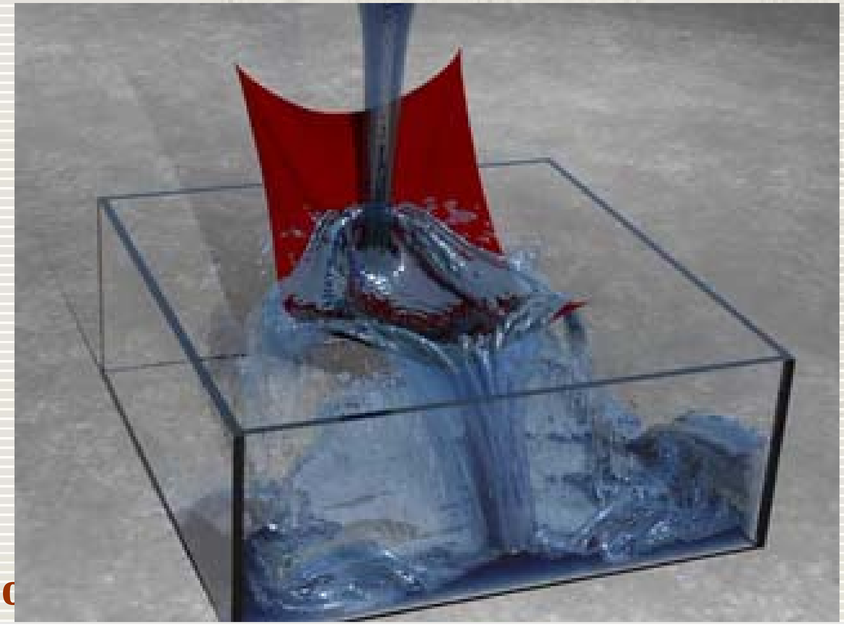

The Piedmont Flood of November 1994:
Numerical Simulations at FISBAT

Flow Visualization


Clusters of Galaxies on the Cray T3D

Simulation



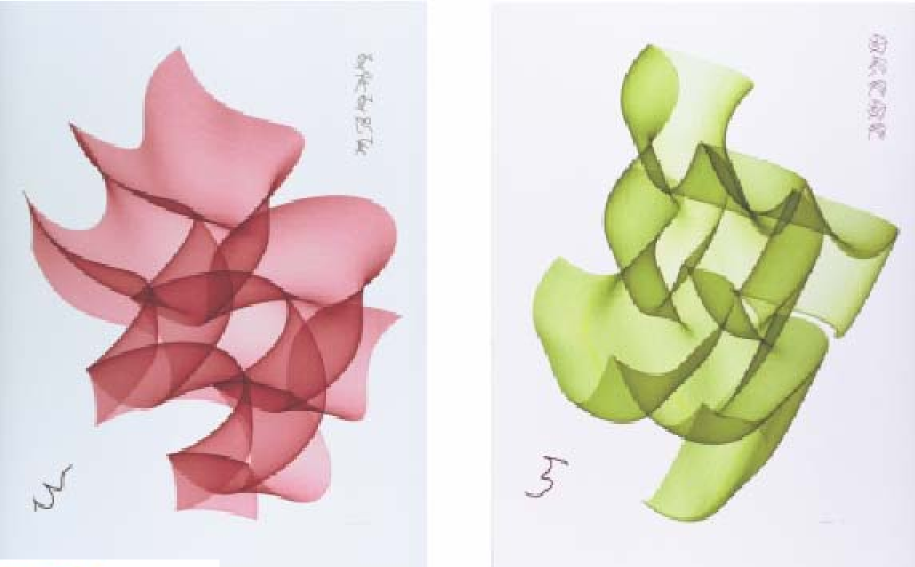


Art

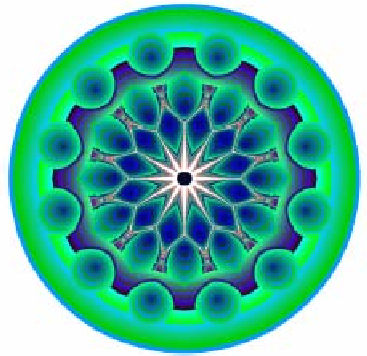
ART GALLERY 2003, SIGGRAPH San Diego

Cat’s eye

Circle face



Cyberflower duet, red & green



Polynomiography: visualizzazione di approssimazioni di zeri di polinomi.

Fig: grado 36

Waves

Art

Surface fairing

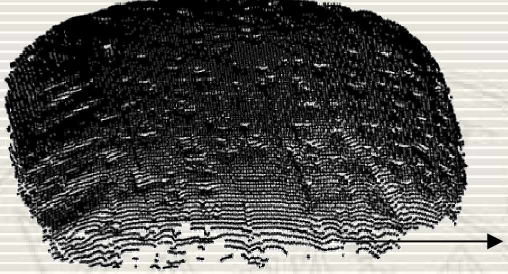
Reverse Engineering and …

…..Rapid Prototyping

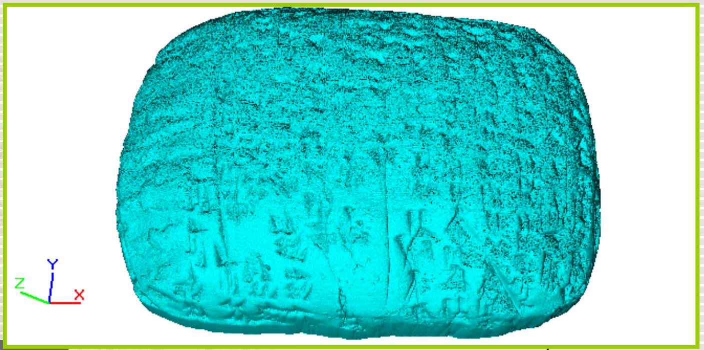
La Tavoletta Cuneiforme

(tesi di Serena Trerè - ENEA -)

Object

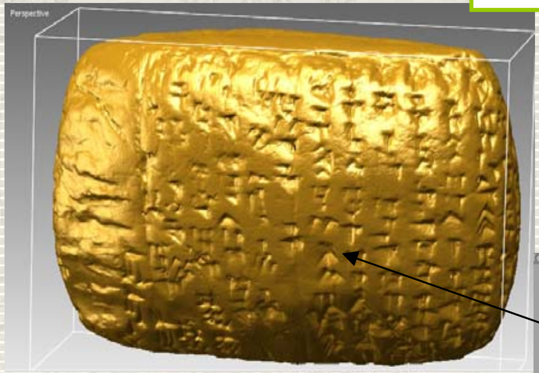


Cloud of points

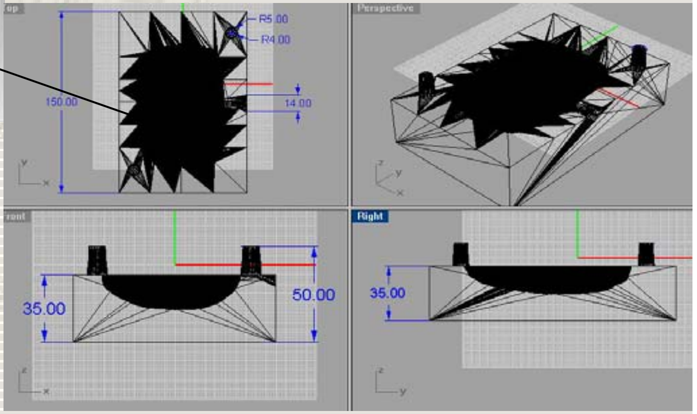


Virtual Model

3D scanner Picza



Rendering



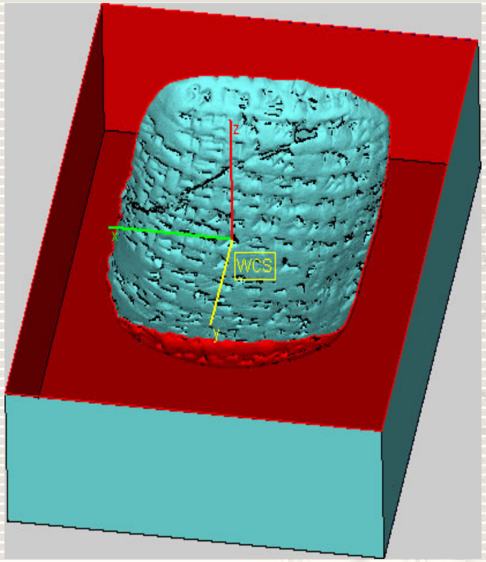
CAD

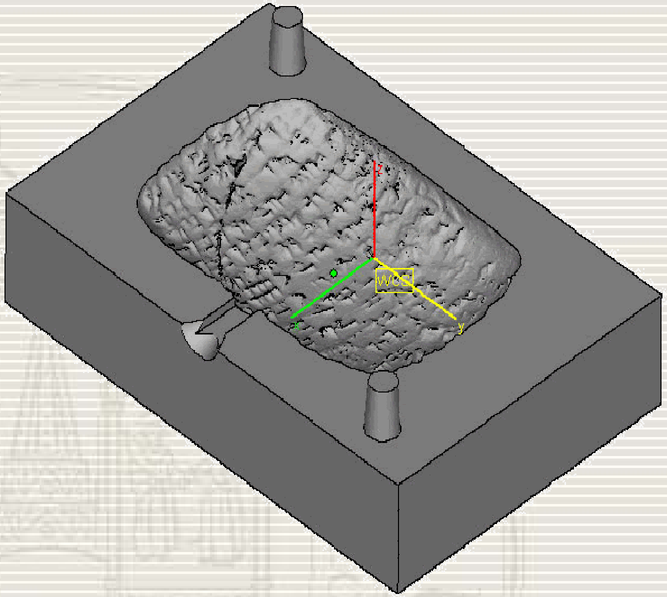
Prototype object

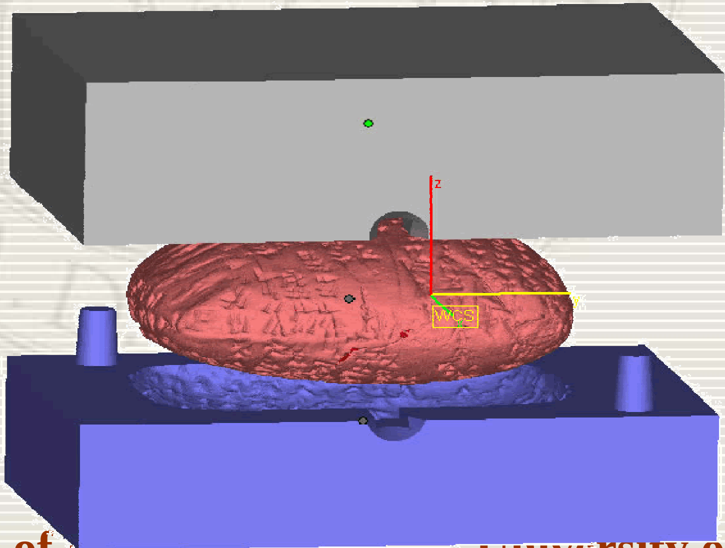
“Thermojet” -ENEA-

University of Bologna

CAD PostProcessing





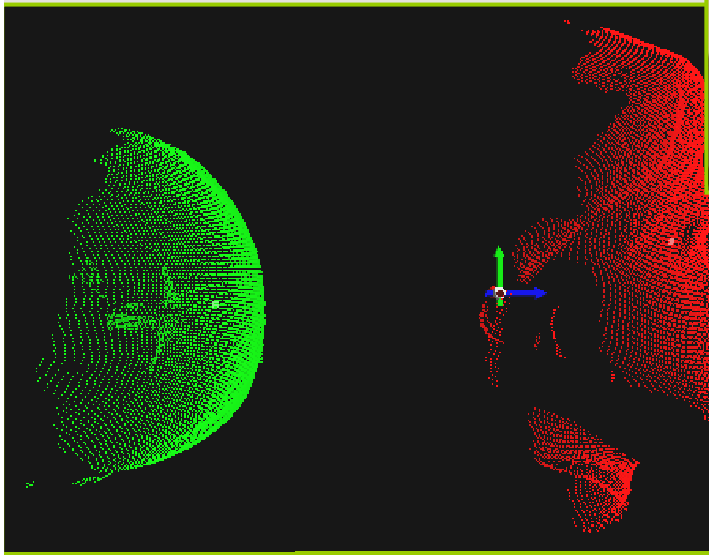


Minerva Project

(tesi di Filippo Veneri)

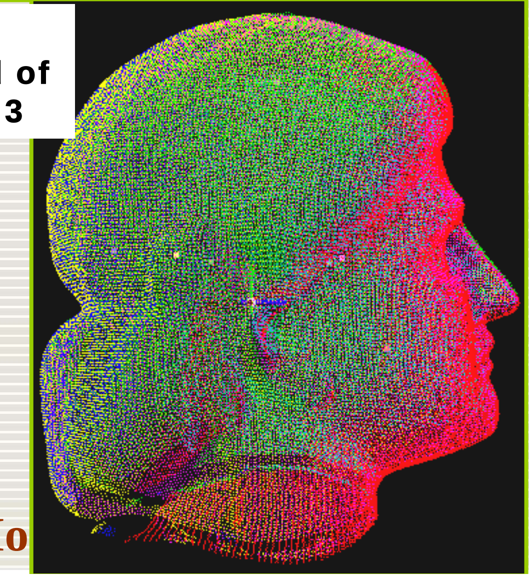
Object

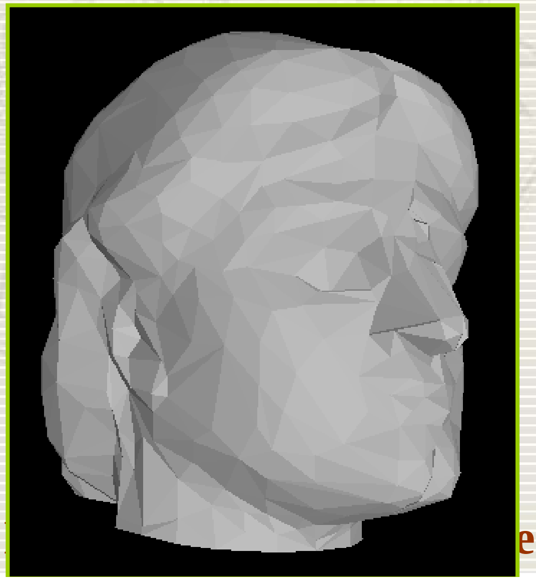
Scanner laser 3D: Minolta VIVID 900

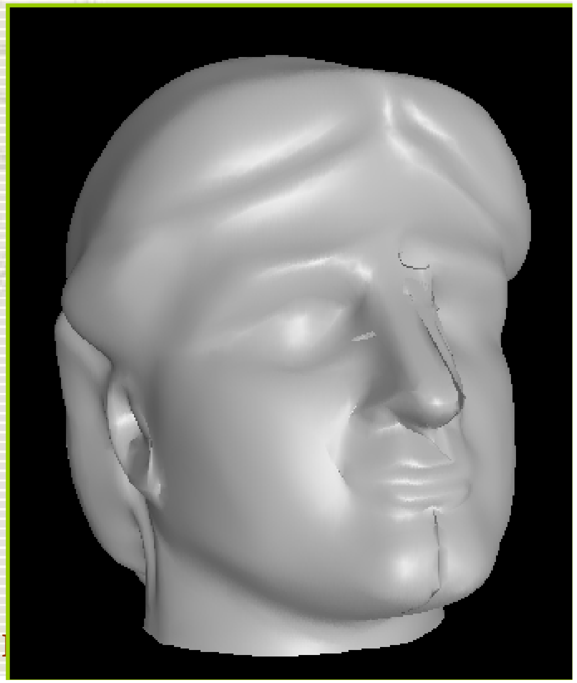


Surface Reconstruction

8 RANGE IMAGE, cloud of points 98503







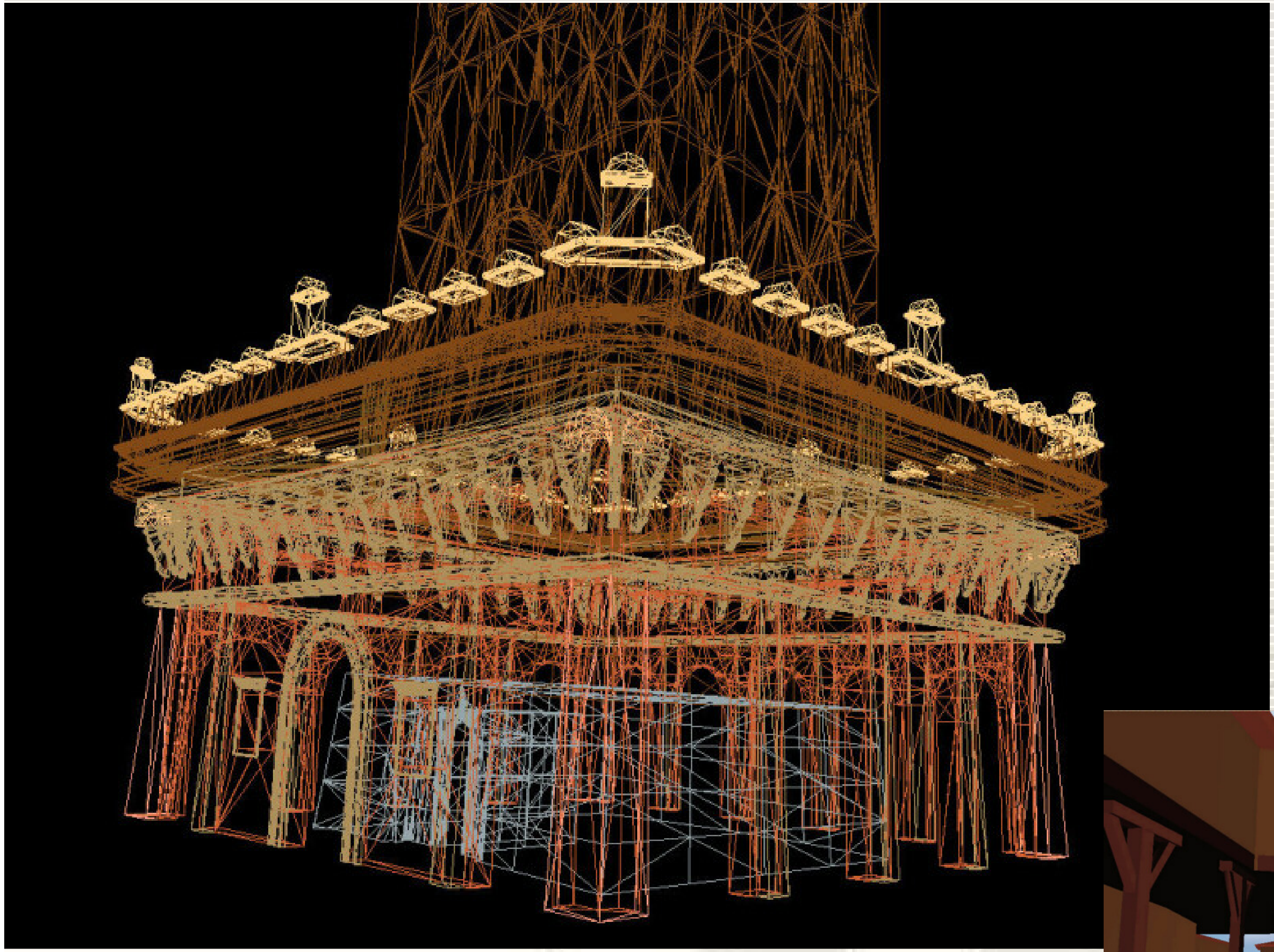
Architectural Walkthroughs

- Virtual paths

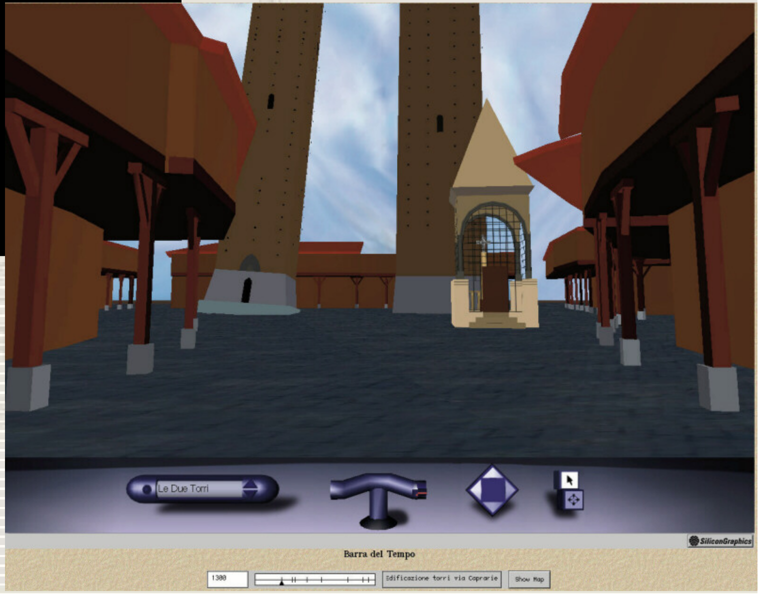
Cineca, Bologna

L'idea alla base del progetto NUME (NUovo Museo Elettronico della città di Bologna) è la realizzazione di un ambiente multimediale che consenta di ripercorrere a ritroso nel tempo la situazione urbanistica della città, partendo da quella attuale sino a quella ricostruita attraverso le fonti storiche. Questi documenti, integrati con le nuove tecnologie della realtà virtuale e resi sotto forma di continuum, consentono di ricreare l'evoluzione storica della città.

Scopo di NUME è quello di dare corpo al concetto di Città Digitale attraverso la realizzazione di un modello tridimensionale e storico del centro di Bologna.



The Torre Asinelli's wireframe model



Electronic publishing

ADIDAS “mechanical legs”

GATORADE: Gatorade “23 vs 39“ presented interesting challenges. The spot involves present-day Michael Jordan playing one-on-one against 1986 Michael Jordan. The challenge was to keep the performances fresh, creating the illusion that two MJs are playing one-on-one.

E-business, E-commerce

Entertainment

- Computer Games
- Film Production
- Special Effects

PIXAR Animation Studios

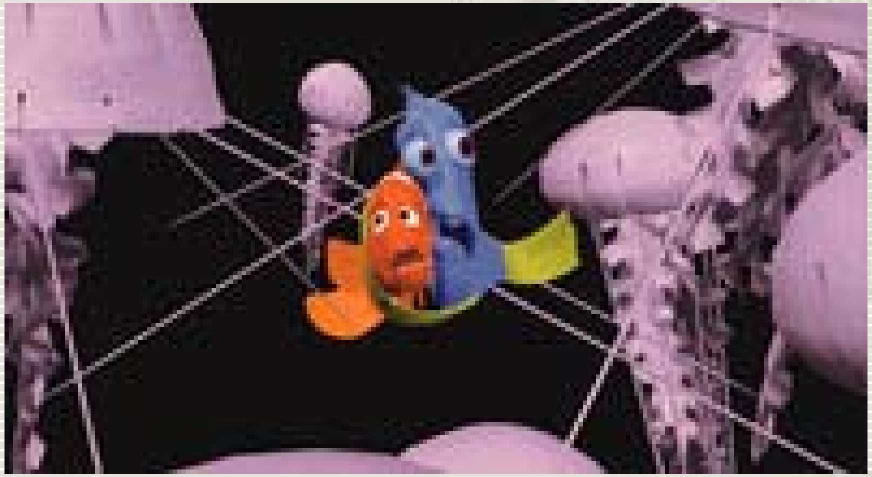
Luxo Junior (1986)
Red's Dream (1987)
Toy Story (1995)
A Bug's Life (1998)
Toy Story 2 (1999)
Monster,Inc. (2001)
Finding Nemo (2003)
The Incredibles (2004)
Cars (2006)

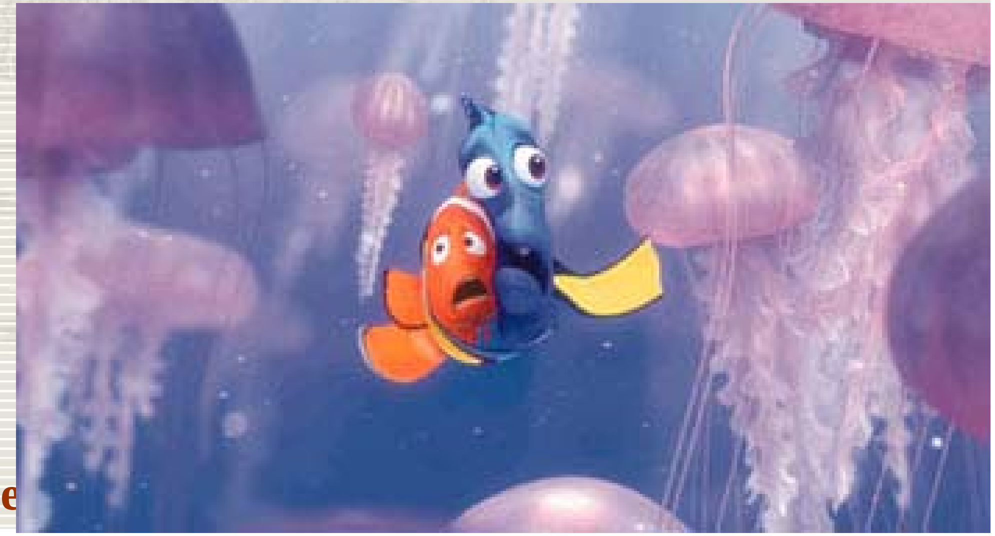
DREAMWORKS

Shrek/Shrek 2 (2003/2005)
Madagascar (2005)

PIXAR Animation Studios

©Disney/Pixar





Film , special effects:

Industrial Light & Magic (ILM)

creature skin and muscles, skin rendering, motion capture, rigid and deformable dynamics, image-based modeling, digital doubles, fluid and smoke simulation, 3D compositing, cloth simulation, and new animation techniques.

Terminator 3

The lord of the rings

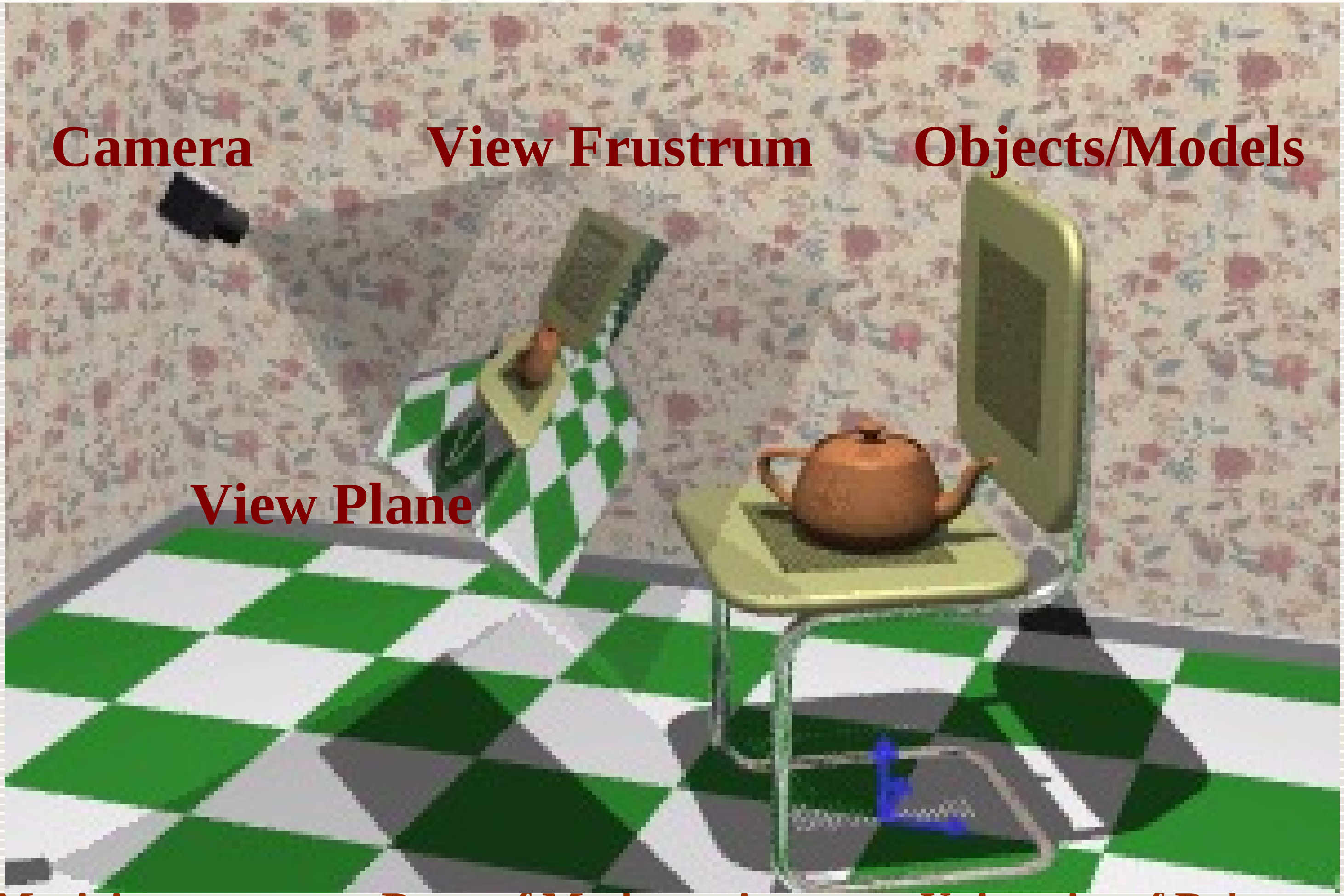
(ESC Entertainment)

The matrix reloaded

Harry Potter

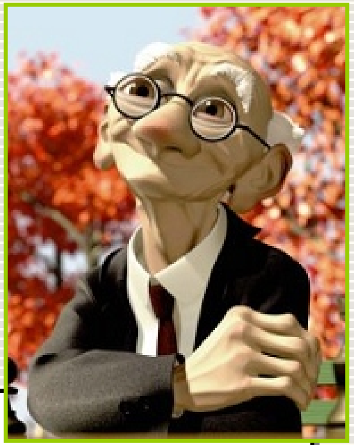
PlayStation.2
Red Dead Revolver
Midtown Madness
Chicago Edition
Checkpoint
Microsoft
Midtown Madness 2
Checkpoint
Microsoft
SEGA
SEGA SATURN
Mr Bones
KA
SEGA
GAME GEAR
ECCO
THE TIDES OF TIME
GA
PlayStation.2
鬼
XBOX
TRANSWORLD
SURF
T
ATARI
PlayStation.2
湾岸
MIDNIGHT CLUB
STREET RACING
T
R
PlayStation.2
湾岸
MIDNIGHT CLUB II
RP
R
NINTENDO 64
RESIDENT EVIL
2
CAPCOM
M
KEN GRIFFEY JR.'S
Slugfest
E
XBOX
TEST DRIVE
OFF-ROAD
WIDE OPEN
HUMMER
E
PlayStation.2
SMUGGLER'S RUN
T
R
PlayStation.2
CRAZY BUMP'S
クレイジーバンプ
かっとびカーバトル!
SYSCOM
PlayStation.2
SMUGGLER'S RUN 2
HOSTILE TERRITORY
T
R
MAJOR LEAGUE
Baseball
FEATURING
KEN GRIFFEY JR
E

CG basic:
Virtual objects, scene
Viewer (camera)



The graphics process

Pixar, ‘Geri’s game’



3D MODELLING

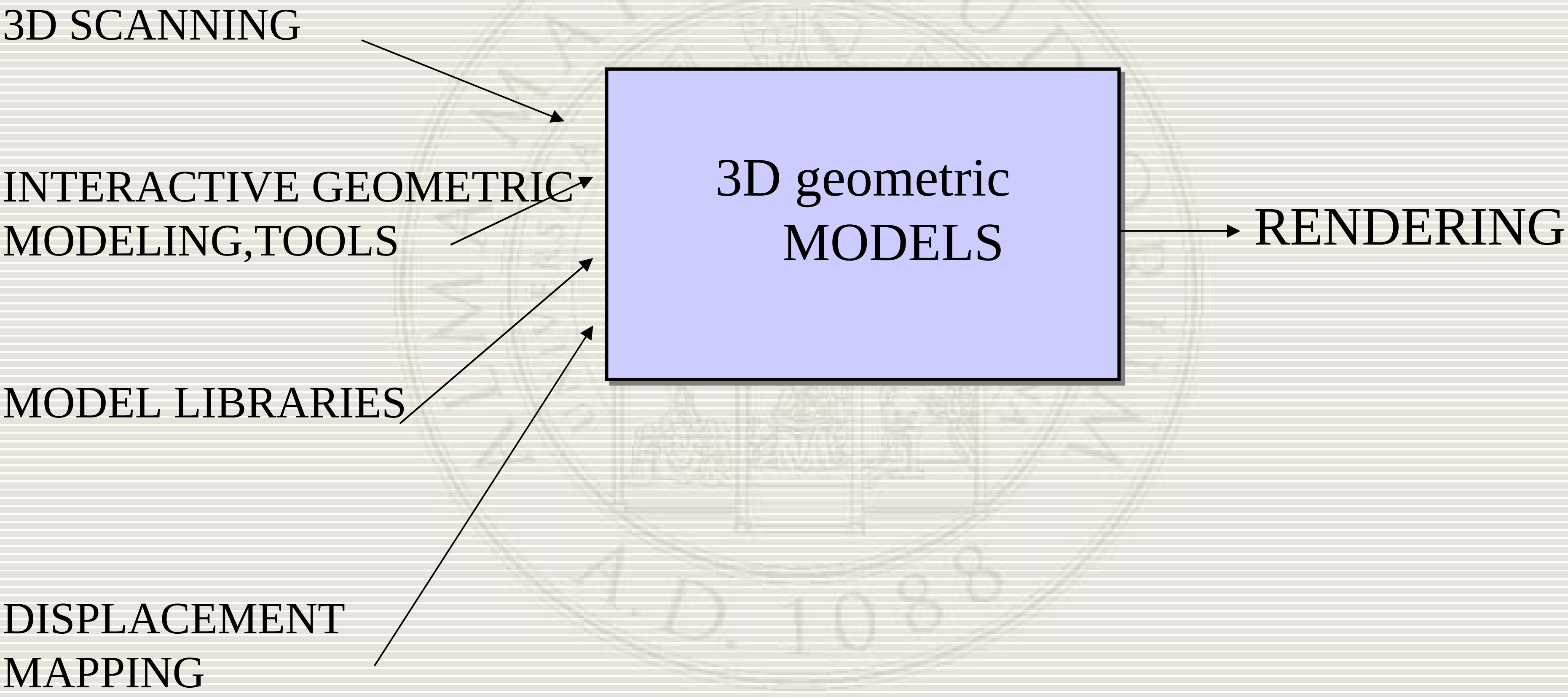
RENDERING

IMAGE STORAGE
DISPLAY

ANIMAZIONE

TEXTURING

The graphics process: Geometric Modeling



3D SCANNING

INTERACTIVE GEOMETRIC MODELING,TOOLS

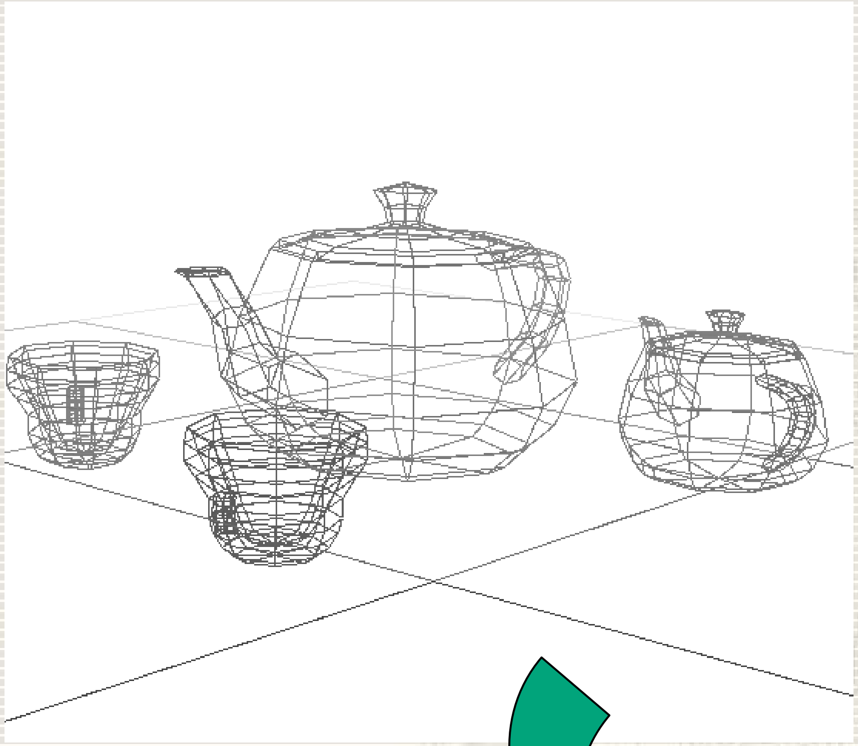
MODEL LIBRARIES

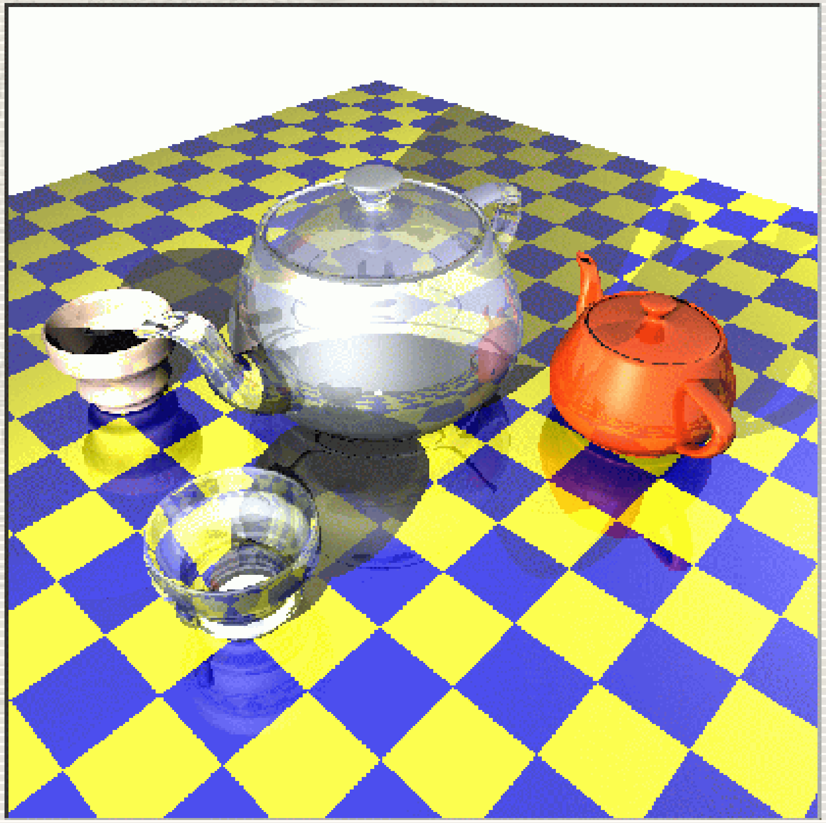
DISPLACEMENT MAPPING

3D geometric MODELS

RENDERING

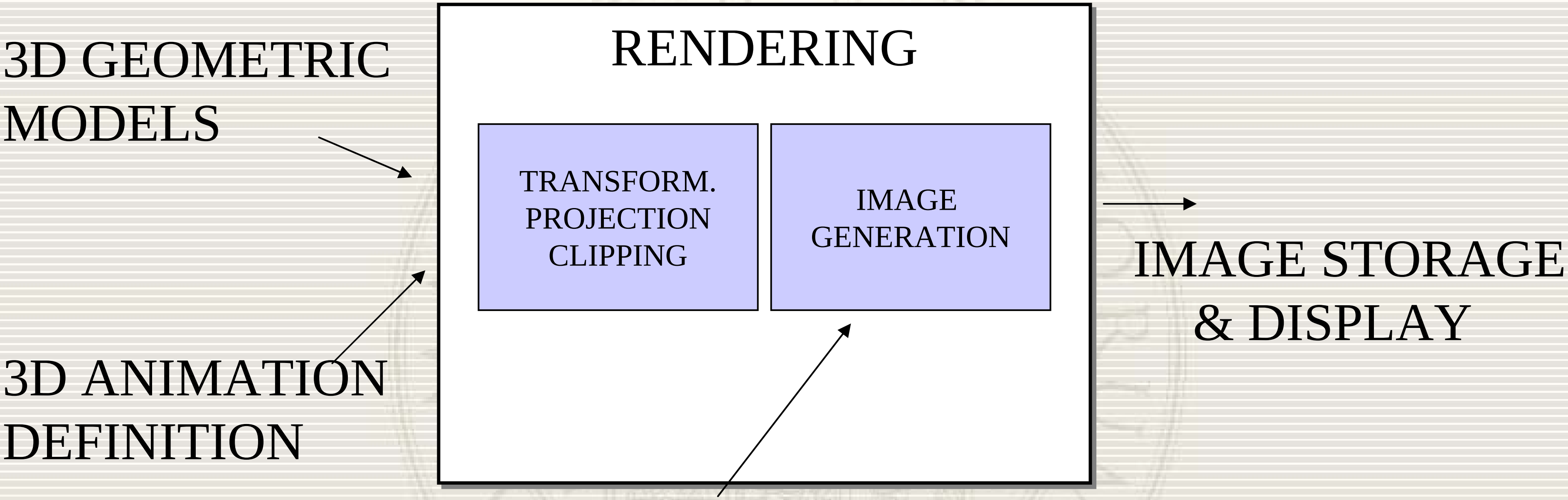
Rendering





Produce bidimensional images from a 3D scene and a camera

The graphics process: RENDERING



3D GEOMETRIC MODELS

3D ANIMATION DEFINITION

RENDERING

TRANSFORM.
PROJECTION
CLIPPING

IMAGE
GENERATION

IMAGE STORAGE
& DISPLAY

TEXTURING
- image painted,
- Scanned images,
- computed images

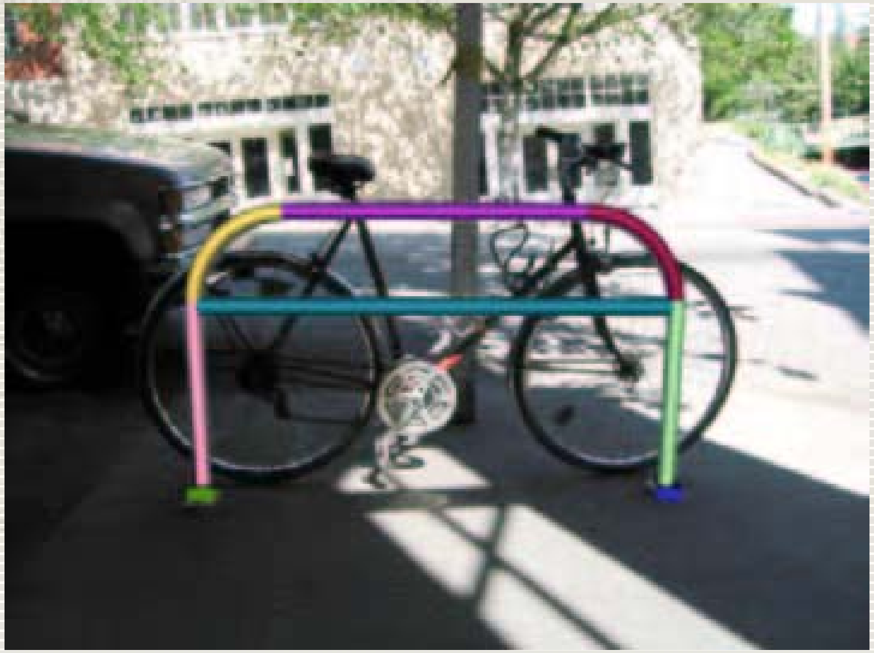
Two steps process:
Visibility +Shading

3D MODELLING

Design the shape of 3D objects

Seeing in 3D

- The world in basic shapes
- Simple but not too simple



NO
STOPS
TOW AWAY
ZONE

Detail for Image Synthesis

- Real shapes are complex
- More detail= more realism

  takes longer to model, longer to render, and occupies more disk space

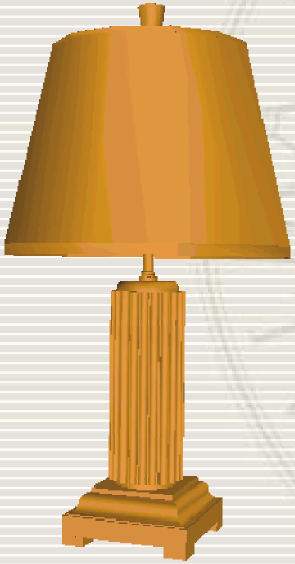
Different detail when required

- Procedural modeling

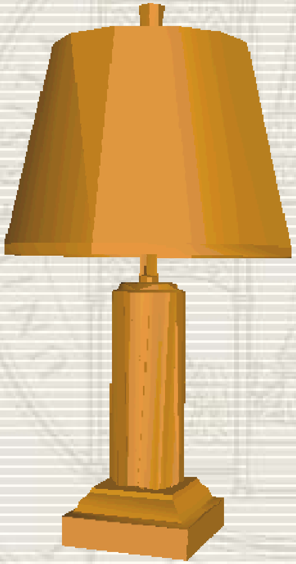
models from **coarse to fine**

Level-of-Detail(LOD):

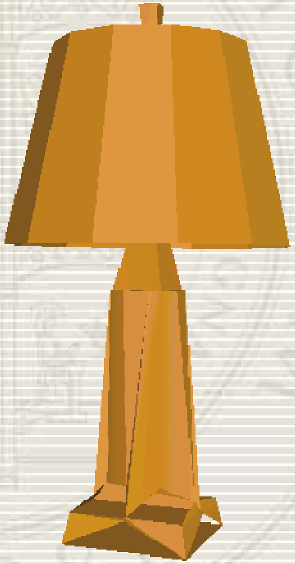
as object gets farther away from viewer, replace it with a lower-polygon version or lower quality texture map. Discontinuous jumps in model detail



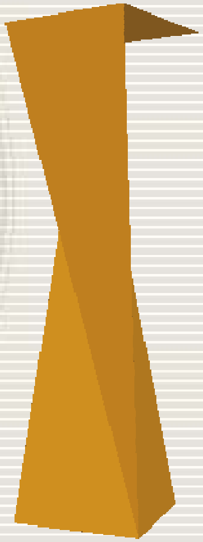
10,108 polys



1,383 polys



474 polys

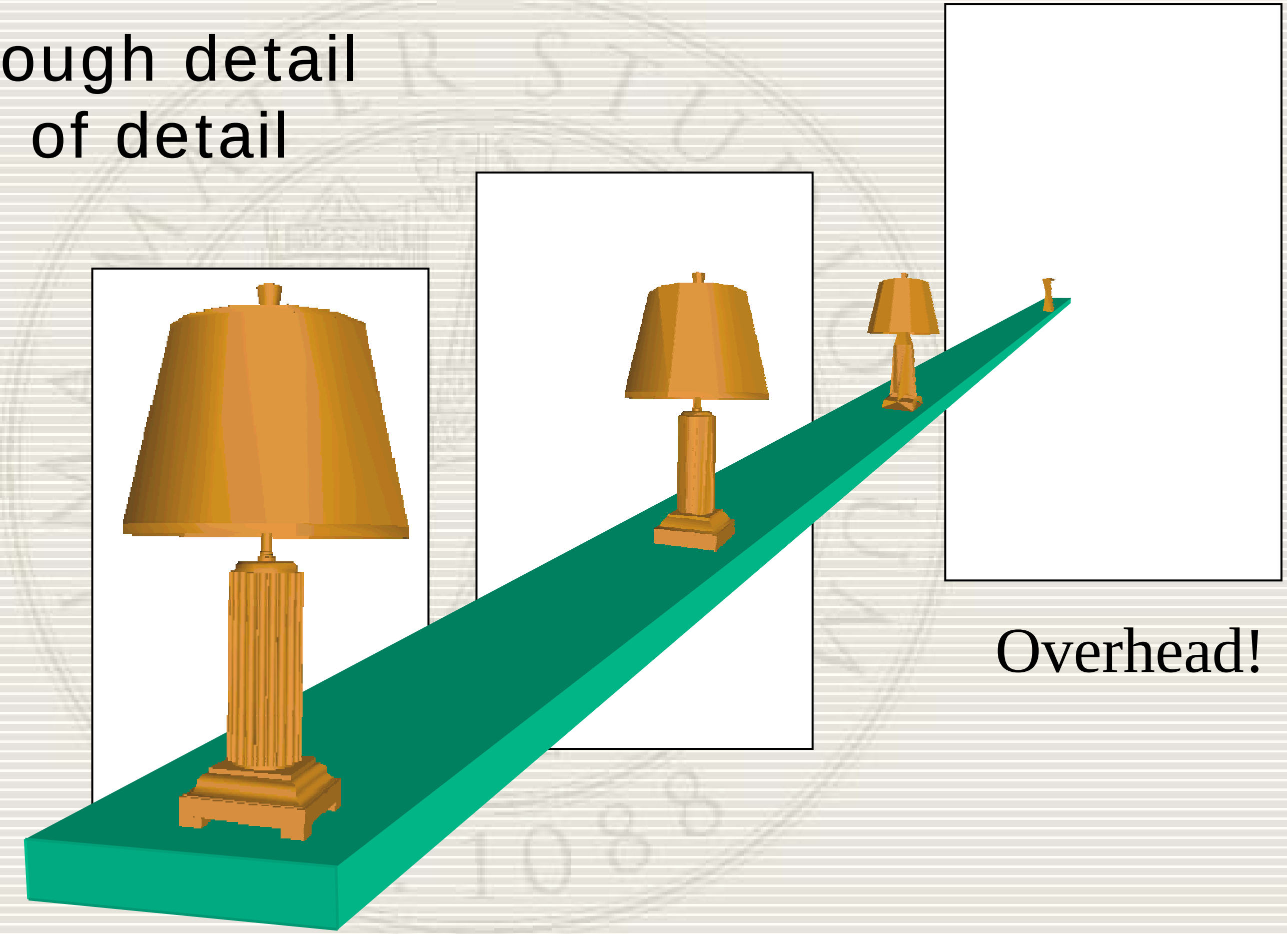


Courtesy IBM

46 polys

LOD:

- Use only enough detail
- Switch level of detail

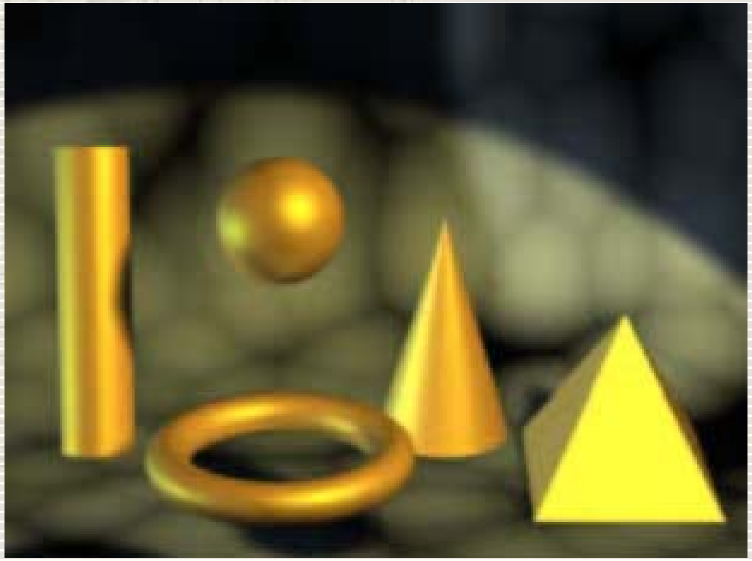


Overhead!

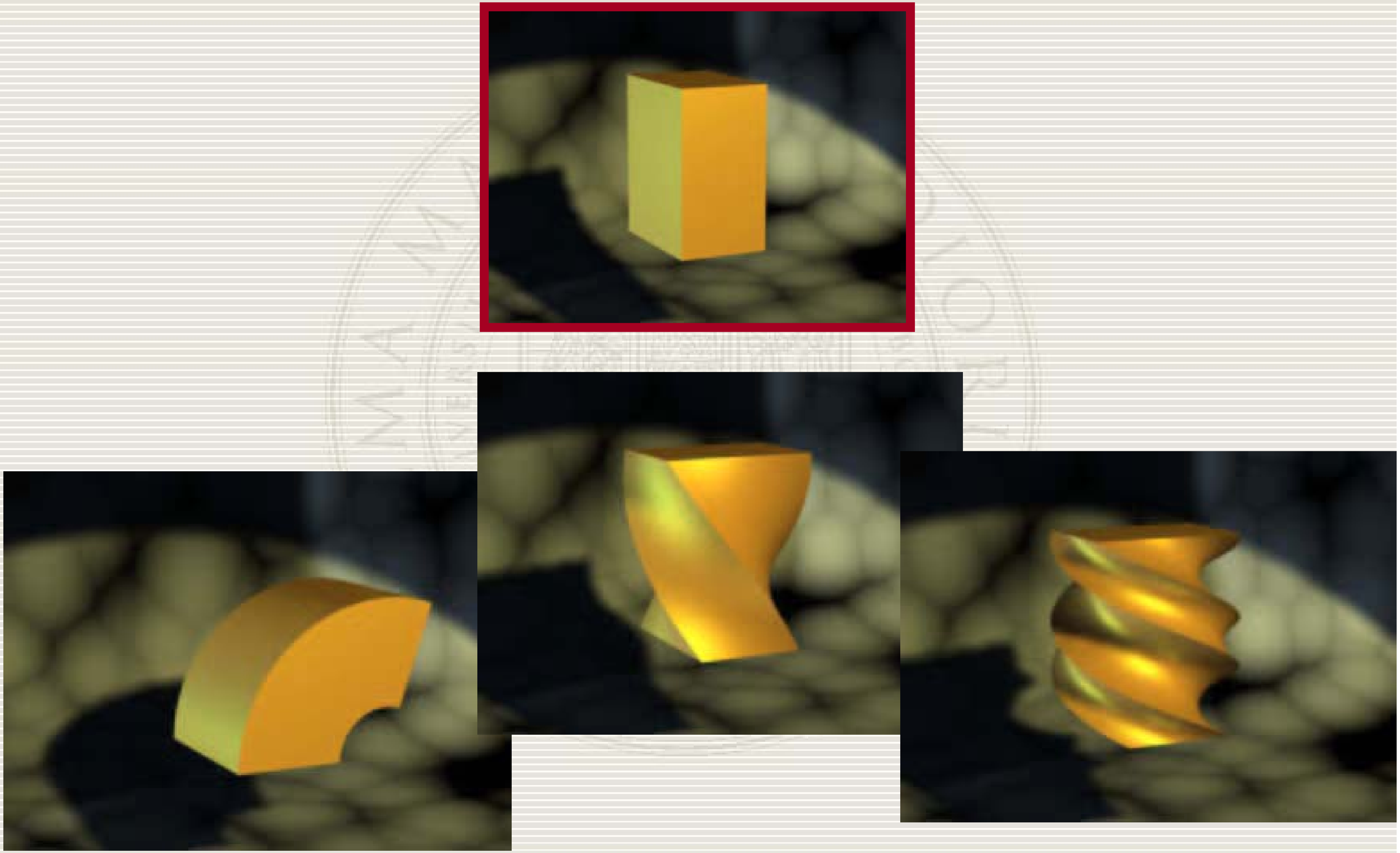
Primitives and instances

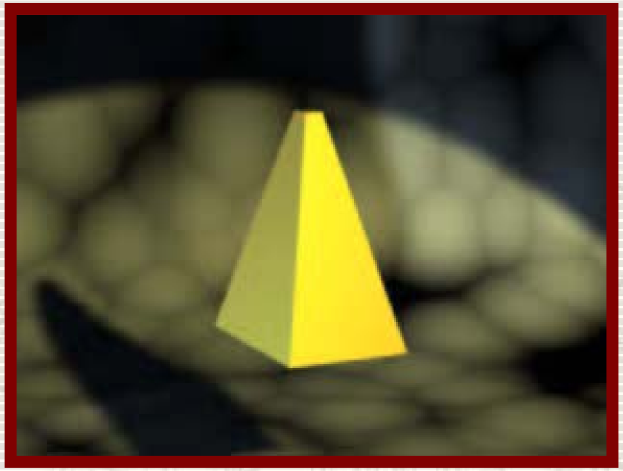
- Divide and Conquer
- Hierarchy of geometrical components
- Shapes are instances of primitives (e.g., spheres, cubes, etc.)

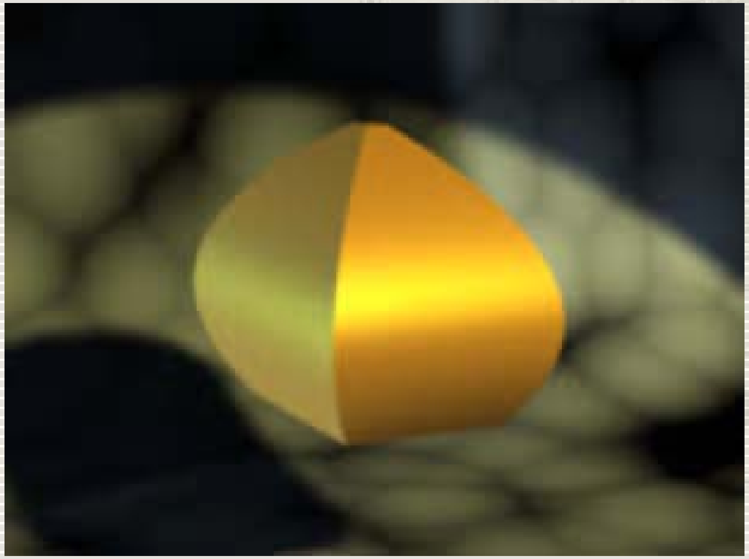
Object to be modeled is (visually) analyzed, and then decomposed into collections of primitive shapes.

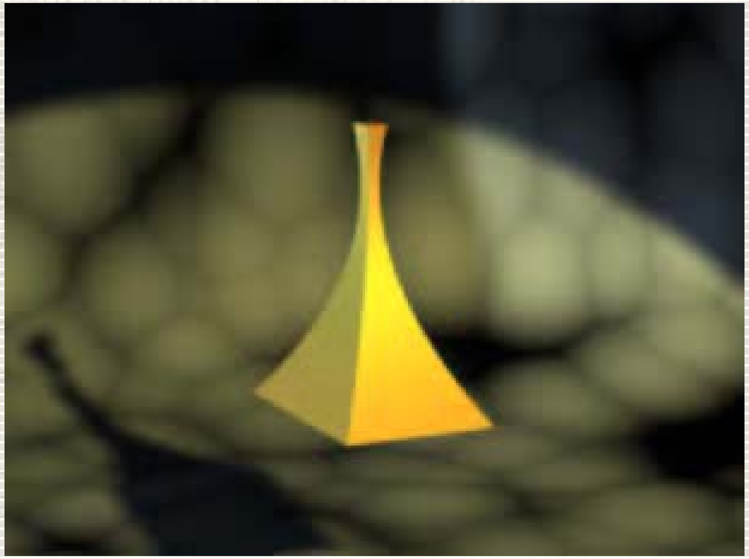


Box Modifiers





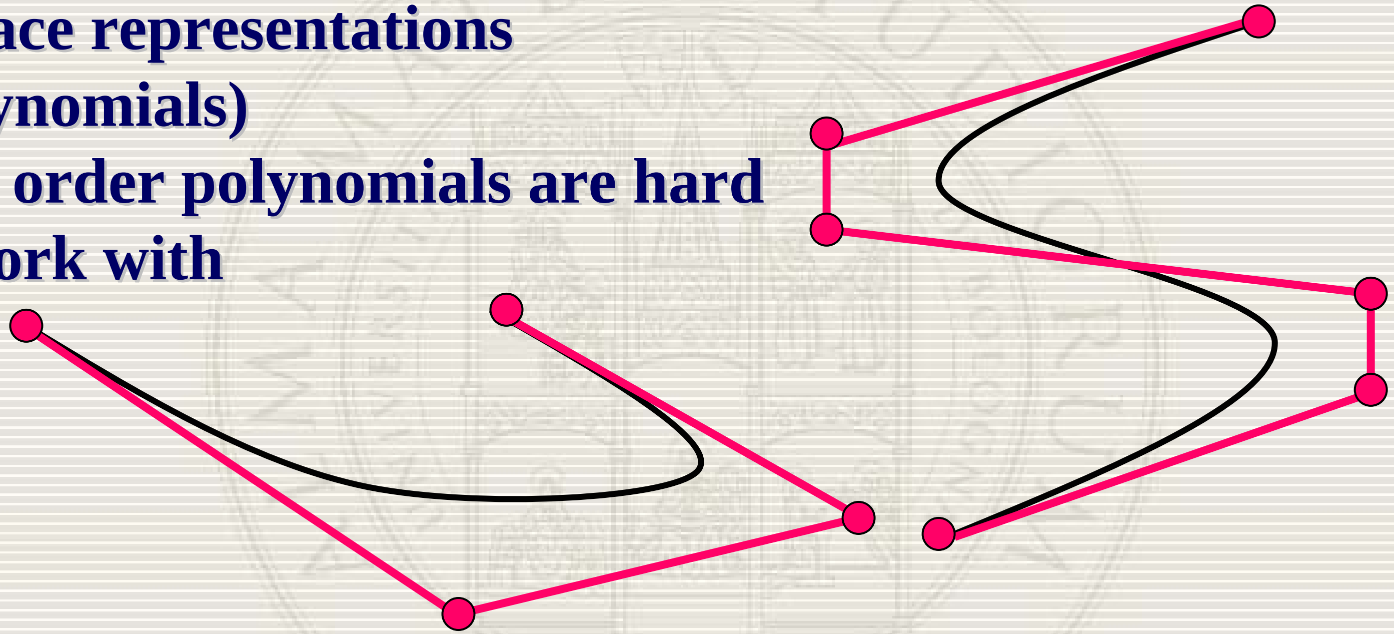




Hierarchical Modeling

Interactive modeling: spline

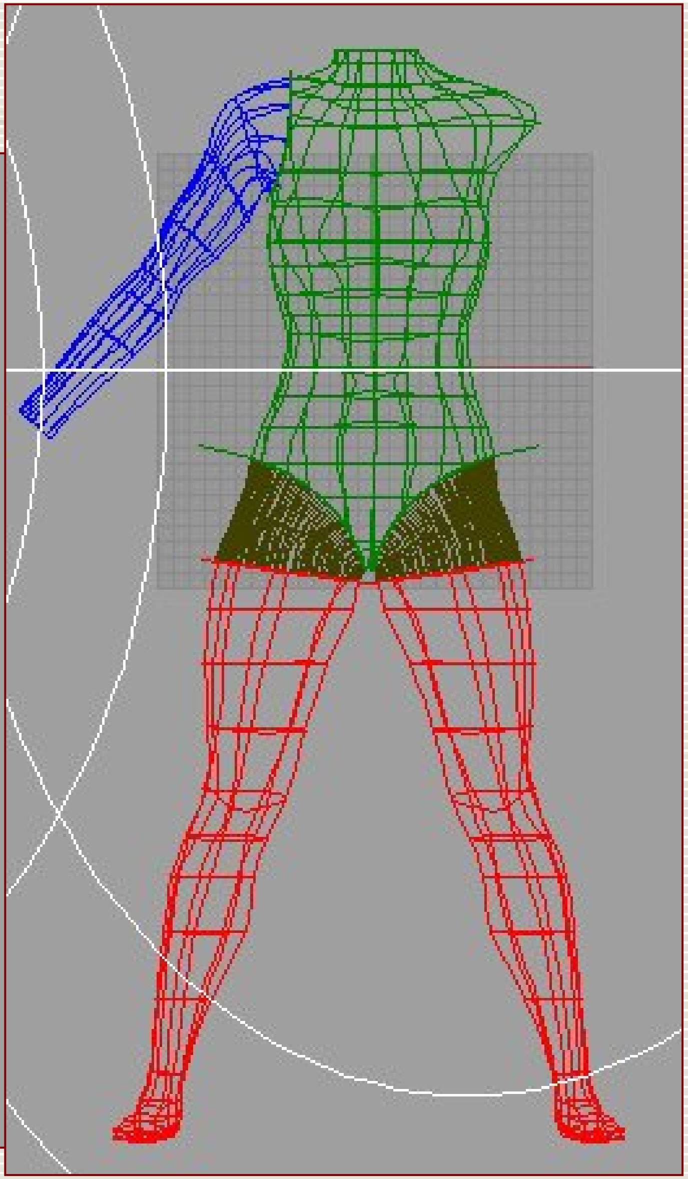
Continuous mathematical surface representations (polynomials)
high order polynomials are hard to work with

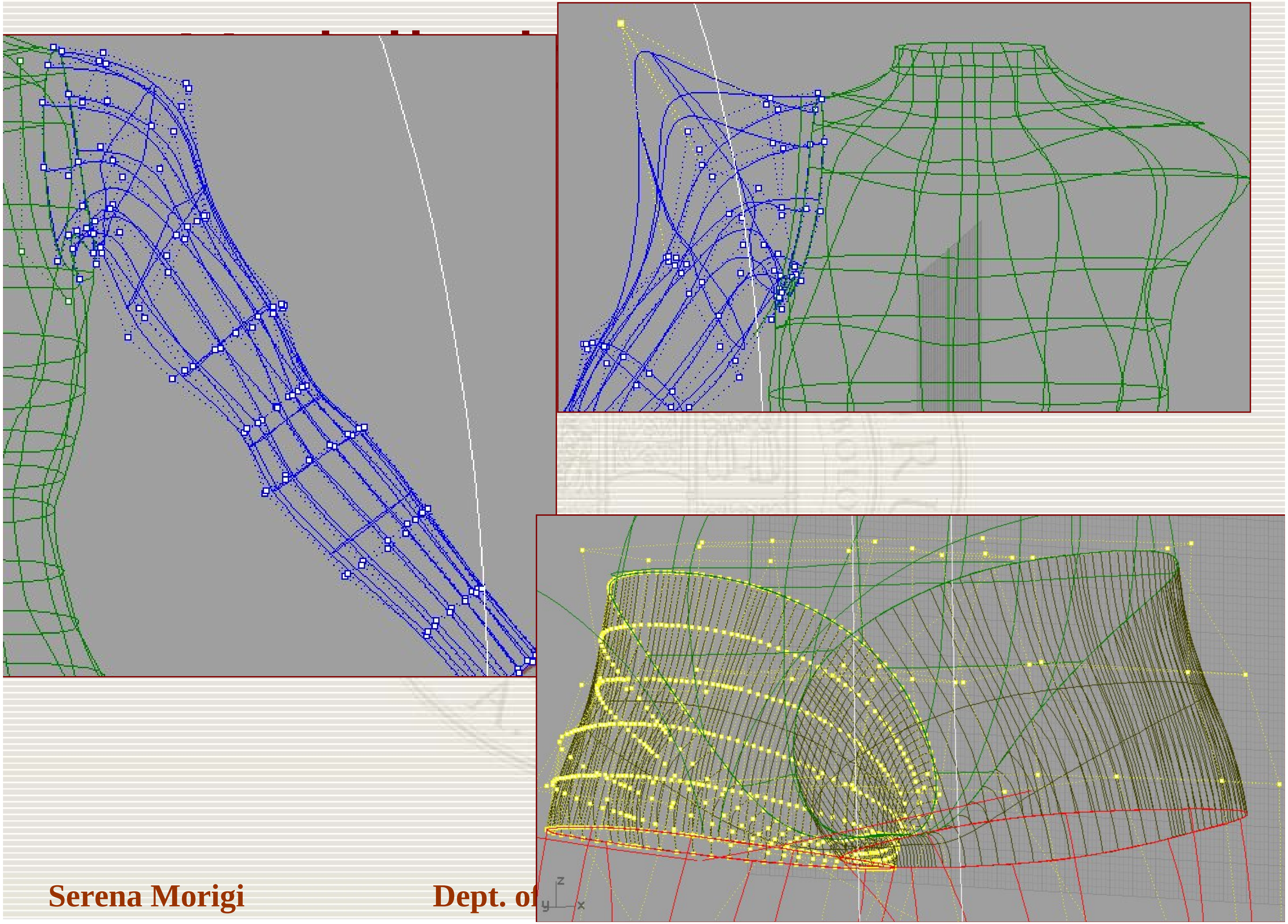


$$C(t)=\sum_{i=0}^{n} P_i B_i^n(t) \quad t\in[0,1] \qquad B_i^n(t)=\binom{n}{i} t^i (1-t)^{n-i}, \quad i=0,..,n$$

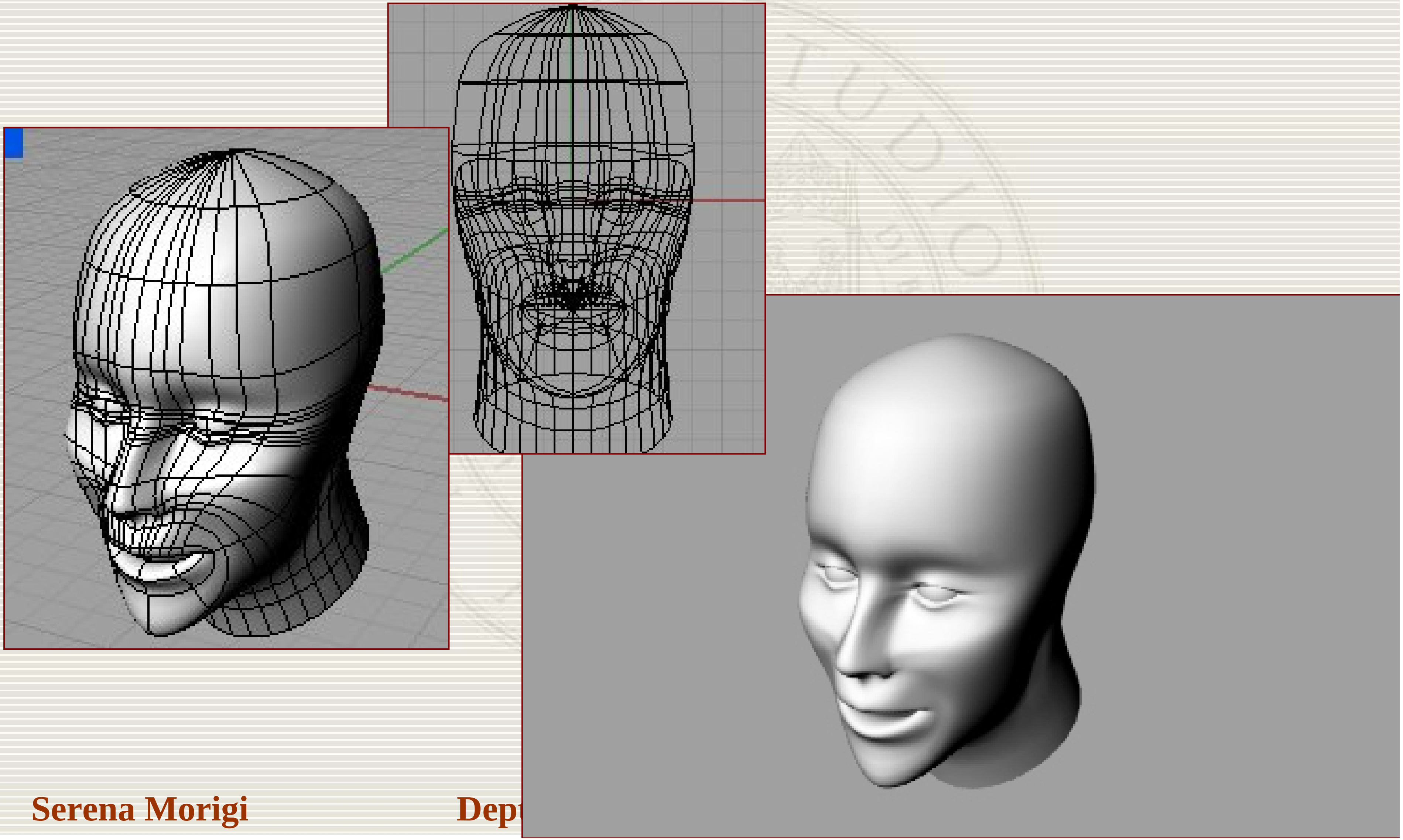
P_i control point **Bernstein basis functions**

Patch: 2D and 3D curved surfaces
Non-Uniform Rational B-Splines (NURBS)



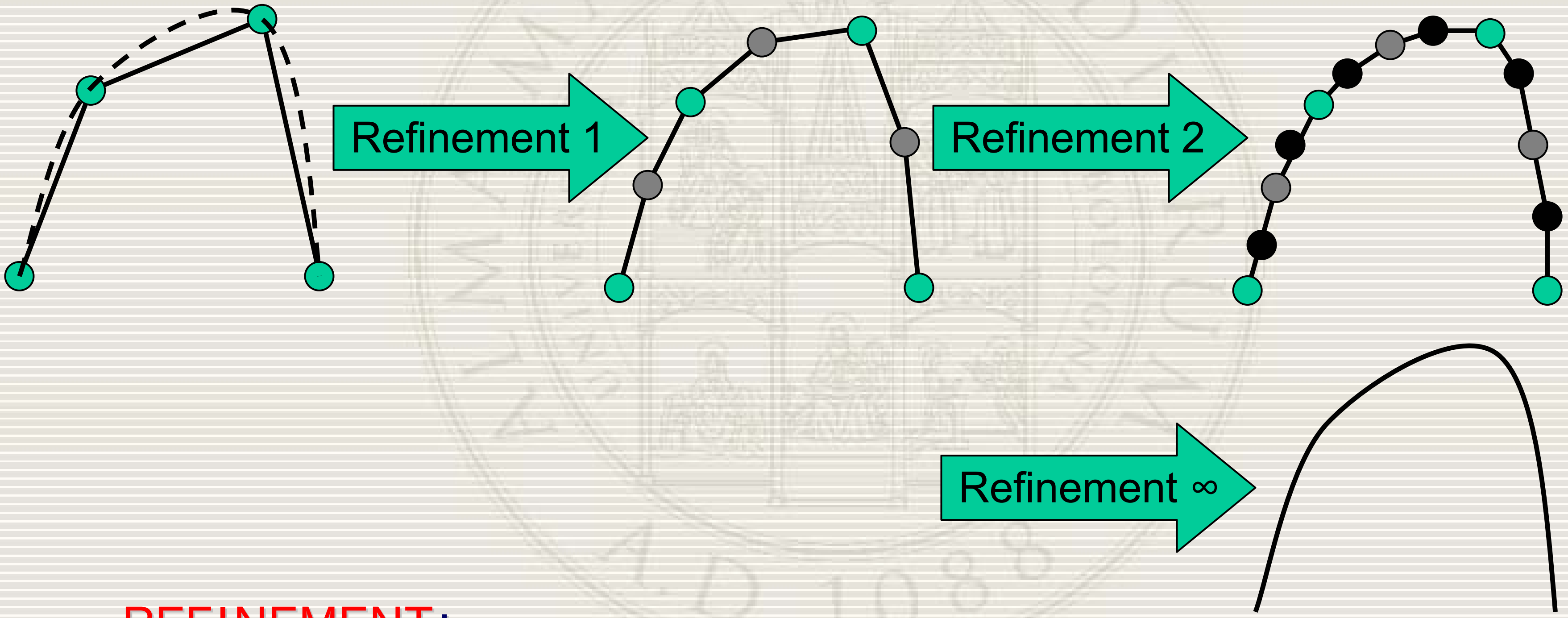

z
y
x

Interactive modeling: patch spline



Subdivision Modelling

Bézier curve, spline and subdivision are based on an algorithm that makes a curve from a control polygon.

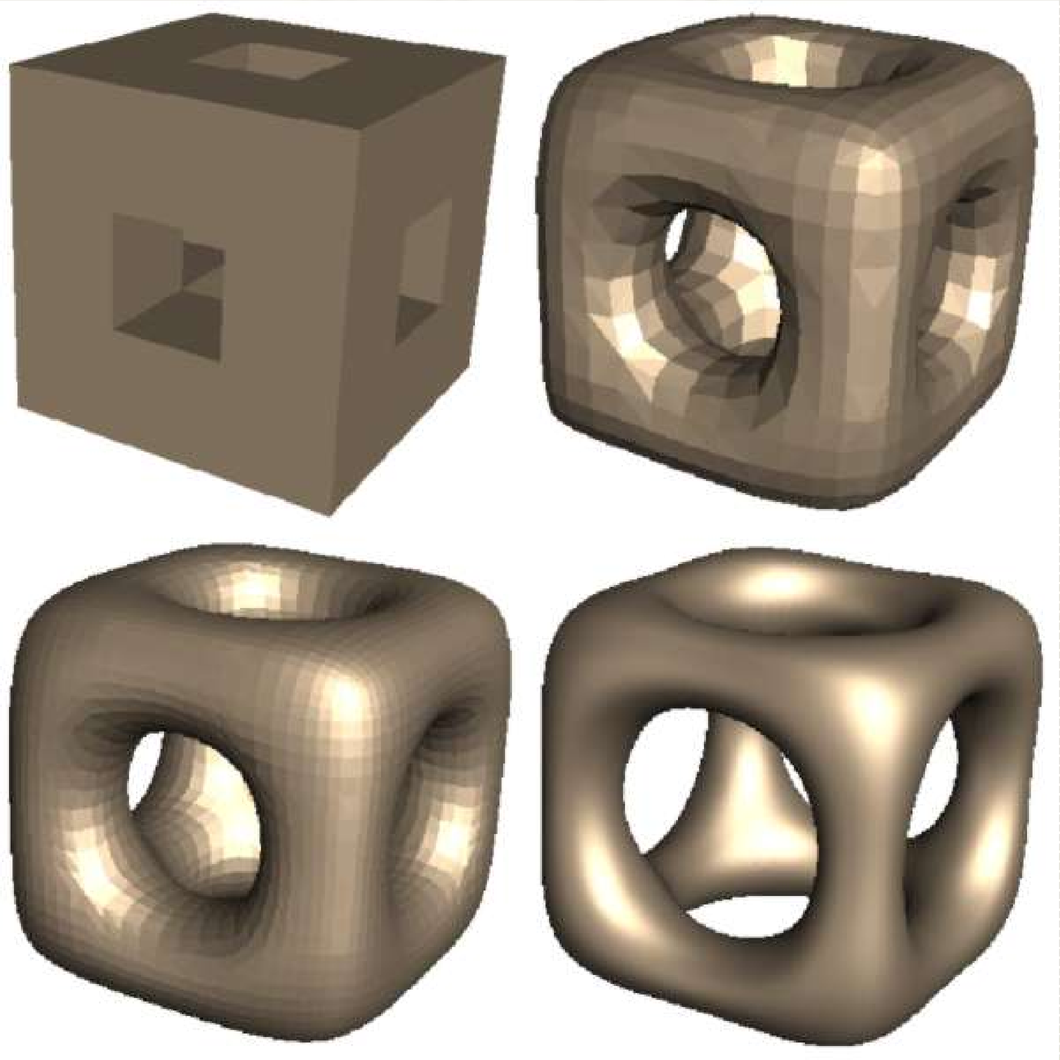


REFINEMENT:

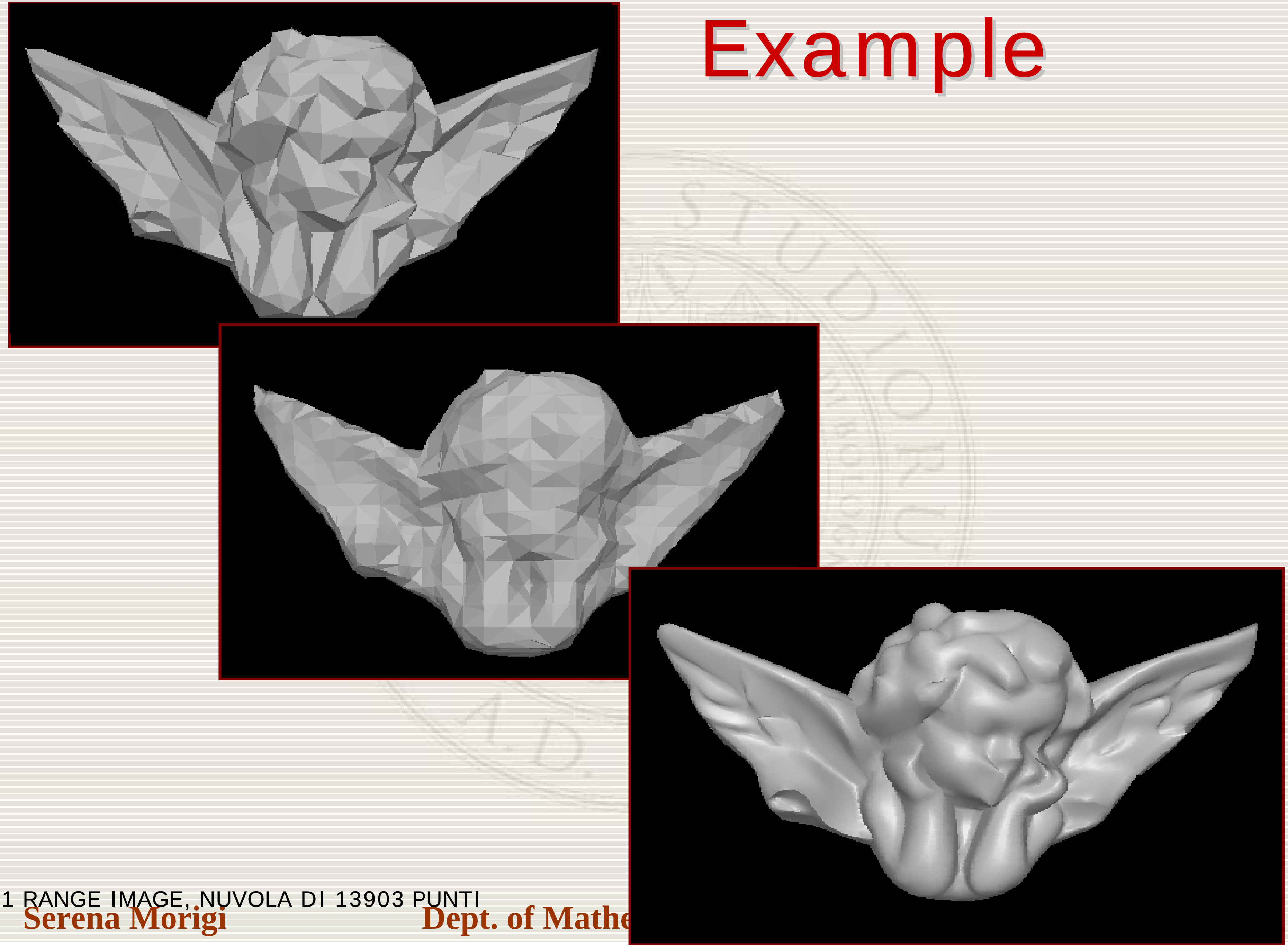
The smoothed final curve is obtained iterating a refinement procedure

Subdivision Surfaces

subdivide triangles into more triangles, moving to a continuous limit surface

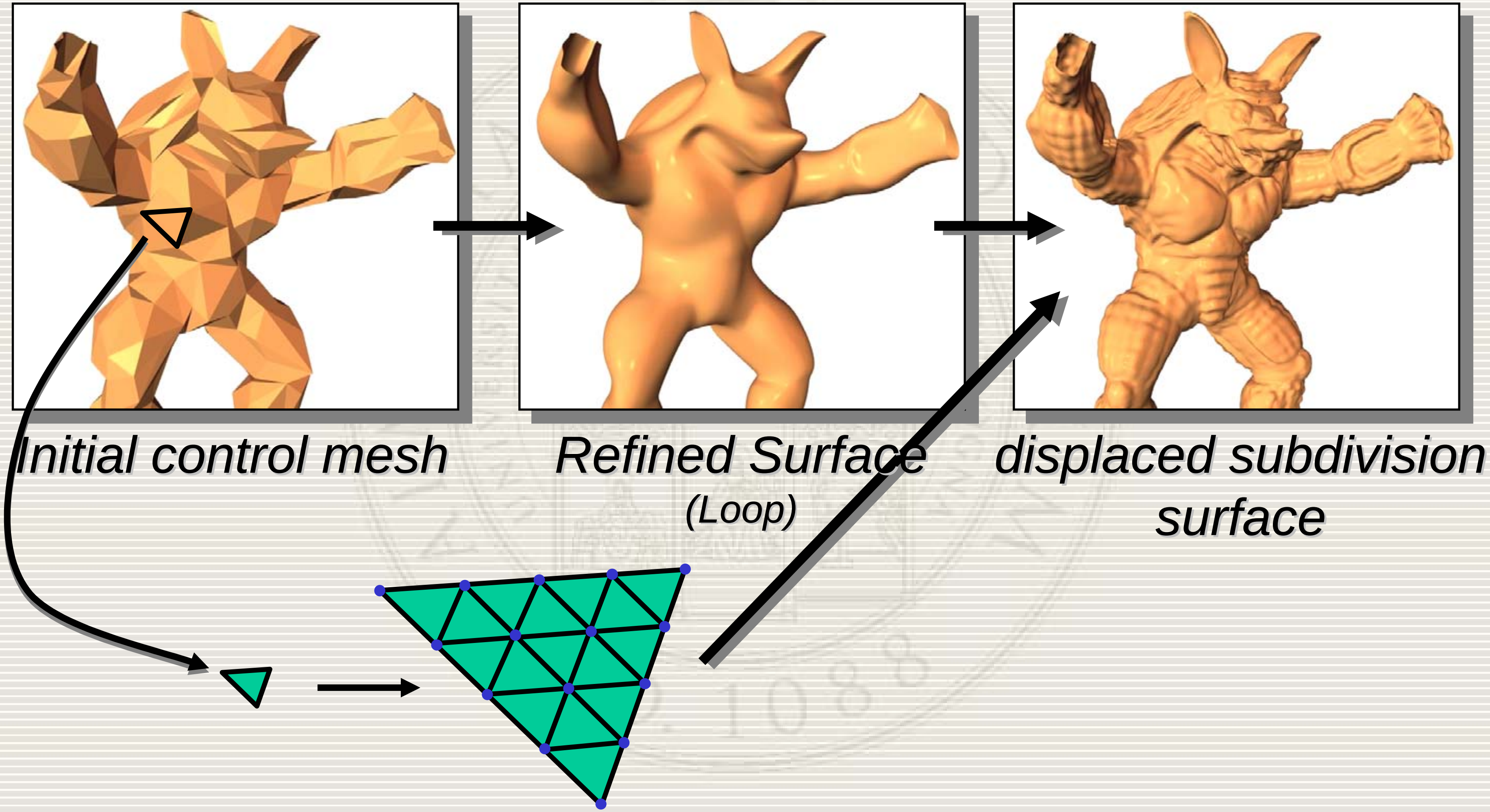


Example



1 RANGE IMAGE, NUVOLA DI 13903 PUNTI

Subdivision surface with displacement mapping



Initial control mesh

Refined Surface

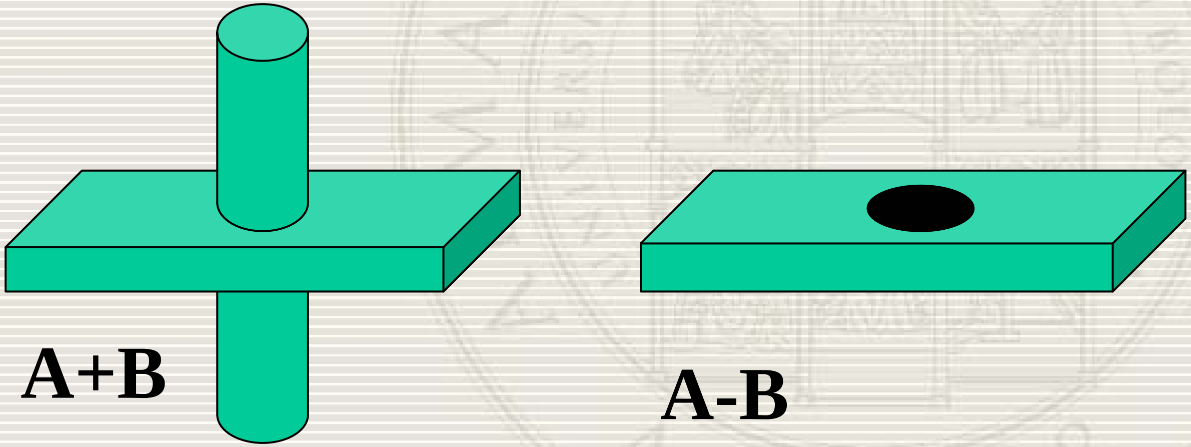
(Loop)

displaced subdivision surface

scalar displacements

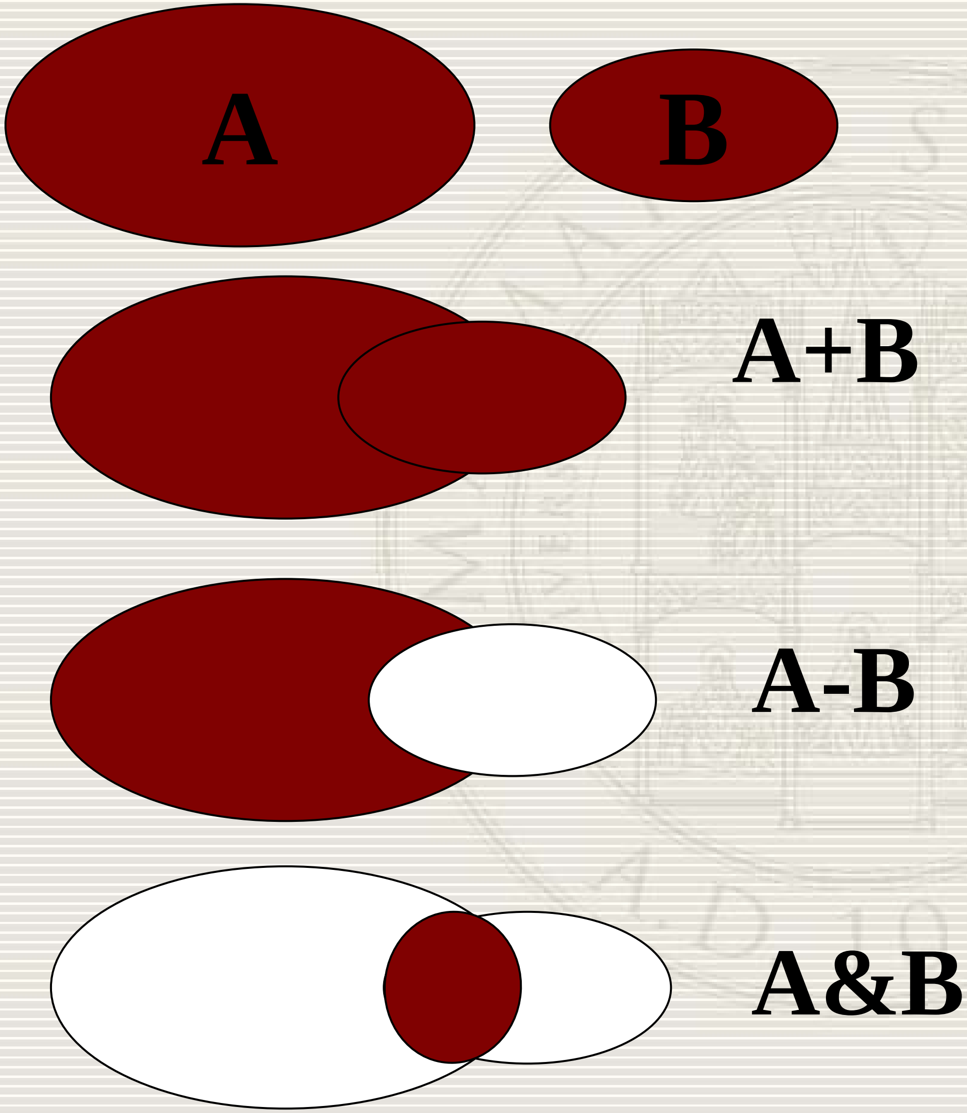
Constructive Solid Geometry

- Combination rules for solids
- Each combines two solids



- CSG tree

CSG



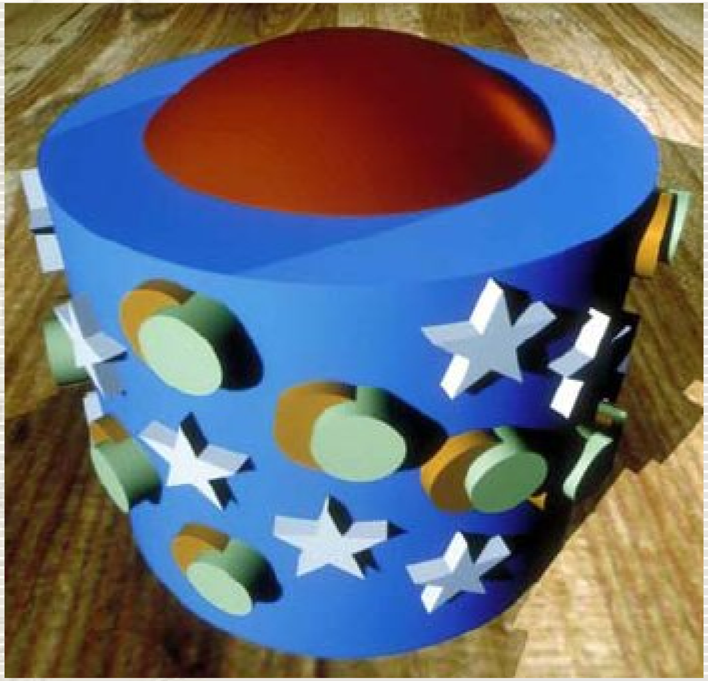
A

B

A+B

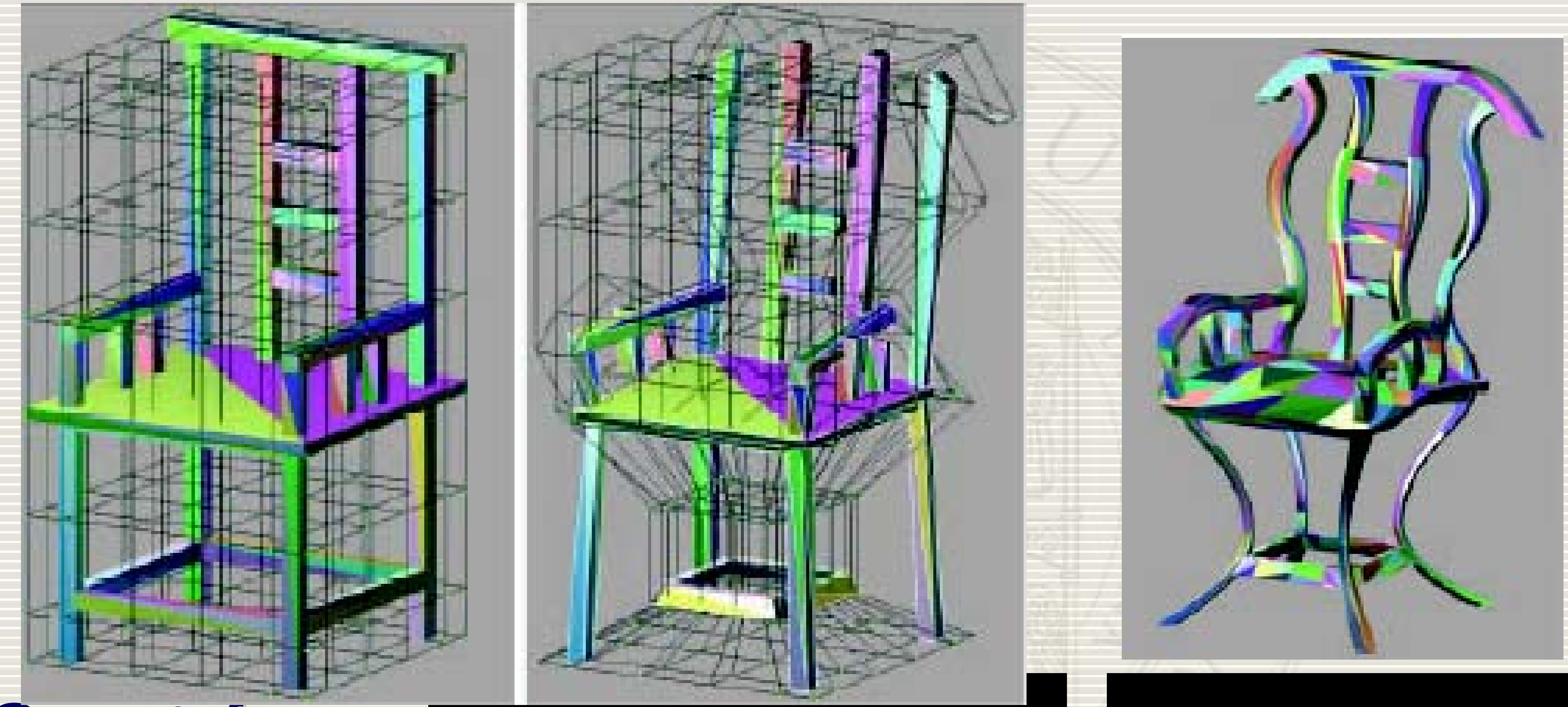
A-B

A&B



Free-form deformation

- Change the space, not the object

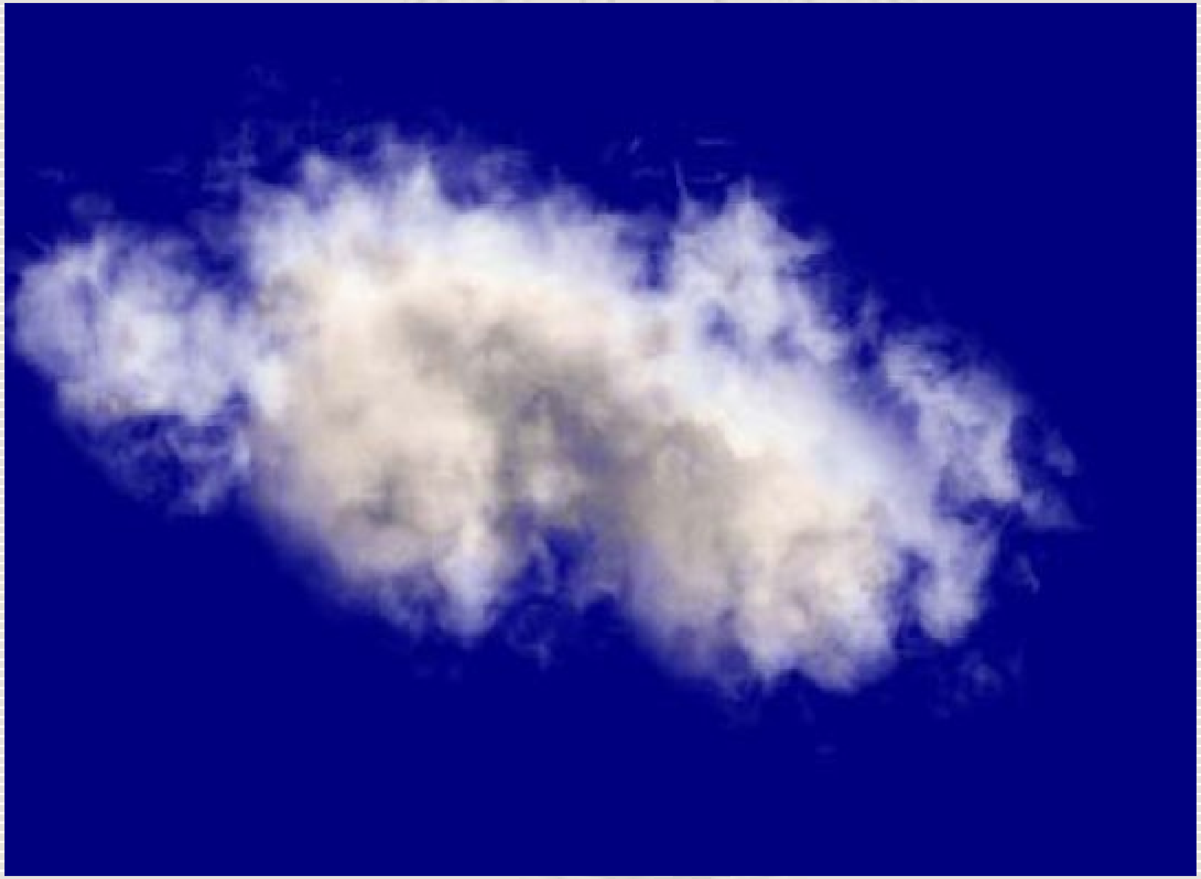


- Great for animation

Volumetric Primitives

- Voxel :
  - Volume that encloses some space (open vs. closed)
- Adaptive (varying sizes):
  - octree
- Uniform (equally-sized):
  - Grids

Example



Procedural Modeling

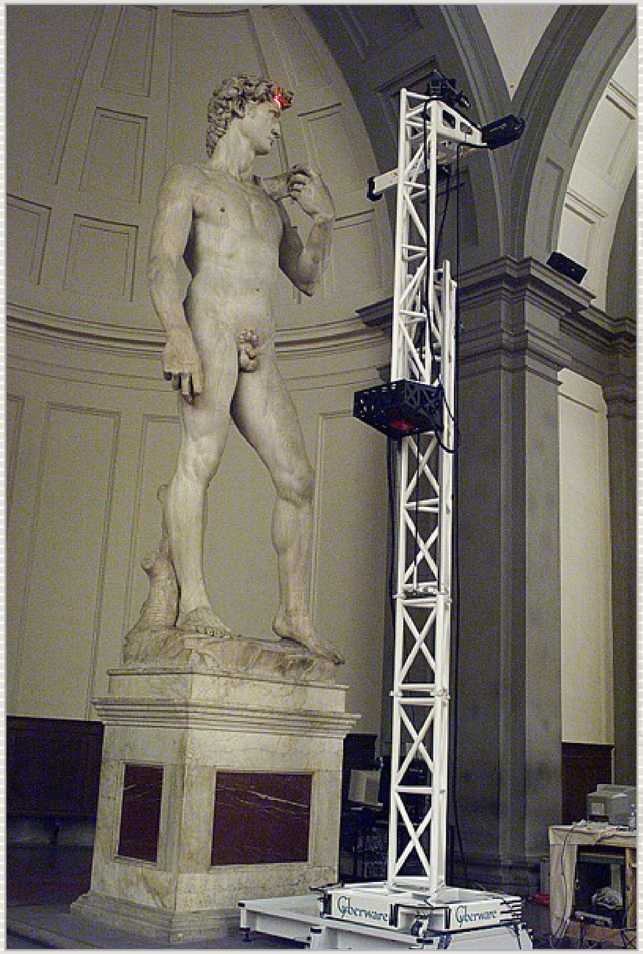
- Fractals
- Shape Grammars
- Particle Systems

Procedural Modeling

Captured Modeling:

Polygonal models (mesh) by 3D scanners

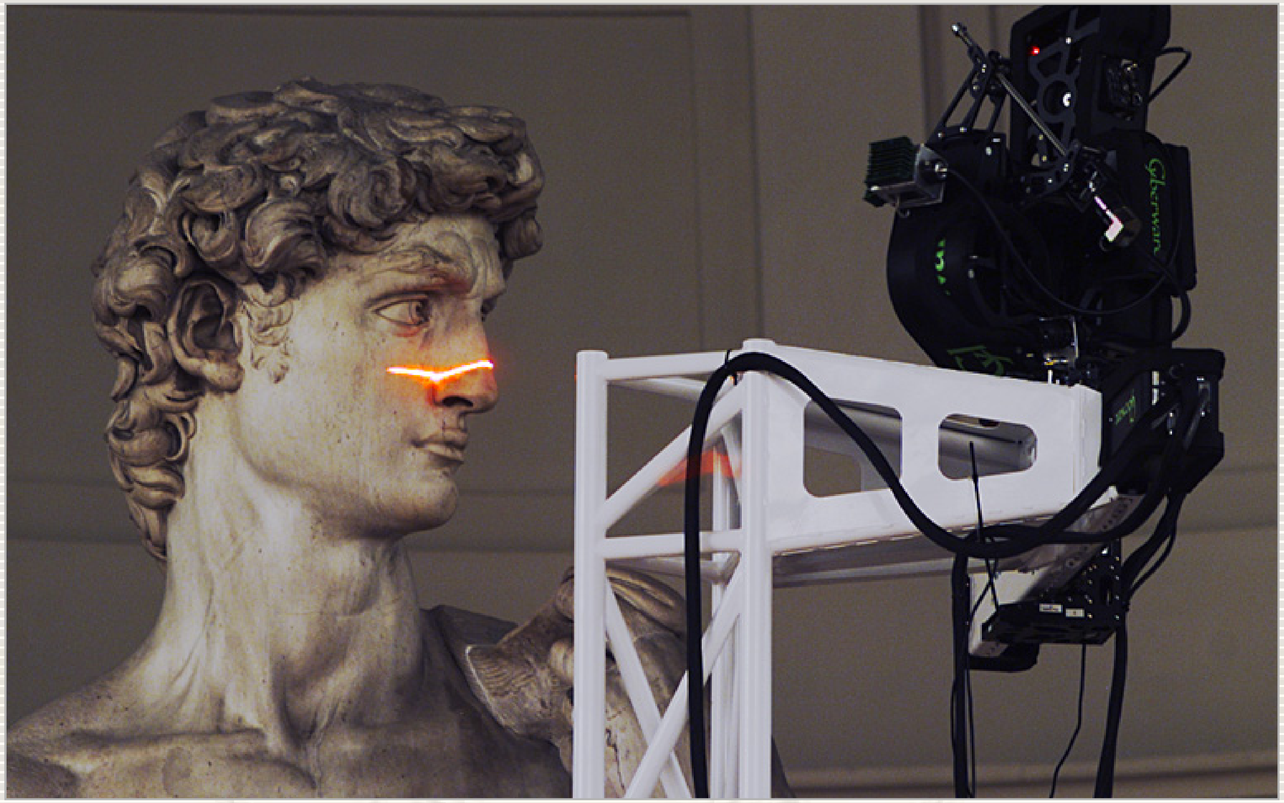
Michelangelo Project: "Scanning the David"



- 480 individually aimed scans
- 2 billion polygons
- 7,000 color images
- 32 gigabytes
- 30 nights of scanning
- 22 people

http://graphics.stanford.edu/projects/mich

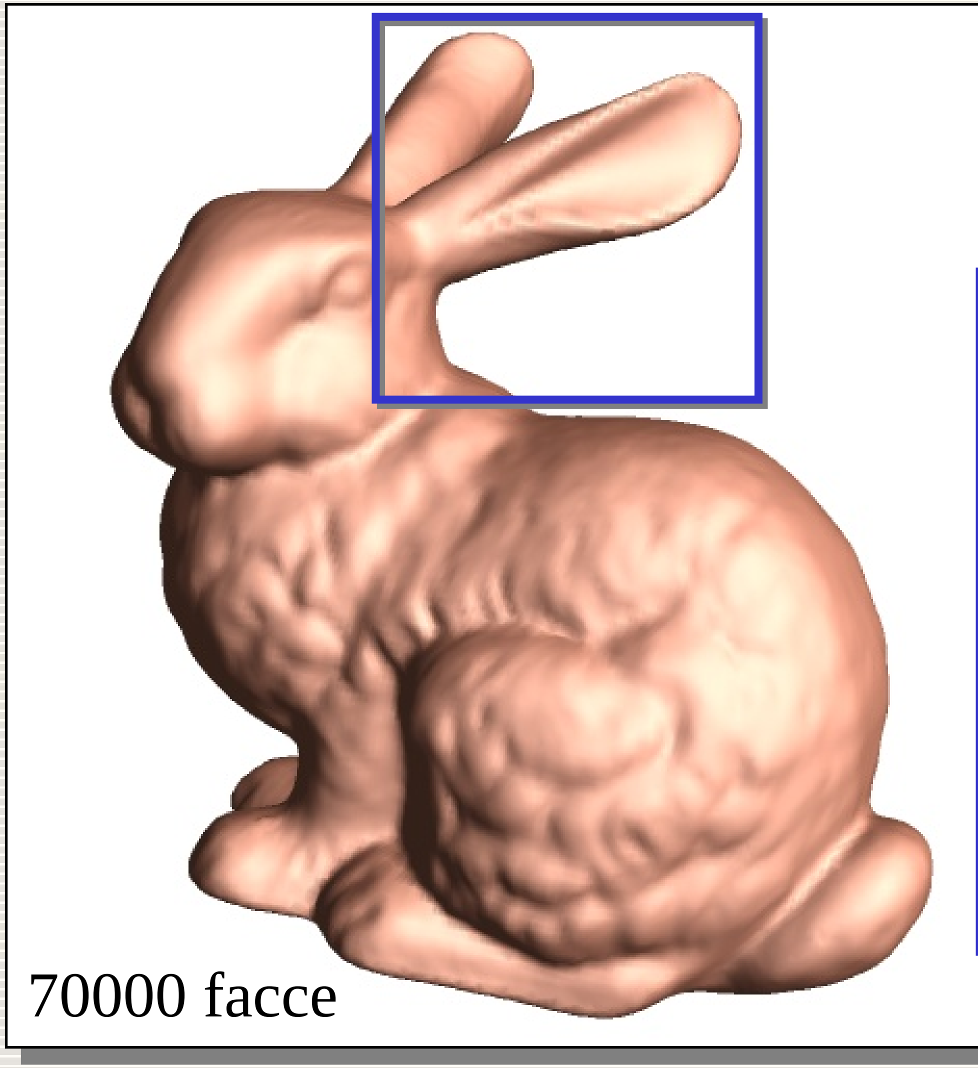
Scanning Michelangelo's David

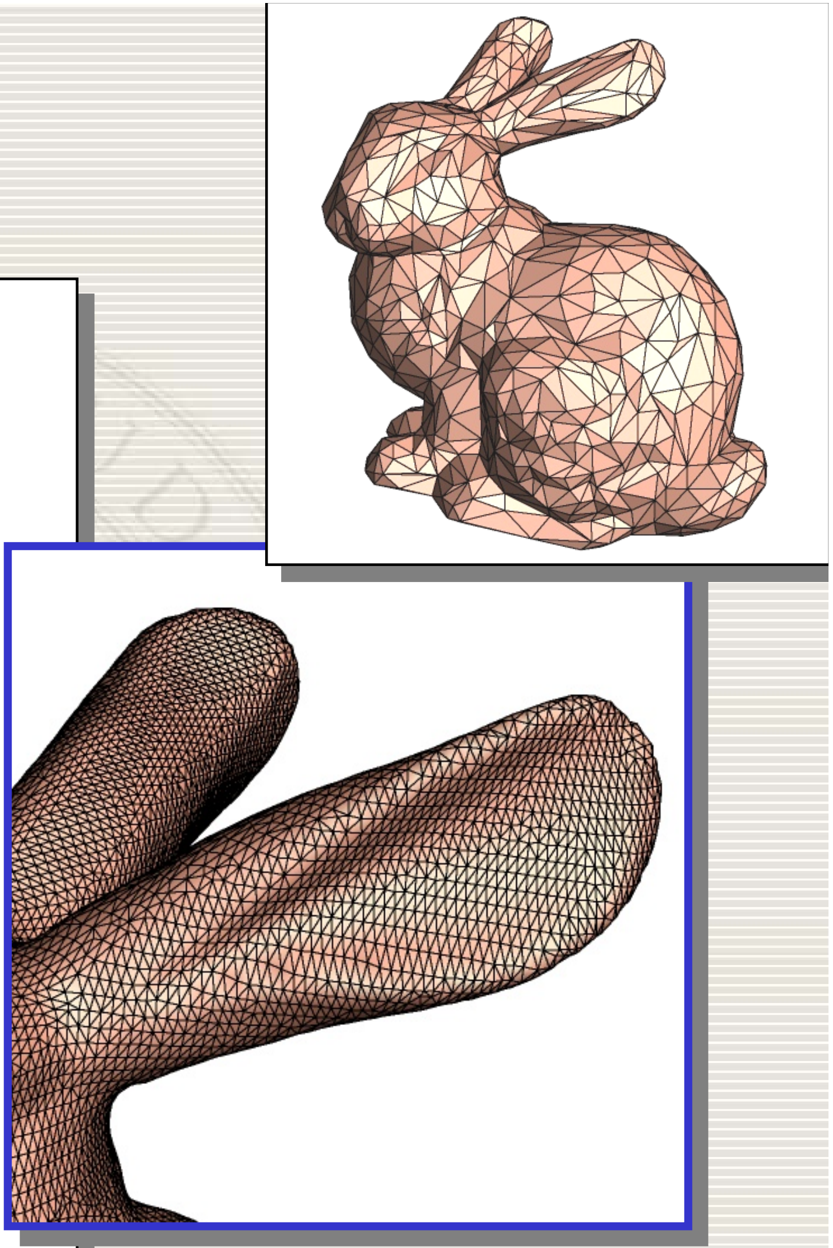


- David is 5 meters tall
- chisel marks need 1/4mm
- dynamic range of 20,000:1

- $20{,}000^2$ = 1 billion polygons
- 14cm wide working stripe
- David was ~30 stripes around

..3D Scanner

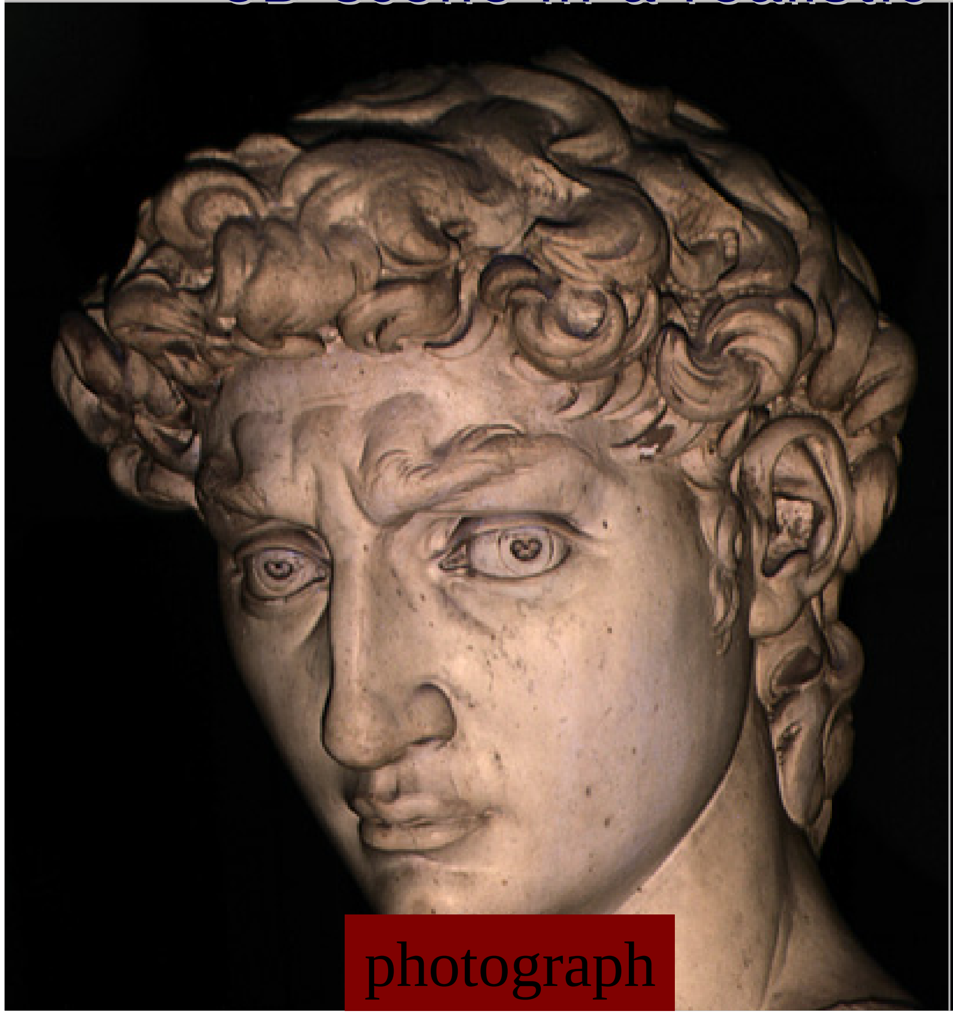




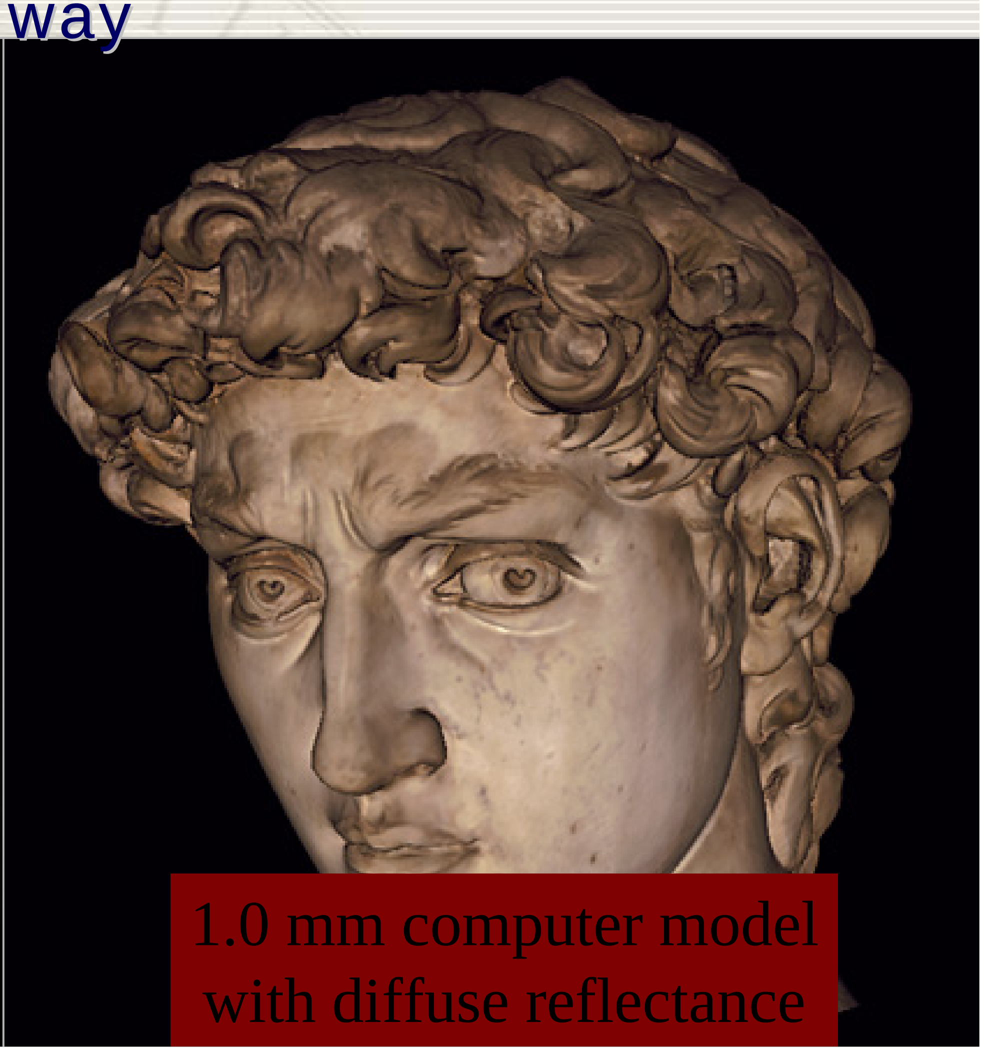
H.Hoppe

RENDERING
from models to images

Photorealistic rendering refers to rendering a 3D scene in a realistic way



photograph



1.0 mm computer model
with diffuse reflectance

Photorealistic Rendering

Simulating nature..

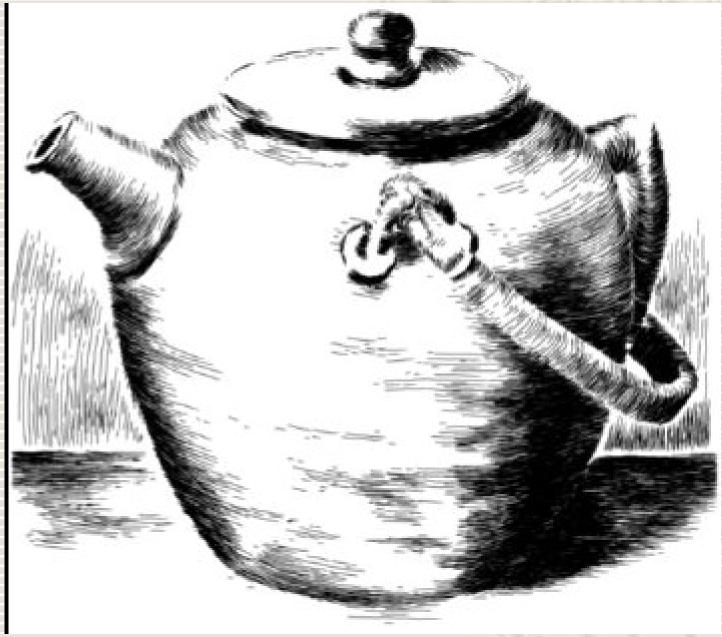
HENRIK WANN JENSEN 1995

.. anything else

nati

Non-photorealistic rendering

- Artistic rendering—trying to evoke hand-drawn or hand-painted styles, such as charcoal sketching, pen and ink illustration, or oil painting (Cartoon rendering style)



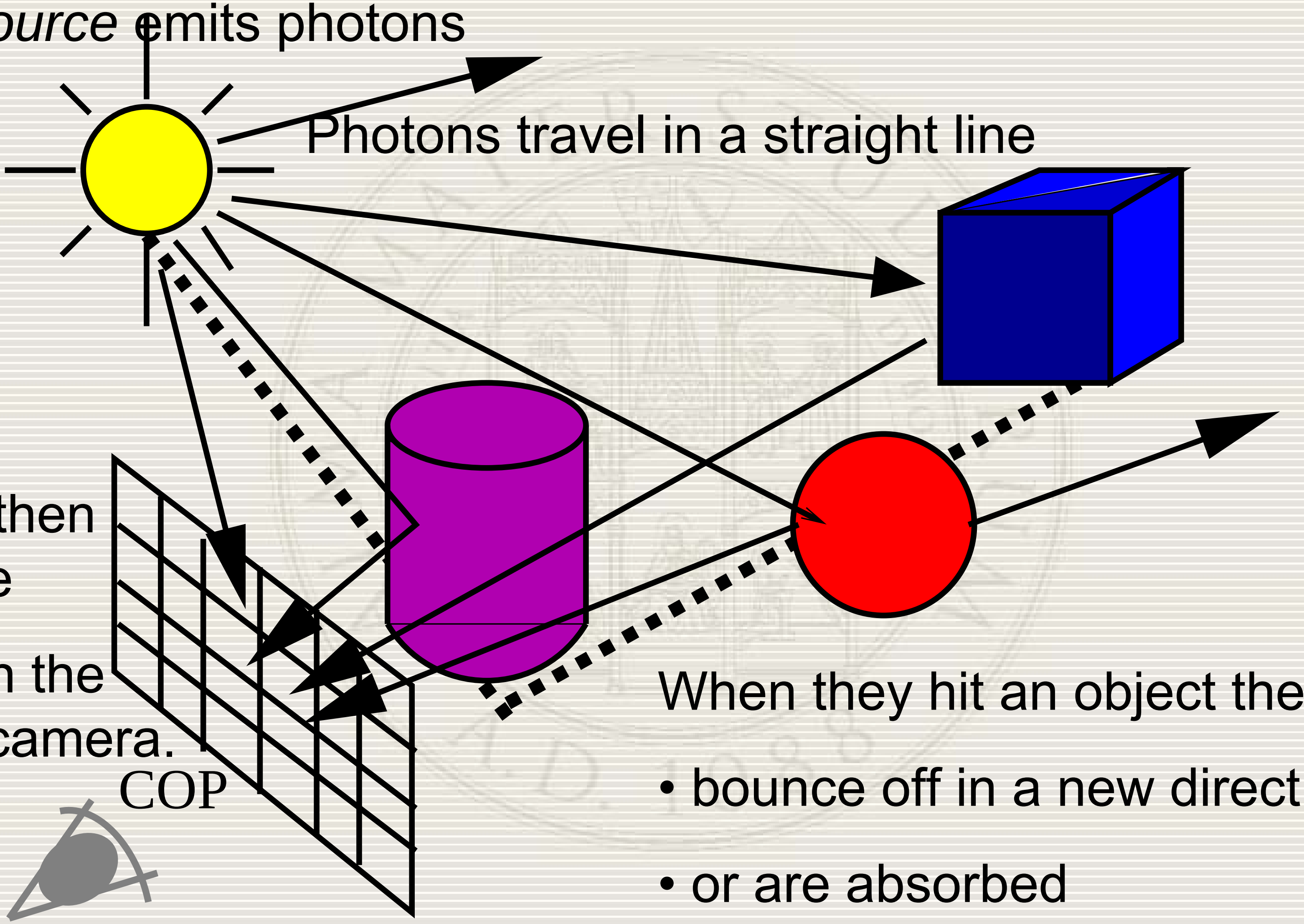
Tonal Art Maps

Rendering:
interaction between light and material

Lighting

- Modeling Lighting
- Reflectance
- Texture
- Shadows (visibility)
- Interreflections

Light



Source emits photons

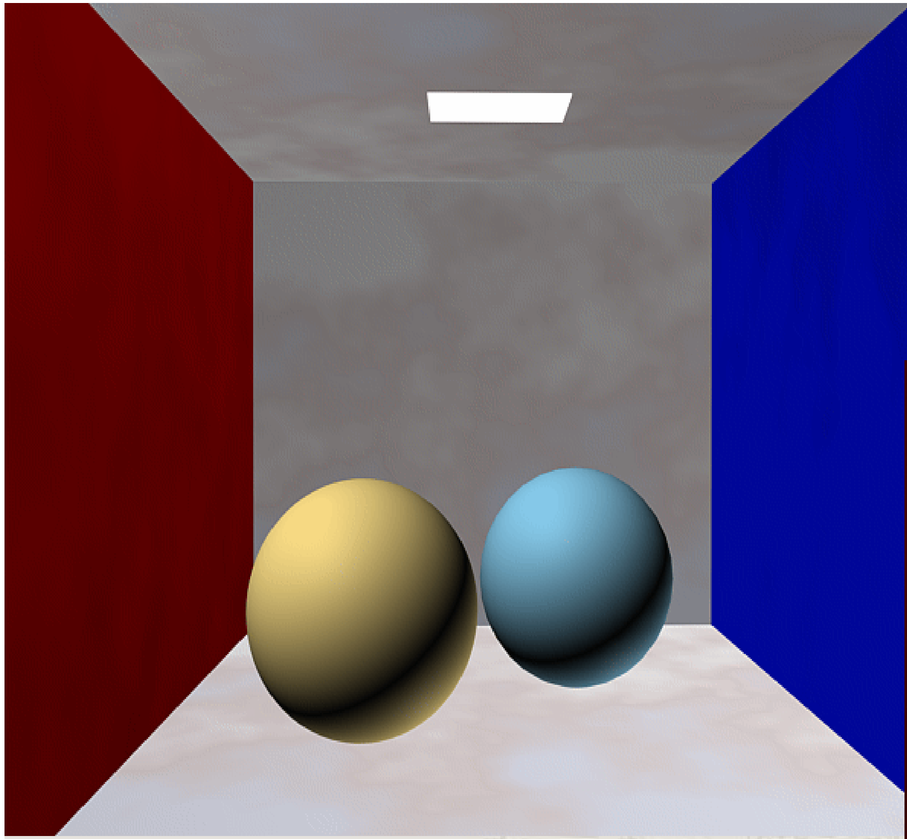
Photons travel in a straight line

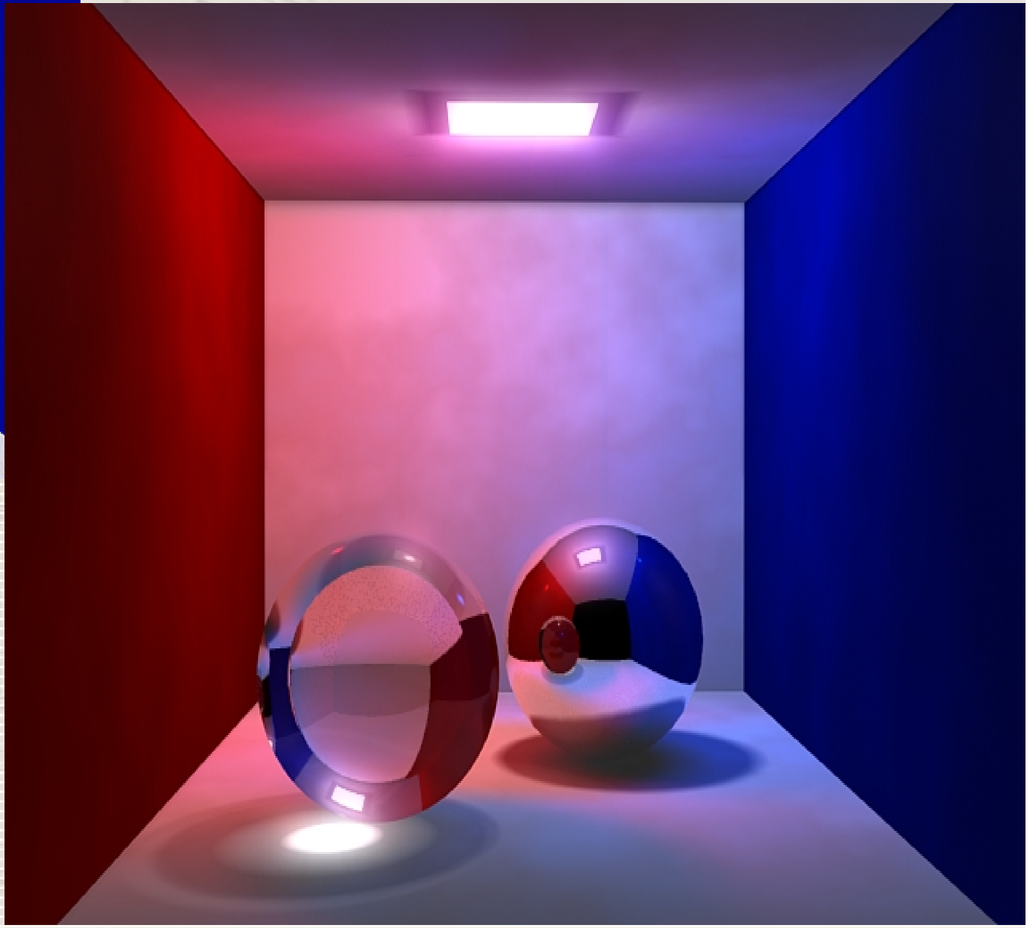
And then some

reach the eye/camera.

When they hit an object they:

- bounce off in a new direction
- or are absorbed

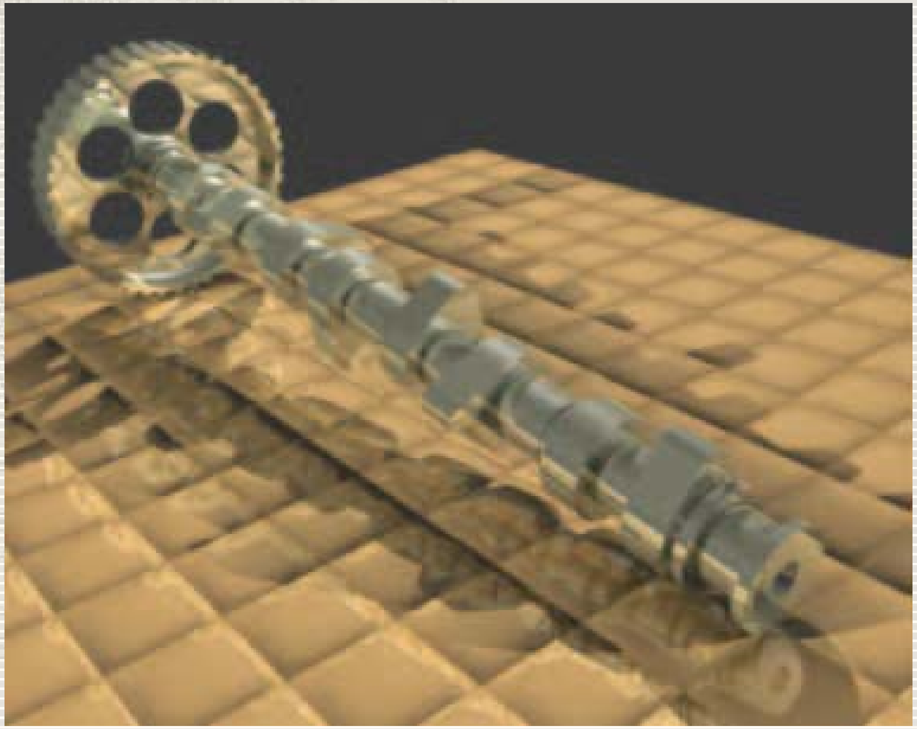




Shadows (visibility)
Interreflections

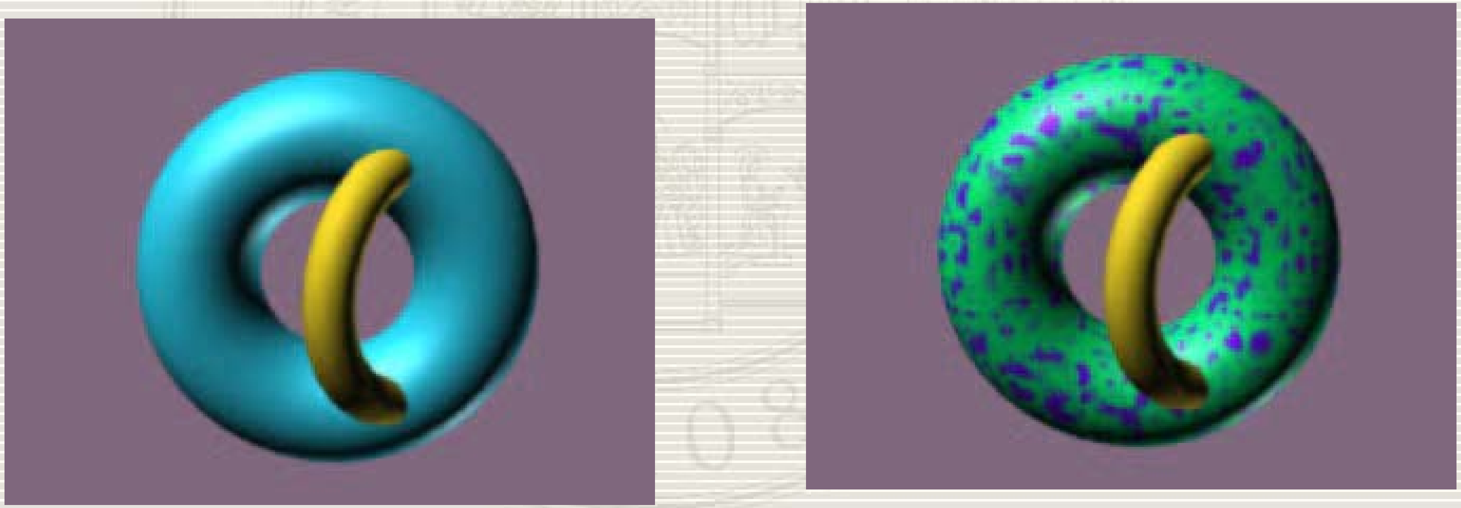
Ray tracing

shoot viewing rays from viewer's eyepoint through each pixel into scene, and see what objects they hit. Return color of object struck first. If object is transparent or reflective, recursively cast ray back into scene and add in reflected / refracted color

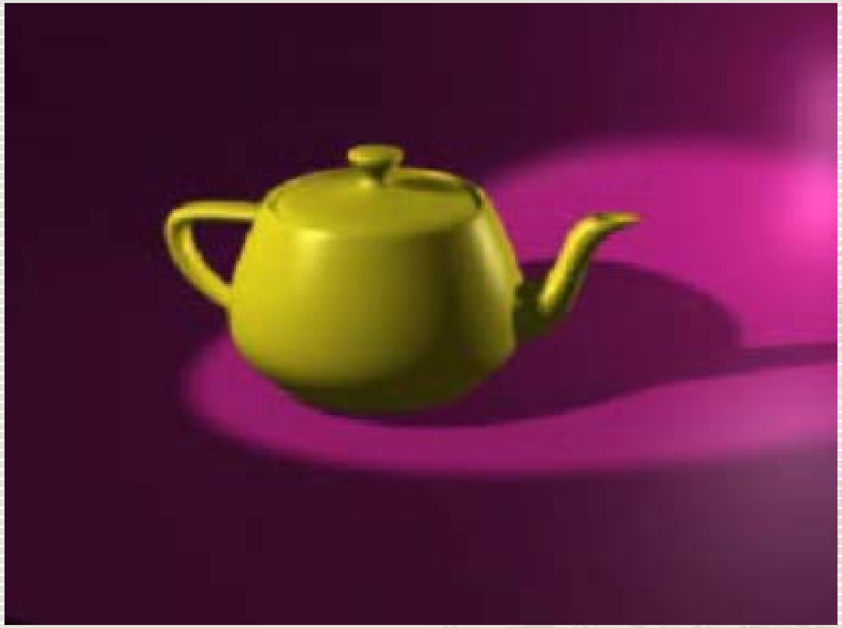


Texture Mapping

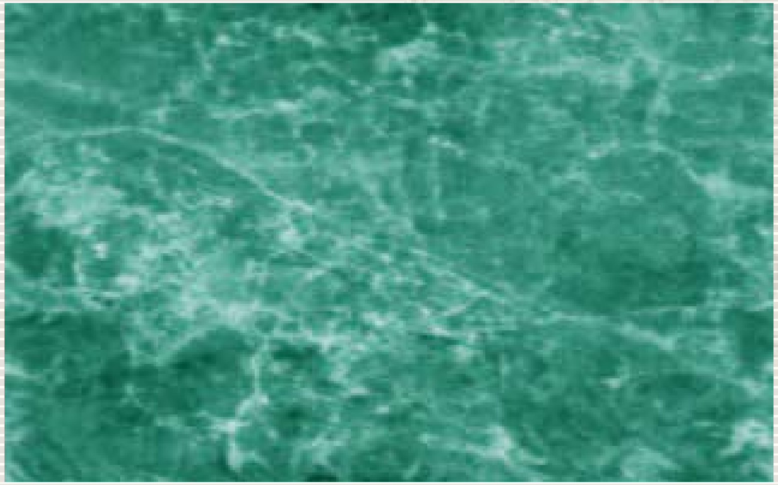
map an image onto surface geometry to create appearance of fine surface detail. A high level of realism may require many layers of textures.

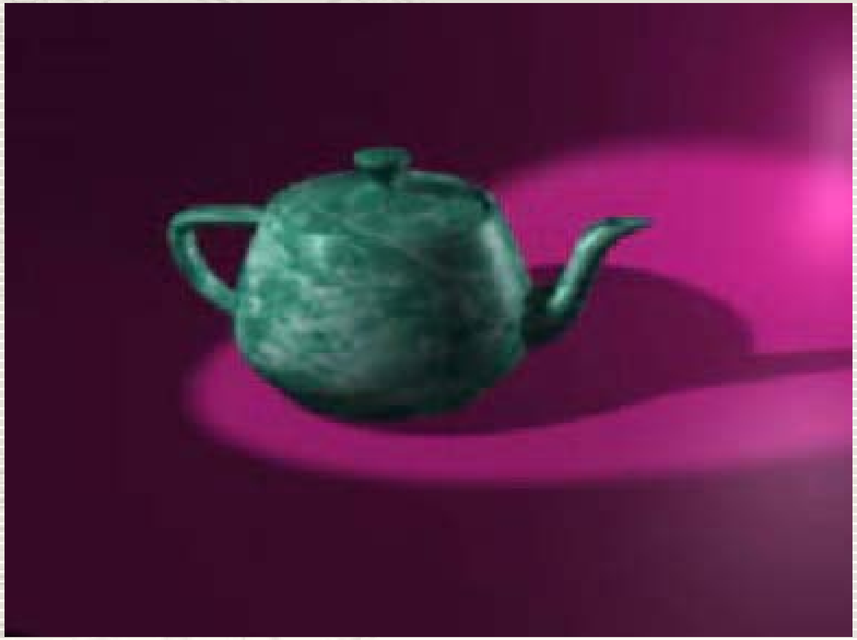


Details created by texturing



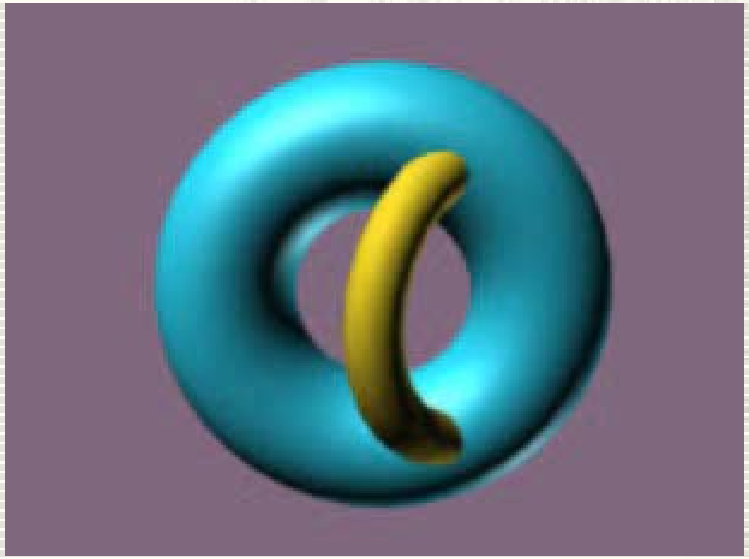
+

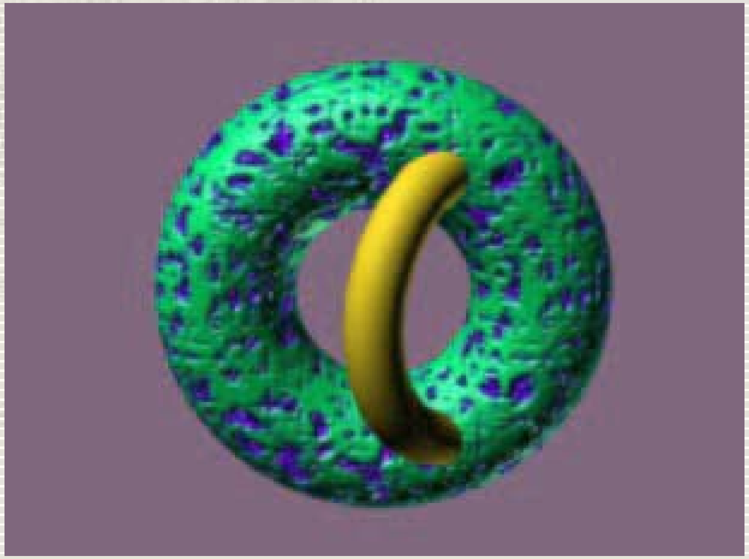




Bump Mapping

fake surface normals by applying height field (intensities in the map indicate height above surface). From height field calculate gradient across surface and use this to perturb the surface normal.



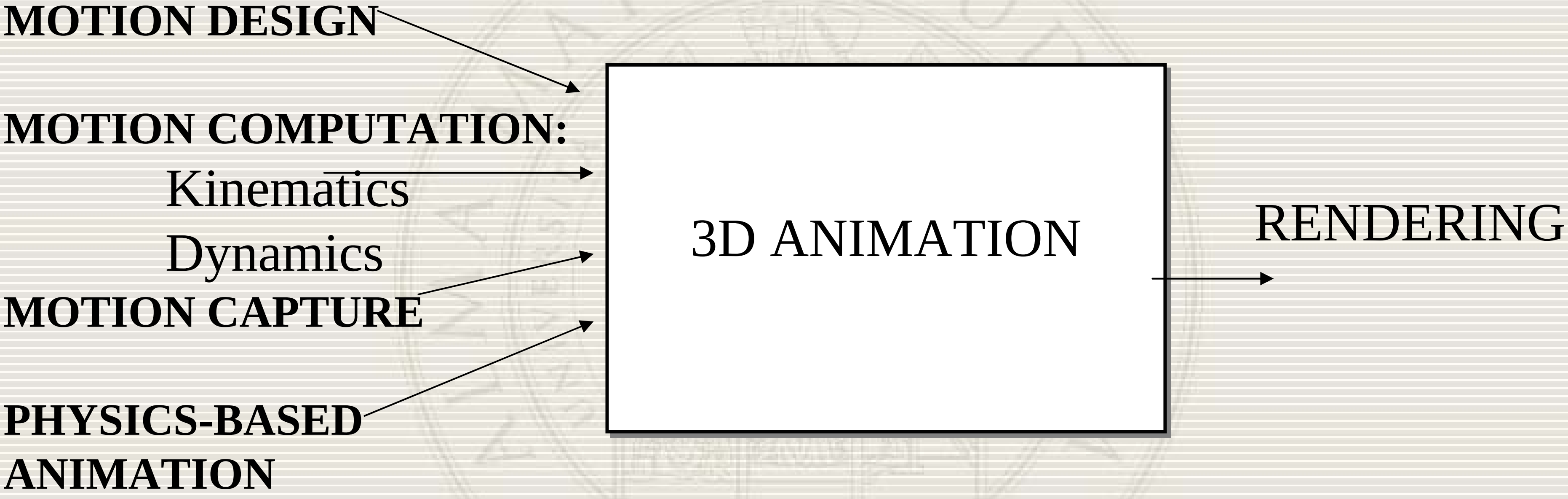


Environmental Mapping

multiple images (textures) which record global reflection and lighting on object. These images are resampled during rendering to extract view- specific information which is then applied as texture to object.

Computer Animation

Making things alive/Making them move



MOTION DESIGN

MOTION COMPUTATION:

Kinematics

Dynamics

MOTION CAPTURE

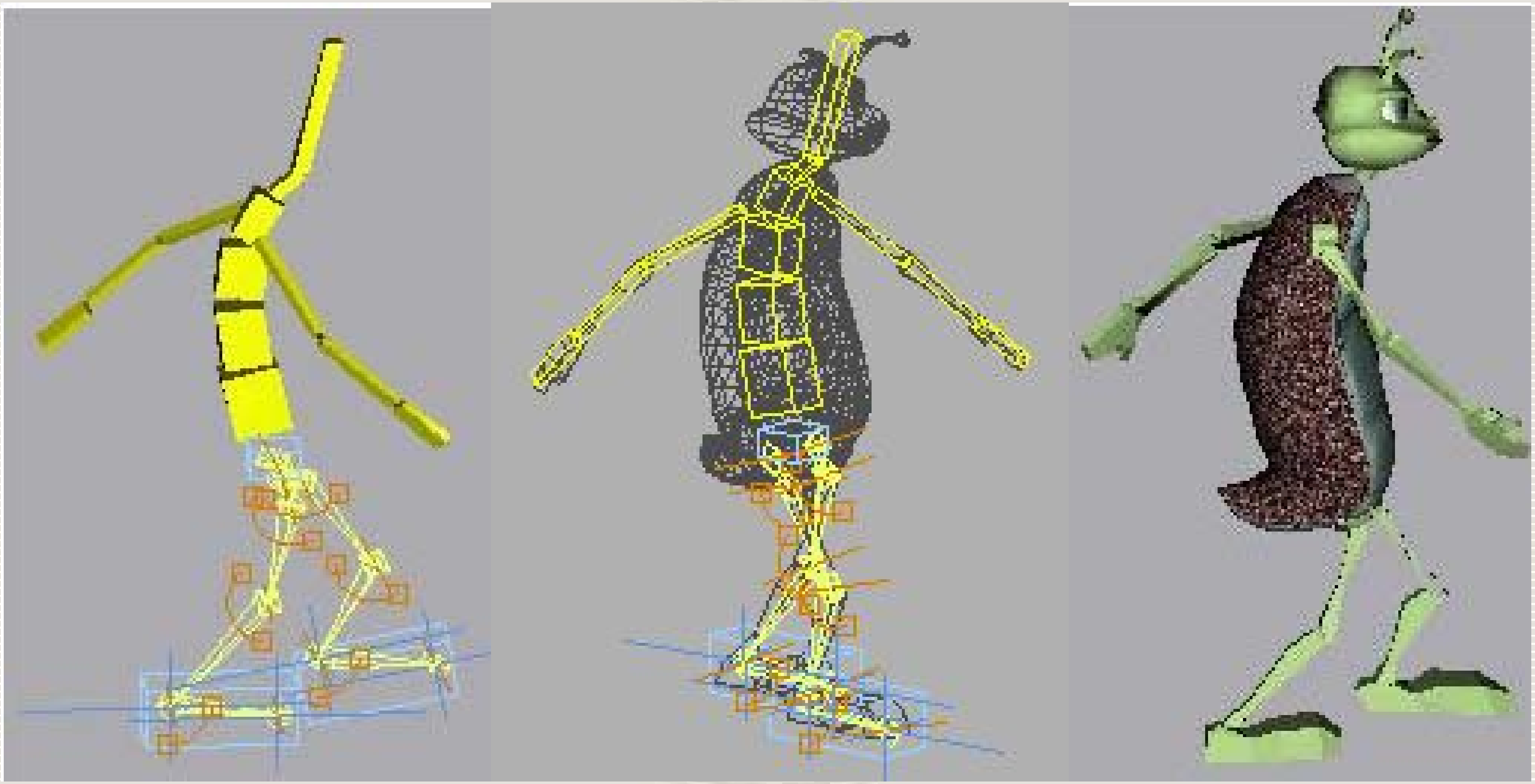
PHYSICS-BASED ANIMATION

3D ANIMATION

RENDERING

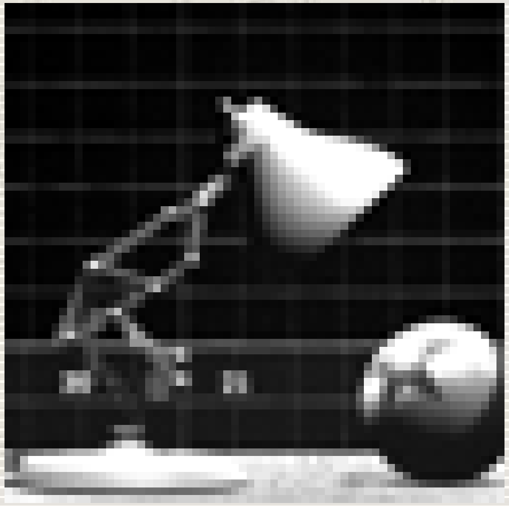
Actor+ motion law

Link, joint, skinned mesh

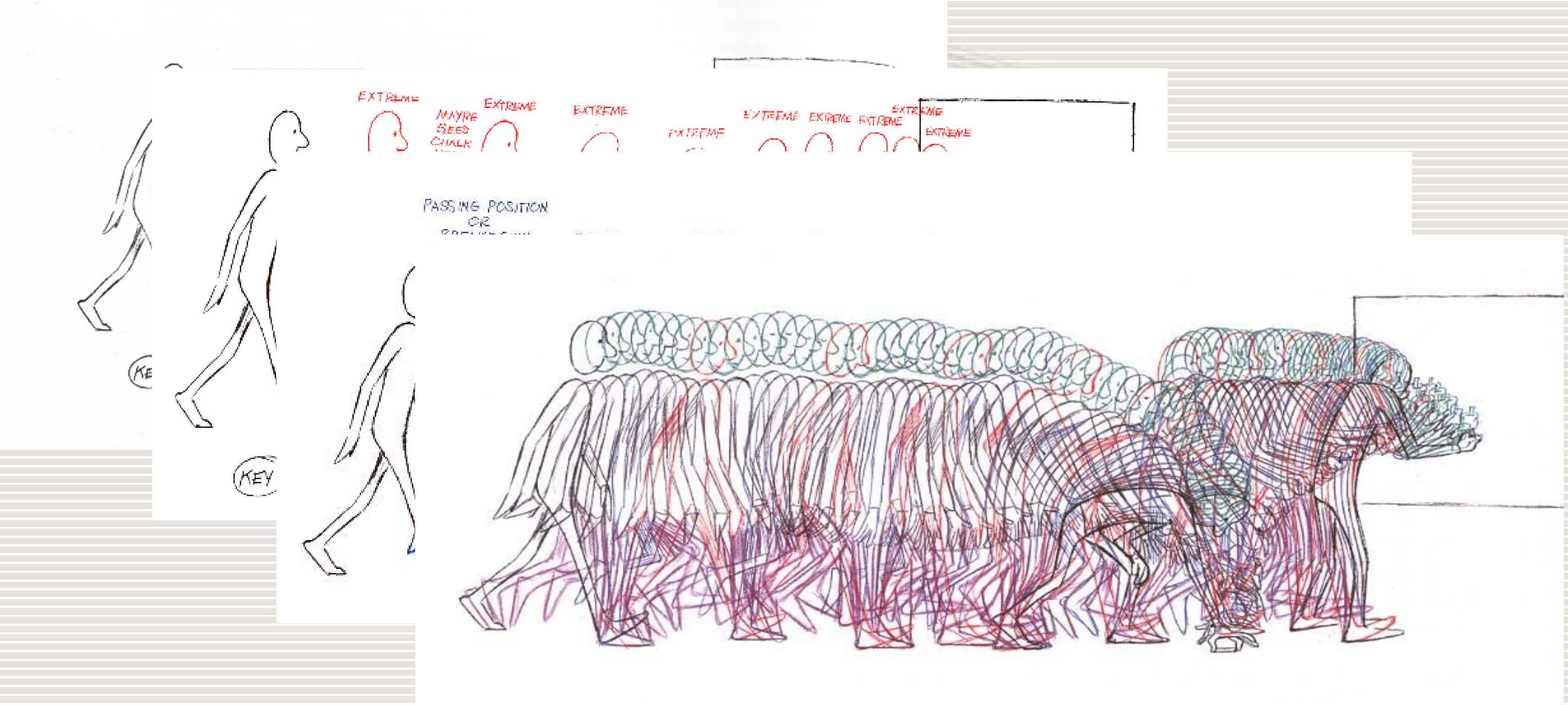


Animating actors

- Keyframe
- Physics-based animation
- Motion Capture



Keyframe Animation



Key Frame: fundamental steps

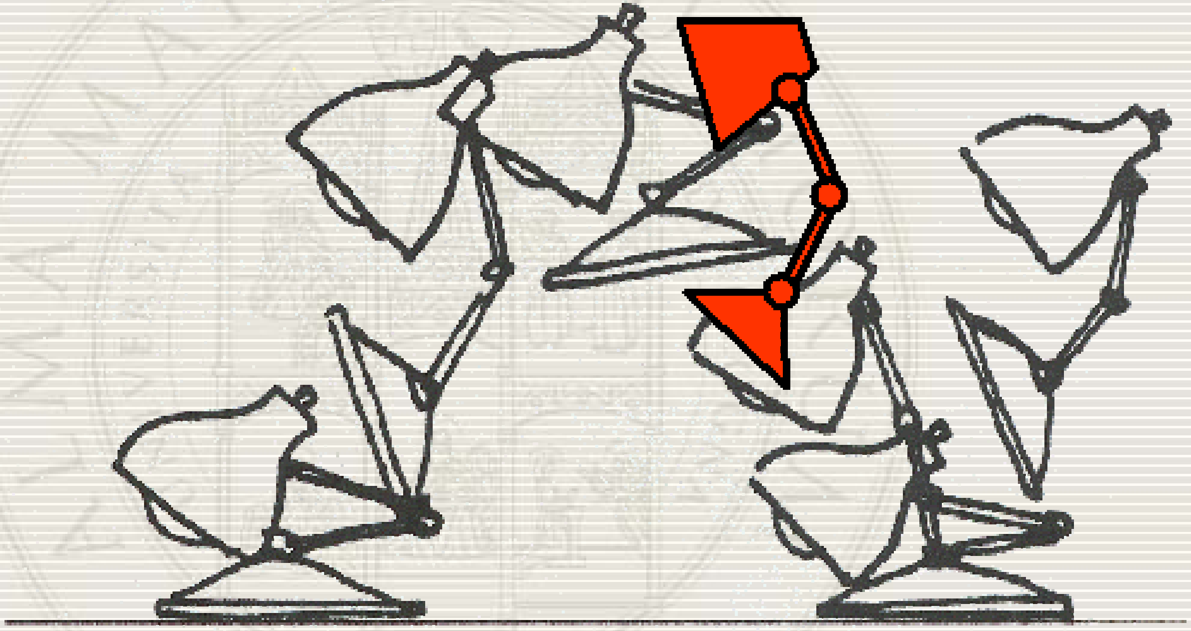
Extremes: fix some critical points inbetween

Breakdown: join smoothly

Keyframe Interpolation

Keyframe

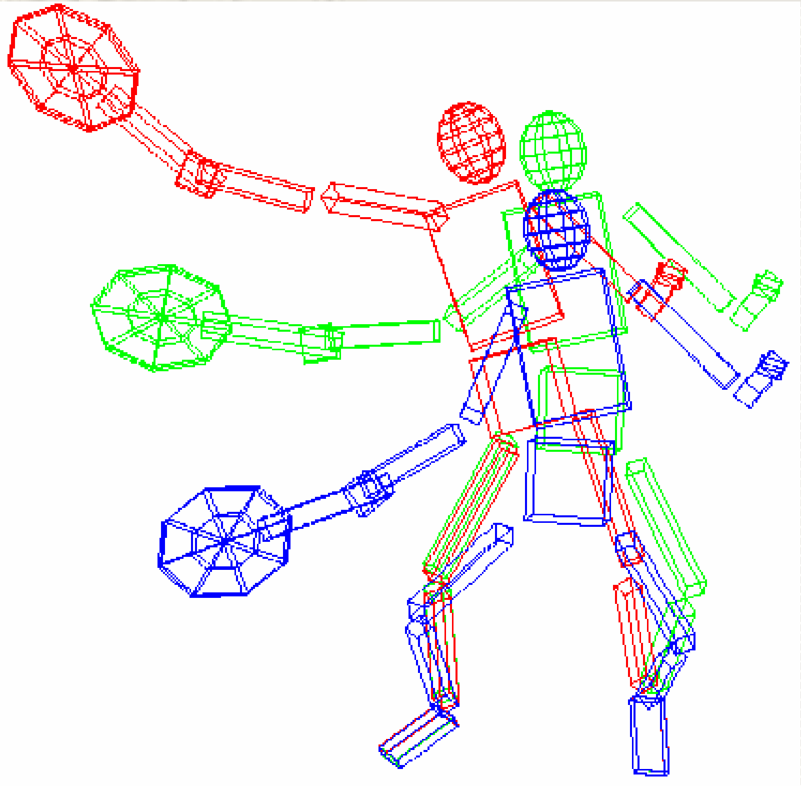
- **Keyframe:**
  define motion from a set of poses



- **Key and inbetween:**
  - Define pose
  - Define keyframe,
  - Compute inbetween for a smooth animation(spline).

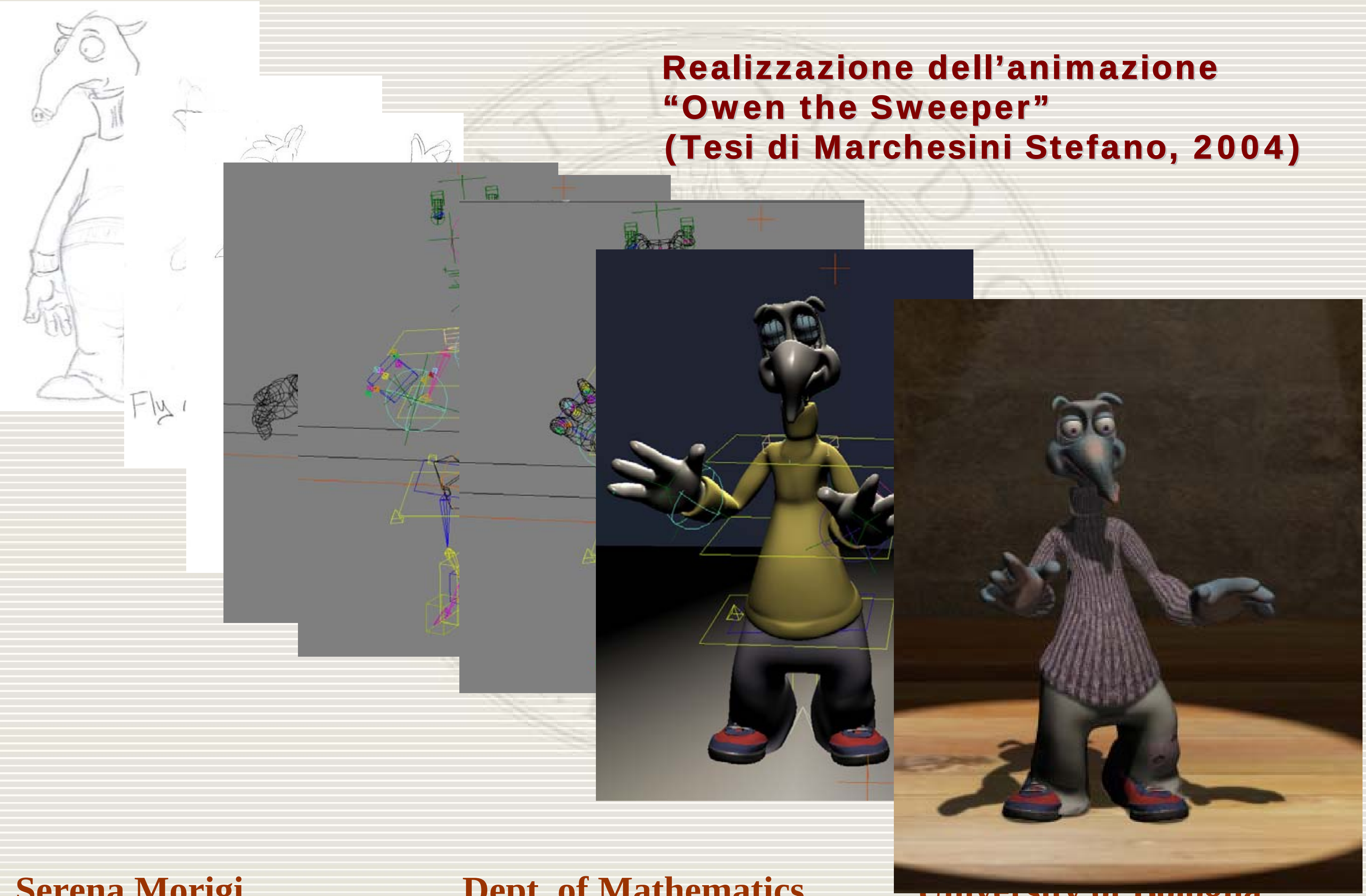
Parametric Interpolation

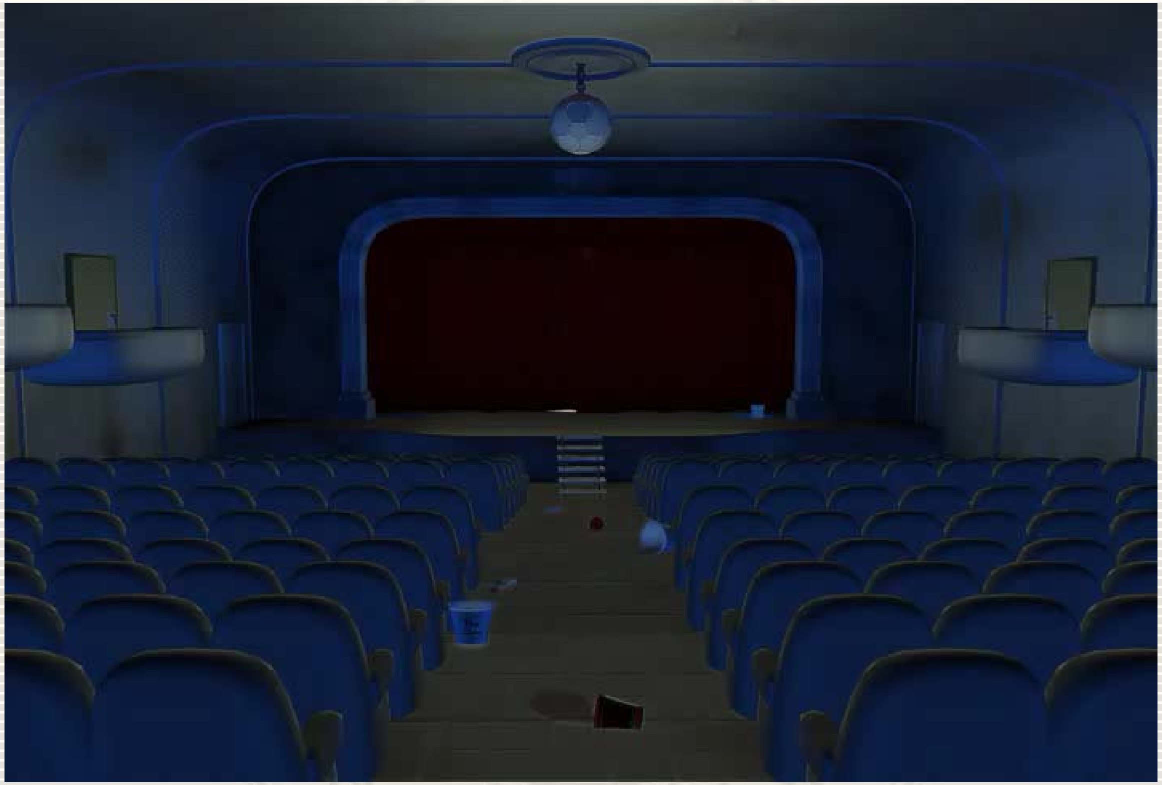
- Any number of parameters:
  - Position, orientation, material (color/texture), light, shape (for nonrigid object)
- Parameter source
  - User;
  - Measured (mocap)
  - procedural



Computer Animation

Realizzazione dell'animazione "Owen the Sweeper" (Tesi di Marchesini Stefano, 2004)



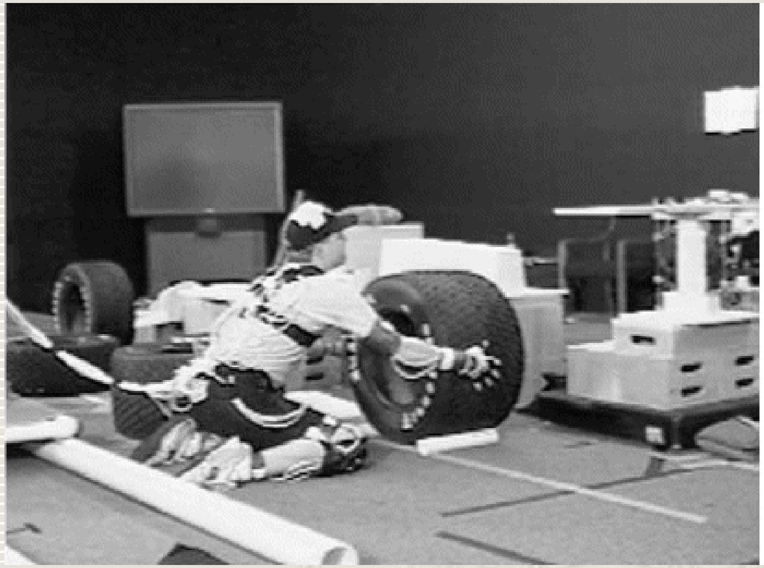


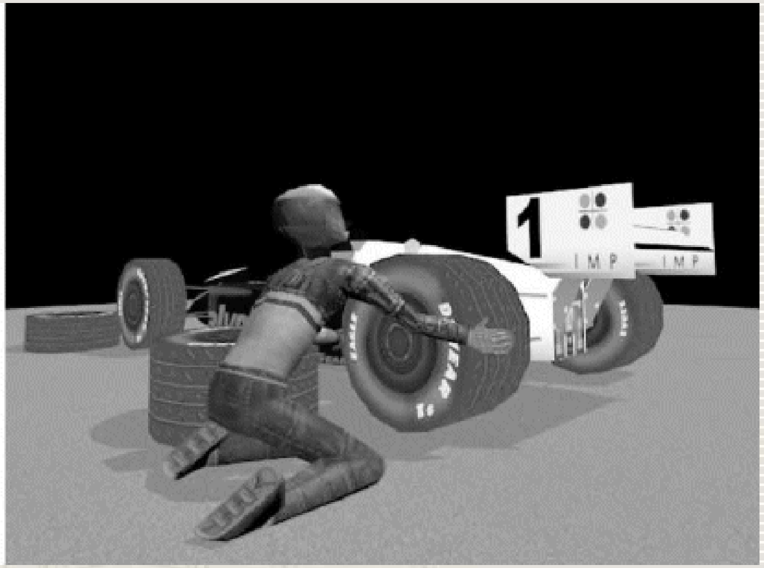
Realizzazione dell’animazione
“Owen the Sweeper”
(Tesi di Marchesini Stefano, 2004)

Motion Capture

Andy Serkis in "Gollum"
"The lord of the rings"

- Captur of motion of (human) actor:
  - Whole body
  - Upper body
  - face
- One way of using a physical device to control animation





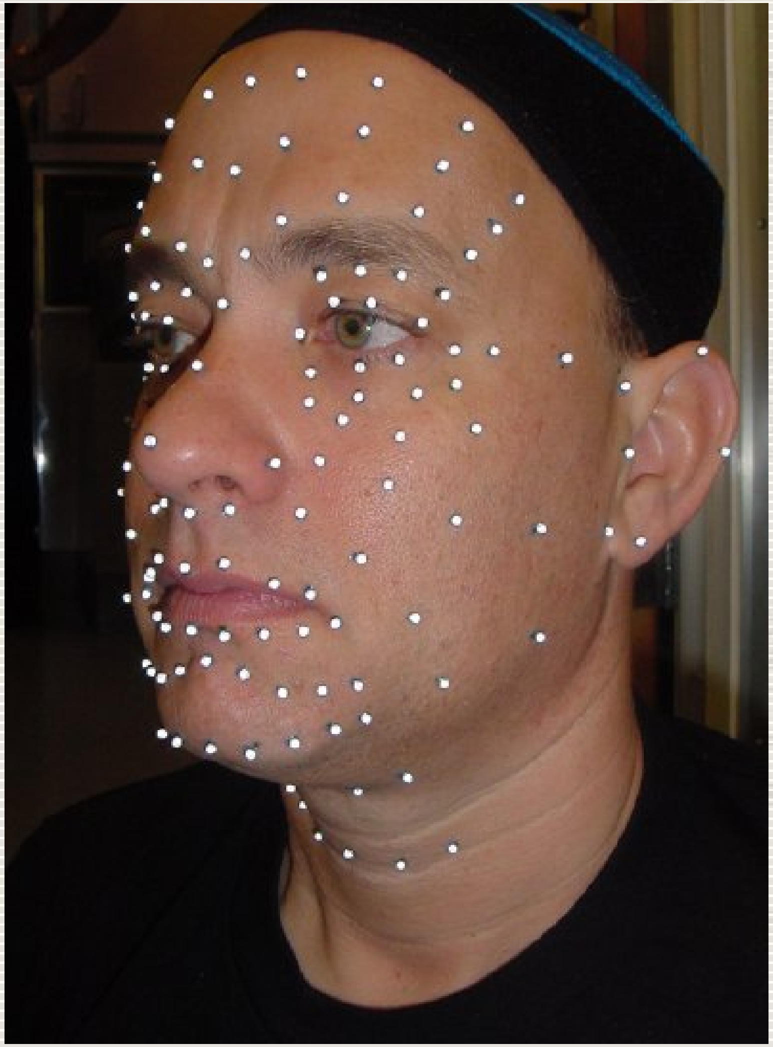
Technologies:

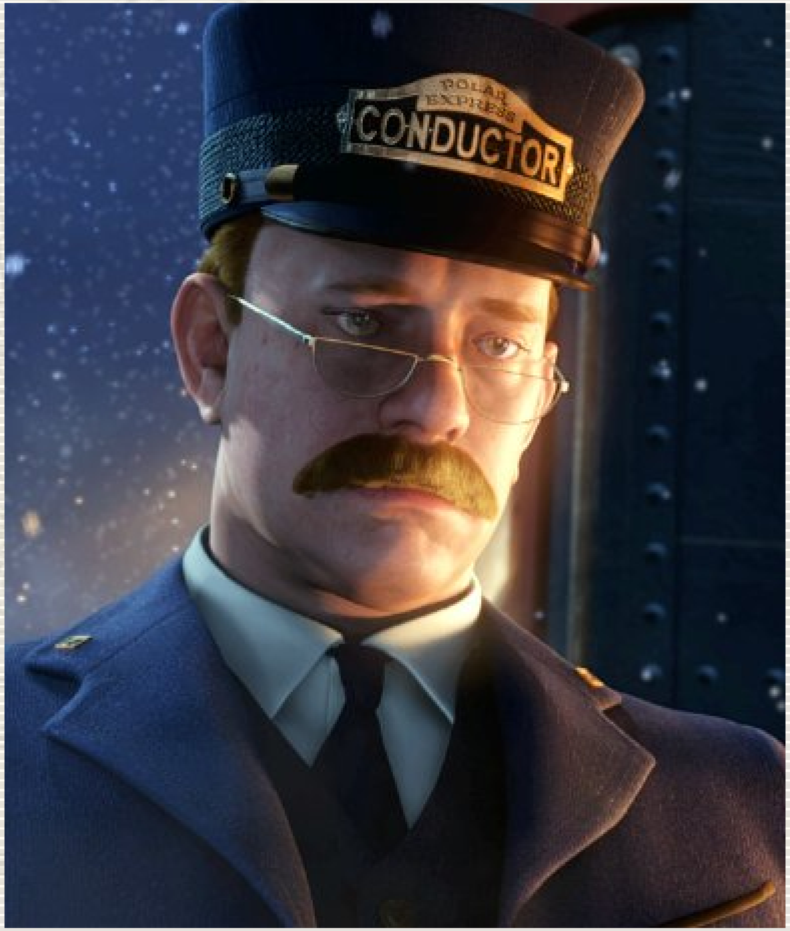
-Magnetic

-Optical passive reflection

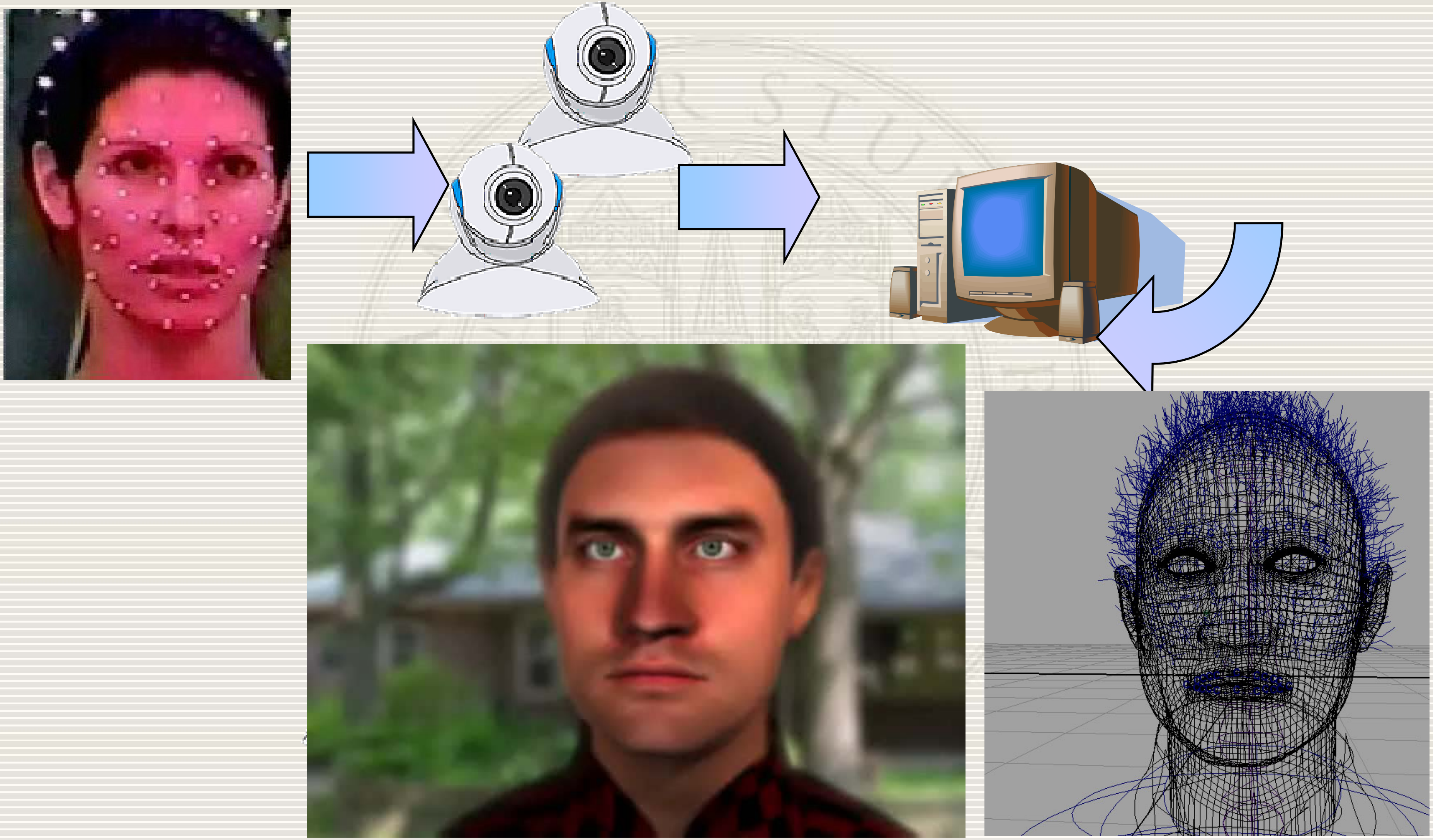
MoCap Motion Capture

“The polar Express” Imageworks, 2005





Motion Capture



Stefano Marchesini Univ. Bologna

Physics-based Animation

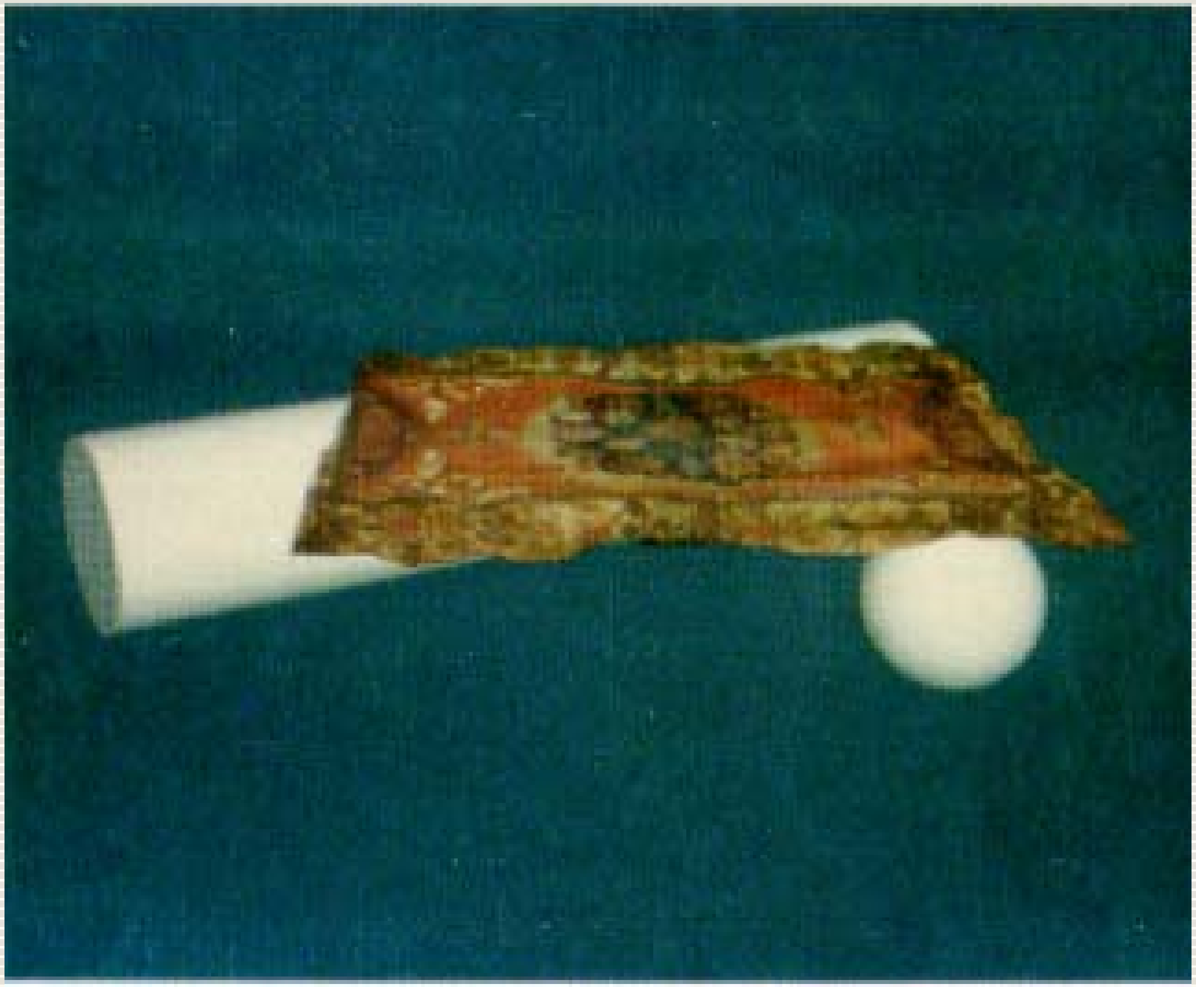
Ideally suited for:

- Large volumes of objects – wind effects, liquids, …
- Cloth animation/draping

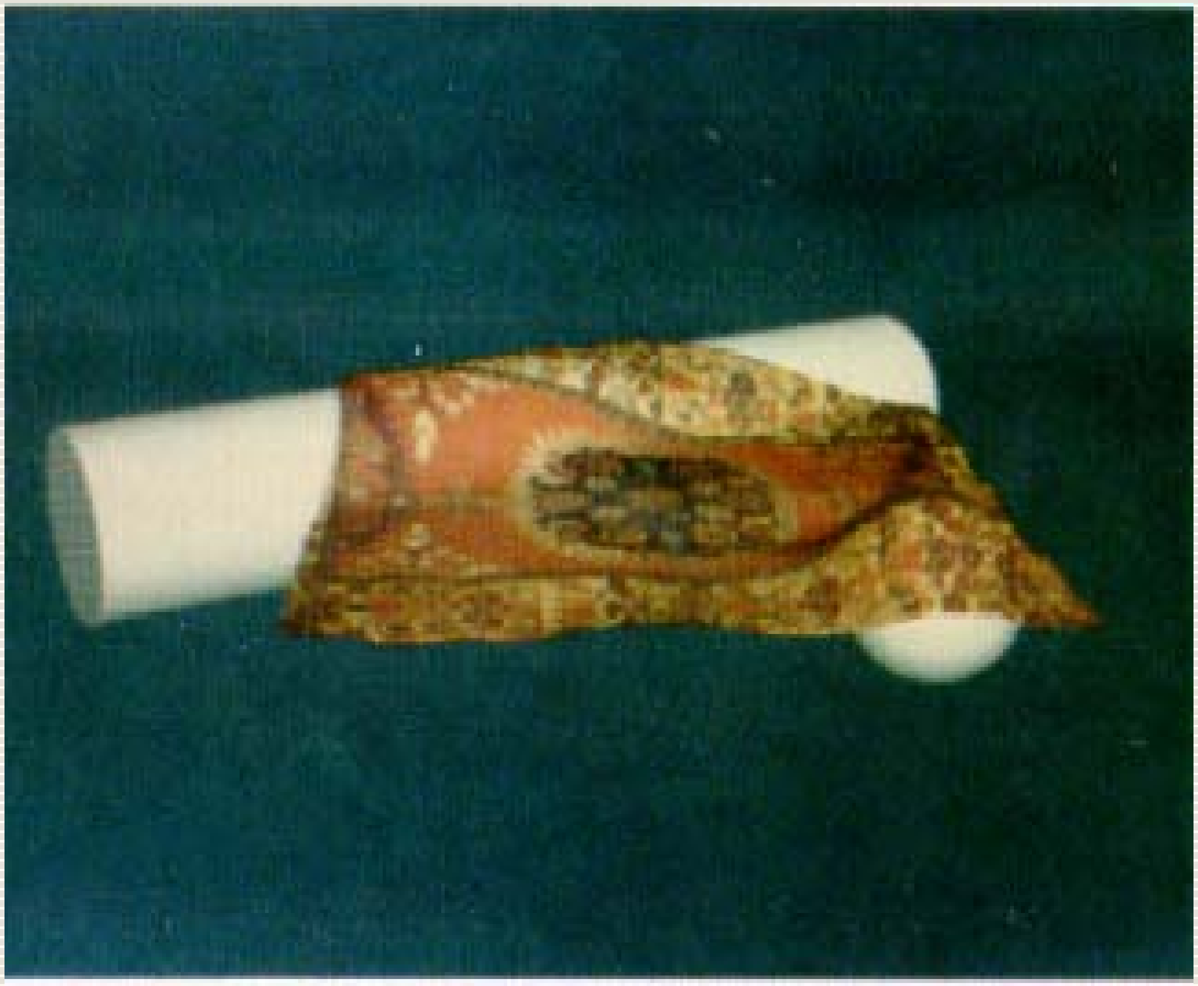
Underlying mechanisms are usually:

- Particle systems
- Mass-spring systems

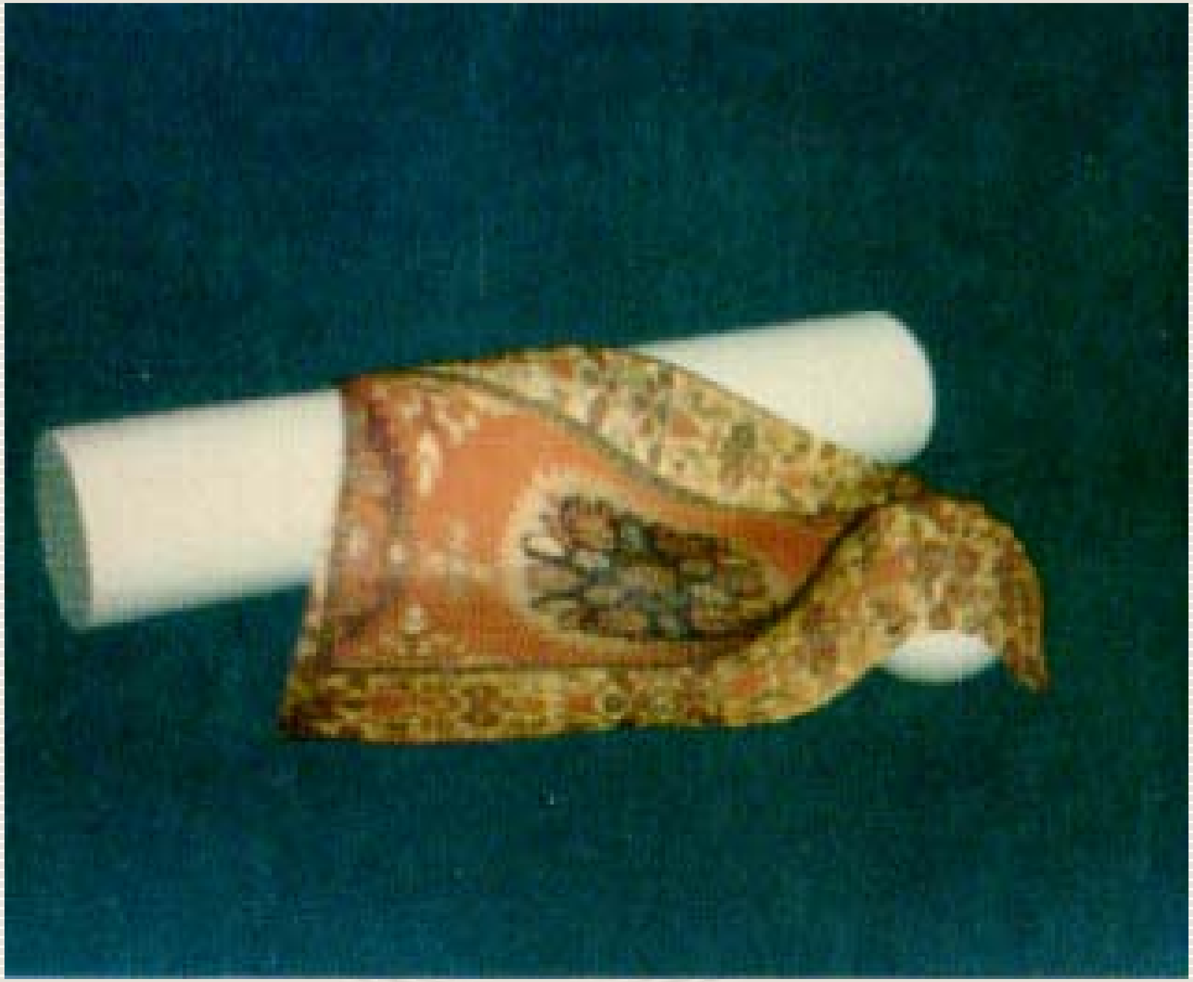
Typically solve ordinary or partial differential equations using iterative methods with some initial/ending boundary values and constraints on conservation of mass/energy/angular momentum



(Terzopoulos, Platt, Barr and Fleischer, SIGGRAPH '87)

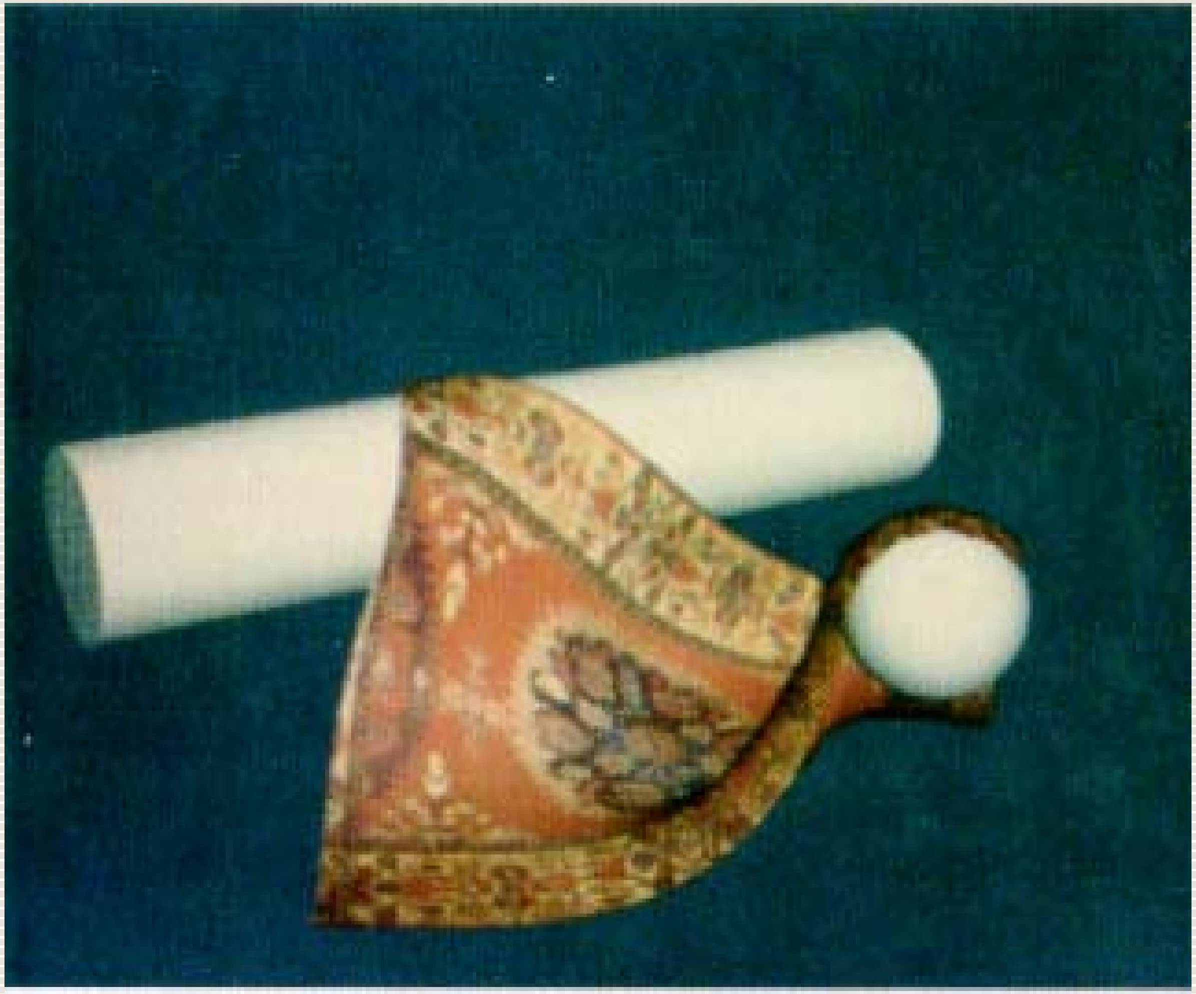


(Terzopoulos, Platt, Barr and Fleischer, SIGGRAPH ’87)



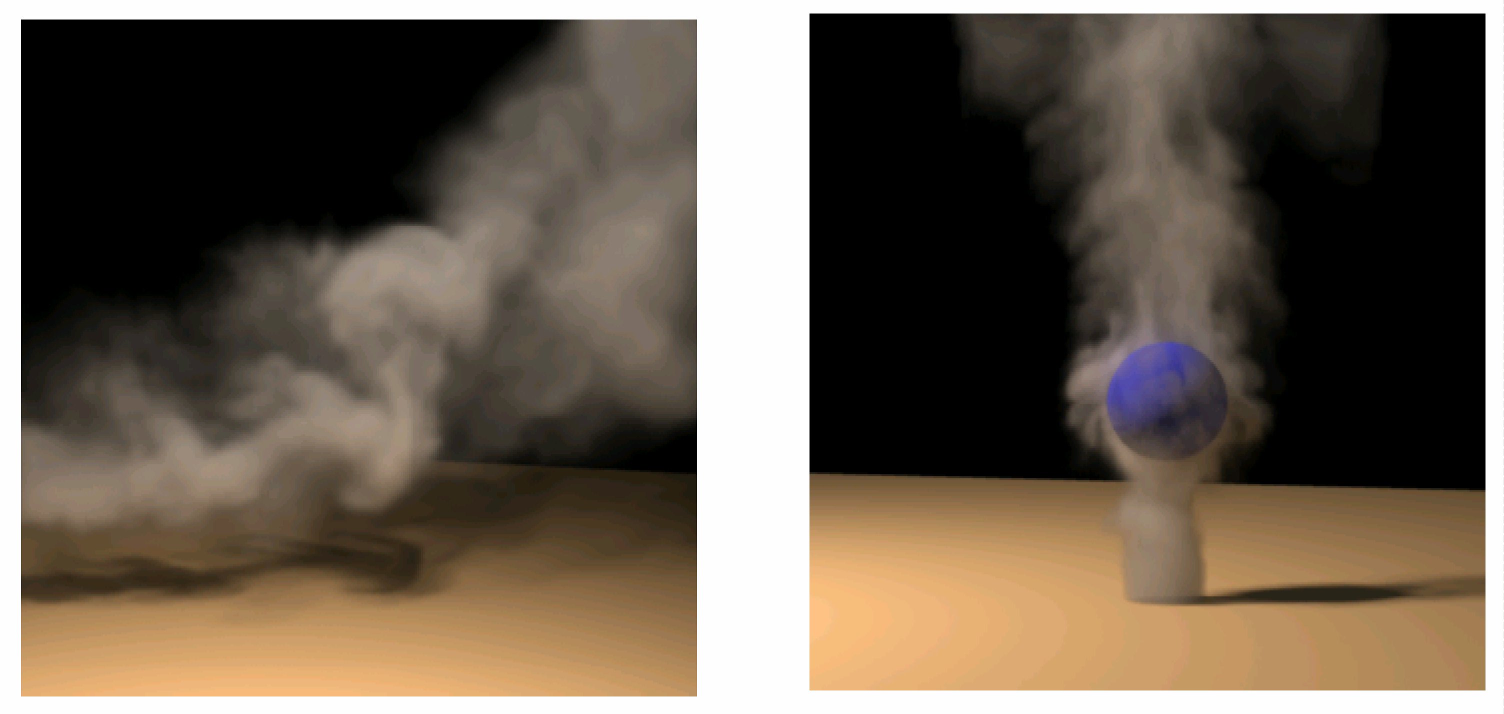
(Terzopoulos, Platt, Barr and Fleischer, SIGGRAPH ’87)

(Terzopoulos, Platt, Barr and Fleischer, SIGGRAPH ’87)



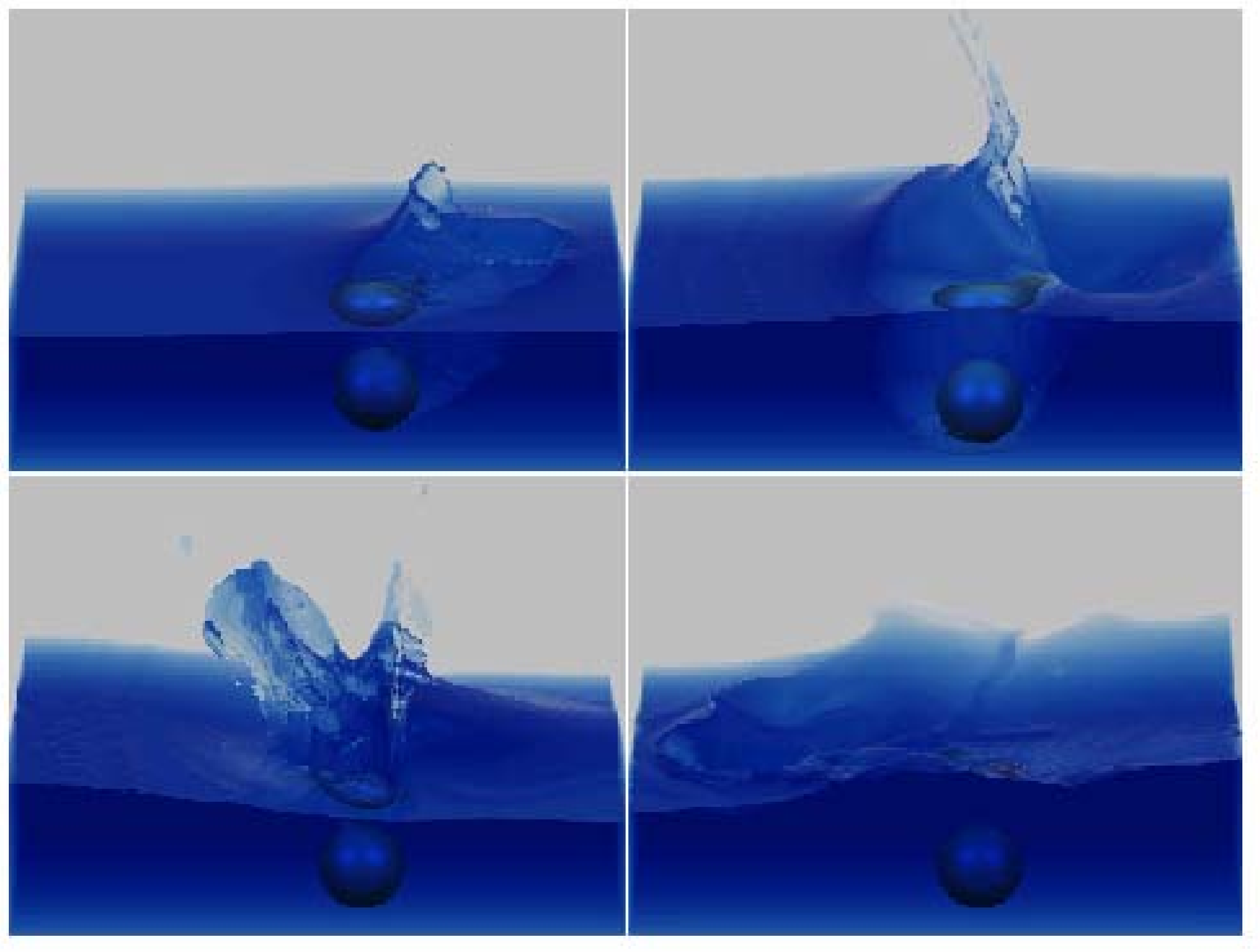
(Terzopoulos, Platt, Barr and Fleischer, SIGGRAPH '87)

Examples



Images from Fedkiw, Stam, Jensen, SIGGRAPH 2001

Examples



Images from Foster & Fedkiw

SIGGRAPH 2001

Physically real motion

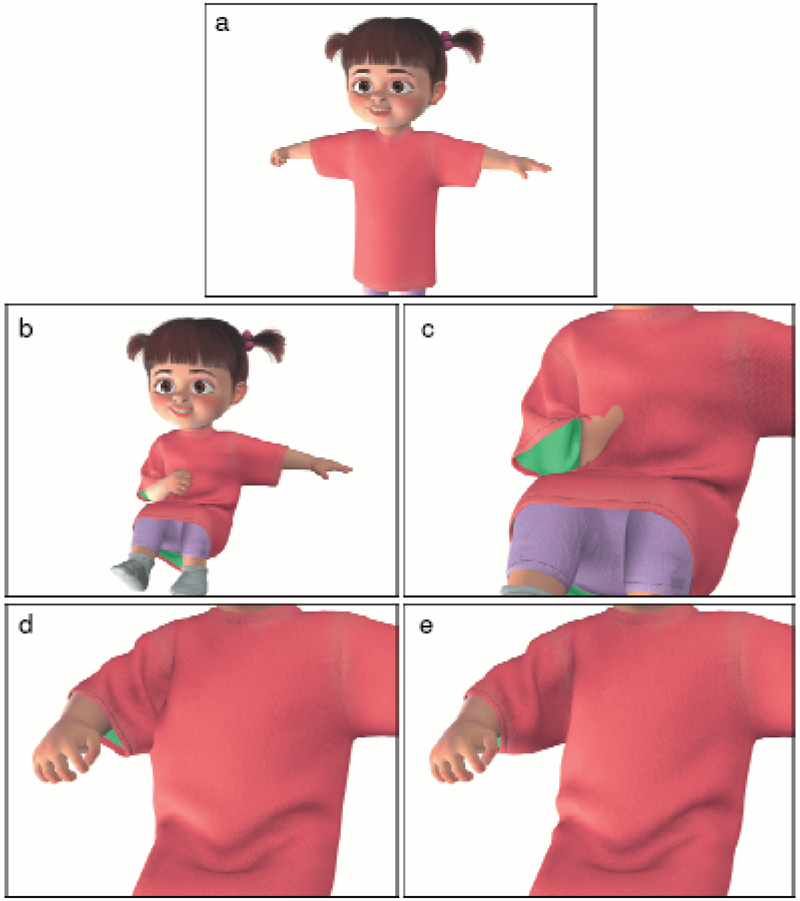


Figure 10: (a) Starting pose. (b) Arm moves in tightly. (c) Close-up view of (b) with right arm invisible. Note how the arm position forces cloth to intersect both itself and the body. (d) Without GIA, a cloth/cloth intersection persists as the arm pulls out, snagging the sleeve. (e) The same frame as (d), but using GIA, the cloth doesn't snag as the arm pulls out.

Graphics Hardware

Real-time graphics

- CPU: general-purpose computer ('60)
- VGA (Video Graphic Array) controller (DPU) (anni '80): special-purpose graphics system
- Graphics Hardware Unit ('90): pipeline graphics system (special-purpose VLSI circuits) SGI and Evans Sutherland design expensive multichip.

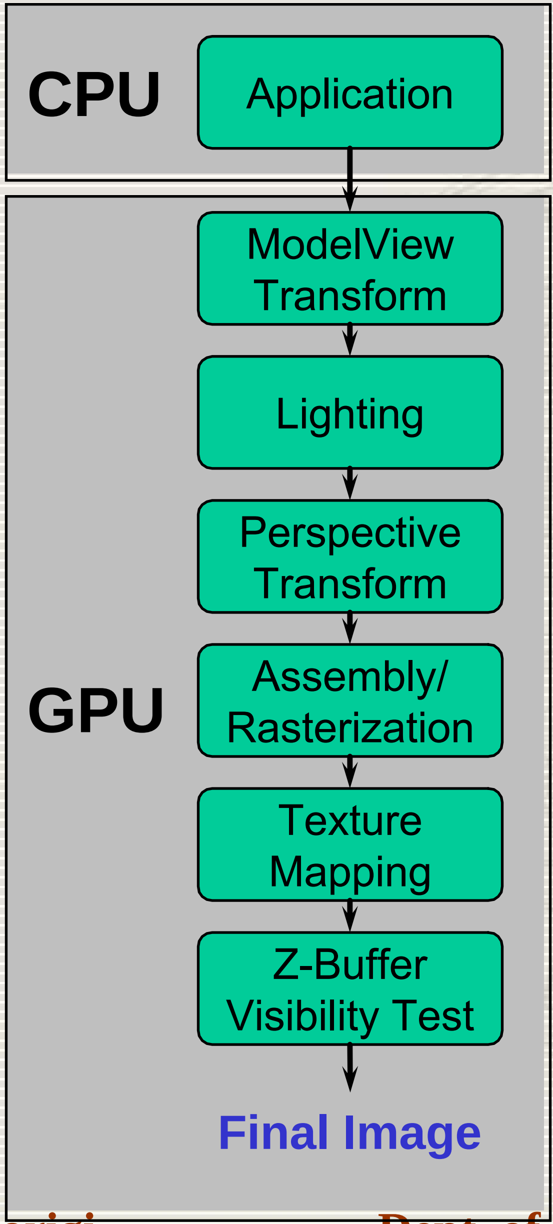
- GPU (Graphics Processor Unit) (end'90): single chip GPU, cheaper, in PC console for video game

- Towards the offline rendering system.

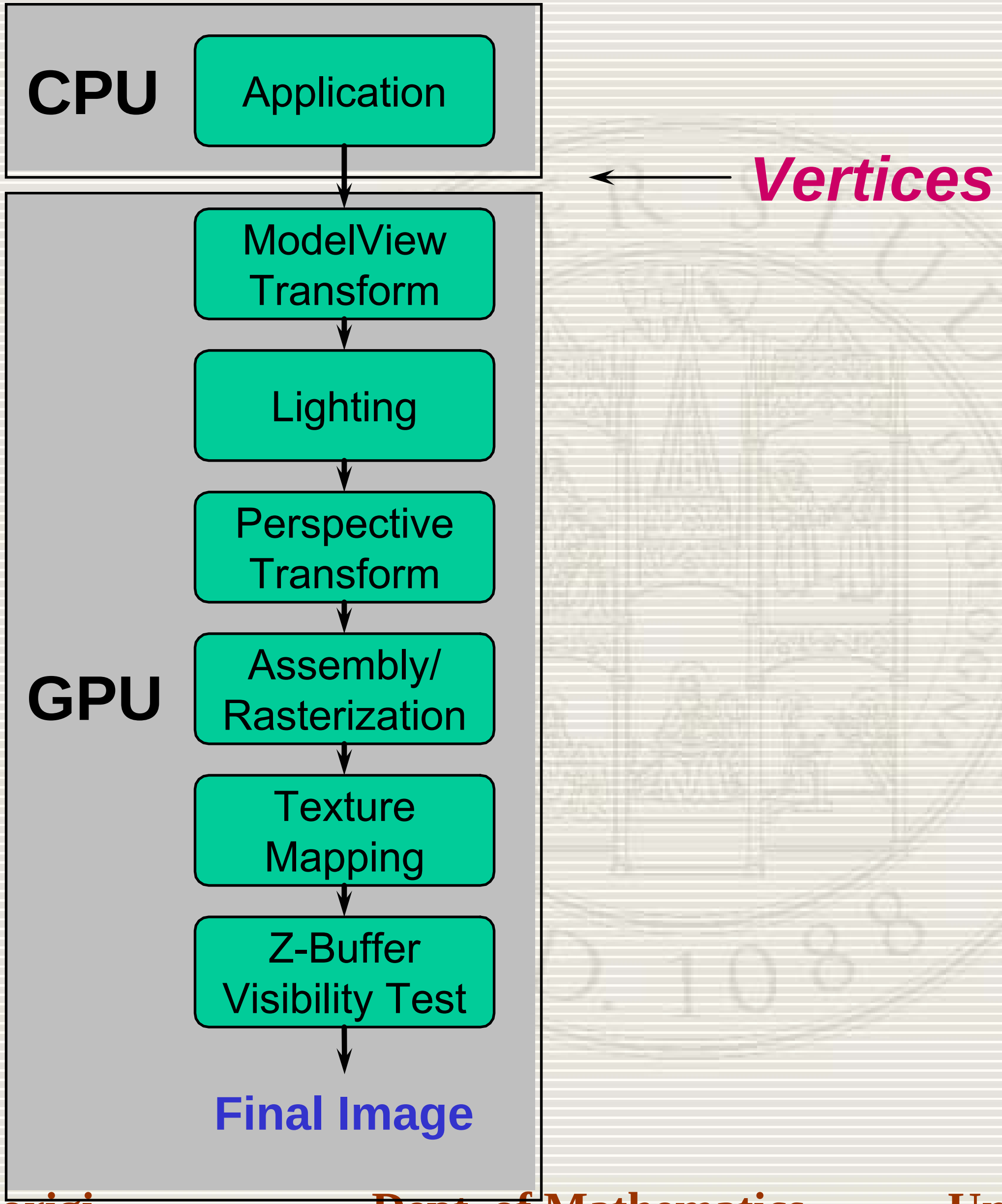
Graphic card generations

- I generation (fino al 1998) single chip GPU. TNT (NVIDIA), RAGE (ATI), Voodoo3 (3dfx), gestione pixel GPU, trasformazioni dei vertici in CPU, set operazioni matematiche su pixel limitato.
- II generation (1999-2000) GeForce2 (NVIDIA), Radeon 7500 (ATI), Savage3D (S3), transform and lighting (T&L) of vertices is done in hardware as well (uses the fixed function pipeline)
- III generation (2001) GeForce3,GeForce4 (NVIDIA), Radeon 8500 (ATI), vertex programmability, graphics card lets programmers download assembly programs to control vertex lighting and shading keeping the speed of the fixed function pipeline with none of the restrictions No pixel programmability.
- IV generation (2002...) GeForce FX family (NVIDIA), Radeon 9700 (ATI),Quadro4 XGL (NVIDIA), nVidia released Cg. Vertex and pixel programmability. Increased use of lighting effects such as bump mapping and shadowing.
- V generation (now) GPGPU General-Purpose GPU: 32 bit floating point throughout the pipeline, GeForce 6+, console videogames, XBOX360

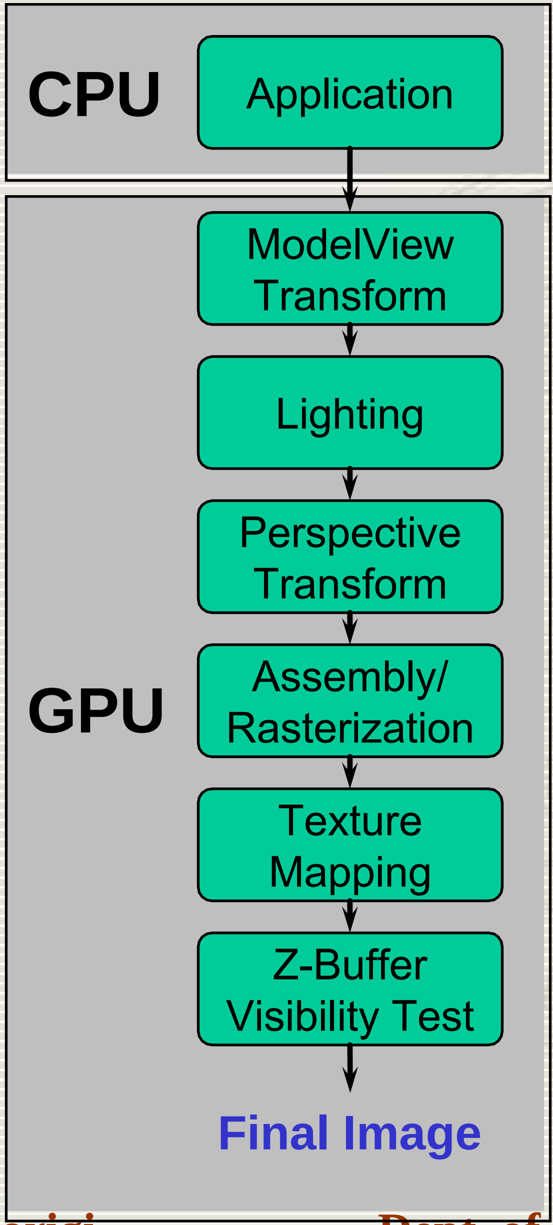
Rendering Pipeline



Rendering Pipeline

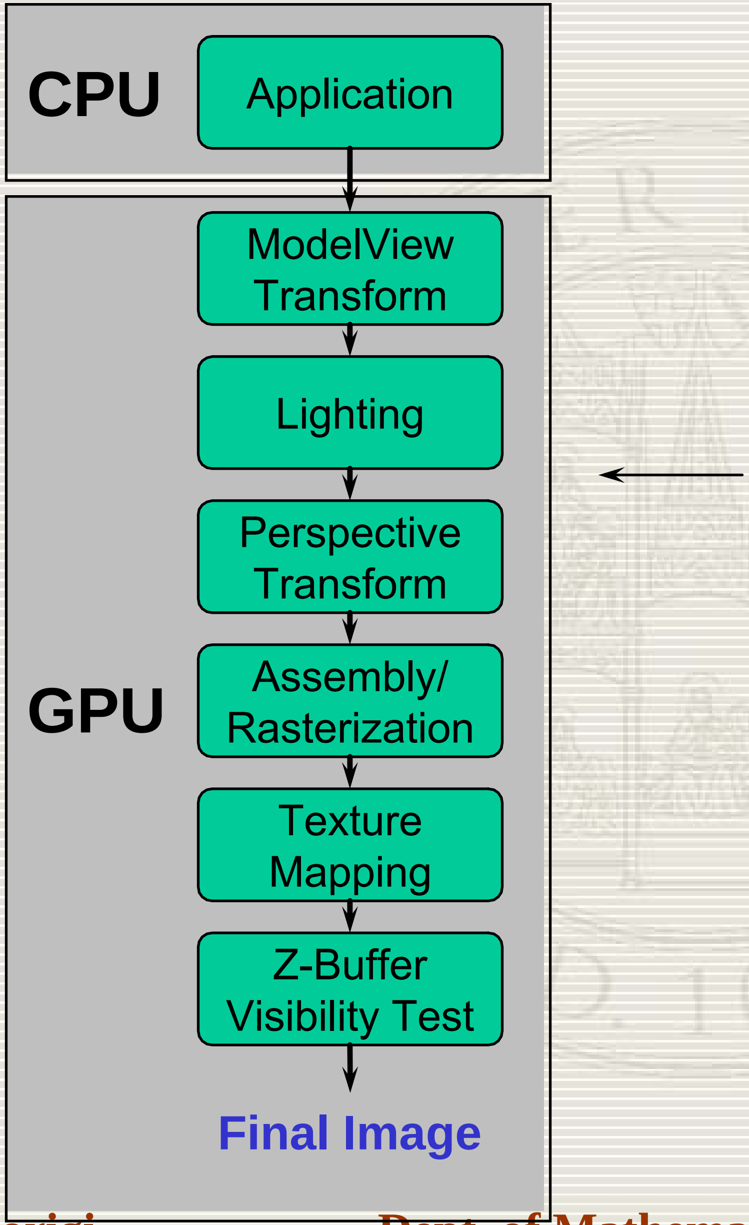


Rendering Pipeline



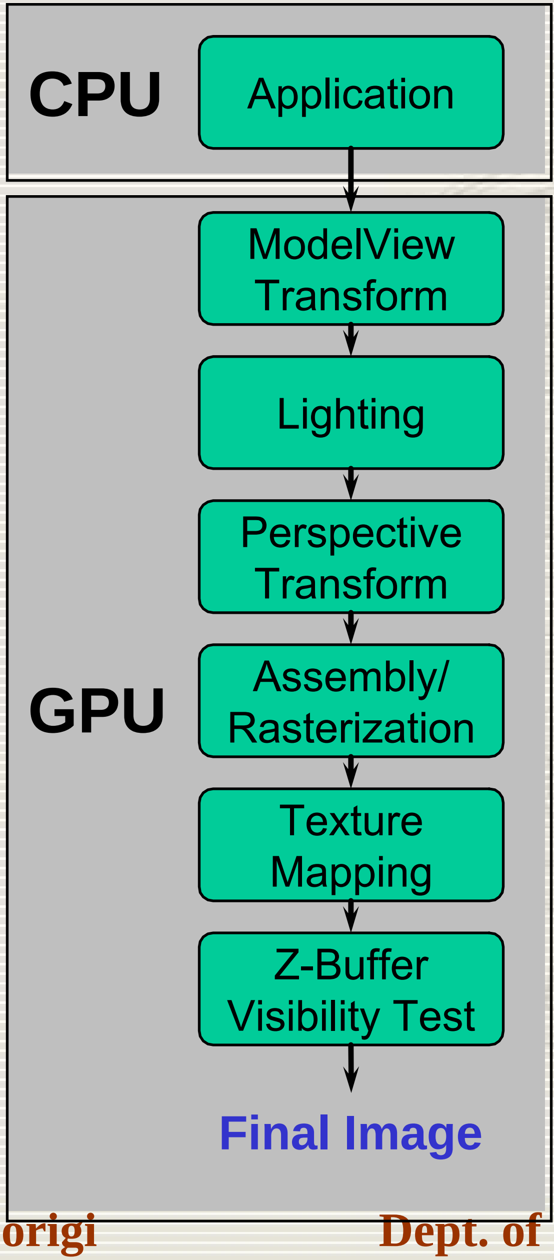
World Space Vertices

Rendering Pipeline



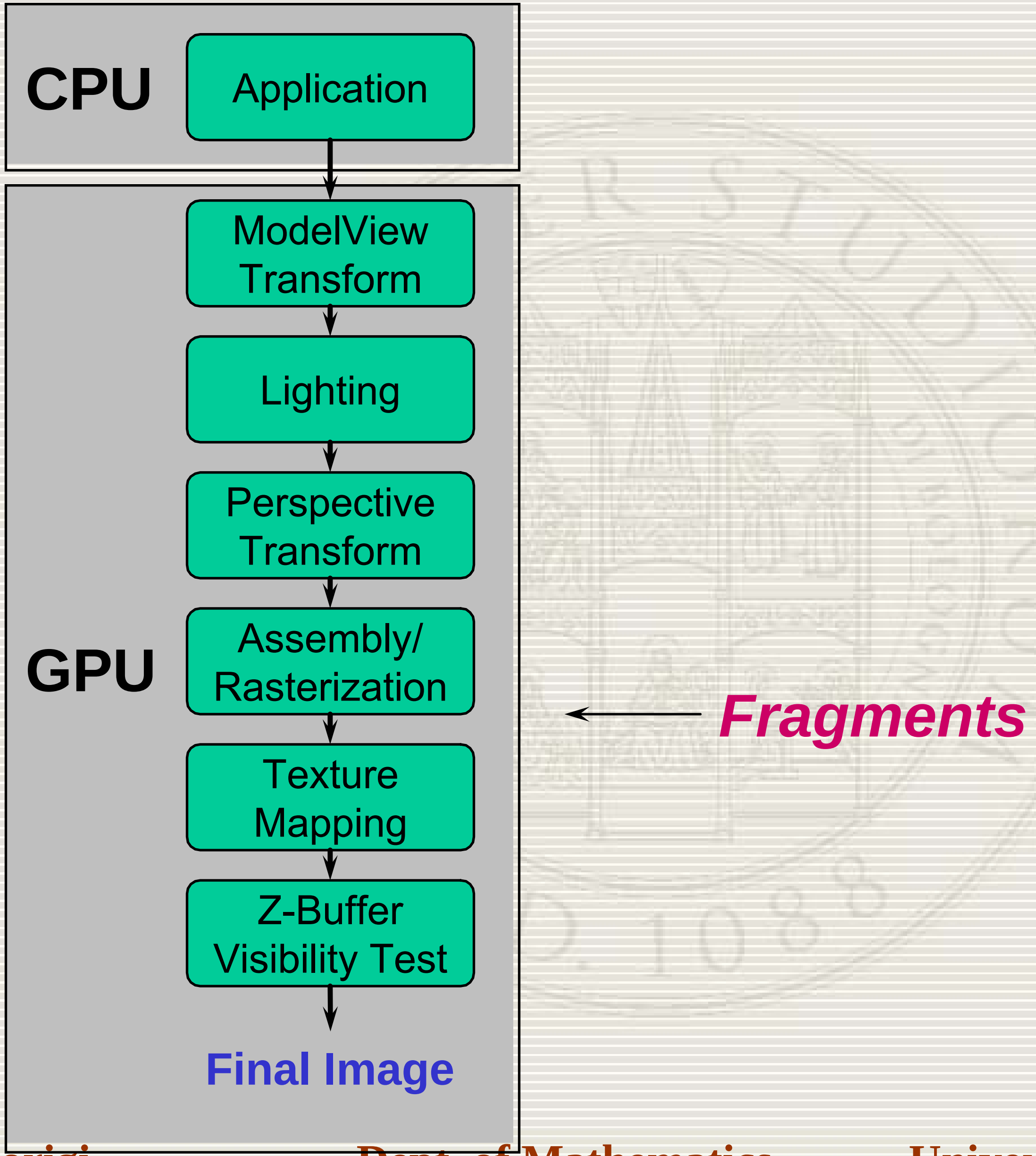
Lit, World Space Vertices

Rendering Pipeline

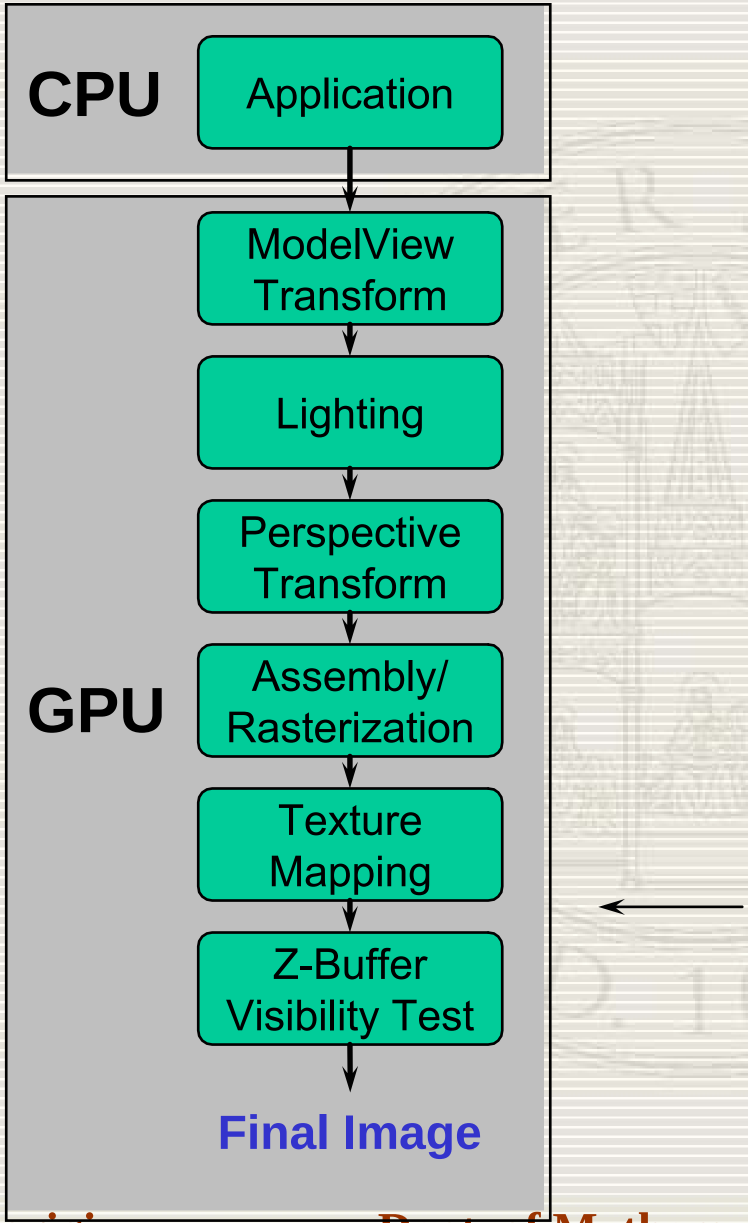


Lit, Screen Space Vertices

Rendering Pipeline

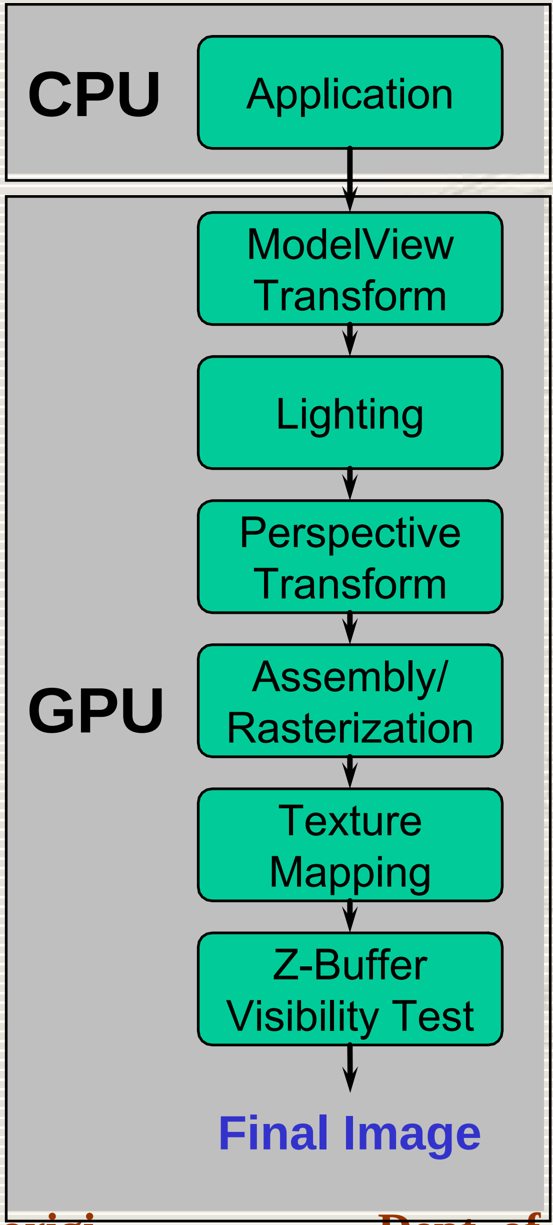


Rendering Pipeline



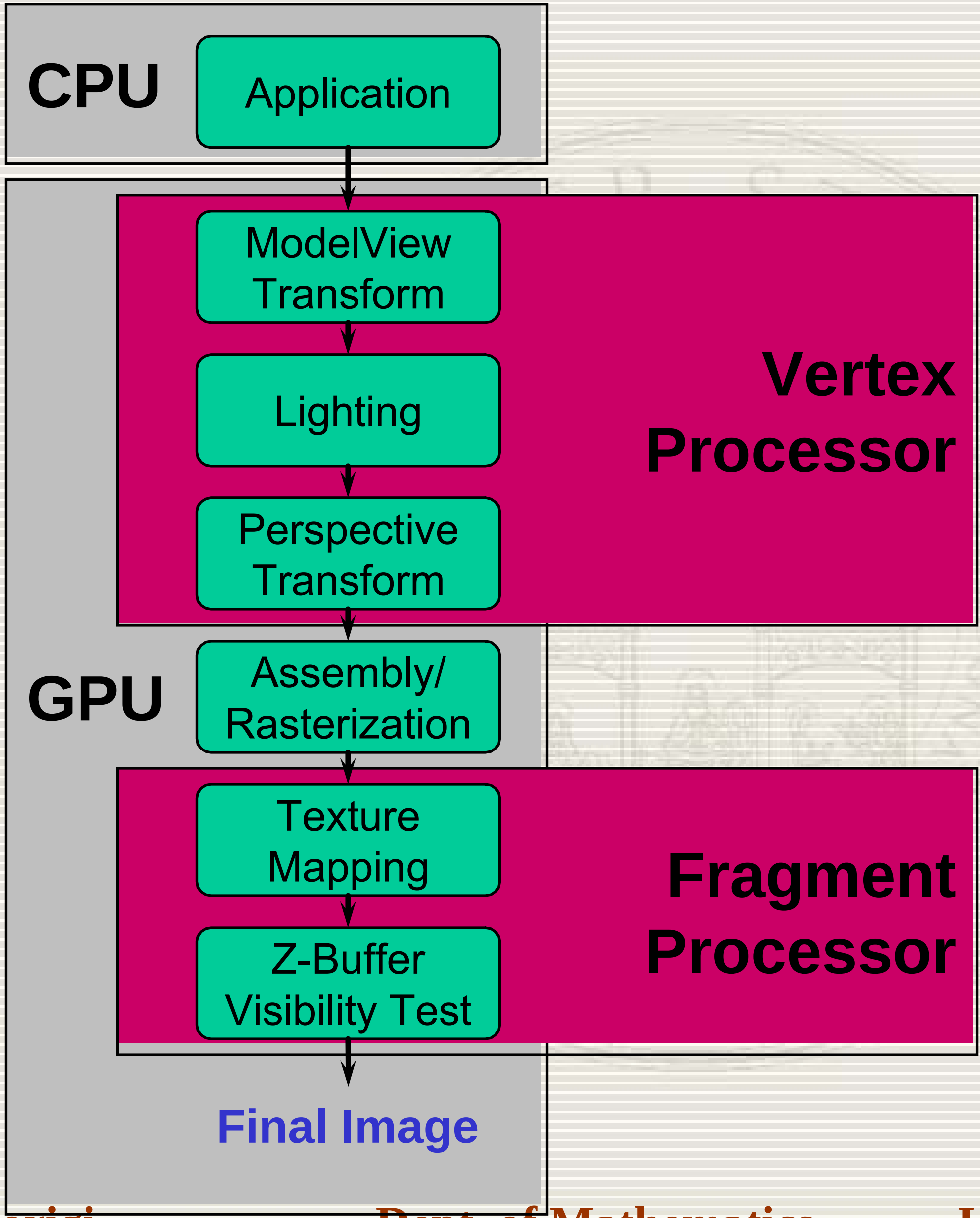
Textured Fragments

Rendering Pipeline



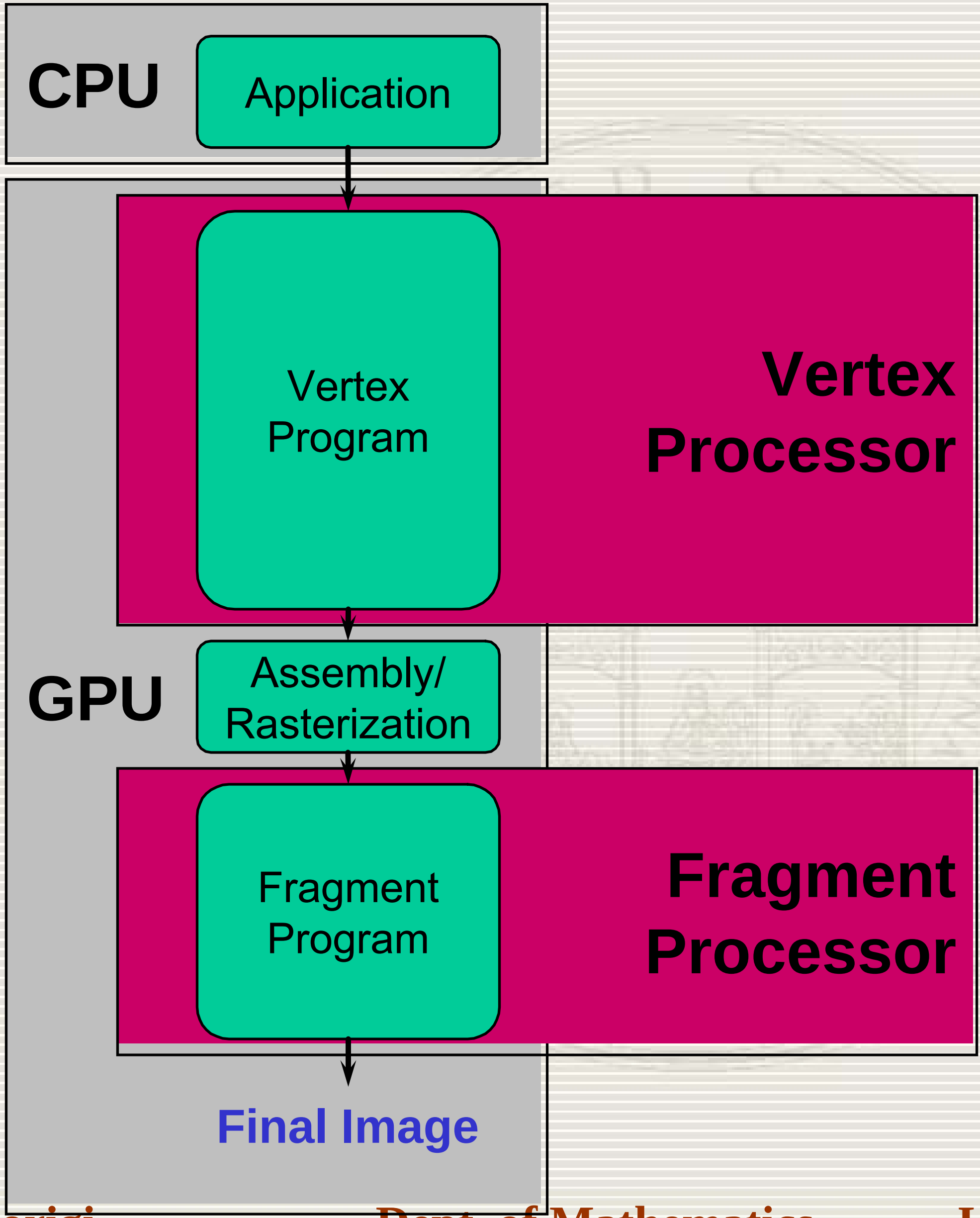
Textured, Visible Fragments

"Fixed Function" Pipeline



the programmer had limited control over how the hardware created the final image. To do non-standard effects, like cartoon shading, required a lot of hackery.

Programmable Pipeline



The programmer simply sends **data** to the card and then can write a program to interpret the data and create an image.

Programmable Pipeline

- **Vertex shader programs** take as input per vertex information (object space position, object space normal, etc.) and per frame constants (perspective matrix, modeling matrix, light position, etc.). They produce some of the following outputs: clip space position, diffuse color, specular color, transparency, texture coordinates, and fog coordinates.

- **Pixel shader programs** take as input the outputs from the vertex shader program and texture maps. They produce a final color and transparency as output. They are often called fragment shaders.
  - Per-Pixel Lighting
  - Environment Mapping, Bump Mapping
  - NPR (Non Photorealistic Rendering)

Programmable Pipeline

- Programming Languages
  - Cg from NVIDIA: Cg is a C-like language that the graphics card compiles in to a program
    The program is run once per-vertex **and/or** per-pixel *on the graphics card*
  - RenderMonkey from ATI
  - HLSL from Microsoft
  - GLSL from OpenGL ARB

GPGPU (General Purpose GPU)

- GPU can be used for things other than graphics!
- Instead of pixels with color, think of a grid with a four-component vector at each cell.
- Instead of frames, think of time-steps.
- Instead of rendering equations, perform any computation you want.

- Why use the GPU? GPU is more specialized than CPU, so can do what it does fast; Parallel, pipelined architecture

FUTURE ….

- GPU as a co-processor for general purpose computation
- Expect shader programs to become even more flexible and powerful
- Programmable ray processing unit!