CS100433
Computer Graphics
计算机图形学

Junqiao Zhao 赵君峤

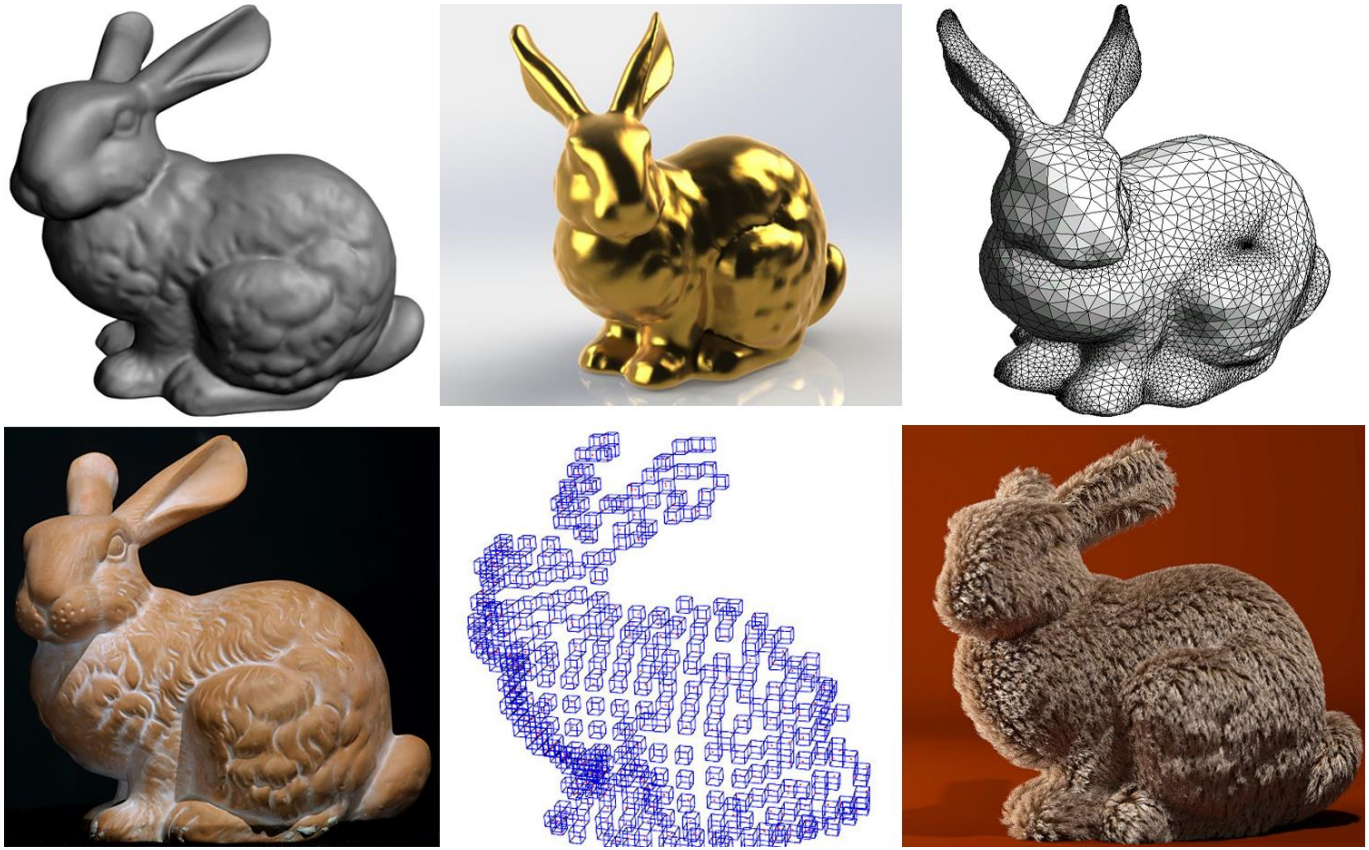
Department of Computer Science and Technology

College of Electronics and Information Engineering

Tongji University

Welcome to CS100433 !

- Computer Graphics



Overview

- 2 credits elective course for undergraduate level
- CG course team
  - **Dr. Junqiao ZHAO (赵君峤)**
  - Prof. Xiaoping WANG (王小平)
- Course website
  - cs1.tongji.edu.cn/courses/CS100433/

My Contact Info.

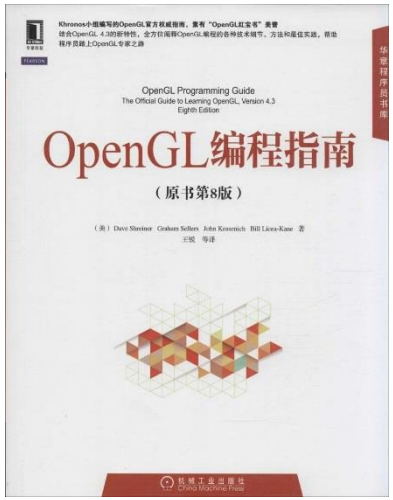
- Email:
  - zhaojunqiao@tongji.edu.cn
- Office:
  - 232 Room SEE Building
- Piazza
  - piazza.com/tongji.edu.cn/fall2019/cs100433

Goal of This Course

- Principles of Computer Graphics
  - Math
  - Algorithms
- Modern OpenGL
  - Version 3.3 + (*date back to 2010*)
- Will not cover:
  - How to use CG software to create CG contents.
  - Advanced graphics techniques, such as *deferred lighting*
  - Animation techniques

Textbook

- Donald Hearn, M. Pauline Baker, 2012, Computer Graphics with OpenGL Fourth Edition, Prentice Hall. 《国外计算机科学教材系列:计算机图形学(第4版)(英文版)》, 电子工业出版社, 第1版 (2012年2月1日)。
- Dave Shreiner, Graham Sellers, John Kessenich, Bill Licea-Kane, 2013, OpenGL Programming Guide:the Official Guide to Learning OpenGL, Version 4.3, Eighth Edition, Addison-Wesley Professional. 《华章程序员书库:OpenGL编程指南(原书第8版)》, 机械工业出版社; 第1版 (2014年10月1日)。



Prerequisites

- Programming skills in C and C++
- Data structures
  - Arrays
  - Linked list
  - Tree/Graph
- Linear Algebra
  - Vector
  - Matrix
- Geometry
  - Point, line, surface, polyhedron

Course Grading

- Assignments — 10%
  - Four assignments
  - Questions and Programs
- Group project — 30%
  - Work in a group of maximum 7 students
  - Three phases (*start on the 9th week*)
    - Proposal
    - MidTerm
    - Final Defense
- Exam — 60%

References

- Please see the website:
  - *cs1.tongji.edu.cn/courses/CS100433*
- ***Be self-motivated!***

- Questions?
- **We need a course assistant.**

- *What do you think is Computer Graphics?*

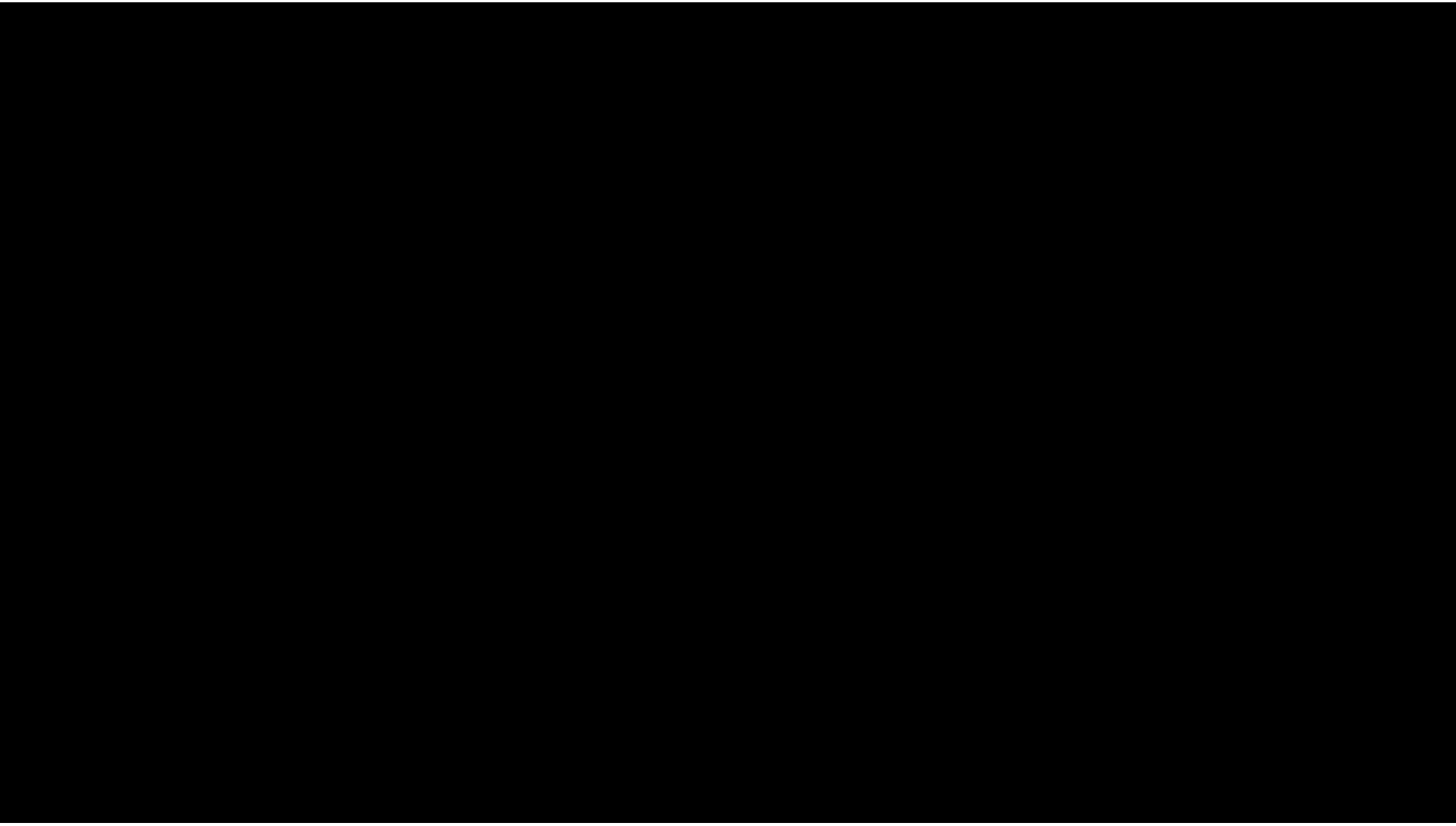
What is Computer Graphics?

- A sub-field of computer science which studies methods for digitally synthesizing and manipulating visual content – *Wikipedia*
- The study of creating, manipulating, and using visual images in the computer – *Steve Marschner*
- *Computer graphics* deals with all aspects of creating images with a computer – *Ed Angel*
  - Hardware
  - Software
  - Applications

What is Computer Graphics?

- **Geometry**: studies ways to represent and process surfaces/primitives
- **Rendering**: studies algorithms to reproduce light transmission
- **Animation**: studies with ways to represent and manipulate motion
- **Imaging**: studies image acquisition or image editing
- **Topology**: studies the behavior of spaces and surfaces.
- **VR/AR**: studies the interaction between virtual world and the reality
- Machine Learning

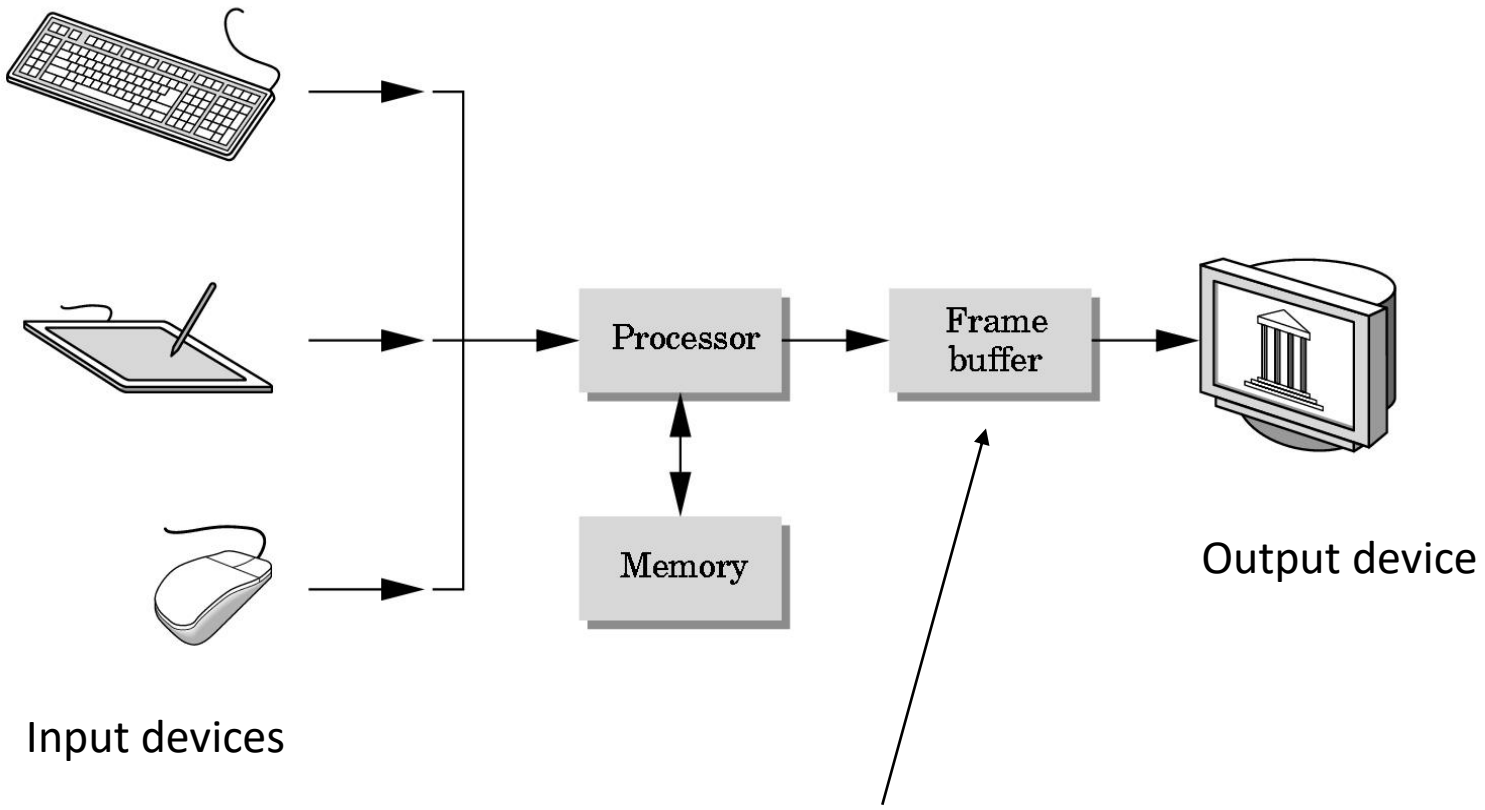
- *Computer Graphics* vs *Computer Vision*
- *Computer Graphics* vs *Image Processing*
- *Computer Graphics vs Virtual Reality*
- *Virtual Reality* vs *Augment Reality*



Microsoft HoloLens

- *How is an image generated on your screen?*

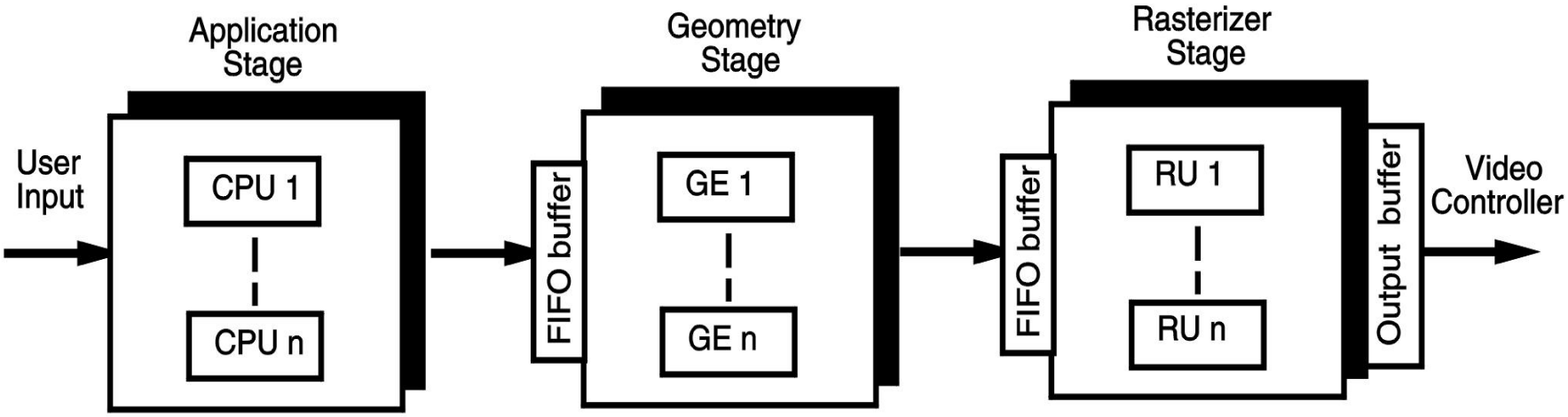
Basic Graphics System



(Angel 2005)

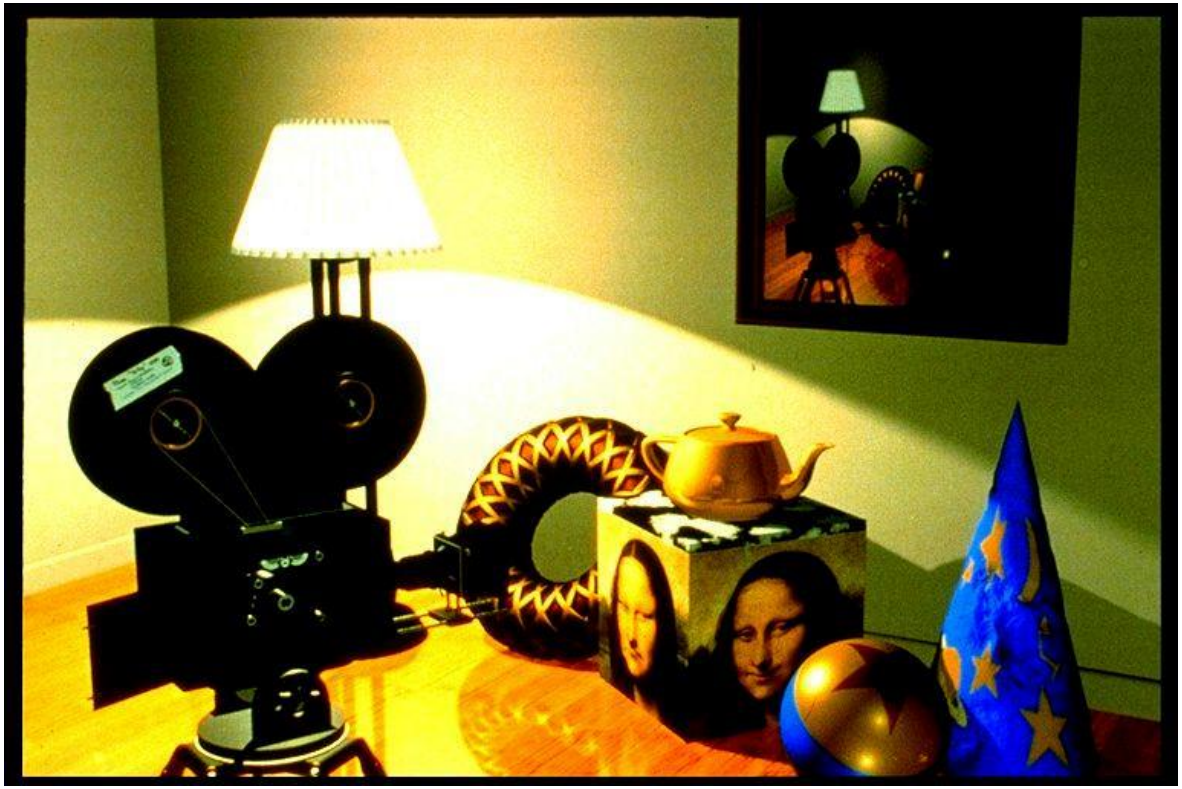
Graphics / Rendering Pipeline

- There are three stages
  - Application Stage
  - Geometry Stage
  - Rasterization Stage



An example thro’ the pipeline…

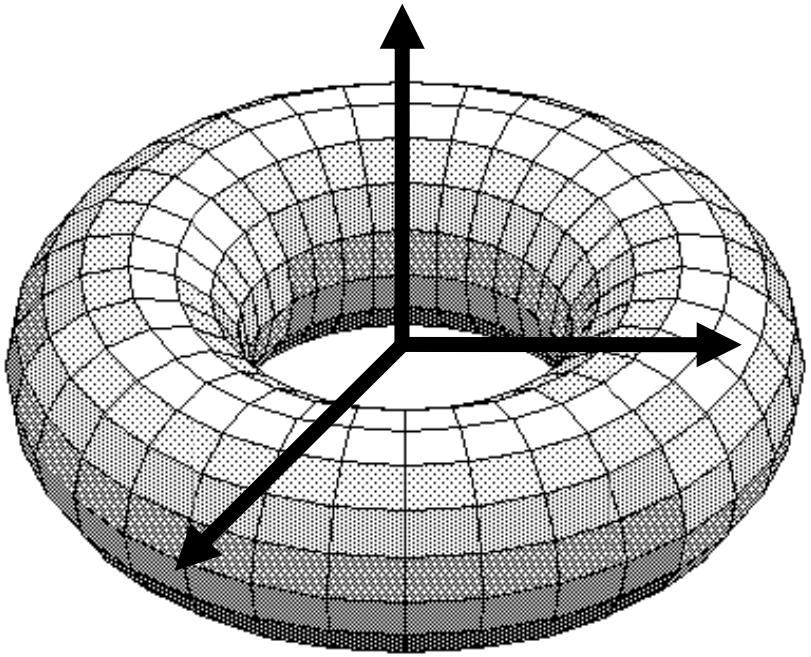
The scene we are trying to represent:

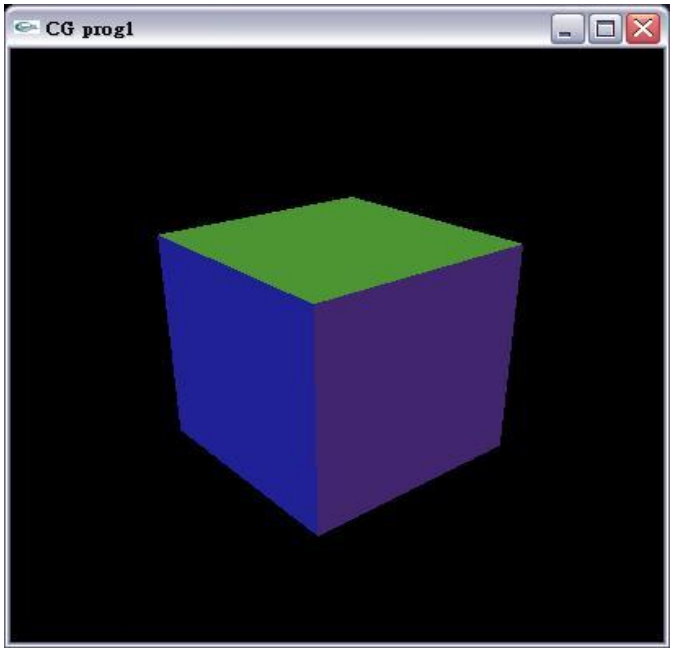


Preparing Shape Models

Designed by polygons, parametric curves/surfaces, implicit surfaces and etc.

Defined in its own coordinate system



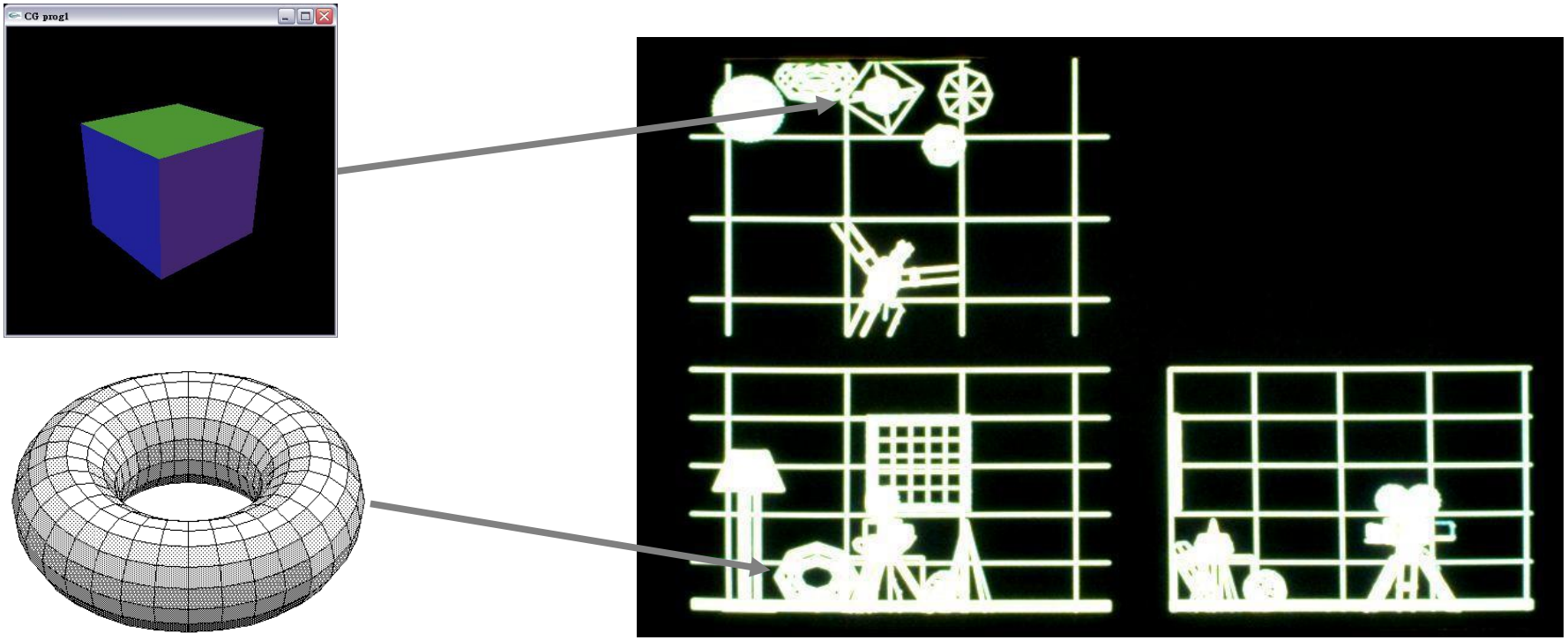


Transformation

Objects put into the scene by applying translation, scaling and rotation

Linear transformation called homogeneous transformation is used

The location of all the vertices are updated by this transformation

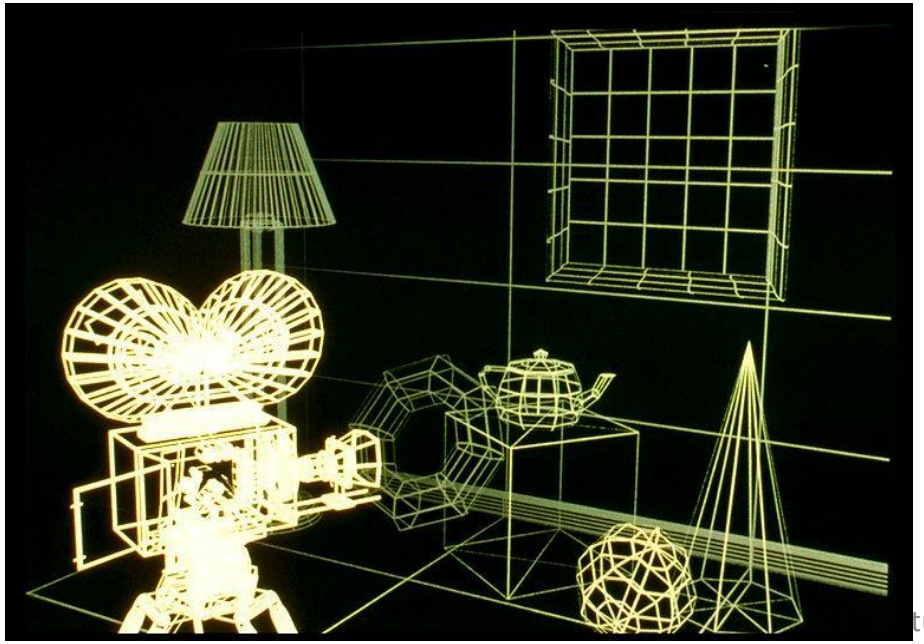


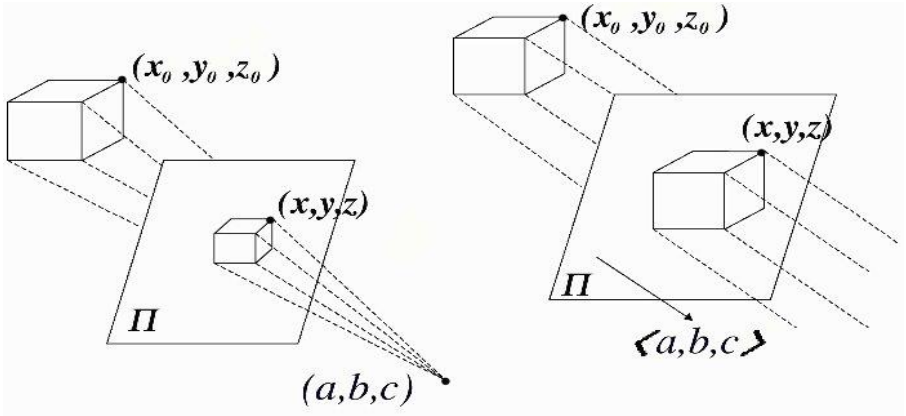
Projection

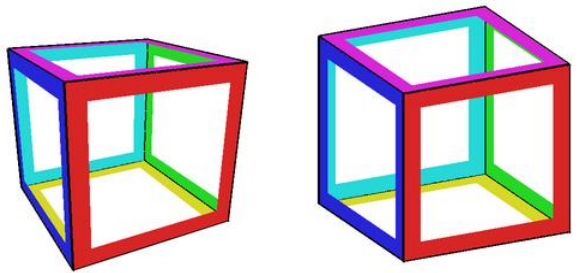
We want to create a picture of the scene viewed from the camera

We apply a perspective transformation to convert the 3D coordinates to 2D coordinates of the screen

Objects far away appear smaller, closer objects appear bigger

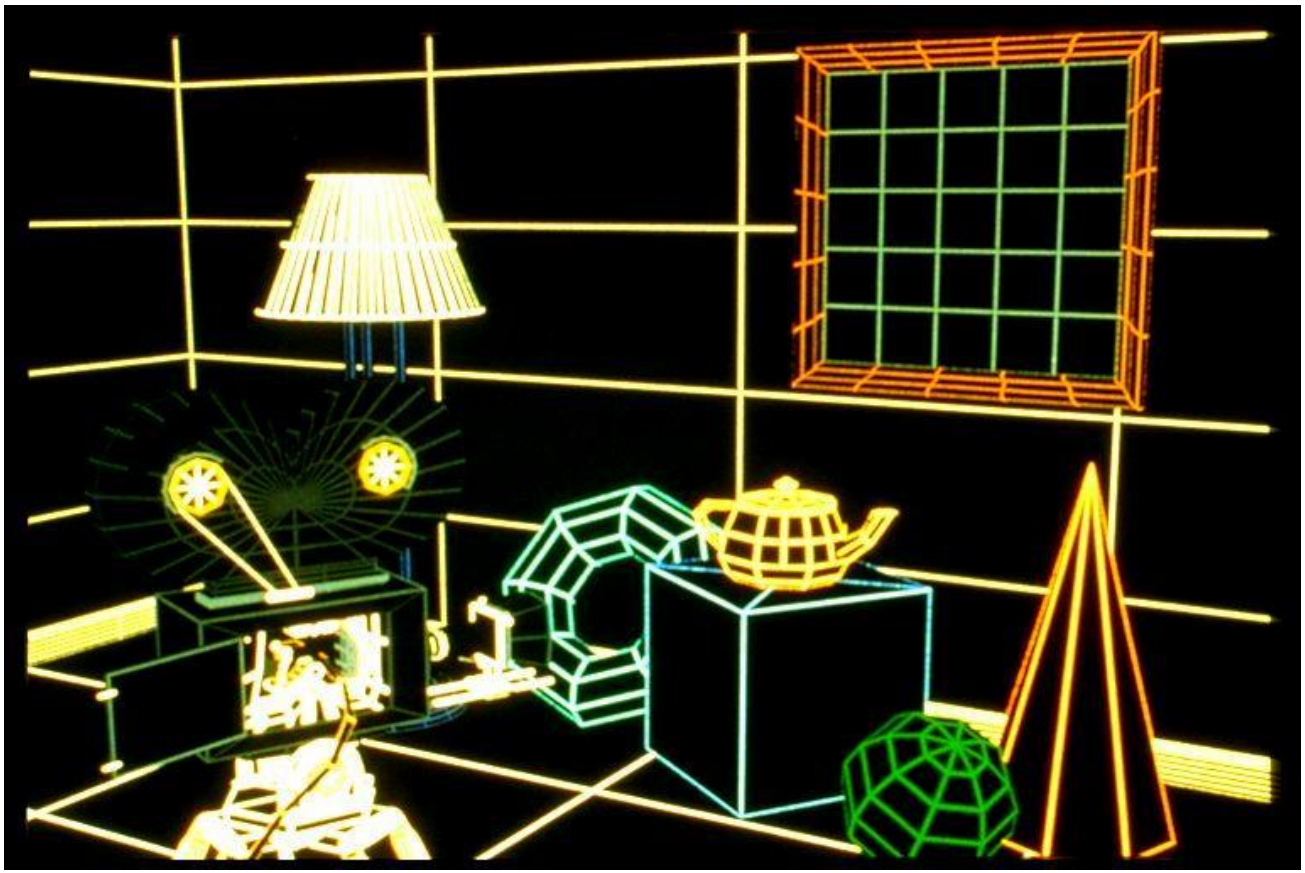






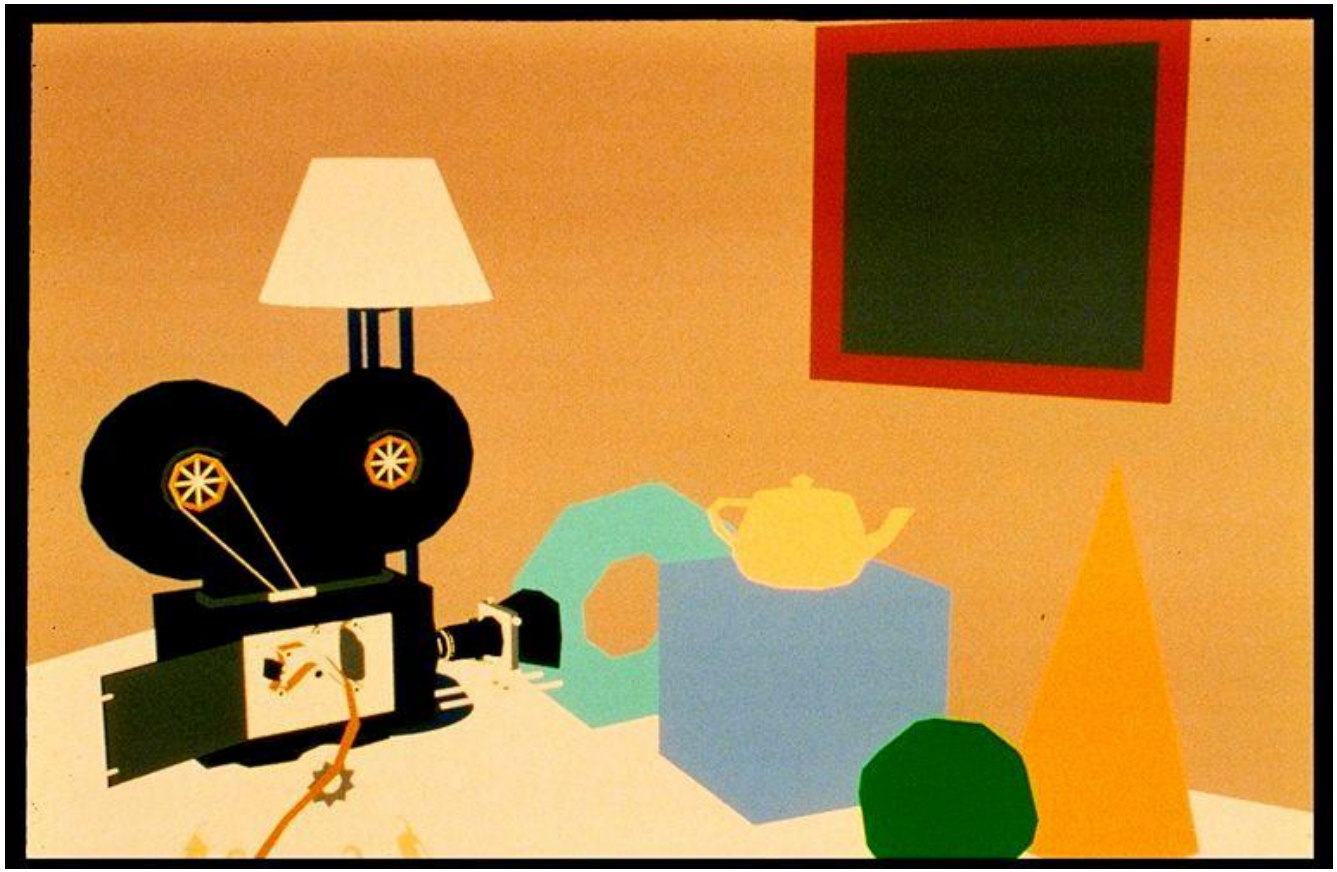
Hidden Surface Removal

Objects occluded by other objects must not be drawn



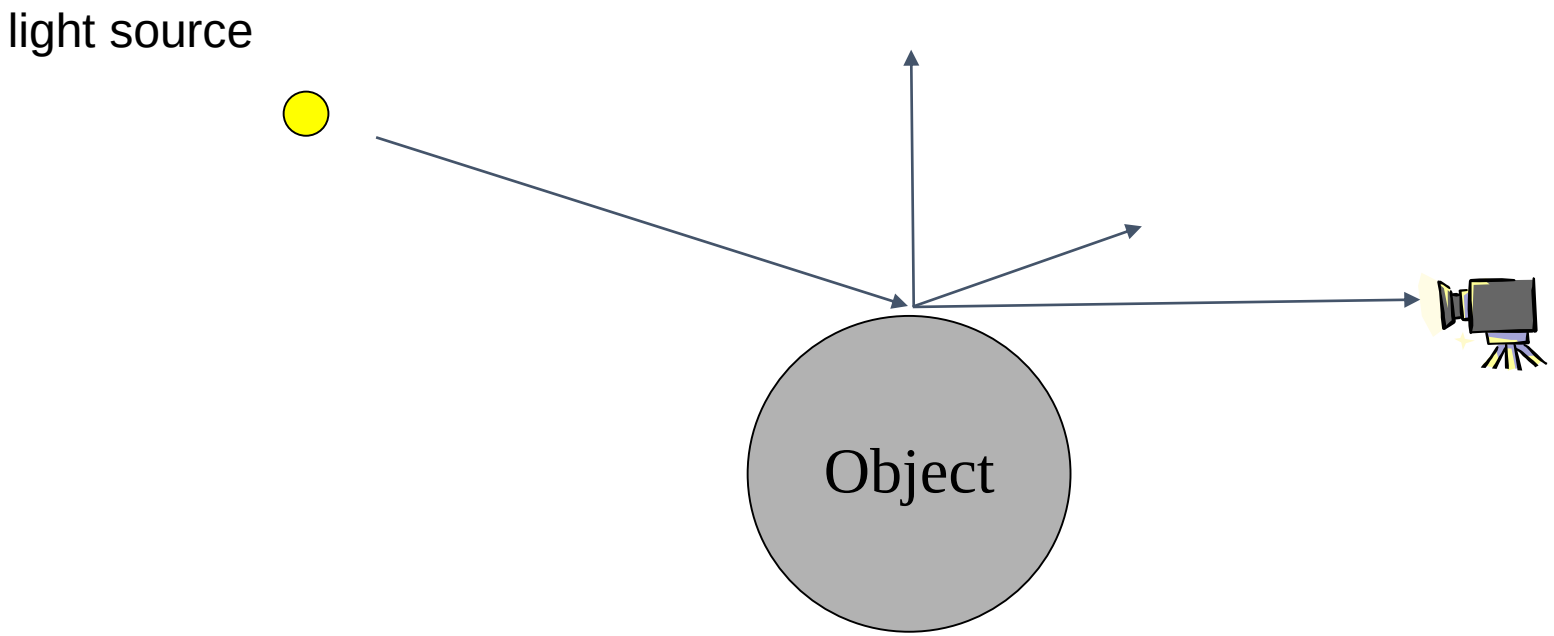
Shading : Constant Shading - Ambient

Objects colours by its own colour



Shading

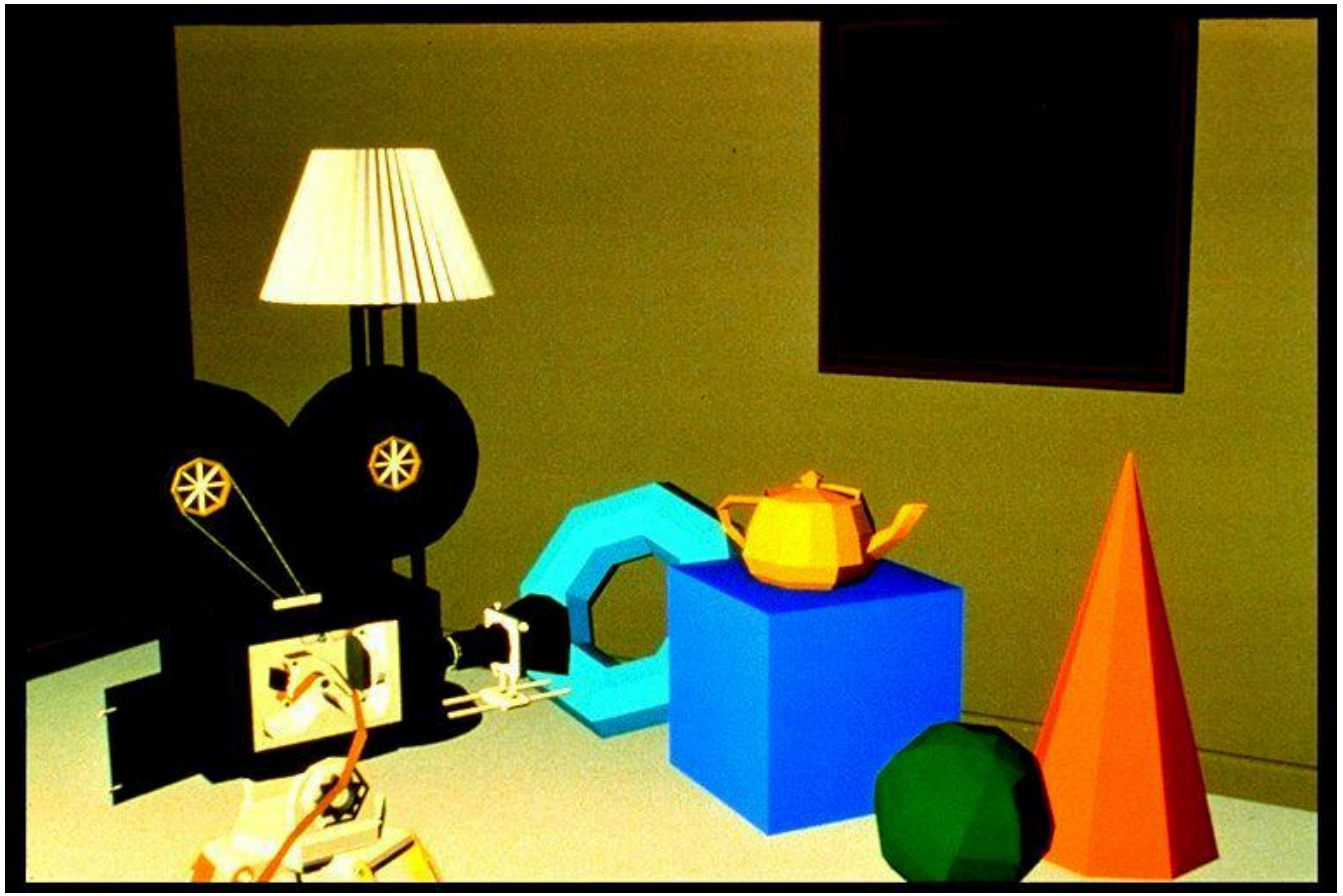
Now we need to decide the colour of each pixels taking into account the object's colour, lighting condition and the camera position



Shading – Flat Shading

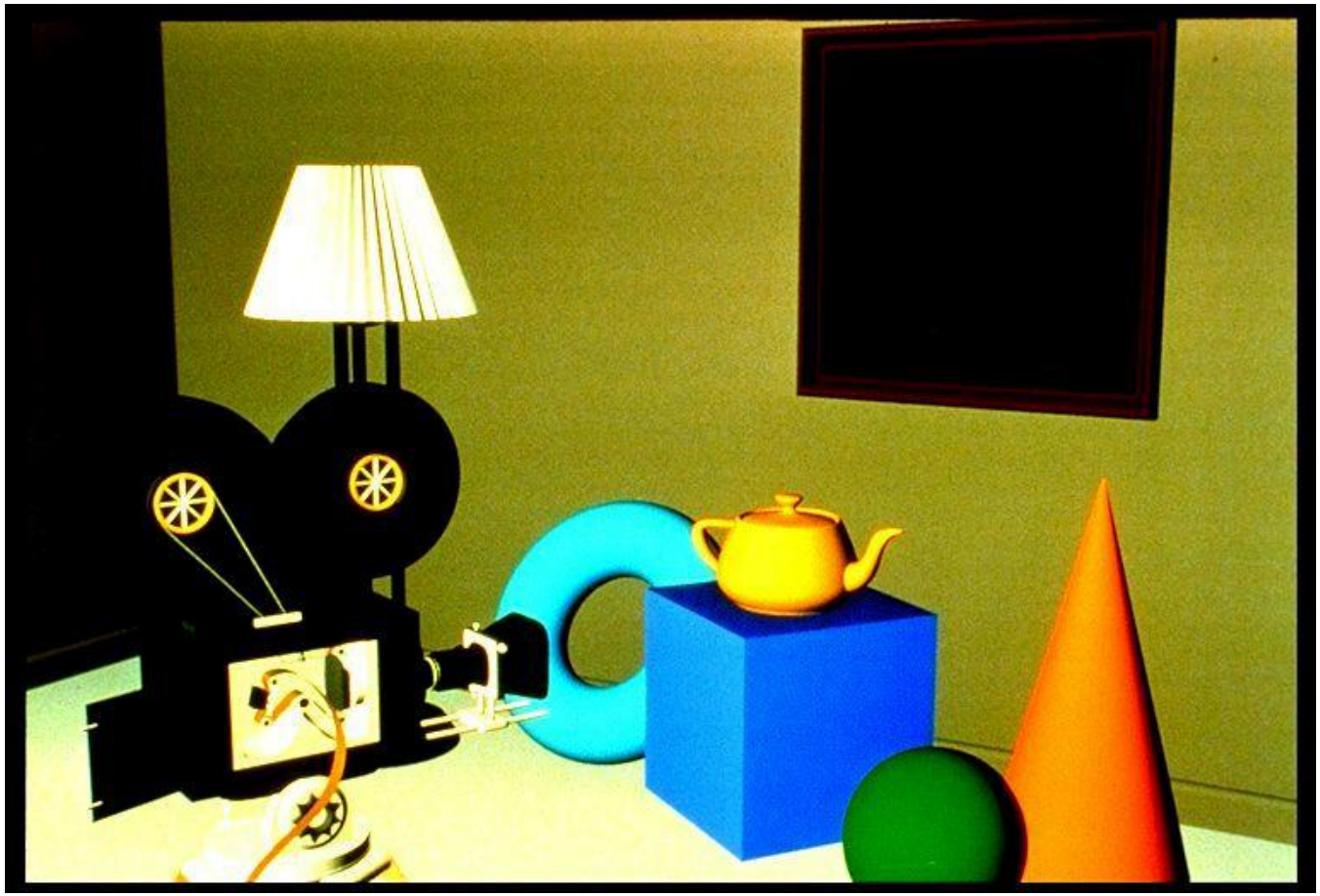
Objects coloured based on its own colour and the lighting condition

One colour for one face

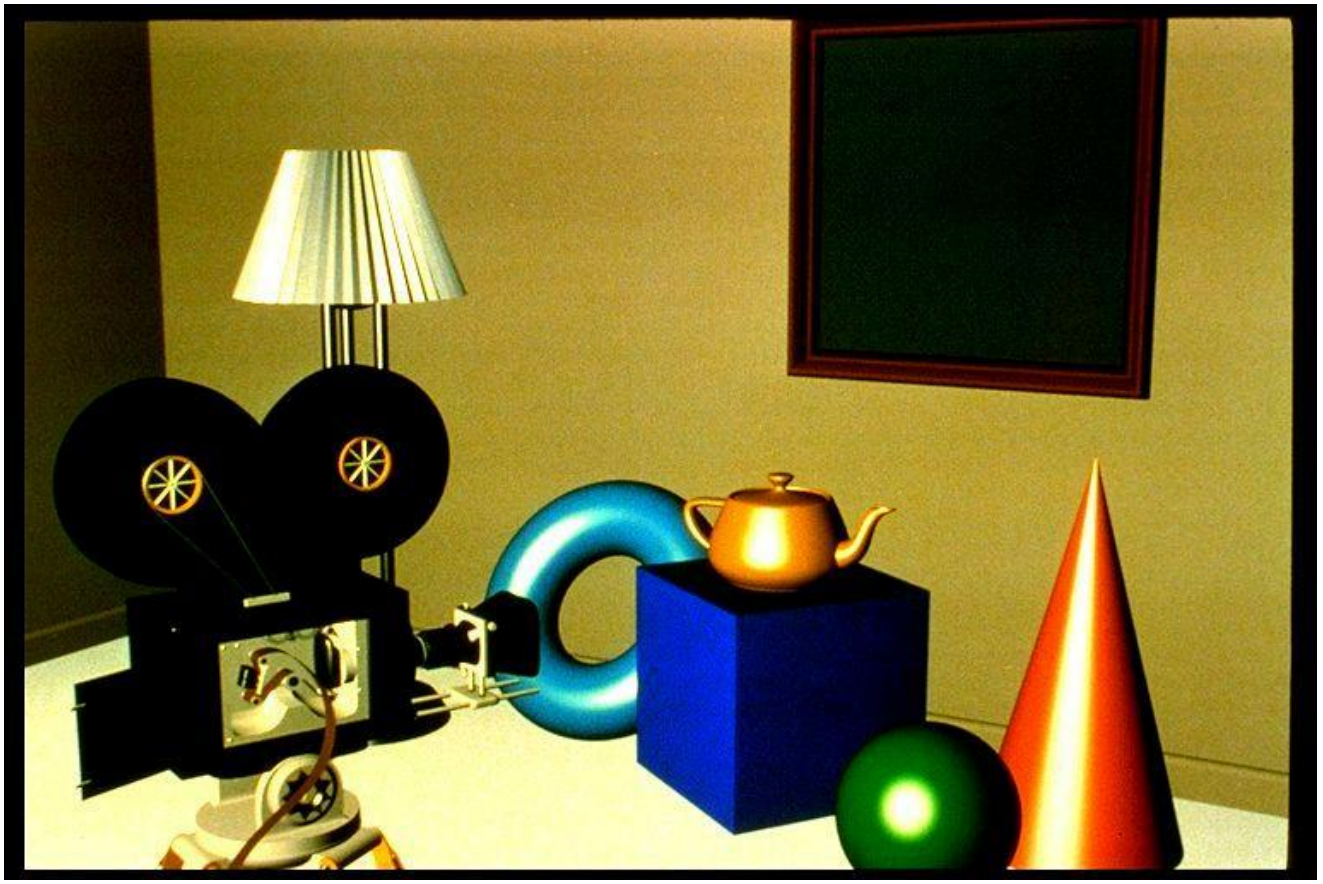


Gouraud shading, no specular highlights

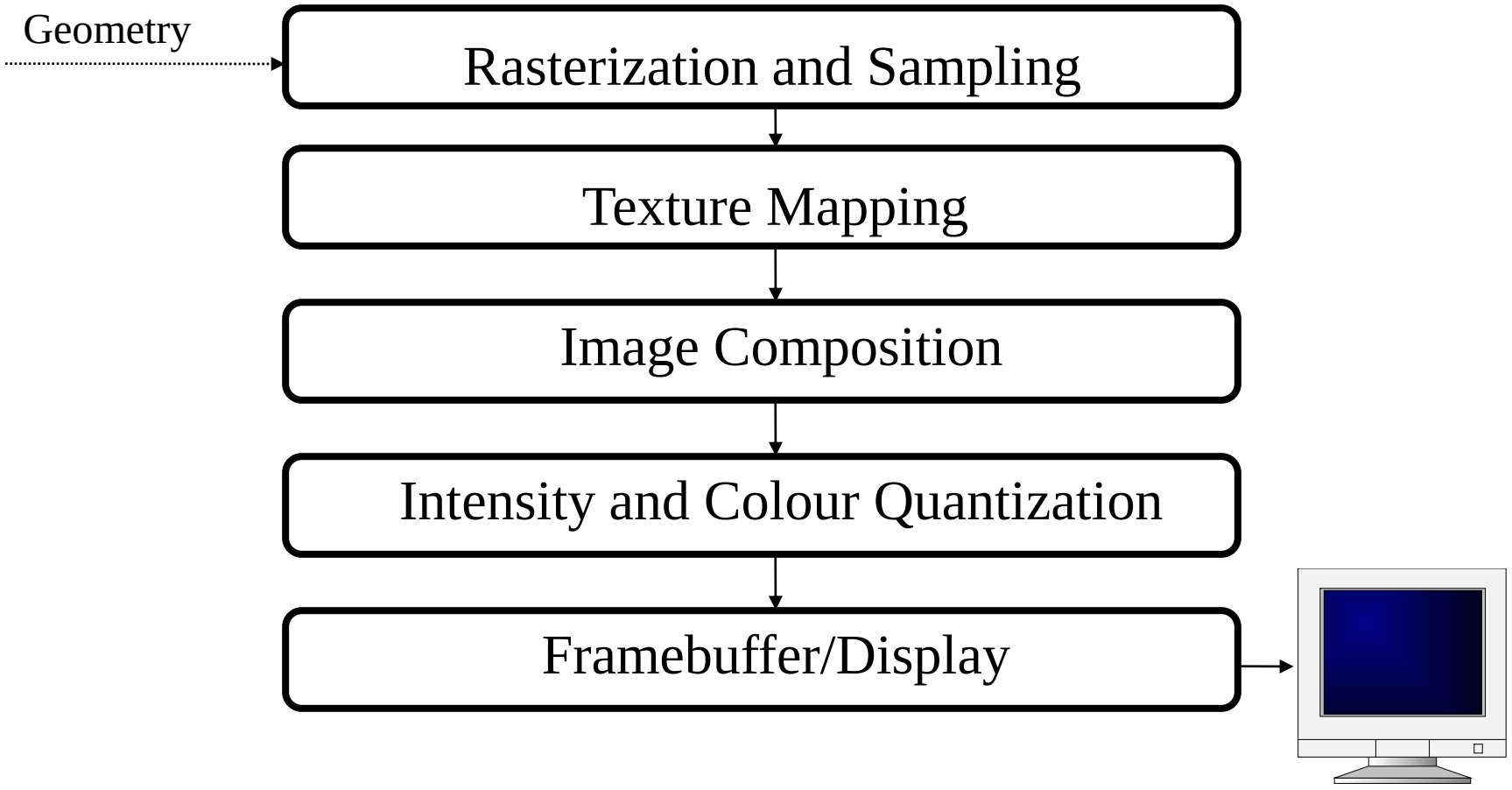
Lighting calculation per vertex



Phong shading

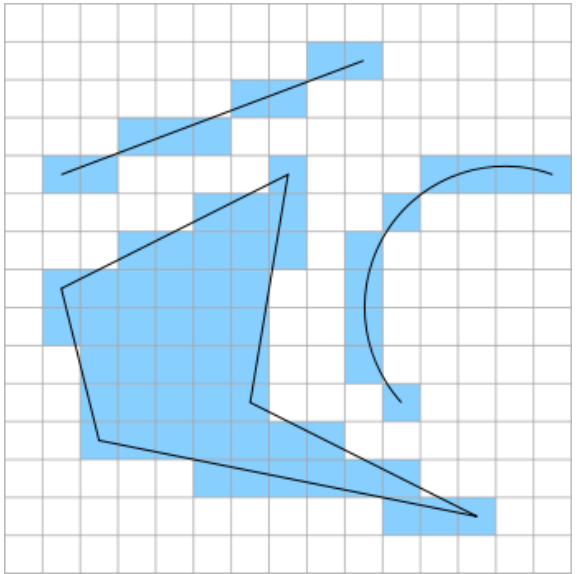


Next, the Imaging Pipeline

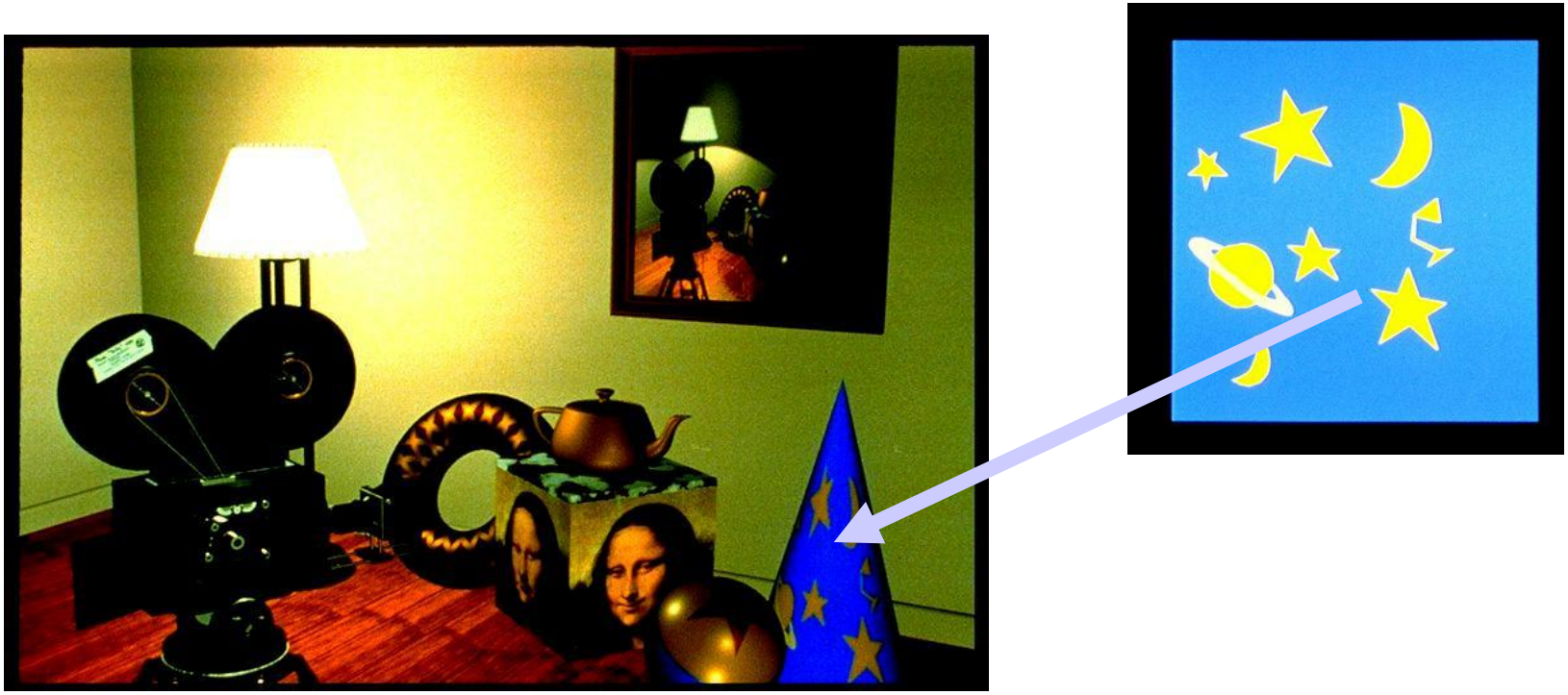


Rasterization

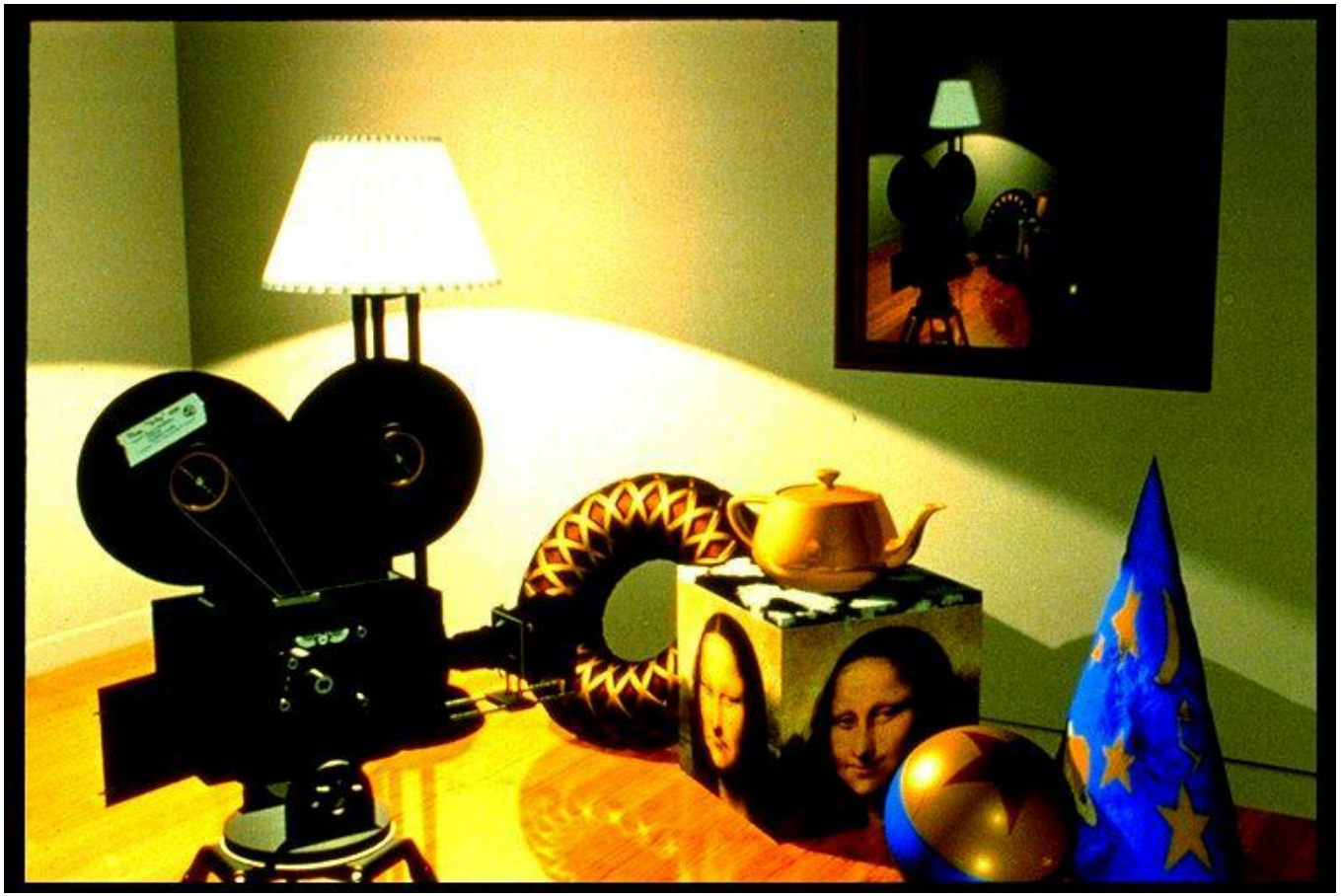
- Converts the vertex information output by the geometry pipeline into pixel information needed by the video display
- Aliasing: distortion artefacts produced when representing a high-resolution signal at a lower resolution.
- Anti-aliasing : technique to remove aliasing



Texture mapping



Reflections, shadows & Bump mapping

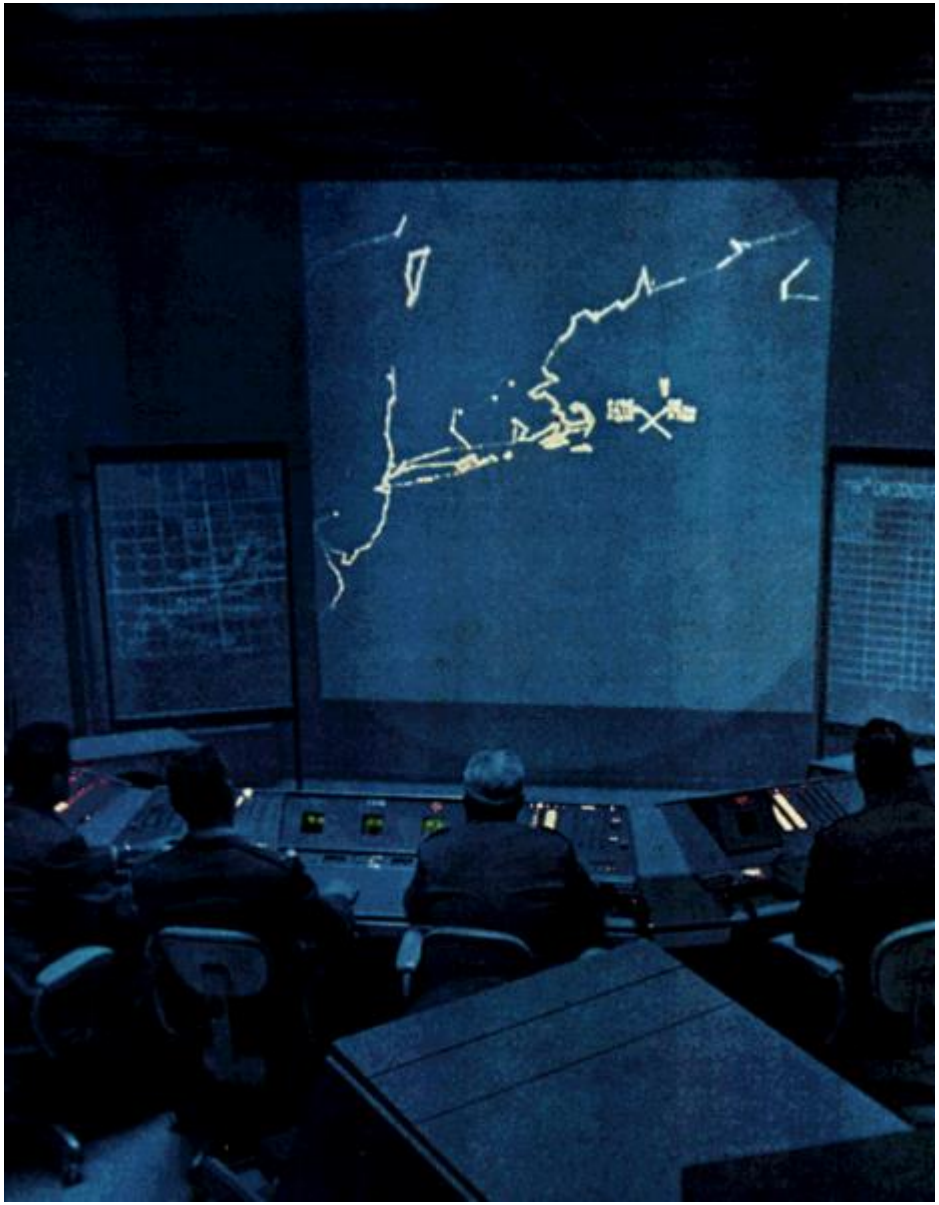


More advanced: Global Illumination

- Questions?

Computer Graphics: 1950-1960

- Computer graphics goes back to the earliest days of computing
  - Strip charts
  - Pen plotters
  - Simple displays using A/D converters to go from computer to calligraphic CRT
- Cost of refresh for CRT too high
  - Computers slow, expensive, unreliable

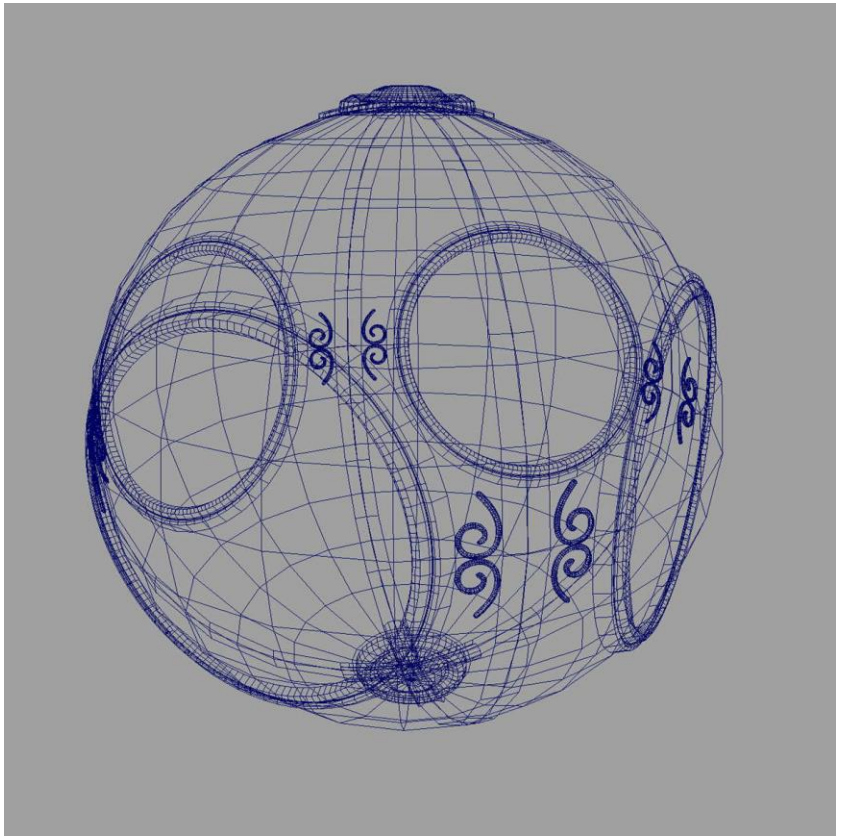


(SAGE project)

Computer Graphics: 1960-1970

- *Wireframe* graphics
  - Draw only lines
- Sketchpad
- Display Processors
- Bézier curve

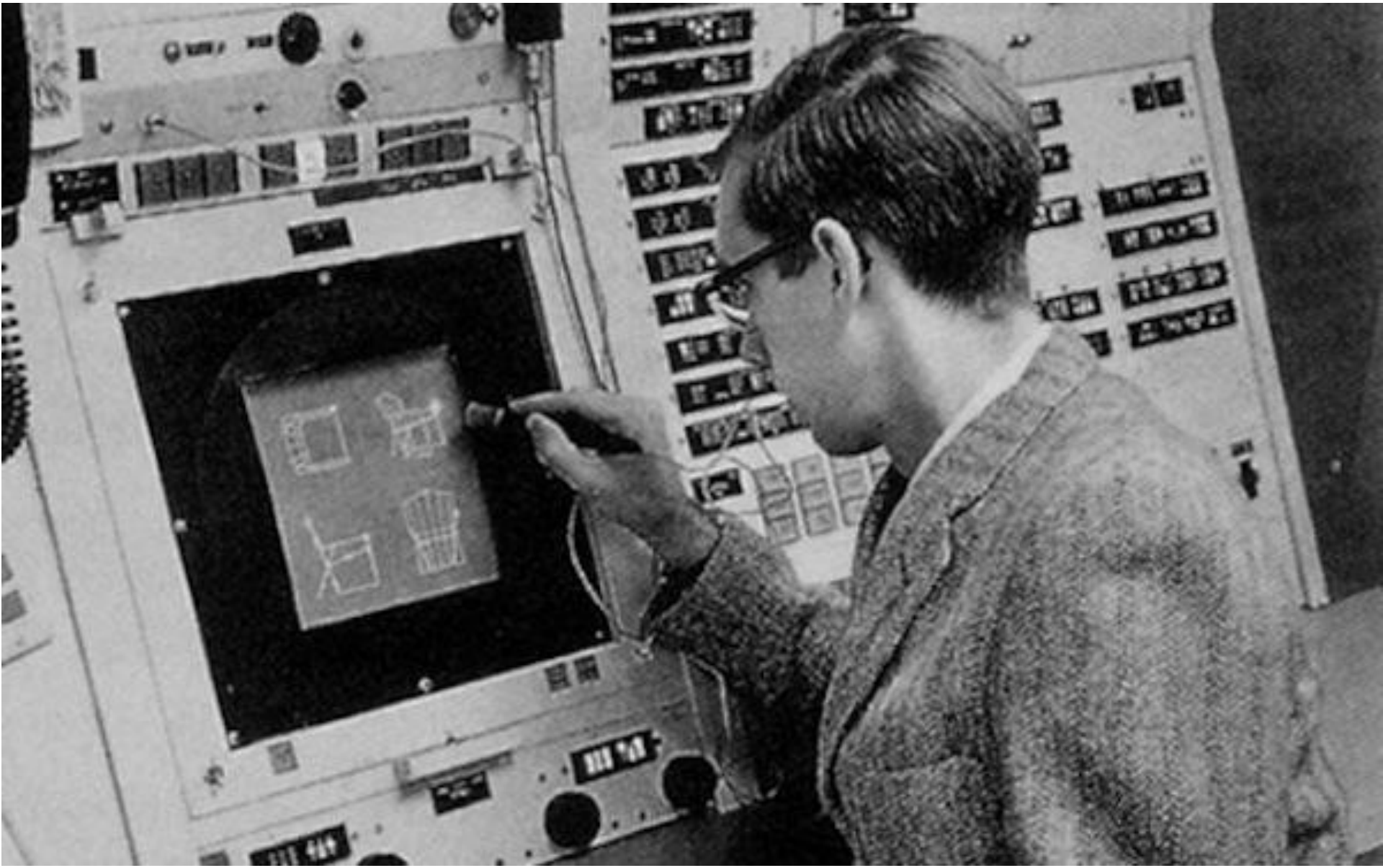
wireframe representation of an object



(Angel 2005)

Sketchpad

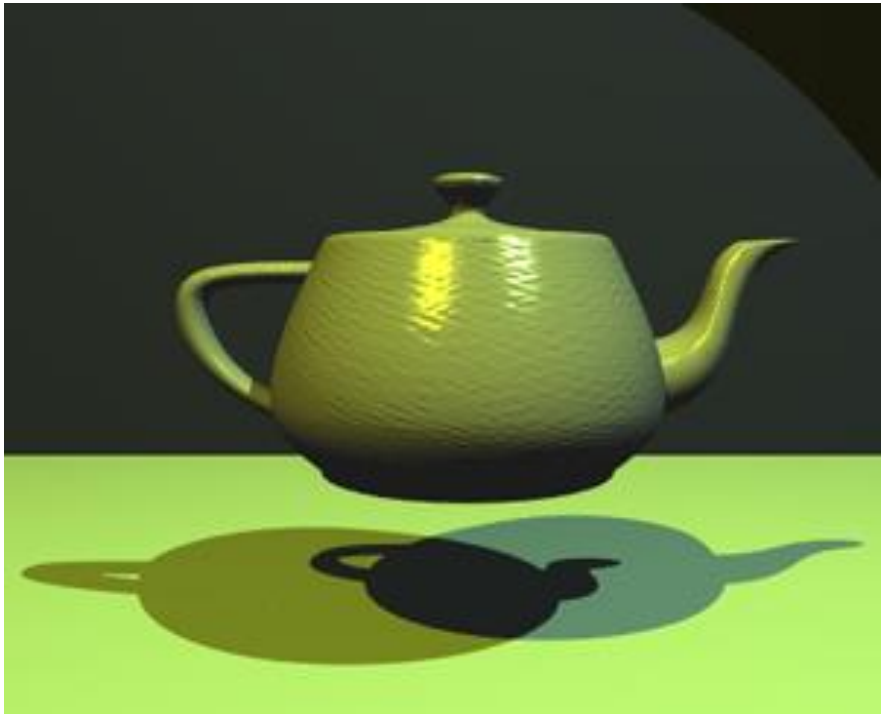
- Ivan Sutherland's PhD thesis at MIT
  - Recognized the potential of man-machine interaction
  - Loop
    - Display something
    - User moves light pen
    - Computer generates new display
  - Sutherland also created many of the now common algorithms for computer graphics
  - HMD



http://www.ifanr.com/231110

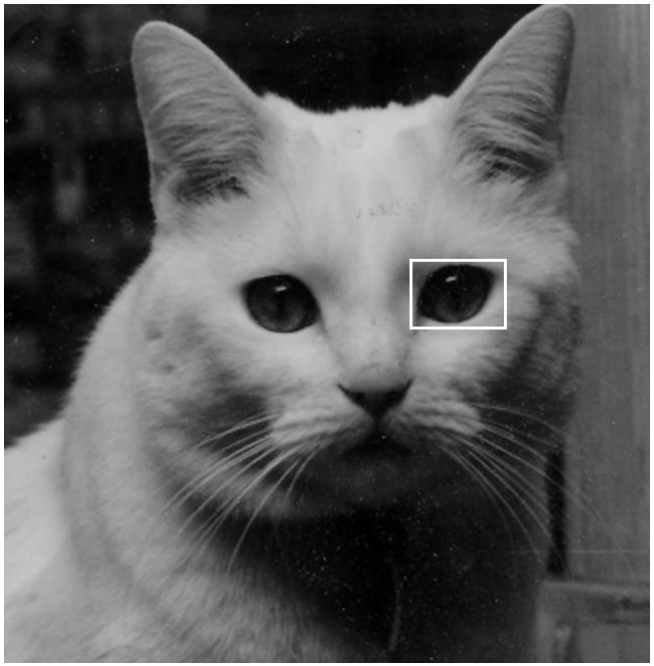
Computer Graphics: 1970-1980

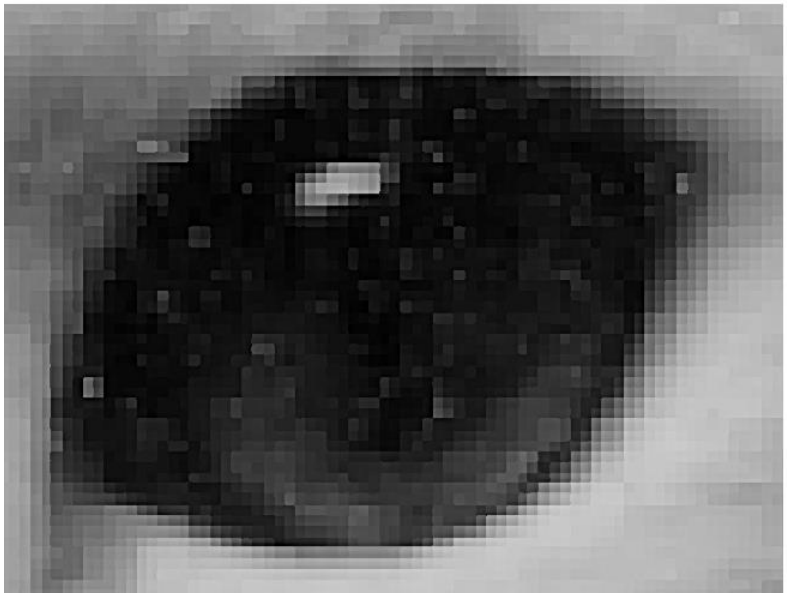
- Raster Graphics
- Gouraud shading and Blinn-Phong shading
- University of Utah
  - Edwin Catmull **Pixar**.
  - John Warnock **Adobe**



Raster Graphics

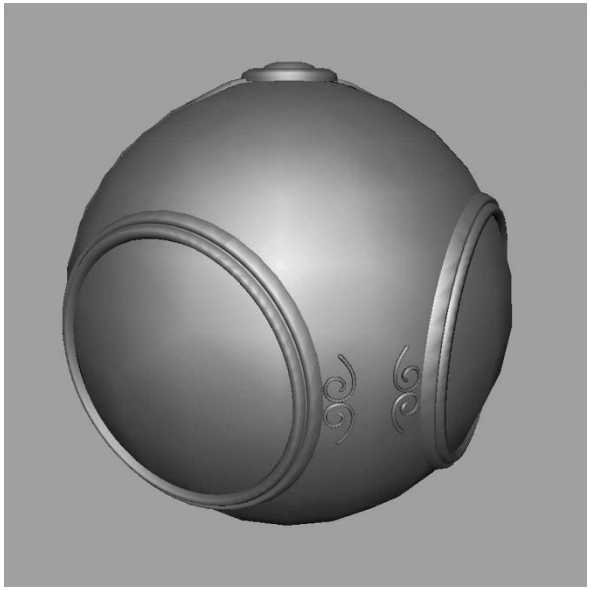
- Image produced as an array (the *raster*) of picture elements (*pixels*) in the *frame buffer*





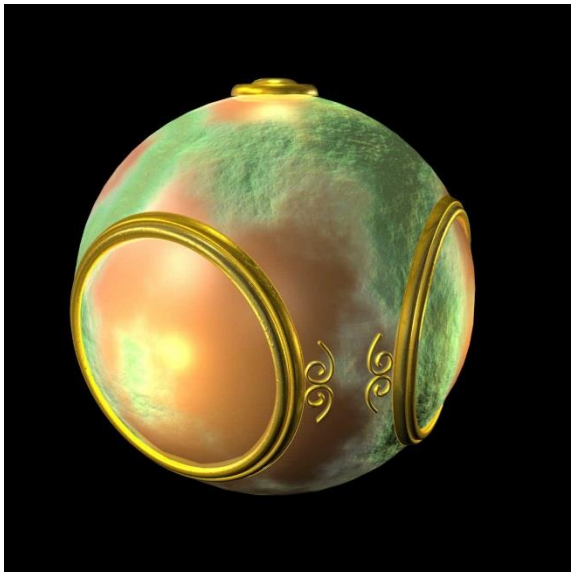
Computer Graphics: 1980-1990

Realism comes to computer graphics



smooth shading

environment mapping



bump mapping

Computer Graphics: 1980-1990

- Special purpose hardware
  - Silicon Graphics geometry engine
    - VLSI implementation of graphics pipeline
- Industry-based standards
  - PHIGS
  - RenderMan
- Graphical user interfaces (GUI)
  - Macintosh
- Computer generated images CGI
  - Star Trek

The Enterprise

Computer Graphics: 1990-2000

- OpenGL API (1992)
- Completely computer-generated feature-length movies (Toy Story) are successful
- New hardware capabilities
  - Texture mapping
  - Blending
  - Accumulation, stencil buffers
- **Autodesk**
- **Id software**

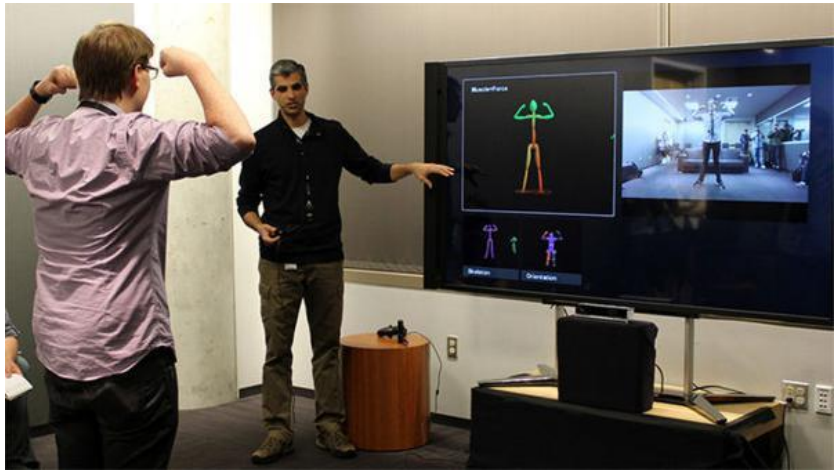
1995 Toy Story

Computer Graphics: 2000-2010

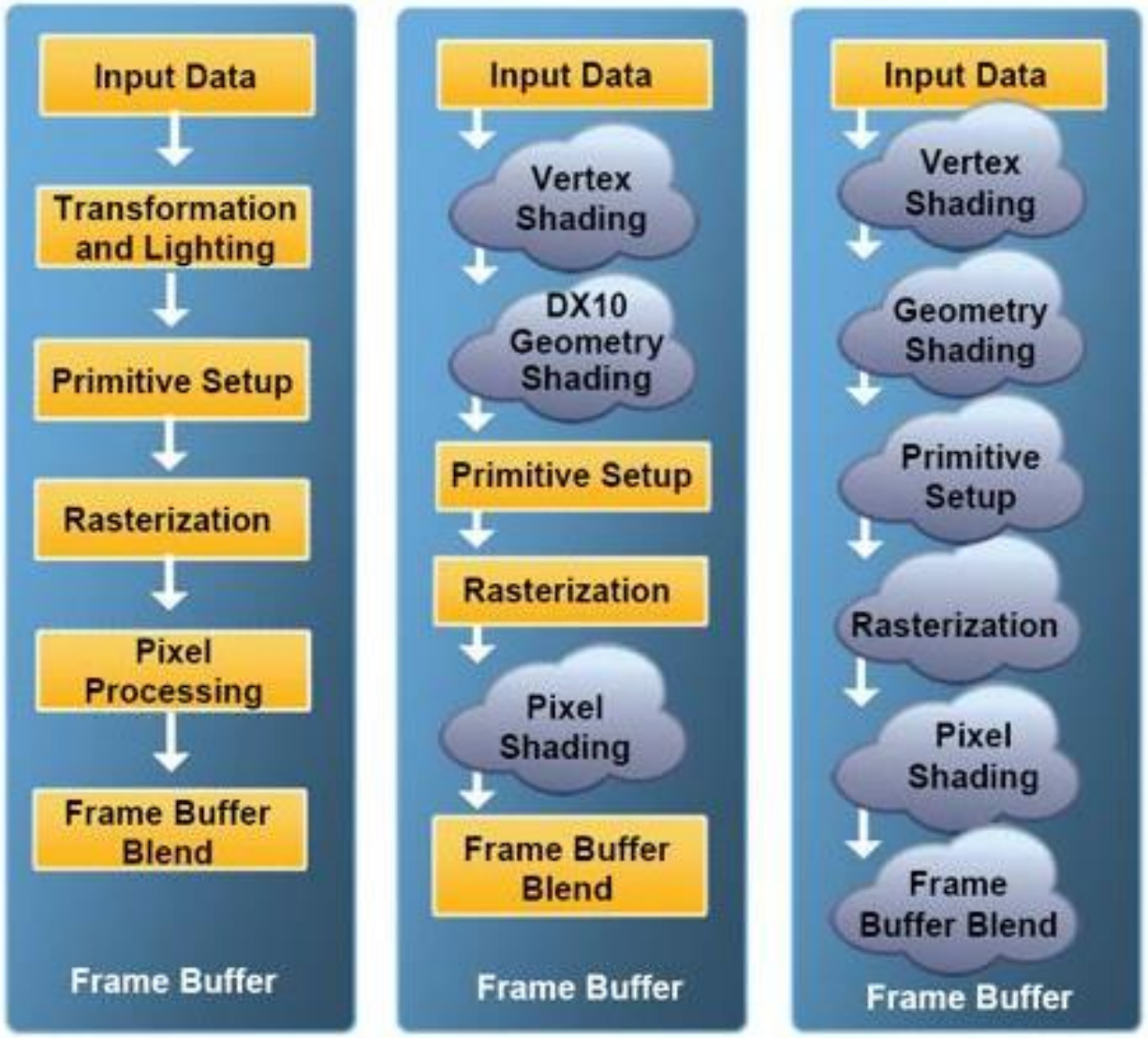
- Photorealism
- Graphics cards for PCs dominate market
  - Nvidia, ATI, 3DLabs
- Game boxes and game players determine direction of market
- Computer graphics routine in movie industry: Maya, Lightwave
- Programmable pipelines
  - GPU

Computer Graphics: 2010-

- Real-time Photorealism
- GPGPU
  - CUDA, OpenCL
- Cloud/Streaming Graphics
  - Nvidia Grid
- Mobile/Web Graphics
  - WebGL
- Advanced HCI
  - Kinect
  - Hololens



The evolving of Graphics Pipeline



(Intel)

Ray-tracing

- Ancient but also modern
- Nvidia RTK built with ray-tracing core!
  - https://www.youtube.com/watch?v=5dRKOMAOdwE

- Questions?

Graphics Applications

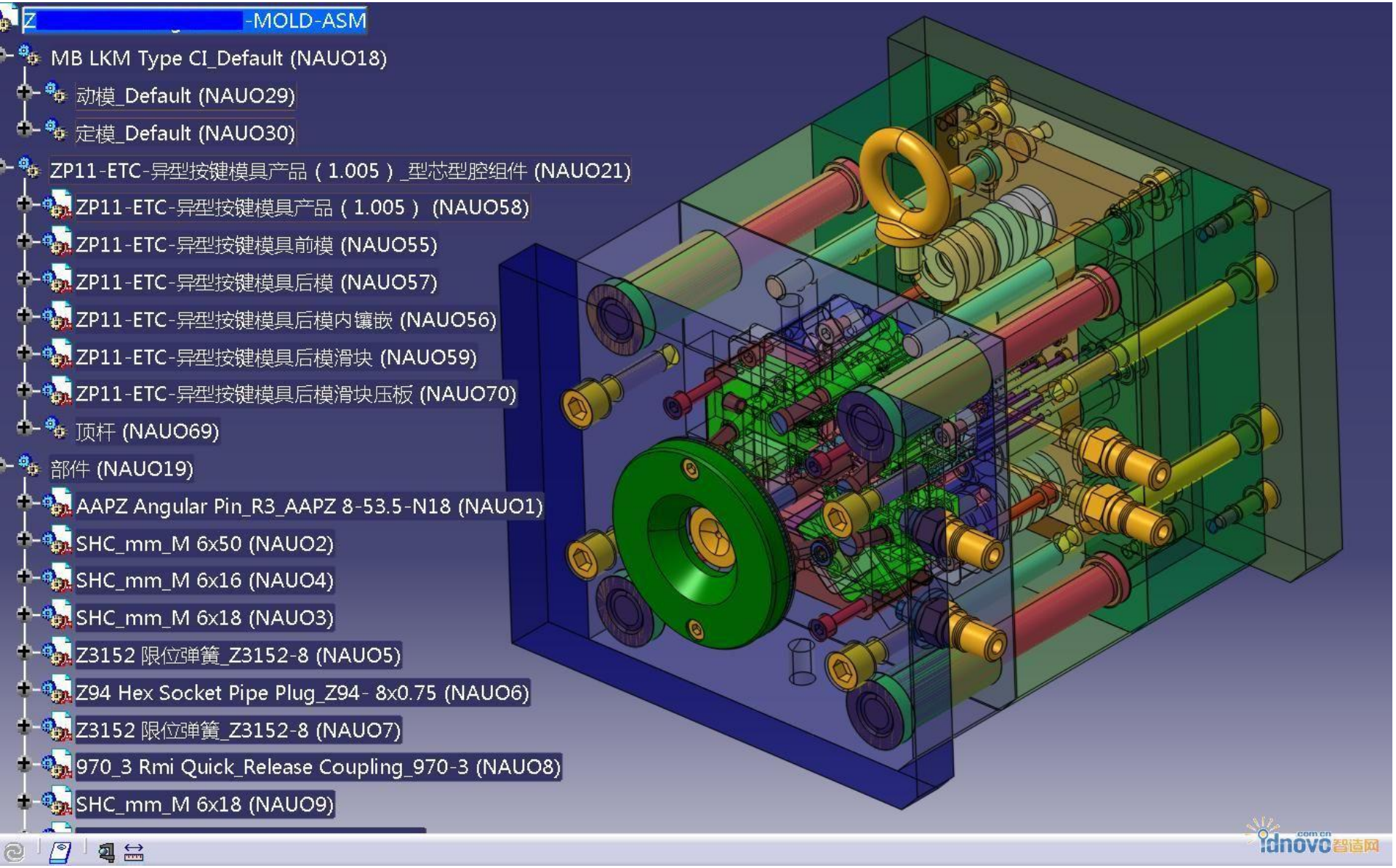
- Entertainment
  - film production
  - film effects
  - games
- Science and engineering
  - computer-aided design
  - visualization (scientific, information)
- Virtual Prototyping
- Cultural Heritage
- Training & Simulation
- Graphic Arts, Fine Art

Graphics Applications

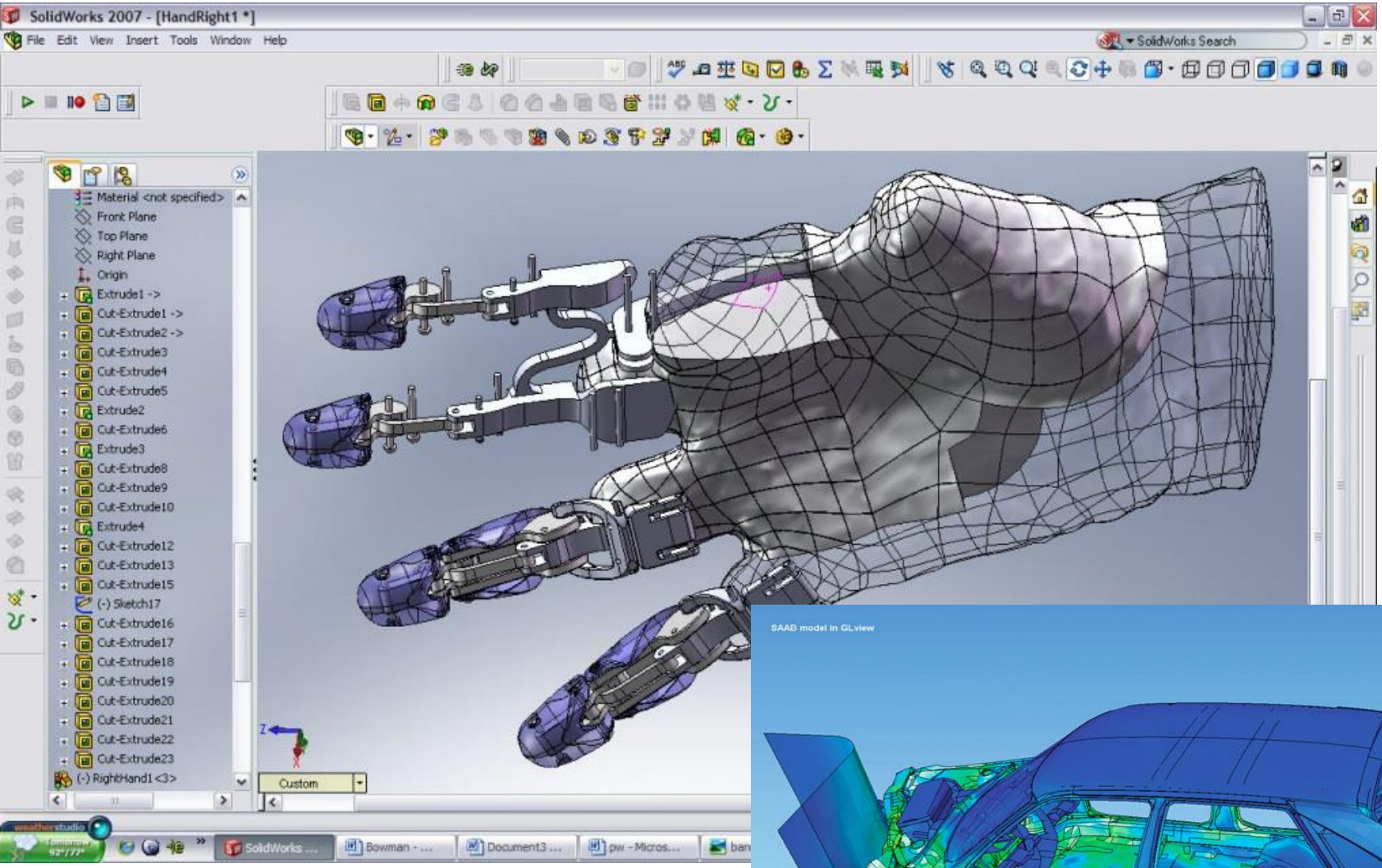
- Entertainment
  - film production
  - film effects
  - games
- Science and engineering
  - computer-aided design
  - visualization (scientific, information)
- Virtual Prototyping
- Cultural Heritage
- Training & Simulation
- Graphic Arts, Fine Art

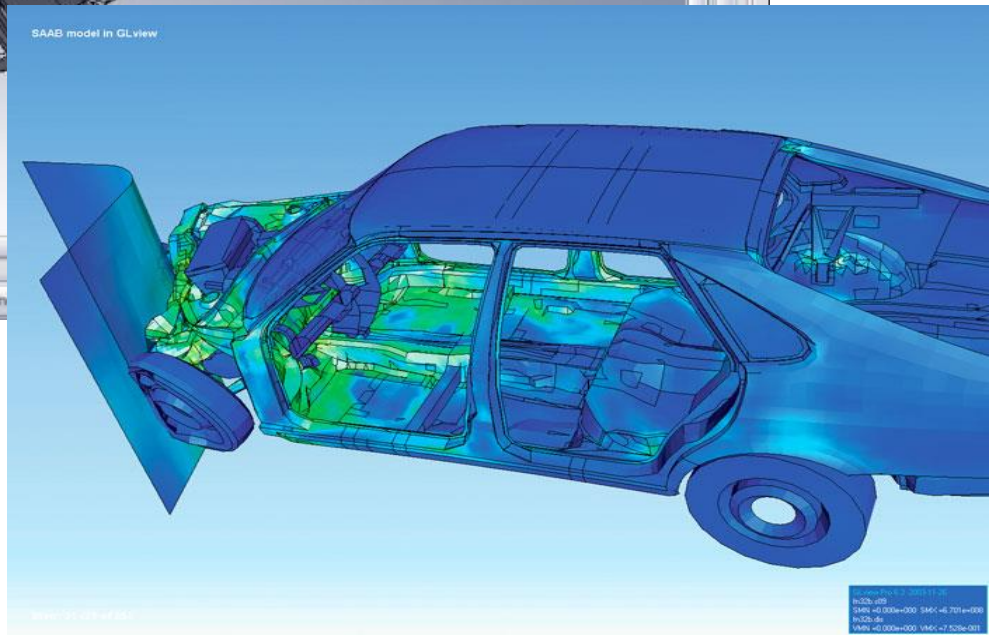
Graphics Applications

- Entertainment
  - film production
  - film effects
  - games
- Science and engineering
  - computer-aided design
  - visualization (scientific, information)
- Virtual Prototyping
- Cultural Heritage
- Training & Simulation
- Graphic Arts, Fine Art



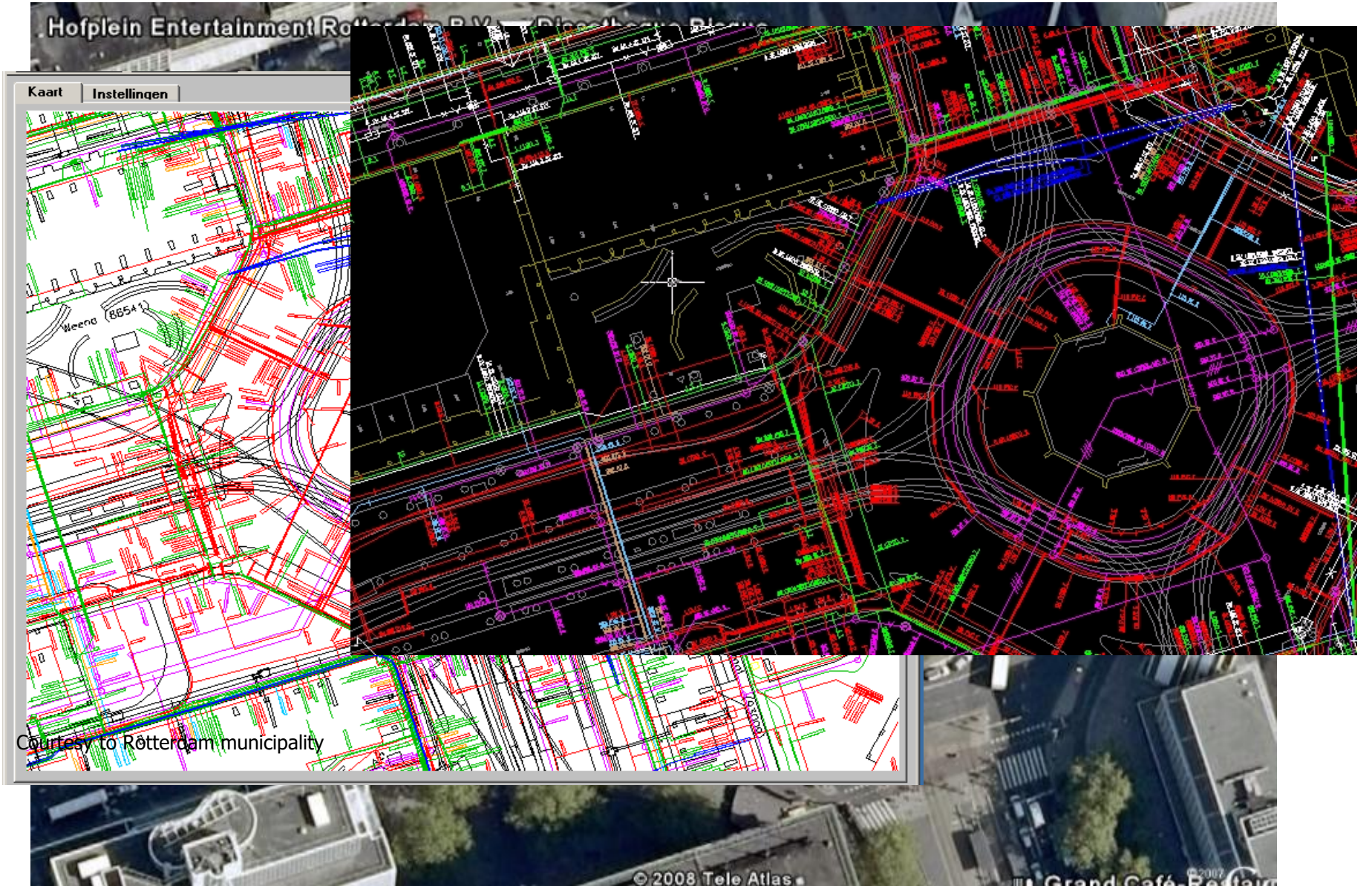
Dassault Catia



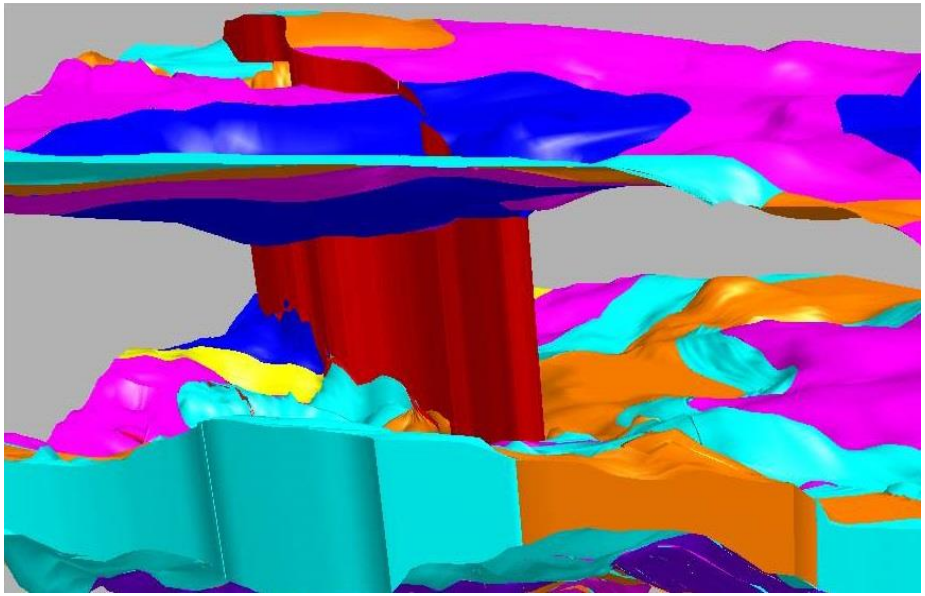


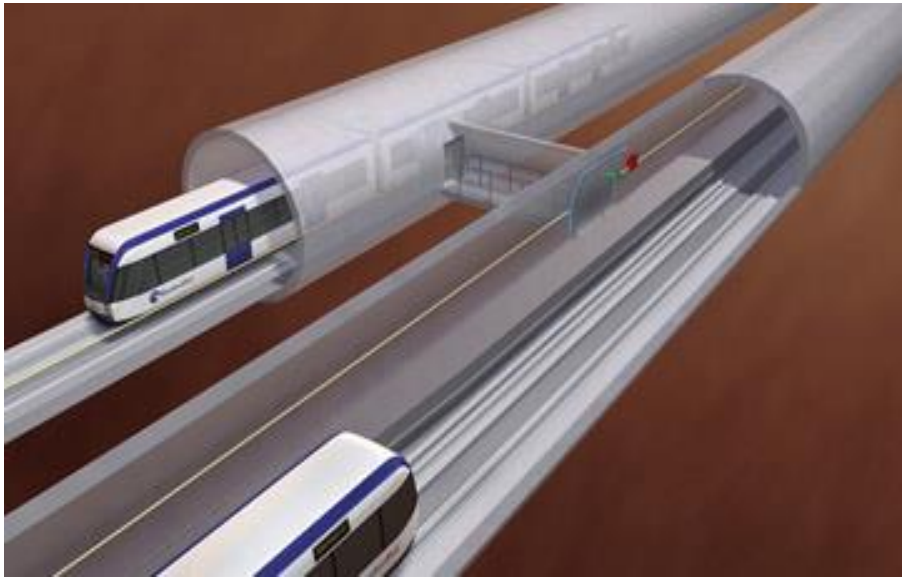
Dassault Solidworks

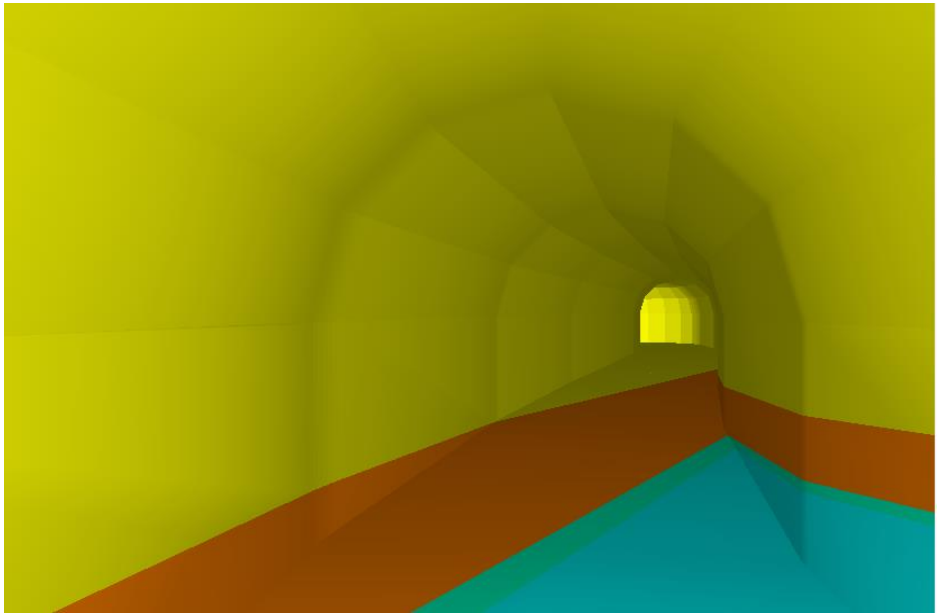
Planning

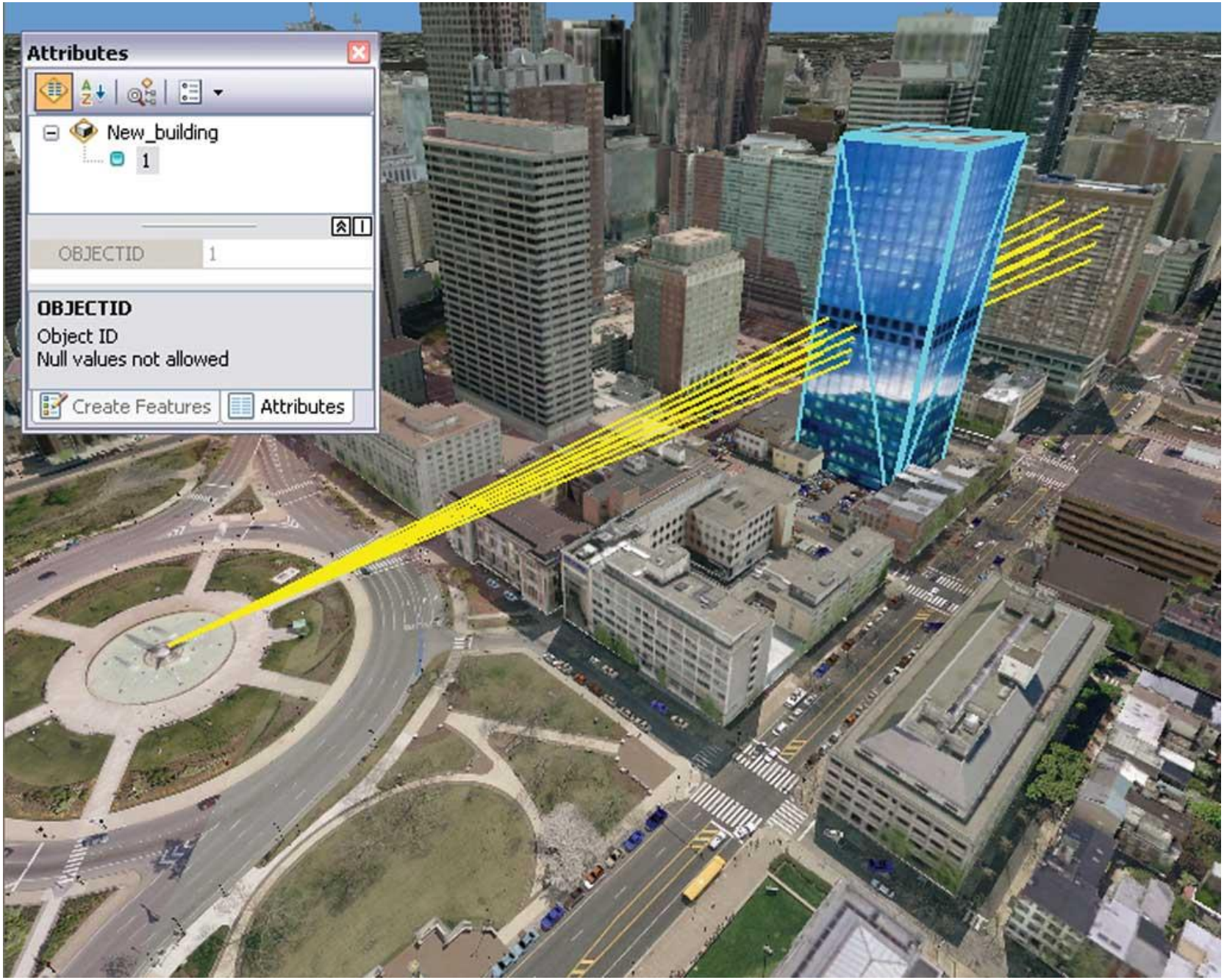


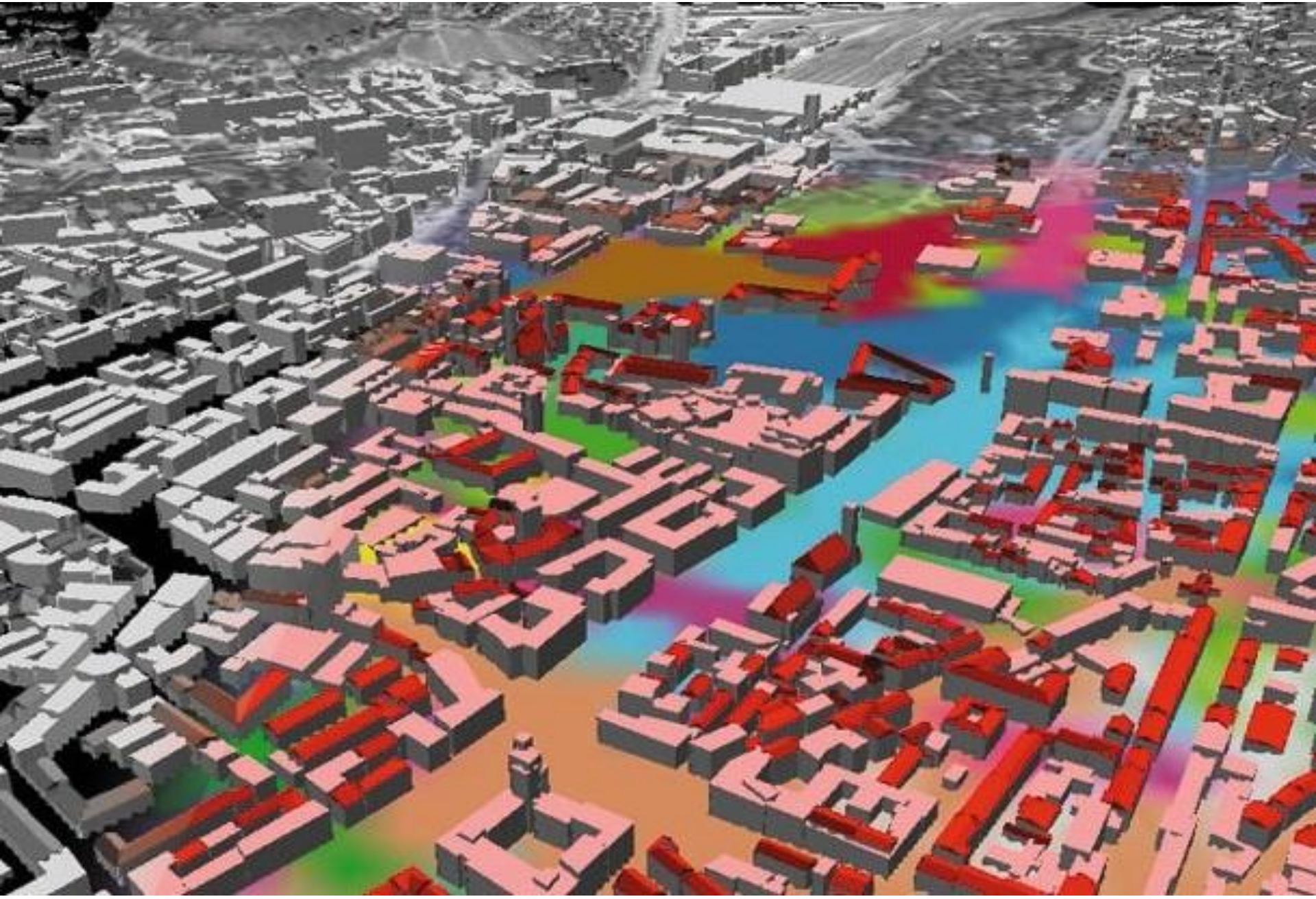
Courtesy to Rotterdam municipality



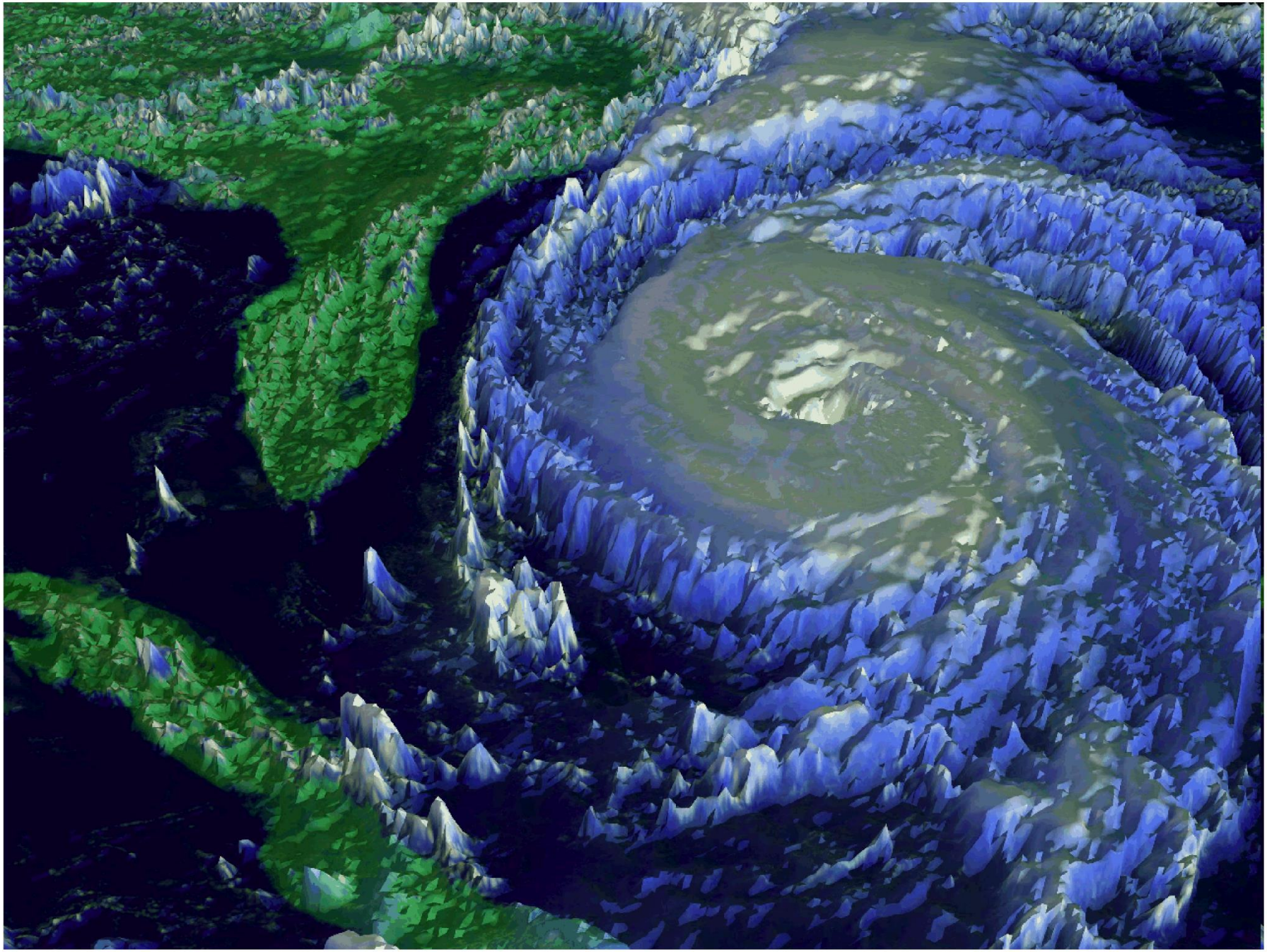


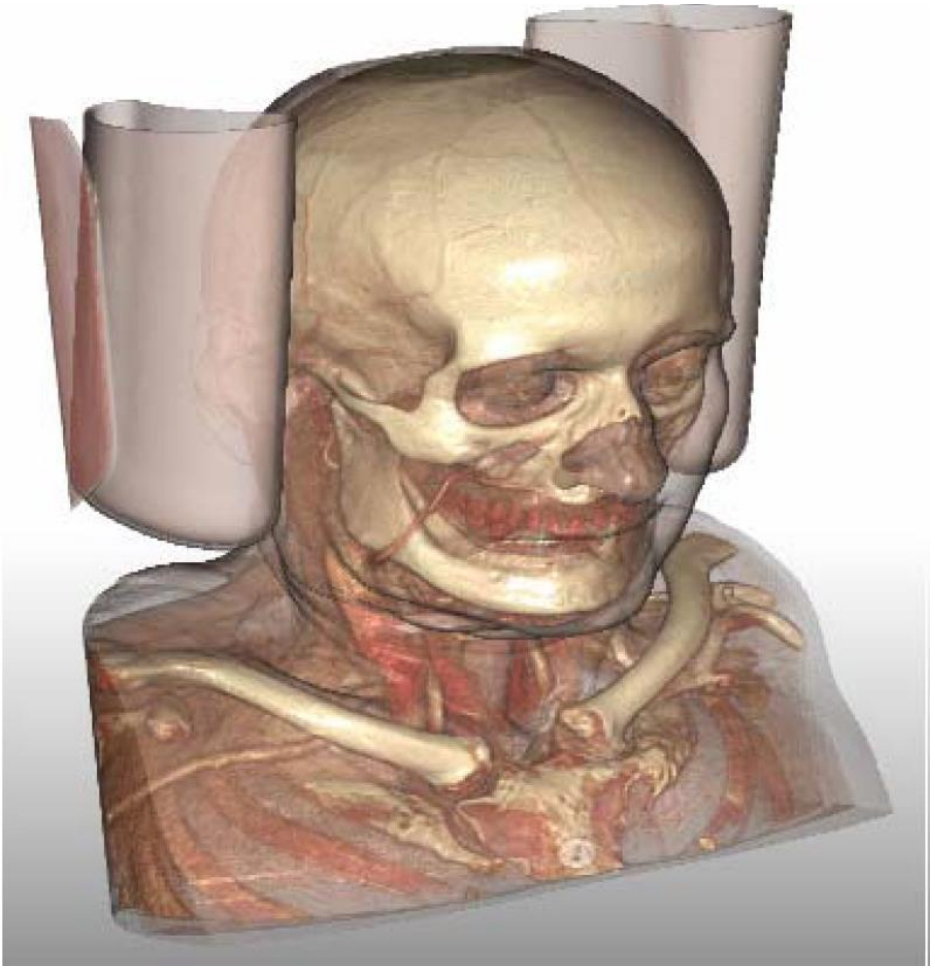


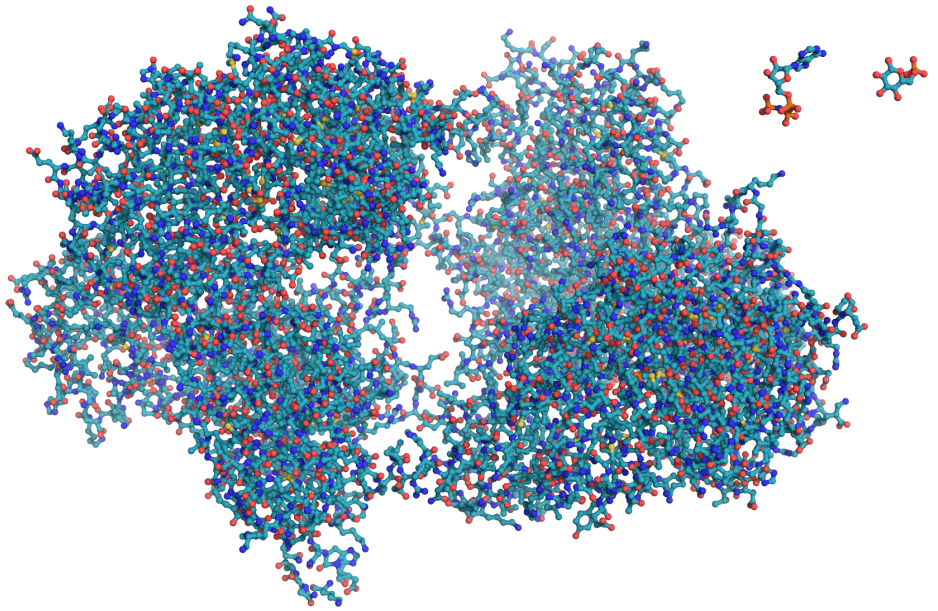

Attributes
New_building
1
OBJECTID
1
OBJECTID
Object ID
Null values not allowed
Create Features
Attributes



(Courtesy to Sisi)





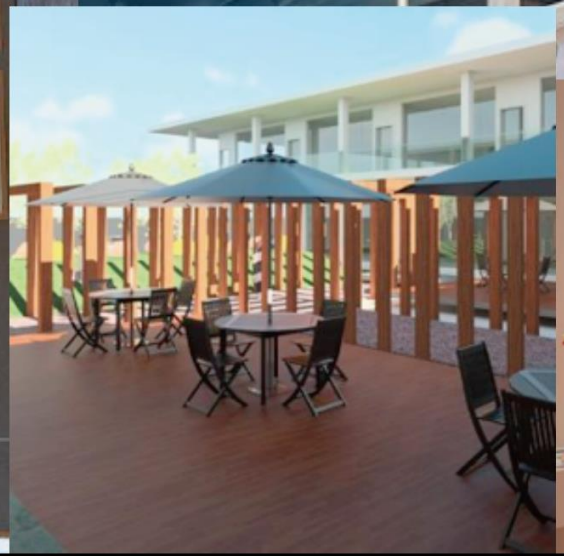


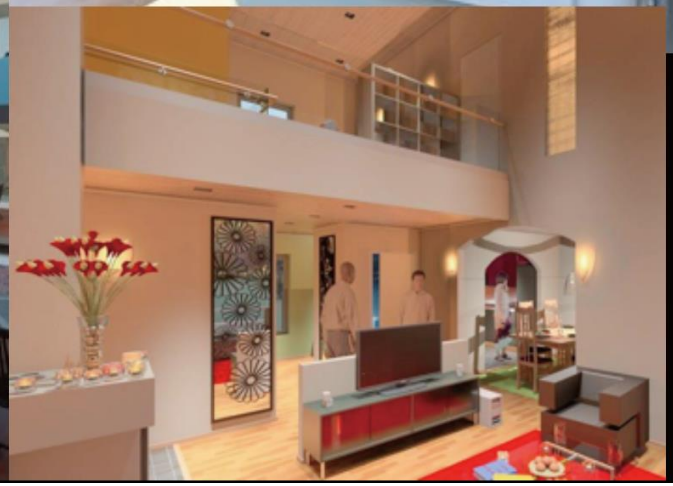
Graphics Applications

- Entertainment
  - film production
  - film effects
  - games
- Science and engineering
  - computer-aided design
  - visualization (scientific, information)
- Virtual Prototyping
- Cultural Heritage
- Training & Simulation
- Graphic Arts, Fine Art

Autodesk® 360 Rendering

Create photorealistic images and panoramas using our Rendering cloud services with your Autodesk® 360



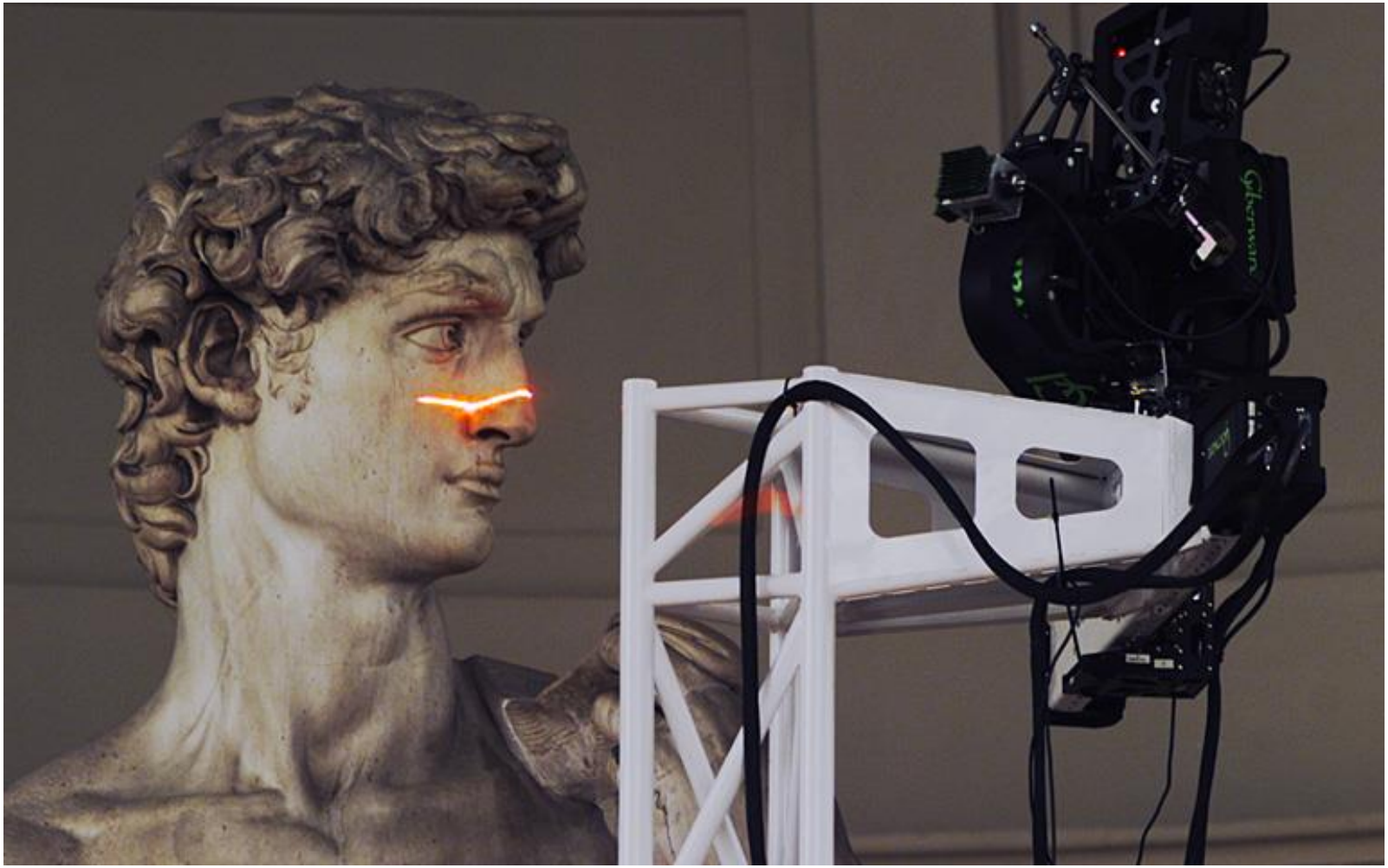


Autodesk 360

Graphics Applications

- Entertainment
  - film production
  - film effects
  - games
- Science and engineering
  - computer-aided design
  - visualization (scientific, information)
- Virtual Prototyping
- Cultural Heritage
- Training & Simulation
- Graphic Arts, Fine Art

The Digital Michelangelo Project



Marc Levoy

The Virtual Pompeii

Courtesy to ESRI

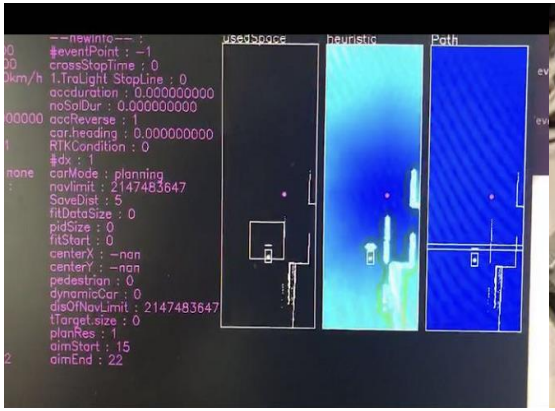
Courtesy to ESRI

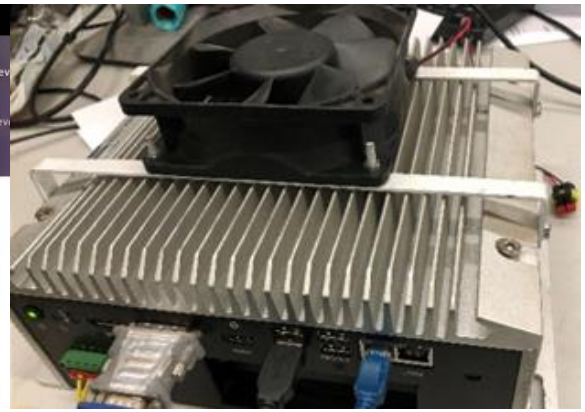
Graphics Applications

- Entertainment
  - film production
  - film effects
  - games
- Science and engineering
  - computer-aided design
  - visualization (scientific, information)
- Virtual Prototyping
- Cultural Heritage
- Training & Simulation
- Graphic Arts, Fine Art


apollo

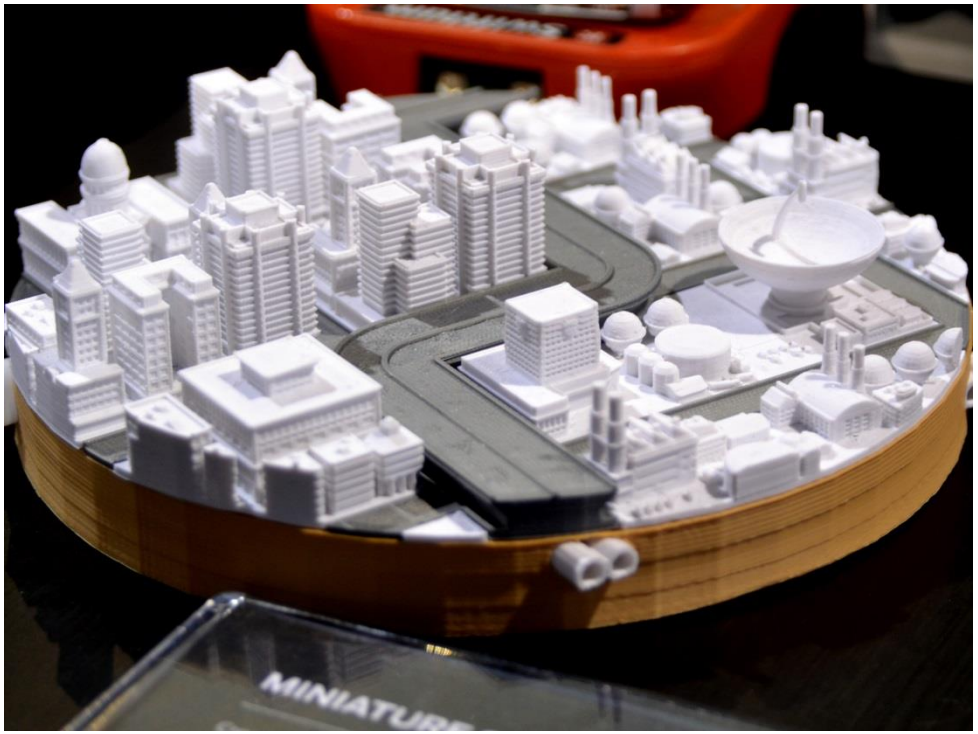

pygame window


--newInfo-- :
#eventPoint : -1
crossStopTime : 0
1.TraLight StopLine : 0
accduration : 0.000000000
noSolDur : 0.000000000
accReverse : 1
car.heading : 0.000000000
RTKCondition : 0
#dx : 1
carMode : planning
navlimit : 2147483647
SaveDist : 5
fitDataSize : 0
pidSize : 0
fitStart : 0
centerX : -nan
centerY : -nan
pedestrian : 0
dynamicCar : 0
disOfNavLimit : 2147483647
tTarget.size : 0
planRes : 1
aimStart : 15
aimEnd : 22
usedSpace
heuristic
Path




Server: 12 FPS
Client: 45 FPS
Vehicle: Lincoln MKZ2017
Map: Town03
Simulation time: 1 day, 2:51:23
Speed: 13 km/h
Heading: -93° SW
Location: (346.3, -85.8)
Height: -0 m
Throttle:
Steer:
Brake:
Reverse:
Hand brake:
Collision:
Number of vehicles: 22
Nearby vehicles
21m Lincoln MKZ2017
46m Lincoln MKZ2017
133m Lincoln MKZ2017
148m Lincoln MKZ2017
185m Lincoln MKZ2017
169m Lincoln MKZ2017
CARLA

3D Printing



The Course Layout

- Part 1 Introduction
  - Introduction to Computer Graphics
    - Applications
    - History
  - Basic Viewing and Modeling
    - Graphics Pipeline
    - 2D and 3D viewing basics
    - Geometric models
  - Introduction to Modern OpenGL
    - GLSL
    - Simple program

The Course Layout

- Part 2 Viewing
  - 2D and 3D Transformation
    - Coordinate frames
    - Homogeneous coordinates
    - Affined transformations
  - 2D and 3D Viewing
    - Clipping window
    - Viewport
    - Camera
    - Projection and viewing frustum
  - Visible Surface Detection
    - Depth
    - Z-buffer

The Course Layout

- Part 3 Rendering
  - Graphics Output Primitives
    - Line, circle
    - Polygon fill
  - Illumination Model and Surface Rendering
    - Illumination model
      - Light sources
      - Reflection model
    - Surface rendering
  - Texture mapping

The Course Layout

- Part 4 Geometric modeling
  - Triangle mesh
  - Curves and Curved Surface
- Part 5 CG techniques
  - Shadows
  - Ray-tracing

References

- Ed Angel, CS/EECE 433 Computer Graphics, University of New Mexico
- Steve Marschner, CS4620/5620 Computer Graphics, Cornell
- Tom Thorne, COMPUTER GRAPHICS, The University of Edinburgh
- Elif Tosun, Computer Graphics, The University of New York

- Questions?