Computer Graphics
Introduction

AMARTYA KUNDU DURJOY
LECTURER
CSE DEPARTMENT (UGV)

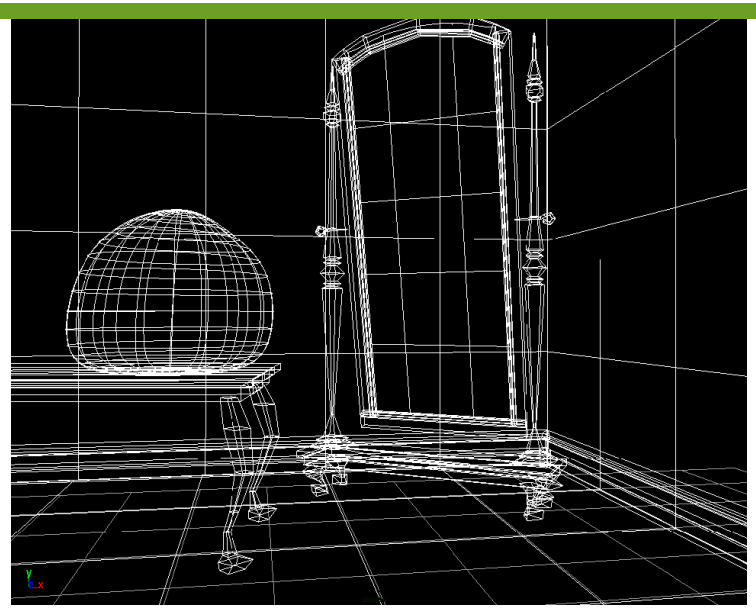
Computer Graphics

- ~ is concerned with the creation and manipulation of **synthetic** images and animations using the aid of computer.

Computer graphics are pictures and movies created using computers – usually referring to image data created by a computer specifically with help from specialized graphical hardware and software.

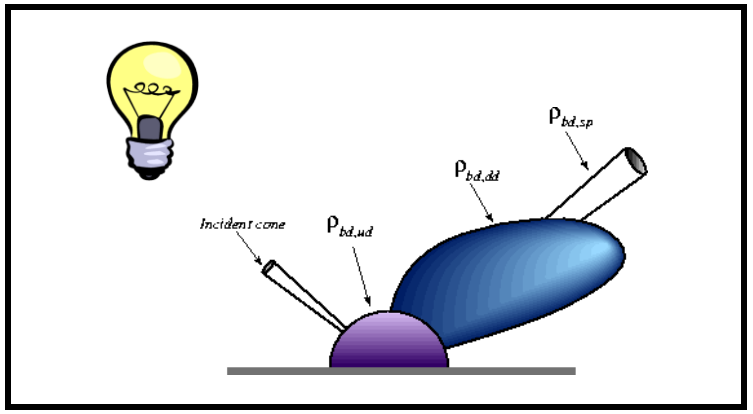
- wikipedia

Computer graphics?



3D geometry

projection



physics

Simulati
on

Computer Graphics

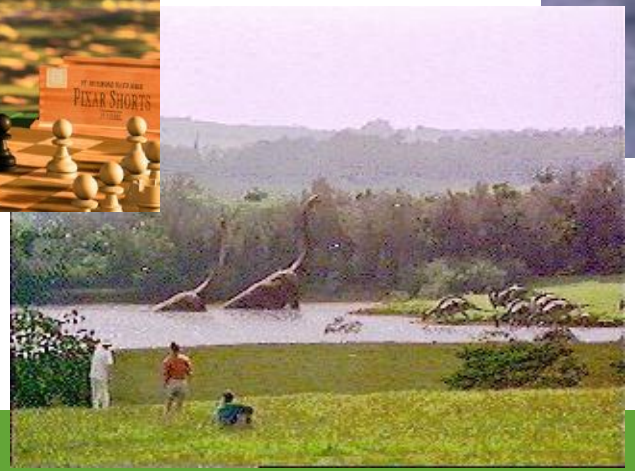
- Imaging
  - Representing 2D images
- Modeling
  - Representing 3D objects
- Rendering
  - Constructing 2D images from 3D models
- Animation
  - Simulating changes over time

Why Study Computer Graphics?

- Graphics is cool
  - I like to see what I'm doing
  - I like to show people what I'm doing
- Graphics is interesting
  - Involves *simulation*, AI, *algorithms*, *architecture*…
- I'll never get an Oscar for my acting
  - But maybe I'll get one for my CG special effects
- Graphics is fun

What drives computer graphics?

- Movie Industry
  - Movie production, Animations,
    Special Effect (super reality)
  - Big budgets and tight schedules
  - Defines our expectations

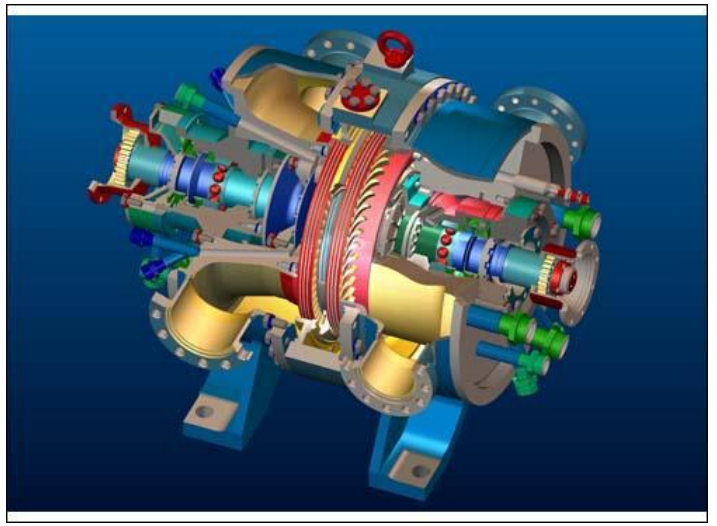


What drives computer graphics?

- Game Industry
  - The newest driving force in CG
    - Why? Volume and Profit
    - This is why we have commodity GPUs
  - Focus on interactivity
  - Cost effective solutions

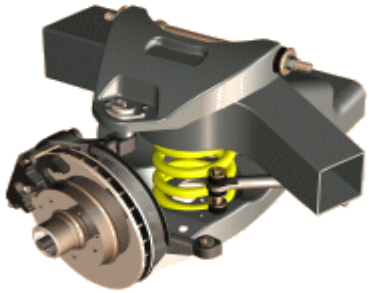
What drives computer graphics?

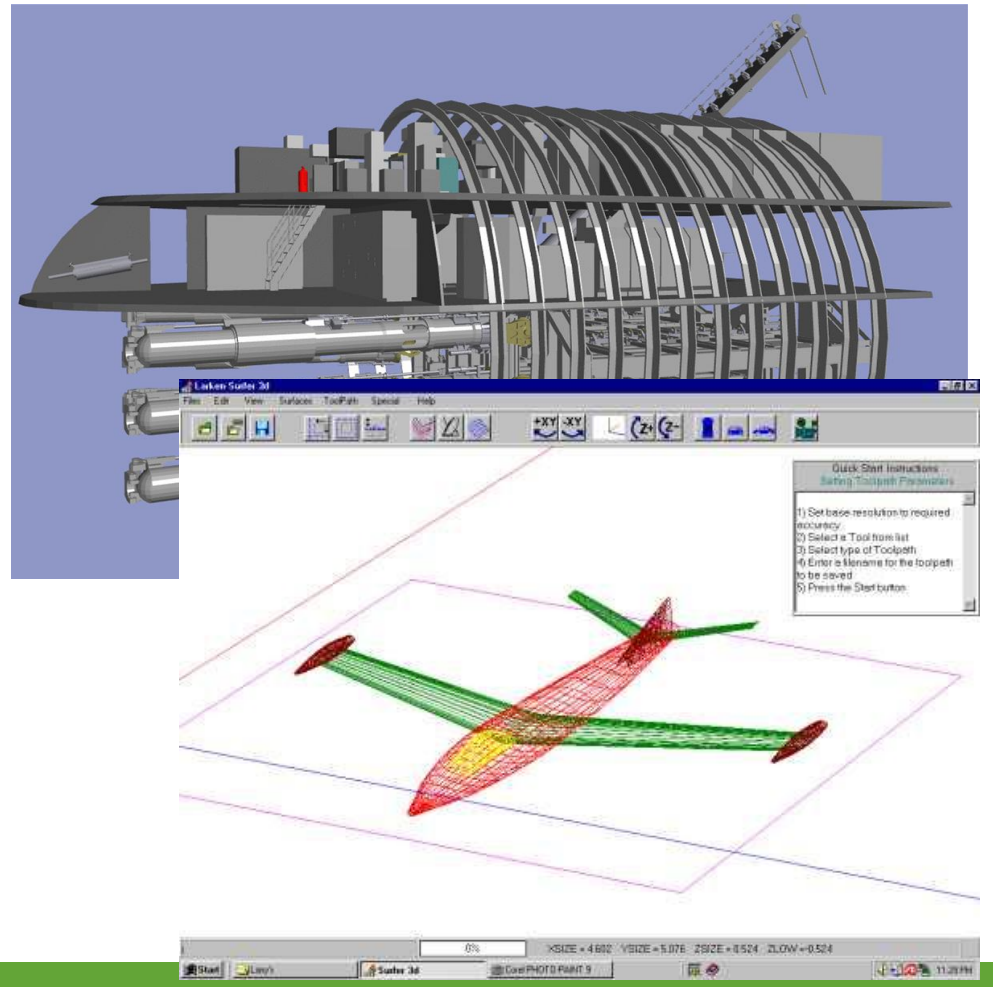
- Computer Aided Design
  - Structural design – design of buildings, bridges, car, aircraft, many more
  - Electrical circuit, mechanical design
  - Integration of computing and display resources
  - Reduced design cycles == faster systems, sooner



Graphics Applications

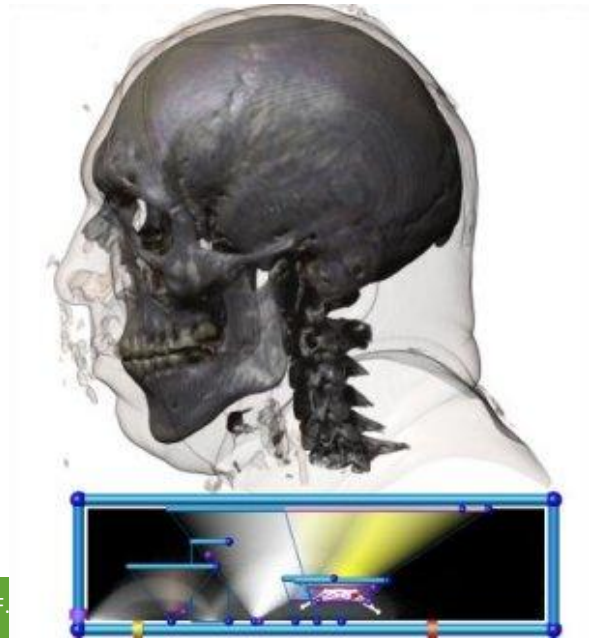
- Computer Aided Design (CAD)
  - Better conceptualization, interaction, transfer of ideas

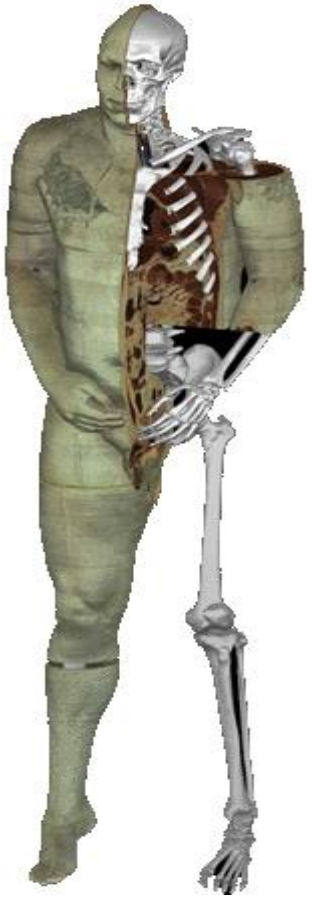




Graphics Applications

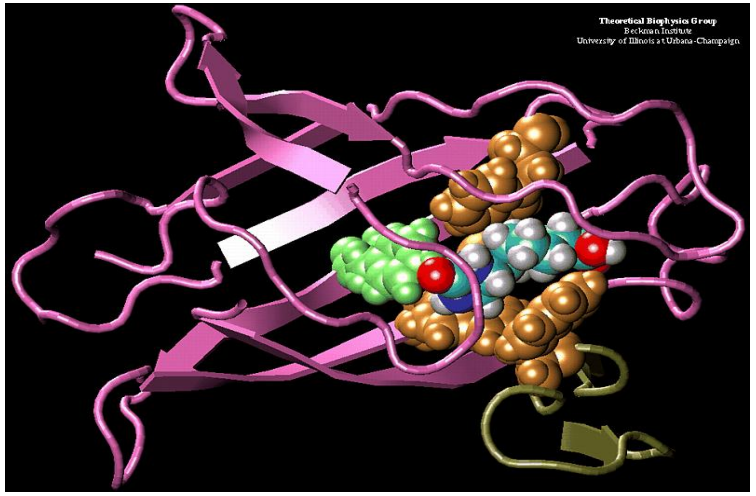
- Medical Visualization
  - Tools for visualizing diagnosis
    - No cheating or tricks allowed
  - Drive issues of precision and correctness
  - Focus on presentation and interpretation of data
  - Construction of models from acquired data

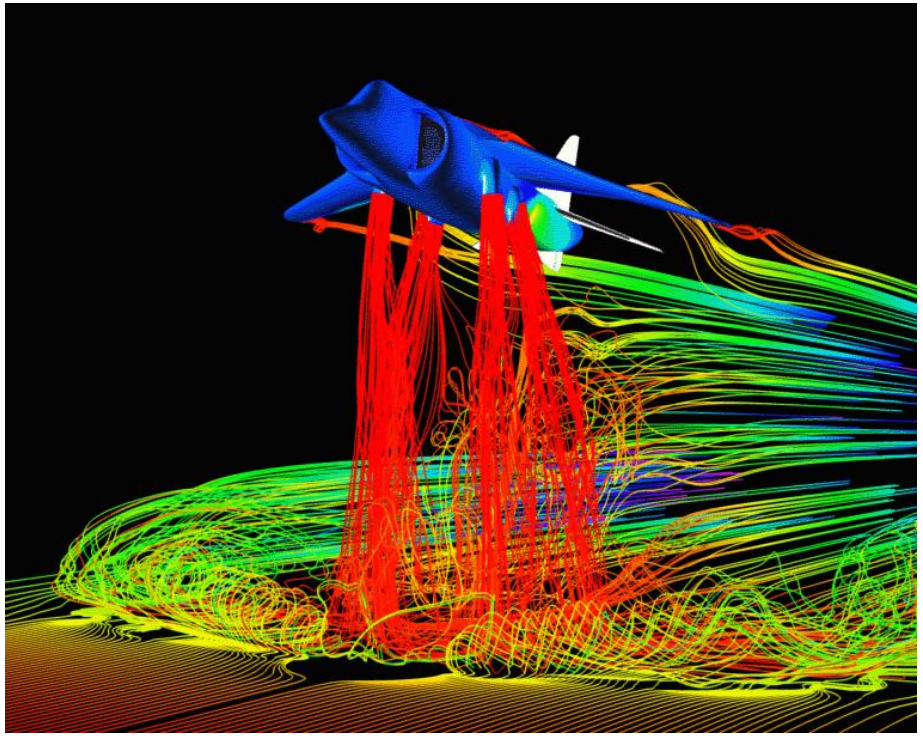


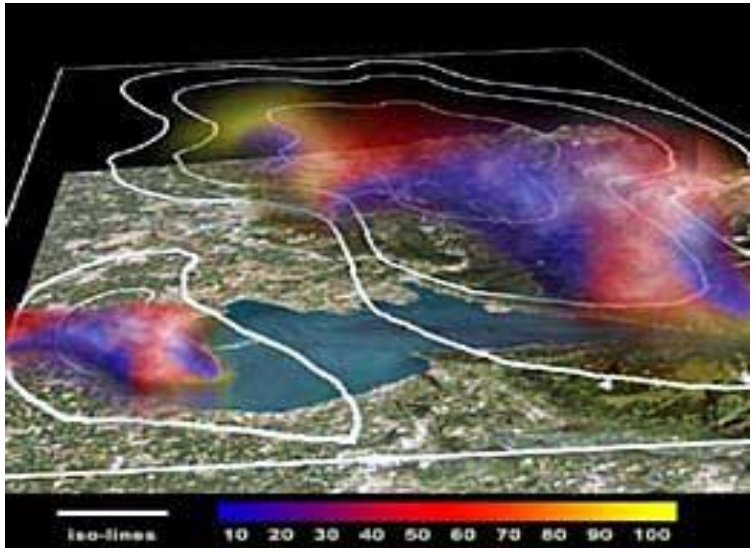


Graphics Applications

- Scientific Data Visualization
  - CG is superb tool for presenting complex scientific data in a way that can be easily grasped







Airflow around a Harrier Jet *(NASA Ames)*

Computer Graphics - Applications

- Education and Training
  - Computer generated model can aid easy understanding (education)
  - Various simulator can be used for practice / training session for aircraft pilot, heavy equipment operators.
- Computer Art, publishing
  - Fine and commercial art
  - Slide, Book, Magazine design

Computer Graphics - Applications

- Computer Vision and Image processing
  - IP - Improving Picture quality
  - CV - Machine perception of visual info
  - CG can help on
- Process control, Cartography, GUI

What's Our Scope?

- Not a Tutorial on Commercial Software
  - 3DMax, Maya, Photoshop, etc.
- Not about Graphics Business
  - 3D online-game, E-commerce, etc.
- Graphics = Algorithm for Visual Simulation
  - Modeling, Rendering, Animation

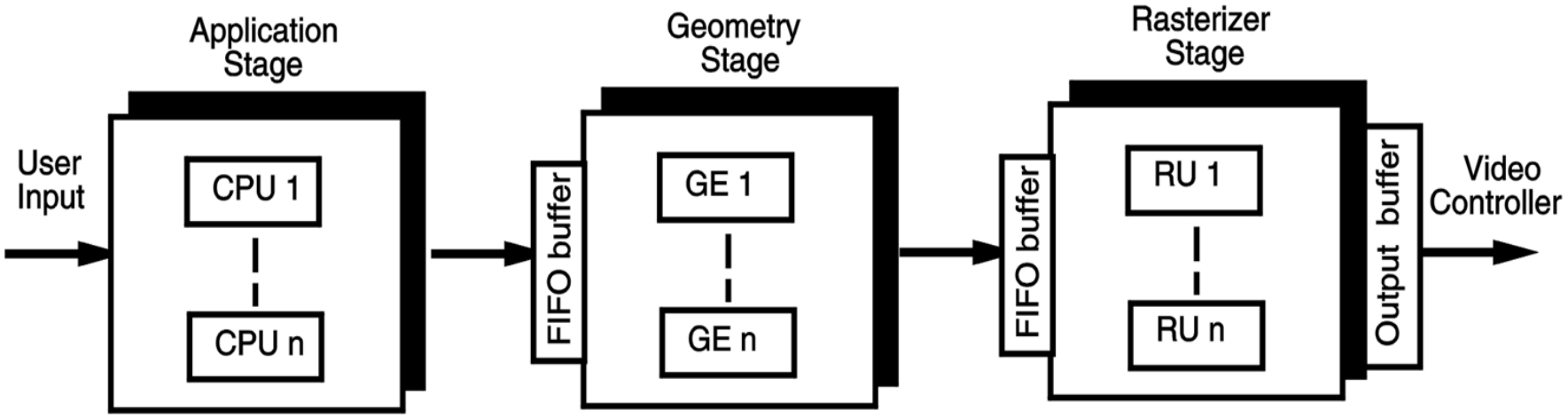
Computer Graphics

- Imaging
  - Representing 2D images
- Modeling
  - Representing 3D objects
    - Surface Modeling
    - Solid Modeling
    - Procedure Modeling
    - Simplification
- Rendering
  - Constructing 2D images from 3D models
    - Polygonal Rendering
    - Global Rendering
    - Texture Mapping
- Animation
  - Simulating changes over time
    - Keyframe Animation
    - Inverse Kinematics
    - Dynamics
    - Motion Capture

Graphics / Rendering Pipeline

There are three stages

- Application Stage
- Geometry Stage
- Rasterization Stage



Application stage

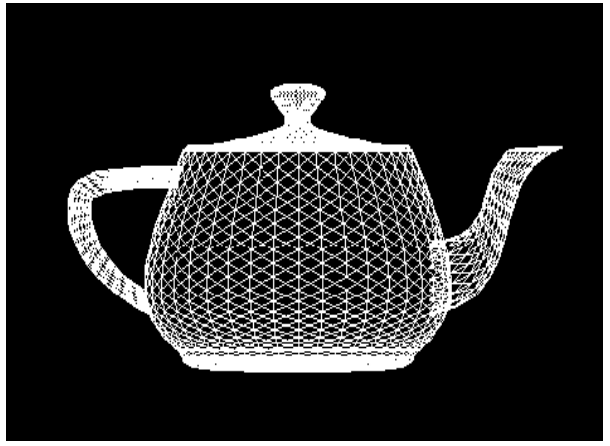
Entirely done in software by the CPU

Read Data

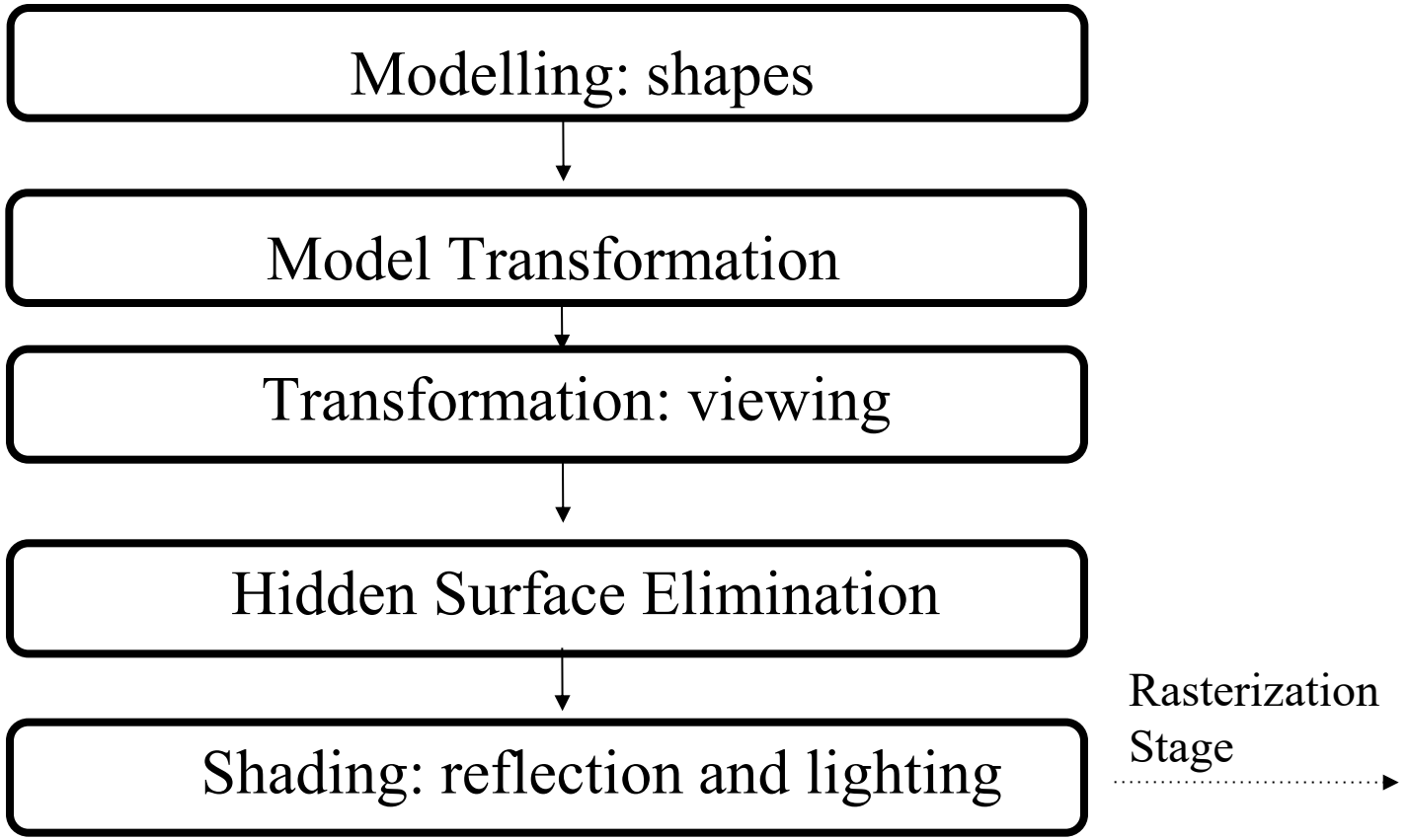
the world geometry database,

User's input by mice, trackballs, trackers, or sensing gloves

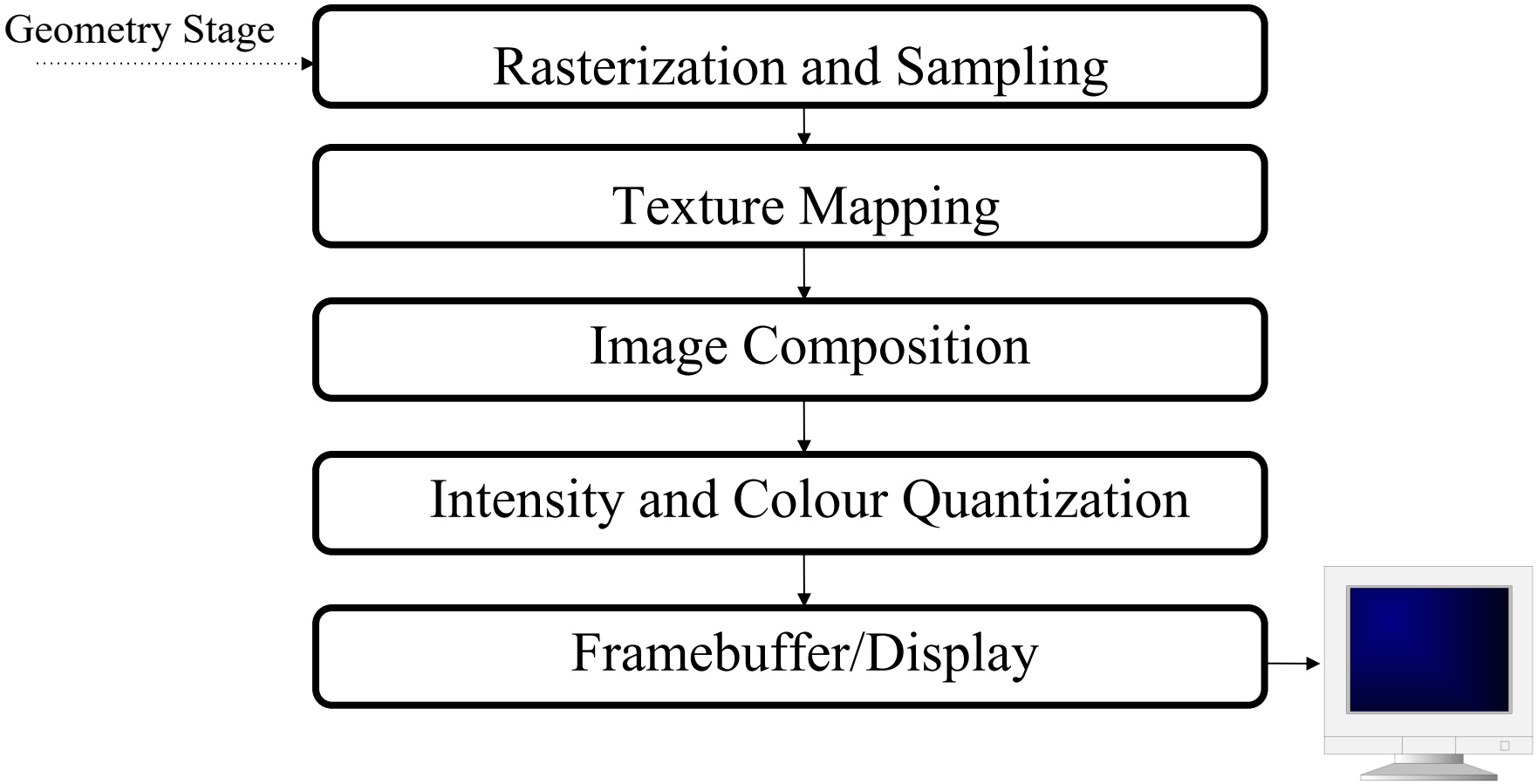
In response to the user's input, the application stage change the view or scene



Geometry Stage

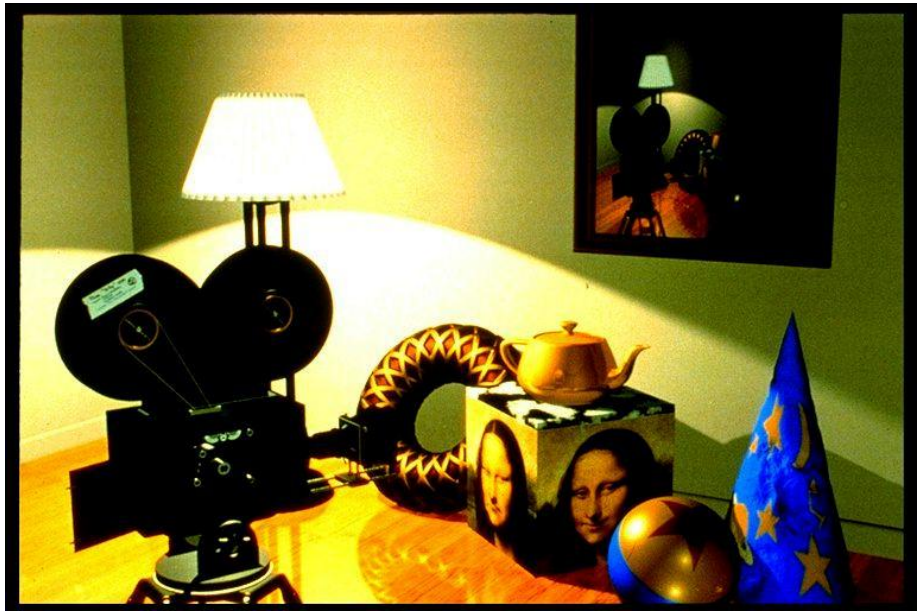


Rasterization Stage



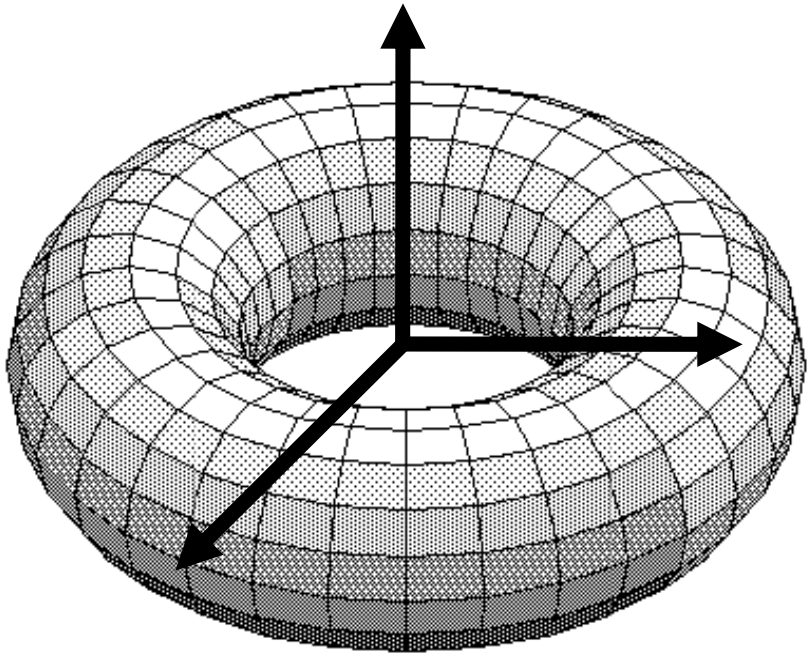
An example thro’ the pipeline…

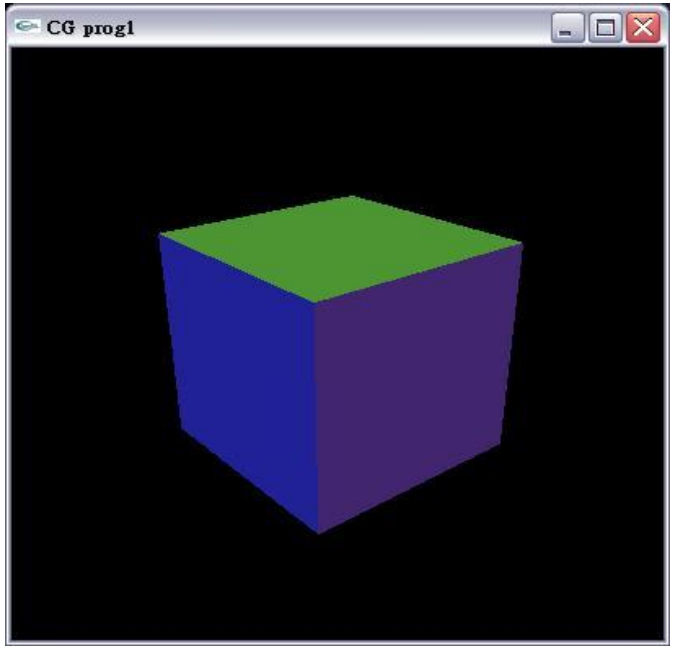
The scene we are trying to represent:



Preparing Shape Models

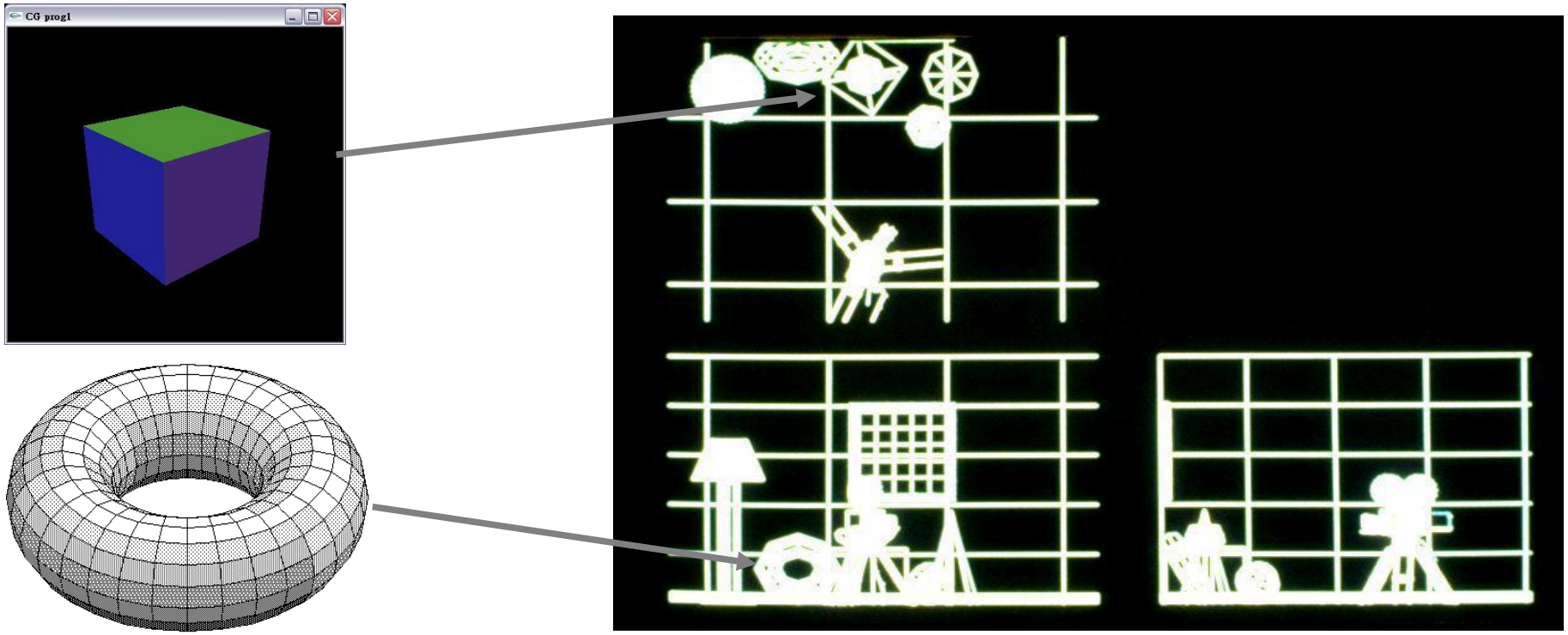
- Designed by polygons, parametric curves/surfaces, implicit surfaces and etc.
- Defined in its own coordinate system





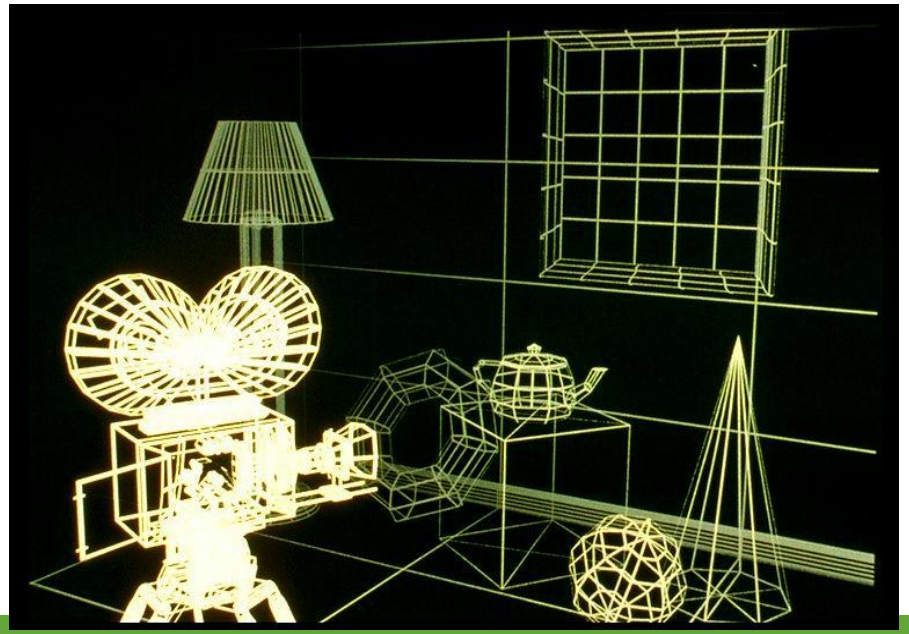
Model Transformation

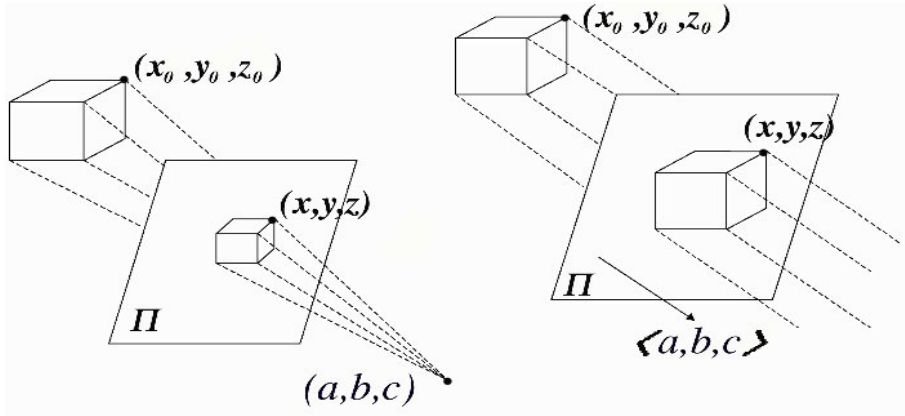
- ✓ Objects put into the scene by applying translation, scaling and rotation
- ✓ The location of all the vertices are updated by this transformation

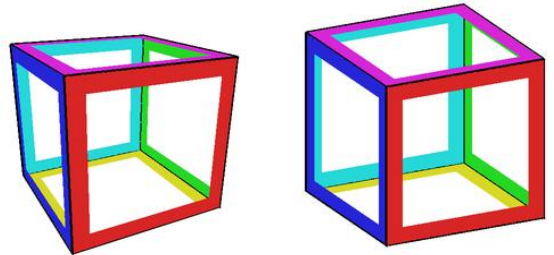


Perspective Projection

- 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

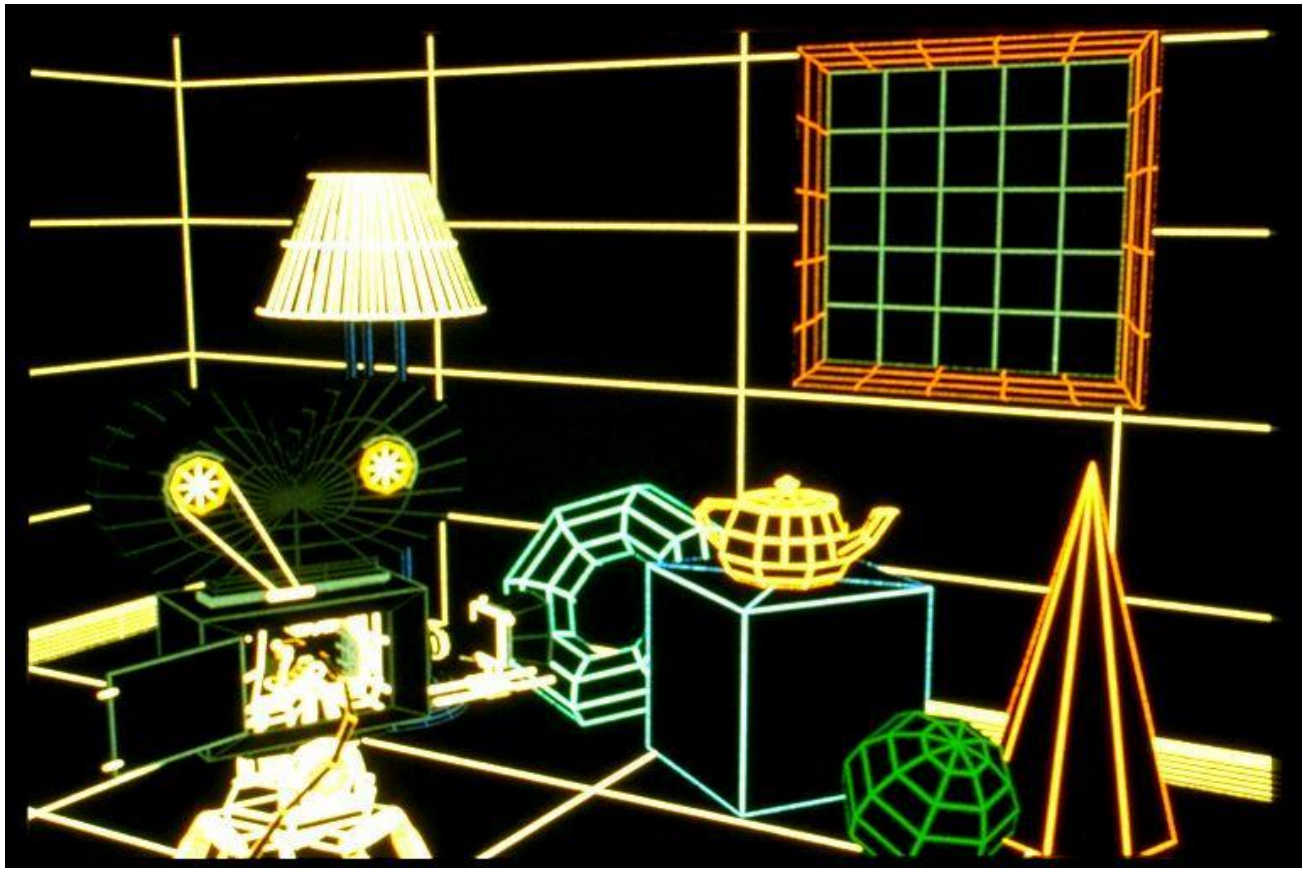






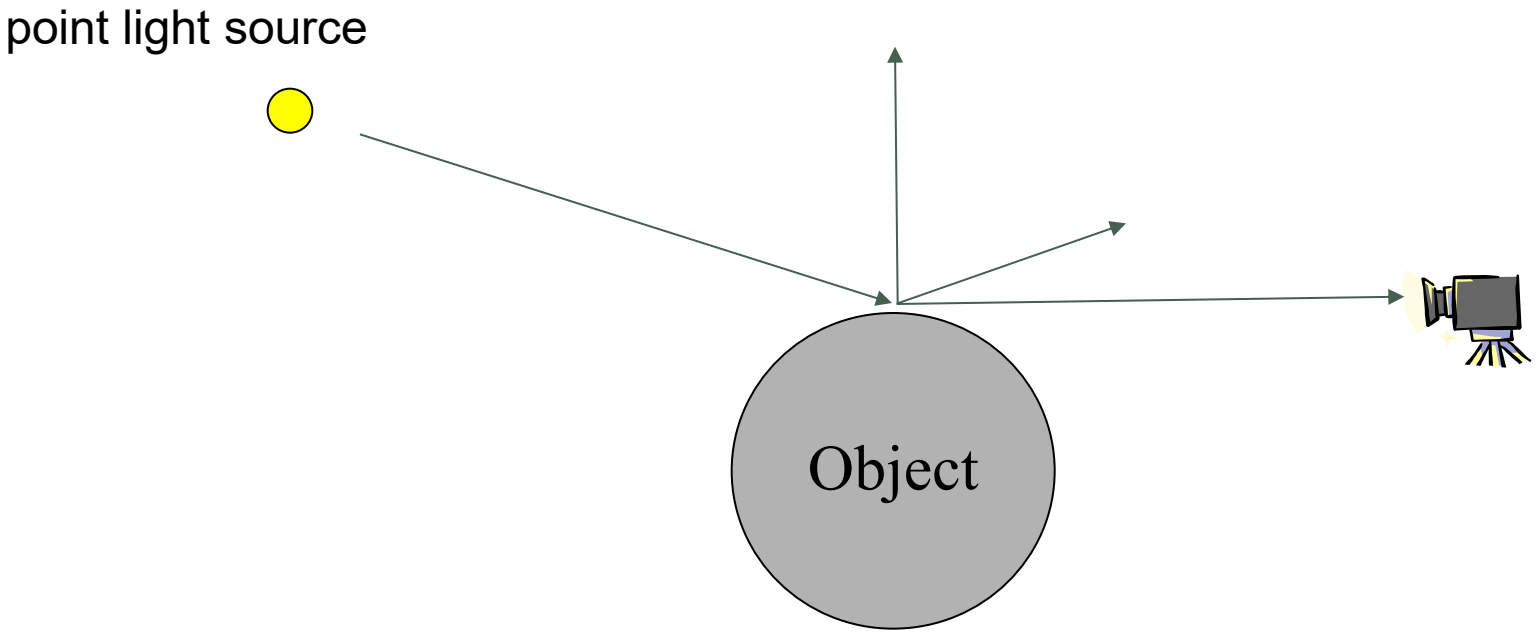
Hidden Surface Removal

- Objects occluded by other objects must not be drawn



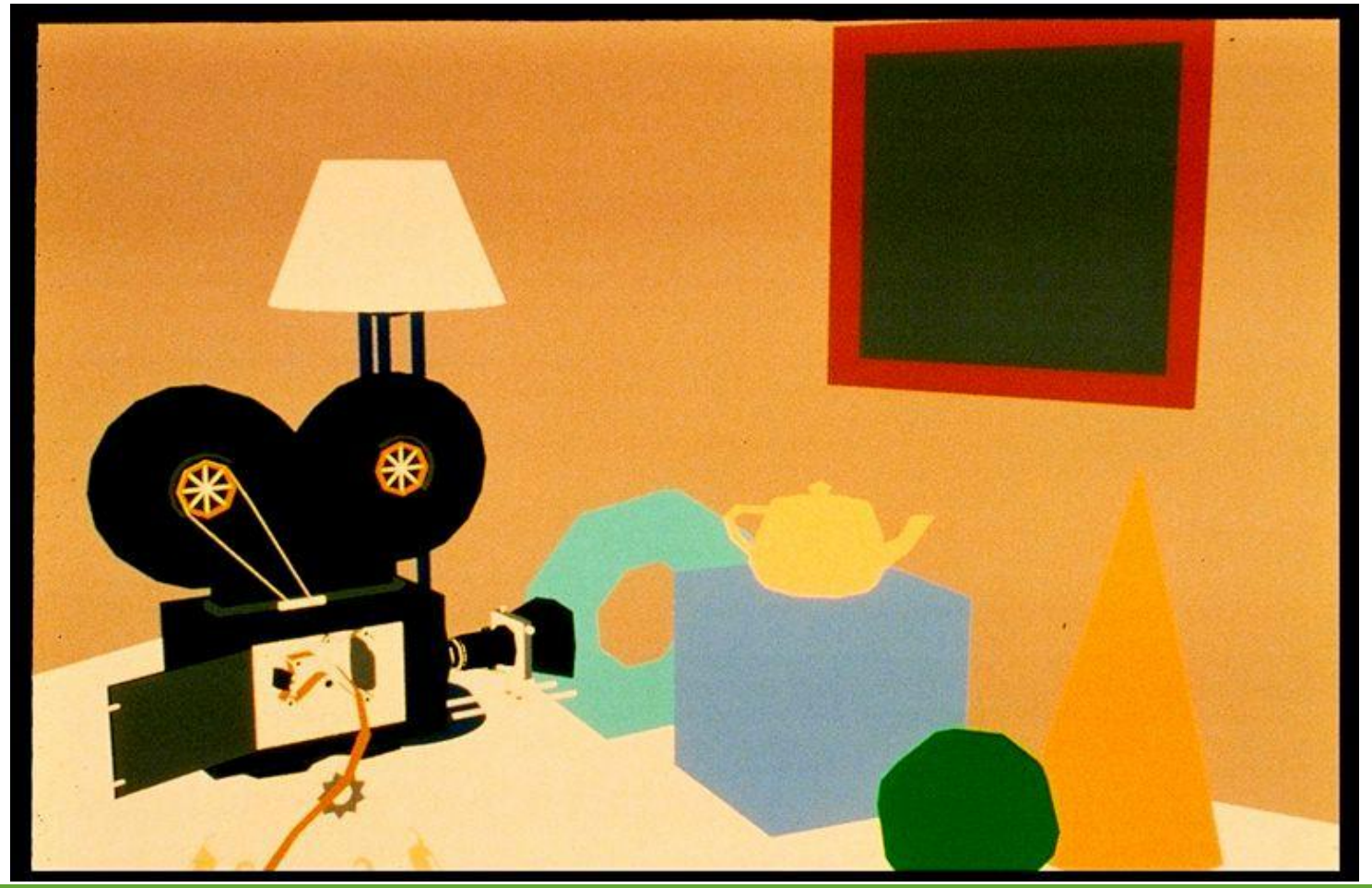
Shading

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



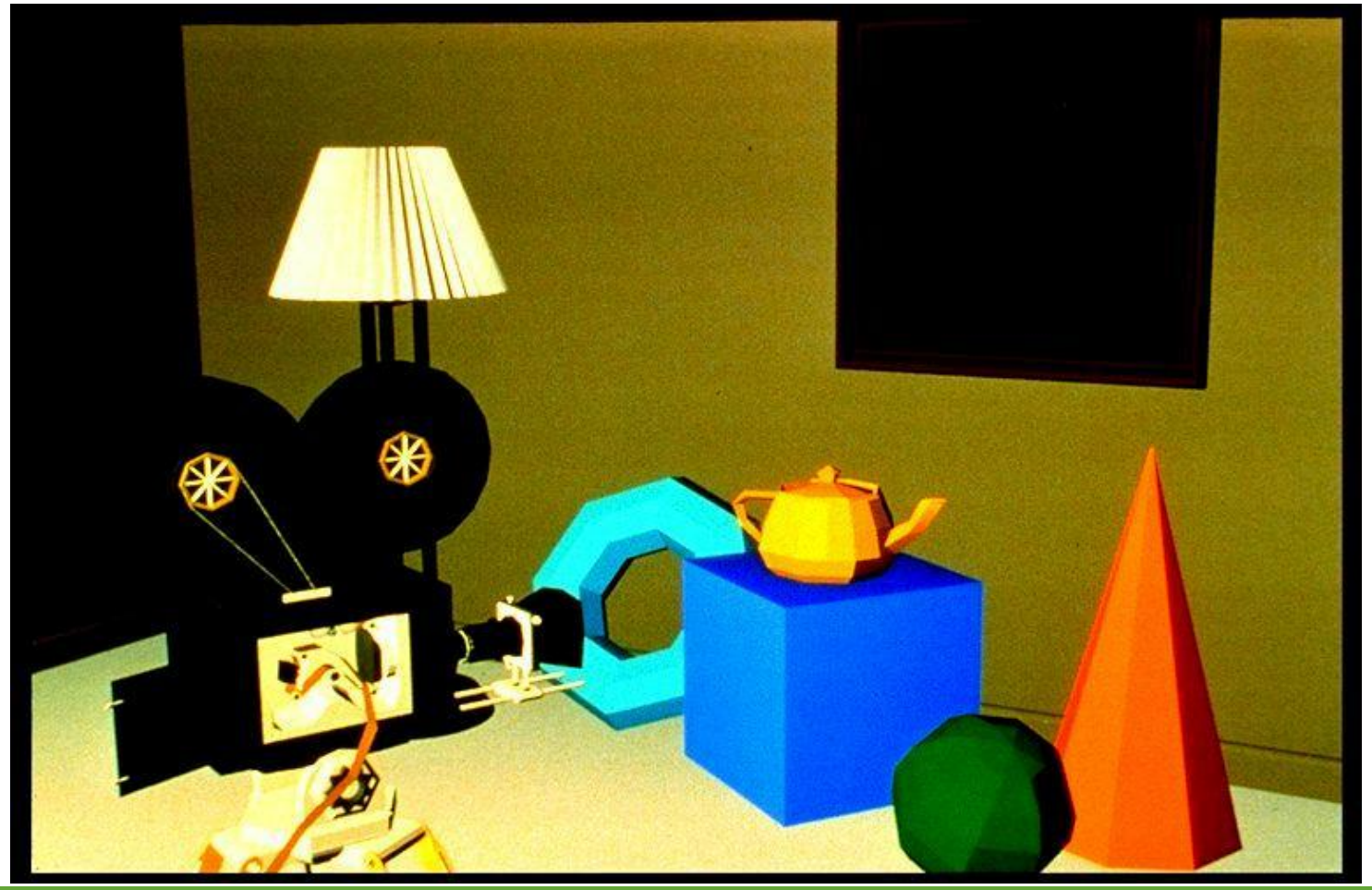
Shading : Constant Shading - Ambient

- Objects colours by its own colour



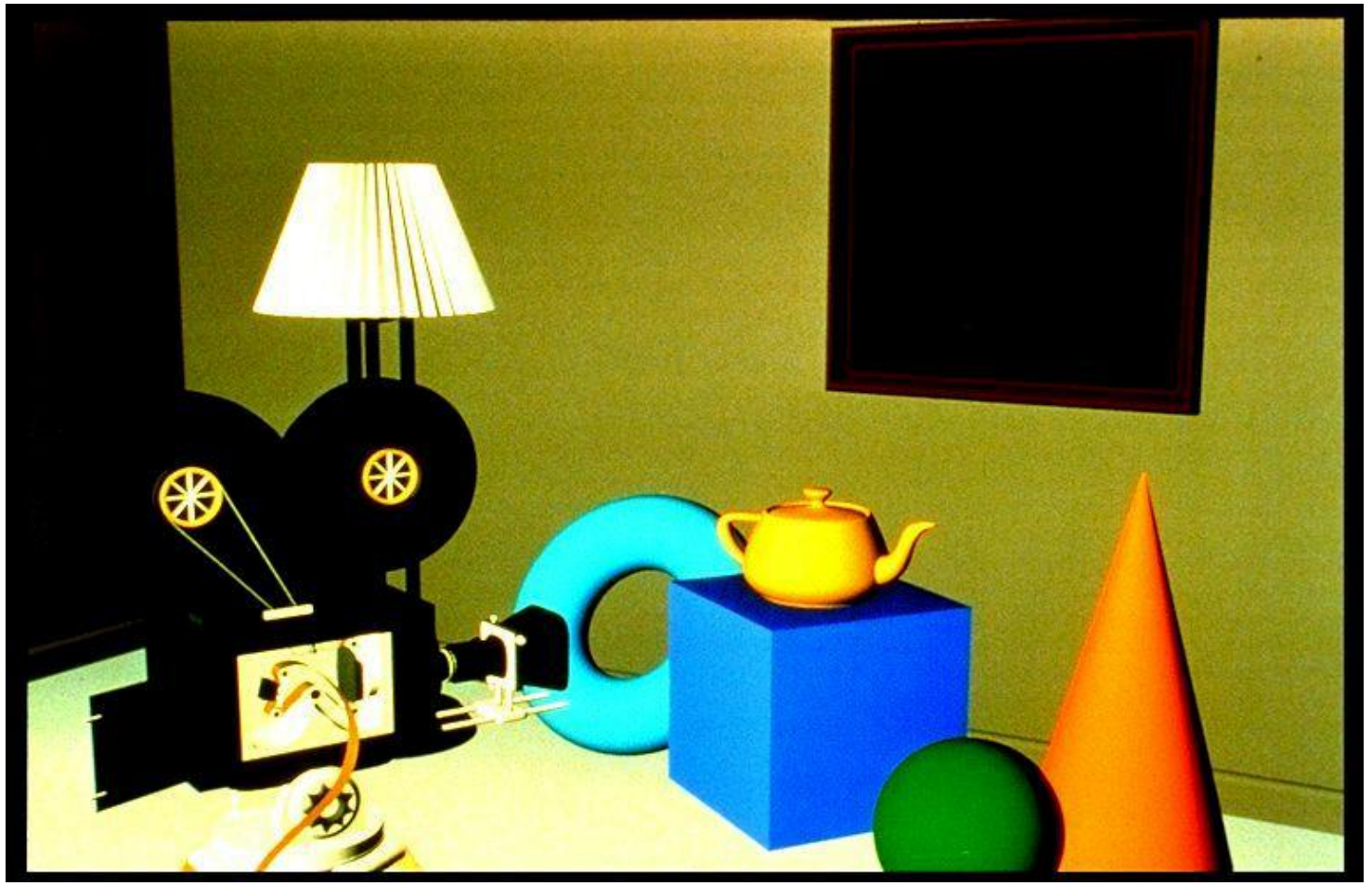
Shading – Flat Shading

- Objects coloured based on its own colour and the lighting condition
- One colour for one face

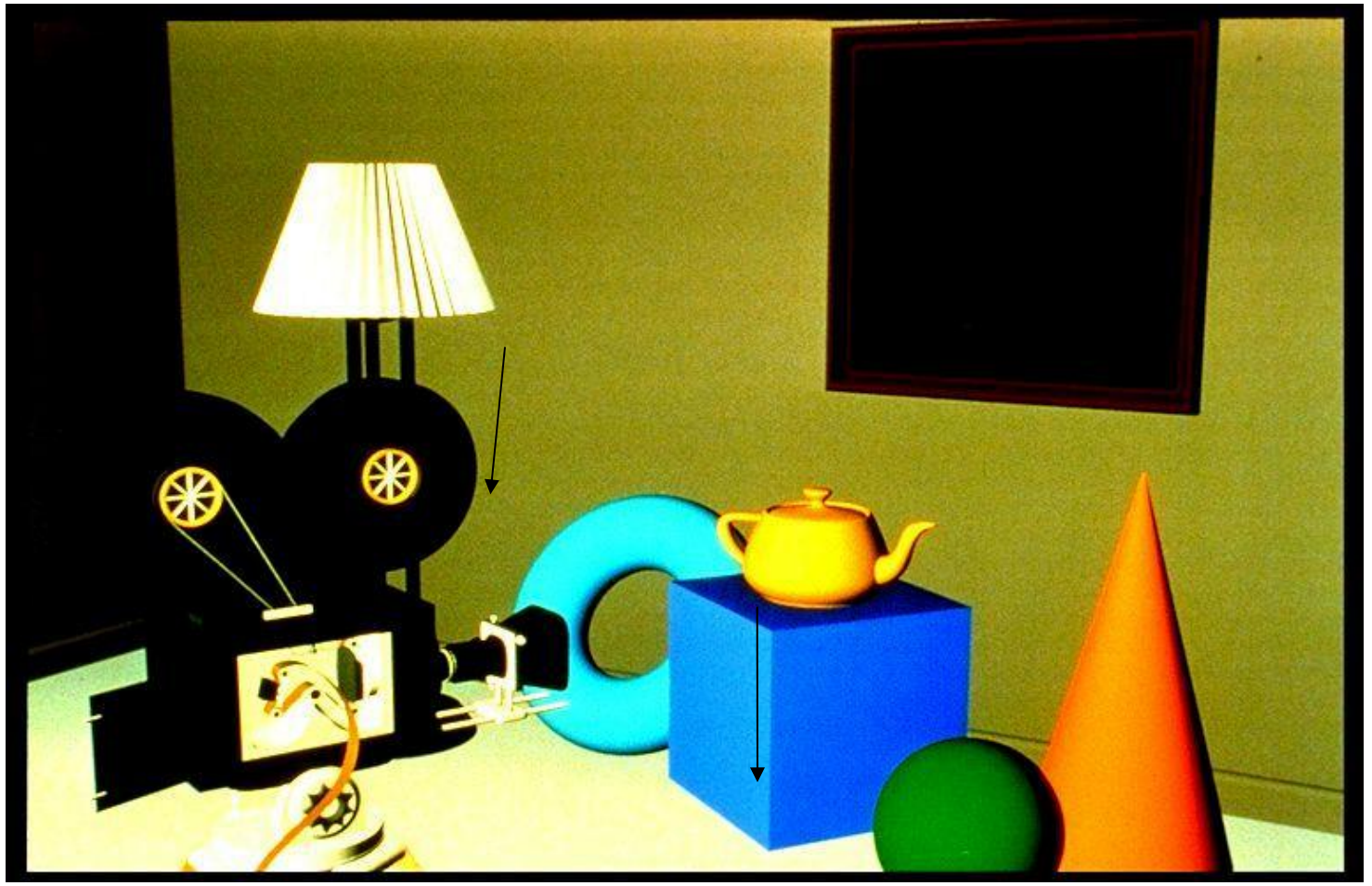


Gouraud shading, no specular highlights

- Lighting calculation per vertex

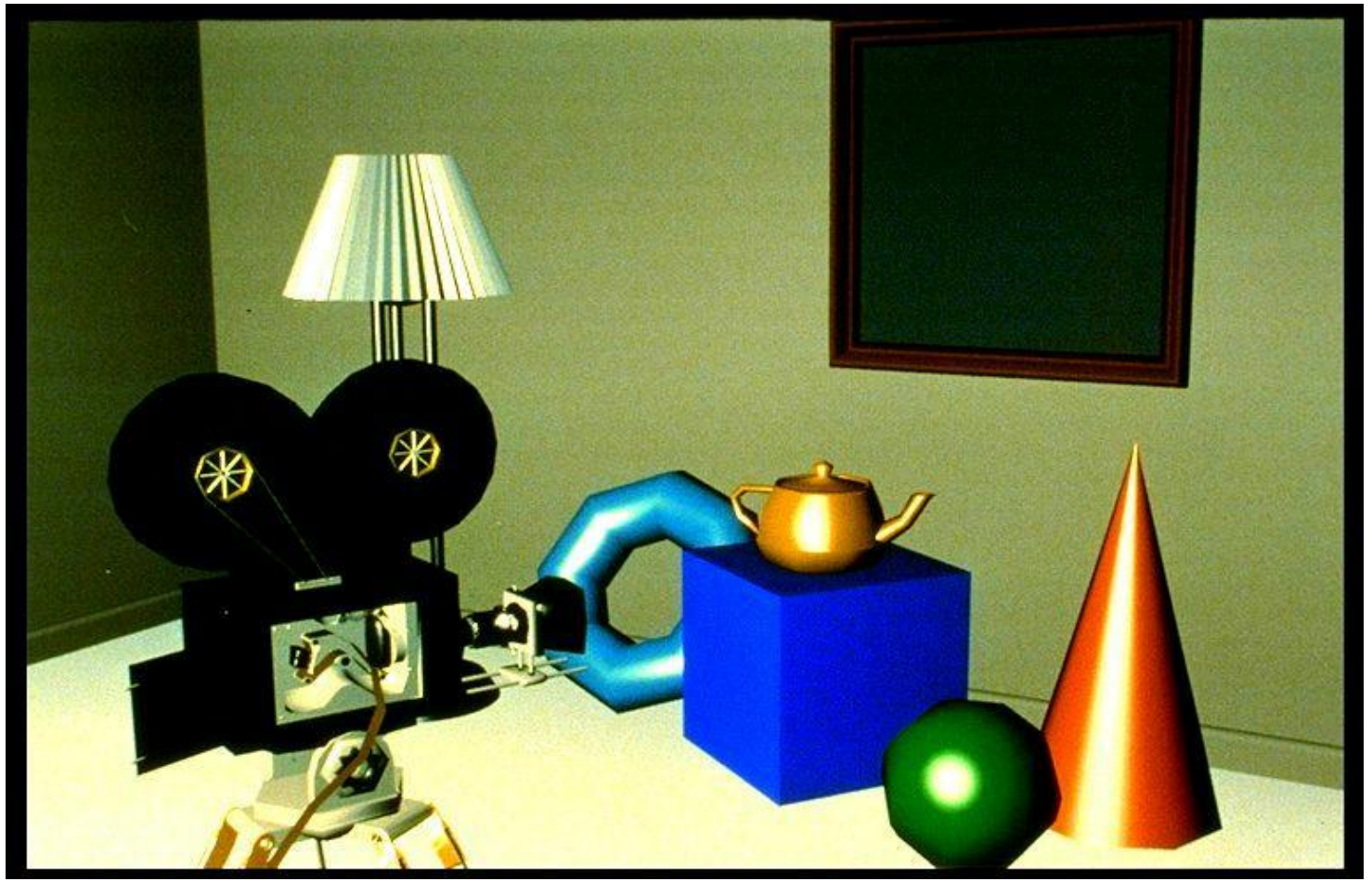


Shapes by Polynomial Surfaces

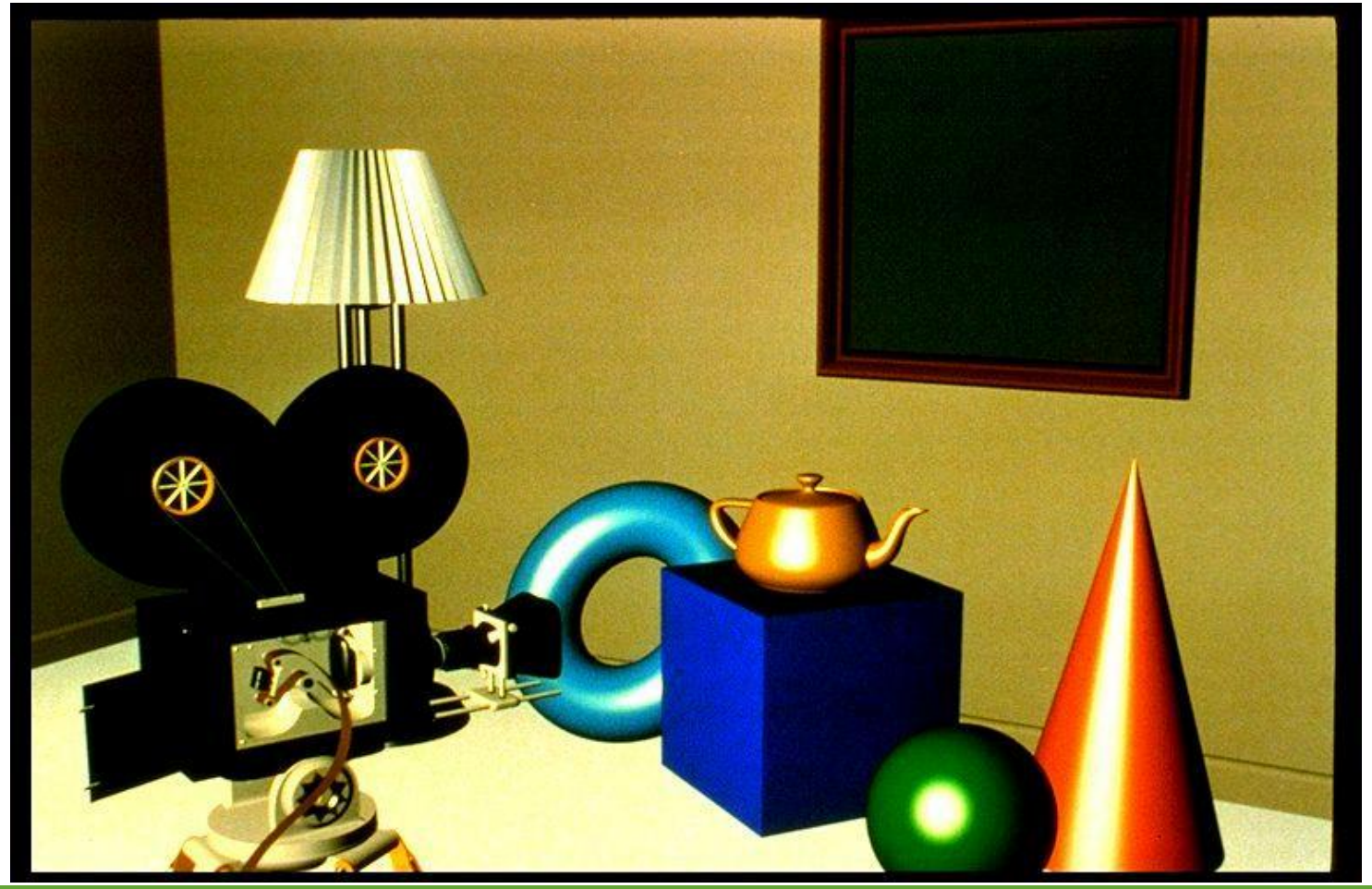


Specular highlights added

- Light perfectly reflected in a mirror-like way



Phong shading

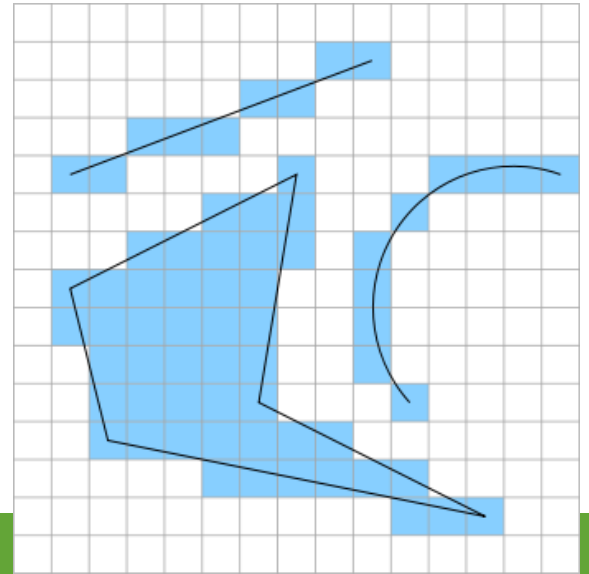


Rasterization

Converts the vertex information output by the geometry pipeline into pixel information needed by the video display

Aliasing: distortion artifacts produced when representing a high-resolution signal at a lower resolution.

Anti-aliasing : technique to remove aliasing

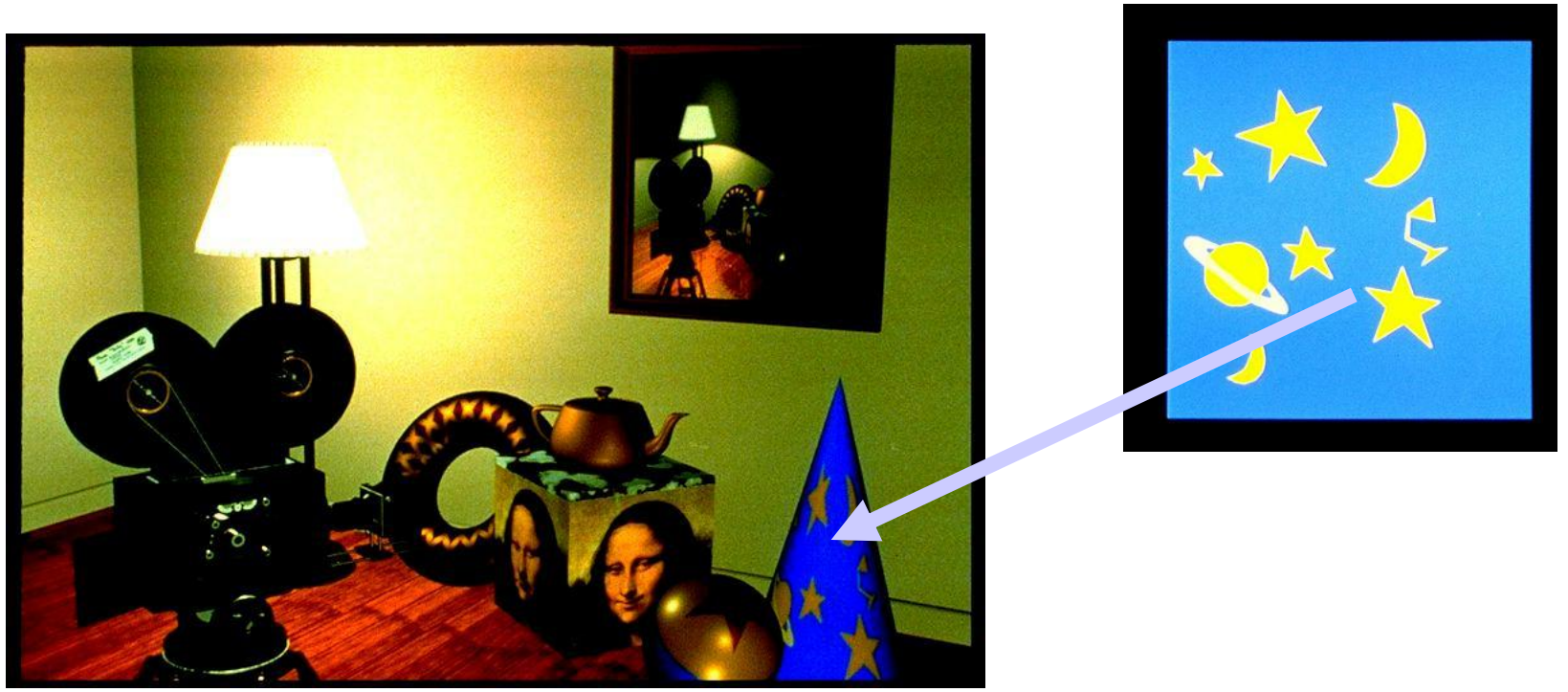


Anti-aliasing

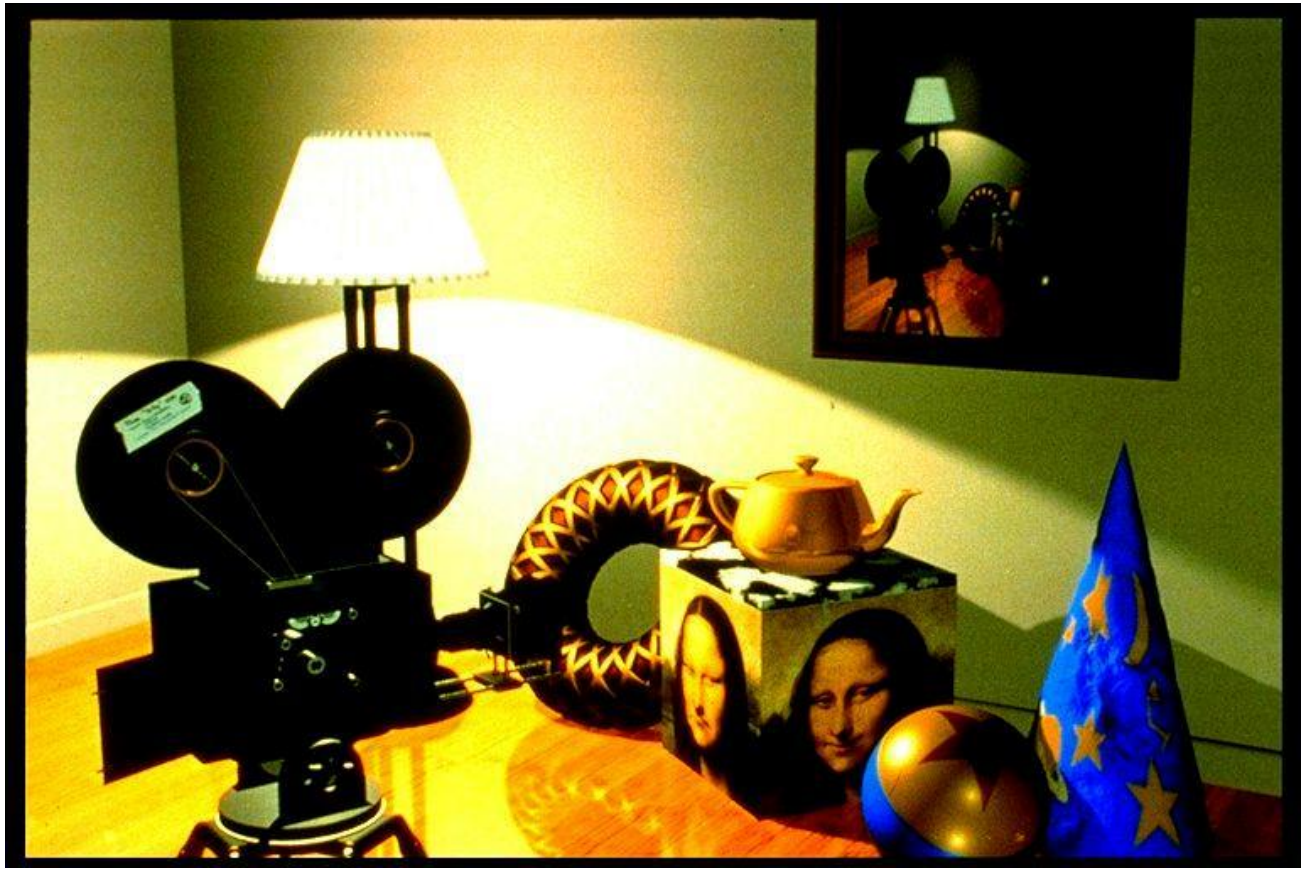
Aliased polygons (jagged edges)

Anti-aliased polygons

Texture mapping

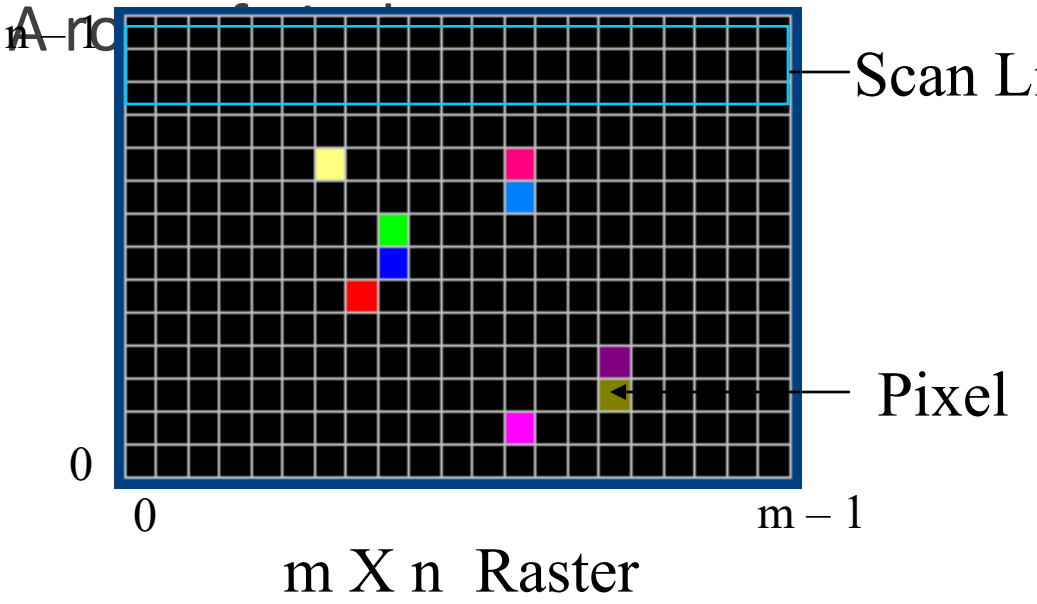


Reflections, shadows & Bump mapping



Graphics Definitions

- Most common display used for CG are Raster displays
  - CRT, Flat panel
- Display surface : On which image is presented
- RASTER : A rectangular array of points or dots on the surface
- PIXEL : One dot or picture element of the raster
- SCAN LINE : A ro



We don't have any pixel like (1.2, 5.8). In raster device co-ordinates can take integer values only.

Graphics Definitions

Vertex

point in 3D

Edge

line in 3D connecting two vertices

Polygon/Face/Facet

arbitrary shape formed by connected vertices

fundamental unit of 3D computer graphics

Mesh

set of connected polygons forming a surface (or object)

:

Graphics Definitions

- The definition of image is stored in a 2 dimensional array in memory called Frame Buffer
- The display hardware produces the image line by line (Raster Lines)
- The high level process of taking a graphics model and producing an image is called Rendering
- The lower level process of drawing individual pixels or filling polygonal regions of the image by modifying the frame buffer is known as Scan Conversion or Rasterization

Frame Buffer

- a video output device that drives a video display from a memory containing the color for every pixel
- A frame buffer may be thought of as computer memory organized as a two-dimensional array with each (x,y) addressable location corresponding to one pixel.
- Bit Planes or Bit Depth is the number of bits corresponding to each pixel.
- A typical frame buffer resolution might be
  - 640 x 480 x 8 (x-resolution x y-resolution x color bit depth)
  - 800 x 600 x 16
  - 1024 x 768 x 24

Conclusion

- Computer graphics: generating 2D synthetic images of a 3D world represented in a computer.
- Main tasks:
  - *modeling*: (shape) creating and representing the geometry of objects in the 3D world
  - *rendering*: (light, perspective) generating 2D images of the objects

Recommended Texts

1. **Computer Graphics**
   - Donald Hearn, M Pauline Baker
2. **Computer Graphics**
   - R. Plastock, Zhigang Xiang
   - (Schaum's Outline Series) McGraw Hill,
3. **Computer Graphics using OpenGL**
   - F S Hill
4. **Computer Graphics: Principles & Practice**
   - J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes
5. **OpenGL Programming Guide**
   - M. Woo, J. Neider, T. Davis, D. Shreiner

- Fundamentals of Computer Graphics
  - Steve Marschner, Peter Shirley
- Mathematics for Computer Graphics
  - John Vince
- Mathematics for 3D Game Programming and Computer Graphics
  - Eric Lengyel