Graphics

CMSC 100

Penny Rheingans
University of Maryland Baltimore County

Where have you seen computer graphics?

Special Effects

- Uses for graphics

- Characteristics

Computer/video Games

- Uses for graphics

- Characteristics

Computer Graphics

- Using computer to generate simulated scenes or worlds
- Can require tricking eye to believe 2D collection of pixels is really a continuous 3D world
- Coding-intensive application with strong basis in creativity and human perception

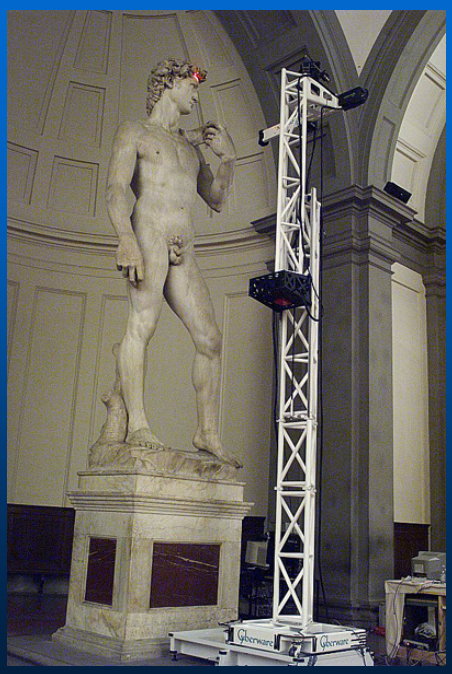
- Five key problems
  - What shape is it?
  - What do you see?
  - What does it look like?
  - How does it move?
  - Why does it have to look like a photograph?

What shape is it?

Modeling Approaches

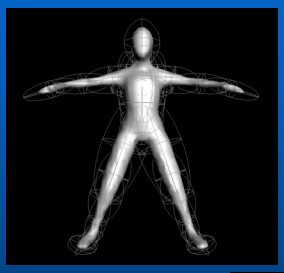
- Modeling problem
  - Define shape, color, and other visual properties
- Modeling solutions
  - Manual primitive creation
  - Scans from physical object
  - Functional descriptions
  - Grammar-based generation
  - Biologically-inspired simulations

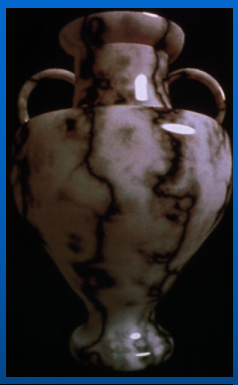
Scanning

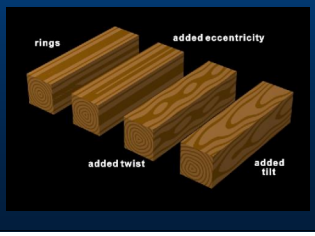


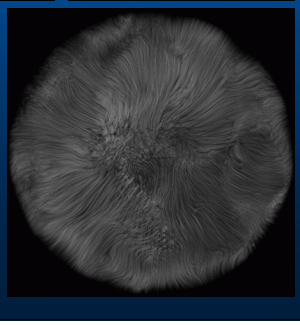
Functional Descriptions

- Define visual attributes with function, defined over space
  - Shape
  - Density
  - Color



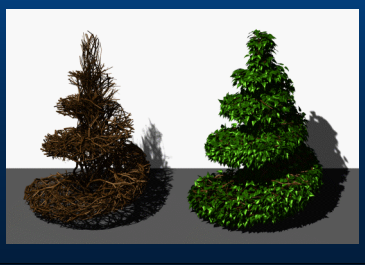






Grammar-based Generation

- Use (mostly) context-free grammars (CFG) to specify structural change over generations
- A CFG G=(V,T,S,P) with
  - V is a set of non-terminals
  - T is a set of terminals
  - S is the start symbol
  - P is a set of productions (rules) of the form:
    - » $A \rightarrow x$, where $A \in V$, $x \in (V \cup T)^*$



Biological Simulations

Mimic developmental process:

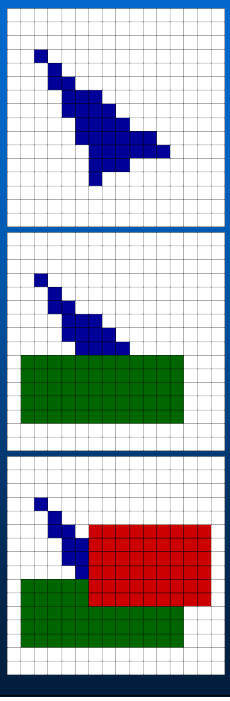
- cellular automata
- reaction diffusion

What do you see?

Painter's Algorithm

- Basic approach
  - Draw polygons, from farthest to closest

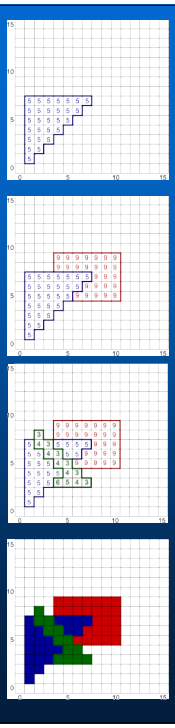
- Given
  - List of polygons {P_1, P_2, …. P_n)
  - An array of Intensity [x,y]
- Begin
  - Sort polygon list on minimum Z (largest z value comes first in sorted list)
  - For each polygon P in selected list do
    - For each pixel (x,y) that intersects P do
      - Intensity[x,y] = intensity of P at (x,y)
  - Display Intensity array



Z-Buffer

- Basic approach
  - Draw polygons, remembering depth of stuff drawn so far

- Given
  - List of polygons {P_1, P_2, …., P_n}
  - An array x-buffer[x,y] initialized to +infinity
  - An array Intensity[x,y]
- Begin
  - For each polygon P in selected list do
    - For each pixel (x,y) that intersects P do
      - Calculate z-depth of P at (x,y)
      - If z-depth < z-buffer[x,y] then
        - Intensity[x,y] = intensity of P at (x,y)
        - Z-buffer[x,y] = z-depth
  - Display Intensity array

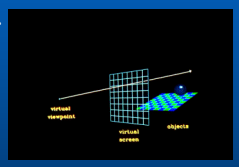


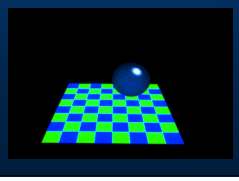
Raytracing

- Basic approach
  - Cast ray from viewpoint through pixels into scene

- Given

```
   List of polygons { P1, P2, ..., Pn }
   An array of intensity [ x, y ]
{
   For each pixel (x, y) {
      form a ray R in object space through
        the camera position C and the pixel
        (x, y)
      Intensity [ x, y ] = trace ( R )
      }
   Display array Intensity
}
```





What does it look like?

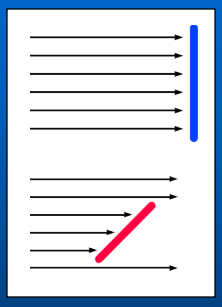
Illumination Approaches

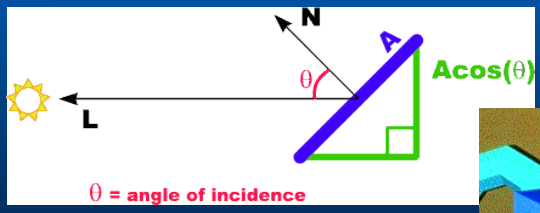
- Illumination problem
  - Model how objects interact with light
- Modeling solutions
  - Simple physics/optics
  - More realistic physics
    - » Surface physics
    - » Surface microstructure
    - » Subsurface scattering
    - » Shadows
    - » Light transport

Simple Optics: Diffuse Reflection

Lambert's Law:

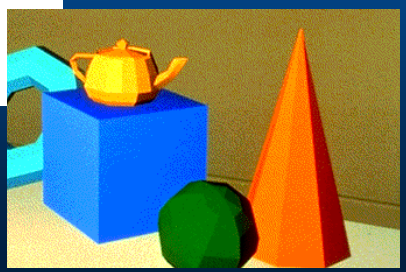
the radiant energy from any small surface area dA in any direction θ relative to the surface normal is proportional to cos θ





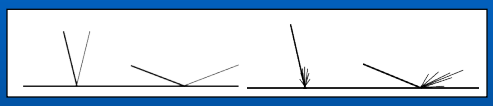
$I_{diff} = k_d I_l \cos \theta$

$= k_d I_l (N \bullet L)$

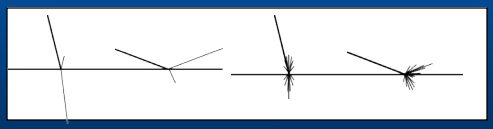


Surface Physics

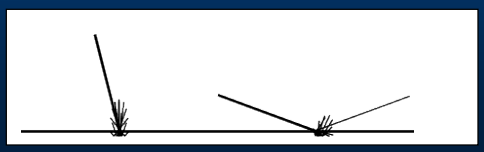
- Conductor (like metal)



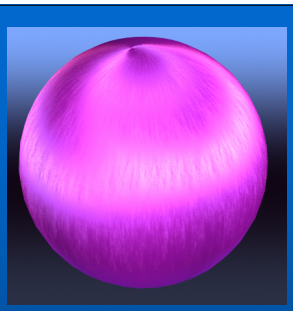
- Dielectric (like glass)

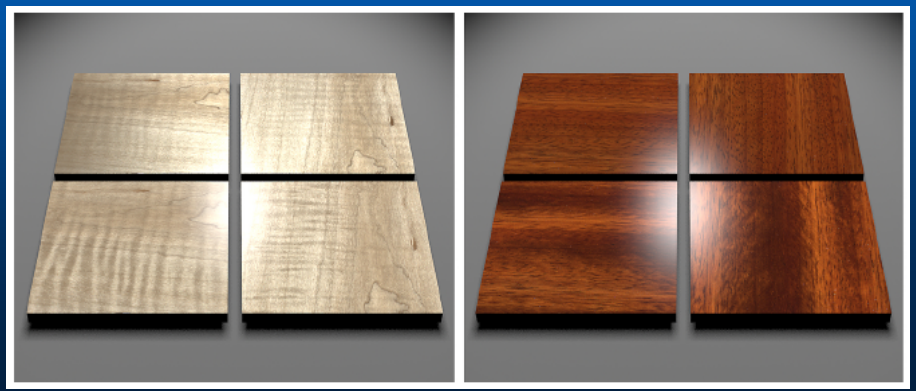


- Composite (like plastic)

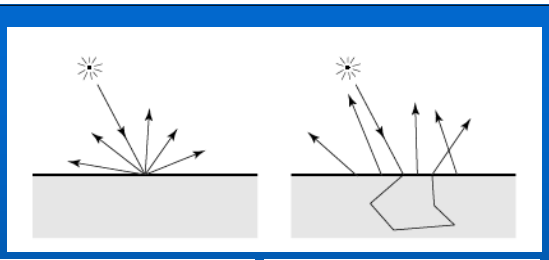


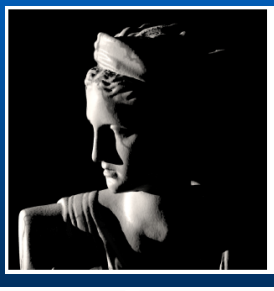
Surface Microstructure

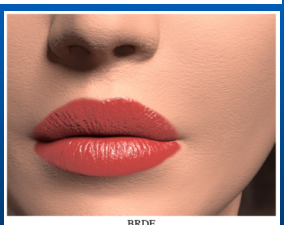




Subsurface Scattering







BRDF

BSSRDF

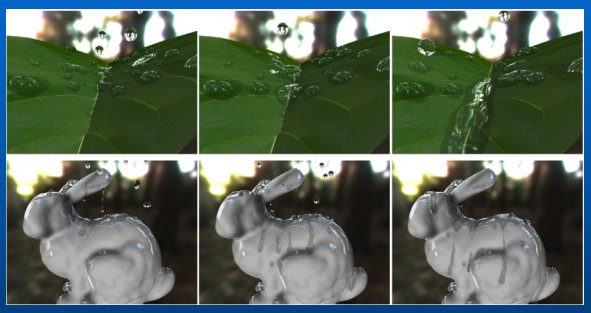
Shadows

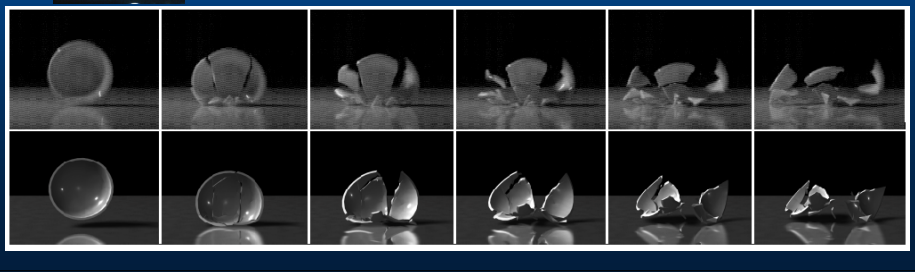
How does it move?

Motion Dynamics Approaches

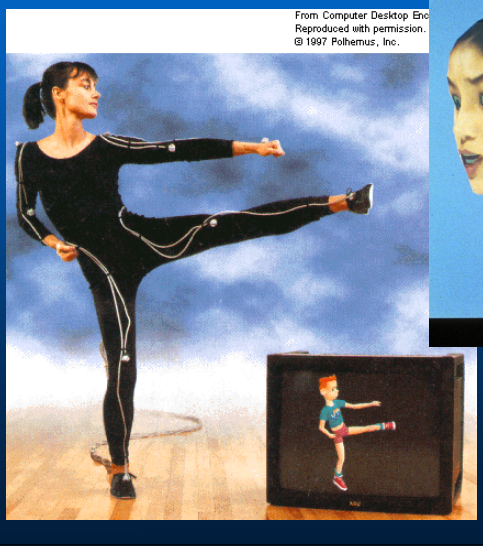
- Motion dynamics problem
  - Define geometric movements and deformations of objections under motion
- Dynamics solutions
  - Simulate physics of simple objects
  - Model structure and constraints
  - Capture motion from reality
  - Simulate group dynamics
  - Use your imagination

Simulate Physics

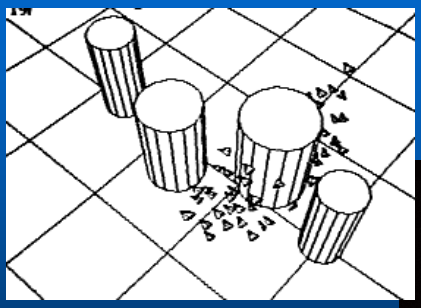




Motion Capture

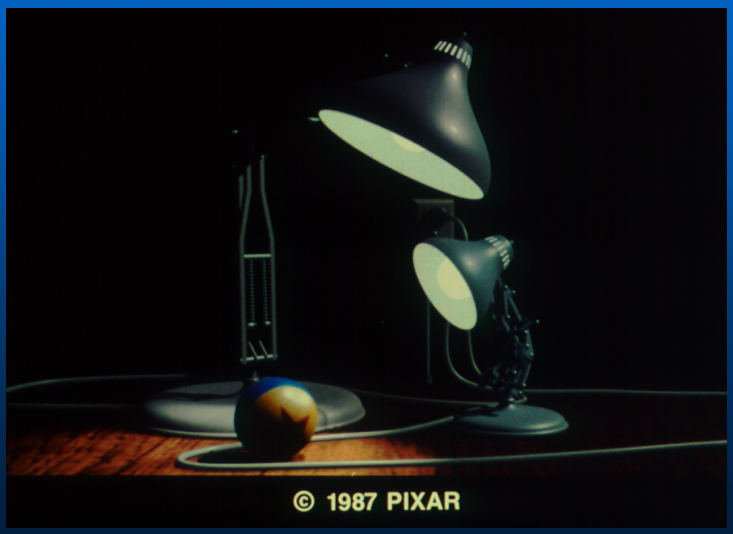


Behavioral Simulation



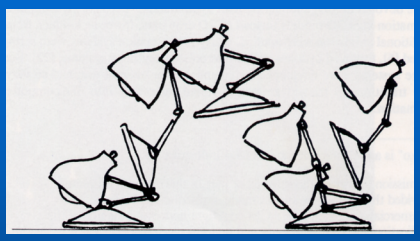
Use Your Imagination

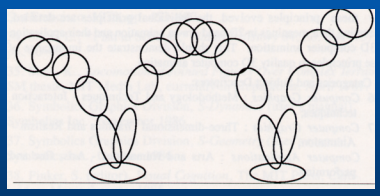
John Lasseter

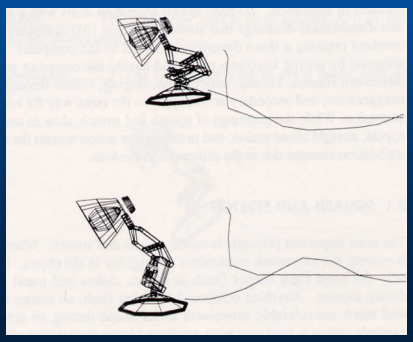


Tricks from Traditional Animation

- **Squash and Stretch**
  - Defining the rigidity and mass of an object by distorting its shape during an action







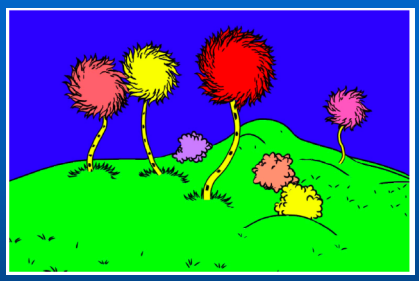
- **Secondary Action**
  - Action that results directly from another action

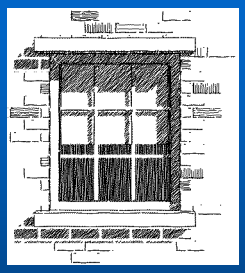
Why does it have to look like a photograph?

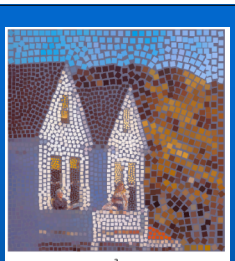
Artistic Rendering Approaches

- Artistic rendering problem (NPR)
  - Produce images from geometric models that are more expressive or mimic alternative media
- Artistic rendering solutions
  - Mimic characteristics of media
  - Physically simulate media
  - Break rules
  - Learn styles

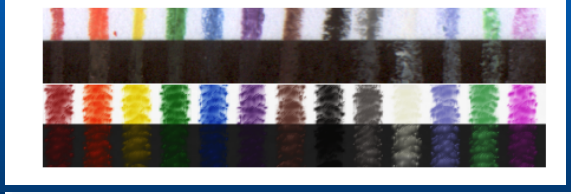
Mimic media

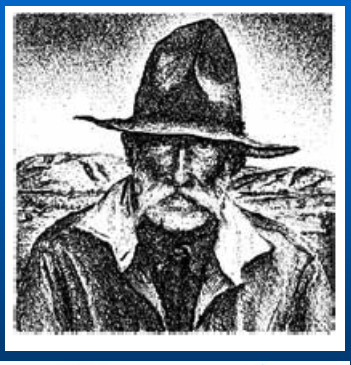


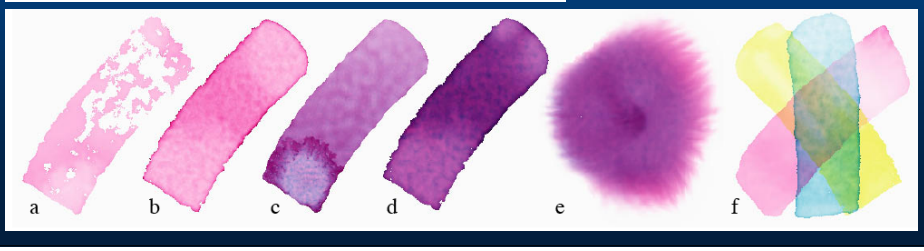




Physical Media Simulation







Break Rules