cheap tricks

- texture mapping
- procedural texture mapping
- bump mapping
- transparency mapping
- cut-out mapping
- depth of field
- lens effects
- jittering
- soft shadows

texture mapping

See cs155 gallery for examples

texture mapping

\[ f(p) : \text{coordinates in the texture map corresponding to surface point } p \]

texture mapping triangle

Input specifies texture coordinates of triangle vertices

texture mapping sphere

e.g. latitude/longitude
texture mapping

- surface
- texture (digital image)
- \( r, g, b \) of texture at \( f(p) \):
  1. use as luminance
  2. use as diffuse and spectral coefficient
  3. use as diffuse coefficient

cheap tricks

- texture mapping
- procedural texture mapping
- bump mapping
- transparency mapping
- cut-out mapping
- depth of field
- lens effects
- jittering
- soft shadows

procedural texture mapping

- create 3d "noise" procedure
- sample to find texture for surface

advantages

- don't need to find a mapping from a (complex) 3d surface to a 2d texture image
- concise representation of texture

invented by Ken Perlin: see his web site at NYU for more details

cheap tricks

- texture mapping
- procedural texture mapping
- bump mapping
- transparency mapping
- cut-out mapping
- depth of field
- lens effects
- jittering
- soft shadows
bump mapping

See gallery for example

bump mapping intuition

a surface appears to have a coarse texture
- jagged silhouette
- surface normals fluctuate across surface

simulate this by randomly perturbing normals in the lighting calculations

Perturbing normals can also create some illusion of this

cheap tricks

• texture mapping
• procedural texture mapping
• bump mapping
• transparency mapping
• cut-out mapping
• depth of field
• lens effects
• jittering
• soft shadows

Texture specifies "existence" of surface

cheap tricks

• texture mapping
• procedural texture mapping
• bump mapping
• transparency mapping
• cut-out mapping
• depth of field
• lens effects
• jittering
• soft shadows

blur based on distance from viewpoint
cheap tricks

- texture mapping
- procedural texture mapping
- bump mapping
- transparency mapping
- cut-out mapping
- depth of field
- lens effects
- jittering
- soft shadows

See paper mentioned in assignment

jittering: anti-aliasing technique

Run rt for example

- cast several ray through pixel neighborhood into scene
- for each:
  - find intersection point (if any) that is closest to eye
  - compute luminance at intersection
  - compute average luminance

soft shadows

See gallery for example
**Soft shadows**

Use jittering in occlusion test.

**Ray tracing**

- Simple ray casting
- Recursive ray tracing
- Modeling transforms
- Cheap tricks
- Optimizations

**Ray tracing complexity**

\[ O(\text{# of intersection tests}) = O(\text{#pixels \times # objects}) \]

Can we reduce the number of intersection tests?

**Optimization**

- Bounding boxes
- Oct-trees
- BSP-trees

**Bounding boxes: intuition**

10,000,000 triangles

Most rays miss the object.

**Bounding boxes**

1. Rule out rays by simple intersection test with bounding box.
2. Perform exhaustive test on remaining rays.
hierarchical coordinates

body xfm
body description
  head translate wrt body
  head rotate
  head description
    (eye1 translate wrt head
     eye1 scale
     eye1 description)
    (eye2 translate wrt head
     eye2 scale
     eye2 description)

intersection

body xfm
body description
  head translate wrt body
  head rotate
  head description
    (eye1 translate wrt head
     eye1 scale
     eye1 description)
    (eye2 translate wrt head
     eye2 scale
     eye2 description)

bounding box intersection

\[ P + \alpha v \]

\[ q = (p + \alpha v, p_z + \alpha v_z) \]
where \( \alpha = (\text{front}-p_z)/v_z \)

optimization

- bounding boxes
- oct-trees
- BSP-trees

We won't cover these. Look on the web for details.