CS770/870

CLASS 9

ADVANCE RENDERING
RAY TRACING
RADIOSITY
LIGHT FIELD
Ray Tracing

- Basic idea: Cast rays out 1 per pixel.
- Calculate light at surface of first intersection.
- Does refraction – easily
- Does cast shadows

- Turner Witted (1978)
Witted (1979)

- When a ray hits a surface, it can generate up to three new types of rays: reflection, refraction, and shadow.
- Reflection: Trace reflection. Whatever surface is intersected becomes what is seen in the reflection.
- Refraction: Ray passes through, and refracted. Whatever surface is intersected becomes what is seen at that point.
- Shadow: Trace a ray to the light source. If blocked, compute ambient. Otherwise apply standard CG lighting
- [Show Witted movie]. Used a micro-film recorder.
Modern ray tracing.

- Whole bundles of rays can be used to get effects like soft shadows, more realistic ambient, etc.
- Used for movie cars, (first pixar ray-traced movie).
Ray Tracing + Radiosity
Radiosity (doing ambient right)

- Derived from heat engineering
- For every polygon, light input = light reflected (diffusely) + light absorbed.
- Some polygons also emit light
- Ultimately all emissions = all absorptions.
- Result: A huge system of equations.
- Must compute the influence of every polygon on every other polygon.

- Don Greenberg, 1986
The radiosity calculation

\[ B_j = \rho_j H_j + E_j \]

- \( B_j \) – Radiosity of surface \( j \)
- \( \rho_j \) – Reflectivity of surface \( j \)
- \( H_j \) – Energy incident on surface \( j \)
- \( E_j \) – Energy emitted by surface \( j \)

\[ H_j = \sum_{i=1}^{N} B_i F_{ij}, \quad j = 1..N \]

- \( H_j \) – Energy incident on surface \( j \)
- \( B_i \) – Radiosity of surface \( i \)
- \( F_{ij} \) – Form factor \( ij \)

\[
\begin{bmatrix}
1 - \rho_1 F_{11} & -\rho_1 F_{12} & \cdots & -\rho_1 F_{1N} \\
-\rho_2 F_{21} & 1 - \rho_2 F_{22} & \cdots & -\rho_2 F_{2N} \\
\vdots & \vdots & \ddots & \vdots \\
-\rho_N F_{N1} & -\rho_N F_{N2} & \cdots & 1 - \rho_N F_{NN}
\end{bmatrix}
\begin{bmatrix}
B_1 \\
B_2 \\
\vdots \\
B_N
\end{bmatrix}
=
\begin{bmatrix}
E_1 \\
E_2 \\
\vdots \\
E_N
\end{bmatrix}
\]
Methods for improving ray tracing

- Use bundles of rays – stochastic sampling.
Volume rendering.
Visible human
Or CAT scans
Or MRI scans

How to look at it?
Slices,
The shape of organs
Volume rendering algorithm

- Used in visualization. E.g. CAT scans, MRI scans
- A 3D volume of data.
- Algorithm. Compute mapping from sensed data to transparency and color. E.g. Bone is opaque and white
- Assume light reaches each voxel
- Trace rays towards the viewpoint, one for each pixel.
- At a set sample points along the ray, calculate orientation of density gradient. Apply lighting model.
- Sum light from furthest to nearest point along ray, taking opacity into account.
But there are no surfaces with volume rendering

- If we actually need to have a model of a chunk of tissue we need another method
Marching squares
Marching cubes

- Consider the problem of making a contour map.
- At a fixed set of heights (or energy levels) we must create a continuous contour.
- Marching squares.
Marching squares

E.g. Contour at 50

48.3
50.7

51.2
53.8

Interpolate to find crossing

Algorithm:
for every contour value {
    for every square {
        1. Determine case
        2. Interpolate edges
        3. Draw line segment
    }
}

How many unique cases?
15 Cases for each square.
• Create a grid over the data set.
• For every square in the grid find which of the 16 cases
• If NOT 0000 OR 1111
• Interpolate along sides to find crossing points.
• Then draw line segment.
Marching Cubes

- 256 cases (8 edges on the cube).
- 15 unique cases
- Polygons instead of lines
Used for medical imaging. Virtual Colon
Light field sensing and rendering slides from Mark Levoy
Light field rendering

flipbook animation (QuickTime VR)  rebinning the rays to create new views
(movie is available at http://graphics.stanford.edu/papers/light)
Devices for recording light fields

- handheld camera [Buehler 2001]
- array of cameras [Wilburn 2005]
- plenoptic camera [Ng 2005]
- light field microscope [Levoy 2006]
Stanford Multi-Camera Array
[Wilburn SIGGRAPH 2005]

- $640 \times 480$ pixels $\times$
- 30 fps $\times$ 128 cameras

- synchronized timing
- continuous streaming
- flexible arrangement
Synthetic aperture photography
one camera’s view

synthetic aperture view

(movie is available at http://graphics.stanford.edu/projects/array)
Conventional versus light field camera
Prototype camera

Contax medium format camera

Kodak 16-megapixel sensor

Adaptive Optics microlens array

125μ square-sided microlenses

\[4000 \times 4000 \text{ pixels} \div 292 \times 292 \text{ lenses} = 14 \times 14 \text{ pixels per lens}\]
Light field rendering
Digital refocusing

- refocusing = summing windows extracted from several microlenses
Example of digital refocusing
Example of digital refocusing
Example of digital refocusing
Example of digital refocusing
Digitally moving the observer

- moving the observer = moving the window we extract from the microlenses
Lytro Camera

Seamless blending of CGI and imagery