Ray Tracing

Announcements: Quiz
• On Friday (3/2), in class
• One 8.5x11 sheet of notes allowed
• Sample quiz (from last year) is posted online
• Focus on “reading comprehension” and
  material for Homeworks 0, 1, & 2

Last Time?
• Keyframing
• Procedural Animation
• Physically-Based Animation
• Forward and Inverse Kinematics
• Motion Capture

Today
• Ray Casting
  – Ray-Plane Intersection
  – Ray-Sphere Intersection
  – Point in Polygon
• Ray Tracing
• Recursive Ray Tracing
• Distribution Ray Tracing

Durer’s Ray Casting Machine
• Albrecht Durer, 16th century

Ray Casting
Find the intersection and normal is the central part of ray casting
For every pixel
Construct a ray from the eye
For every object in the scene
Find intersection with the ray
Keep if closest
Shade depending on light and normal vector
**A Note on Local Shading**

- Surface/Scene Characteristics:
  - surface normal
  - direction to light
  - viewpoint
- Material Properties
  - color/texture
  - diffuse (matte)
  - specular (shiny)
- More later!

**Ray Representation?**

- Two vectors:
  - Origin
  - Direction (normalized is better)
- Parametric line (explicit representation)
  - \( P(t) = \text{origin} + t \times \text{direction} \)

**3D Plane Representation?**

- Plane defined by
  - \( P_o = (x,y,z) \)
  - \( n = (A,B,C) \)
- Implicit plane equation
  - \( H(P) = Ax + By + Cz + D = 0 \)
  - \( = n \cdot P + D = 0 \)
- Point-Plane distance?
  - If \( n \) is normalized, distance to plane, \( d = H(P) \)
  - \( d \) is the signed distance!

**Explicit vs. Implicit?**

- Ray equation is explicit \( P(t) = R_o + t \times R_d \)
  - Parametric
  - Generates points
  - Harder to verify that a point is on the ray
- Plane equation is implicit \( H(P) = n \cdot P + D = 0 \)
  - Solution of an equation
  - Does not generate points
  - Verifies that a point is on the plane

**Ray-Plane Intersection**

- Intersection means both are satisfied
- So, insert explicit equation of ray into implicit equation of plane & solve for \( t \)
  \[
  P(t) = R_o + t \times R_d \\
  H(P) = n \cdot P + D = 0 \\
  n (R_o + t \times R_d) + D = 0 \\
  t = -D + n \cdot R_o / n \cdot R_d
  \]

**Additional Housekeeping**

- Verify that intersection is closer than previous \( P(t) < t_{\text{current}} \)
- Verify that it is not out of range (behind eye) \( P(t) > t_{\text{min}} \)
Normal

- Needed for shading
  - diffuse: dot product between light and normal
- Normal of a plane is constant!

Ray-Triangle Intersection

- Use barycentric coordinates:
  - \( P(\alpha, \beta, \gamma) = \alpha a + \beta b + \gamma c \)
  - with \( \alpha + \beta + \gamma = 1 \)
  - If \( 0 < \alpha < 1 \) & \( 0 < \beta < 1 \) & \( 0 < \gamma < 1 \)
    then the point is inside the triangle!

How Do We Compute \( \alpha, \beta, \gamma \) ?

- Ratio of opposite sub-triangle area to total area
  - \( \alpha = A_c/A \) \( \beta = A_b/A \) \( \gamma = A_a/A \)
- Use signed areas for points outside the triangle

Using Cramer’s Rule…

- Used to solve for one variable at a time in system of equations

\[
\begin{vmatrix}
\alpha_c - R_{ox} & \alpha_c - c_x & R_{dx} \\
\alpha_c - R_{oy} & \alpha_c - c_y & R_{dy} \\
\alpha_c - R_{oz} & \alpha_c - c_z & R_{dz}
\end{vmatrix}
\]

\[
\begin{vmatrix}
\alpha_c - b_x & \alpha_c - R_{ox} & R_{dx} \\
\alpha_c - b_y & \alpha_c - R_{oy} & R_{dy} \\
\alpha_c - b_z & \alpha_c - R_{oz} & R_{dz}
\end{vmatrix}
\]

\[
\begin{vmatrix}
\alpha_c - b_x & \alpha_c - c_x & \alpha_c - R_{ox} \\
\alpha_c - b_y & \alpha_c - c_y & \alpha_c - R_{oy} \\
\alpha_c - b_z & \alpha_c - c_z & \alpha_c - R_{oz}
\end{vmatrix}
\]

\[
\begin{vmatrix} A_1 \end{vmatrix}
\]

Sphere Representation?

- Implicit sphere equation
  - Assume centered at origin (easy to translate)
  - \( H(P) = P \cdot P - r^2 = 0 \)

Ray-Sphere Intersection

- Insert explicit equation of ray into
  implicit equation of sphere & solve for \( t \)

\[
P(t) = R_o + tR_d \quad H(P) = P \cdot P - r^2 = 0
\]

\[
(R_o + tR_d) \cdot (R_o + tR_d) - r^2 = 0
\]

\[
R_dR_d^2 + 2R_dR_t + R_oR_o - r^2 = 0
\]
Ray-Sphere Intersection

• Quadratic: \( at^2 + bt + c = 0 \)
  - \( a = 1 \) (remember, \( ||\vec{R}_d|| = 1 \))
  - \( b = 2\vec{R}_d \cdot \vec{R}_o \)
  - \( c = \vec{R}_o \cdot \vec{R}_o - r^2 \)

• with discriminant \( d = (b^2 - 4ac) \)
• and solutions \( t_{\pm} = \frac{-b \pm \sqrt{d}}{2a} \)
• What does it mean if there are no solutions, 1 solution, or 2 solutions?

Today

• Ray Casting
• Ray Tracing
  – Shadows
  – Reflection
  – Refraction
• Recursive Ray Tracing
• Distribution Ray Tracing

Questions?

How Can We Add Shadows?

Find the point to be shaded
For every light,
Construct ray from point to light
For every object
Find intersection of ray with object
If no objects between point and light
Add contribution from light

Mirror Reflection

• Cast ray symmetric with respect to the normal
• Multiply by reflection coefficient (color)

Reflection

• Reflection angle = view angle
• \( \vec{R} = \vec{V} - 2(\vec{V} \cdot \vec{N})\vec{N} \)
Transparency

- Cast ray in refracted direction
- Multiply by transparency coefficient (color)

Qualitative Refraction

From “Color and Light in Nature” by Lynch and Livingston

Refraction

\[
I = N \cos \theta_i - M \sin \theta_i \\
M = (N \cos \theta_j - 1) / \sin \theta_j \\
T = N \cos \theta_j + M \sin \theta_j \\
\eta_i \sin \theta_i = \eta_T \sin \theta_T \\
\sin \theta_j = \frac{\eta_i}{\eta_T} \sin \theta_T
\]

- Total internal reflection when the square root is imaginary
- Don’t forget to normalize!

Total Internal Reflection

From “Color and Light in Nature” by Lynch and Livingston

Questions?
Readings for Today: (read one…)


Today

- Ray Casting
- Ray Tracing
- Recursive Ray Tracing
- Distribution Ray Tracing

Ray Tracing

- Trace ray
  - Intersect all objects
  - Color = ambient term
  - For every light
    - Cast shadow rays
    - Color += local shading term
  - If mirror
    - Color += color_refl * trace reflected ray
  - If transparent
    - Color += color_trans * trace transmitted ray

- Does it ever end?

The Ray Tree

- Stopping criteria:
  - Recursion depth
    - Stop after a number of bounces
  - Ray contribution
    - Stop if reflected / transmitted contribution becomes too small

Ray Debugging

- Visualize the ray tree for single image pixel

Today

- Ray Casting
- Ray Tracing
- Recursive Ray Tracing
- Distribution Ray Tracing
- Soft shadows
- Antialiasing (getting rid of jaggies)
- Glossy reflection
- Motion blur
- Depth of field (focus)
Shadows

- one shadow ray per intersection per point light source

point light source

Shadows & Light Sources

no shadow rays

one shadow ray

Soft Shadows

- multiple shadow rays to sample area light source

area light source

clear bulb

clear bulb

Antialiasing – Supersampling

- multiple rays per pixel

point light

area light

Glossy Reflection

- multiple reflection rays

perfect mirror

justin legakis
Motion Blur

- Sample objects temporally

Depth of Field

- multiple rays per pixel

Ray Tracing Algorithm Analysis

- Ray casting
- Lots of primitives
- Recursive
- Distributed Ray Tracing Effects
  - Soft shadows
  - Anti-aliasing
  - Glossy reflection
  - Motion blur
  - Depth of field

Raytracing & Epsilon

Reading for Tuesday 3/6:

- Goral, Torrance, Greenberg & Battaile “Modeling the Interaction of Light Between Diffuse Surfaces”, SIGGRAPH '84