CENG 477
Introduction to Computer Graphics
Ray Tracing: Geometry
Ray Tracing

• In ray tracing, we model the propagation of light and its interaction with materials to create realistic images.
• Different from reality, we assume that the rays originate from the eye (or the camera).
• This allows us to avoid processing rays that will not be visible to the eye (or the camera).
Components

- In RT, we have the following components:
  - Camera (or eye)
  - Image plane
  - Objects
  - Light sources
  - and lots of rays!
Camera

- Camera represents the origin of the rays that we will trace
- It is represented by a position \((e)\) and orientation \((u, v, w)\)
- These vectors are orthogonal to each other

- The position and the orientation are defined with respect to a global (or world) coordinate system
- This global system has origin at \((0, 0, 0)\) and its three axis are: \((1, 0, 0)\), \((0, 1, 0)\), \((0, 0, 1)\)
Camera

- The $\mathbf{u}$, $\mathbf{v}$, $\mathbf{w}$ vectors of the camera have the following meaning:
  - $\mathbf{v}$: up vector
  - $\mathbf{w}$: opposite of gaze vector
  - $\mathbf{u}$: $\mathbf{v} \times \mathbf{w}$ with $\times$ representing cross-product
Image Plane

- Image plane is a surface on which the final image is formed
- It is divided into pixels through which rays will be cast
- It is represented by its:
  - Resolution (nx, ny)
  - Distance to the camera
  - Left, right, top, bottom coordinates

- The image plane is typically centered and orthogonal with respect to the camera coordinates (not the world coordinate system)
Objects

- Objects consist of mathematically defined geometrical shapes or meshes made up of triangles
Objects

- It is possible to model complex shapes using a large number of small triangles or quadrilaterals.
Objects

- Objects may have different materials that affect their appearance

James P. O'Shea
Left to right: High-gloss, diffuse, mirrored, semi-gloss
Light Sources

• Light sources provide the illumination in the scene
• The geometrical relationship between objects and light sources may produce shadows
• Typically, three types of light sources are used:
  – Ambient
  – Directional
  – Positional (Point)
Rays

- A ray, or half-line, is a 3D parametric line with a half-open interval, usually \([0, \infty)\)

\[ r(t) = o + td \]

- Both \(o\) and \(d\) are element of \(\mathbb{R}^3\) and \(t\) is in range \([0, \infty)\)
- For a fixed \(t\), \(r(t)\) represents a point on this line
Ray Tracing

• The basic algorithm:

\textbf{for} each pixel \textbf{do} \\
\hspace{1em} compute viewing (eye, primary) rays \\
\hspace{1em} find first object hit by ray and its surface normal $n$ \\
\hspace{1em} set pixel color to value computed from hit point, light, and $n$
Computing Eye Rays

\[ m = e + w \text{distance} \]
\[ q = m + lu + tv \]

\[ s = q + suu - svv \]

How to find \( su \) and \( sv \)?

\[ su = (r - l)(i + 0.5)/n_x \]
\[ sv = (t - b)(j + 0.5)/n_y \]

How to write the final eye ray equation?

\[ r(t) = e + (s - e)t = e + dt \]

What information did we use to derive this?

- \( e \): location of the camera
- \( \text{distance} \): camera-image plane distance
- \( n_x \): image width
- \( n_y \): image height
- \( u, v, w \): camera vectors
- \( r, l, t, b \): image plane borders (in \( uvw \) space)
Some Example Values

- \( \mathbf{u} = (1, 0, 0) \)
- \( \mathbf{v} = (0, 1, 0) \)
- \( \mathbf{e} = (0, 0, 0) \)
- \( n_x = 1024 \)
- \( n_y = 768 \)
- \( l = -1, r = 1 \)
- \( b = -1, t = 1 \)
- \( d = 1 \)

Compute the ray equation passing through the pixel (256, 192):

\[
\mathbf{r}(t) = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + t \begin{bmatrix} -0.5 + 1/1024 \\ 0.5 - 1/768 \\ -1 \end{bmatrix}
\]

Where is this ray at \( t = 0, t = 1, t = 2 \)?
Ray-Object Intersections

• **Goal:** To decide at what point, if any, a 3D line (ray) intersects a 3D surface
Parametric Lines

• A 2D line can be represented as: \( y - mx - b = 0 \). This is called the implicit form

• A parametric 2D line can be represented as:
  \[
  x(t) = 2 + 7t \\
  y(t) = 1 + 2t
  \]

• A 3D parametric line (ray) can be written as:
  \[
  x(t) = 2 + 7t \\
  y(t) = 1 + 2t \\
  z(t) = 3 - 5t
  \]

• Alternatively, in vector form, we can write \( \mathbf{r}(t) = \mathbf{o} + td \) where \( \mathbf{o} = (2, 1, 3) \) and \( \mathbf{d} = (7, 2, -5) \)
Ray (Reminder)

- A ray is a half-line represented by $r(t) = o + td$ with $t \geq 0$.

- We want to know if a ray intersects an object with $t$ in the interval $[t_{\text{min}}, t_{\text{max}}]$
  - If $t < t_{\text{min}}$, the object is too close (maybe in front of the image plane).
  - If $t > t_{\text{max}}$, the object is too far (outside the range we want to consider).
Implicit Surfaces

• Rays will intersect surfaces, so we need to know how to represent surfaces
• In implicit form a surface can be written as \( f(x, y, z) = 0 \)
• Why is it called implicit?
  – You can test whether a point is on the surface, but you cannot generate points on the surface
• Another way to write: \( f(p) = 0 \) where \( p = (x, y, z) \)
• A ray will intersect this surface if:

\[
f(r(t)) = f(o + td) = 0
\]
Ray-Plane Intersection

• Consider the plane equation written in vector form as:

$$(p - a) \cdot n = 0$$

If you expand this, you’ll get the familiar $Ax + By + Cz + D = 0$ equation.

• Here, $a$ is a point on the plane, $n$ is the normal vector of the plane.

• $a$ and $n$, are known quantities and $p$ is the variable.
Ray-Plane Intersection

- Simply plug $r(t) = o + td$ into the previous equation:

$$ (o + td - a).n = 0 $$

- Solving for $t$, we get:

$$ t = (a - o).n / (d.n) $$

- If $t$ is in $[t_{min}, t_{max}]$, the ray hits the plane and it is within the limits of our desired viewing range

- What if $d.n = 0$?
Ray-Sphere Intersection

• A sphere can be represented as:

\[(x - c_x)^2 + (y - c_y)^2 + (z - c_z)^2 - R^2 = 0\]

• where \(\mathbf{c} = (c_x, c_y, c_z)\) is the center and \(R\) is the radius

• In vector form, we can rewrite this as:

\[(\mathbf{p} - \mathbf{c}) \cdot (\mathbf{p} - \mathbf{c}) - R^2 = 0\]

• Again, plug in the ray equation to find \(t\):

\[(\mathbf{o} + t\mathbf{d} - \mathbf{c}) \cdot (\mathbf{o} + t\mathbf{d} - \mathbf{c}) - R^2 = 0\]
Ray-Sphere Intersection

• This gives:

$$(d \cdot d)t^2 + 2d \cdot (o - c)t + (o - c) \cdot (o - c) - R^2 = 0$$

• Note that, this is a quadratic equation in $t$:

$$At^2 + Bt + C = 0$$

• The solution is:

$$t = \frac{-d \cdot (o - c) \pm \sqrt{(d \cdot (o - c))^2 - (d \cdot d)((o - c) \cdot (o - c) - R^2)}}{d \cdot d}$$

• What if the determinant is less than zero?
Ray-Triangle Intersection

• So far, we have been using implicit equations to represent surfaces: \( f(x, y, z) = 0 \)

• Ray-triangle intersection is best found if the triangle is represented using a **parametric form**:

\[
\begin{align*}
  x &= f(u, v) \\
  y &= g(u, v) \\
  z &= h(u, v)
\end{align*}
\]

• For instance, a sphere in parametric form can be represented as:

\[
\begin{align*}
  x &= \cos \Phi \sin \Theta \\
  y &= \sin \Phi \sin \Theta \\
  z &= \cos \Theta
\end{align*}
\]
Ray-Triangle Intersection

• Two techniques are possible:
  – Lengthy but simple
  – Shorter but somewhat more complex
Lengthy But Simple Method

• First, intersect the ray with the plane of the triangle:
  – The plane normal can be found by cross product
  – The plane equation can be found from the normal and a point

\[ n = (c - b) \times (a - b) \]

\[ f(p) = (p - a) \cdot n = 0 \]
Lengthy But Simple Method

• If the ray intersects triangle’s plane, we need to determine if it is inside the triangle
  – How to make an inside check?

\[ \text{If } p \text{ is on the same side of } \overrightarrow{ab} \text{ as } c \text{ and} \]
\[ \text{If } p \text{ is on the same side of } \overrightarrow{bc} \text{ as } a \text{ and} \]
\[ \text{If } p \text{ is on the same side of } \overrightarrow{ac} \text{ as } b \]
• Then \( p \) is inside the triangle
• Otherwise, it is outside the triangle
Lengthy But Simple Method

- To check this, we use cross and dot products

\[ \mathbf{c} \times (\mathbf{p} - \mathbf{a}) = (\mathbf{p} - \mathbf{b}) \times (\mathbf{p} - \mathbf{a}) \]

Pay attention to the order of the terms in cross-products!
Alternative Method (Faster)

• A faster method involves using barycentric coordinates
• Imagine the triangle positioned on a non-orthogonal grid:
Barycentric Coordinates

• Any point on this grid is characterized by two coordinates
Barycentric Coordinates

- Zooming into our triangle:

\[ \alpha = 1 \]
\[ \beta = 0 \]
\[ \beta = 0.5 \]
\[ \beta = 1 \]
\[ \alpha = 0 \]
\[ \alpha = 0.5 \]
Barycentric Coordinates

- If $0 \leq \alpha \leq 1$ and $0 \leq \beta \leq 1$ then we are in this region:
Barycentric Coordinates

- On top that, if $0 \leq \alpha + \beta \leq 1$ then we are inside the triangle

\[ \begin{align*}
\beta &= 0 \\
\alpha &= 1 \\
\beta &= 0.5 \\
\alpha &= 0.5 \\
\beta &= 1 \\
\alpha &= 0
\end{align*} \]
Barycentric Coordinates

- Note that there is also a hidden third coordinate, $\gamma$: 

\[ a = \beta \left( \frac{1}{3} \right) = \alpha \left( \frac{2}{3} \right) = \gamma \left( -\frac{1}{3} \right). \]
Barycentric Coordinates

- As a rule, \( \alpha + \beta + \gamma = 1 \):

\[
\begin{align*}
\beta &= 0 \\
\alpha &= 1 \\
\gamma &= -1
\end{align*}
\]
Barycentric Coordinates

• Any point inside the triangle plane can be parametrized by these three coordinates with the following conditions:

\[ p(\alpha, \beta, \gamma) = \alpha a + \beta b + \gamma c \]

with the constraints

\[ 0 < \alpha < 1 \]
\[ 0 < \beta < 1 \]
\[ 0 < \gamma < 1 \]
Barycentric Coordinates

- Also, the coordinates can be computed from area ratios (but we will not use this):

\[ \begin{align*}
\alpha &= \frac{A_a}{A} \\
\beta &= \frac{A_b}{A} \\
\gamma &= \frac{A_c}{A}
\end{align*} \]

where

\[ A = A_a + A_b + A_c \]
Back to Intersection

• Note that we can eliminate one of the parameters:
  \[ \alpha = 1 - \beta - \gamma \]

  \[ p(\beta, \gamma) = a + \beta(b - a) + \gamma(c - a) \]

• The point \( p \) is inside (or on) the triangle if and only if:
  \[ \beta + \gamma \leq 1 \]
  \[ 0 \leq \beta \]
  \[ 0 \leq \gamma \]

• Plug the ray equation \( r(t) = o + td \) into \( p \):
  \[ o + td = a + \beta(b - a) + \gamma(c - a) \]
Ray-Triangle Intersection

• How to solve for t?
• Expand from the vector form into individual coordinates:

\[o_x + td_x = a_x + \beta(b_x - a_x) + \gamma(c_x - a_x)\]
\[o_y + td_y = a_y + \beta(b_y - a_y) + \gamma(c_y - a_y)\]
\[o_z + td_z = a_z + \beta(b_z - a_z) + \gamma(c_z - a_z)\]

• The unknowns here are \(t, \beta, \) and \(\gamma\)
• We have 3 equations and 3 unknowns
Ray-Triangle Intersection

- Rewrite this system in matrix form:

\[
\begin{bmatrix}
ax - bx & ax - cx & dx \\
ay - by & ay - cy & dy \\
az - bz & az - cz & dz \\
\end{bmatrix}
\begin{bmatrix}
\beta \\
\gamma \\
t \\
\end{bmatrix}
= 
\begin{bmatrix}
ax - ox \\
ay - oy \\
az - oz \\
\end{bmatrix}
\]

\[
A = 
\begin{bmatrix}
\beta \\
\gamma \\
t \\
\end{bmatrix}
= 
\begin{bmatrix}
ax - ox \\
ay - oy \\
az - oz \\
\end{bmatrix}
\]

- And solve for $t$, $\beta$, and $\gamma$ using Cramer’s rule
Ray-Triangle Intersection

- Cramer’s rule:

\[
\beta = \frac{\begin{vmatrix} a_x - o_x & a_x - c_x & d_x \\ a_y - o_y & a_y - c_y & d_y \\ a_z - o_z & a_z - c_z & d_z \end{vmatrix}}{|A|}
\]

\[
\gamma = \frac{\begin{vmatrix} a_x - b_x & a_x - o_x & d_x \\ a_y - b_y & a_y - o_y & d_y \\ a_z - b_z & a_z - o_z & d_z \end{vmatrix}}{|A|}
\]

\[
t = \frac{\begin{vmatrix} a_x - b_x & a_x - c_x & a_x - o_x \\ a_y - b_y & a_y - c_y & a_y - o_y \\ a_z - b_z & a_z - c_z & a_z - o_z \end{vmatrix}}{|A|}
\]

where \(|.|\) denotes the determinant
Finding the Determinant

• If $A$ is equal to:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

• Then $|A|$ is given by:

$$|A| = a(ei - hf) + b(gf - di) + c(dh - eg)$$
Ray-triangle Intersection

• Use this to find the determinants of the other terms and compute $t$, $\beta$, and $\gamma$.
• The ray will intersect the triangle if:

$$t_{\min} \leq t \leq t_{\max}$$
$$\beta + \gamma \leq 1$$
$$0 \leq \beta$$
$$0 \leq \gamma$$

• Ray-triangle intersection is the most important as any complex object can be represented using a set of triangles
Complex Models

- Bunny model composed of 725,000 triangles

From Stanford University
Complex Models

• Buddha model composed of 9,200,000 triangles

From Stanford University
Complex Models

- Dragon model composed of 5,500,000 triangles

From Stanford University
Complex Models

• Armadillo model composed of 7,500,000 triangles

From Stanford University
Complex Models

• Lucy model composed of 116,000,000 triangles

From Stanford University
Intersection Cost

- Let’s compute how many intersection tests we need to perform to render a model composed of 1,000,000 triangles with an image size of 1024x1024
- $1,000,000 \times 1024 \times 1024 \approx 1,000,000,000,000$ (one trillion)
- And this is just the intersection – realistic shading is generally more costly (next week)
- That’s why ray tracing is very slow
- However, ray tracing can be accelerated by using:
  - Multiple computers
  - GPUs
  - Acceleration structures
Realism

- Intersection tests give us the surface position that is hit by a ray
- To create realistic images, we need to compute realistic models of light-surface interaction at that point on the surface
- This will be the topic of the next week

From ACM Siggraph