Dielectrics and Constructive Solid Geometry

Presented by: Richard Olney, Ashima Shrivastava
Dielectrics: Transparency and Refraction
Physics: Snell’s Law

\[ n \sin \theta = n_t \sin \phi. \]

Refractive indexes \( n \) and \( n_t \) where \( n_t > n \)

Refractive indexes of air is 1.0
Physics: Some Derivations

\[ n \sin \theta = n_t \sin \phi. \]

\[ \sin^2 \theta + \cos^2 \theta = 1 \]

\[ \cos^2 \phi = 1 - \left( n^2 (1 - \cos^2 \theta) \right) / n_t^2 \]
Physics: Orthonormal basis

\( n \) and \( b \) form an orthonormal basis

\[
\begin{align*}
  t &= \sin \phi \ b - \cos \phi \ n \\
  d &= \sin \theta \ b - \cos \theta \ n \\
  b &= (d + \cos \theta \ n)/\sin \theta
\end{align*}
\]
Physics: Salve for $t$

\[ t = \frac{(n^2 (d + \cos \theta \ n))}{n_t^2} - \cos \phi \ n \]

\[ = \frac{(n^2 (d+n (d \cdot n)))}{n_t^2} \]

\[ - n \sqrt{1-(n^2(1-(d \cdot n)^2))/n_t^2} \]

Total Internal Reflection if the discriminant is < 0
Schlick approximation

\[ R(\theta) = R_0 + (1 - R_0) (1 - \cos \theta)^5 \]

\[ R(\theta) = R_0 + (1 - R_0) (1 - d \cdot n)^5 \]

\[ R_0 = \left(\frac{n_t - 1}{n_t + 1}\right)^2 \]

\( \theta \) angle in air, and \( n = 1.0 \)
**Light Attenuation**

\[ I = k \exp(-Cx) \]

\[ I(0) = I_0 \]

\[ I(x) = I_0 \exp(-Cx) \]

\[ I(1) = aI_0 \]

\[ I_0a = I_0 \exp(-C) \implies -C = \ln(a) \]

\[ I(s) = I(0)e^{\ln(a)s} \]

\( I(s) \) is the intensity of the beam at distance \( s \) from the interface, but \( a \) generally needs to be hand tuned.
Implementation

Introduce a dielectric material, with information on light attenuation constants.

Check for dielectric materials before you would check for mirror reflection.

Then use the following pseudo-code to spawn your new rays.
### Pseudo Code

if \((p)\) is on a dielectric
\[ r = \text{reflect}(d, n) \]
if \((d \cdot n < 0)\)
\[ \text{refract}(d, n, n, t) \]
\[ c = -d \cdot n \]
\[ k_r = k_g = k_b = 1 \]
else
\[ k_r = \exp(-a_r t) \]
\[ k_g = \exp(-a_g t) \]
\[ k_b = \exp(-a_b t) \]
if \(\text{refract}(d, -n, 1/n, t)\)
\[ c = t \cdot n \]
else
return \(k \times \text{traceray}(p + tr)\)
\[ R_0 = \left(\frac{n-1}{n+1}\right)^2 \]
\[ R = R_0 + (1 - R_0)(1 - c)^5 \]
return \(k(R \text{traceray}(p + tr) + (1 - R) \text{traceray}(p + tt))\)

# refract function returns false if there is total internal reflection, # otherwise fills \(t\)
Constructive Solid Geometry (CSG)

- Constructive solid geometry is using combinations of basic 3D shapes to create models.
- Major advancement in solid modeling.
- The shapes are joined using various operations.
- Final geometry is created.
There are three steps for solid modeling using constructive solid geometry:

- Primitive solids
- Boolean operations
- Creating binary tree
The very first solid that we assemble to create any solid geometry are the primitive solids.
Performing boolean operations on the solids to generate a new geometry

- A union B
- A difference B
- A intersection B
Boolean operations conti.

For every object you can compute where the ray enters and exits the object.

Compute the intersection between the ray and A and B separately (which gives inside range for each object) to compute intersection between ray and object AB. Then compute inside range for AB.
To represent CSG method we use binary tree.

Binary trees are used to show how different primitives are used to create new solids.

- Leaves - Primitive solids
- Root - Resulting solid
CSG - Example

Create CSG of following solid -

Steps -

- Find primitive used - 2 cuboid
- Boolean operation - A - B
- Tree structure - WhiteBoard

(WhiteBoard figure)
Question

Create CSG model for the given geometry:

- Primitive solid -
- Boolean geometry -
- Tree -
Thank you!!

See chapter 13 of *Fundamentals of Computer Graphics, Fourth Edition* by Marschner and Shirley for more details on these topics.