Total Internal Refraction and Fiber Optics

Chris DiPastina [dipasc]
Rennselaer Polytechnic Institute [@rpi.edu]

ABSTRACT

Light refraction inside of translucent material involves a critical angle at which the light refracting through the material is completely refracted back into the object. With fiber optics, this critical angle is very low. The project seeks to modify the HW3 ray-tracer to produce the effects of fiber optics material.

INTRODUCTION

1.1 BACKGROUND

Refraction of light through translucent materials is used throughout computer graphics, with a focus on limiting the bounces of rays and efficient gathering of indirect light. What we seek to accomplish is the accurate depiction of the interaction of light in fiber optics. When light refracts off of a surface, the sine of the angle of refraction is equal to the sin of the initial angle multiplied by the coefficient of the refracted surface over the coefficient of the initial surface. When the ratio of the refracted coefficient to the initial coefficient is 0, this causes Total Internal Reflection (TIR) which refracts all of the light from the ray, reflecting nothing. In fiber optics, the goal is to have the lowest possible ratio, meaning that the refraction coefficient for the refraction surface must be as low as possible. Using 0 results in errors dividing by 0, so we set this value to a low value near to zero. We simulate the interaction between the surfaces by modifying a ray tracer to refract when it passes between surfaces with varying refraction coefficients.

IMPLEMENTATION

2.1 Calculating Refraction

The ray-tracer must be altered so that rather than simply calculating the mirror direction, the algorithm must calculate the direction to refract the light in. Using Snell's Law, we know that

Sin(1)*C1=Sin(2)*C2 where

Sin(1) is the sine of the angle of entry

Sin(2) is the sine of the angle of refraction

C1 is the refraction Coefficient for the initial material

C2 is the refraction Coefficient for the material being entered

However, we need to calculate a direction vector for the angle, so we must use the following equation for vector refraction

Cos(1) is the cosine for the angle of incidence

Cos(2) is the cosine for the angle of refraction

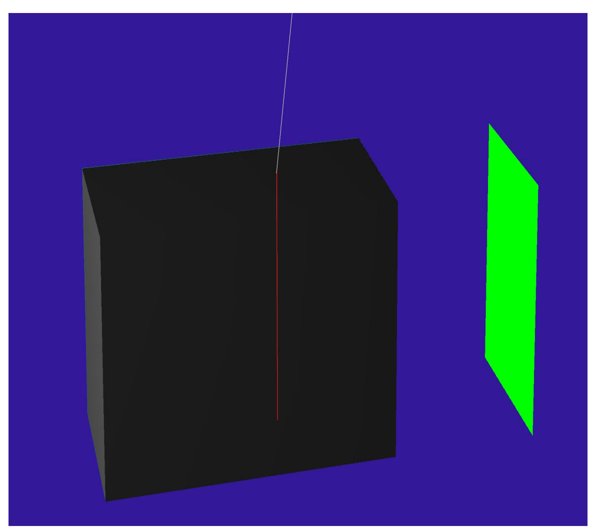
n is the normal for the surface

I is the vector of the incoming ray

v is the vector for the refracted ray

```
        Cos(1)=n DOT(-I)
        Cos(2)= sqrt(1-((C1/C2)^2)(1-(cos(1)^2)
if(Cos(1)<0):
        v=(C1/C2)I+((C1/C2)Cos(1)+Cos(2))n
else:
        v=(1+(2*Cos(2))*n
```

We then set the intersection to alter a value in the hit to convey whether the ray has intersected the backside or the frontside of the mesh. This allows us to swap C1 and C2 when the ray is leaving the material. If this is not done, the computer thinks the ray is entering the material and will send the ray in the wrong direction



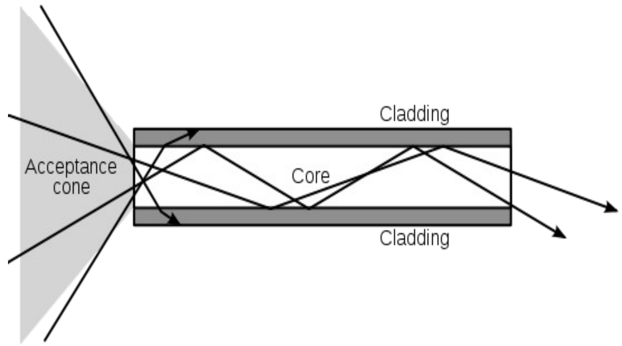
A ray bounces incorrectly because the computer does not check which side the ray is colliding with.

2.2 Obstacles

One obstacle is choosing the correct coefficients to result in Total Internal Reflection. The goal is to cause the Critical Angle so that light reflecting off the sides is contained but light striking the front is released. The critical angle is defined as
C=arcsin(C2/C1)
The index of refraction for fiber optics is about 1.48. This will cause light to reflect out of the object at undesirable locations along the length of the object, which is why most fiber optics are coated in a material which causes light hitting the sides to be reflected back into the inner material.

Another issue is the number of bounces required. Normally this is entered in the command line, whereas in our implementation we will bounce the ray until it leaves the mesh. This will allow the ray to eventually leave the mesh, whereas limiting to a number of bounces will cause the ray to become lost in the material. We will use the bounce count to limit the bounces for rays that have trouble getting out.

Additionally, to allow for variation in material, each face is altered to allow for a different index of refraction, passed in with the object file. Two separate values are stored for the back and front of the face. This was done to allow for the creation of models with realistic models such as fiber optics with cases around the core material (Cladding).[wiki]. Cladding is essential in order to refract the rays down the tube. Cladding works by having a higher index of refraction than the inner material, which will cause the light ray to reflect back into the inner material. Rather than build geometry around the object, this allows us to simulate cladding around a face without actually constructing it. Constructing the cladding could be done by building a mesh around the inner material and storing the data the same way. The outer mesh would hold the index of refraction for the cladding on the back facing index and the air on the outer index. This would allow for a more accurate depiction of the physical process but would slow the computation to compute what amounts very little extra data. This assumes that the goal is to model the inner transmission of light, rather than to simulate the interaction of the light and the cladding.

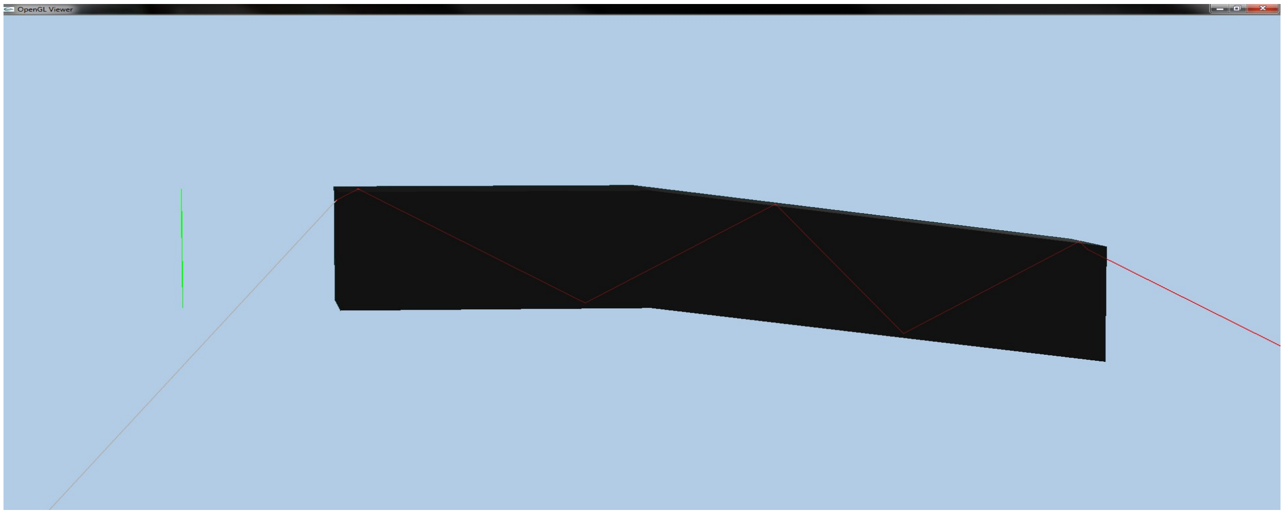


http://upload.wikimedia.org/wikipedia/commons/thumb/4/46/Optical-fibre.svg/550px-Optical-fibre.svg.png
An example of fiber optics with cladding. We simulate the cladding without actually modeling it, however the procedure of modeling the cladding could be represented using our model.

Lastly, with high refraction indexes on the mesh, it is nearly impossible for the light to leave the mesh. To this end, we would like the light to be able to pass through the mesh if Cos(1) is not negative. If Cos(1) is positive, we will send back a different vector to represent the light escaping from the mesh. This vector is defined:
V=I+(2Cos(1))n.

RESULTS

In the end, there was a problem with getting the vector equation to work properly. Instead of reflecting inward, pieces bouncing off the inner sides were escaping into the scene. To patch this, whenever the incident index of refraction is smaller than the index of the material the ray is moving into, reflection was used. This situation approximates the result of a proper fiber optic simulation, if the critical angle were zero at the interface between the core and the cladding, although it is not very accurate. This has the unfortunate side effect of not allowing any light through the mesh. The result caused rays to move as pictured below.



REFERENCES

http://en.wikipedia.org/wiki/Snell's_law#cite_note-11

http://en.wikipedia.org/wiki/File:Optical-fibre.svg

M. Berger, T. Trout. "Ray tracing mirages". IEEE Computer Graphics and
Applications, 11(5), pp. 36-41, May 1990.

M. Berger, T. Trout. "Ray tracing mirages". IEEE Computer Graphics and

Applications, 11(5), pp. 36-41, May 1990.

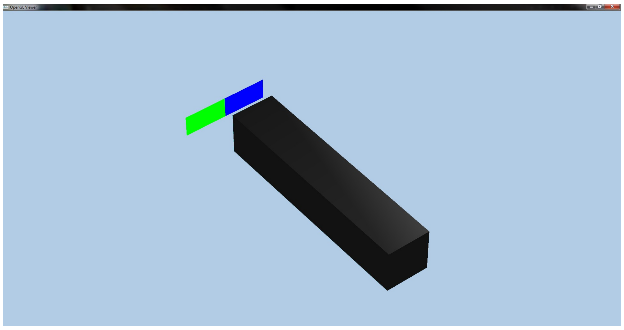
D. Gutierrez, O. Anson, A. Muoz, F. J. Seron. Chasing the green

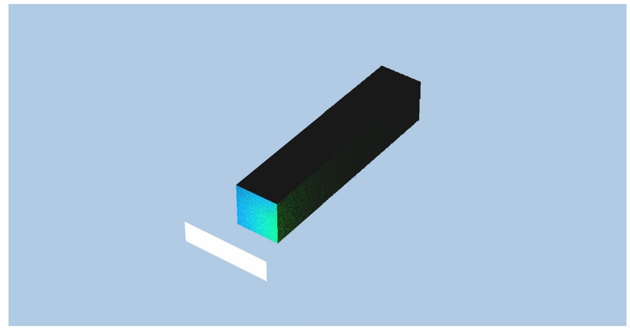
flash: A global illumination solution for inhomogeneous media. SCCG
(Spring Conference on Computer Graphics). ISBN 1-58113-914-4,
pp. 97-105. Slovakia, 2004.

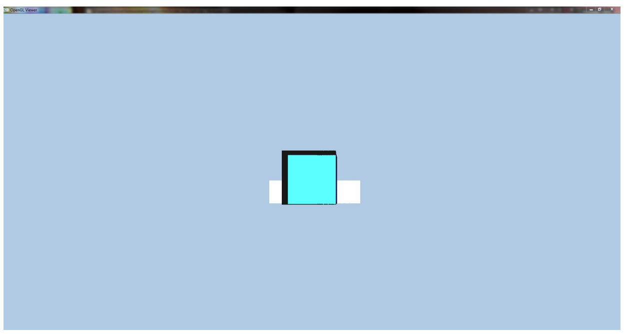
Olivier Genevaux, rederic Larue, and Jean-Michel Dischler. “Interactive Refraction on Complex Static Geometry using Spherical Harmonics”

Scott T Davis and Chris Wyman. “Interactive Refractions with Total Internal Reflection”. ACM , International Conference on Real-time rendering. New York, NY, USA,

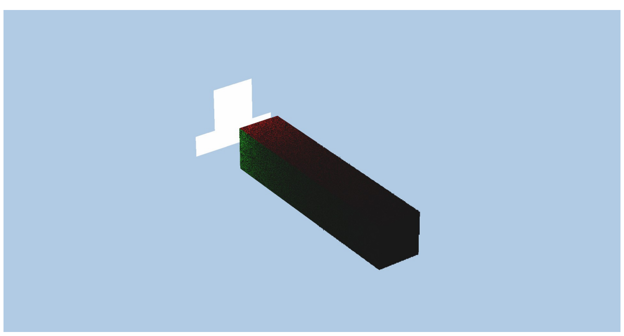
Turner Whitted. “An Improved Illumination Model for Shaded Display”. Bell Laboratories, Holmdel, New Jersey.



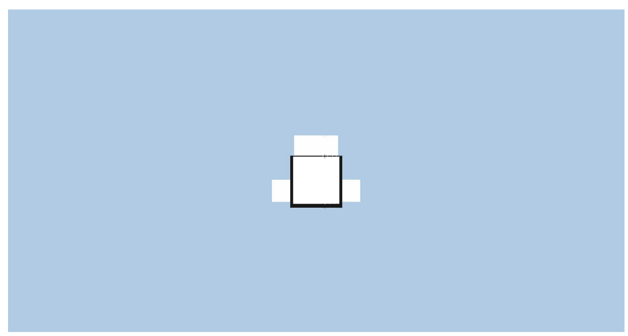


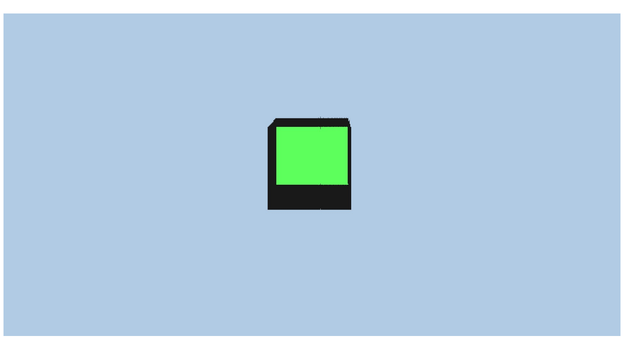


A piece of fiber optic material with two lights, blue and green, produces a mixture of both colors in the end of the material.



Three lights of equivalent power in each of the three primary colors (Red, Green, Blue) result in a white light





The green end of the bent tube. The light will be carried to the end no matter what the shape, so long as it is sealed properly.