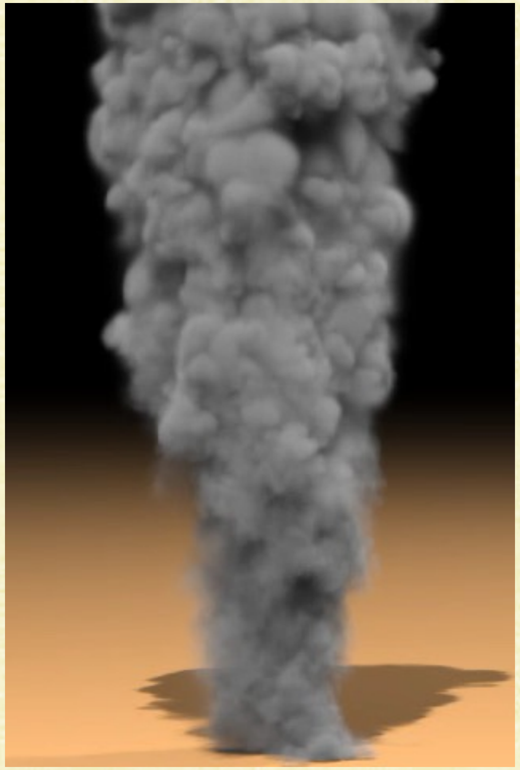
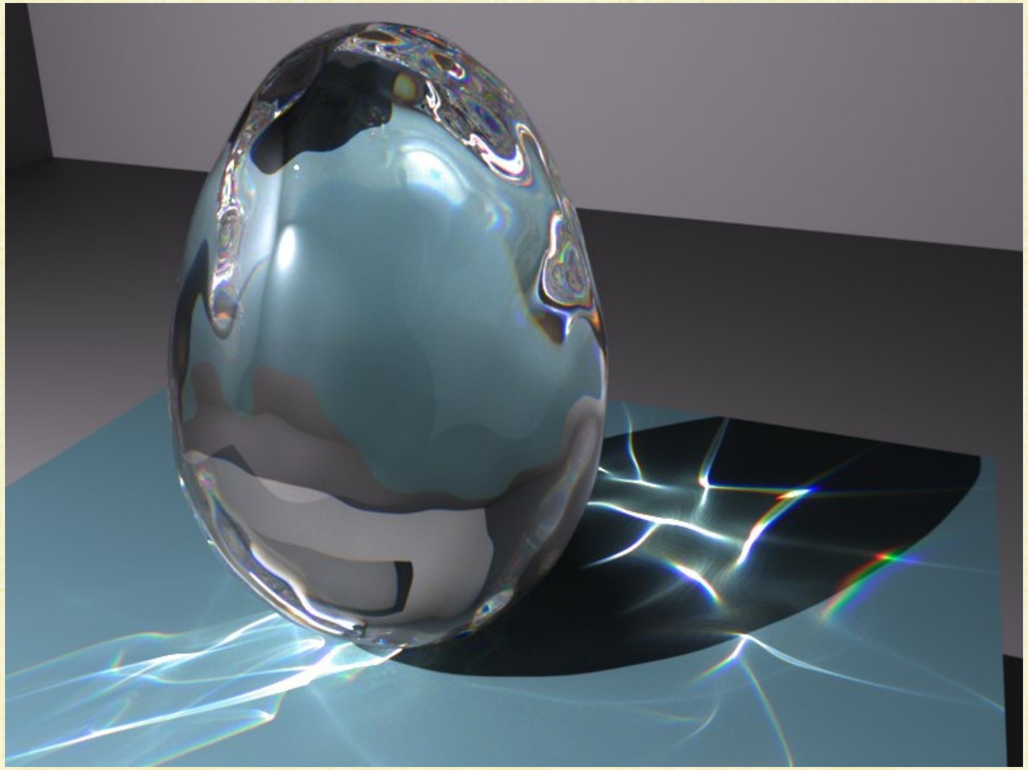
Recursive Ray Tracing

Reflection and Transmission

- A shadow ray is cast to each light source, and the total contribution from all light sources is accumulated:

$$(k_R, k_G, k_B)\left(\sum_{lights} V_{light} I_{light} \max(0, \cos\theta_{light}) + I_{ambient} \prod_{lights} (1 - V_{light})\right)$$

- $V_{light} = 1$ for visible light sources, and $V_{light} = 0$ for occluded light sources
- $I_{ambient}$ is added to fully shadowed regions where $\prod_{lights}(1 - V_{light}) \neq 0$
- To summarize: $(k_R, k_G, k_B)(L_{diffuse} + L_{ambient})$

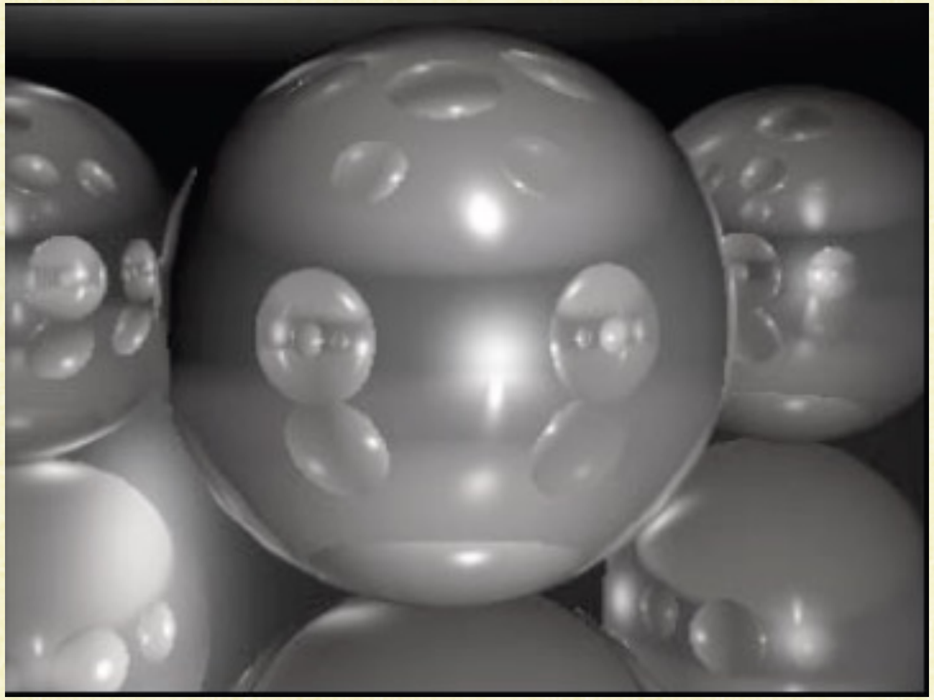
- Mirror-like reflection can also contribute to the color at an intersection point
- Transparency allows other objects to be seen through a surface, allowing those objects to contribute to the color as well
- In summary: $(k_R, k_G, k_B)(L_{diffuse} + L_{ambient}) + L_{reflect} + L_{transmit}$

Scaling Coefficients

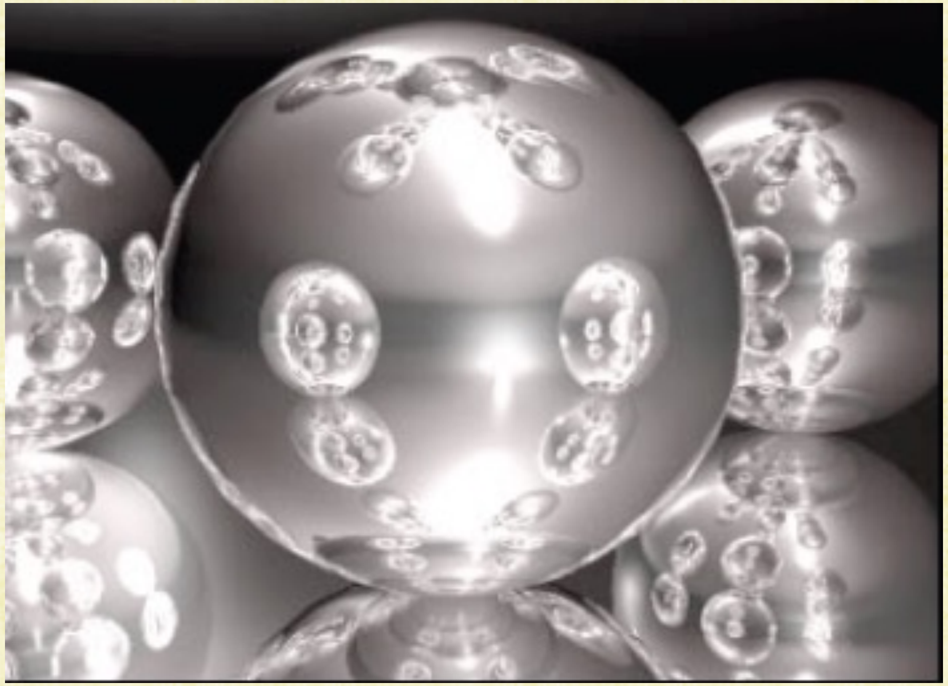
- Scaling coefficients are added in front of every lighting contribution

$$(k_R, k_G, k_B)\left(k_d L_{diffuse} + k_a L_{ambient}\right) + k_r L_{reflect} + k_t L_{transmit}$$

- Coefficients are typically adjusted relative to each other to get the desired “look”
- Then, all the coefficients are scaled together for overall brightness/darkness
- Note: each term adds light to the image, making it brighter (so it might over-saturate)



less reflection (darker)



more reflection (brighter)

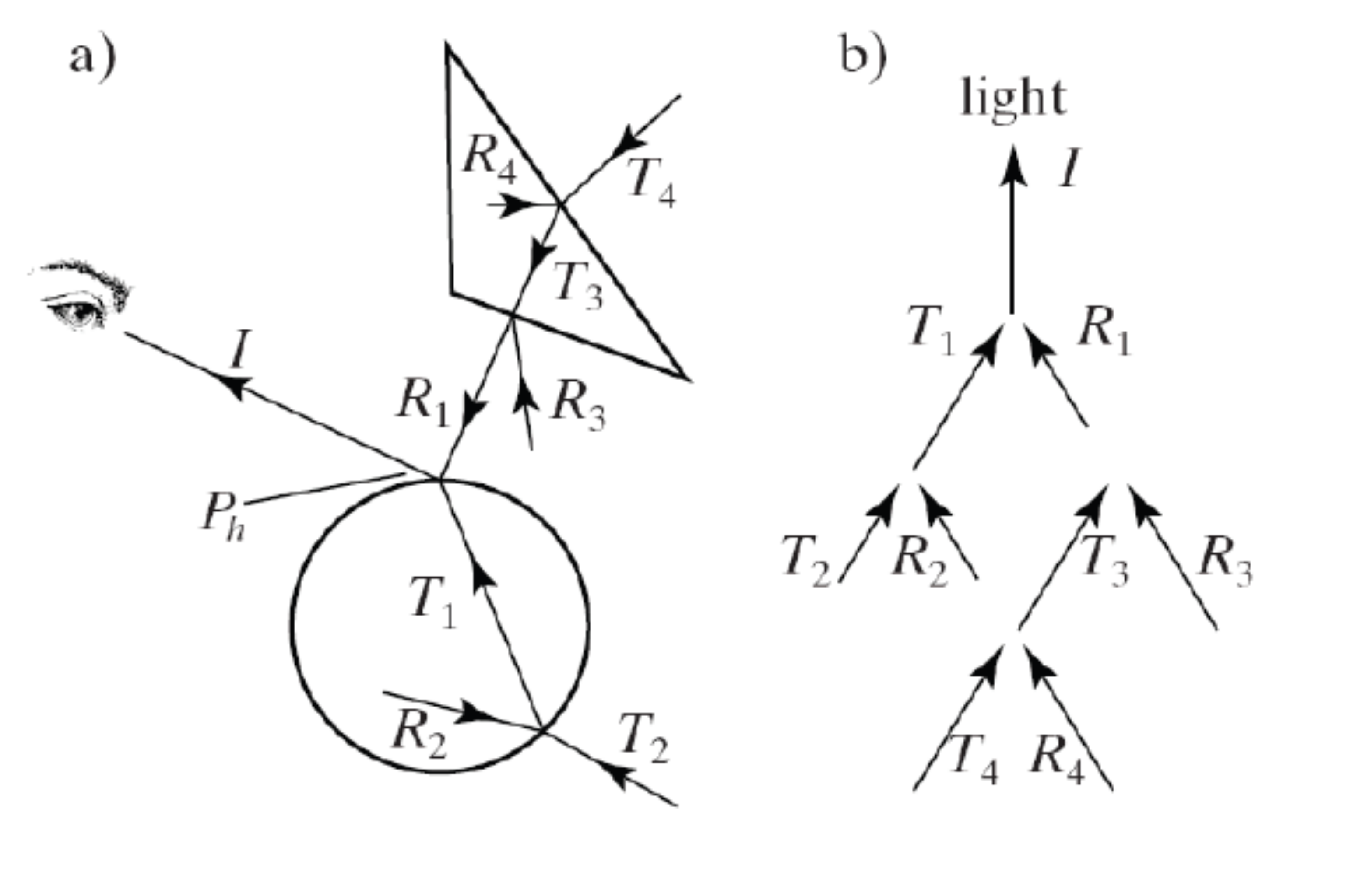
Recursion

- $L_{reflect}$ and $L_{transmit}$ are treated the same way pixel color is treated

- A ray is constructed for the reflection direction and intersected with scene geometry (just like what is done for camera rays through pixels)
  - the result is stored in $L_{reflect}$
- A ray is constructed for the transmission direction and intersected with scene geometry (just like what is done for camera rays through pixels)
  - the result is stored in $L_{transmit}$

- $L_{reflect}$ and $L_{transmit}$ depend on the color computed from the geometry that their rays intersected
- Those intersection points have colors of their own, also computed via: shadow rays, ambient and diffuse shading, and additional reflection and transmission
- Thus, even more rays need to be spawned

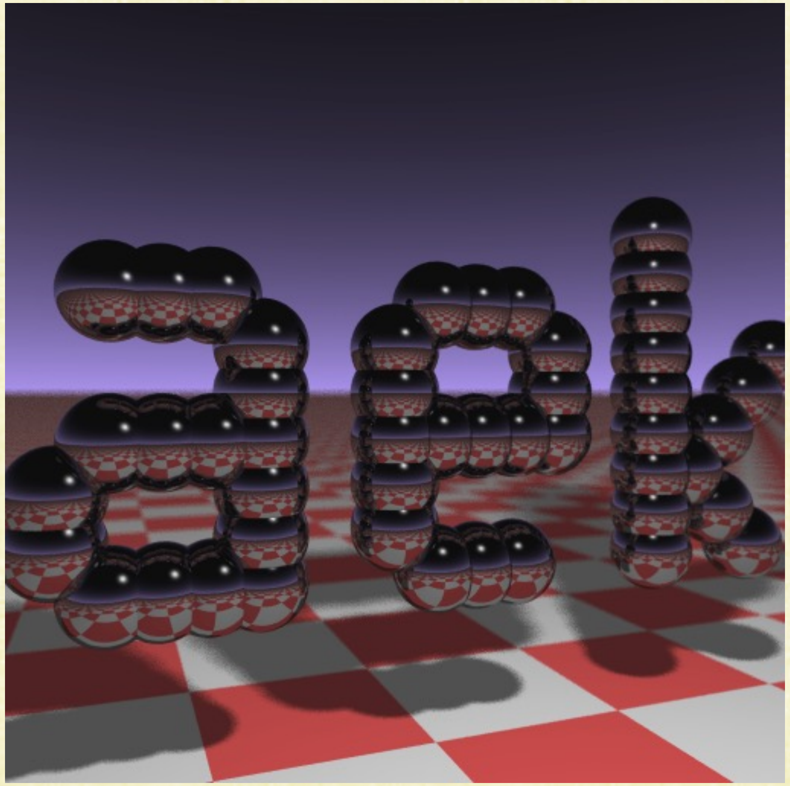
Ray Tree Example



Code Simplicity

- Recursion allows for stunning imagery with minimal code, as demonstrated by these 1337 characters printed on the back of a business card

```
#include <stdlib.h>   // card > aek.ppm
#include <stdio.h>
#include <math.h>
typedef int i;typedef float f;struct v{
f x,y,z;v operator+(v r){return v(x+r.x
,y+r.y,z+r.z);}v operator*(f r){return
v(x*r,y*r,z*r);}f operator%(v r){return
x*r.x+y*r.y+z*r.z;}v(){}v operator^(v r
){return v(y*r.z-z*r.y,z*r.x-x*r.z,x*r.
y-y*r.x);}v(f a,f b,f c){x=a;y=b;z=c;}v
operator!(){return*this*(1/sqrt(*this%*
this));}};i G[]={247570,280596,280600,
249748,18578,18577,231184,16,16};f R(){
return(f)rand()/RAND_MAX;}i T(v o,v d,f
&t,v&n){t=1e9;i m=0;f p=-o.z/d.z;if(.01
<p)t=p,n=v(0,0,1),m=1;for(i k=19;k--;)
for(i j=9;j--;)if(G[j]&1<<k){v p=o+v(-k
,0,-j-4);f b=p%d,c=p%p-1,q=b*b-c;if(q>0
){f s=-b-sqrt(q);if(s<t&&s>.01)t=s,n=!(
p+d*t),m=2;}}return m;}v S(v o,v d){f t
;v n;i m=T(o,d,t,n);if(!m)return v(.7,
.6,1)*pow(1-d.z,4);v h=o+d*t,l=!(v(9+R(
),9+R(),16)+h*-1),r=d+n*(n%d*-2);f b=l%
n;if(b<0||T(h,l,t,n))b=0;f p=pow(l%r*(b
>0),99);if(m&1){h=h*.2;return((i)(ceil(
h.x)+ceil(h.y))&1?v(3,1,1):v(3,3,3))*(b
*.2+.1);}return v(p,p,p)+S(h,r)*.5;}i
main(){printf("P6 512 512 255 ");v g=!v
(-6,-16,0),a=!(v(0,0,1)^g)*.002,b=!(g^a
)*.002,c=(a+b)*-256+g;for(i y=512;y--;)
for(i x=512;x--;){v p(13,13,13);for(i r
=64;r--;){v t=a*(R()-.5)*99+b*(R()-.5)*
99;p=S(v(17,16,8)+t,!(t*-1+(a*(R()+x)+b
*(y+R())+c)*16))*3.5+p;}printf("%c%c%c"
,(i)p.x,(i)p.y,(i)p.z);}}
```



http://fabiensanglard.net/rayTracing_back_of_business_card/

Termination

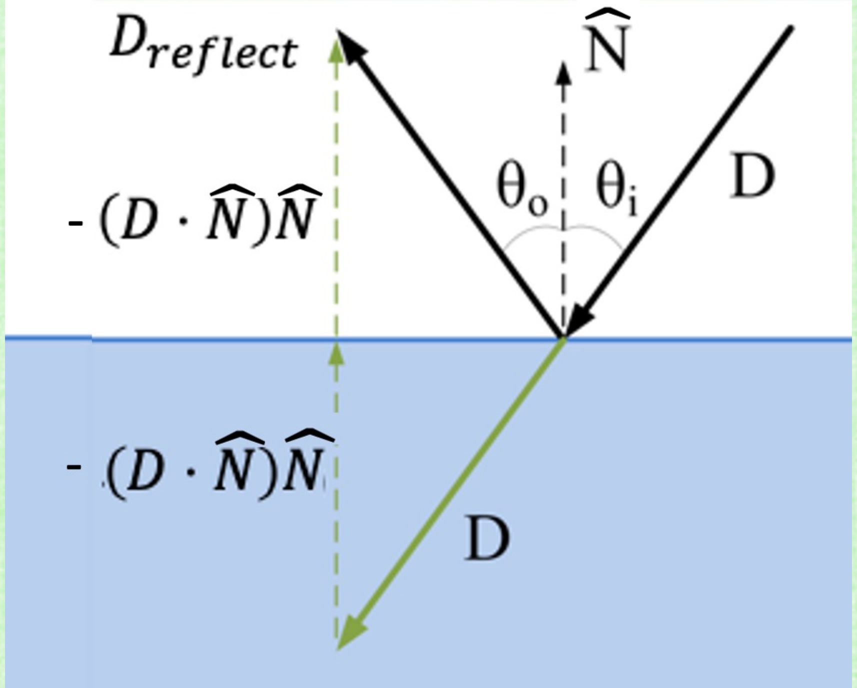
- If every intersected point continued to depend on reflected/transmitted rays, rays would be spawned indefinitely
- Eventually, one hits the recursion limit (depending on hardware) that prevents stack overflow

- If k_d and k_a are frequently nonzero, the reflected/transmitted contributions are eventually diminished enough that one can terminate the recursion (with imperceptible error)
- Terminate by using an arbitrary value for $L_{reflect}$ and/or $L_{transmit}$ (without tracing the associated ray)

- When there is not enough ambient/diffuse lighting (e.g. mirrors, bubbles, etc.), nearly 100% of the lighting is sought recursively via reflected/transmitted rays
- Then, the arbitrary values can show up in the pixel color (which is undesirable)
- So, choose realistic termination colors when possible (common choices: sky color, background color, etc.)

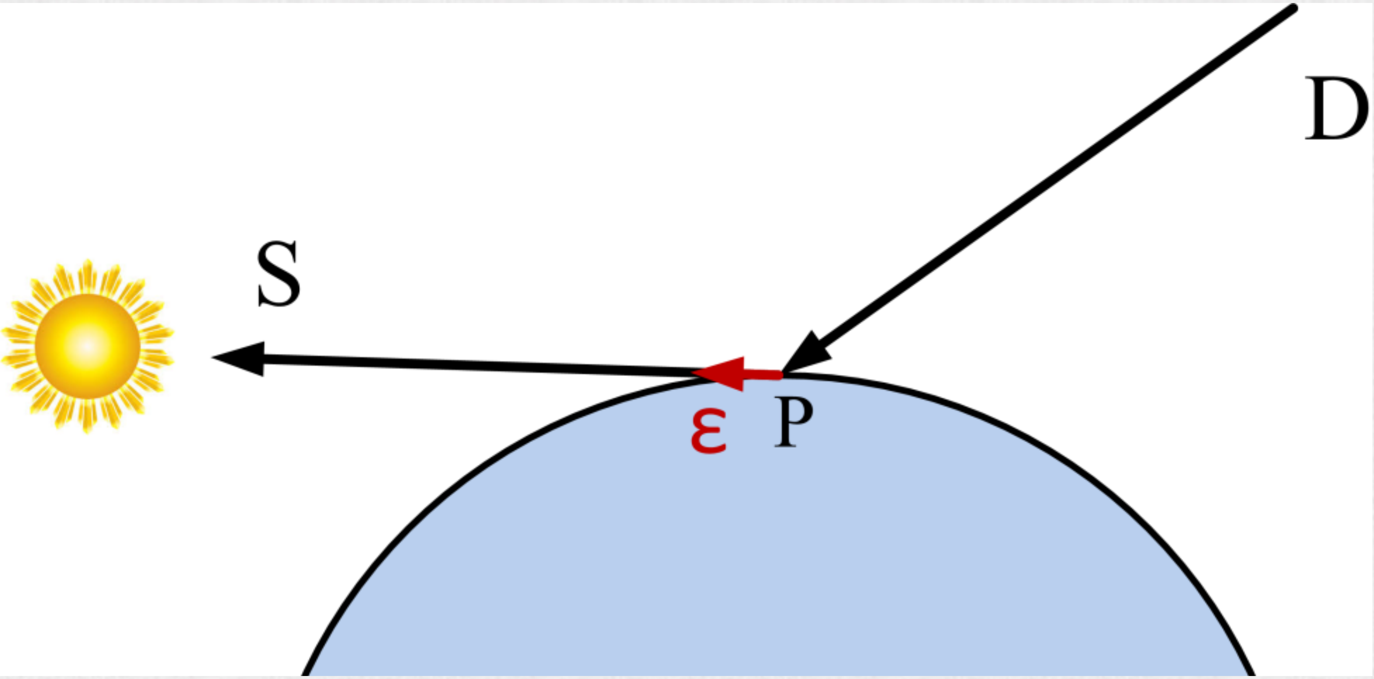
Reflected Ray

- Given an incoming ray $R(t) = A + Dt$, and (outward) unit normal $\hat{N}$, the angle of incidence is defined via $D \cdot \hat{N} = -\|D\|_2 \cos\theta_i$
- Mirror reflection: incoming/outgoing rays make the same angle with $\hat{N}$, i.e. $\theta_o = \theta_i$
  - Note: all the rays and the normal are all coplanar
- Reflected ray direction: $D_{reflect} = D - 2(D \cdot \hat{N})\hat{N}$
- Reflected ray: $R_{reflect}(t) = R(t_{int}) + D_{reflect}t$



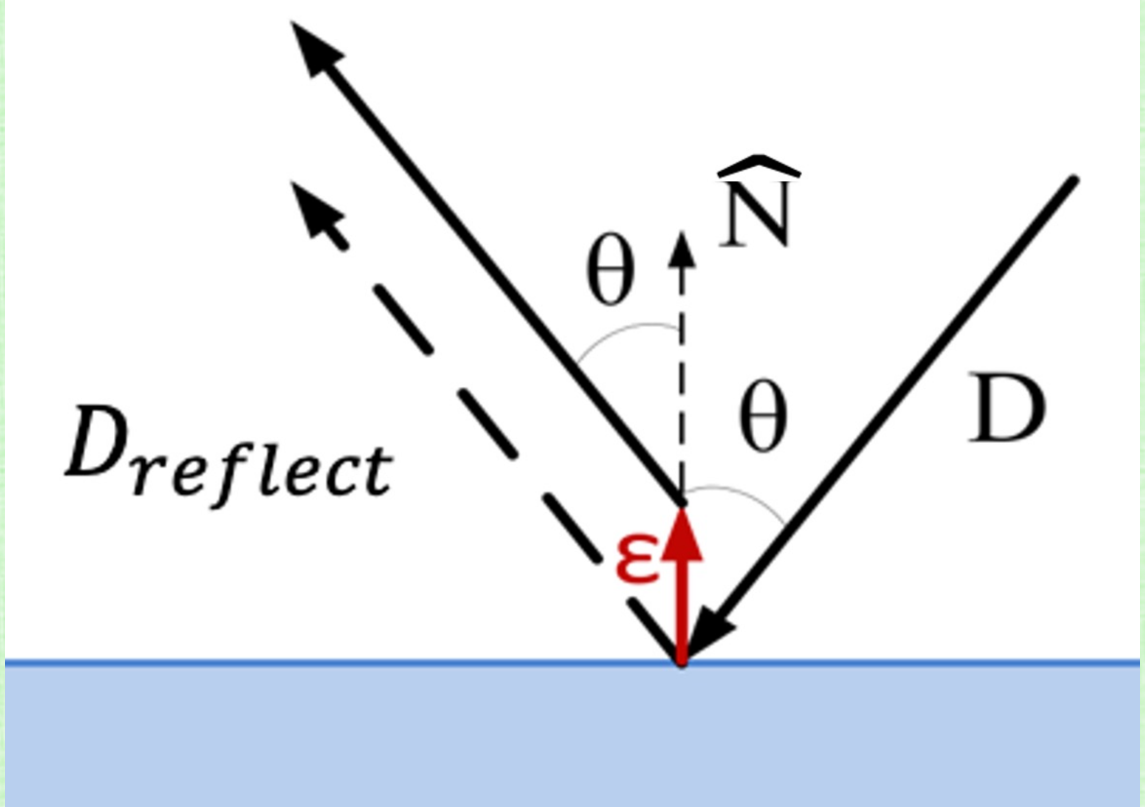
Recall: Spurious Self-Occlusion

- A simple solution is to use $t \in (\epsilon, t_{light})$ for some $\epsilon > 0$ large enough to avoid numerical precision issues
- This works well for many cases
- However, grazing shadow rays may still incorrectly re-intersect the object

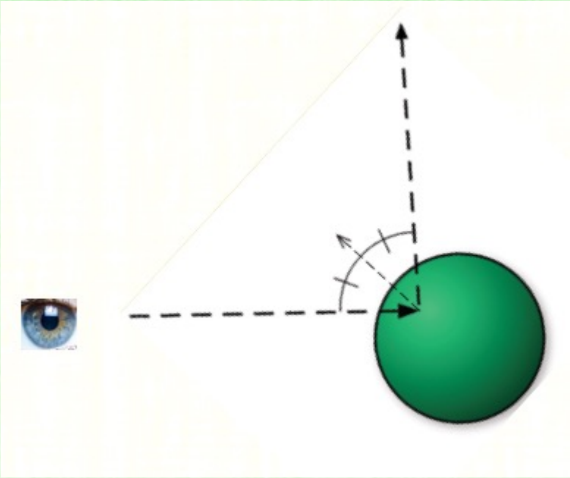


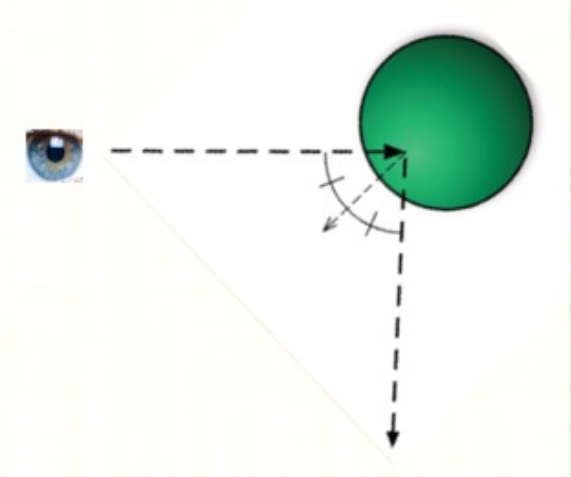
Spurious Self-Occlusion

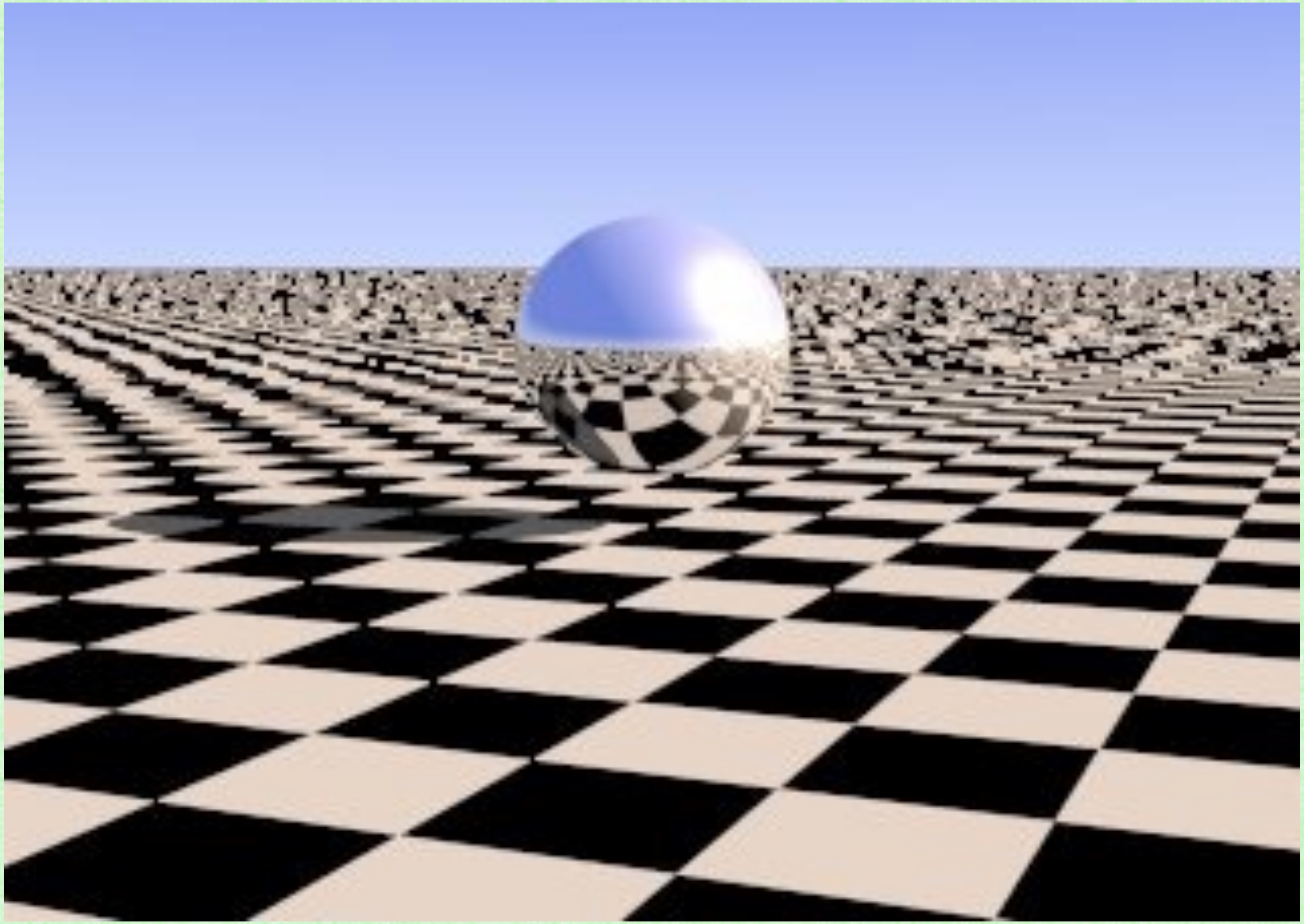
- Perturb the starting point of the reflected ray to $R(t_{int}) + \epsilon\hat{N}$
- The ray direction does not need to be modified (dissimilar to shadow rays)
- The new reflected ray is $R_{reflect}(t) = R(t_{int}) + \epsilon\hat{N} + D_{reflect}t$ with $t \in [0, \infty)$
- Need to be careful that the new starting point isn't inside (or too close to) any other geometry



Reflections





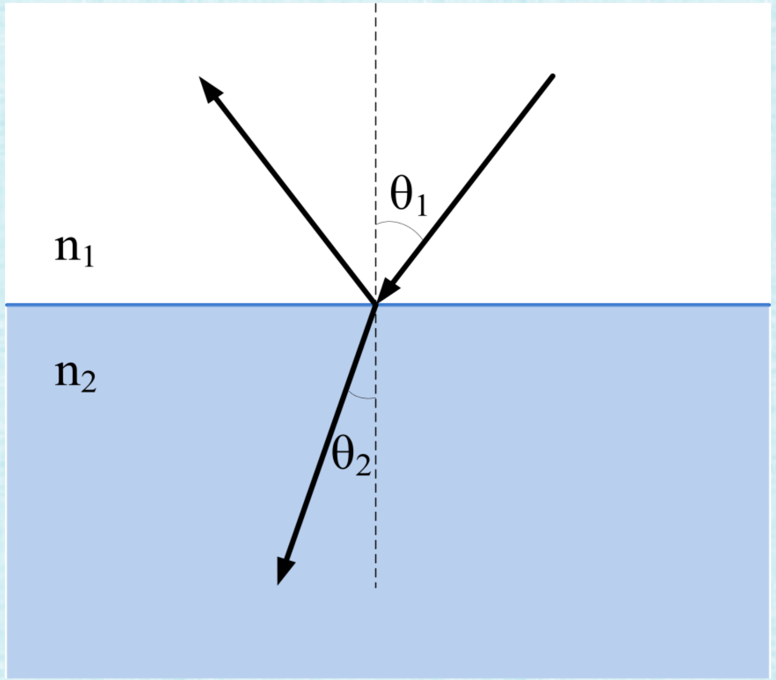


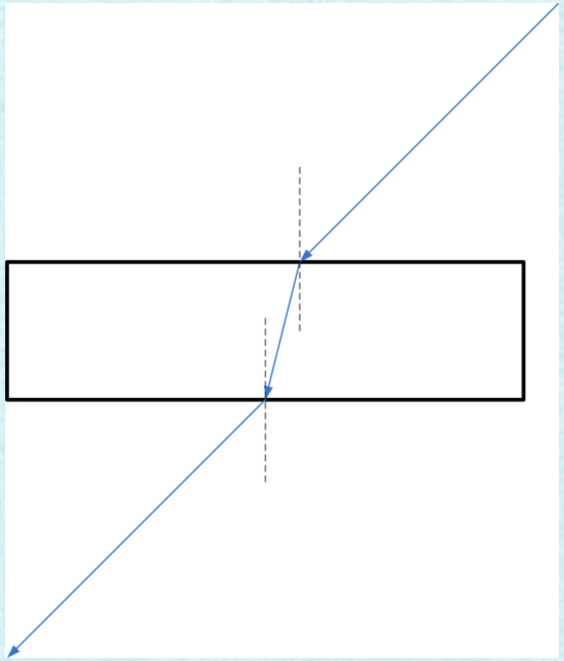
Transmission

- The angle of incidence and angle of transmission (or refraction) are related via Snell's Law:

$$\frac{\sin\theta_1}{\sin\theta_2} = \frac{v_1}{v_2} = \frac{n_2}{n_1}$$

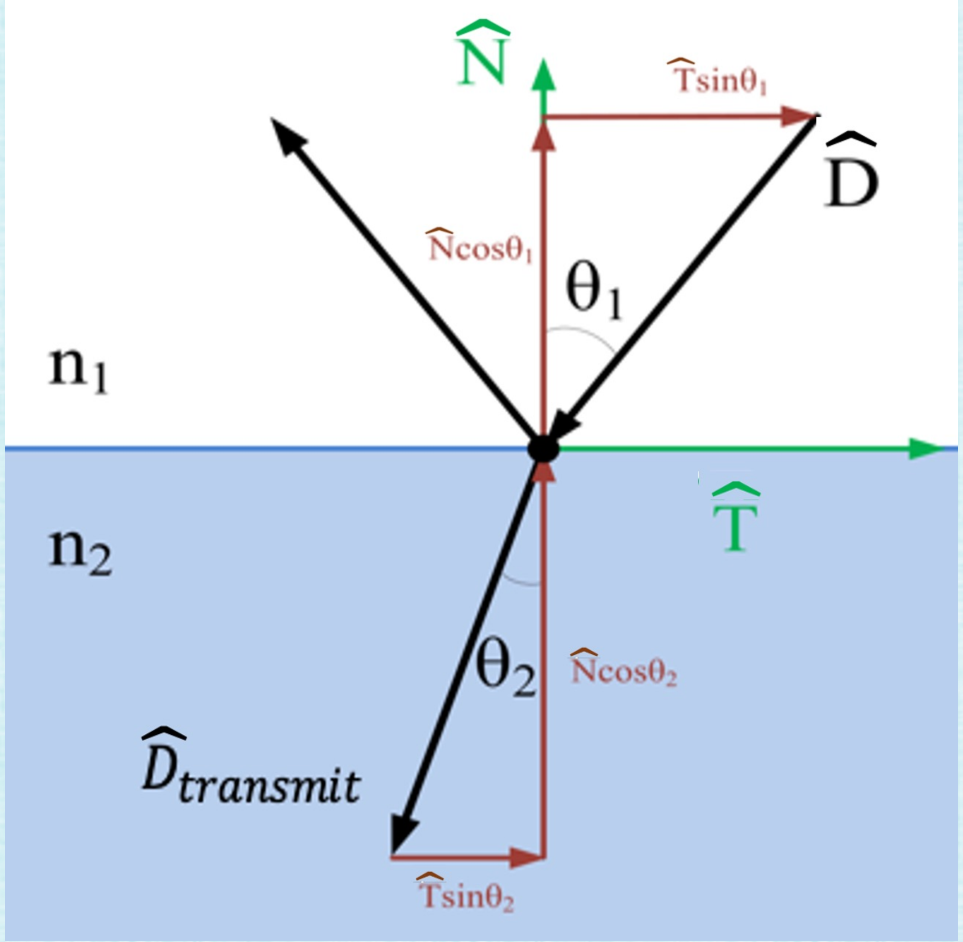
- Incoming/outgoing angles: θ_1, θ_2; phase velocities: v_1, v_2; indices of refraction: n_1, n_2





Transmitted Ray

- $\hat{D}$ is the (unit) incoming ray direction, $\hat{N}$ is the (outward) unit normal, and $\hat{T}$ is the unit tangent in the plane of $\hat{D}$ and $\hat{N}$, so that $\hat{D} + \hat{N}cos\theta_1 + \hat{T}sin\theta_1 = 0$
- $\hat{D}_{transmit}$ is the (unit) transmitted ray direction, so $\hat{D}_{transmit} + \hat{T}sin\theta_2 + \hat{N}cos\theta_2 = 0$



Transmitted Ray

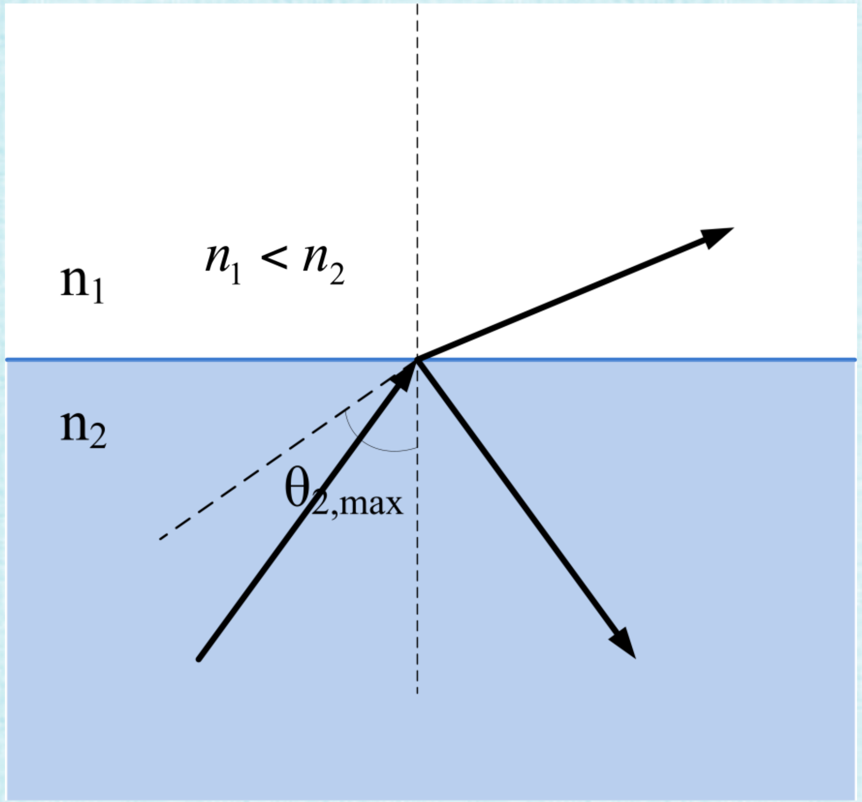
- $\hat{D}_{transmit} = -\hat{T}sin\theta_2 - \hat{N}cos\theta_2 = (\hat{D} + \hat{N}cos\theta_1)\frac{sin\theta_2}{sin\theta_1} - \hat{N}\sqrt{1 - \sin^2\theta_2}$
- Using Snell's Law: $\hat{D}_{transmit} = (\hat{D} + \hat{N}cos\theta_1)\frac{n_1}{n_2} - \hat{N}\sqrt{1 - \left(\frac{n_1}{n_2}sin\theta_1\right)^2}$
- $\hat{D}_{transmit} = \hat{D}\frac{n_1}{n_2} + \hat{N}\left(\frac{n_1}{n_2}cos\theta_1 - \sqrt{1 - \left(\frac{n_1}{n_2}\right)^2(1 - \cos^2\theta_1)}\right)$
- $cos\theta_1 = -\hat{D}\cdot\hat{N}$ leads to $\hat{D}_{transmit} = \hat{D}\frac{n_1}{n_2} - \hat{N}\left(\frac{n_1}{n_2}\hat{D}\cdot\hat{N} + \sqrt{1 - \left(\frac{n_1}{n_2}\right)^2\left(1 - (\hat{D}\cdot\hat{N})^2\right)}\right)$

- When the term under the square root is negative, there is no transmitted ray (all the light is reflected, i.e. total internal reflection)

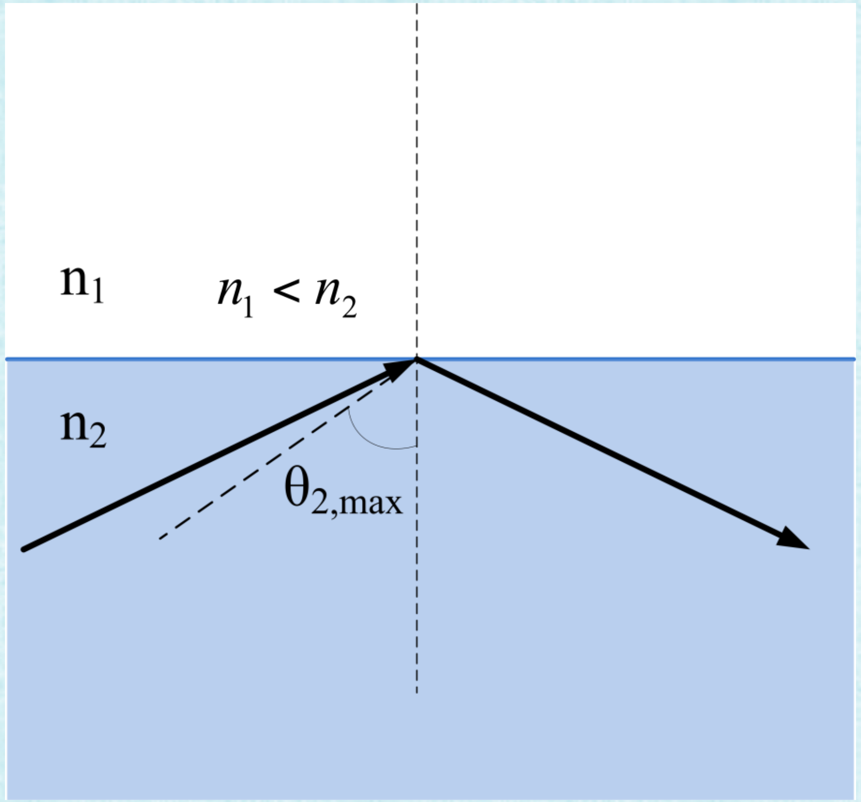
- Note: This equation works regardless of whether n_1 or n_2 is bigger
- Note: Add $\epsilon > 0$ to avoid self intersection, or offset in the negative normal direction (while avoiding other nearby geometry, etc.)

Total Internal Reflection

- When light goes from a higher index of refraction to lower index of refraction, no light is transmitted when the incident angle exceeds a critical angle
- In such a case, all the light reflects



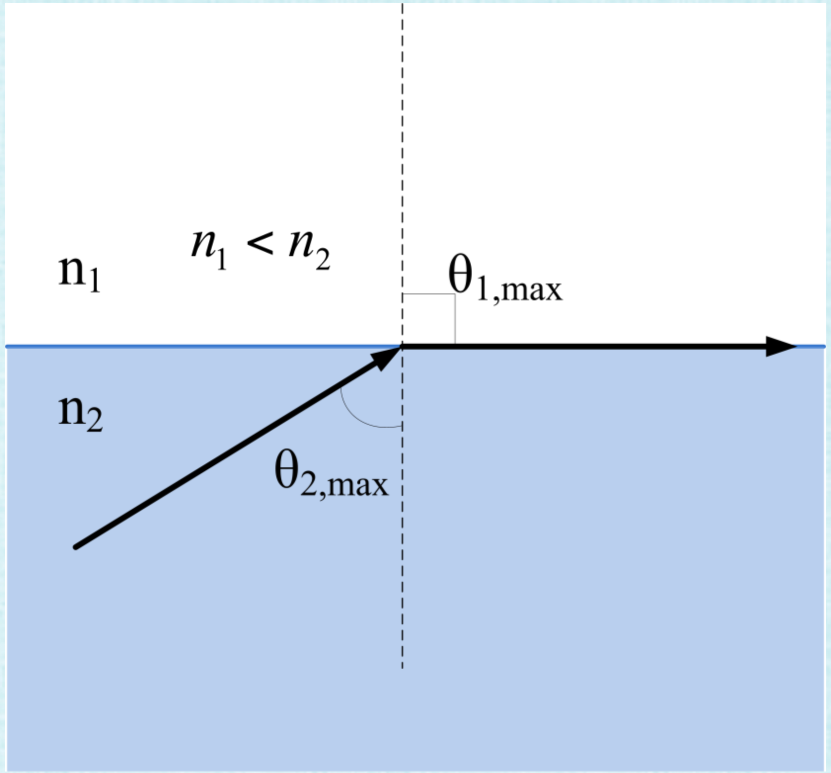
when $\theta_2 < \theta_{2,max}$, both reflection and transmission occur



when $\theta_2 > \theta_{2,max}$, only reflection occurs

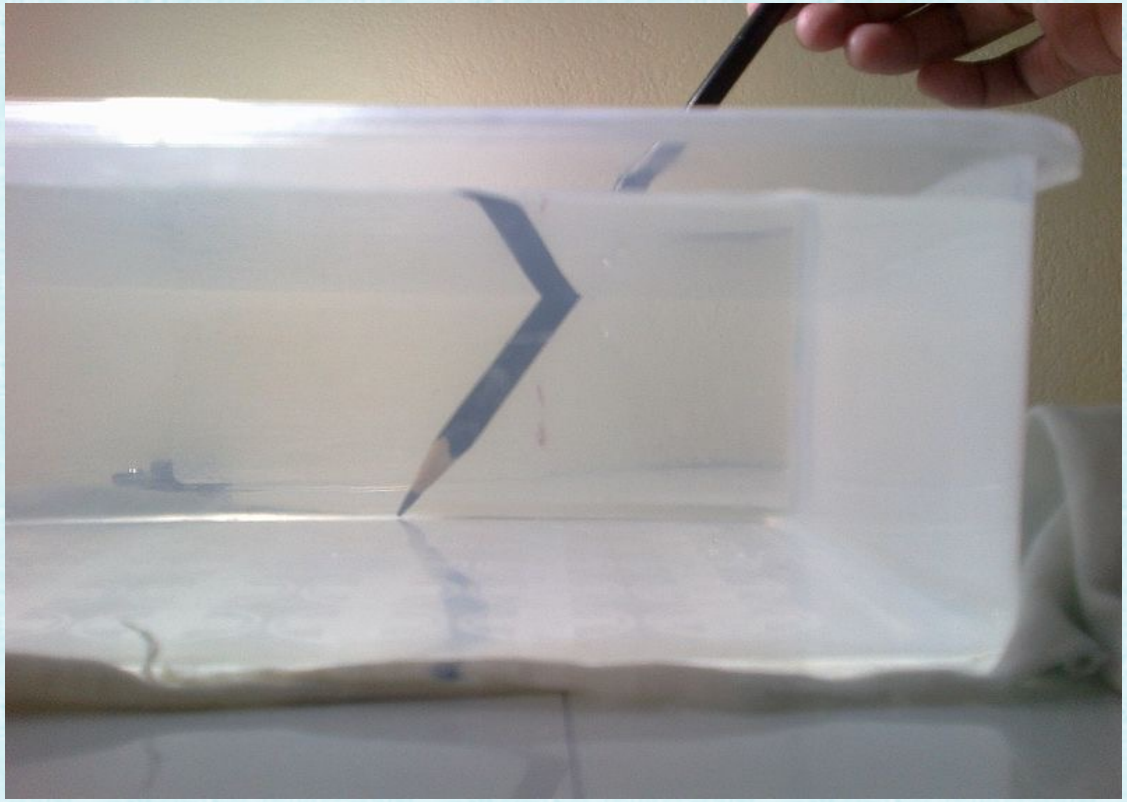
Critical Angle

- $\theta_1 = \frac{\pi}{2}$ is the maximum angle for transmission
- $\sin\left(\frac{\pi}{2}\right) = 1$ and Snell's Law becomes $\frac{1}{\sin\theta_2} = \frac{n_2}{n_1}$ or $\theta_2 = \arcsin\left(\frac{n_1}{n_2}\right)$
  - Note: this can only occur when $n_1 < n_2$



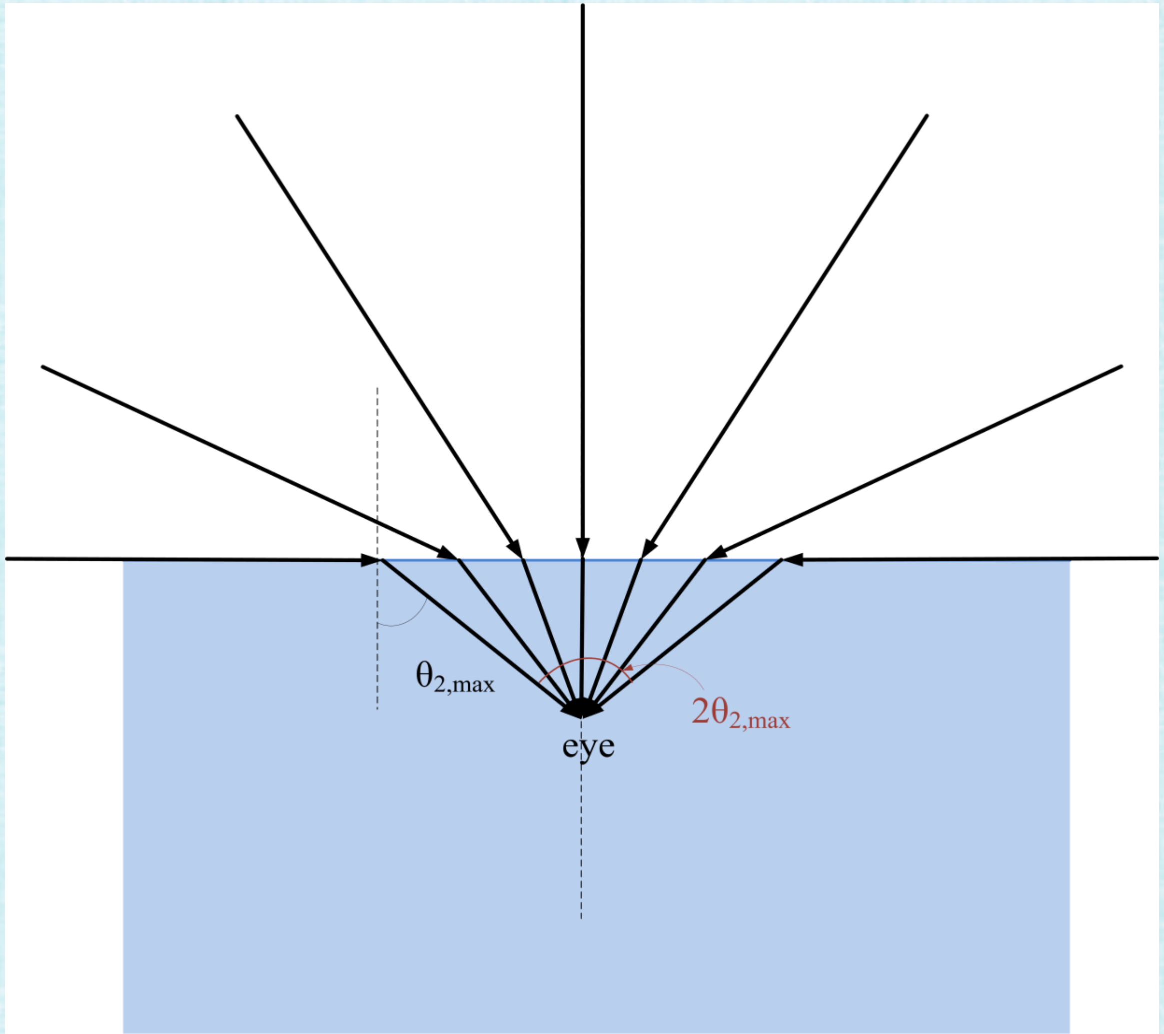
Total Internal Reflection

- Responsible for many interesting and impressive visuals in both glass and water



Snell's Window

- Yes, fish can see you standing on the shore!



Snell's Window

Reflection vs. Transmission

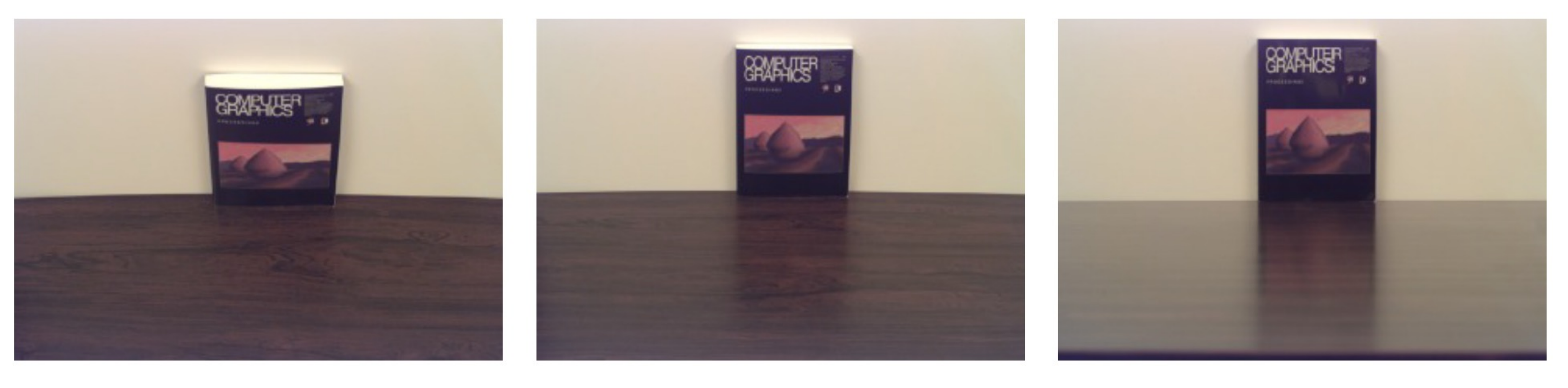
- The amount of transmission vs. reflection decreases as the viewing angle goes from perpendicular (overhead) to parallel (grazing)

Perpendicular (overhead) view:
more transmission, less reflection

Parallel (grazing) view:
more reflection, less transmission

Reflection vs. Transmission

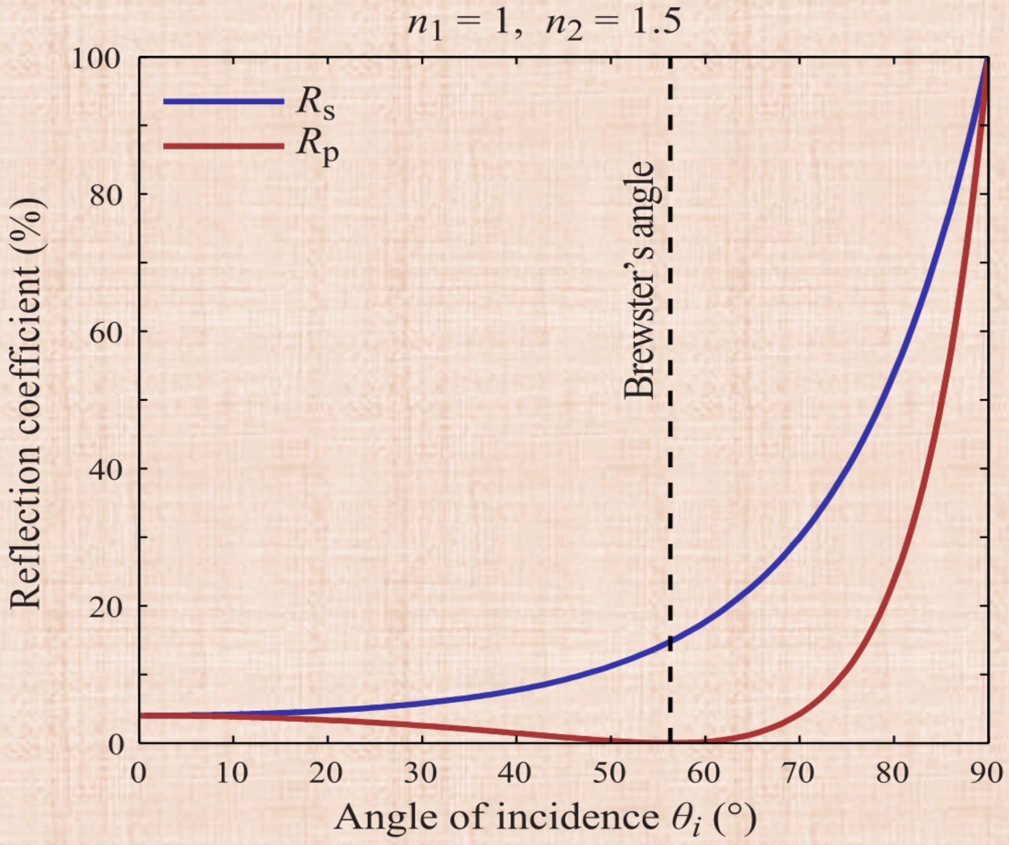
- Even for opaque objects (that lack transmission), reflection behaves similarly



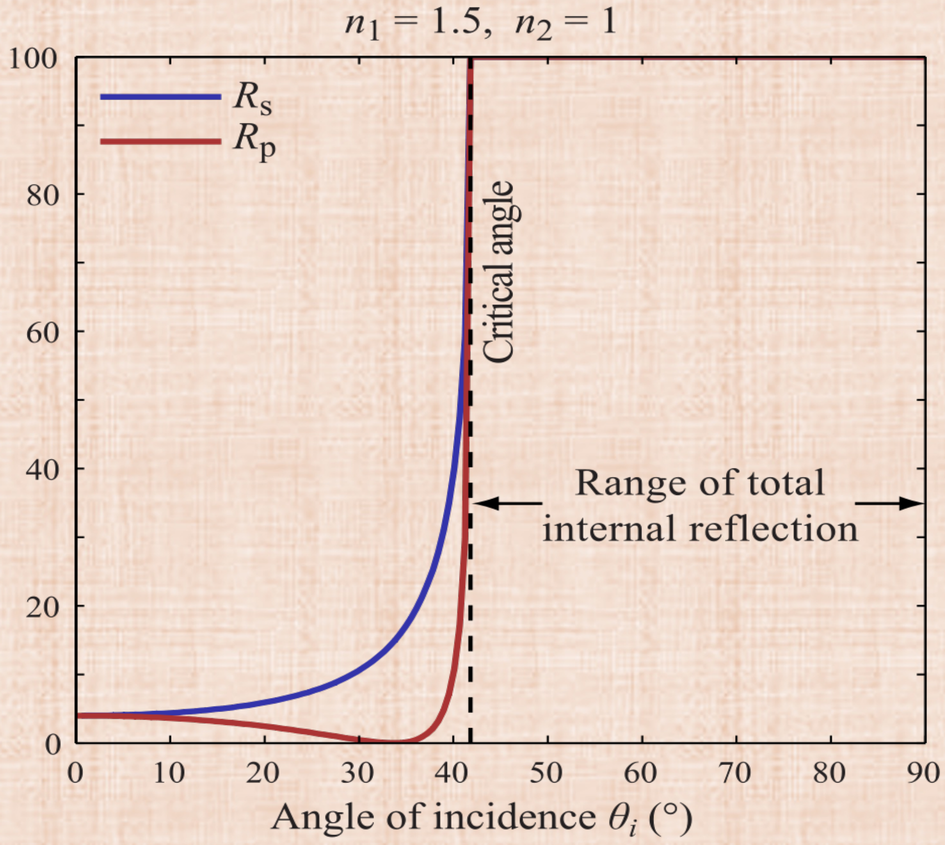
As the viewing angle changes from overhead to a grazing angle (from left to right), the amount of reflection off of the table increases (one can better see the book's reflection)

Fresnel Equations

- The proportion of reflection gradually increases as the viewing angle goes from perpendicular (coincident with the normal) to parallel (orthogonal to the normal)



Light entering a denser material
(e.g. from air into water)



Light leaving a denser material
(e.g. exiting water into air)

Fresnel Equations

- Light is polarized into 2 parts, based on whether the plane containing the incident, reflected, refracted rays is parallel (p-polarized) or perpendicular (s-polarized) to the electric field
- The Fresnel equations approximate the fraction of light reflected as:

$$R_p = \left| \frac{n_1 \cos\theta_t - n_2 \cos\theta_i}{n_1 \cos\theta_t + n_2 \cos\theta_i} \right|^2 \qquad R_s = \left| \frac{n_1 \cos\theta_i - n_2 \cos\theta_t}{n_1 \cos\theta_i + n_2 \cos\theta_t} \right|^2$$

- Transmission (if it occurs) is calculated as the remaining light:

$$T_p = 1 - R_p \qquad T_s = 1 - R_s$$

- For <u>unpolarized</u> light (a typical assumption in ray tracing), assume:

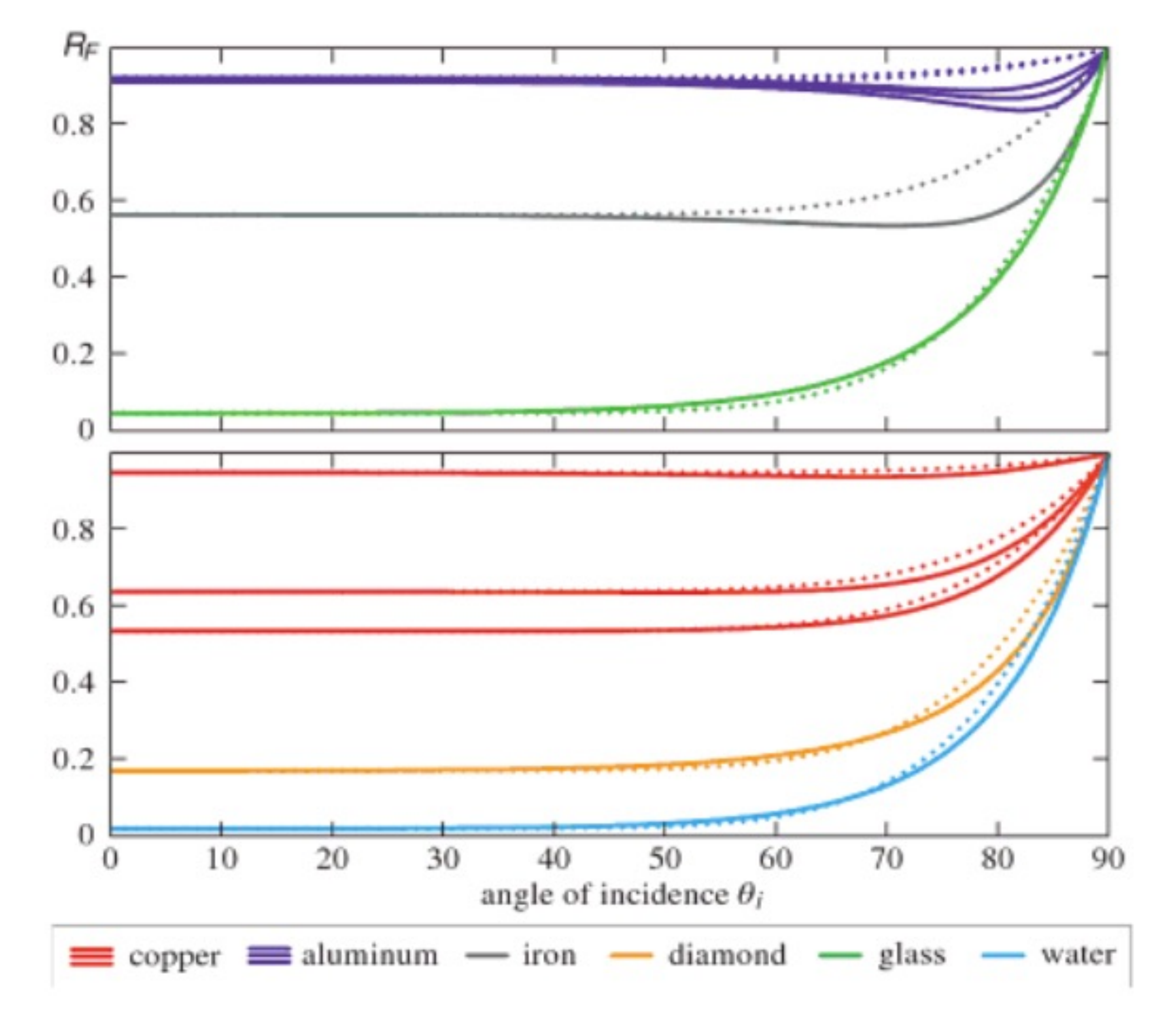
$$R = \frac{R_p + R_s}{2} \qquad T = 1 - R$$

Schlick's Approximation

- Approximate reflection via:

$$R(\theta_i) = R_0 + (1 - R_0)(1 - \cos\theta_i)^5$$

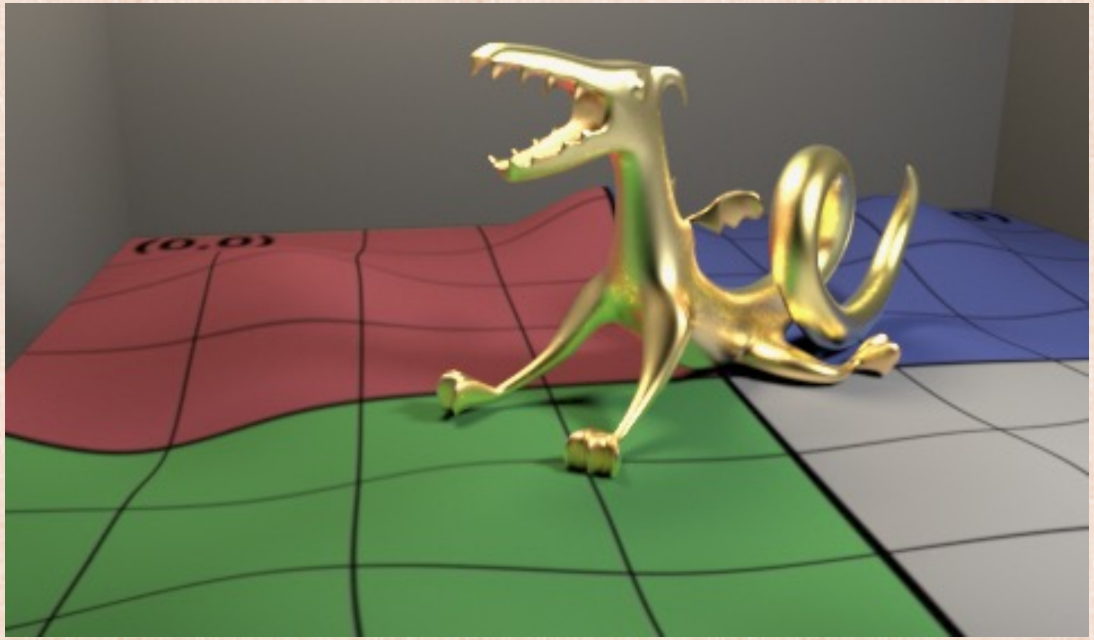
$$R_0 = \left(\frac{n_1 - n_2}{n_1 + n_2}\right)^2$$



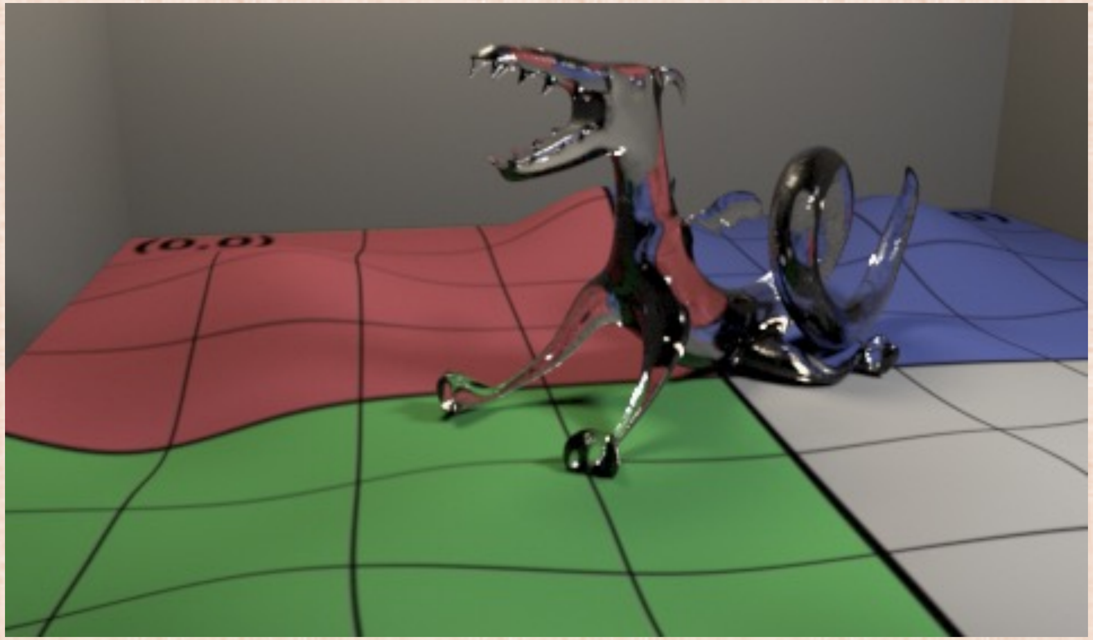
Fresnel (solid lines)
Schlick (dotted lines)

Conductors vs. Dielectrics

- Conductors (of electricity, e.g. metals) mostly reflect light (low absorption, no transmission)
- The amount reflected doesn't change much with viewing angle
  - see copper, aluminum, iron on the last slide
- Thus, k_r can be approximated as a constant (independent of viewing direction) for conductors
- In contrast, k_r varies significantly with viewing angle for dielectrics (e.g. glass, water)



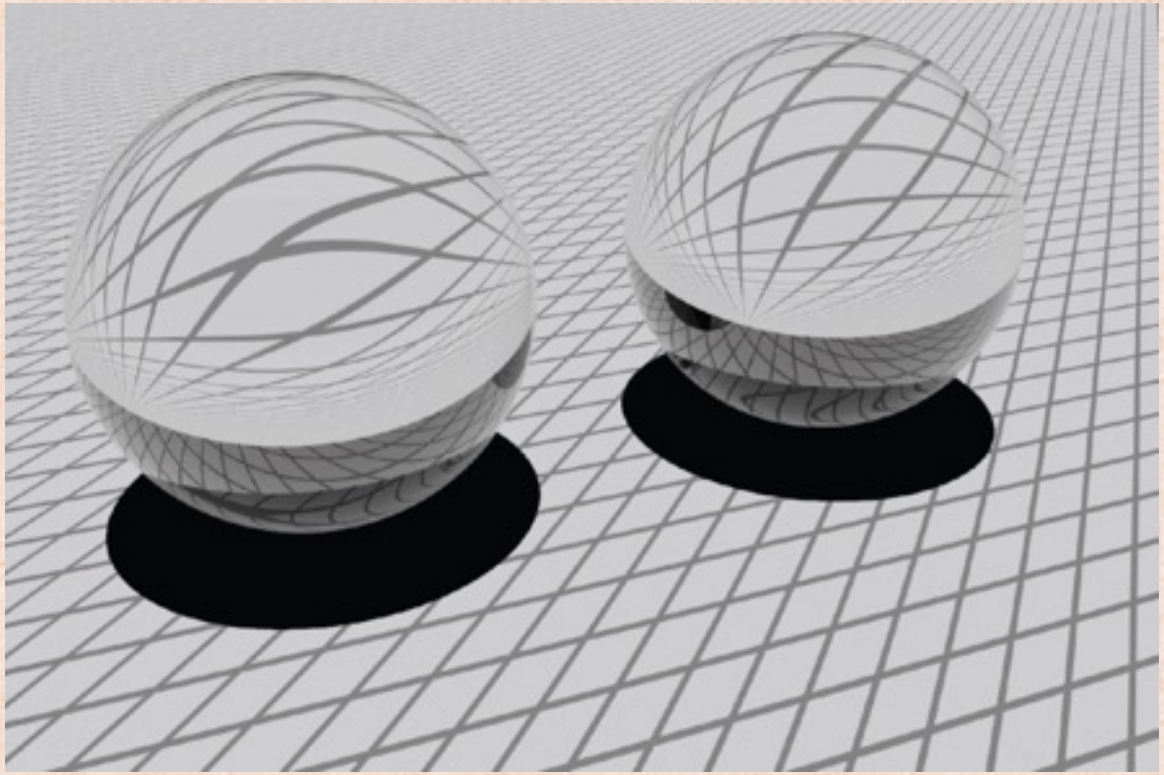
Conductor



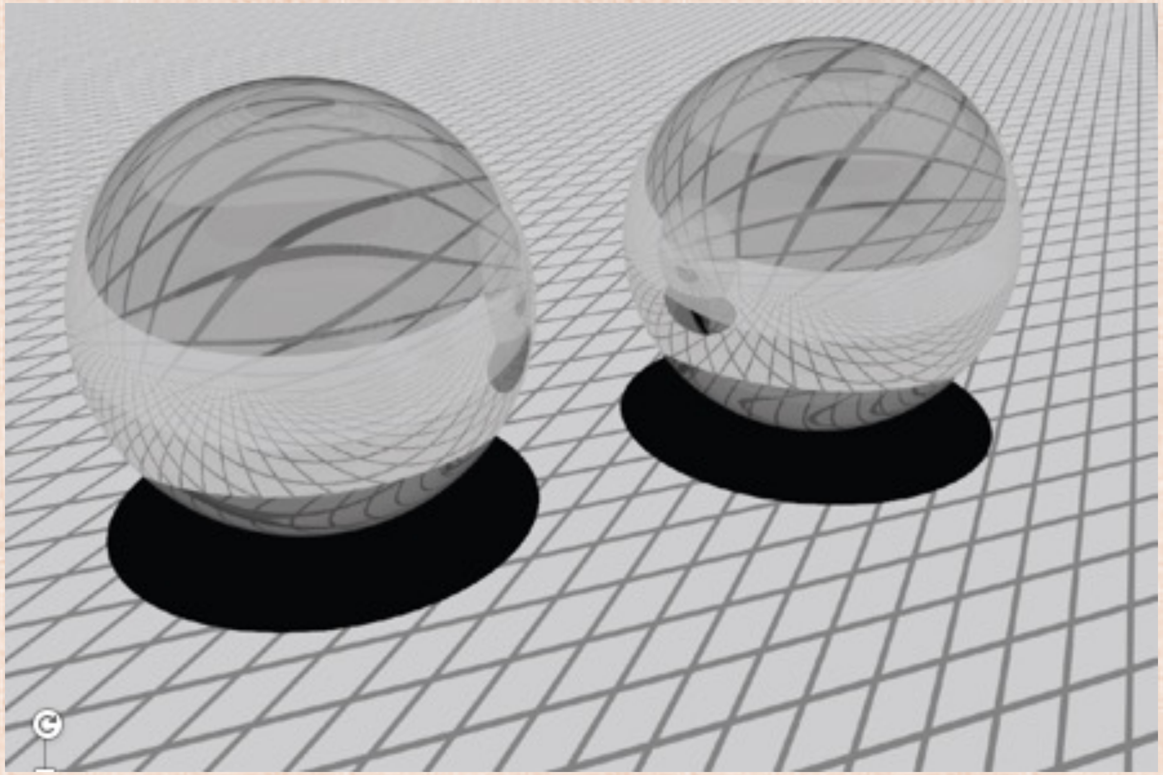
Dielectric

Curved Surfaces

- The viewing angle can vary (from perpendicular to parallel) across the surface of an object
- The amount of reflection vs. transmission similarly varies
- Capturing this is especially important for dielectrics



Correct reflection vs. transmission
(based on viewing angle)



Incorrect reflection vs. transmission
(no dependence on viewing angle)

Attenuation

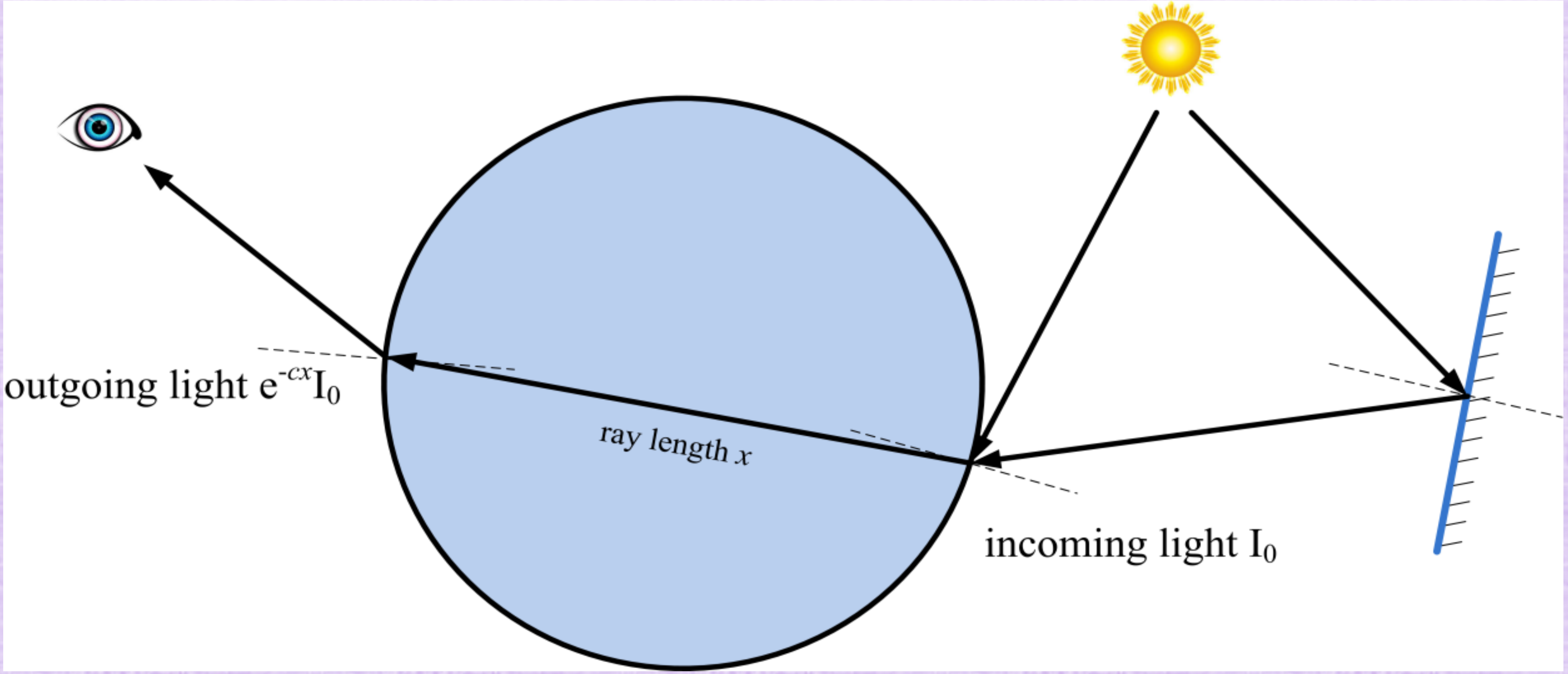
- Light is absorbed and scattered as it travels through material
- This attenuates the amount of light traveling along a straight line
- The amount of attenuation depends on the distance traveled (through the material)
- Different colors (actually, different wavelengths) are attenuated at different rates

Example:

- Shallow water is clear (almost no attenuation)
- Deeper water attenuates all the red light and looks bluish-green
- Even deeper water attenuates all the green light too, and looks dark blue
- Eventually all the light attenuates, and the color ranges from blackish-blue to black

Beer's Law

- For homogeneous media, attenuation can be approximated by Beer's Law
- Light with intensity I is attenuated over a distance x via the Ordinary Differential Equation (ODE): $\frac{dI}{dx} = -cI$ where c is the attenuation coefficient
- This ODE has an exact solution: $I(x) = I_o e^{-cx}$ where I_o is the original amount of light

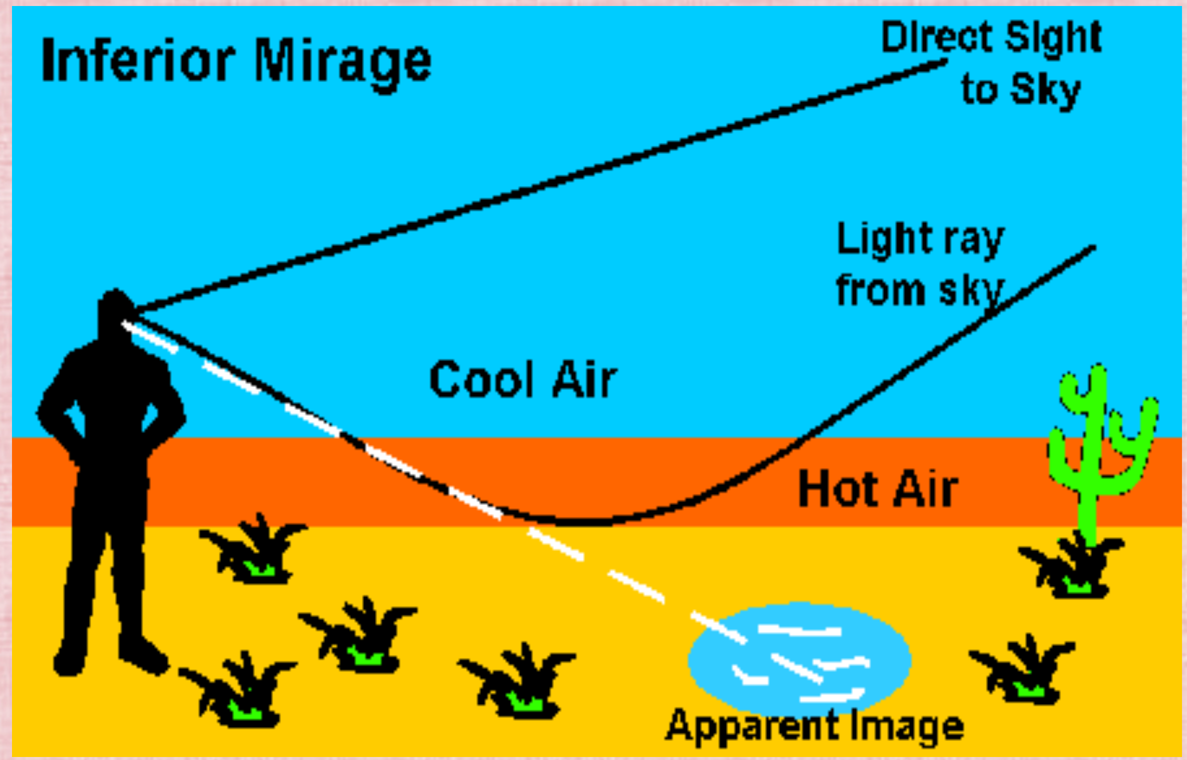


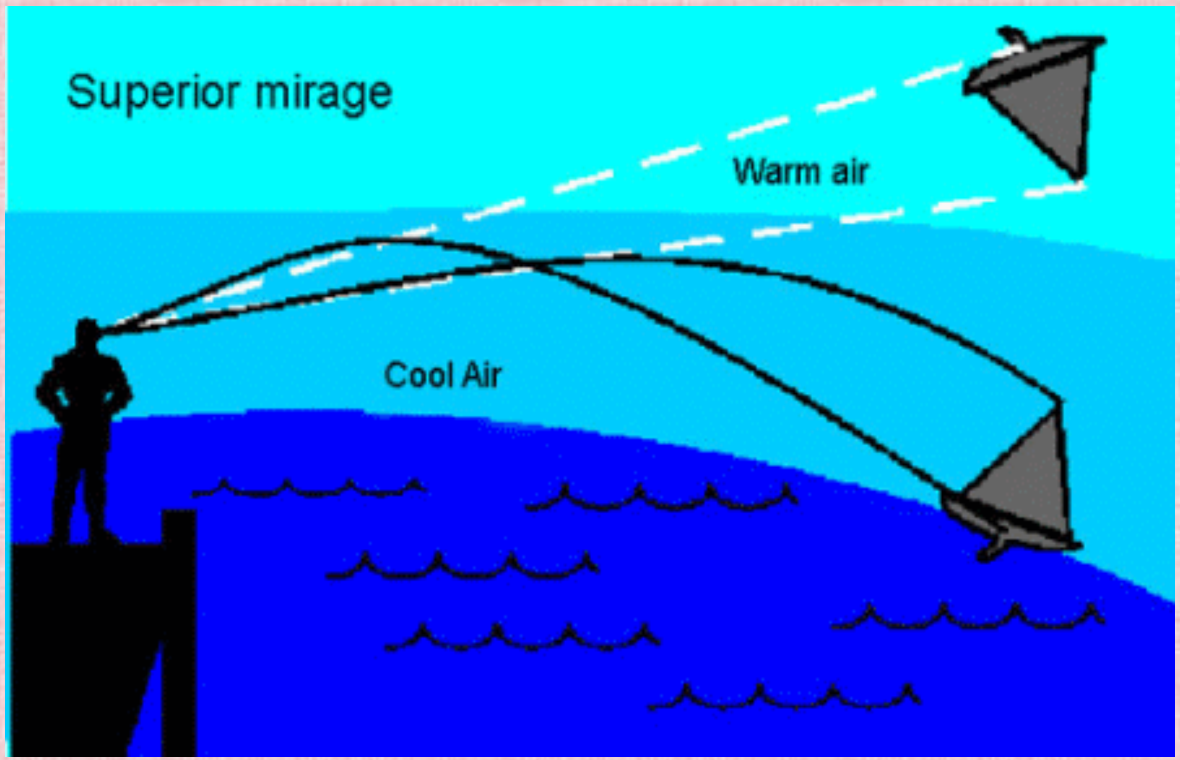
Beer’s Law

- The color of a transparent object is described by three attenuation coefficients: c_R, c_G, c_B
- Shadow rays are also attenuated

Atmospheric Refraction

- Light continuously bends (following a curved path) as it passes through varying temperature gases (with varying density)
- The density variations cause similar variations in the index of refraction



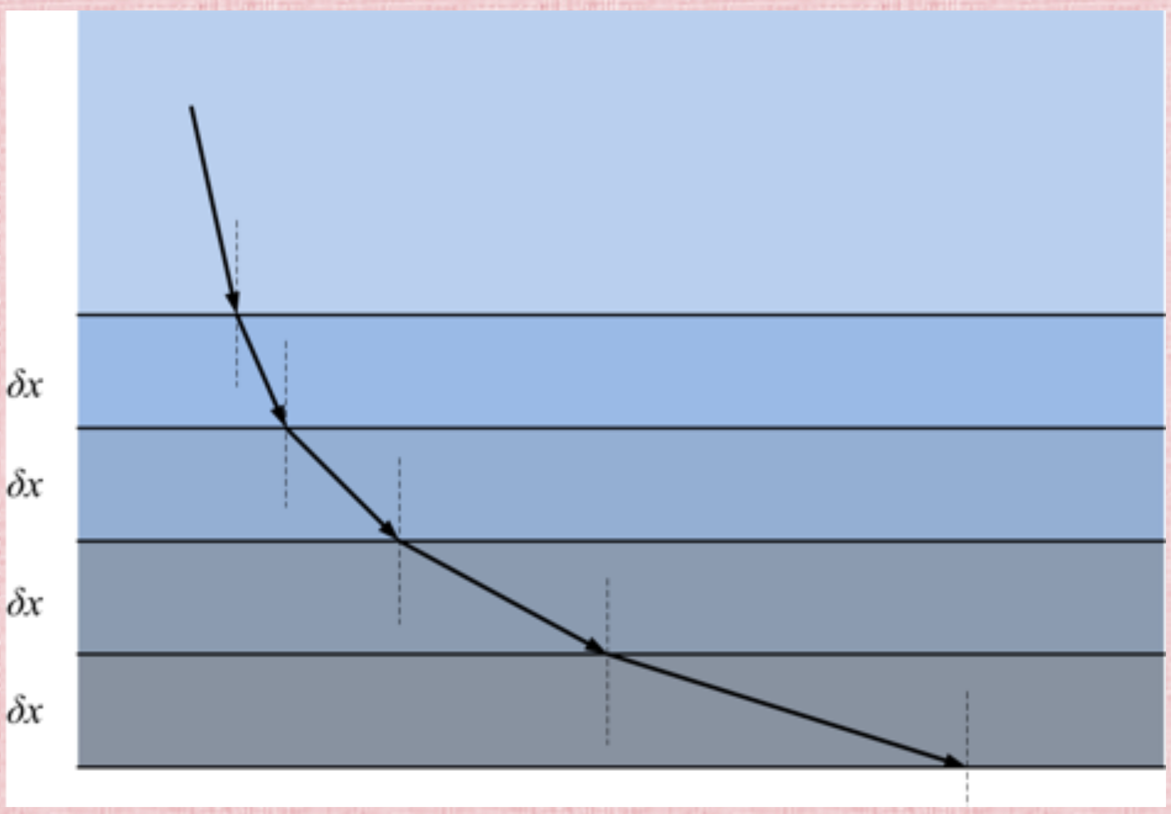


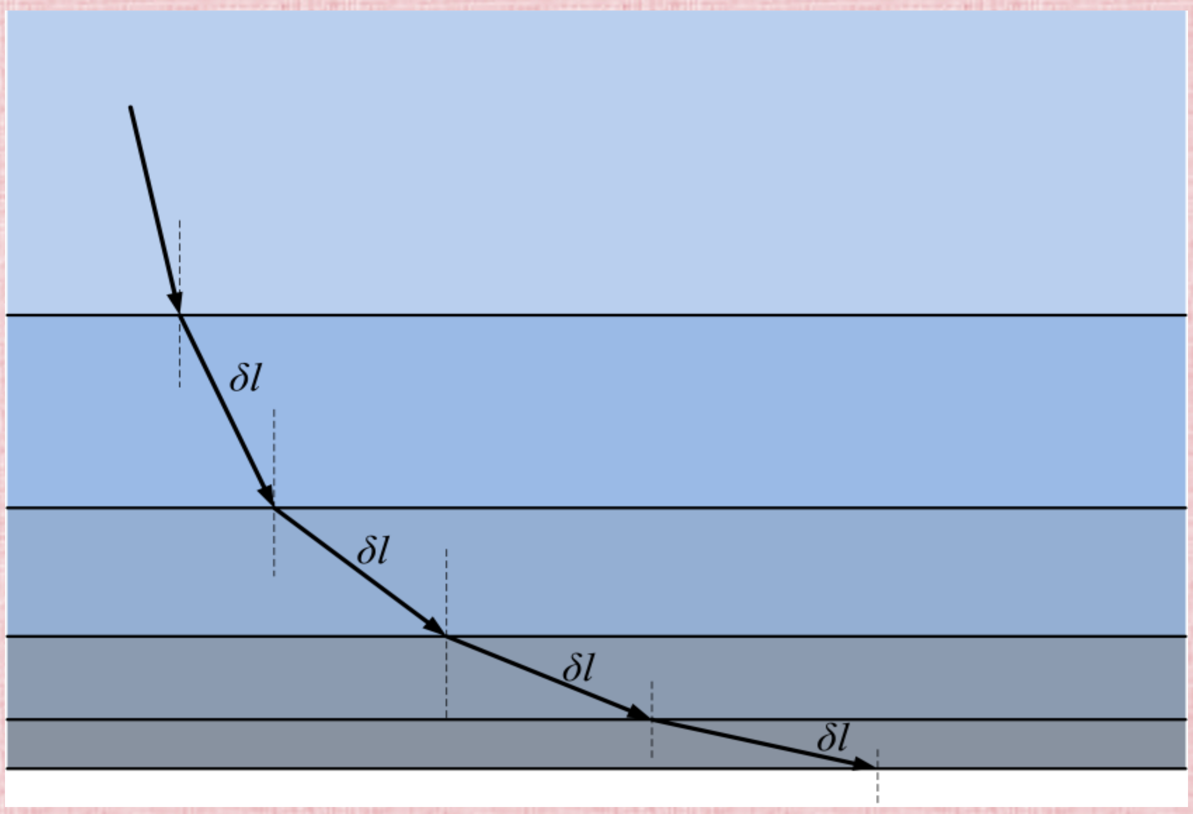
Inferior Mirage

Superior Mirage (March 2021, England)

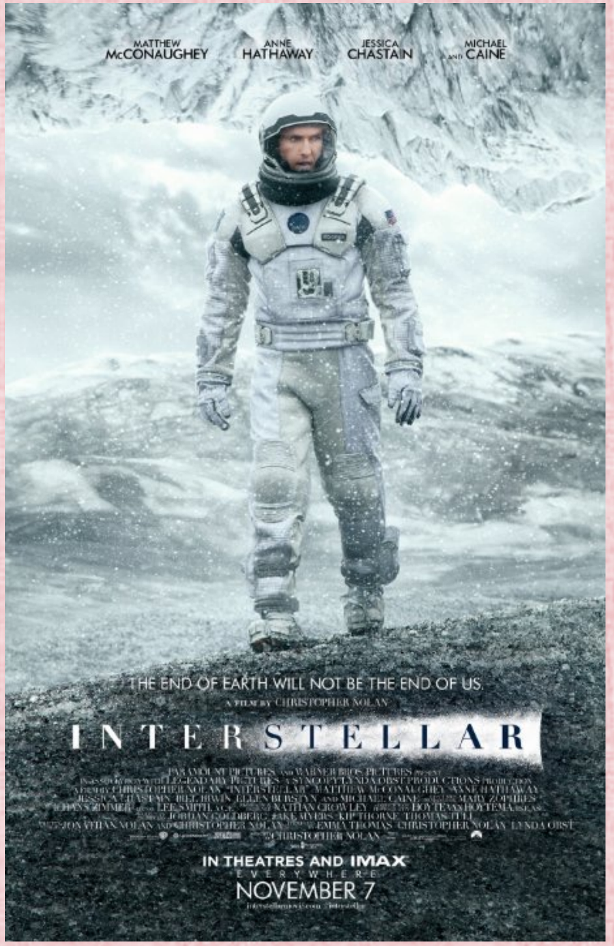
Atmospheric Refraction

- Bend ray traced rays as they go through varying air densities
- Change the direction between every interval in the vertical direction (left) or along the ray direction (right)





Gravity can bend light too!



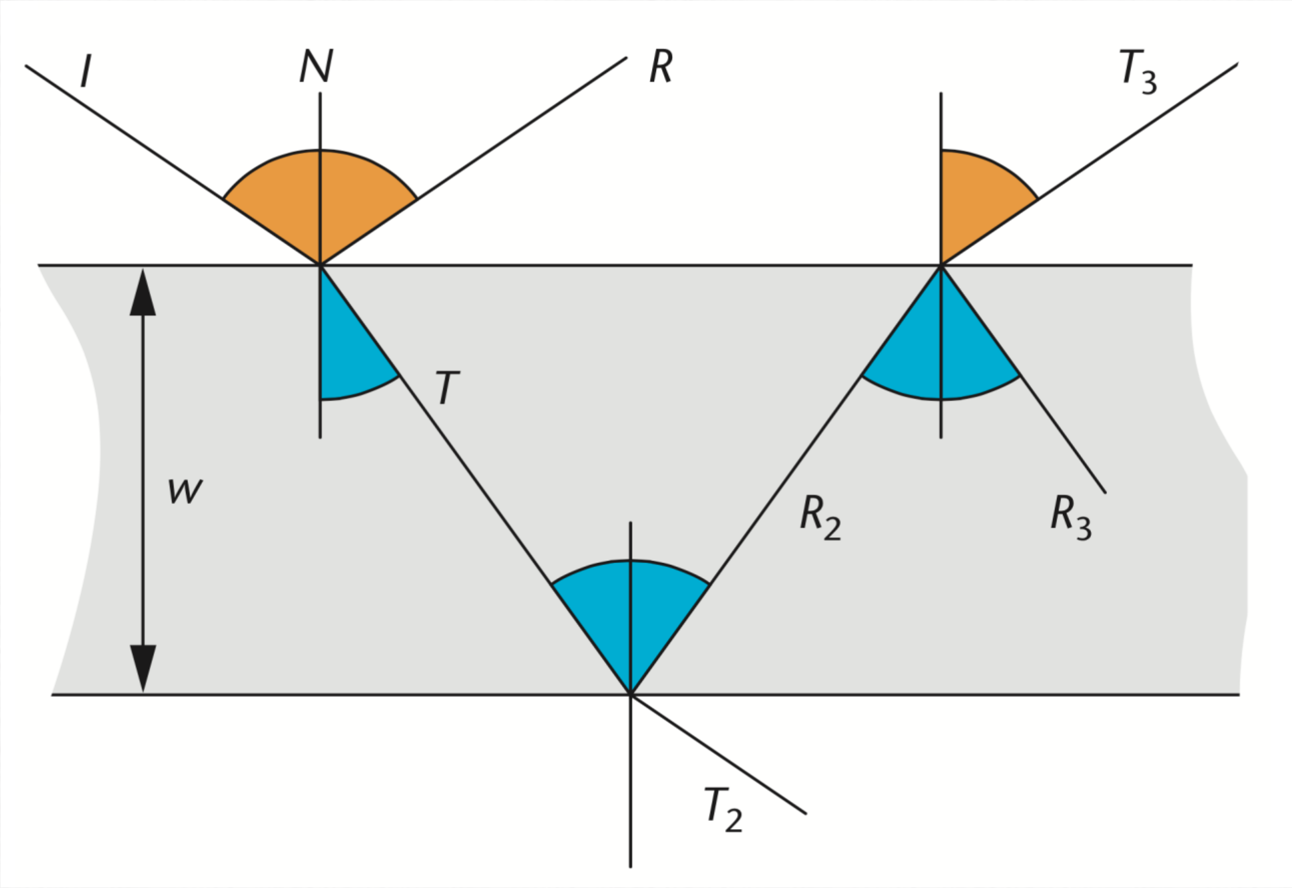
http://www.wired.com/2014/10/astrophysics-interstellar-black-hole/
https://www.businessinsider.com/interstellar-anniversary-learned-about-black-holes-2019-11

Iridescence

- A surface can gradually change color as the viewing angle or the lighting change

Iridescence

- Various light waves are emitted in the same direction giving constructive/destructive interference



http://www.glassner.com/wp-content/uploads/2014/04/CG-CGA-PDF-00-11-Soap-Bubbles-2-Nov00.pdf