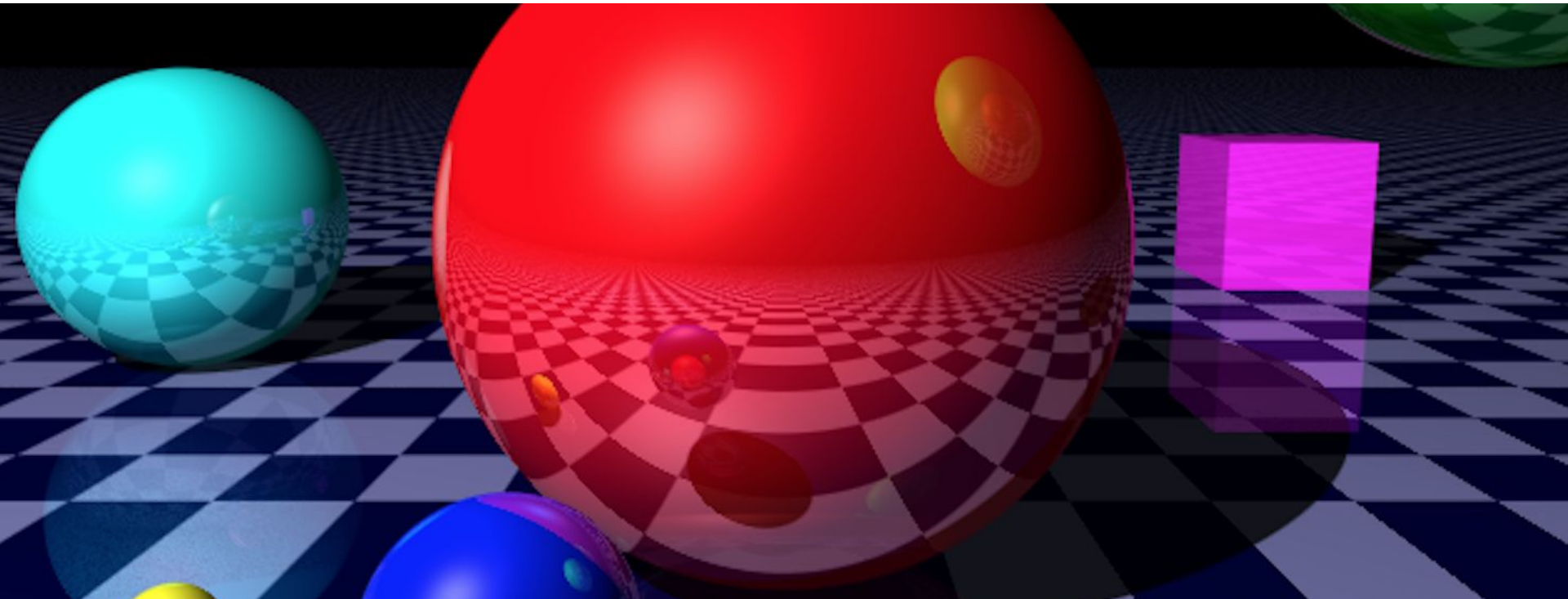
Ray Tracing:
Modeling Realistic Light in Computer Graphics

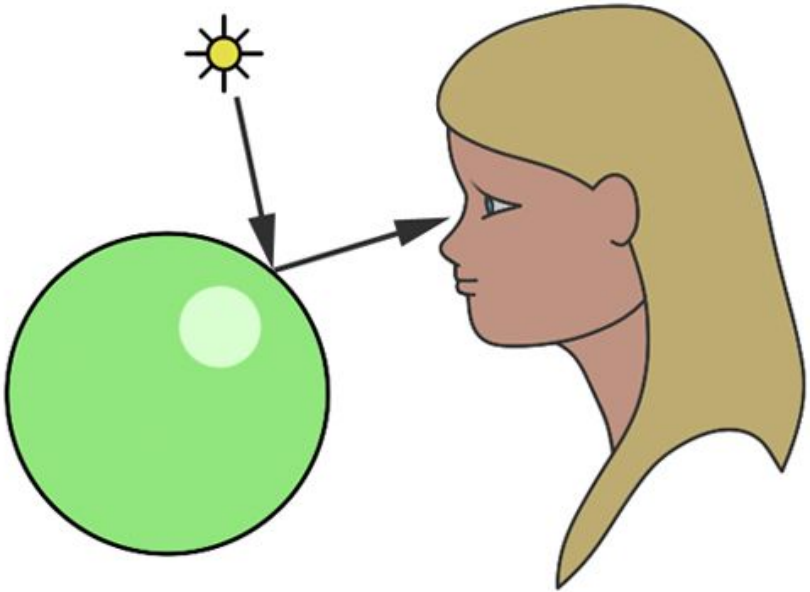
Kiet Tran

Project in

COMP 465: Interactive Computer Graphics

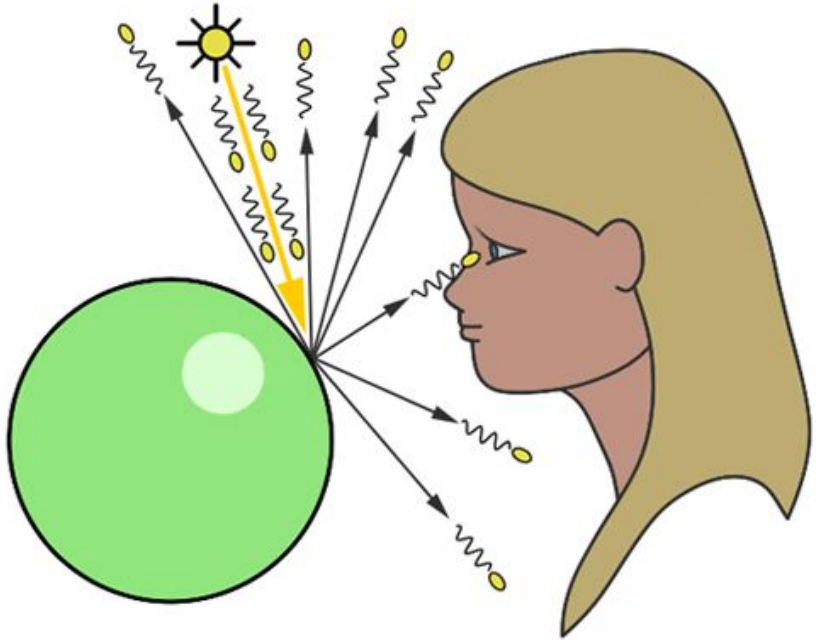


How light works in real life (basics)



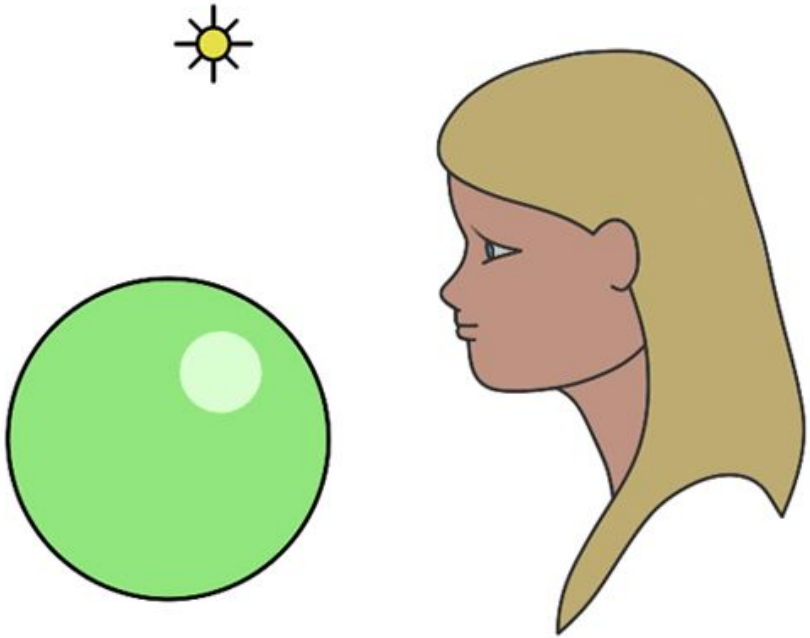
Scratchapixel.com

But few light rays reach the eye ...



Scratchapixel.com

Ray Tracing = Real life, but backwards



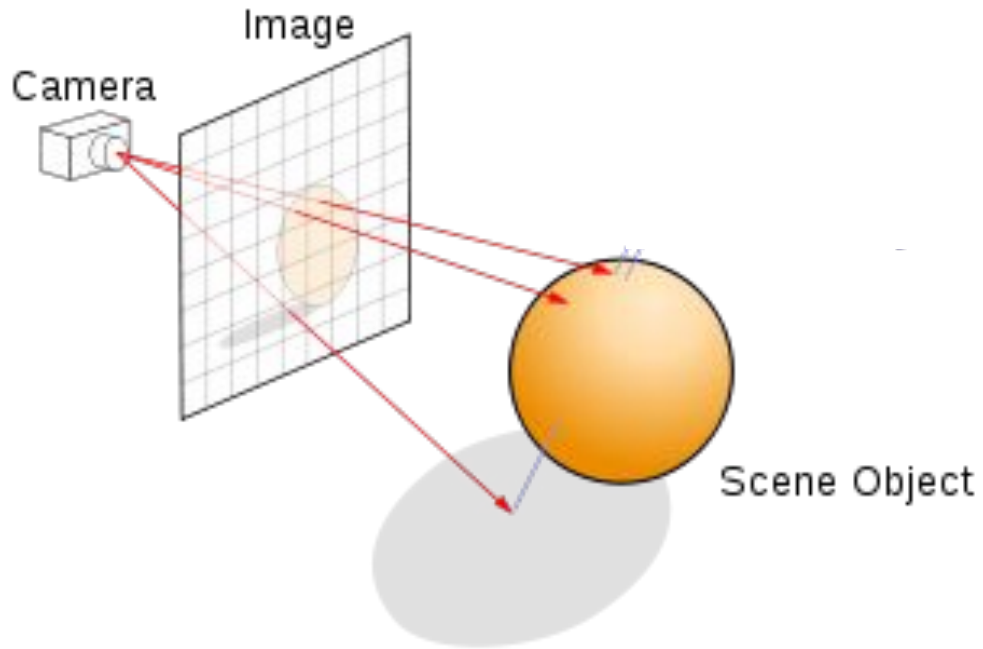
Scratchapixel.com

Computer Graphics Vocab

Camera

Image Plane

Scene Object(s)

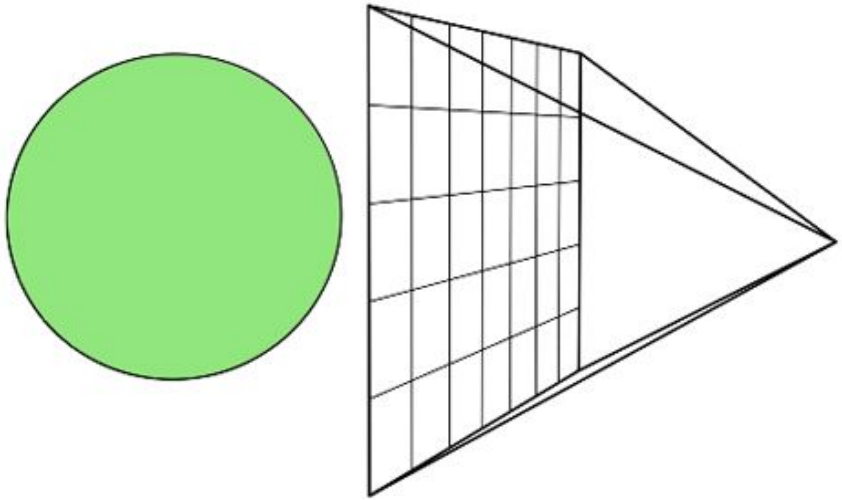


Wikipedia.com

Ray Tracing Basics

For every pixel in the image plane:

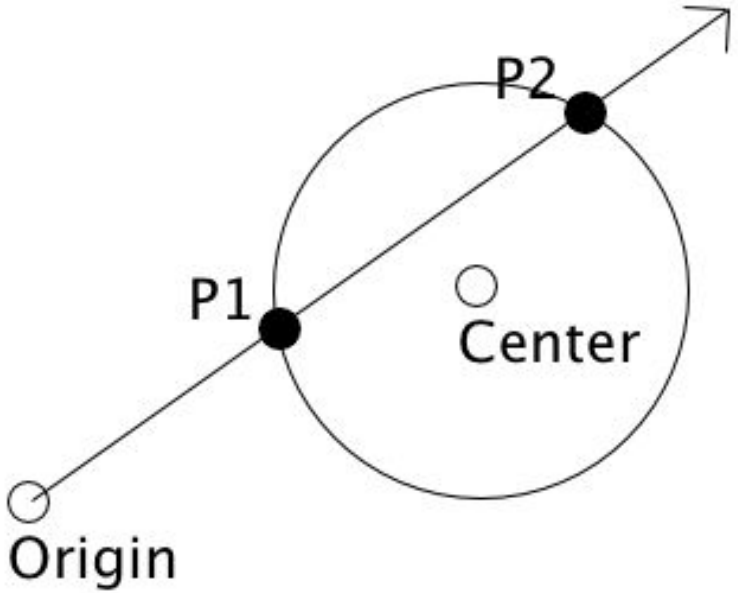
- **Emit a light ray**
- **Check if the light ray intersects with anything**



Scratchapixel.com

Math : Sphere-Ray intersection

- Point ***x*** on Sphere:
  - $||x - center||^2 = radius^2$
- Point ***x*** on Ray:
  - **x = origin + direction * t**
- Combine the 2 equations, **solve for t**



kylehalladay.com

Math : Sphere-Ray intersection

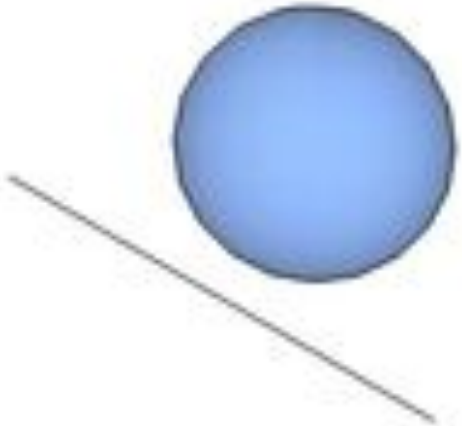
Substitute $x = o + dt$ into $||x - c||^2 = r^2$

$$
\begin{aligned}
& ||o + dt - c||^2 = r^2 \\
\Leftrightarrow\ & (o + dt - c) \cdot (o + dt - c) = r^2 \\
\Leftrightarrow\ & t^2(d \cdot d) + 2t(d \cdot (o - c)) + (o - c) \cdot (o - c) - r^2 = 0
\end{aligned}
$$

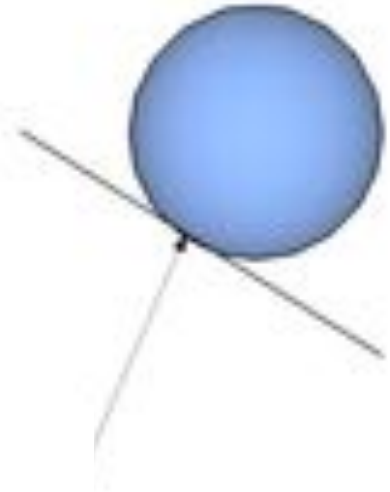
$$
t = -(d \cdot (o - c)) \pm \sqrt{(d \cdot (o - c))^2 - (||o - c||^2 - r^2))}
$$

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

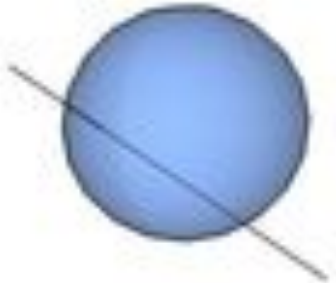
< 0



= 0

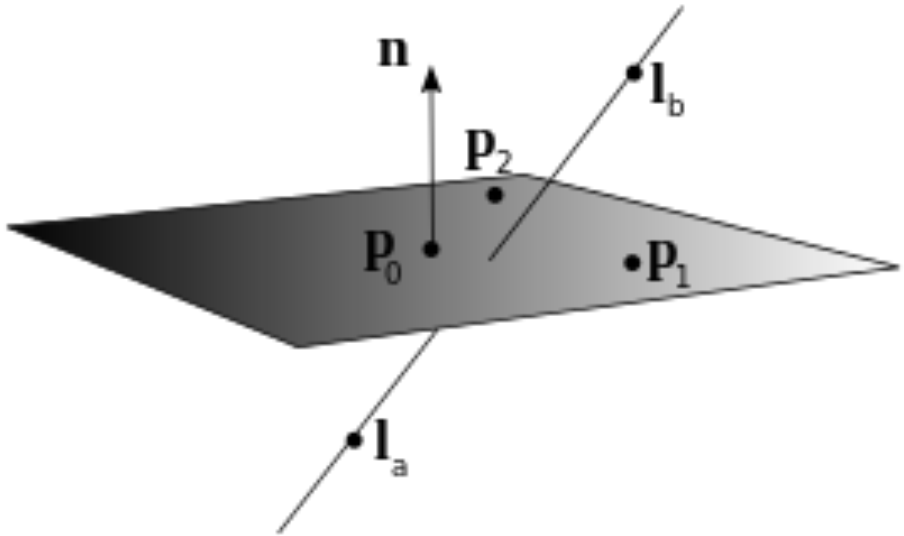


> 0



Math : Plane-Ray intersection

- Points ***x*** and x_0 on Plane:
  - $(x - x_0) \cdot n = 0$
- Point ***x*** on Ray:
  - **x = origin + direction * t**
- Combine the 2 equations, **solve for t**



Wikipedia.com

Math : Plane-Ray intersection

Substitute $x = o + dt$ into $(x - x_0) \cdot n = 0$

$$(o + dt - x_0) \cdot n = 0$$
$$\Leftrightarrow (d \cdot n)t + (o - x_0) \cdot n = 0$$

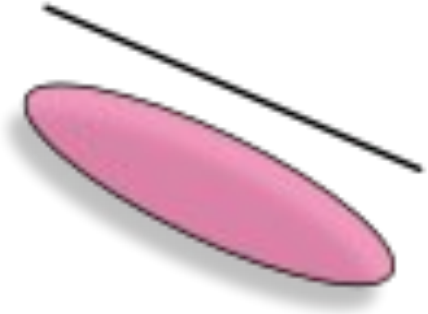
$$t = \frac{(x_0 - o) \cdot n}{d \cdot n}$$

$$t = \frac{(x_0 - o) \cdot n}{d \cdot n}$$

≠ 0

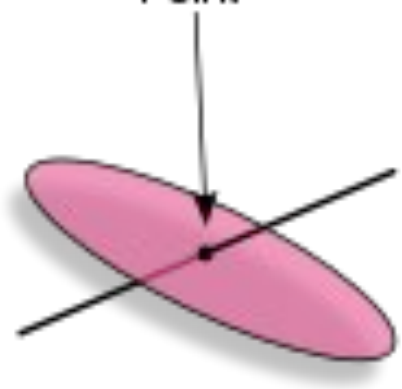
= 0

No intersection



≠ 0

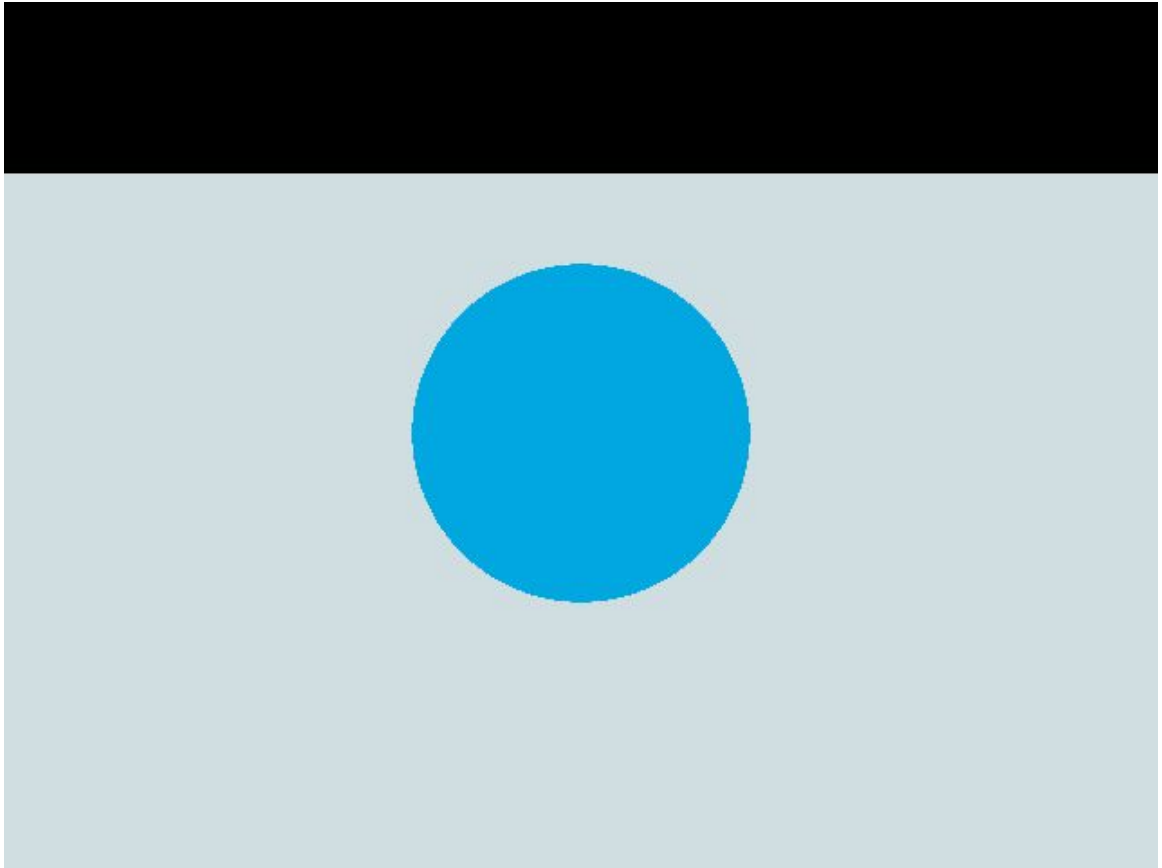
Point



= 0

= 0

Line contained in plane

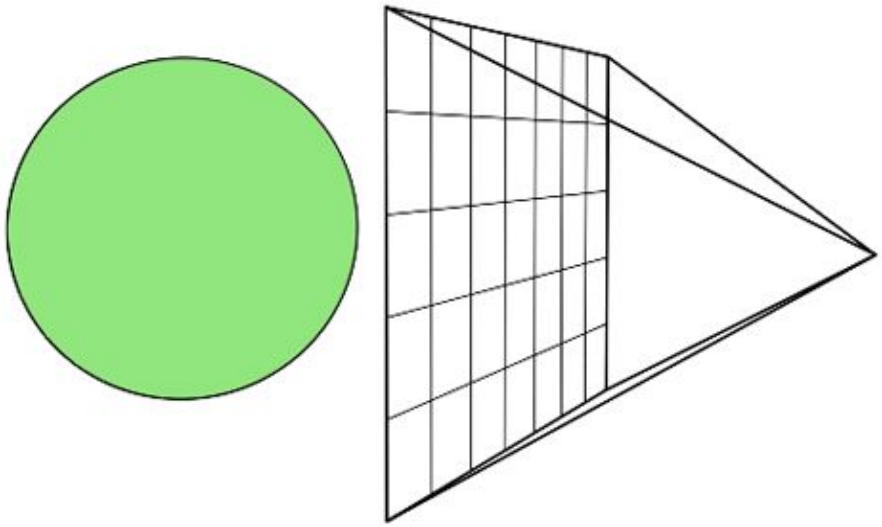


Send a light ray into the world

Ray Tracing, **Now with Shadows**

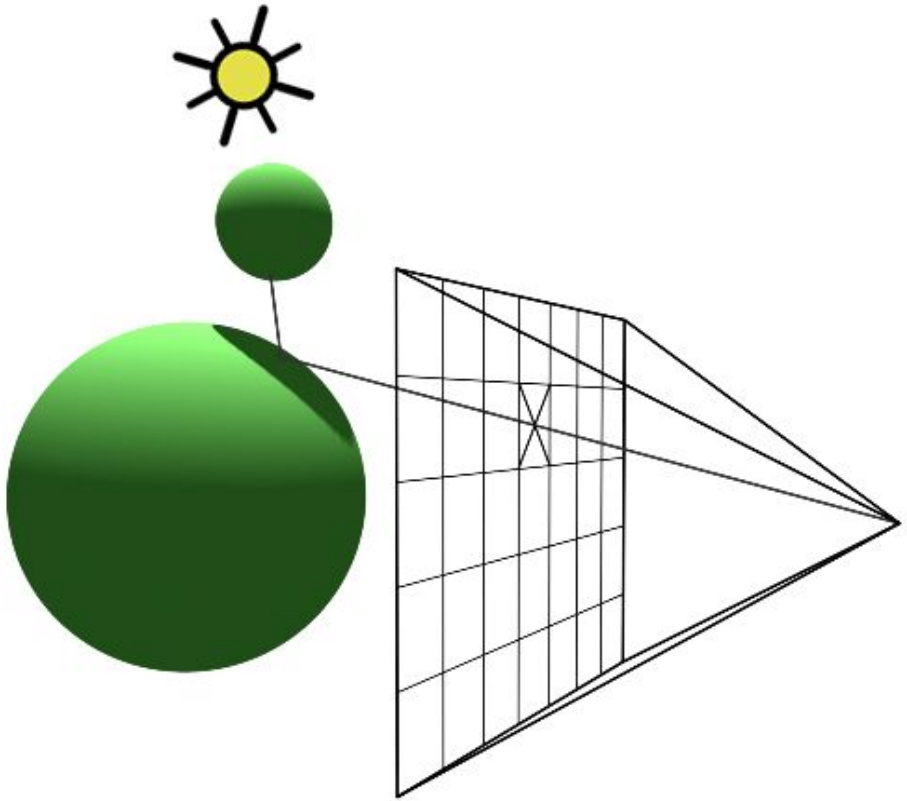
For every pixel in the image plane:

- Emit a light ray
- Check if the light ray intersects with anything
- **Apply shadows**



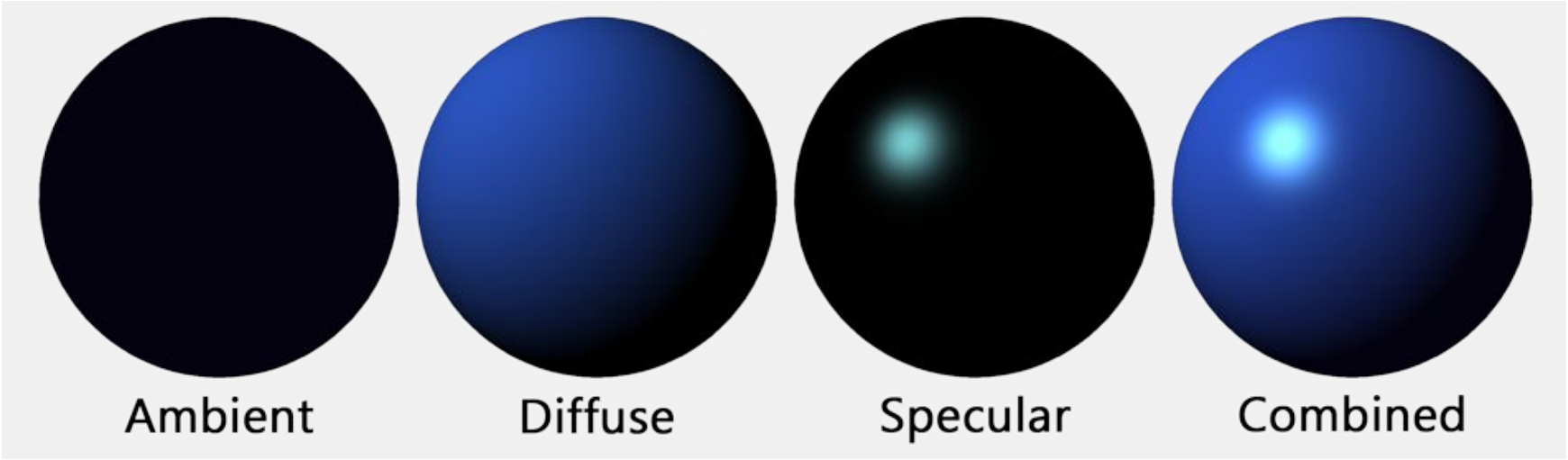
Scratchapixel.com

Shadows

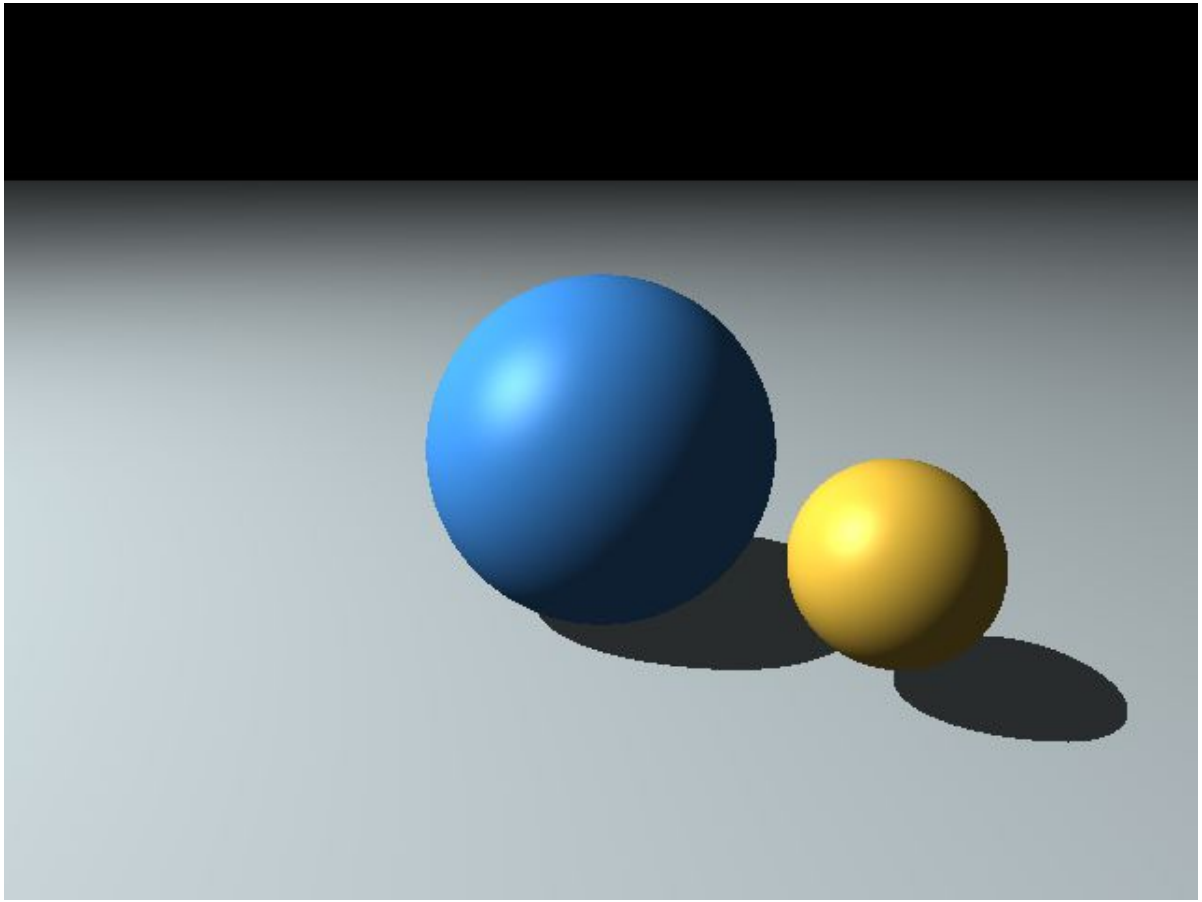


Scratchapixel.com

Shading



clara.io

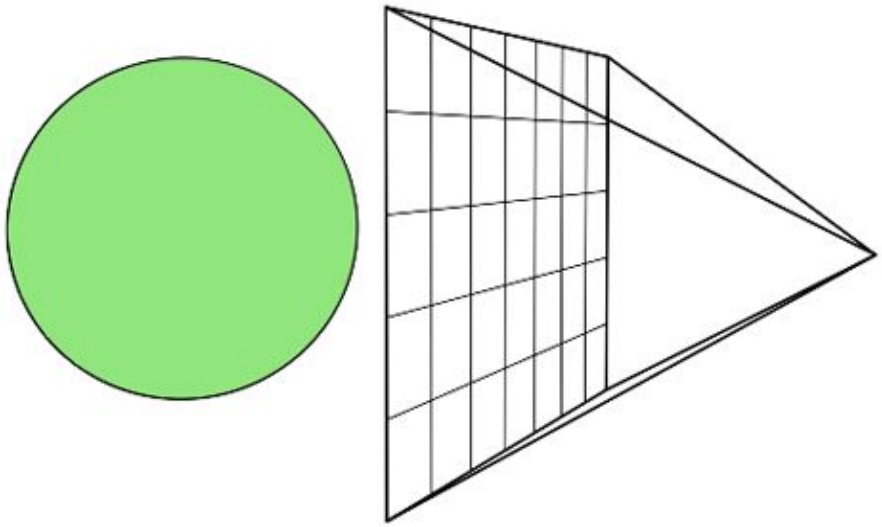


Let there be light

Ray Tracing, **Now with Reflections**

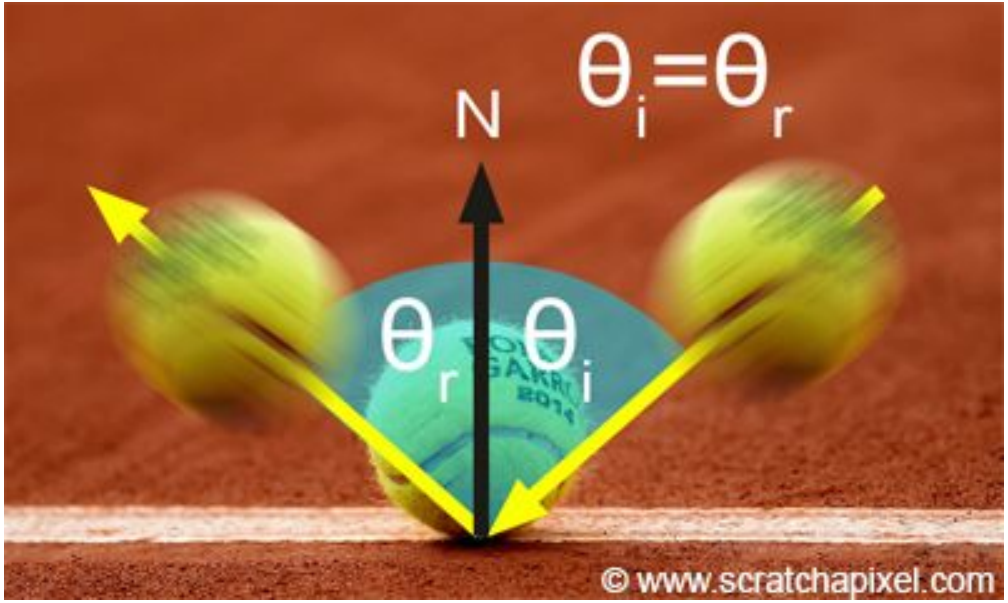
For every pixel in the image plane:

- Emit a light ray
- Check if the light ray intersects with anything
- Apply shadows
- **Keep light ray bouncing, check for reflections**



Scratchapixel.com

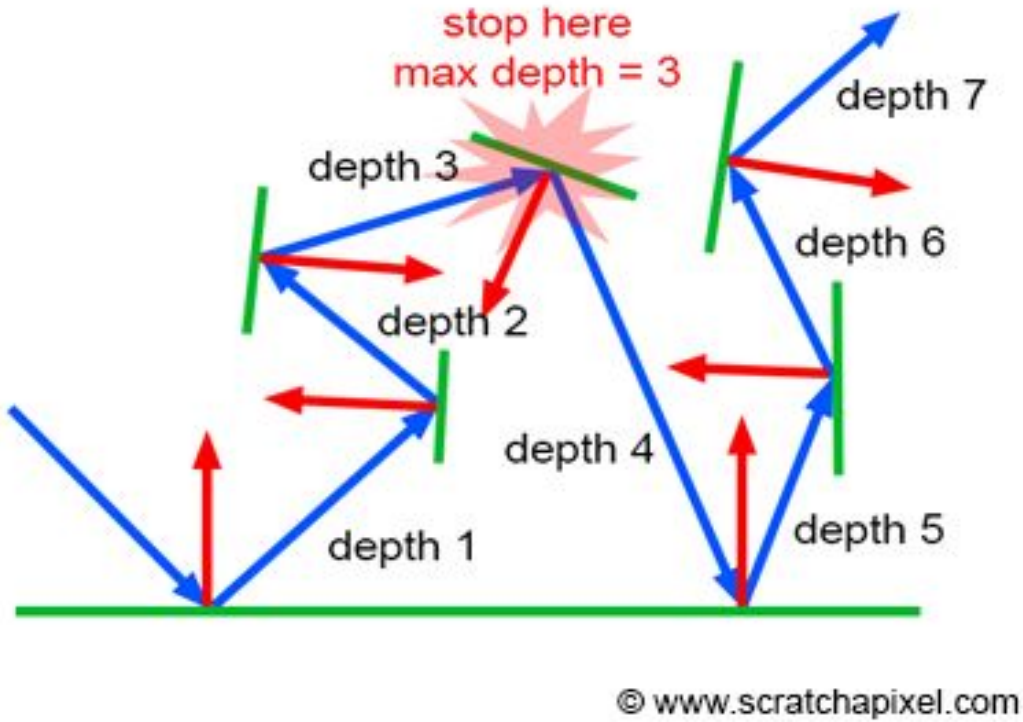
Reflections

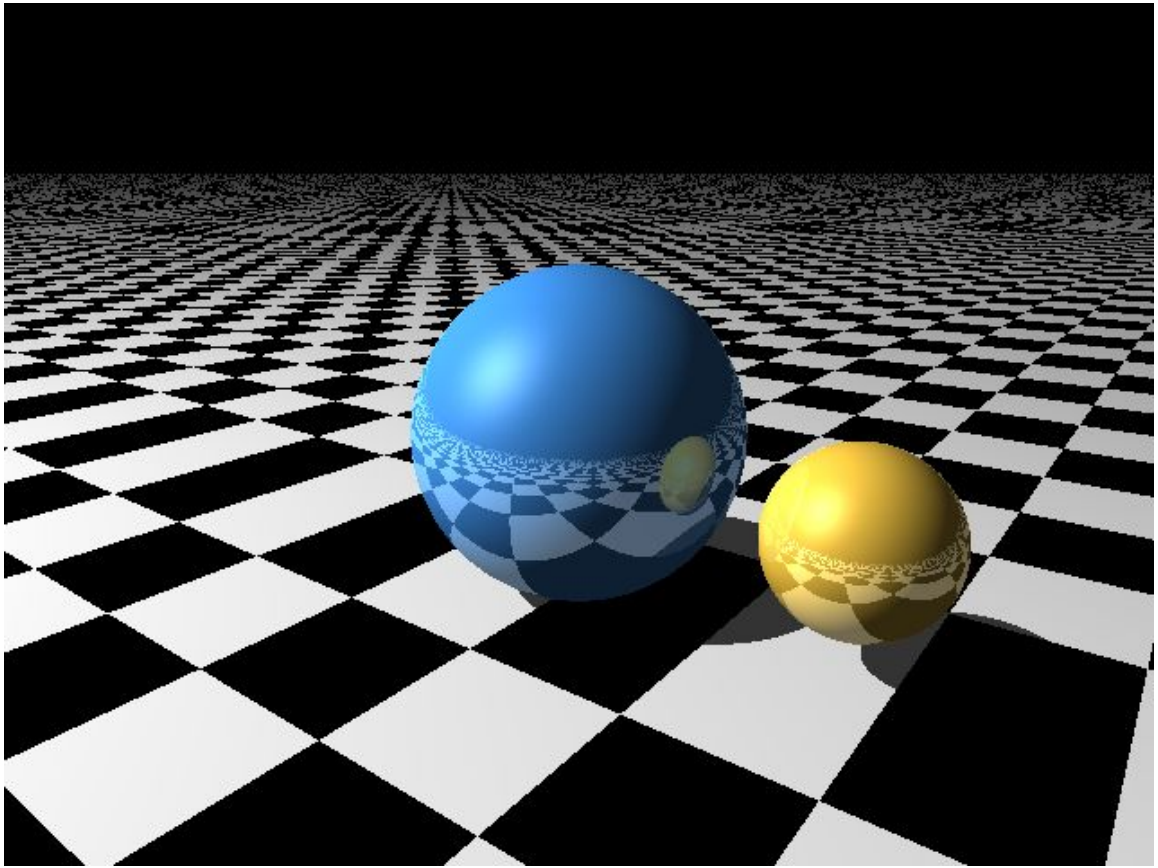


$$R = I - 2(N \cdot I)N$$

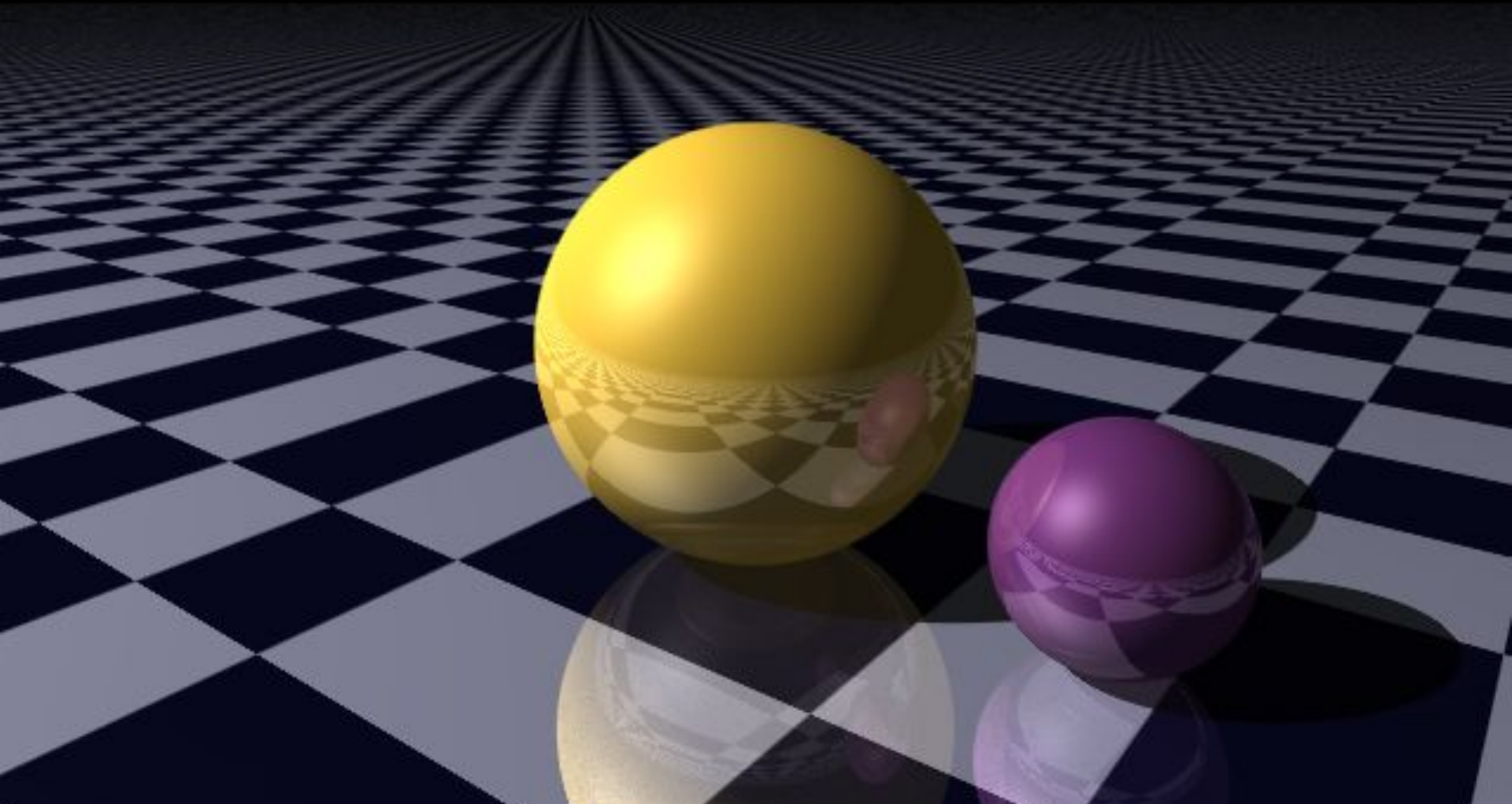
Reflections: **doesn’t happen once**

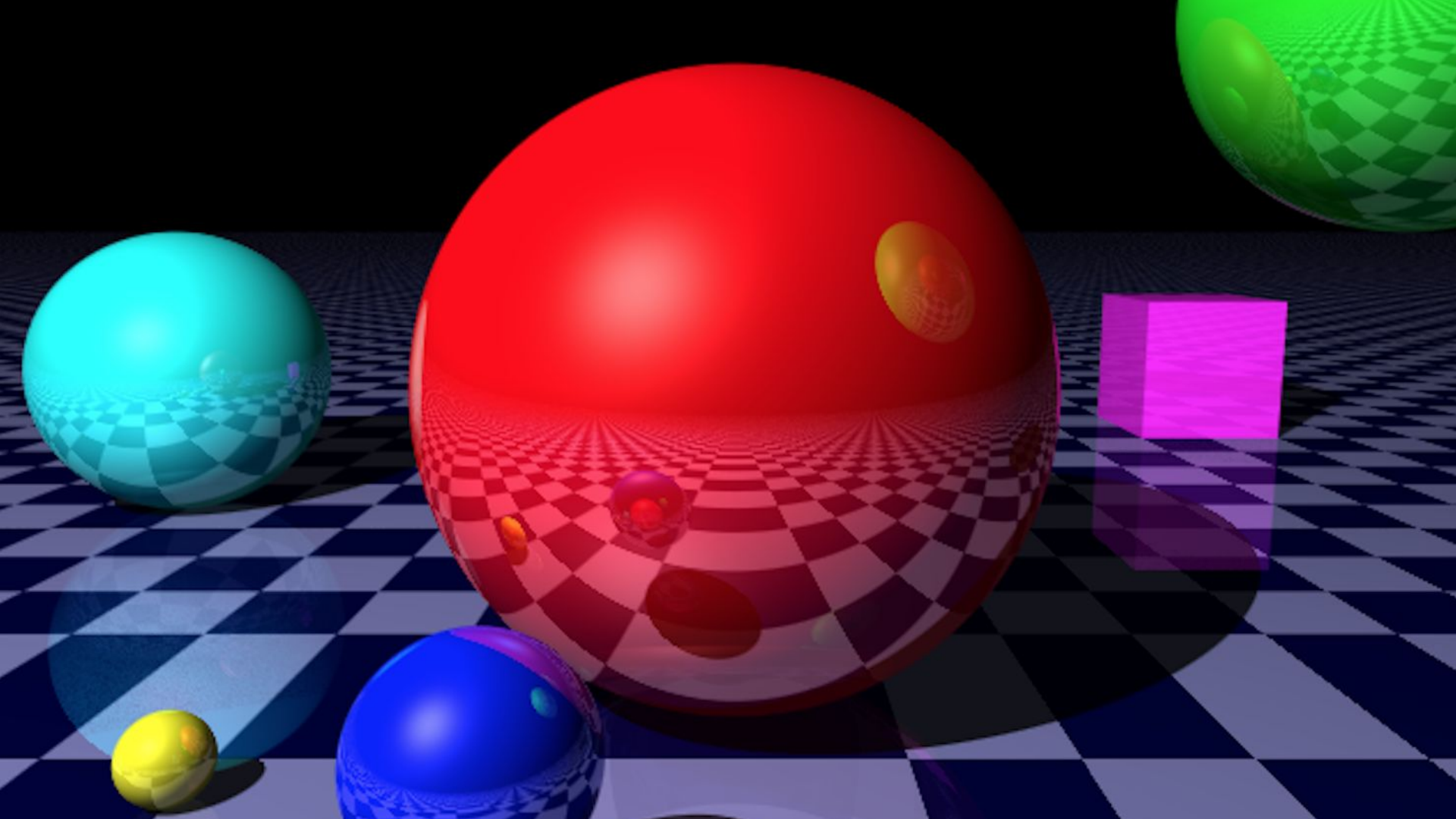
Reflections: **keep bouncing (& trace the light rays)**





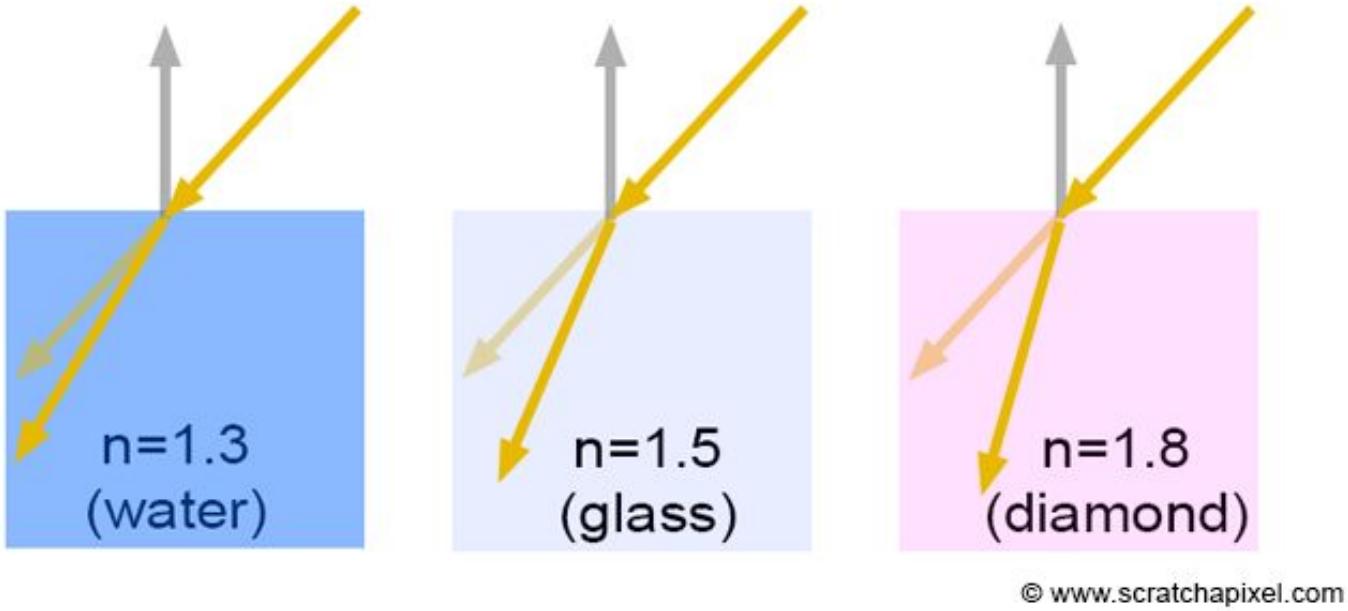
Reflections!



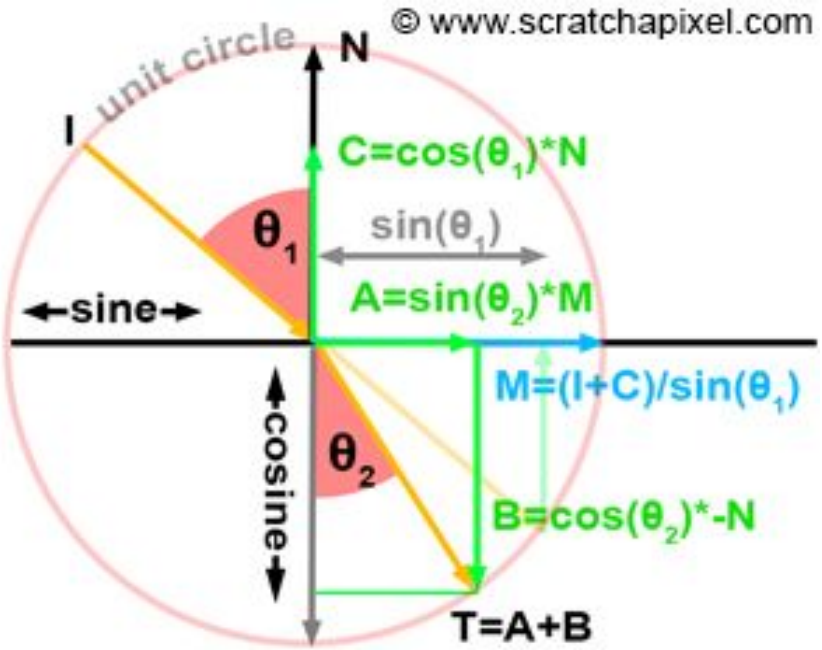


How about Refractions?

Refractions, in different environments



Refractions



$$T = \frac{\eta_1}{\eta_2}(I + \cos(\theta_1)N) - N\sqrt{1 - \left(\frac{\eta_1}{\eta_2}\right)^2 \sin^2(\theta_1)}.$$

Refractions + Reflections: **Fresnel**

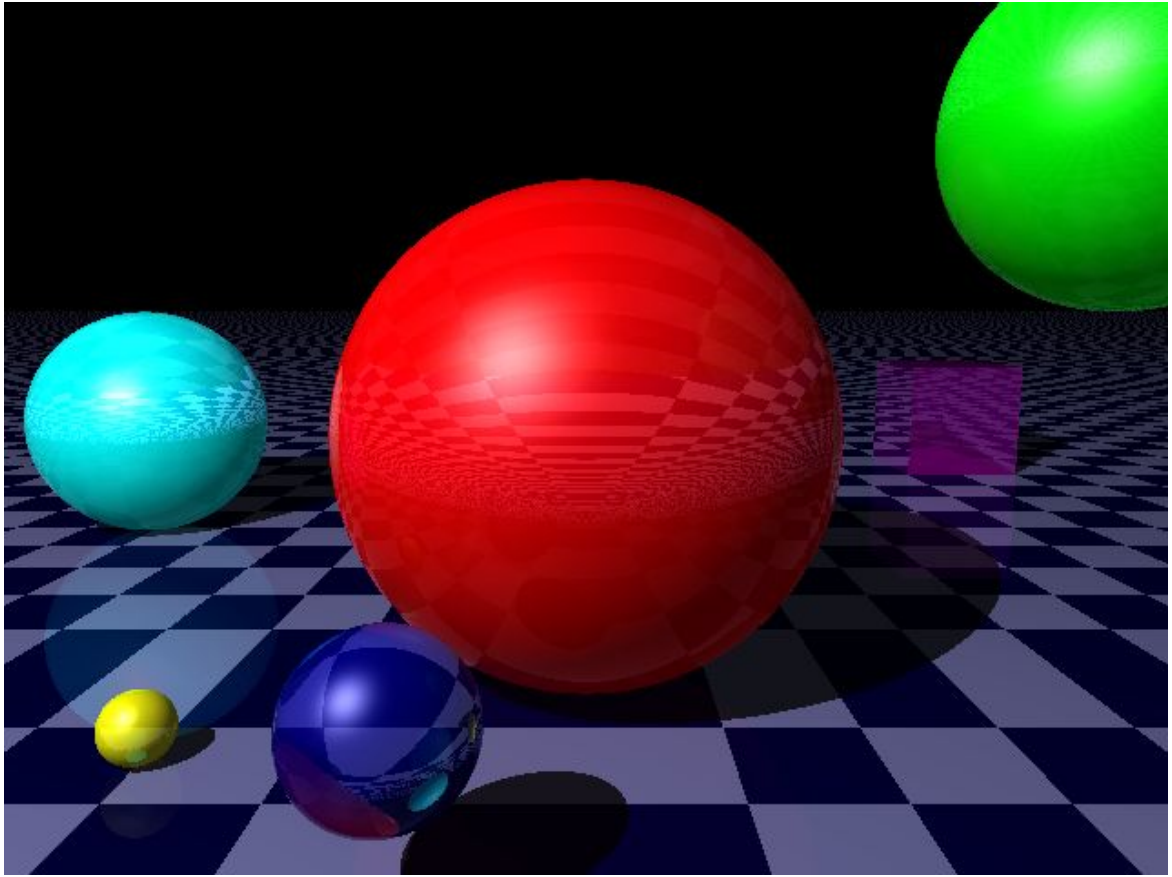
$$F_{R\parallel} = \left(\frac{\eta_2 \cos\theta_1 - \eta_1 \cos\theta_2}{\eta_2 \cos\theta_1 + \eta_1 \cos\theta_2} \right)^2,$$
$$F_{R\perp} = \left(\frac{\eta_1 \cos\theta_2 - \eta_2 \cos\theta_1}{\eta_1 \cos\theta_2 + \eta_2 \cos\theta_1} \right)^2.$$

Reflection amount

$$F_R = \frac{1}{2}(F_{R\parallel} + F_{R\perp}).$$

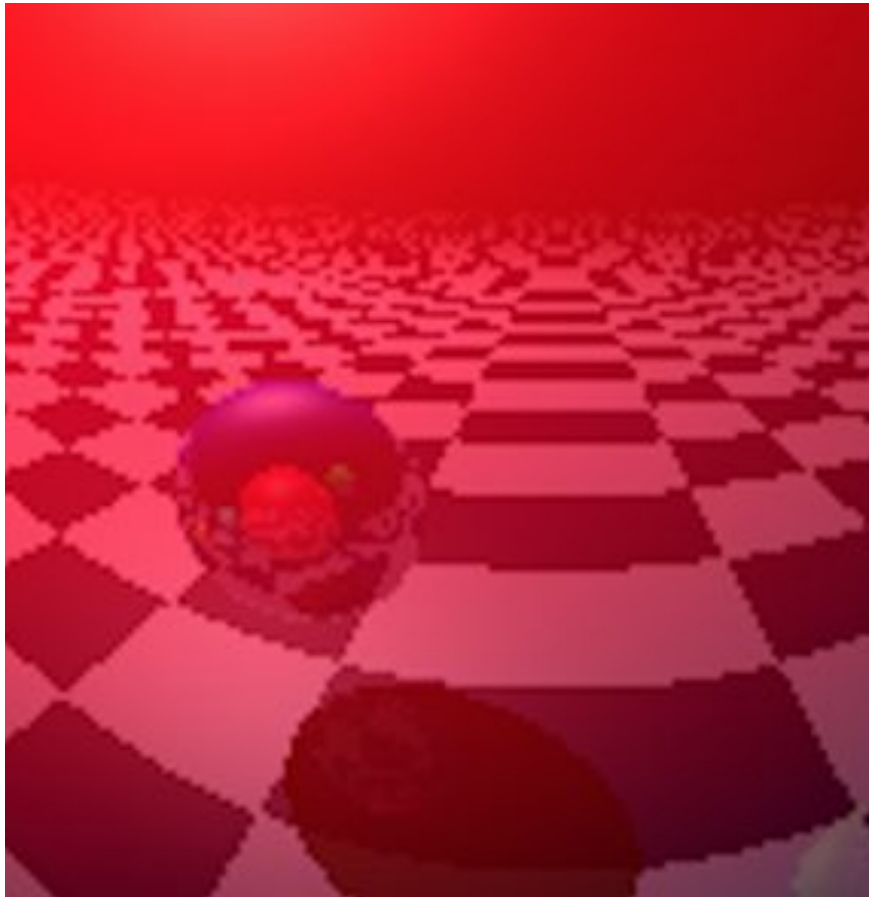
Refraction amount

$$F_T = 1 - F_R.$$

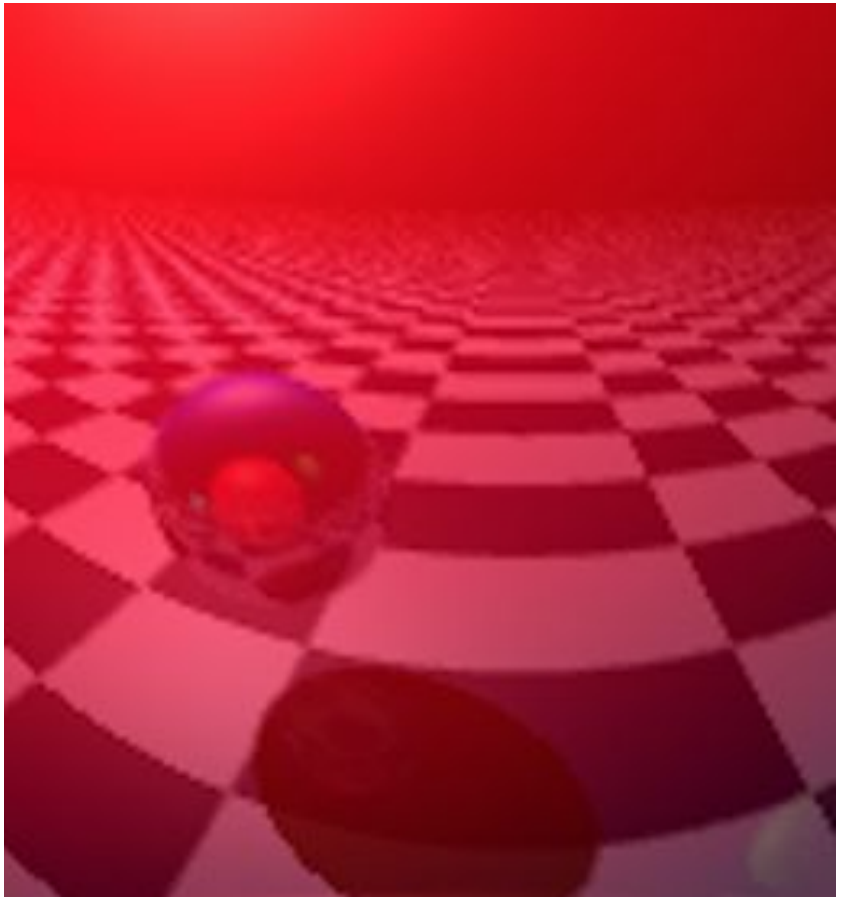


Refractions???

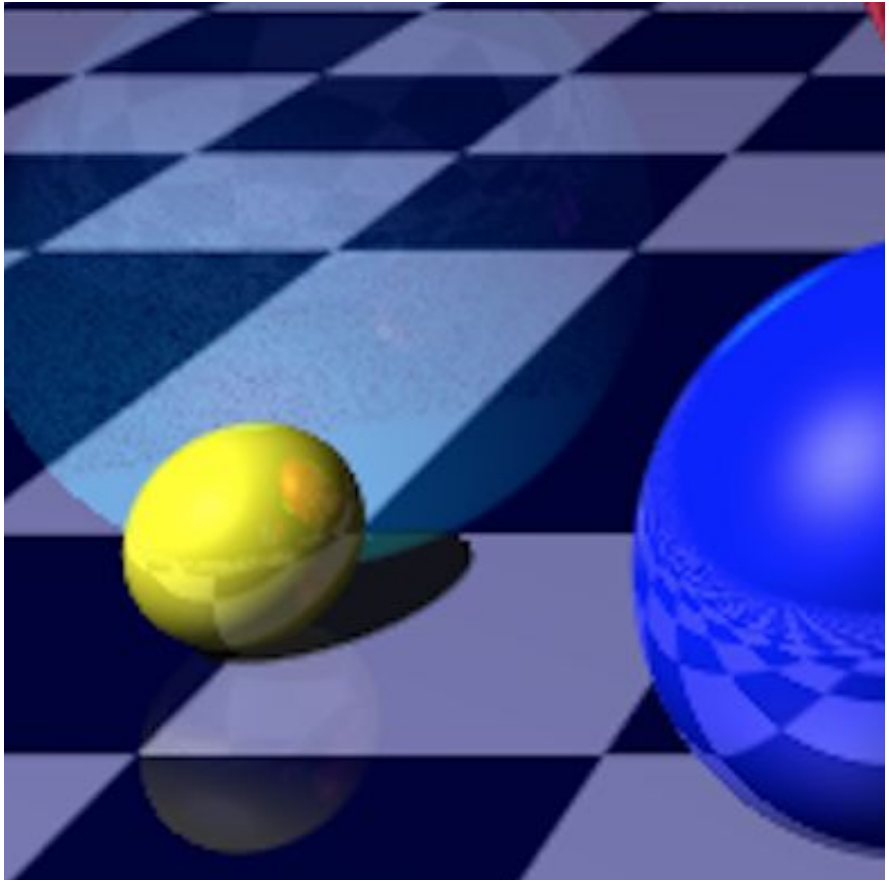

Anti-Aliasing



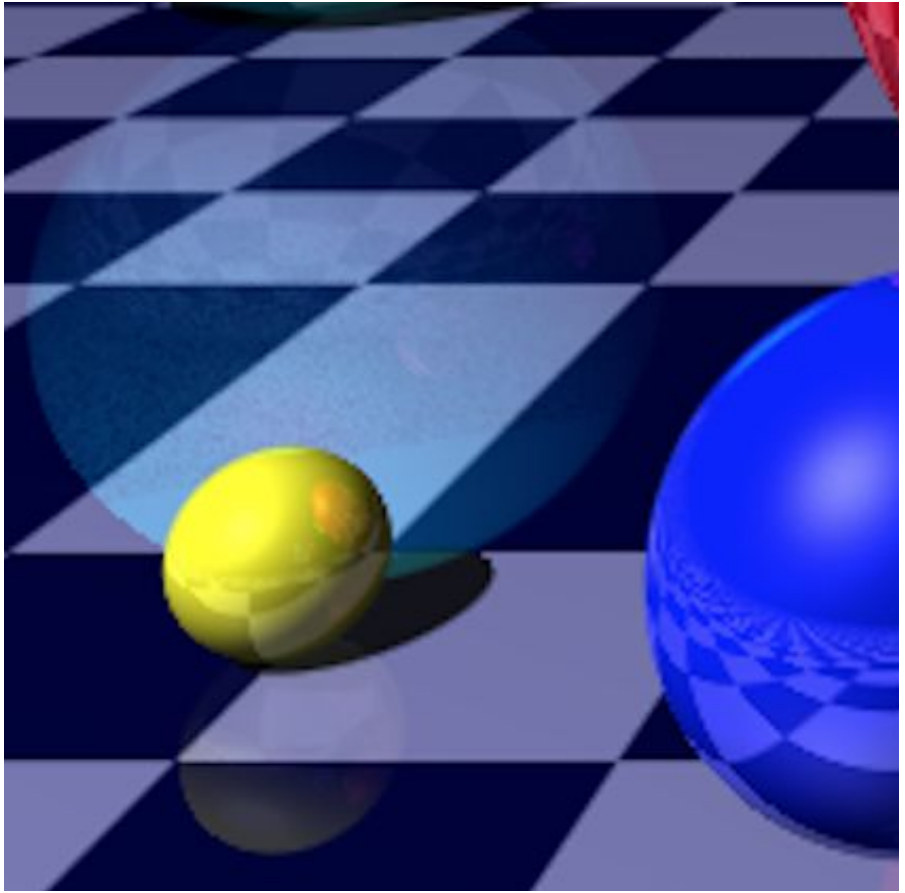
No Anti-Aliasing



Some Anti-Aliasing

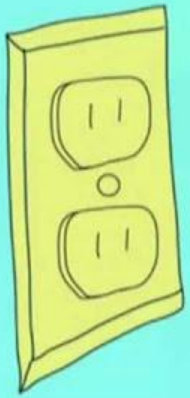


Anti-Aliasing Depth = 10



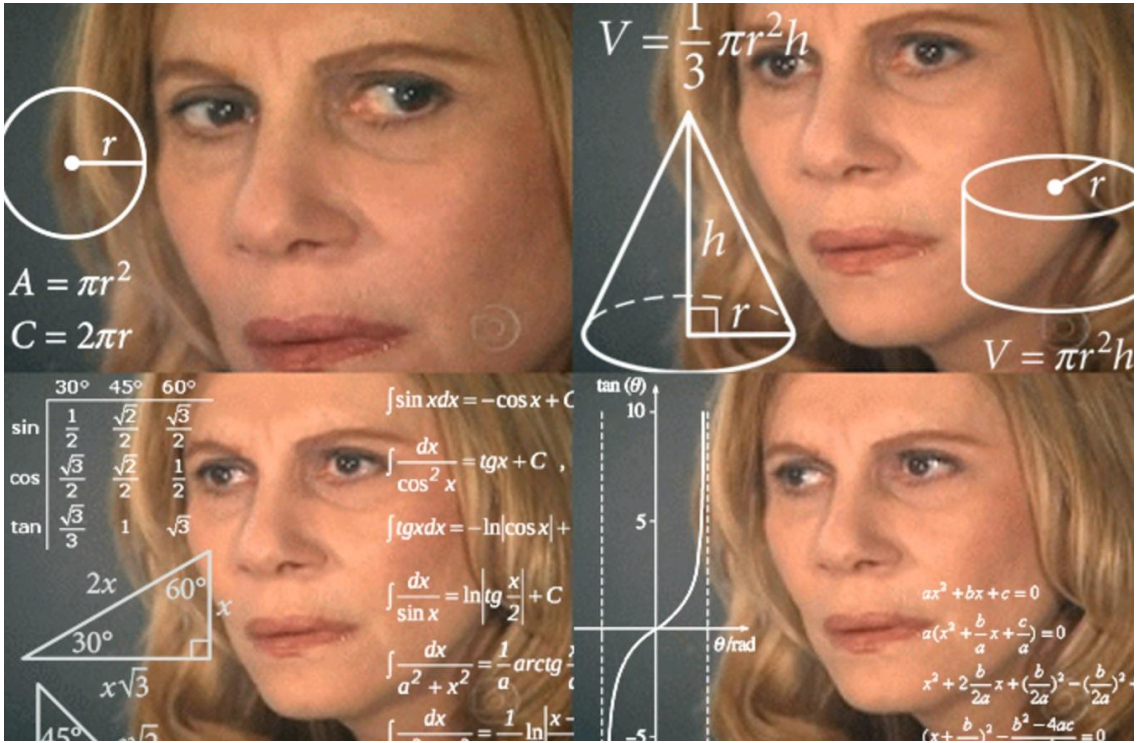
Anti-Aliasing Depth = 20

TECHNICAL DIFFICULTIES

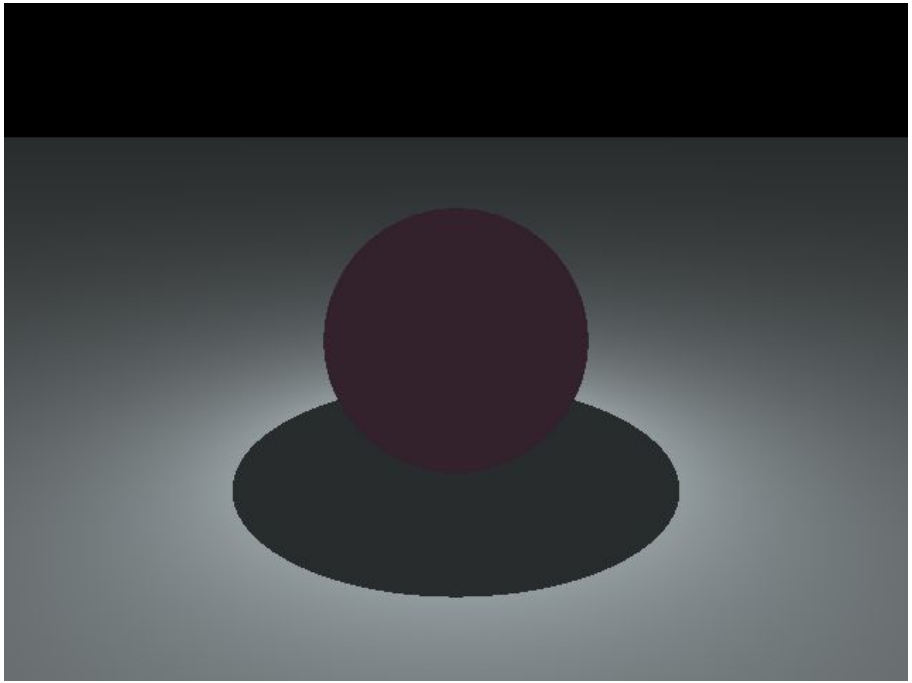


PLEASE STAND BY

01 | Math!

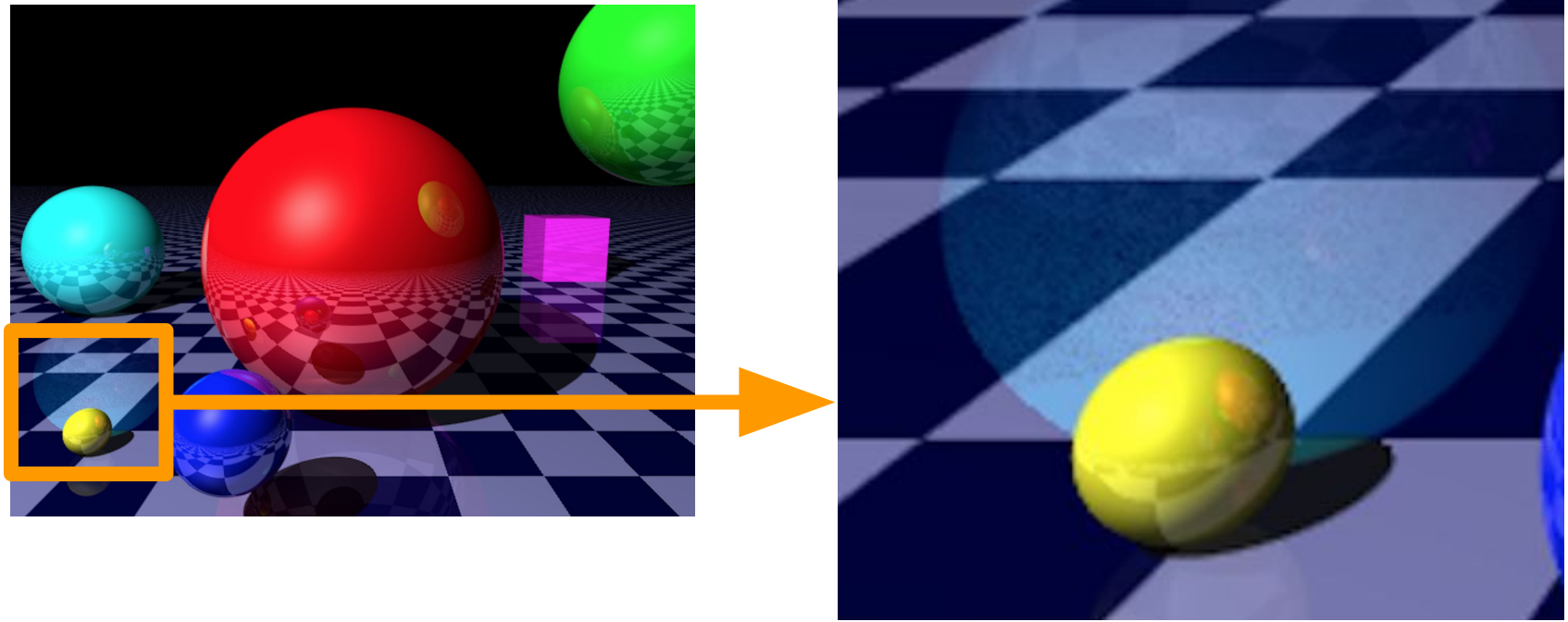

r
$A = \pi r^2$
$C = 2\pi r$
$V = \frac{1}{3}\pi r^2 h$
h
r
r
$V = \pi r^2 h$
30° 45° 60°
sin
cos
tan
$\int \sin x dx = -\cos x + C$
$\int \frac{dx}{\cos^2 x} = tgx + C$,
$\int tgx dx = -\ln|\cos x| +$
$\int \frac{dx}{\sin x} = \ln\left|tg\frac{x}{2}\right| + C$
$\int \frac{dx}{a^2 + x^2} = \frac{1}{a} arctg$
2x
60°
x
30°
$x\sqrt{3}$
tan (θ)
10
5
θ/rad
$ax^2 + bx + c = 0$
$a(x^2 + \frac{b}{a}x + \frac{c}{a}) = 0$
$x^2 + 2\frac{b}{2a}x + (\frac{b}{2a})^2 - (\frac{b}{2a})^2$
$(x + \frac{b}{}) - \frac{b^2 - 4ac}{} = 0$

02 | Shadow Implementation

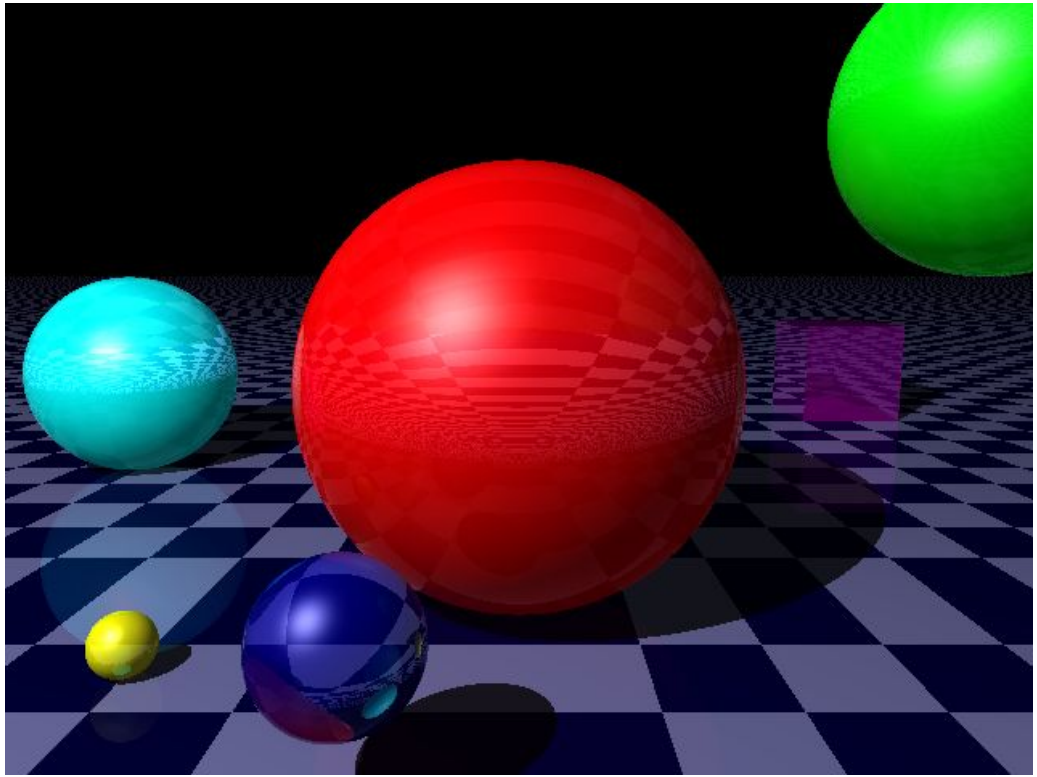


Buggy shadow

03 | Shadow “Acnes”



04 | Refraction - fail :(



Future Work

01 | Removing Shadow Acnes

02 | Implement transparency

03 | Add an Interactive component