

# CS230 : Computer Graphics

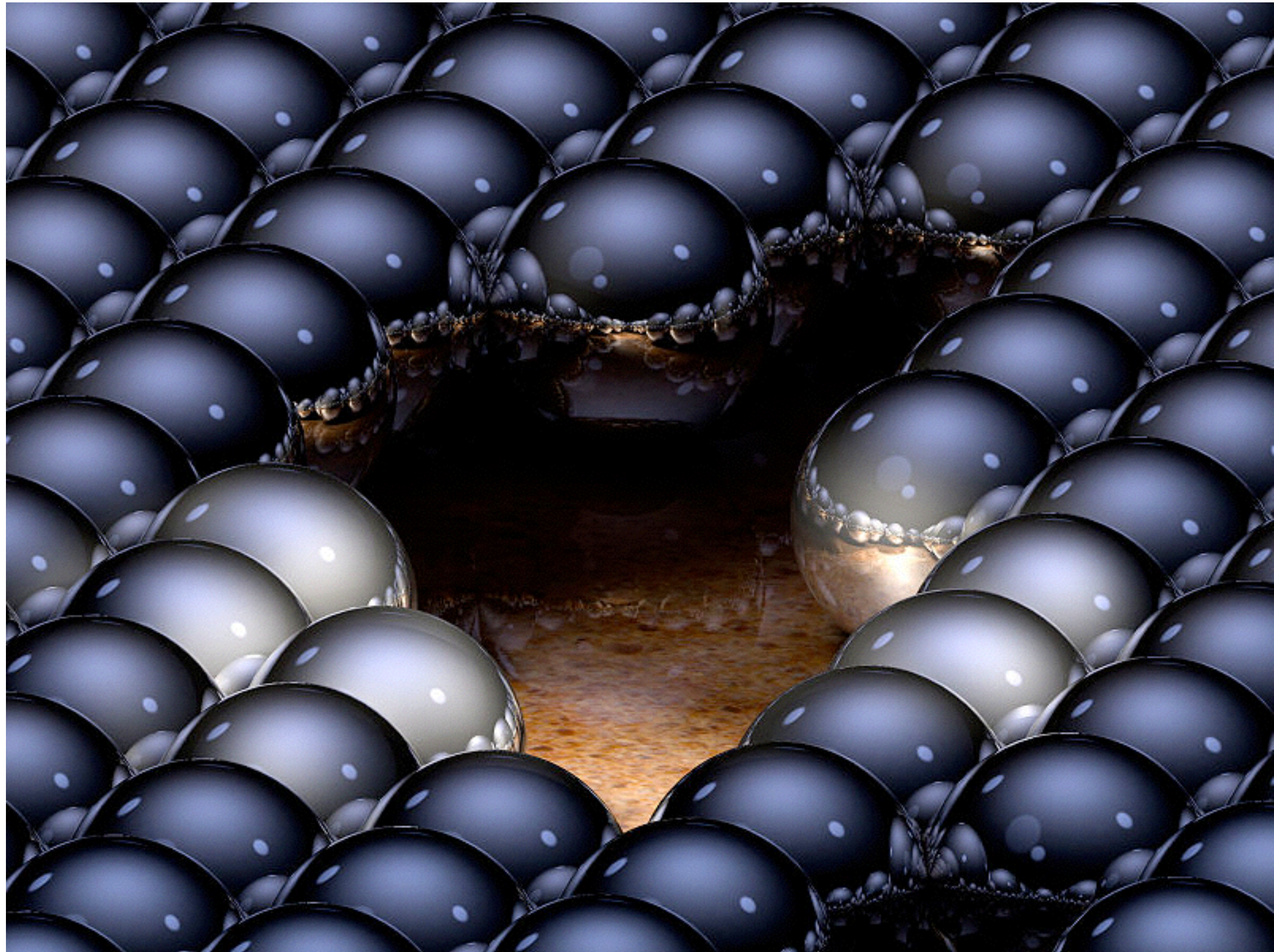
Tamar Shinar  
Computer Science & Engineering  
UC Riverside

# Ray Tracing



Wikimedia Commons

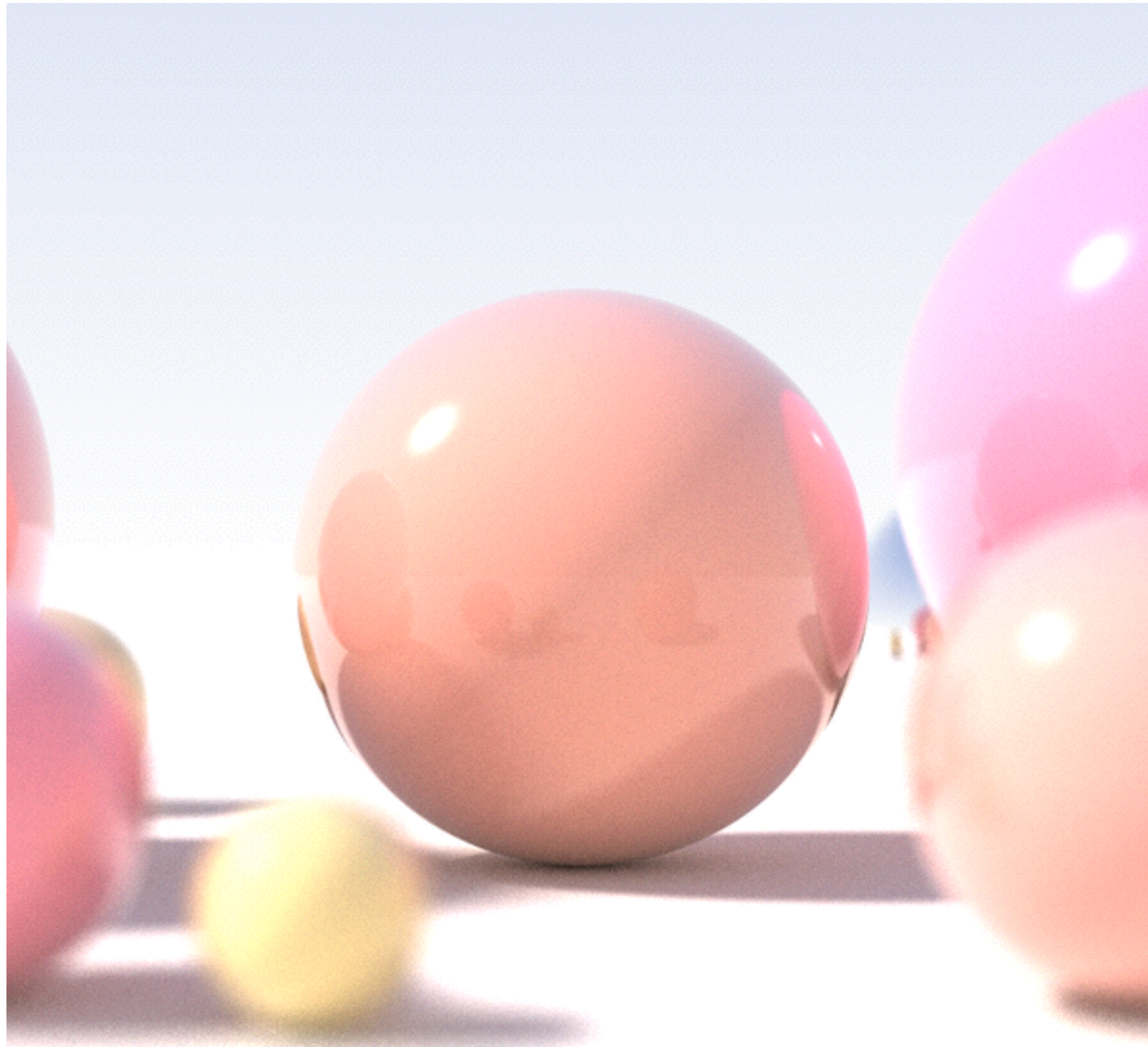




up to 16 reflections per ray

Greg L., Wikimedia Commons





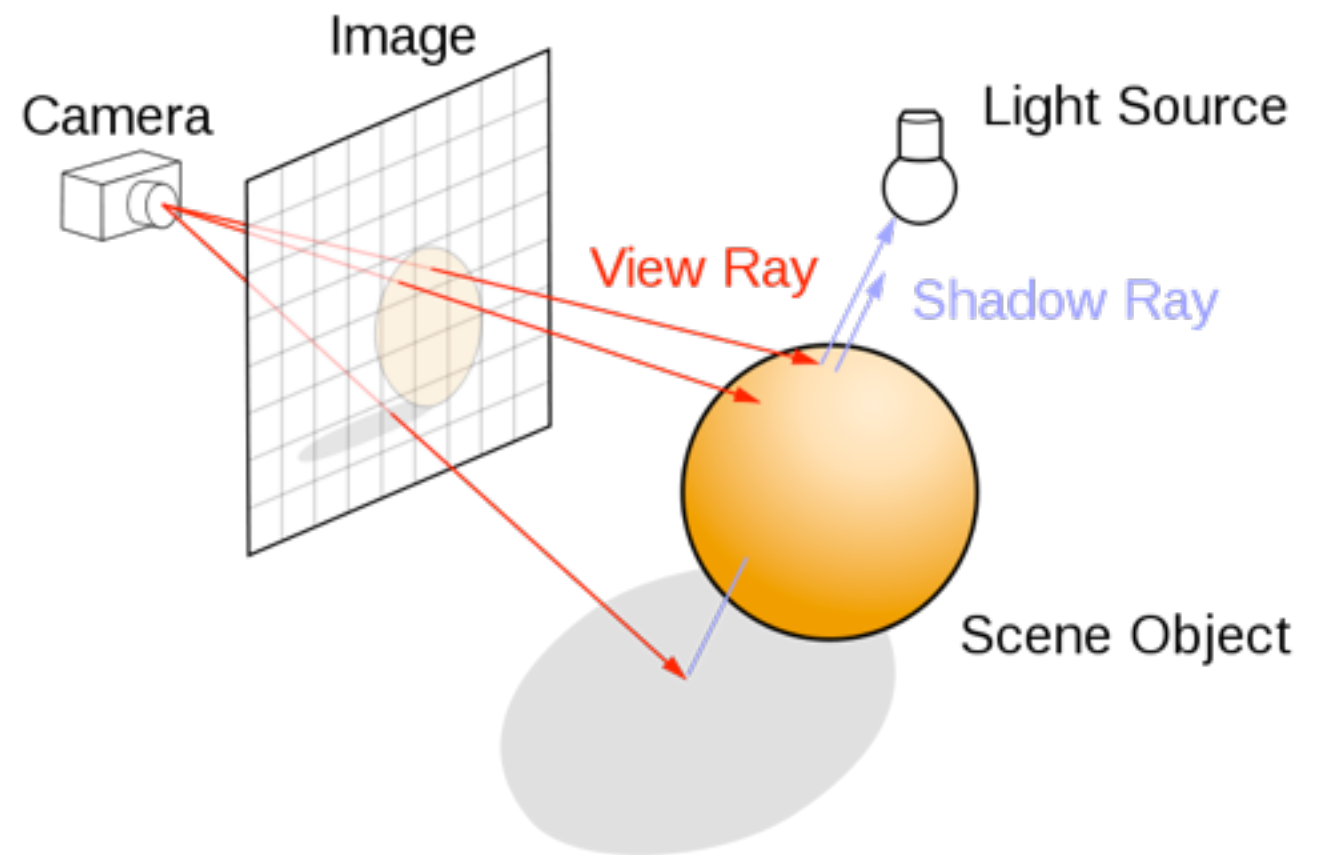
Wikimedia Commons

shallow depth of field, area light sources, diffuse interreflection

# Basic Algorithm

for each pixel

1. **cast view ray:**  
compute view ray  
from camera through  
pixel into scene
2. **intersect:** find  
intersection of ray  
with closest object
3. **shade:** compute the  
color of the  
intersection point

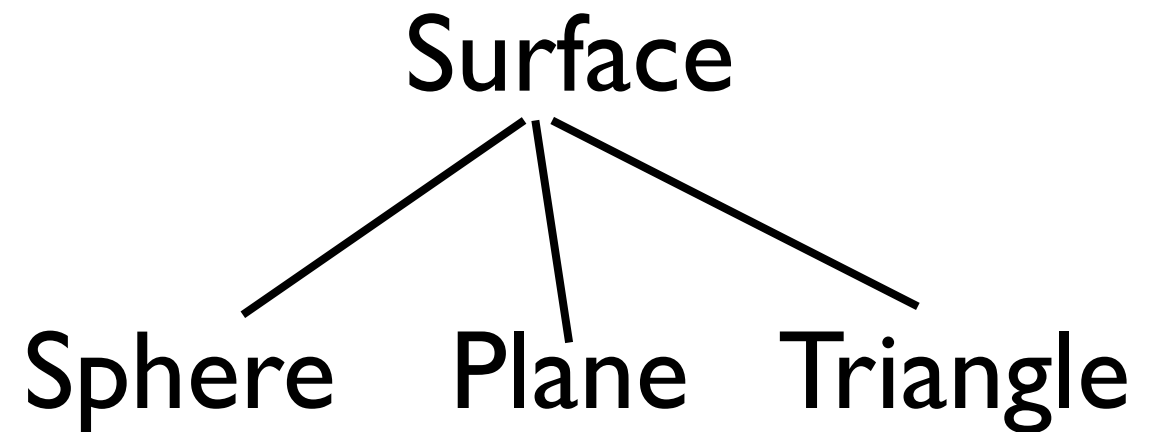


# Ray Tracing Program

```
for each pixel do  
  compute viewing ray  
  if ( ray hits an object with t in [0, inf] ) then  
    compute n  
    evaluate shading model and set pixel to that color  
  else  
    set pixel color to the background color
```

# Object-oriented design

```
class Surface
{
    public:
        bool Intersection(RAY& ray)=0;
        Box Bounding_Box()=0;
}
```



**Other objects: Ray, Light,  
Material, Camera, Film, World**



# Simple Ray Tracer

