CmpE 460 : Computer Graphics

Assignment 2 : Recursive Ray Tracing & Diffuse Shading

Emre Girgin & 2016400099

Method

In recursive ray tracing, we shoot a ray to the center of each pixel and if this ray intersects with an object in world space, it is reflected from that intersection point as well. This process is repeated until a max depth (max reflection count) is reached. The algorithm may be written like this:

```
for each pixel p do
    r = ray from the eye through p;
    color of p = Trace(r, 0);
endfor

Trace(r, d)
      if d > d_max then return background color; endif
      q = Intersect(r);                          // q: object surface point
      if q = null then
        return background color;
      endif
      c = AccLightSource(q);                                  // c: color
      if object (q) is reflective (coherently) then
        r_r = ray towards inverse direction of reflection;
        c += Trace(r_r, d + 1);
      endif
end
```

r_r, the reflected ray, is obtained by the formula $r_r = D - 2(D \cdot N))N$ where D is the direction vector of the coming ray (coming ray is r in our case) and N is the surface normal at the intersection point.

```
AccLightSource(q)
    c = ambient intensity + own emission;          // c: color
    for each lightsource l do
        r = ray from q towards l;
        if Intersect(r) = null then
            c += diffuse intensity;
            c += specular intensity;
        endif
    endfor
    return c;
end
```

We only used **ambient** and **diffuse shading** methods for the sake of the simplicity and not specular intensity.

The algorithms are taken from http://www.fsz.bme.hu/~szirmay/ray.pdf

The diffuse shading is calculated as the following: $max(0, N \cdot r_s) * c_l$ where N is the surface normal at the intersection point and the r_s is the secondary ray that has the direction to the light source and has the origin at the intersection point and c_l is the diffuse component of the object.

Input & Output

a. Config File

I've used a config file to get spheres. This is a text file. It's lines represent the spheres. Each line consists of 7 integers and 2 floats separated with white space. The first three represent the coordinates of the center of the sphere, x, y, z, respectively. The second three represents the RGB value of the sphere ranging from 0 to 255, red, green, blue, respectively. The seventh entry (integer) represents the radius of the sphere. The last two floats represent the ambient and diffuse coefficients of the sphere, respectively. (0.1 and 1.0 are recommended) All coordinates and radius are in screen space.

Example:

50 50 300 255 0 0 20 0.1 1.0

100 100 600 0 255 0 60 0.1 1.0

b. Running

python rayTracer.py --config_file config.txt

The output is named result.png in the working directory.

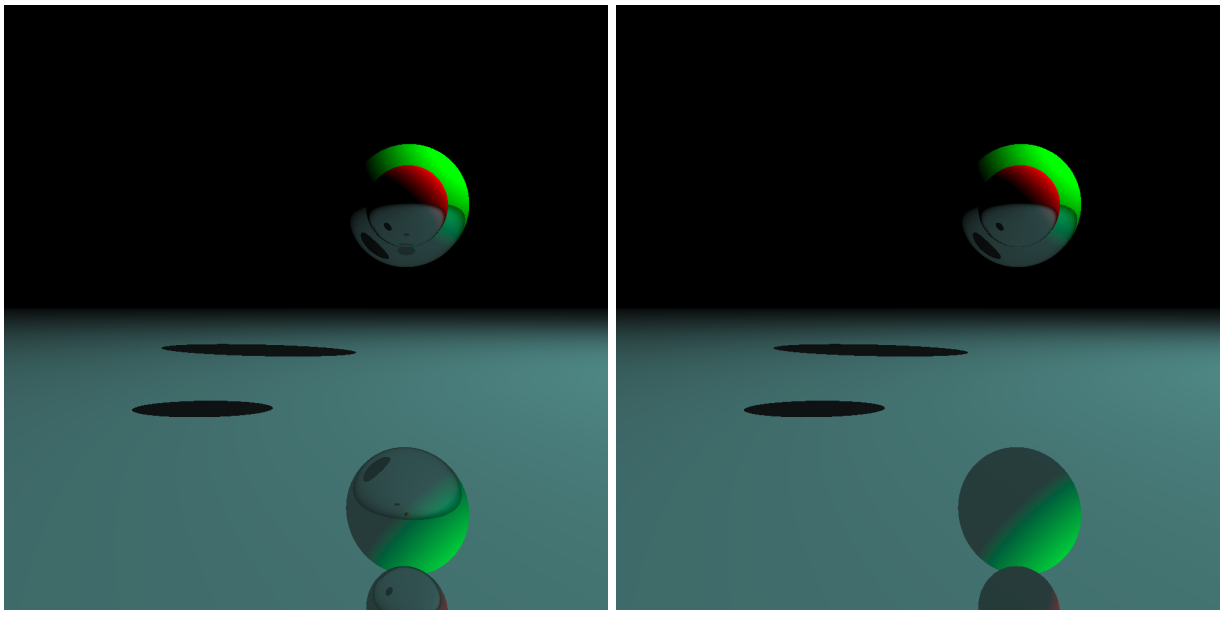
c.Requirements

Numpy = 1.19.2

Matplotlib = 3.3.2

Experiments

I've chosen max depth as 2, meaning that a ray can be reflected two times, at most.

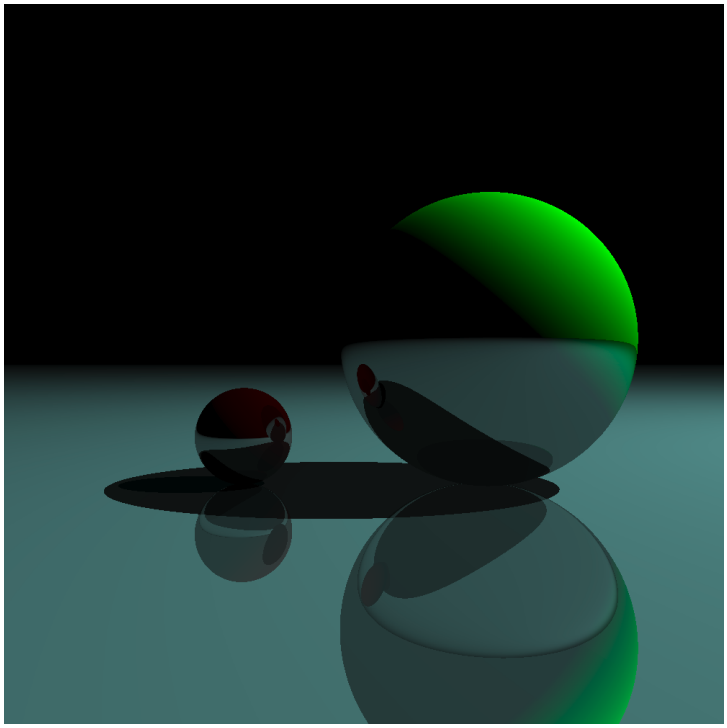


The effect of depth. Left has max depth 2, right has max depth 1.

Config File:

50 50 300 255 0 0 20 0.1 1.0

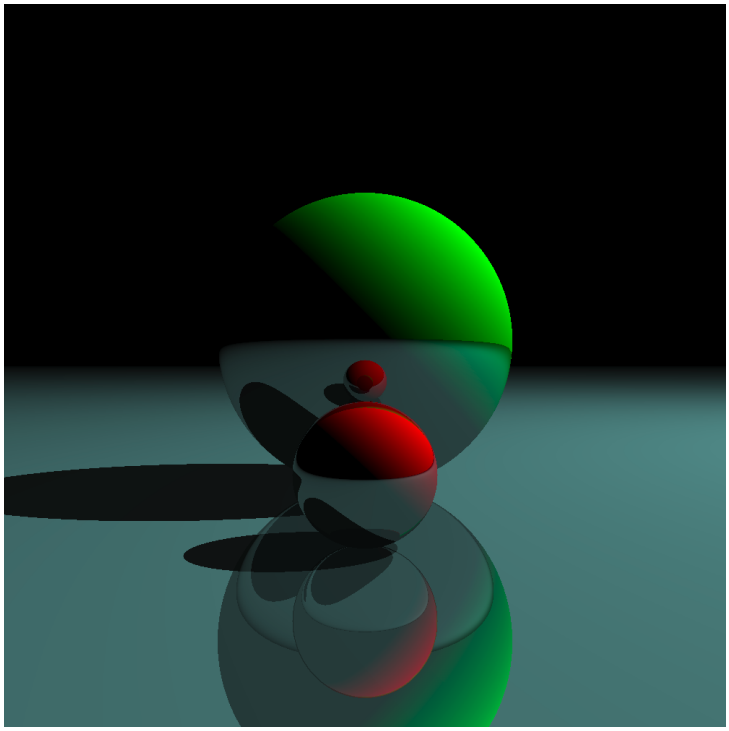
100 100 600 0 255 0 60 0.1 1.0



Config File:

-50 -30 300 255 0 0 20 0.1 1.0

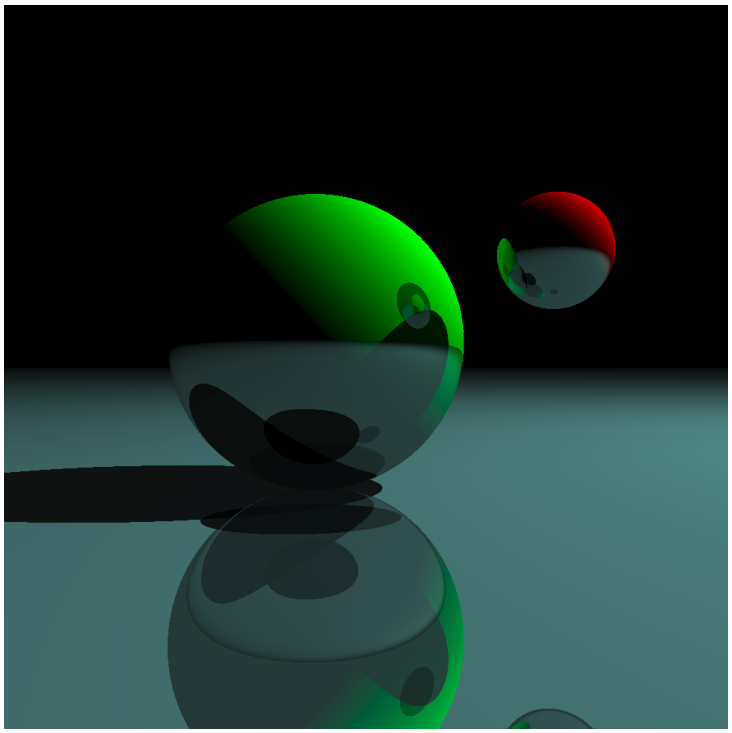
50 10 300 0 255 0 60 0.1 1.0



Config File:

0 -30 200 255 0 0 20 0.1 1.0

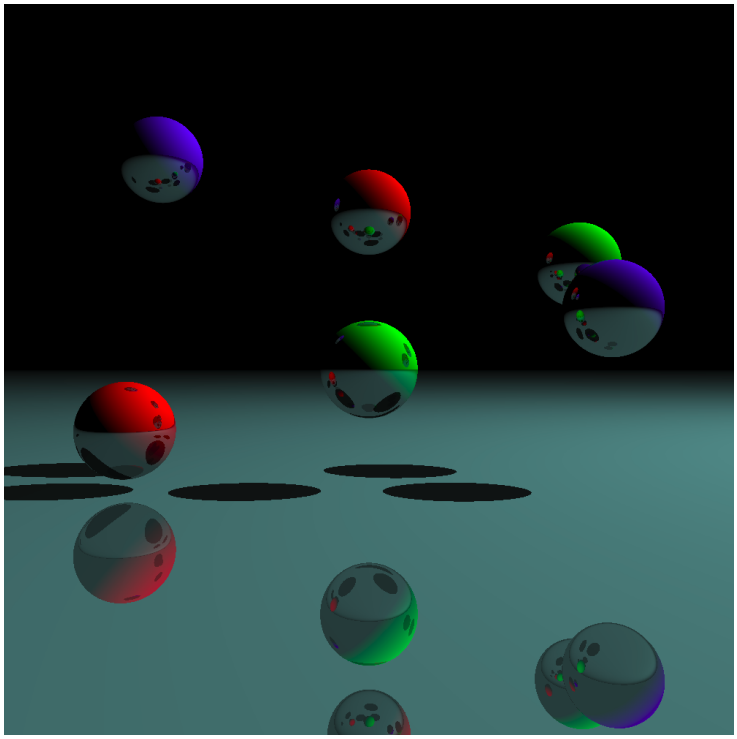
0 10 300 0 255 0 60 0.1 1.0



Config File:

65 40 250 255 0 0 20 0.1 1.0

-20 10 300 0 255 0 60 0.1 1.0



Config File:

-100 -25 300 255 0 0 20 0.1 1.0

0 0 300 0 255 0 20 0.1 1.0

100 25 300 0 0 255 20 0.1 1.0

-100 100 350 0 0 255 20 0.1 1.0

0 75 350 255 0 0 20 0.1 1.0

100 50 350 0 255 0 20 0.1 1.0