

Implementing Ray-Tracing Algorithm

qiangzhiwen 515030910367

1 Question

Write a program to implement the ray tracing or ray casting algorithm to synthesize an image of some virtual environment (with some graphical objects of your own choice). The image should contain the show effect of the virtual environment.

2 Principle

- Deciding the primary ray of the scene.
- Check every object of the scene to see if it intersects with the ray.
- If no, this point is illuminated, otherwise, that object cast a shadow on it.
- Incorporating computations for both reflection and refraction.

3 My Work

- Create Point Class in three dimension, overload its basic operation
- Create Sphere Class using the Point Class, create the properties including position, radius, surface color, reflectivity, transparency and emission color.
- Construct the trace function, which takes a ray as argument then test if this ray intersects any of the geometry in the scene. If does, compute the intersection point, then normalize at the intersection point, and shade this point depending on the surface property (transparent, reflective, diffuse). Otherwise it returns the background color.
- Construct the render function, which compute a camera ray for each pixel, then trace it and return a color. Return the color of the sphere at the intersection point if the ray hits a sphere, otherwise return the background color.
- Construct the main function, which create the scene composed of 6 spheres and 1 light. Then, we call the render function once the scene is ready.

4 How to run

I run this program on Ubuntu 14.04 environment. You could use IDE like codeblocks to build this program, after compiling, open the *raytracing.ppm* file and you will see the results.

5 Output

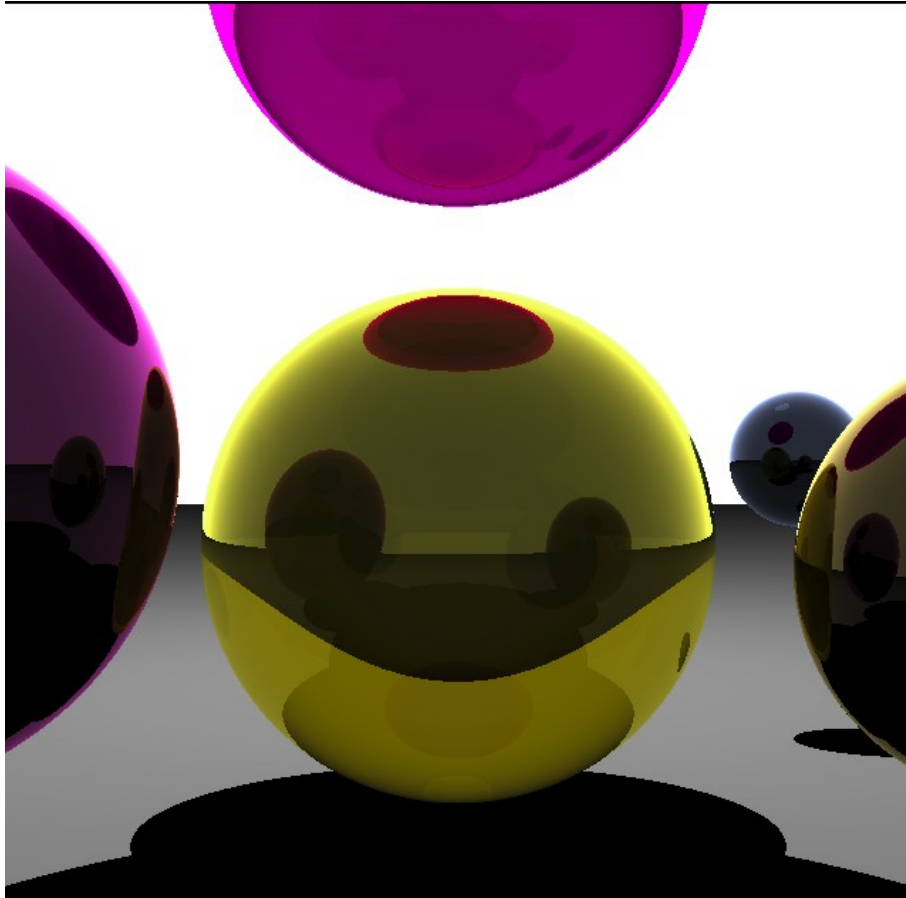


Figure 1: Result of ray tracing