Name: Nathaniel Wong
netID: nwy590
EECS 351-2 at Northwestern University
Intermediate Computer Graphics

Project B: Ray Tracing different objects and materials

Goals

The goal of the project is to demonstrate a complete ray tracing system with a library of materials, custom objects, light sources, a Canonical Viewing Volume, user adjustable camera position and angle, camera projection.

To achieve this, I provide 3 different scenes demonstrating the various materials and phenomena that result from the ray tracing system.

User Guide

To run the program, unzip the project folder and open the webpage WongNathaniel_ProjectB.html. You will see a two side by side square canvases on the top of the screen that will display the simulation. The user controls are as follows:

W,A,S,D: Zoom the camera in/out or strafe the camera sideways
Mouse: Pan the camera
T: trace a new image on the right side of the screen.

Buttons:

- Super Sampling: toggle the number of color samples used for each pixel rendered (1x1 to 4x4)
- Jitter: add randomness to ray tracing
- Scene Number: cycle between the 3 scenes
- Recursion Depth: change the number of times mirror rays are recursively used
- Toggle Lamps: turn lights on and off
- Position sliders: move the lamp around in the world

Code Guide

The ProjectB.js file is a webGL animation script that creates a webGL context. It includes **Shader** programs that instruct the hardware shader on the generation of the particles. An drawAll function**,** updates and renders the objects, or colors each pixel in turn.

The project is **interactive** since it contains mouse and keyboard event handlers that enable custom user input to have an effect on the simulation. See GUIBox.js with notable functions **myKeyPress() myMouseMove()** and **myMouseUp()** for details.

The following classes are used to create the ray tracing program:

- CCamera: A camera class with describes the viewing and projection values for the CVV. Rays are translated into world coordinates from pixel coordinates using this class.
- CGeom: A class containing a library of geometrical objects, including a **ground plane, disk, sphere, cylinder, and Cube.** Also contains transformation functions for rays between model and world coordinates.
- CHit: A class depicting an intersection between a CRay and a CGeom object
- CImgBuf: a 2D float and int buffer containing the color values for each
- CLamp: A lamp object containing values for its position, ambient, diffusive, and specular light.
- CMatl: A class containing support for **23 different materials**, each with its own ambient, diffusive, and specular reflectance, shininess, emissivity.
- CRay: Depicts a single ray with and origin point and a direction
- CScene: The overall scene object that produces a ray traced image.

Results

Illustrations of the 3D scenes:

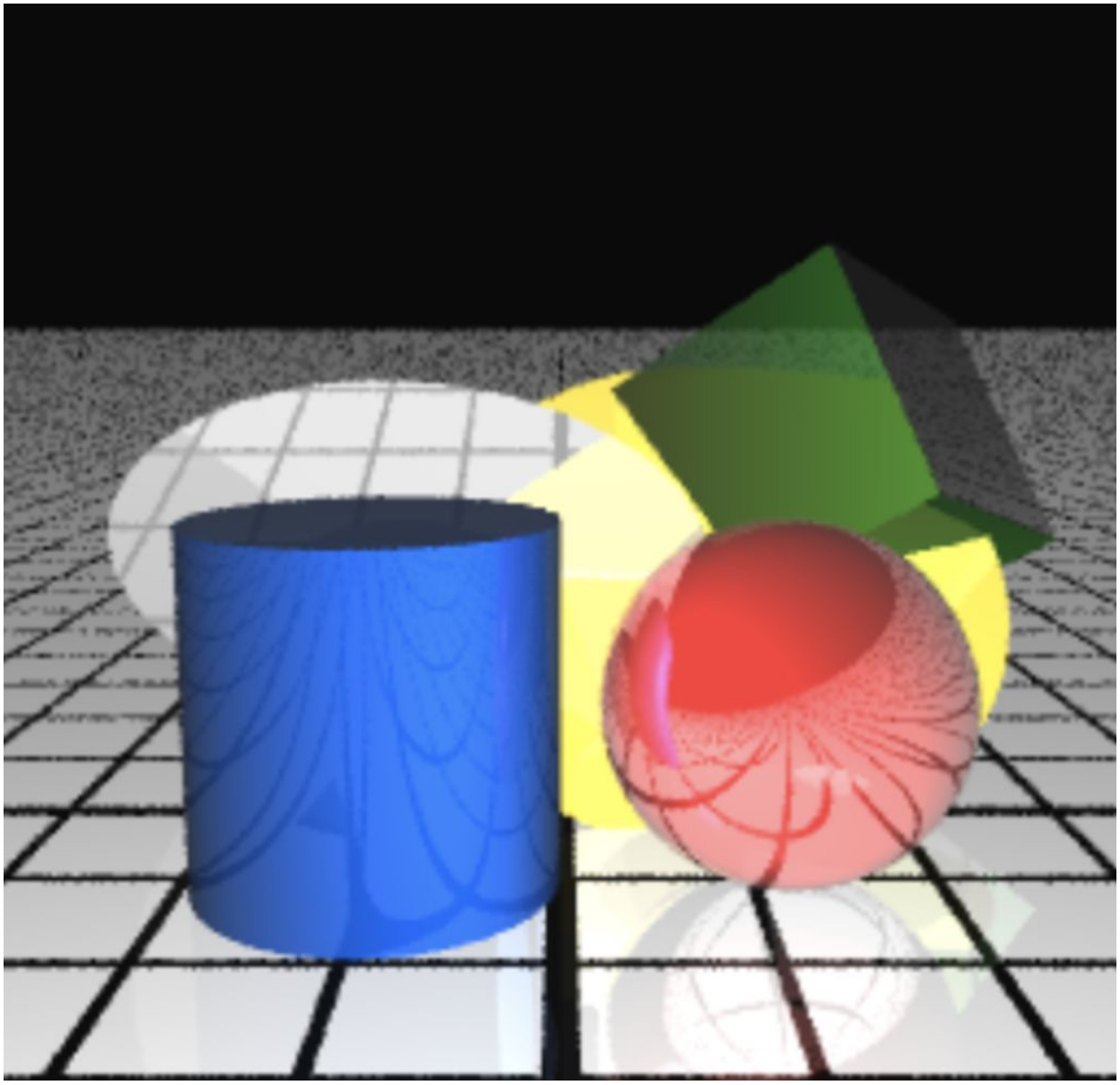

Figure 1: Scene shows various shapes including cylinder, sphere, cube disk and ground plane, with different materials.

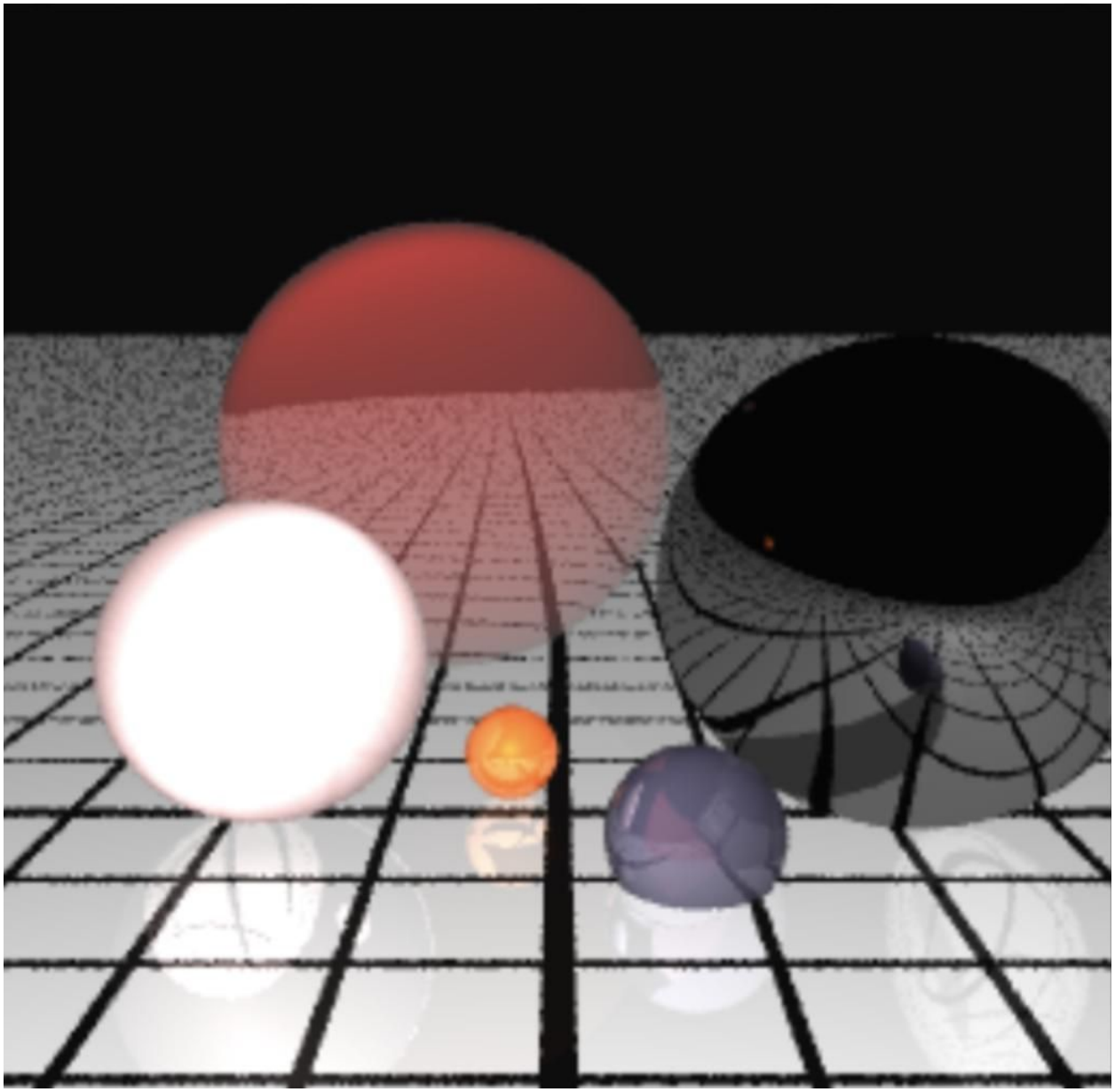

Figure 2: Reflective spheres of different sizes and a range of reflective and non-reflective materials

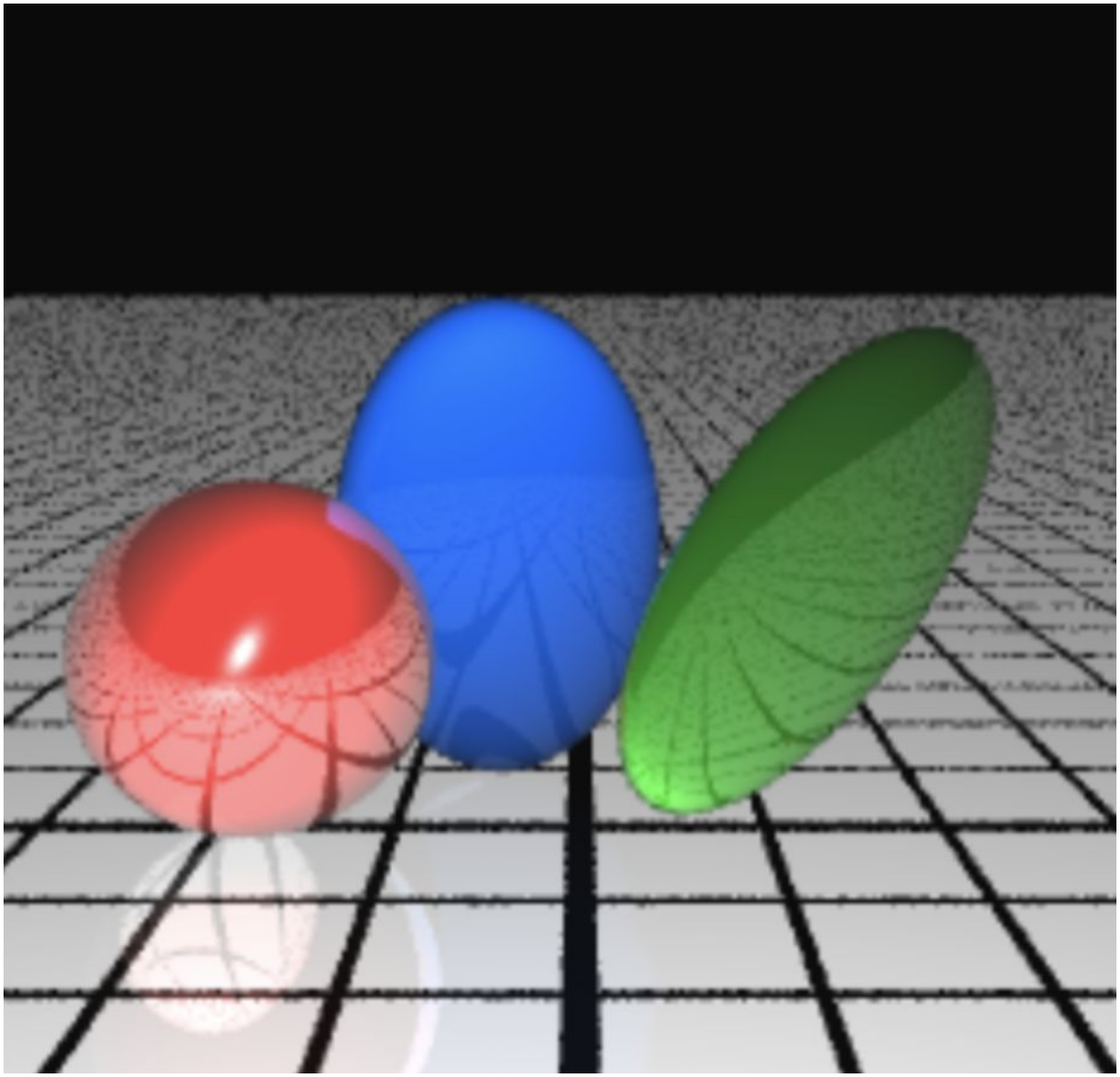

Figure 3: Showing off 3 spheres including transformations for scaling, translation and rotation