

Lab 6 Report

For step1, instead of using the three slabs method, I used the bounding method. There are two points: the minimum extent and the maximum extent of the box. The method can be found in this article: <https://www.scratchapixel.com/lessons/3d-basic-rendering/minimal-ray-tracer-rendering-simple-shapes/ray-box-intersection>. I used most of the code from the article but made changes so that it works with the ray-sphere intersection code.

For step 2, I read and followed the NYU raytracing notes. I defined the axis by having two vectors, pa and pb. Pa is the vector with the minimum y-value and pb is the vector with the maximum y-value. I then followed the algorithm with equation from the site: <https://mrl.cs.nyu.edu/~dzorin/rendering/lectures/lecture3/lecture3.pdf>. I was unable to do the raytracing for the cylinder caps.

Below is the code for step 3:

```
Box *b1 = new Box(Vec3f(0.5, -0.5, -8), Vec3f(2.5, 1.5, -6));
b1->materialType = DIFFUSE_AND_GLOSSY;
b1->diffuseColor = Vec3f(0.6, 0.7, 0.8);

Cylinder *c1 = new Cylinder(Vec3f(0.5, -1.5, -8), Vec3f(0.5, 1.5, -8), 1.0);
c1->materialType = DIFFUSE_AND_GLOSSY;
c1->diffuseColor = Vec3f(0.6, 0.7, 0.8);

// objects.push_back(std::unique_ptr<Sphere>(sph1));
// objects.push_back(std::unique_ptr<Sphere>(sph2));
objects.push_back(std::unique_ptr<Box>(b1));
objects.push_back(std::unique_ptr<Cylinder>(c1));
```

Below is the output image:

