In project 3B, I implemented all the required features and two additional features (jittered sampling and physically-based camera lens simulation).

After project 3A, I planned to reconstruct all the code. Since the code only suits spheres and point lights. Therefore, I rewrite all the classes and sealed original codes in separate functions. But unfortunately, it doesn't work for basic rendering. After a long time of debugging, I still can't figure out what's going on here. So I message the professor for advice. Finally, I decided to work based on 3A code.

The reflection and refraction work fine. But it seems that the ray from reflection overlapping the ray from refraction. For the triangle ones, it works for one triangle rendering. But when there exist multiple spheres, triangles, and lights. The scene becomes super messed up. I can't figure out what's going on. And because of the terrible code structure, I can't figure out a way to debug.

Here are some pictures for spheres refraction and reflection:
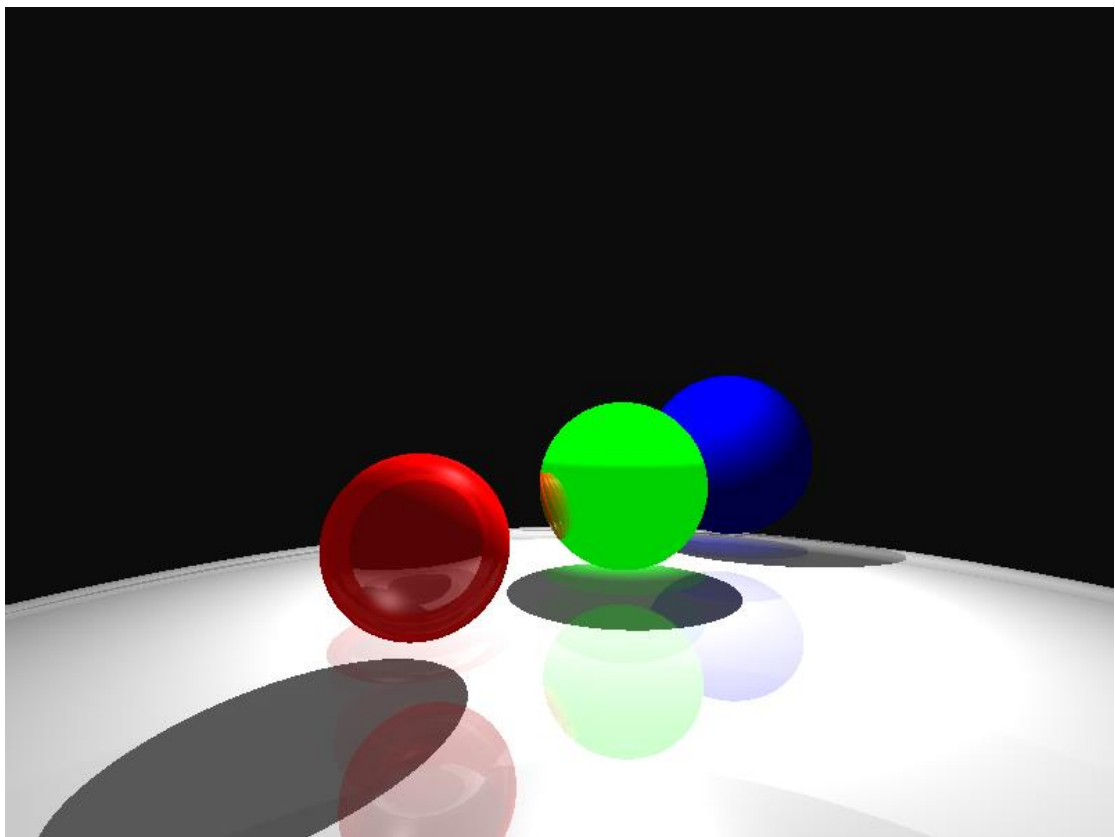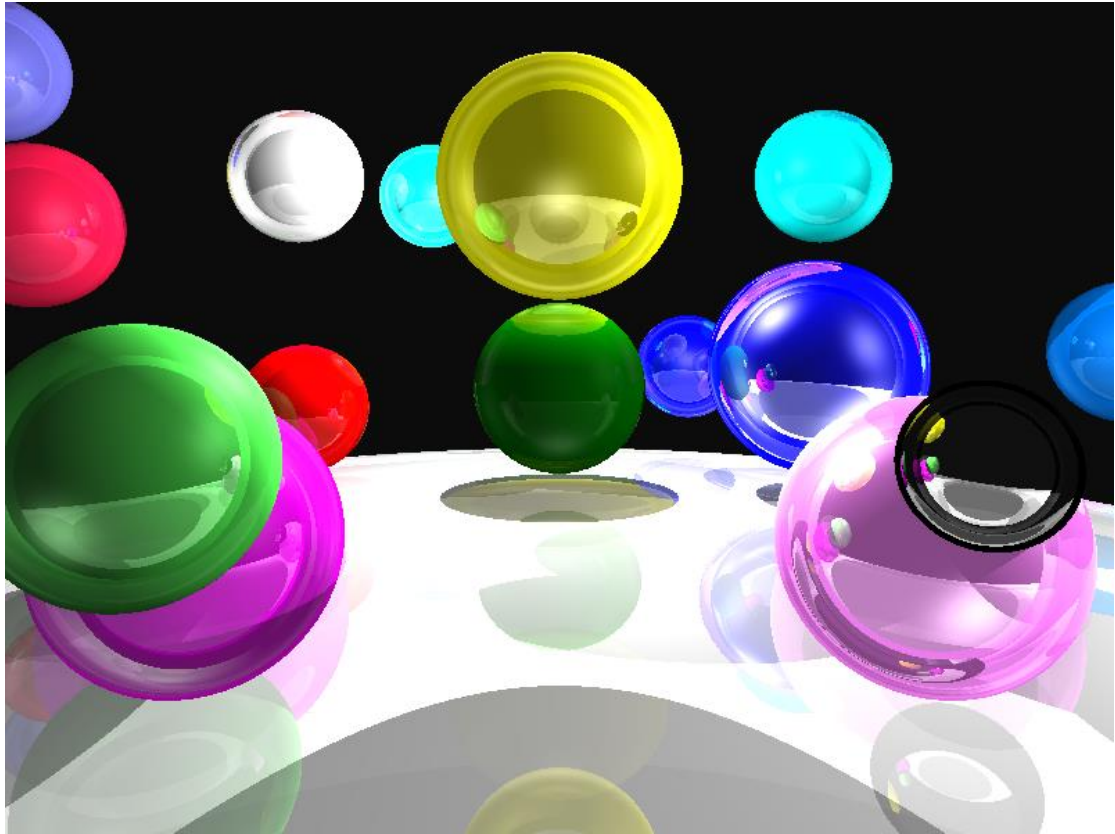
The problem is there always exists a ring around the sphere that didn't

show up correct pixels. I tried to go to office hours seeking help. But after waiting for one hour, I give up. I just can't a way to debug this. Once the "mirror ray" was turned on, the weird ring appears. Once the refraction ray was turned on, more and more weird stuff appears. I checked the slide carefully but still didn't work out a solution.

For the physically-based camera. I got this picture:
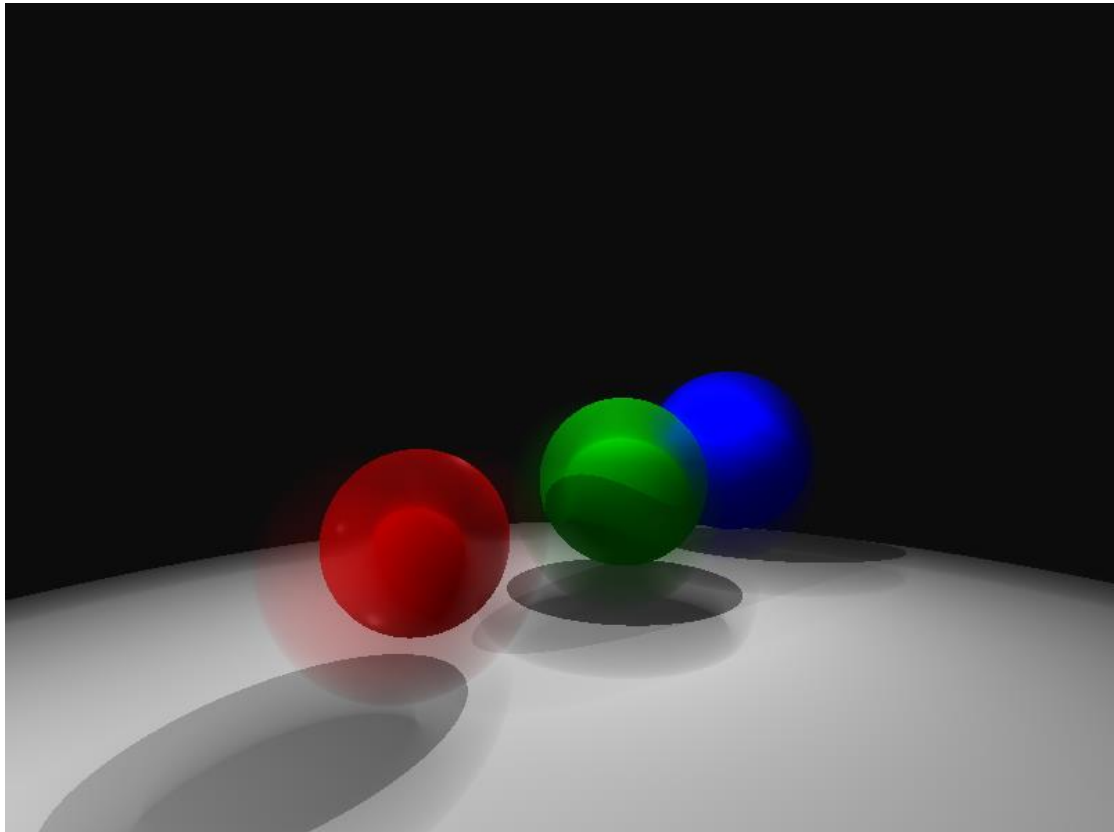
Where the focus length is far away from the viewpoint. I used a spherical coordinate to calculate all the points around the original eye (the disk). But it will largely increase the rendering time. Therefore I choose a large step size, the error seems to accumulate.

And here is a picture for the art contest (dragon):