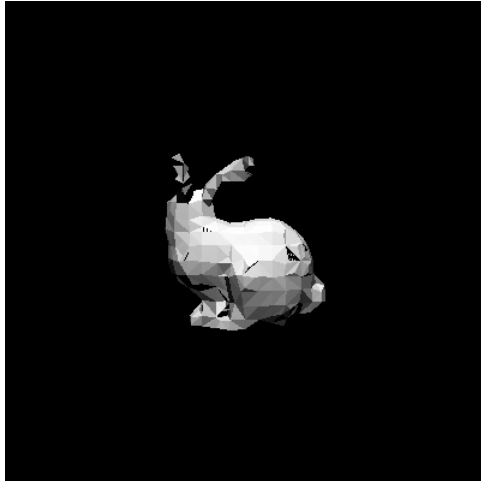


MAT594G - Realistic Image Synthesis (Fall 2011)
Course Website: <http://mat.ucsb.edu/594G/>
Author: RJ Duran
Email: rjduranjr@gmail.com (rjduran@mat.ucsb.edu)
Date: 11/1/11
Homework: #3

1.1 Ray-Triangle Intersection

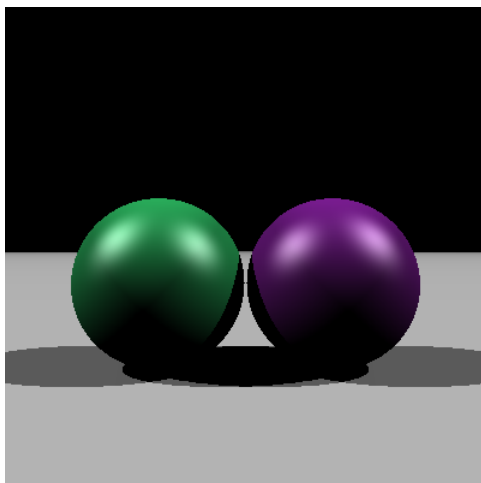
I implemented the ray-triangle intersection method described in the CS465 paper by Steve Marschner. It took a few hours to implement and debug. This 400 px x 400 px image took about 7 mins to render. Out of all the problems on the homework, this was by far the easiest to implement.



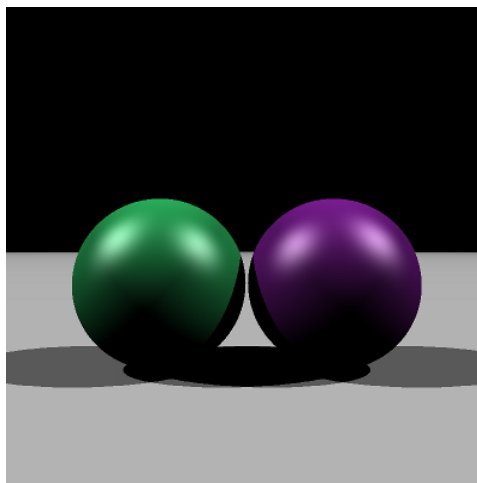
1.2 Distributed Effects

Pixel Super Sampling (Anti-aliasing)

1st attempt at implementing distributed ray tracing. This is a simple implementation of regular sampling with $n = 2$ (each pixel is divided up into 2×2). I increased the n value to 4 and saw a tiny change in quality. I then increased n to 16 and implemented uniform random distribution per ray. It took a long time to render and showed minor improvements. Typically it took twice as long to render each time I doubled the N value.



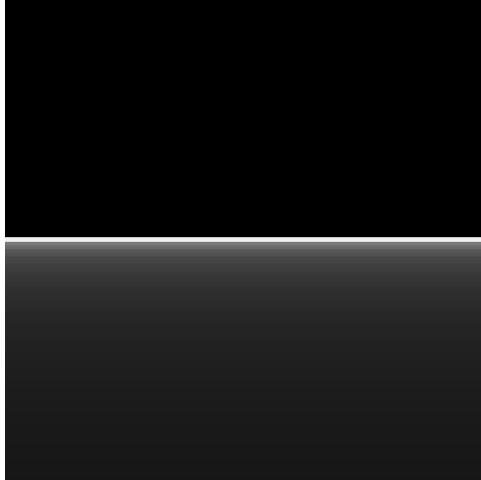
PIXEL_N = 2



PIXEL_N = 4

Area Lights

I couldn't actually get this working properly. I ran into trouble with the positioning of primitives in the scene and having them show up at the scale desired. This is what I get currently with the positioning of objects and camera.



Depth Of Field

This is another method I wasn't able to get fully working. I think it was due to the same issues that came up with area lights. I was able to make the objects but not display them properly.



Summary

At first in this assignment I started exploring option 2 using PBRT but ran into compiler errors trying to get it working as the book described so I switched to option 1. The ray-triangle intersection was easy to implement and only took a few hours. I started to run into trouble when I wanted to do motion blur. This led me to try to complete the area light problem. I wasn't able to completely get this because of an issue with my positioning of objects. For some reason I can't get objects to show properly in the scene. I just ran out of time to debug it.

I think after working on this assignment I would like to work more with a rendering system like PBRT or LuxRender. I did experiment with LixRender a little bit but opted for trying to get my code to work. Now I know I should have probably gone the other route. Overall I probably spent about 25 hrs on all this.

Answers to questions:

* **Did one effect take much more time than the others to achieve an acceptable level of realism?**

Anti-aliasing didn't take too long to implement. It helps the image look better in general. I wasn't able to get the other's working so I can't really compare the realism.

* **Which effect seemed to impart the biggest boost to photorealism per sample?**

From what I can tell motion blur seems to give the most realistic impression. Area lighting also looks pretty realistic.

* **Which effect contributed the least amount of realism to the scene?**

I would say glossy reflections.

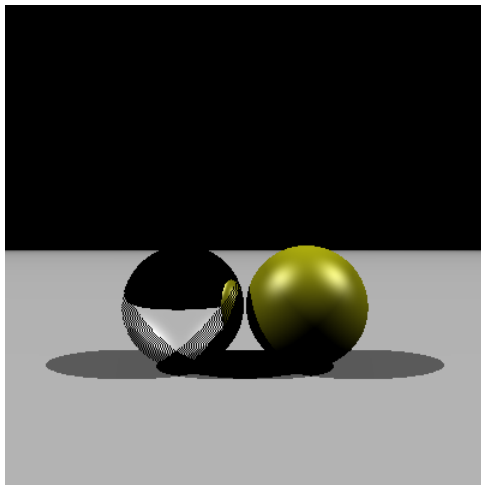
* **If your compute budget was increased to 8 hours, what would you do differently in order to create a more realistic image?**

I would probably work with a different renderer.

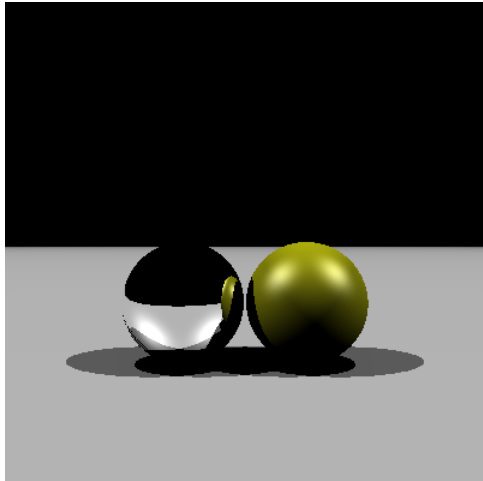
Other Issues Resolved (From HW2)

My issue with the all the shadows not being included under the object was due to not accumulating the color values from light sources when a shadow ray intersected with an object.

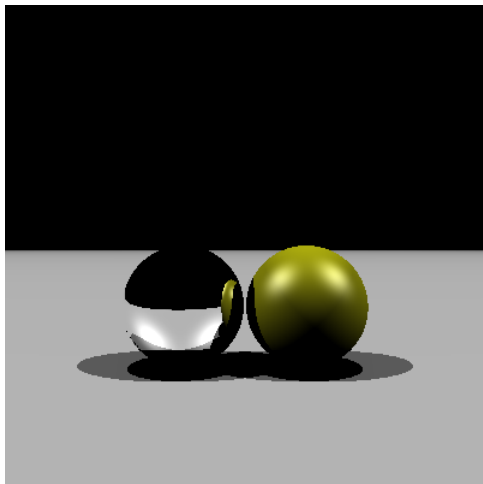
Reflective object artifacts semi-fixed. As I increase the distance of the intersection point for reflections above the mirror sphere along the normal direction with a point offset number I start to reduce the artifacts that show up. However, I don't get the same shadow effect below the center of the two spheres as shown in the HW2 handout.



```
shadowRayOrigin += 0.01*n;
```



`shadowRayOrigin += 0.1*n;`



`shadowRayOrigin += 0.2*n;`