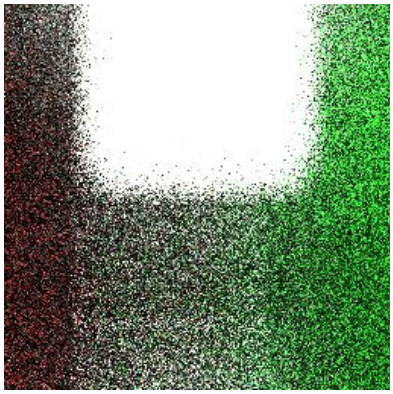
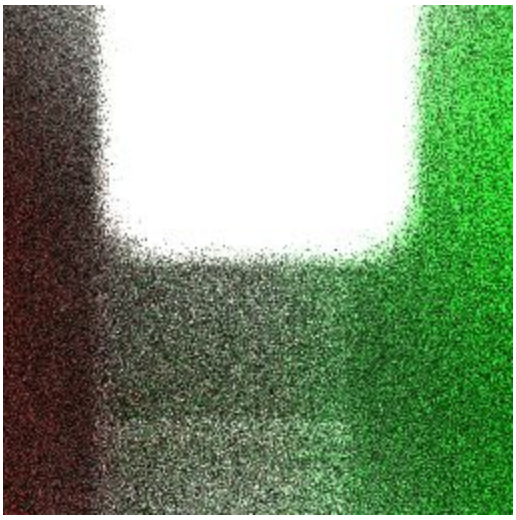
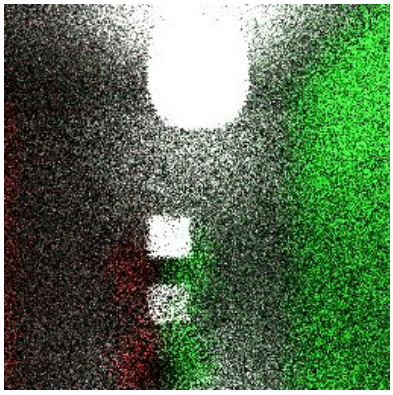
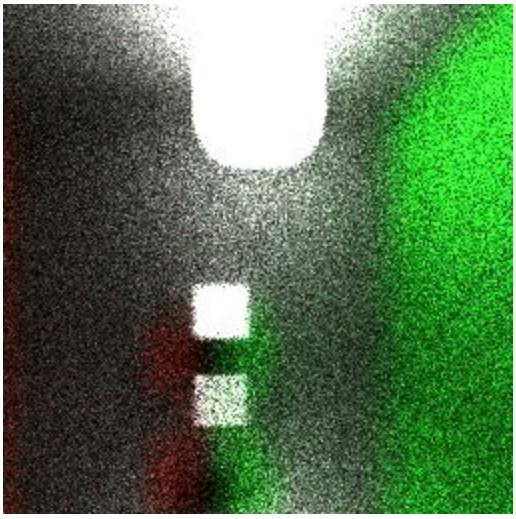
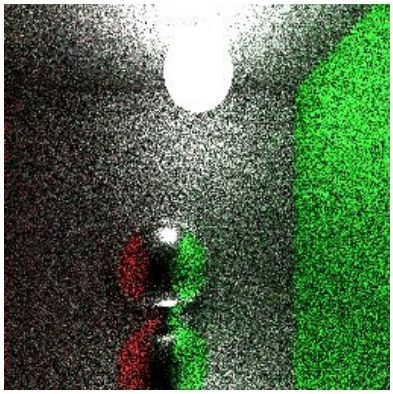
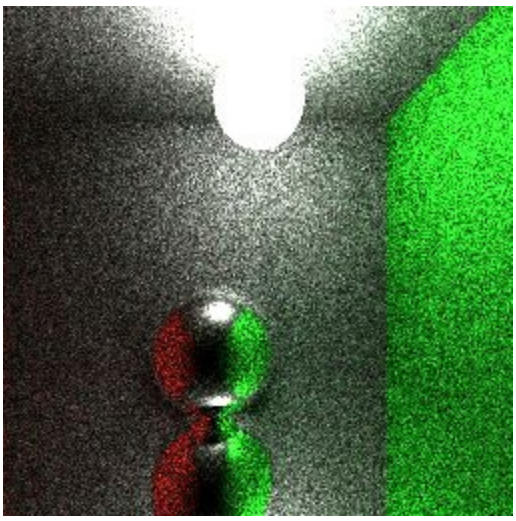
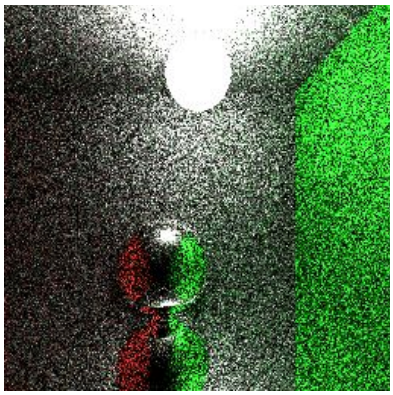
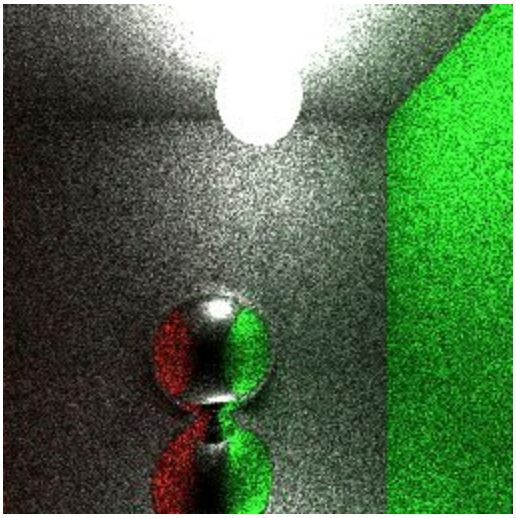
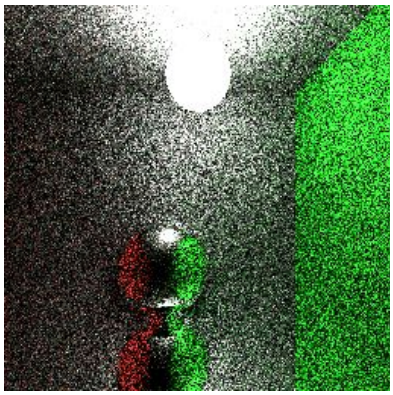
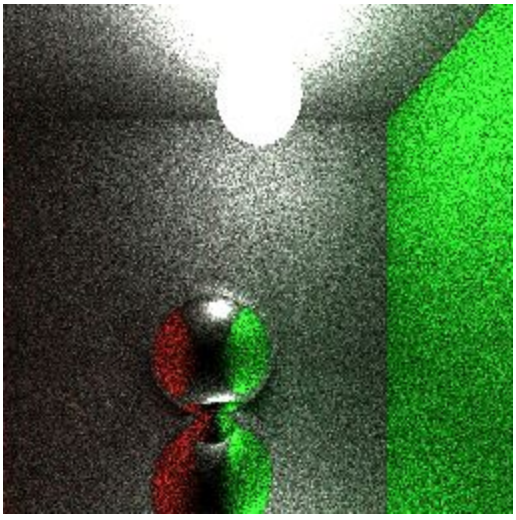
CPSC591 Assignment 3
Brian Yee
00993104
T01

Each image was rendered at 255x255 resolution.

100 samples per pixel	400 samples per pixel	pixel jitter range
		[-1/2, 1/2]
		[-1/10, 1/10]

		[-1/100, 1/100]
		[-1/1000, 1/1000]
		[-1/10000, 1/10000]

Task #1: Generate float random number between the float range min and max

I am generating the number by taking a random number between [0,1] and multiply it by the delta between min/max and offset the number by min.

Task #2: Write the code for creating a random 3D ray (hemisphere sampling)

I translated the formulas from the slides into code.

Task #3: Write the code for the core of the Path Tracing MC

Copied ktot value from the slides, and taking a random number from [0, ktot] by multiplying ktot by a random float from [0, 1].

Task #4: Which procedure will you call to find the direction of the diffuse ray?

I am using HemisphereSampling to find the diffuse ray function as it was the best/only function that fits the description.

Task #5: Jitter (pertub) camera's (x,y) position

To find the jitter divisor constant *c*, i just increased it until the results started to look similar.

The current implementation (adding random jitter values to the x,y coordinates) leads to square looking artifacts (visible in the first two images). This could be improved by keeping the jitter value within a certain radius instead.