Ray Tracing Part 2: Monte Carlo Path Tracing

170050099 - B Nikhil

170050080 - Chagam Dileep Kumar Reddy

image.cpp & image.hpp:

1. Added function *sample_pixels()* to image class which returns random sample of subpixel coordinates for a given pixel, which is then used to sample multiple rays through that image pixel
2. Then averaged over the radiance obtained from the above sampled rays in *rt.cpp* .

light.cpp & light.hpp:

1. Created a new class *Area_light_t* for spherical lights sources and added all the related functions like *intersect()*, *sample_from_light()* and *direct()*.
2. Multiple points on the light source are randomly sampled using *sample_from_light()* which gives smooth shadows upon averaging.
3. *intersect()* uses the *sphere_t intersect()* function to check intersections.
4. Local illumination for each shadow ray is computed in the same way as done for *point_light*.

integrator.cpp:

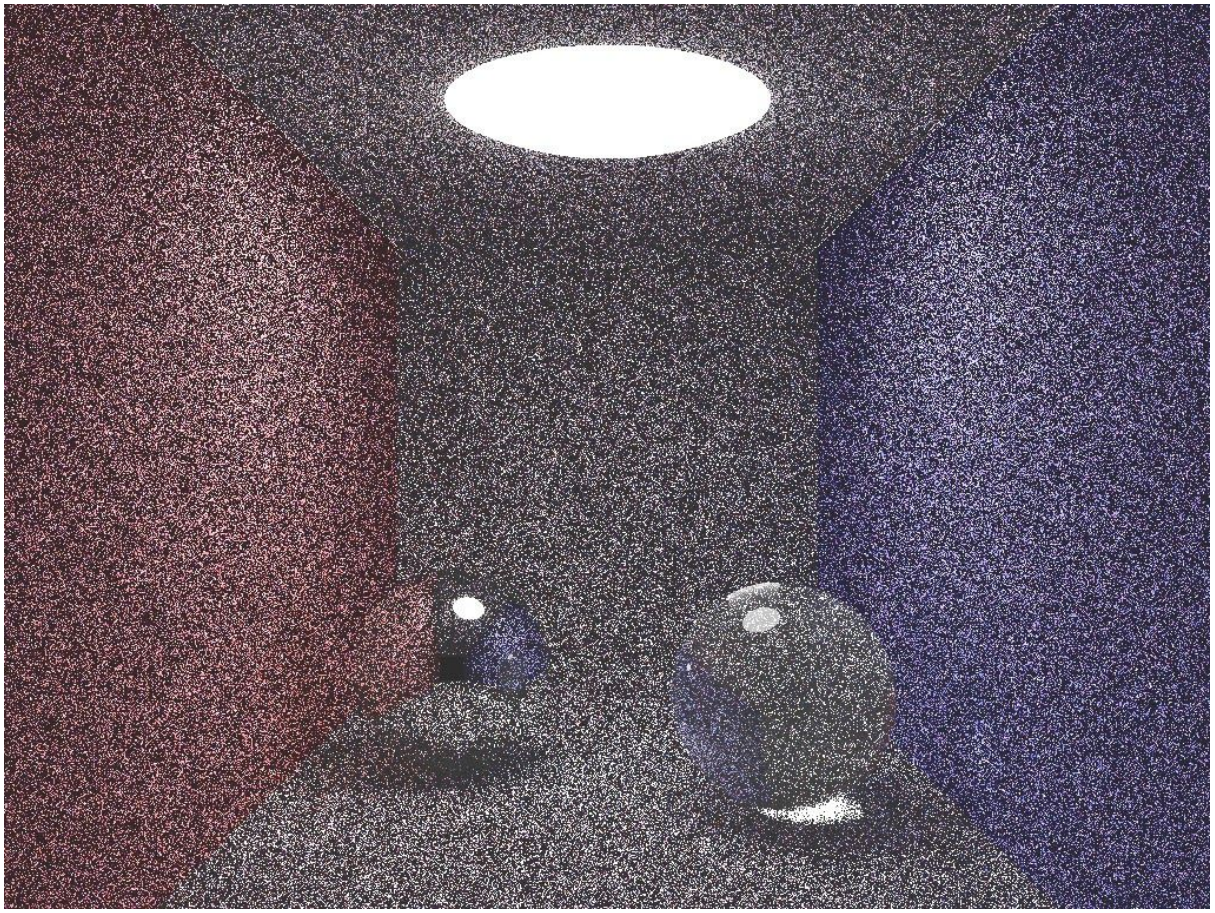
1. Collects light using backward ray tracing.
2. Upon intersection with an object, depending on the object characteristics, only one ray is sampled in path tracing.
3. For ideal diffuse surfaces, ray is sampled uniformly from hemisphere.
4. For ideal mirror surfaces, BRDF is dirac-delta function around reflected ray, so we sampled reflected ray itself.
5. For glass surfaces, which can both transmit and reflect, so the sampled ray is either transmitted ray or reflected ray or both which is chosen using russian roulette as done in smallpt.

cornellscene.xml:

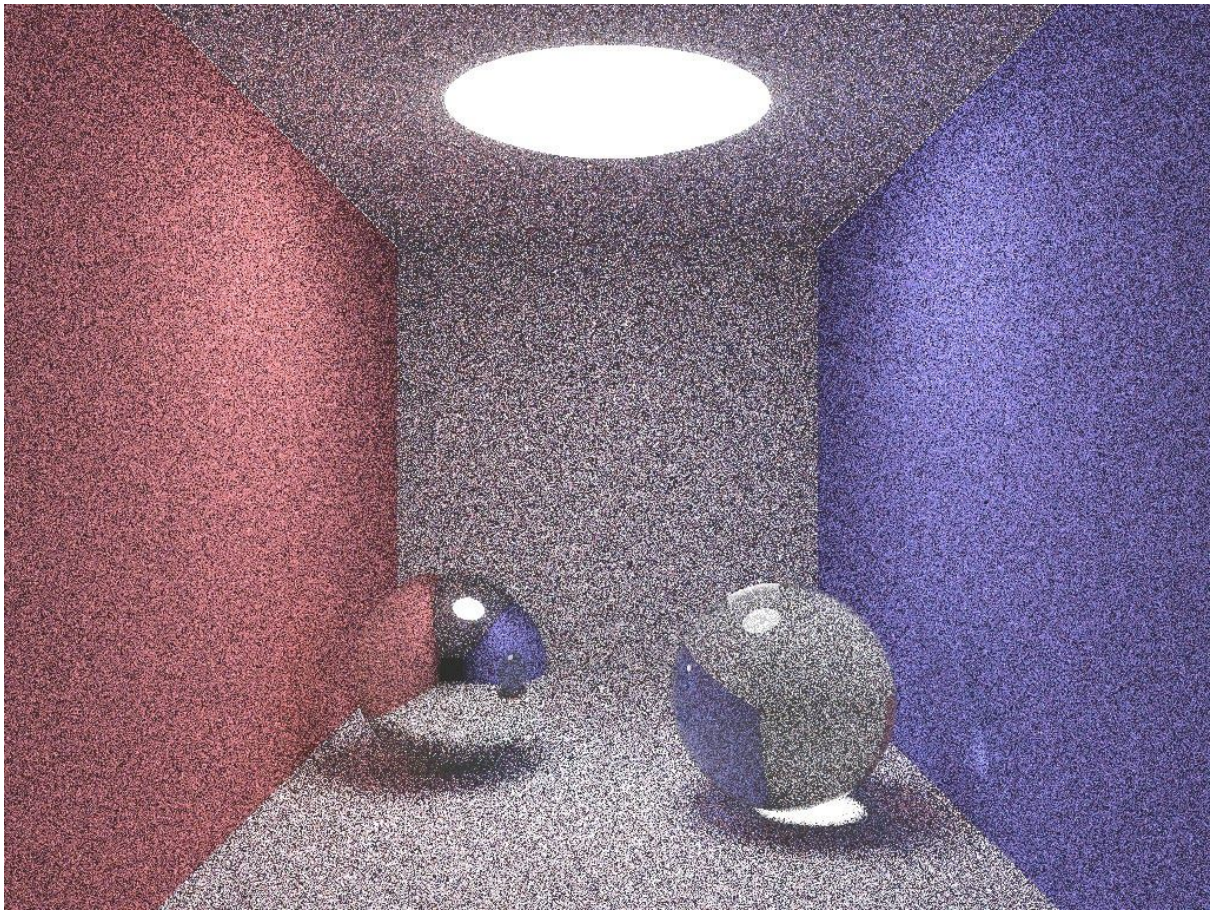
1. Walls for the bounding box are large diffuse spheres.
2. One of the spheres has glass surface while other has mirror surface.
3. Area light source of spherical shape is added to the scene file.

Results: our rendered images for cornellscene2.xml

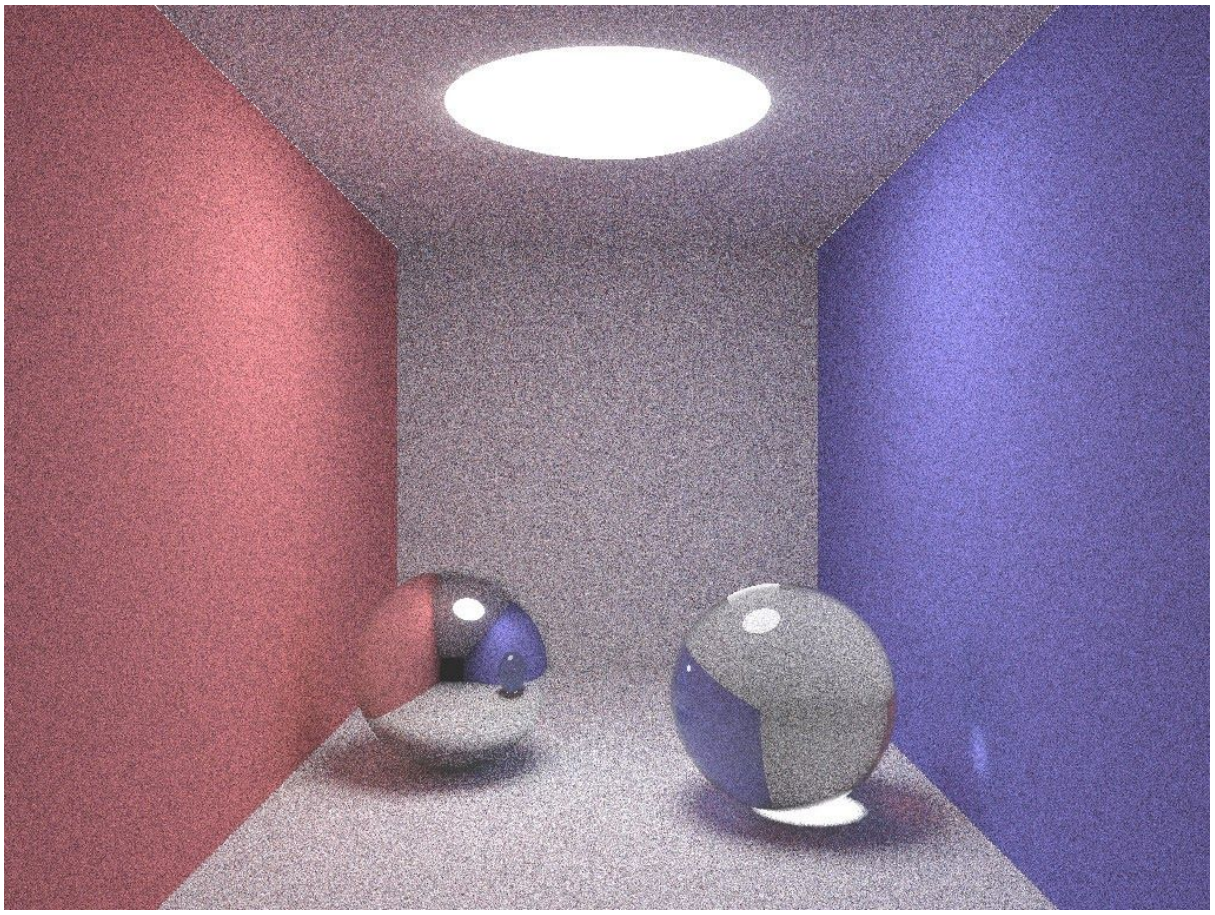
Sample_per_pixel = 8, execution time = 26sec -



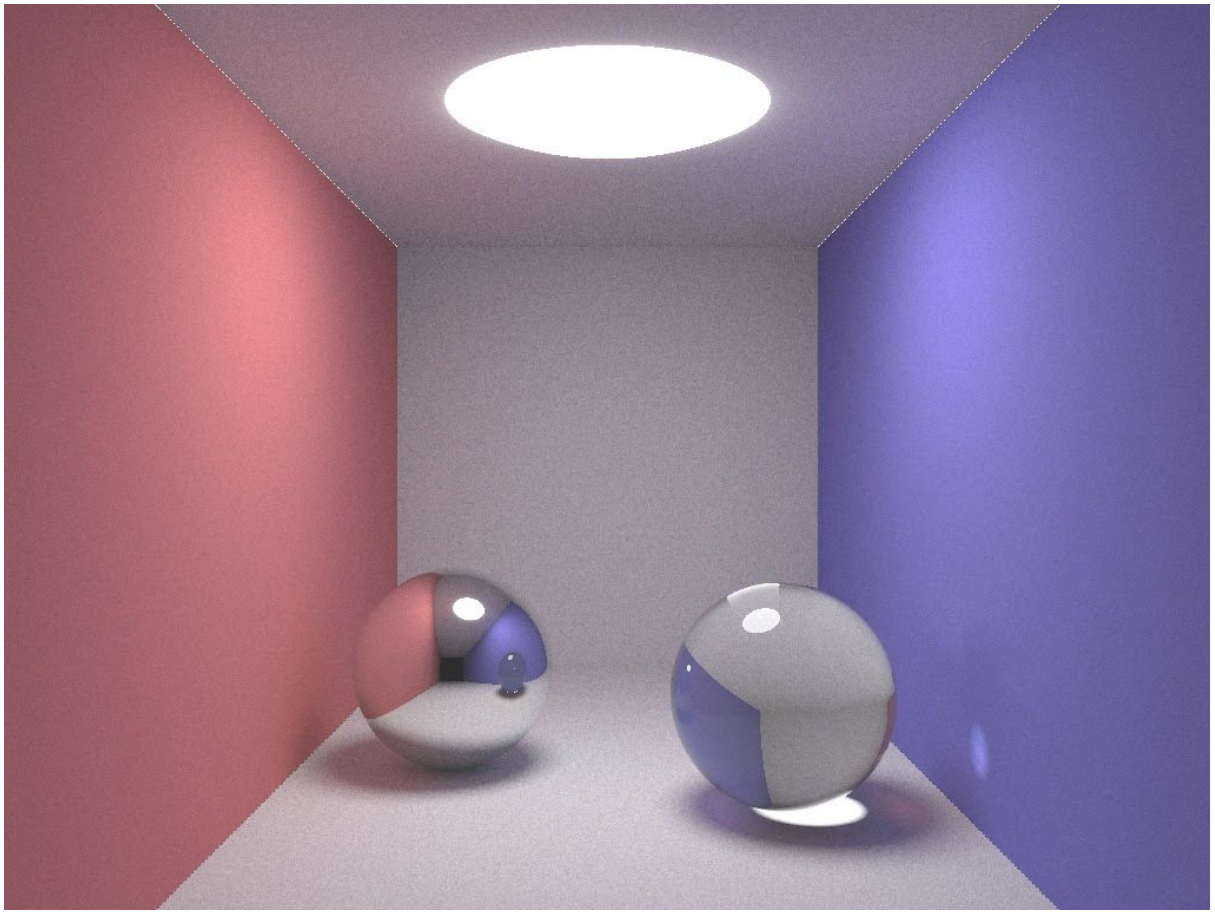
Sample_per_pixel = 40, execution time = 1min33sec



Sample_per_pixel = 200, execution time = 8min36sec



Sample_per_pixel = 5000, execution time = 1hr35min



Instructions:

1. In the parent directory run: make
2. ./rt scenes/<name of .xml file>.xml
3. open <name of .xml file>.ppm file to view output image