Embree pathtracer alteration for school purpouses.

Please note that all images were computed in controlled (as far as can be on windows :) environment on the same machine using Realtime process priority with no other programs running. Resolution of the pathtraced images was set to 510x510. These images were downsampled for this documentation.
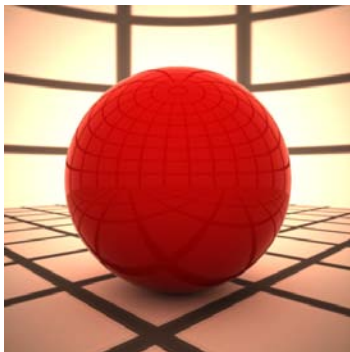
1) **Original algorithm image examples (these are reference images for Russian roulette mod)**
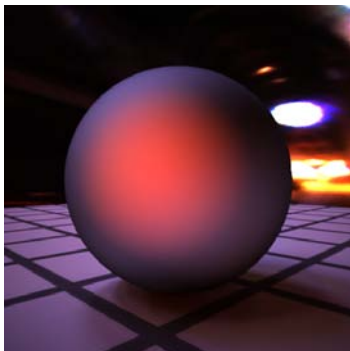   Cornell box Area :

   60000 spp, 3.00132e+007

   Sphere Carpaint :


   60000 spp, 6.86766e+006 ms

   Sphere Phong :

   60000spp, 6.17378e+006 ms

2) **While cycle** replacement of the recursion call - the same number of samples now computes in 97% time (average)
i.e. : build times for Cornell Box Area, Sphere Carpaint and Sphere Phong are 2.80422e+007 ms, 6.63242e+006 ms and 6.56285e+006 ms respectively. L2 error -> 0 and hence the efficiency -> infinity for all the pictures.

16x Differential images :
Cornell Box Area :
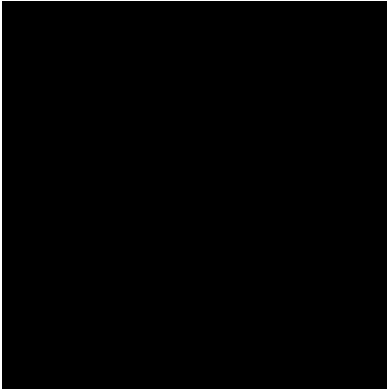
Sphere Carpaint :

Sphere Phong :

On some of the 9 scenes I tested occurred 1 pixel error which I believe isn't the indication of non-functioning algorithm, but rather some inconsistency in the original embree pathtracer. This error is not visible on the 16x diff image. The difference image needs to have 2^20 exposition set for this aberration to be visible.

3) **Russian roulette**: implementing this meant increase in computation time. This is most probably caused by the random generator used for computing the q-factor of survival. The length of the ray was 1(min) to 19 with the mean value being around 2-3. All these statistics are scene dependent.

I experimented with the full Russian roulette solution as well as utilizing the heuristics in which the Roulette is active only after first lambertian surface hit. Both algorithms yield comparable results with the heuristics making the computation slightly faster in general(10% for some scenes). Although I didn't run the full statistics, the variance in the computation time was guestimated to be up to 10%.

The images below represents the same images as mentioned in section Reference images above. Fist of each pair is the image generated with the algorithm utilizing the Russian roulette with the 1st lambertian hit heuristics. Second picture shows the 16x difference between the first image and reference image rendered with the original algorithm.

Results :

a) Cornell Box Area (10000spp) – Russian roulette without heuristics :
   L2 error : 0.0488
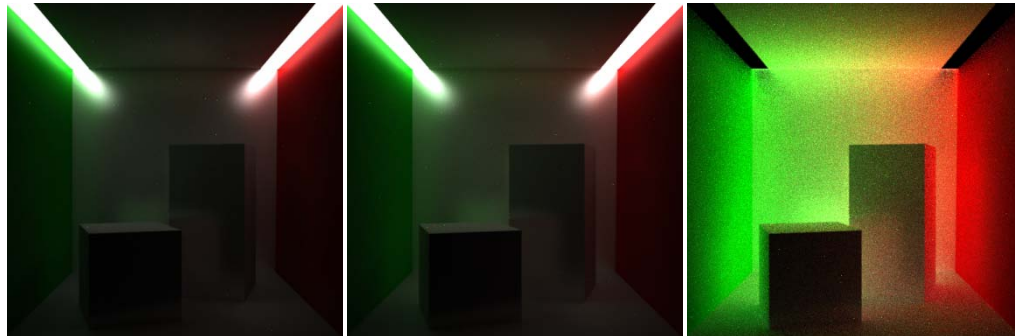   Efficiency : 0.0019871 s^{-1}
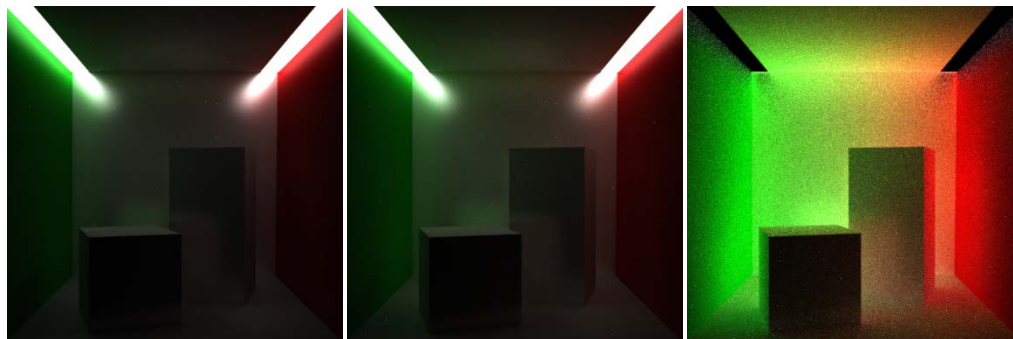
Original pathtracer	Russian roulette	16x difference image
5.01135e+006 ms	1.03127e+007 ms	
	Ray length : (min/max/avg)	
	1/8/2.65692	

b) Cornell Box Area (10000spp) – Russian roulette with heuristics :
   L2 error : 0.0487
   Efficiency : 0.0019776 s^{-1}

Original pathtracer   Russian roulette w/ heuristics   16x difference image
5.01135e+006 ms   1.0383e+007 ms
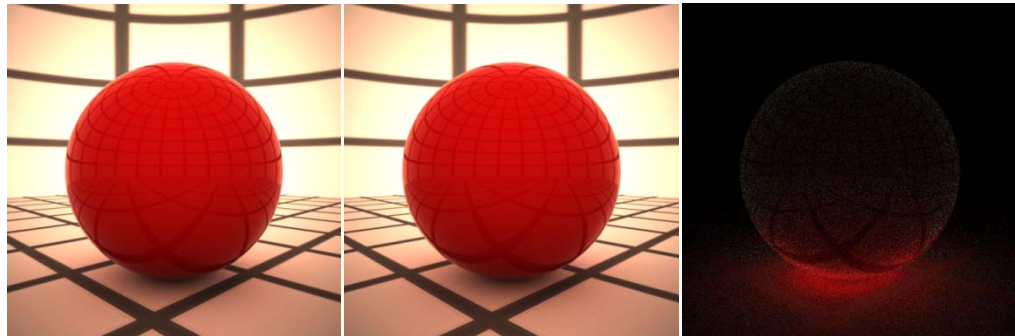   Ray length : (min/max/avg)
                             1/8/2.62004

c) Sphere Carpaint (10000spp) – Russian roulette without heuristics :
   L2 error : 0.0075
   Efficiency : 0.049993 s^{-1}

Original pathtracer   Russian roulette w/ heuristics   16x difference image
1.1446e+006 ms   2.66722e+006 ms
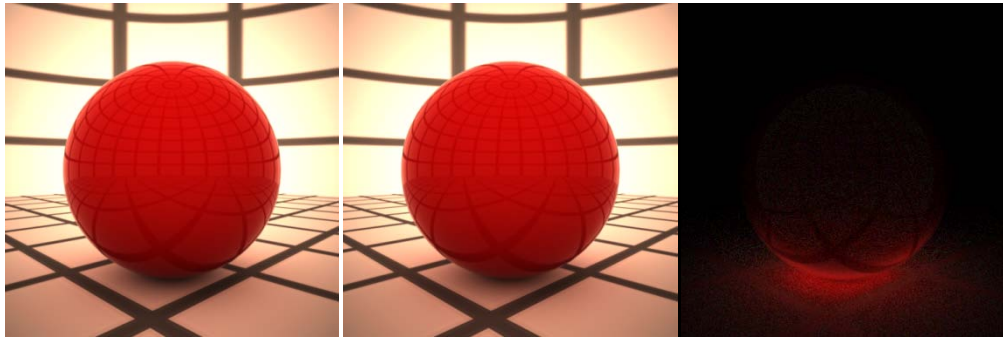   Ray length : (min/max/avg)
                             1/NA/NA

d) Sphere Carpaint (10000spp) – Russian roulette with heuristics :
   L2 error : 0.0071
   Efficiency : 0.056225 s^{-1}

Original pathtracer   Russian roulette w/ heuristics   16x difference image
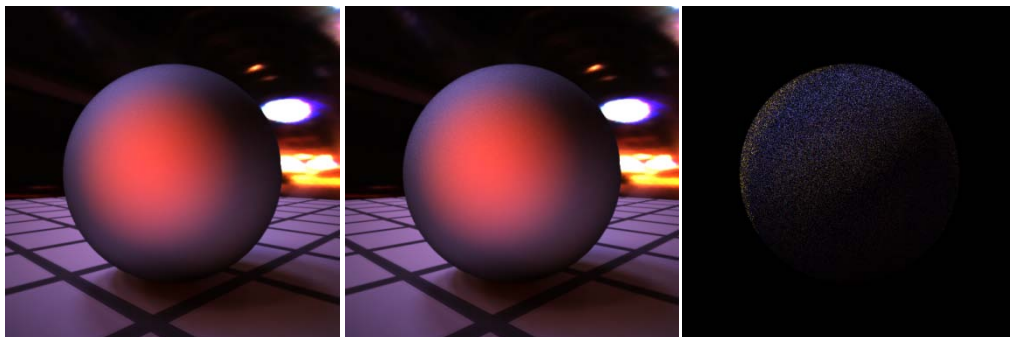1.1446e+006 ms   2.50588e+006 ms
Ray length : (min/max/avg)
1/NA/NA

e) Sphere Phong (10000spp) – Russian roulette without heuristics :
   L2 error : 0.0014
   Efficiency : 0.28266 s^{-1}

Original pathtracer   Russian roulette w/ heuristics   16x difference image
1.0289e+006 ms   2.52776e+006 ms
Ray length : (min/max/avg)
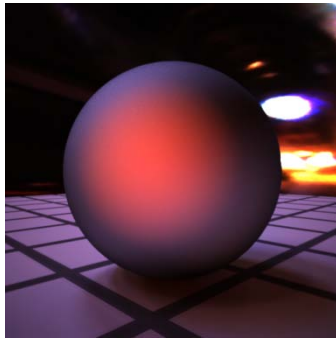1/2/1.169547

f) Sphere Phong (10000spp) – Russian roulette with heuristics :
L2 error : 0.0014
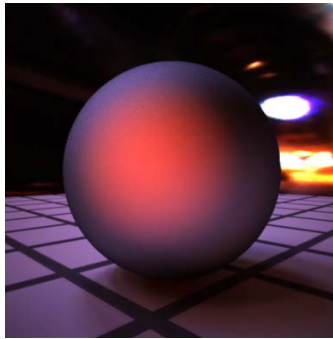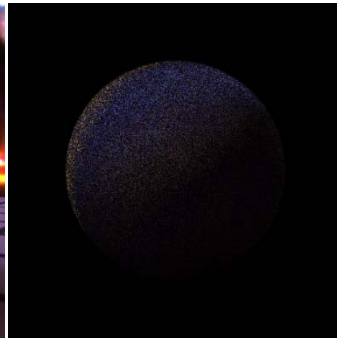Efficiency : 0.28310 s^{-1}

Original pathtracer   Russian roulette w/ heuristics   16x difference image
1.0289e+006 ms   2.52332e+006 ms
                             Ray length : (min/max/avg)
                             1/2/1.168142

4) The next step is to sample only one light source in case of more lightsources in the scene. The particular light source is chosen according the pre-computed distribution function. This pdf usually reflect the usefulness of each and every light source for given geometry illumination computation. This method can be used exclusively or after 1st lambertian reflection (same heuristics used in Russian roulette).

Efficiency of this algorithm suffers again from the use of the random generator to obtain a particular light source for given light source probability distribution.

Reference images were rendered with the Russian roulette & 1st lambertian hit heuristics 10000spp.

a) Cornell Box Area modified (10000spp) – without "1st lambertian hit heuristics" :
    L2 error : 0.0082
    Efficiency : 0.006245 s^{-1}
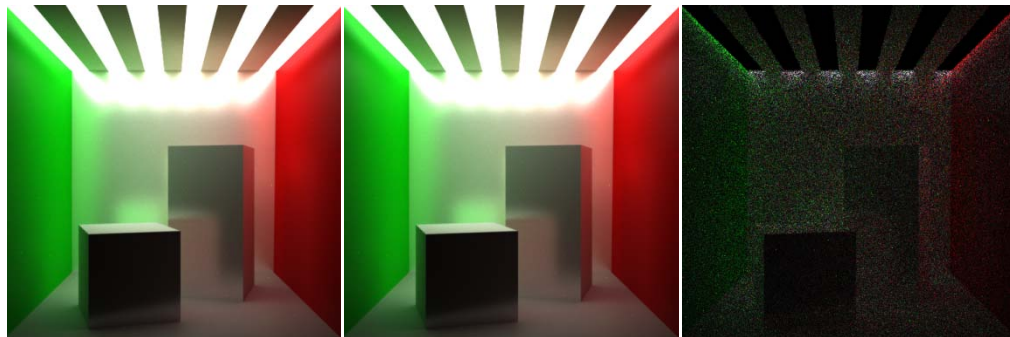
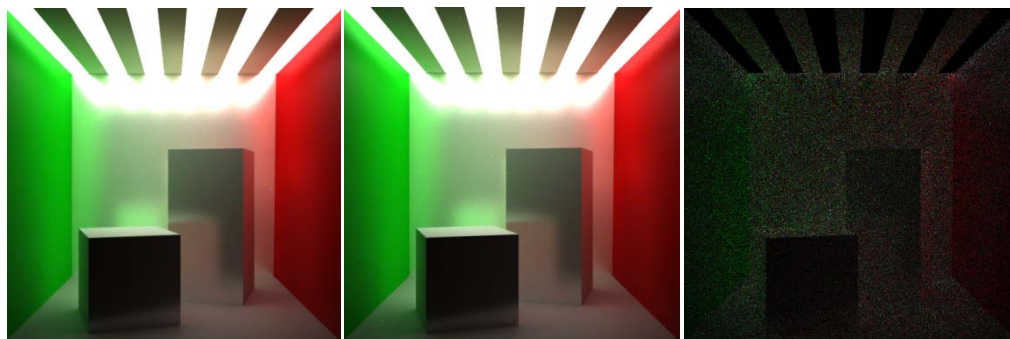Russian roulette pathtracer Russian roulette w/ o heuristics 16x difference image
1.18866e+007 ms   1.95247e+007 ms

b) Cornell Box Area modified (10000spp) – with "1st lambertian hit heuristics" :
    L2 error : 0.0109
    Efficiency : 0.006429 s^{-1}

Russian roulette pathtracer Russian roulette w/ heuristics   16x difference image
1.18866e+007 ms   1.4268e+007 ms

c) Cornell Box Area orig (10000spp) – without "1st lambertian hit heuristics" :
L2 error : 0.0099
Efficiency : 0.00973 s^{-1}

Russian roulette pathtracer Russian roulette w/o heuristics 16x difference image
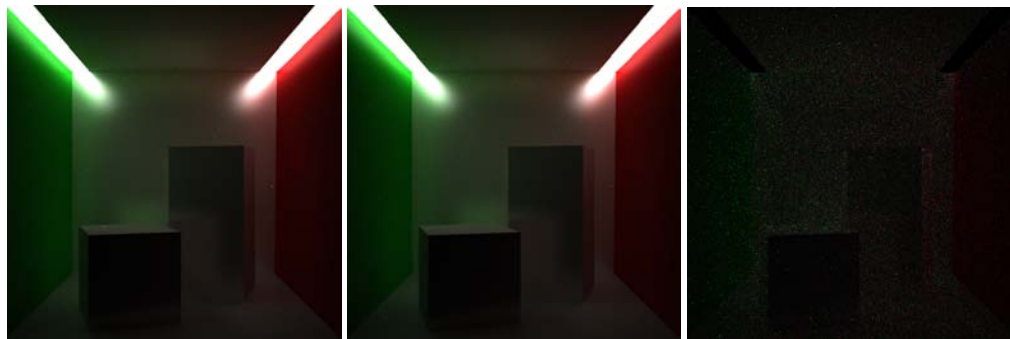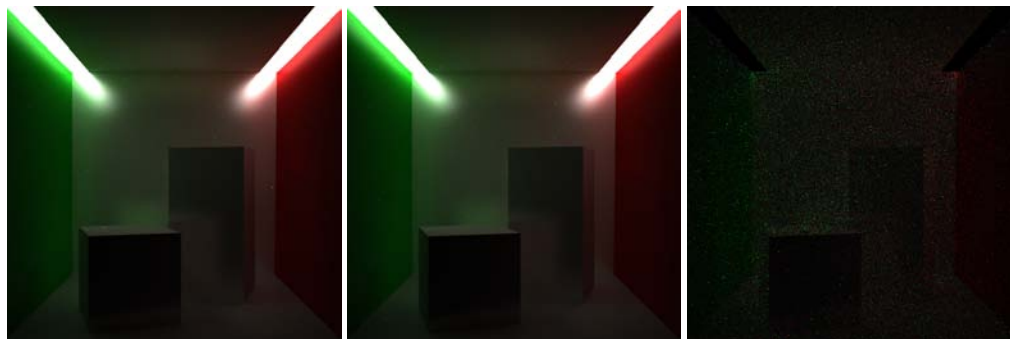1.0383e+007 ms   1.03747e+007 ms

d) Cornell Box Area orig (10000spp) – with "1st lambertian hit heuristics" :
L2 error : 0.0118
Efficiency : 0.00588 s^{-1}

Russian roulette pathtracer Russian roulette w/ heuristics   16x difference image
1.0383e+007 ms   1.43939e+007 ms