Assignment 1 Raytracer

Taekyu Shin in Advanced Computer Graphics

I studied 'A Fast Voxel Traversal Algorithm for Ray Tracing.'

: http://www.cse.yorku.ca/~amana/research/grid.pdf

I also studied this website(Spatial Subdivision) :
http://www.devmaster.net/articles/raytracing_series/part4.php

And I tried to use the resources here.

Usage >

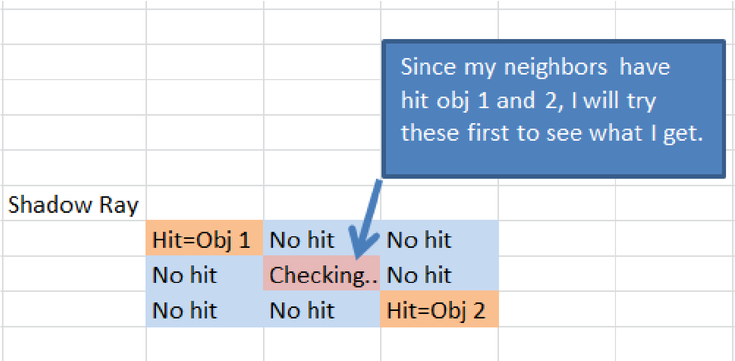
Trace.exe <NFF file name>

Other methods may fail. Putting wrong filenames will fail as well.

I developed in Visual Studio 2010. Therefore, loading in VS 2009 may fail.

The techniques I used:

1. Uniform Grid. – However, it does not have to have unform lengths. Programmers can change the lengths of each x, y, z in whatever way they want. Avoidance of multiple intersection through grids is also implemented.
2. Axis Aligned Bounding Box – I use AABB only to determine in which uniform grid the ray resides or objects are. I do not use AABB to intersect with. I thought that it might slow down the performance.
3. Shadow ray intersection optimization. – I use shadow ray optimization technique. It checks nearby grids for the result of neighboring shadow rays. Try the successful objects of the neighbors first.



Since my neighbors have hit obj 1 and 2, I will try these first to see what I get.

Shadow Ray			
	Hit=Obj 1	No hit	No hit
	No hit	Checking..	No hit
	No hit	No hit	Hit=Obj 2

4. Multi-threading : Simplest kind of multi-threading. I used threadbeginex() and threadend(), the built-in functions in C. I have tried different threads. I usually use 7 to 16 threads. Experimentally, the performance depends on NFF files.
    a. The threads go through each pixels. However, they go through closely together.(Like a checkerboard.) I give thanks to Marc Olano and Wallace Brown for the idea. It helped me because I was actually thinking about what to choose between the checkerboard style and different styles.

- Note
    - I did not implement Cones, so tree.nff does not work.
    - When tracing mount.NFF, there is precision problem. It seems to be hitting the same wall that it is leaving. It is hardly noticeable.

<Tables>

Performance Comparison

*** Please note that I use 100 grids in (-12.5, -12.5, -6) to (12.5, 12.5, 6). It is actually like a small teapot in a stadium. However, it still speeds up. BLANK is NOT tested results due to amount of time consumption.

	Original	New	Speed Up
Balls1	0.594	1.261	0.471055
Balls2	3.276	1.329	2.465011
Balls3	28.336	1.476	19.19783
Balls4	264.702	2.384	111.0327
Balls5	2634.87	9.366	281.3229
Balls6		73.658	
gears1	94.633	6.074	15.58001
gears2	605.999	8.888	68.18171
gears3		10.322	
gears4	4096.02	15.39	266.1481
gears5		16.47	
mount1	1.362	1.458	0.934156

mount2	4.302	1.452	2.96281
mount3	15.751	1.64	9.604268
mount4	63.898	1.661	38.4696
mount5	240.894	2.074	116.1495
mount6		3.194	
tetra1	0.129	0.78	0.165385
tetra2	0.518	0.709	0.730606
tetra3	0.1888	0.745	0.253423
tetra4	6.818	0.752	9.066489
tetra5	24.264	0.854	28.41218
tetra6	98.2	1.026	95.7115
tetra7	424.259	1.78	238.3478

(UNIT : Seconds)

I used Gears2.NFF to determine the effect of how many grids and sizes of grids on the performance.

Below is the comparison based on different number of grids.

# of Grids						
20	50	70	100	150	200	250
46.797	15.521	11.725	9.114	8.29	10.03	13.51

(UNIT:Seconds)

I also used Gears4.NFF to show the performance based on the number of threads.

# of Thread	1	2	7	10	20	50	100	200	500
	61.087	37.102	16.33	16.418	15.78	15.075	15.015	15.287	15.133

Thank you very much.

Taekyu Shin