Gebze Technical University

CSE461

Assignment #1

Ray Tracing

Author

Gökbey Gazi KESKİN

Date

26.03.2023

Compilation and Running

You should install boost library for the parser module I used.

On Ubuntu: `sudo apt-get install libboost-all-dev`
You can compile the code as follows

g++ -o executable_name *.cpp headers/*.hpp -lpthread -Wall -Wunused

And run as:

./executable_name example_inputs/house.xml

Or:

./executable_name example_inputs/house.xml -threaded

First one is the single threaded implementation and the second one is multi-threaded implementation. It uses std::thread::hardware_concurrency() and runs the possible parallel thread amount of threads.

Performance

Depending on the object amount on the scene, rendering process can take up to ~10 minutes in my system. In order to achieve a better rendering time, I implemented a multi-threaded version but, multi-threaded version has it's own overheads such as, while single-threaded version calculates the pixel value and writes it immediately to the .ppm file, Multi-threaded version calculates the pixel values and keeps it in a buffer. After each thread finishes calculating, it writes the buffer to the .ppm file. This means threaded version goes through 2 seperate theta(n^2) loops while the single-threaded version goes through one. Other than that, multi-threaded version also has overhead of creating a large buffer and creating and managing threads. Despite all these overheads, multi-threading speeds up the process up to ~3 times. As can bee seen in the tests below, when scene gets more complex (more meshes), the time difference between two version gets more too.

Results of Performance Tests

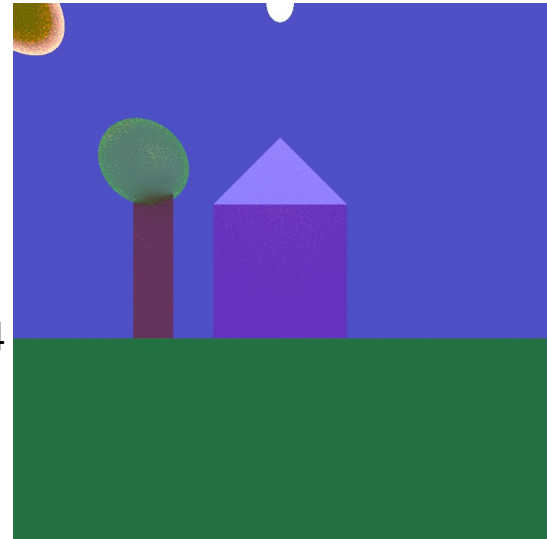I created 3 complex XML files which consists of tens of objects.

Maximum ray tracing depth is 50 for all tests.

Results for the first XML file:

There are 1 point light, 2 spheres, 3 meshes (each consists of 2 triangles) and, one triangle on this scene.

Elapsed Time of Single Threaded Version: 124.314 seconds

Elapsed Time of Multi Threaded Version: 61.812 seconds

Results for the second XML file:

There are 1 point light and 10 spheres in this scene.

Elapsed Time of Single Threaded Version: 119.502 seconds

Elapsed Time of Multi Threaded Version: 97.316 seconds

Results for the third XML file:

There are 1 point light, 2 sphere and 2 meshes (each consists of 12 triangles) in this scene.

Elapsed Time of Single Threaded Version: 567.899 seconds

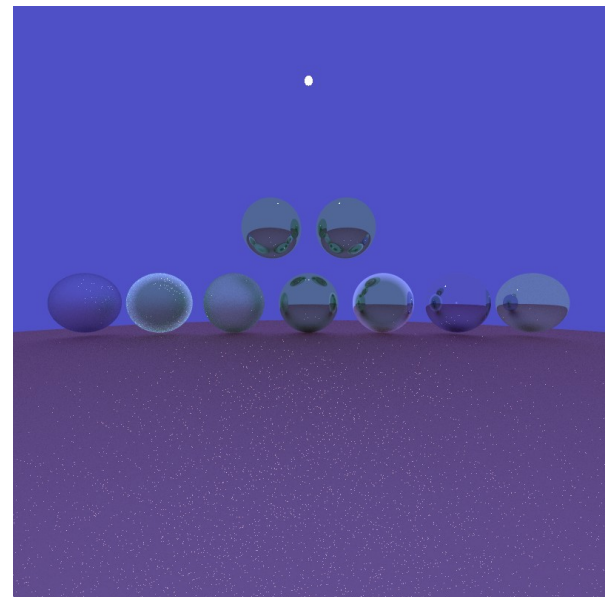Elapsed Time of Multi Threaded Version: 198.143 seconds

<AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>0 0 0
<SpecularReflectance>0 0 0
<MirrorReflectance>0 0 0
<PhongExponent>1000

<AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>0 0 0
<SpecularReflectance>0 0 0
<MirrorReflectance>0 0 0
<PhongExponent>0

Point Light

<AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>1 1 1
<SpecularReflectance>1 1 1
<MirrorReflectance>0 0 0
<PhongExponent>1

<AmbientReflectance>0.5 0.5 0.8
<DiffuseReflectance>0 0 0
<SpecularReflectance>1 1 1
<MirrorReflectance>0 0 0
<PhongExponent>1000

<AmbientReflectance>0.5 0.5 0.8
<DiffuseReflectance>1 1 1
<SpecularReflectance>0 0 0
<MirrorReflectance>0 0 0
<PhongExponent>1

<AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>1 1 1
<SpecularReflectance>0 0 0
<MirrorReflectance>1 1 1
<PhongExponent>1

<AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>0 0 0
<SpecularReflectance>1 1 1
<MirrorReflectance>0 0 0
<PhongExponent>1

<AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>0 0 0
<SpecularReflectance>0 0 0
<MirrorReflectance>0 0 0
<PhongExponent>1

AmbientReflectance>0.5 0.8 0.5
<DiffuseReflectance>0 0 0
<SpecularReflectance>0 0 0
<MirrorReflectance>1 1 1
<PhongExponent>1