Delft University of Technology

Computer Graphics

CSE2215

Project: Ray Tracer, Group 48

Authors:

Sina Şen (4821629)

Julian Biesheuvel (4550188)

Paco Pronk (4842707)

Marilotte Koning (4700163)

Kevin Nanhekhan (4959094)

Joran Heemskerk (4927095)

October 31, 2019

Contents

1 Minimum Requirements for the Implementation **2**
- 1.1 Perform ray intersections with planes, triangles, and bounding boxes. . . . 2
- 1.2 Compute shading at the first impact point (diffuse use and specular). . . . 2
- 1.3 Perform recursive raytracing for reflections to simulate specular materials. . . . 2
- 1.4 Calculate hard shadows from a point light. . . . 2
- 1.5 Calculate soft shadows from a spherical light centered at a point light. . . . 3
- 1.6 Show an interactive display in OpenGL of the 3D scene and a debug ray tracer. . . . 4
- 1.7 Implement a (simple) acceleration structure. . . . 4
- 1.8 Show a scene created by the group, and directly loaded into the application. . . . 5

2 Possible Extensions for the Implementation **6**
- 2.1 Utilizing interpolated normals to smooth objects. . . . 6
- 2.2 Extending the debugger to show the nth reflection of a ray via the key- board, or triggering a ray highlighting and showing command line output of the selected ray's properties . . . 6
- 2.3 Showing refraction behavior through dense object . . . 7
- 2.4 Allowing modification of triangles within the ray tracer. . . . 7
- 2.5 Supporting refraction and the display of transparent objects. . . . 8
- 2.6 Supporting soft shadows and other types of light sources. . . . 8
- 2.7 A numerical evaluation of the performance of your ray tracer. . . . 8
- 2.8 Multicore support of the ray tracer. . . . 8

3 Sources **9**

1 Minimum Requirements for the Implementation

1.1 Perform ray intersections with planes, triangles, and bounding boxes.

This extension is code based only, see source files for implementation. Done by: Julian Biesheuvel, Joran Heemskerk

1.2 Compute shading at the first impact point (diffuse use and specular).

Done by: Paco Pronk, Sina Şen, Julian Biesheuvel

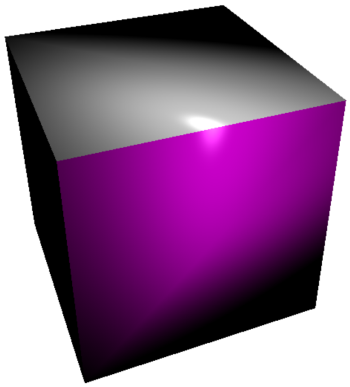
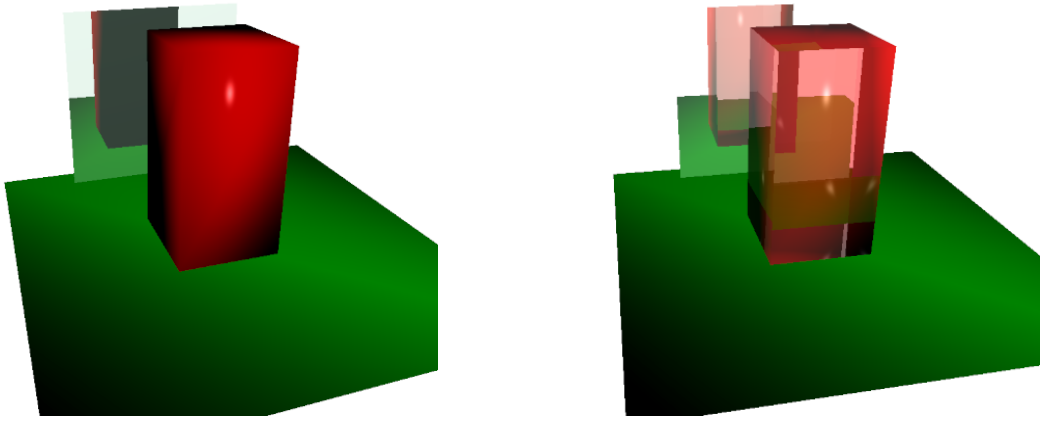


Figure 1: Diffuse and specular components from direct light

1.3 Perform recursive raytracing for reflections to simulate specular materials.

Done by: Paco Pronk, Marilotte Koning



(a) Showing only recursive reflection (b) Showing both recursive reflection and refraction

Figure 2: Recursive raytracing

1.4 Calculate hard shadows from a point light.

Done by: Kevin Nanhekhan

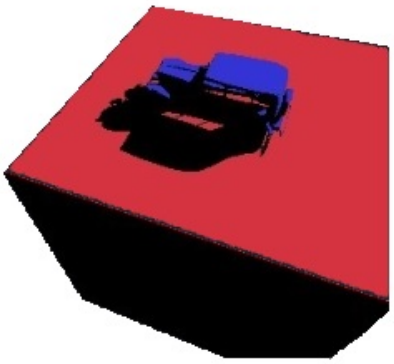


Figure 3: Hard shadow of a car object onto the cube

1.5 Calculate soft shadows from a spherical light centered at a point light.

Done by: Kevin Nanhekhan

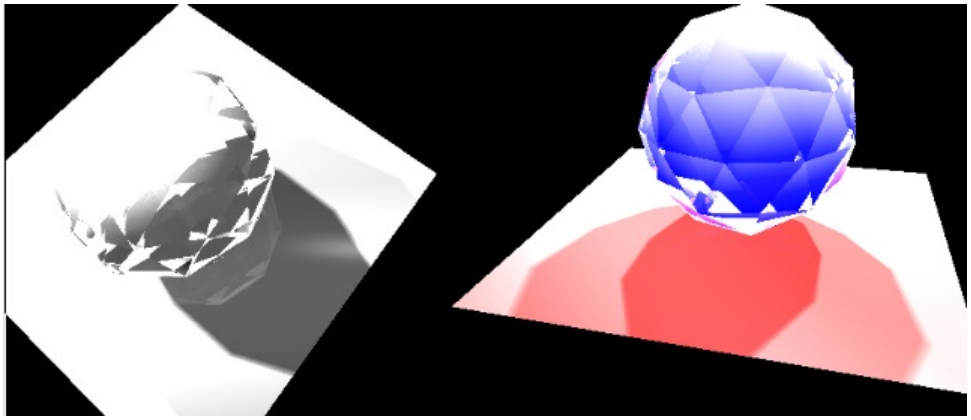


Figure 4: soft shadow of an iso sphere onto the plane

1.6 Show an interactive display in OpenGL of the 3D scene and a debug ray tracer.

Done by: Joran Heemskerk

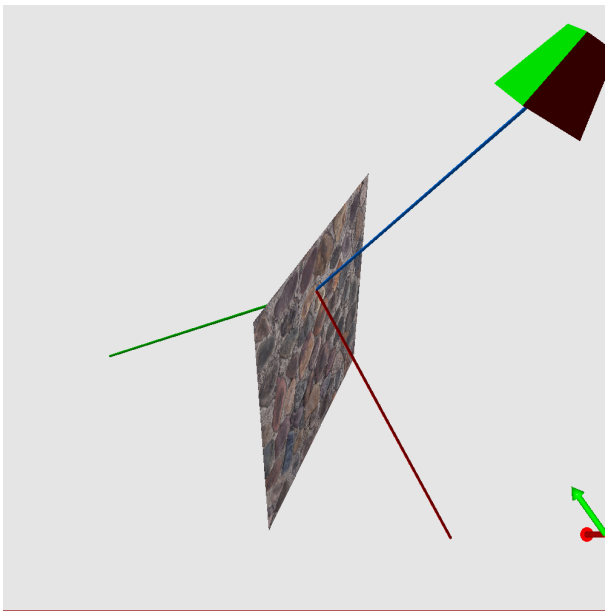
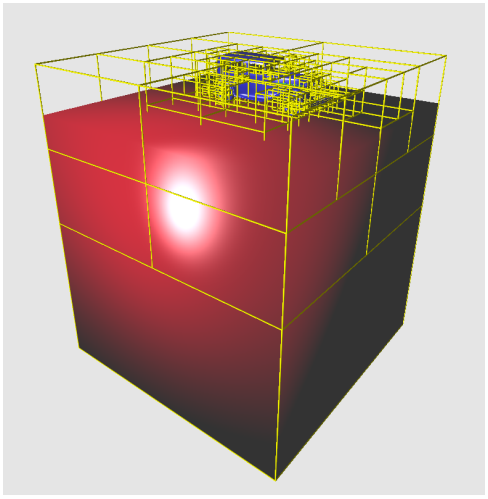


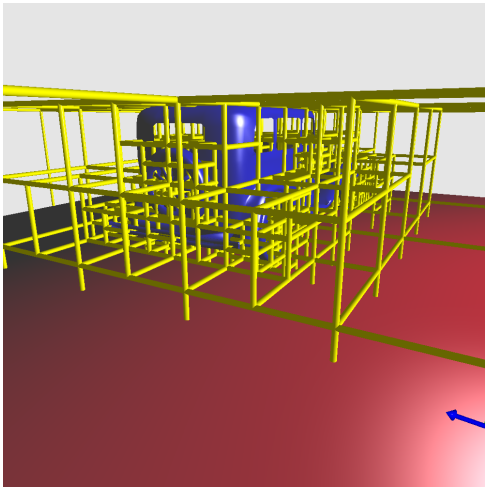
Figure 5: Showing Reflection (red) and refraction (green) against a place object.

1.7 Implement a (simple) acceleration structure.

Done by: Joran Heemskerk, (Sina Şen)



(a) Acceleration structure zoomed out

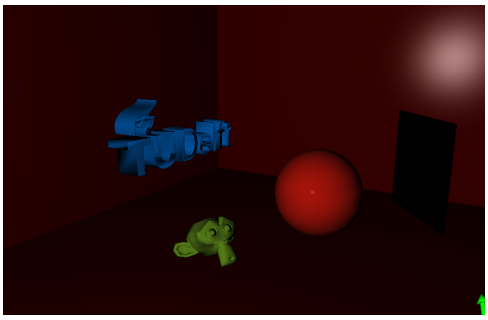


(b) Zoomed in on the car

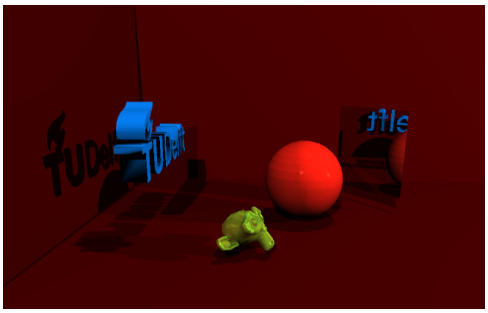
Figure 6: Acceleration structure shown with dodgeColorTest (150 vertices per box)

1.8 Show a scene created by the group, and directly loaded into the application.

Done by: Julian Biesheuvel, Sina Şen



(a) Scene as seen in the application



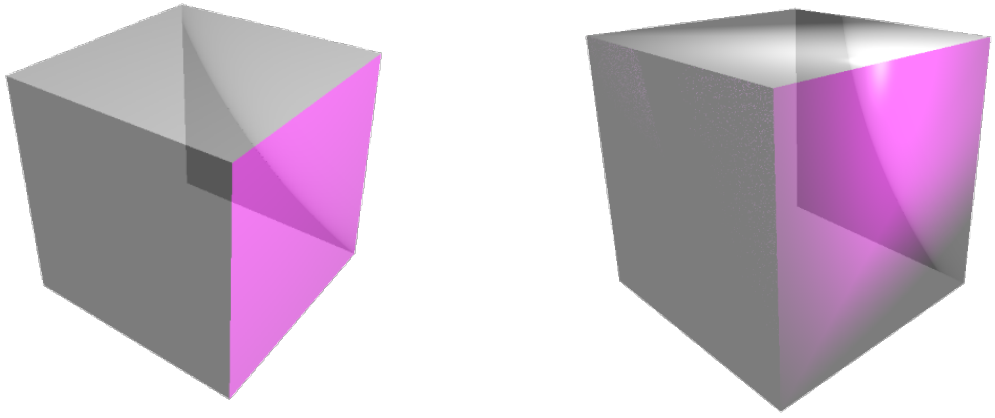
(b) Render of the Scene

Figure 7: The scene created by our group, both the preview and the ray-traced image

2 Possible Extensions for the Implementation

2.1 Utilizing interpolated normals to smooth objects.

Done by: Julian Biesheuvel



(a) Not interpolated normals

(b) Interpolated normals

Figure 8: The effect of interpolated normals.

2.2 Extending the debugger to show the nth reflection of a ray via the keyboard, or triggering a ray highlighting and showing command line output of the selected ray's properties

Done by: Joran Heemskerk

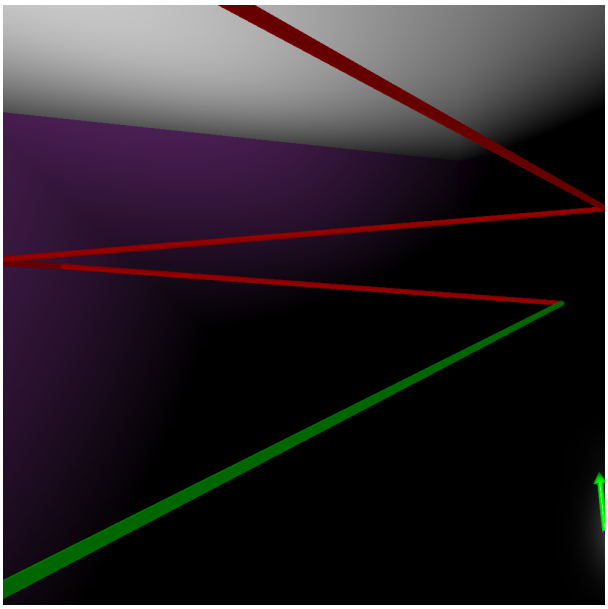


Figure 9: Internal reflections within a cube

Number of reflections can be changed with 1-9 on the keyboard (and 0 for no reflections).

2.3 Showing refraction behavior through dense object

Done by: Paco Pronk, Marilotte Koning, (Joran Heemskerk)

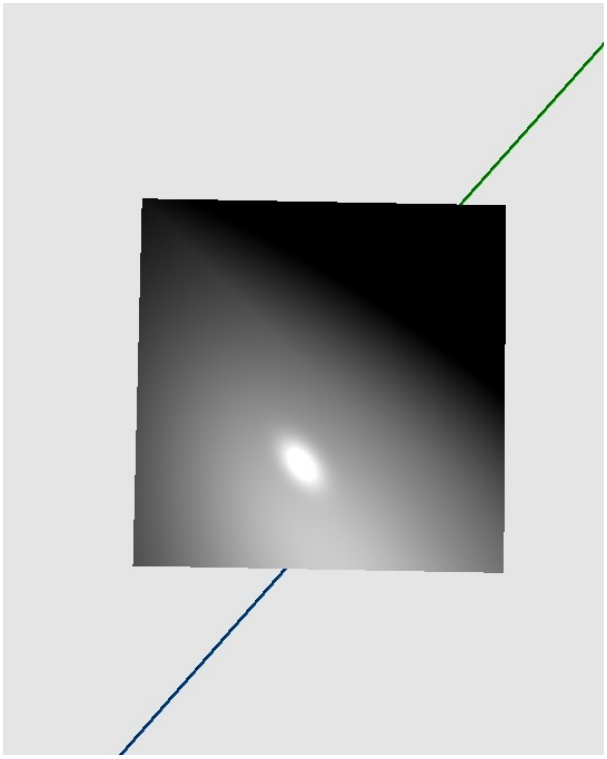


Figure 10: Top-down view showing refraction behavior through dense object

2.4 Allowing modification of triangles within the ray tracer.

Done by: Sina Şen

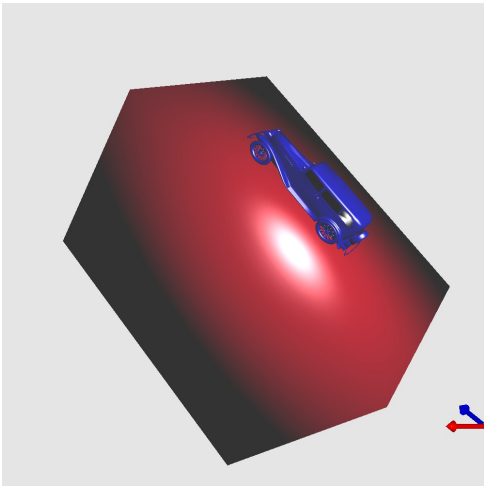


Figure 11: Translation, Rotation and Scaling illustrated in one image

2.5 Supporting refraction and the display of transparent objects.

Done by: Paco Pronk, Marilotte Koning

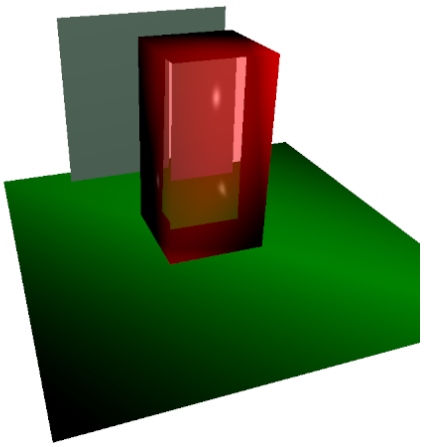
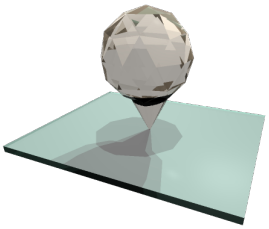


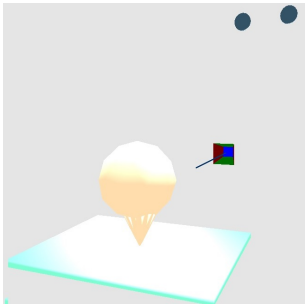
Figure 12: Refraction of a cube

2.6 Supporting soft shadows and other types of light sources.

Done by: Kevin Nanhekhan



(a) soft shadow caused by multiple light sources



(b) previewer with multiple light sources

Figure 13: Soft shadows with different kinds of lights.

2.7 A numerical evaluation of the performance of your ray tracer.

This extension is delivered as an individual file, see submission. Done by: Sina Şen

2.8 Multicore support of the ray tracer.

This extension is code based only, see source files for implementation. Done by: Julian Biesheuvel

3 Sources

- Slides Prof. Dr. E.Eisemann
- Interpolated Normals
  - https://computergraphics.stackexchange.com/questions/5006/do-i-use-barycentric-coordinates-to-interpolate-vertex-normal
  - https://codeplea.com/triangular-interpolation
  - https://courses.cs.washington.edu/courses/cse457/17au/assets/lectures/ray-tracing-1pp.pdf
- Triangle Intersection
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/ray-tracing-rendering-a-triangle/barycentric-coordinates
- Box Intersection
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/minimal-ray-tracer-rendering-simple-shapes/ray-box-intersection
- Sphere Intersection
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/minimal-ray-tracer-rendering-simple-shapes/ray-sphere-intersection
- Multi-Threading
  - https://medium.com/@phostershop/solving-multithreaded-raytracing-issues-with-c-11-7f018ecd76fa
- Shadows (Hard, soft and multiple)
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/introduction-to-shading/ligth-and-shadows
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/introduction-to-shading/shading-spherical-light
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/introduction-to-shading/shading-multiple-lights
- Recursive Ray tracing
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/introduction-to-shading/reflection-refraction-fresnel
  - https://www.scratchapixel.com/lessons/3d-basic-rendering/ray-tracing-overview/light-transport-ray-tracing-whitted
- Numerical Evaluation
  - https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5560912
  - https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8244923
  - http://www.cemyuksel.com/research/papers/rt_performance_CGI18.pdf