Rays-Crashing

Project Report

Etienne Moutot [etienne.moutot@ens-lyon.org]

08/01/2017

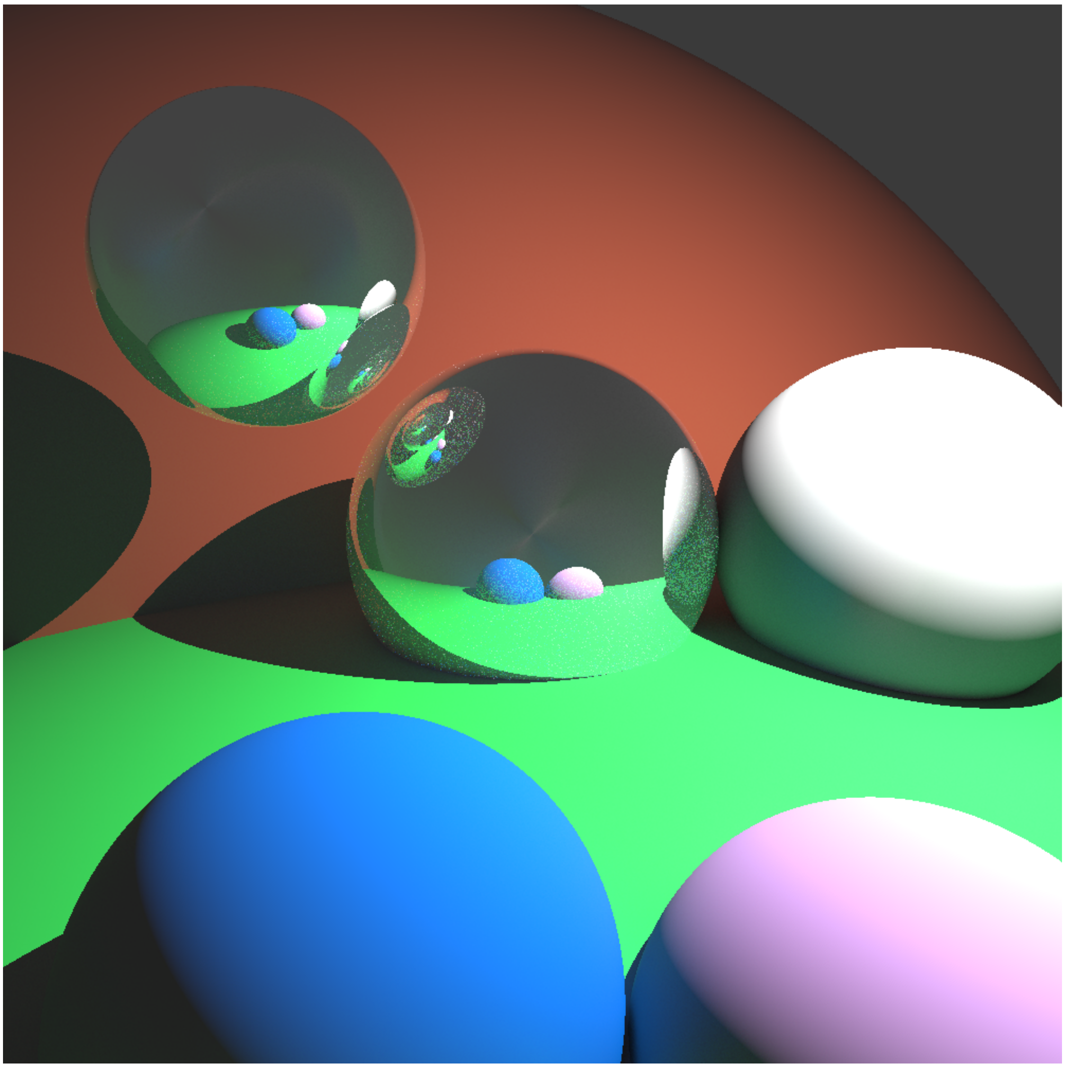

Figure 1: 1000x1000, 500 rays per pixels. Render time: 3min 25sec.

1 Presentation

Rays-crashing is a basic raytracer which is able to render only scenes containing spheres at the moment. It has the following features:

- Direct illumination
- Materials: Diffuse, Specular and Transparent

- Indirect illumination using Monte-Carlo method
- Parallelization
- Gamma correction

2 Using Rays-Crashing

To build the project, you can clone it from github: `https://github.com/coma94/rays-crashing.git` Then build and lunch it:

```
cmake .
make
./rays-crashing
```

The dependencies are only OpenMP.

For the moment the scene is generated in src/main.ccp. You need to rebuild rays-crashing every time you change the scene.

The rendering parameter are also build in the main. To change the size of the output image, change the 1000, 1000 at line 59:

```
Camera cam = Camera(Vector3(0,0,20), 1.05, 1000, 1000);
```

For the filename of the rendered file and the number of rays lunched for the indirect lightning, see line 65:

```
cam.render("bla.png", 100);
```

3 Code Organization

The directory `include/` contains the C++ headers and `src/` the sources files. Several classes are defined and used in this project:

Image (image.h/cpp): C++ layer over CIming, to store images in memory and save them.

Vector3 : 3-dimensional vector with double coordinates, and a lot of operation between them (addition, products, norm, ...).

Material (objects.h/cpp): Contains properties of the material of objects.

Ray : A ray. Contains a method ton compute its intersection with a sphere.

Sphere (objects.h/cpp): A sphere of the scene with a material. Contains a method ton compute its intersection with a ray.

Light (objects.h/cpp): Contains properties of the light of the scene.

Scene (objects.h/cpp): The scene itself, containing objects, a light and a method to compute the value of a point seen by a specific ray.

Camera (objects.h/cpp): The camera, given a scene it renders it into a png file.