Light dispersion and high quality caustics

Modeling and Simulation of Appearance - Final project

César Borja (800675) and Nerea Gallego (801950)

January 11, 2024

1 Objectives

The primary objectives of our project is to generate high quality caustics (Fig. 1) and light dispersion effects (Fig. 2). To do so, we have implemented two new features in Nori:

- A Photon Mapping integrator that favours the generation of caustics.
- Wavelength-based dispersion.

Figure 1: Caustic example.

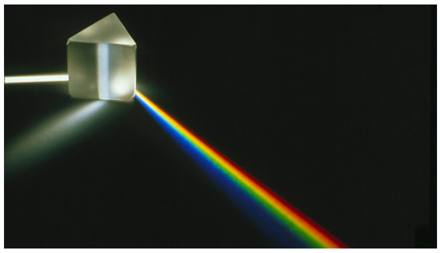

Figure 2: Light dispersion example.

2 Photon Mapping

This section explains the key choices made for our Photon Mapping approach, inspired in [1].

First, we generate and store **two different photon maps** in the first stage of the algorithm:

- **"Global" photon map**: Only sampled photons that intersect diffuse objects are stored. This photon map stores the global illumination, including indirect light interactions.
- **"Caustic" photon map**: Only sampled photons that interact with non-diffuse objects (and are not absorbed) are stored. As a result, this photon map stores all the caustics of the scene.

There are two main reasons for separating the photon maps. One is to optimise sampling. Caustics often require a higher density of photons to accurately capture the intricate light patterns. By dedicating a separate photon map to caustics, we can control the number of photons specifically for these critical areas. The other is efficiency. The global photon map is concerned with capturing indirect light interactions across the scene. The caustics photon map, on the other hand, focuses only on specific surfaces that produce caustics. By having a dedicated caustics photon map, computational resources can be focused on calculating and tracking photons relevant to caustics effects, reducing the overall computational cost.

About the way we store the photon maps, we use a KDTree data structure (one per photon map). We have used the nanoflann [2] API for creating and managing the KDTrees.

3 Wavelength-based dispersion

This section explains the way we have implemented wavelength-based dispersion [3].

First, we add a new field in the BSDFQueryRecord class for representing a wavelength (in nanometers). By setting this wavelength parameter, we indicate that the ray is monochromatic.

Then we created a new type of material, the dispersive material. This material is based on a dielectric material. The main difference is related to the *sample()* method. Instead of having a fixed index of refraction, we calculate the index of refraction based on the wavelength of the incident ray. It is important to note that the calculation of the refractive index is not physically based on any existing material. To make the effect more visible, we have extended the range over which refractive indices are calculated (i.e. refractive indices vary significantly across the visible spectrum).

Finally, when a photon intersects a dispersive material during photon mapping, the new sampled photon resulting from this interaction will be monochromatic. So, before sampling it, we randomly select its wavelength value from among 16 different values (evenly distributed across the visible spectrum). The direction of the new photon will then vary depending on the wavelength value.

4 Results

In this section are some examples of caustics, with and without light dispersion.

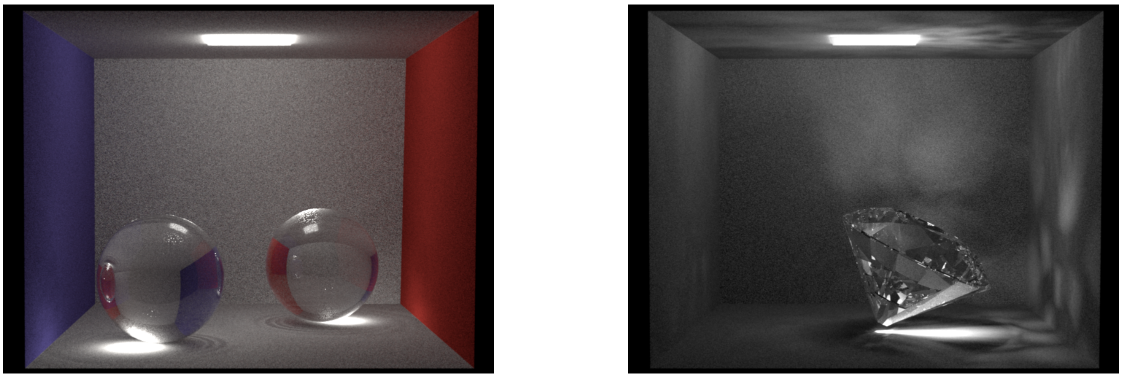

Figure 3: White caustics examples generated with Photon Mapping. No light dispersion

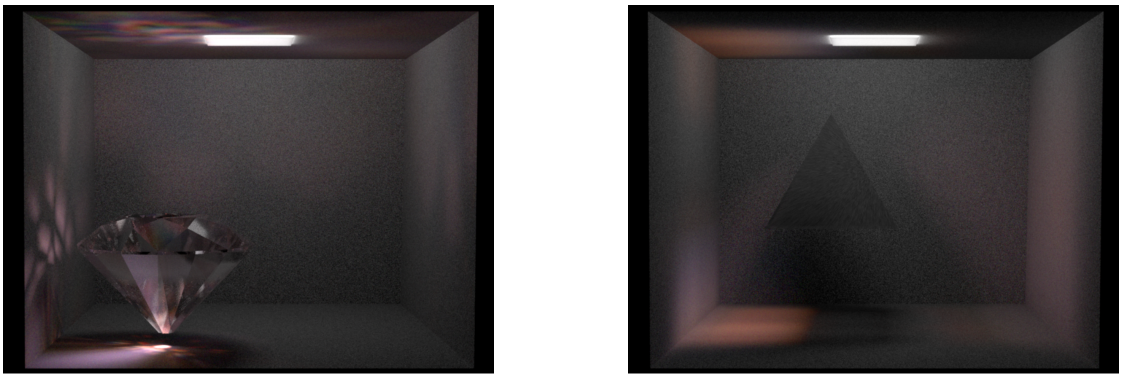

Figure 4: Caustics with light dispersion

References

[1] Henrik Wann Jensen. *Realistic image synthesis using photon mapping.* Vol. 364. Ak Peters Natick, 2001.

[2] *nanoflann.* `https://jlblancoc.github.io/nanoflann/`.

[3] Yinlong Sun, F David Fracchia, and Mark S Drew. "Rendering light dispersion with a composite spectral model". In: *Diamond* 2.37.17 (2000), pp. 0–044.