

The Beauty of Rust

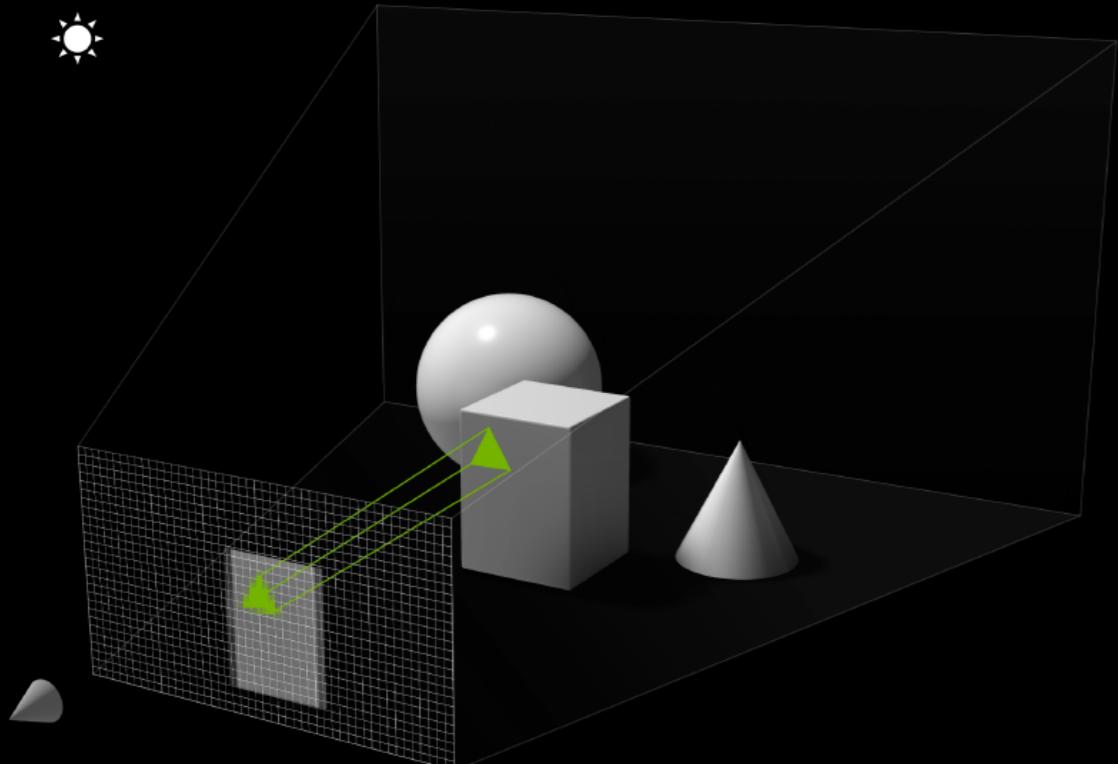
Practical Session: Ray Tracing

Peter Zdankin

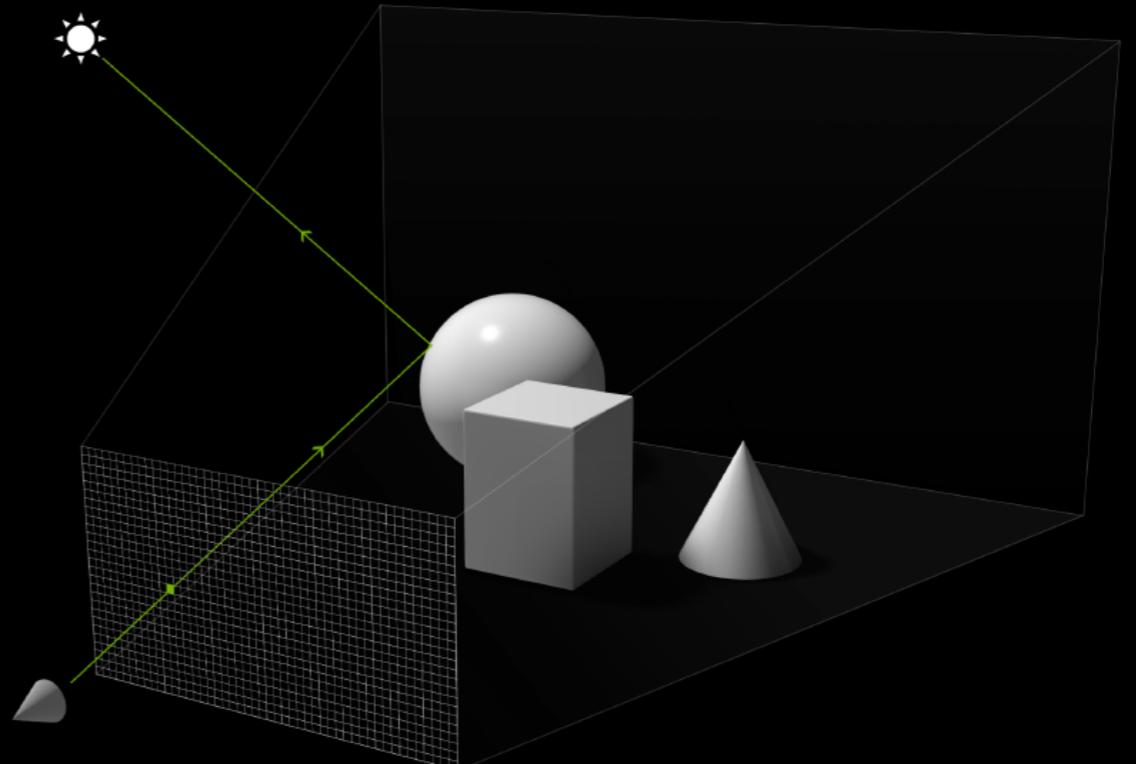
Ray what?

- Ray Tracing is a technique to render photorealistic images
- Simulate the bouncing of light in a scene
- Physically correct rendering (in comparison to rasterization)
- Relatively simple to explain
- Computationally heavy

THE HOLY GRAIL OF GRAPHICS



RASTERIZATION



RAY TRACING

THIS PRESENTATION IS EMBARGOED UNTIL SEPTEMBER 14, 2018

Ray Tracing vs Rasterization (early days)



Precomputed Radiance Transfer (hard)

Left to right: Unshadowed, shadowed, interreflect for 1 bounce , interreflect for 2 bounces





GEFORCE
RTX™

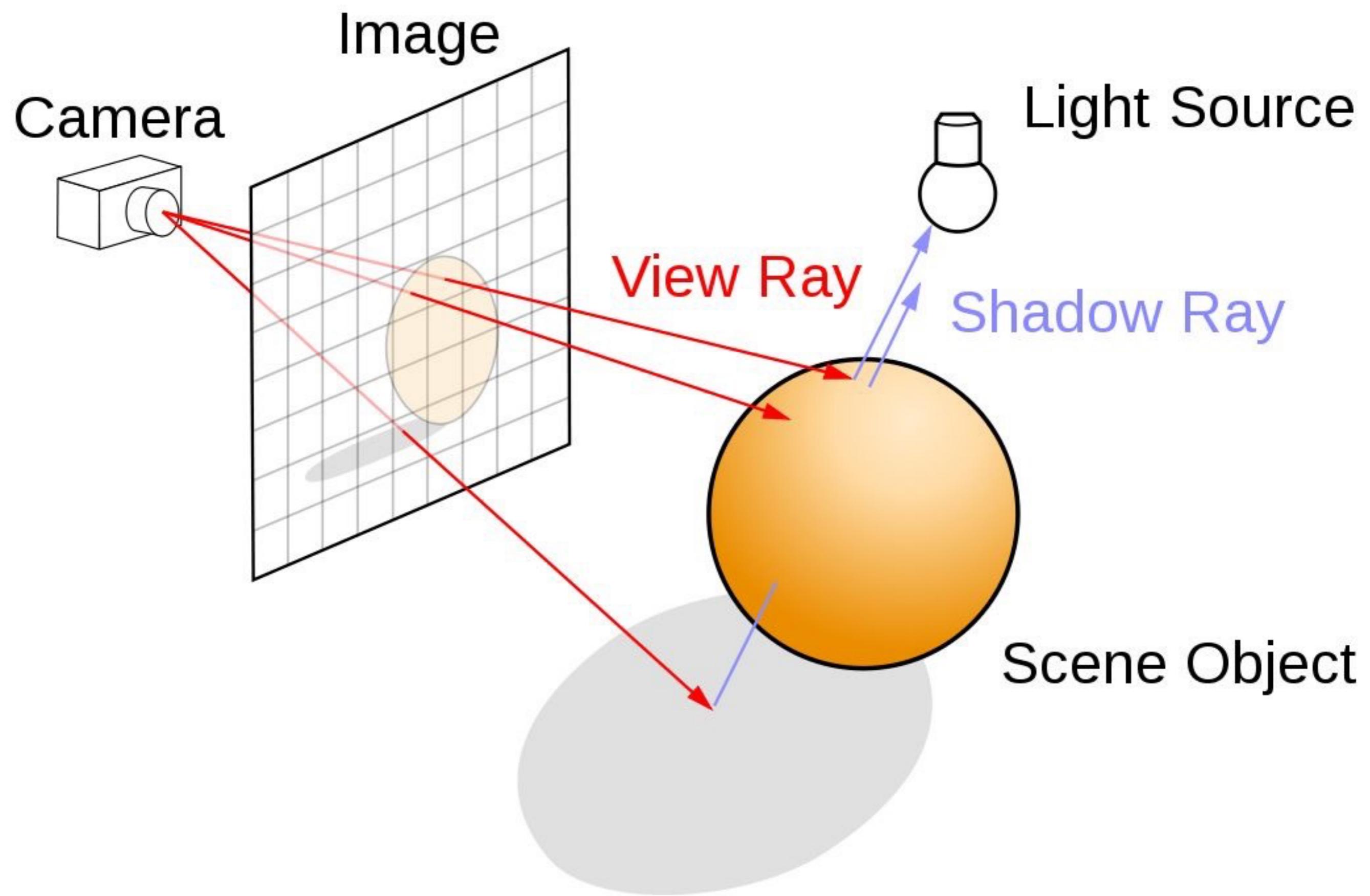
RTX. IT'S ON.

RTX
OFF

RTX
ON

CYBERPUNK
2077





The Ray Tracing Algorithm

- Pick a camera position and viewing direction
- Pick a virtual screen
- Shoot a ray from the camera to each pixel of the virtual screen
- Follow the ray
 - Hit nothing? Background color
 - Hit something? Add color and shoot ray from reflected surface

It's not rocket science!

- You are equipped with almost everything for this task!
- You can write a full ray tracer in one weekend!
- In fact: There is a book for it
- [https://raytracing.github.io/books/
RayTracingInOneWeekend.html](https://raytracing.github.io/books/RayTracingInOneWeekend.html)

Todays agenda

- Starting with the example code I provide for you, we will try to follow the book along!
- Step by step