Rareș Dumitru

Nationality: Romanian **Date of birth:** 14/09/2003 **Gender:** Male

Phone number: (+40) 773345002 **Email address:** dumitru.rares14@gmail.com

LinkedIn: https://www.linkedin.com/in/dumitrurares14/

GitHub: https://github.com/dumitrurares14

Itch: https://rares-dumitru.itch.io/

Website: https://www.rares-dumitru.dev/

Home: Kapelhof 31, 5038 AT Tilburg (Netherlands)

ABOUT ME

A self-motivated student, passionate about game development with a drive towards research. Graphics enthusiast with familiarity in engine and tools development too, I am particularly interested in global illumination techniques, voxel engines, and creative uses of ray marching and ray tracing.

DIGITAL SKILLS

General Programming

C/C++ / C# / JavaScript / Lua / HTML/CSS / Python

Graphics Programming

WebGPU / GLSL/HLSL / Vulkan / OpenCL / OpenGL / Compute

Project collaboration

Scrum / GitHub / Git / Miro / Teams / Unreal Engine / Unity

Industry passions

Voxels / Graphics APIs / Global Illumination / Ray Marching / Ray Tracing / Path Tracing / Procedural Generation / Procedural rendering

LANGUAGE SKILLS

Mother tongue(s): Romanian

Other language(s): English

EDUCATION AND TRAINING

Bachelor

Breda University of Applied Sciences [01/09/2022 – Current]

City: Breda | **Country:** Netherlands | **Website:** https://www.buas.nl/en | **Field(s) of study:** Creative Media and Game Technologies

Certificate of professional competences

"Grigore Antipa" College of Sciences Brașov [2022]

City: Brașov | **Country:** Romania

Attesting :

1. Realization of the design and structure of software products necessary for the implementation of software systems, software applications, databases, web pages, (customer-oriented software)
2. Customization, configuration, and modification of software applications in order to adapt the client's information systems

High School

"Grigore Antipa" College of Sciences Brașov [09/2018 – 05/2022]

City: Brașov | **Country:** Romania | **Website:** https://csantipa.ro/new/ | **Field(s) of study:** Mathematics and Informatics

CONFERENCES AND SEMINARS

[12/11/2024 – 14/11/2024] Breda

Graphics Programming Conference (GPC) 2024

I attended GPC in 2024 and had the opportunity to present how I approached Global Illumination creatively using cellular automata and cone tracing. I studied and developed this technique as a stretch goal during one of my university projects.

Link: https://www.graphicsprogrammingconference.nl/

PORTFOLIO

Visual showcase of all my work

Link: https://dumitrurares14.github.io/

PROJECTS

[09/2024 – 10/2024]

Wexel engine

- Custom voxel engine made as a self-study project with a friend in 32 hours of tracked work.
- Made in WebGPU and Javascript+HTML.
- A chunk of 128x128x128 voxels inside a rasterized box, fully ray-traced shadows, and rendering with basic PBR support.
- Gained experience with WebGPU and WGSL while having fun with voxels.

[05/2024 – 07/2024]

Owlet

- Custom tech engine project that ended up with an RTS game during Block D in Year 2 at BUAS.
- The engine is made in C++ and DirectX12 with hardware accelerated ray traced shadows using DXR.
- Worked on the material system, grass system, particle system, procedural sky and post processing effects tools and editor for all the systems mentioned.
- Gained experience in engine and graphics programming as well as developing tools for designers and artists; my very first interaction with DirectX12 and hardware-accelerated ray tracing. Got comfortable working with SCRUM in a bigger team with colleagues across disciplines.

Link: https://buas.itch.io/owlet

[04/2024 – 04/2024]

Cone tracing cellular automata GI

- Stretch goal explored by me during the OpenGL voxel team project.
- I presented the explored technique at GPC 2024 with a small talk.
- Cellular automata is used for direct light propagation and cone tracing for indirect calculation and direct sampling. Volumetric and specular reflections are done also with cone tracing.
- Gained experience in theoretical research and prototyping, global illumination techniques, cone tracing, cellular automata, and advanced voxel approaches.

[04/2024 – 06/2024]

Voxel engine

- Custom voxel engine project made in a team during Block C in Year 2 at BUAS.
- The engine is made in C++ and OpenGL with additional cross-platform support for PlayStation 5.
- Worked on rendering and tools and editor for procedural generation of caves using an advanced implementation of cellular automata.
- Gained experience in rendering voxels, making tools for voxel engines, and handling engine architecture and memory. Also gained a basic understanding of cross-platform development and more experience working with SCRUM in a team entirely made of programmers.

[01/2024 – 03/2024]

rVox Engine

- Custom voxel engine project made during Block B in Year 2 at BUAS.
- The engine is made in C++ and OpenGL with OpenCL for the rendering/compute pipeline.
- Per voxel lighting with software ray tracing.
- The project was followed up by a short blog post targeted towards enthusiast beginners who want to get into the voxel world.
- Gained experience in voxel engines, rendering with ray marching and lighting solutions.

Link: https://dumitrurares14.github.io/Per-Voxel-Lighting-for-Dummies/

[11/2023 – 01/2024]

Lambda engine

- Custom engine project made during Block A in Year 2 at BUAS.
- The engine is made in C++ with support for scripting in Lua.
- Gained game engine architecture experience with a project that supports serialization, particle systems, resource manager, GLTF loading, entity component systems, complex scene hierarchies

[06/2023 – 08/2023]

Mowdown

- Unreal Engine game made in a team project during Block D in Year 1 at BUAS.
- Worked on the grass-cutting system, other mechanics(respawn, shoot, movement) and developed tools for play testing/design such as a heatmap that shows where players spend most time during a play session.
- Gained experience in Unreal engine, Blueprints, and working in a team following SCRUM methodology.

Link: https://mesibby.itch.io/mowdown

[04/2023 – 06/2023]

2D Ray tracer

- 2D Ray Tracer made during Block C in Year 1 at BUAS.
- The ray tracer is written entirely in C++ and runs only on the CPU.
- This was followed by a quick snake spin-off to showcase the capabilities of the renderer.
- Gained experience in ray tracing techniques.