

Moritz Falk

Öhringerstraße 21, 70435 Stuttgart

Moritz@falkhf.de

Objective

Programmer with experience in graphics programming and engine development. Co-founder of Dihedron Software GmbH. Seeking a role focused on systems programming and performance optimisation.

Projects

Blick Video Editor at Dihedron Software GmbH

2026 — ongoing

- Developing a modern lightweight cross platform NLE with an immediate mode UI in a small team. Working on the core UI and rendering system on macOS and Windows.
- Utilising modern graphics API features to develop a compute shader driven, tile based SDF renderer with many flexible features and a caching system for energy efficiency. Constructed as a shader virtual machine with minimal involvement of CPU side logic.

Developing the ARC project at AnyDesk

2024 — 2026

- Developing a modern immediate mode UI library as part of a team of 4 people. Mainly doing renderer and platform code, but also in Team leading role in the last 4 months.
- An example renderer that is part of the library incorporates a hash grid to only redraw dirty regions by leveraging the fixed function pipeline's stencil buffer and early depth/stencil testing to cull pixel shader invocations. This is especially relevant for UI applications that want to be energy efficient.

Development of a 3D game in a custom engine

2020 – ongoing

- Multiple iterations of a voxel ray tracing based dynamic global illumination system. Includes the voxelisation of 3D meshes on the GPU and the generation of a more cache friendly acceleration grid structure. Additionally, spatial and temporal denoising is applied.
- Simple developer UI using microui for displaying time measurements to check performance, inspecting entities and controlling debug views.
- Stochastic alpha using interleaved gradient noise for simplified integration of transparent elements into the deferred shading pipeline. These elements currently are deferred decals and volumetric billboards. The latter is an unusual method for rendering volume data. The purpose of these elements is to evoke a painterly style.
- A separate tool for generating the volumetric billboard data as a custom file format. A slightly different voxelisation method taking advantage of MSAA is used here.

Arcade Retro Racer for a 10 day Game Jam as part of a team of 4 2023

- Coordinating as an interdisciplinary team under high time pressure.
- Creating a pseudo 3D effect similar to old arcade games like Outrun. Achieved mostly by scaling and moving sprites in a plausible fashion. Correct

placement of sprites and similar “tweaky” tasks were enabled by implementing a custom tweak system. Which, by loading the source code of the game and using a hashtable and a macro, can reload values for certain variables at runtime. A kind of very bare bones in game editor.

Dynamic Diffuse Global Illumination 2022-2023
implementation using D3D11

- First serious involvement with a graphics API and writing shaders. Reading the NVIDIA research paper to guide the implementation. It works but is not very performant, since no acceleration structure is used and some shaders are written naively.
- Multiple disadvantages of this approach to global illumination became noticeable. Such as efficiently placing probes and dealing with dynamic objects.

Software Rasterizer 2022

- Gaining a better understanding of the standard transformation hierarchy in 3D rasterisation. Especially clip space and the canonical viewing volume.
- The basis was “tinyrenderer” (can be found on github), which was then extended by realtime rendering, a dynamic camera and a naive clipping method.

CPU Ray Tracer 2022

- Implementation of a bvh acceleration structure as well as a single server multiple consumer multithreading scheme to achieve “realtime” performance.
- Customizing loading of .obj data and caching the result in a custom file format, since re-parsing the obj everytime for bigger models turned out to be quite slow.
- Removing the CRT out of curiosity and to minimize external dependencies.

Programming languages

C/C++, Odin, hlsl, msl