

Daniel Martinez Amigo

COMPUTER SYSTEMS ENGINEER

Alcala de Henares, Madrid, Spain

☎ (+34) 611684999 / (+44) 7756691924 | ✉ danimtz@outlook.com | 🌐 danimtz.github.io/ | 📷 danimtz | 📺 danimtz15

Education

University of Bath

M.ENG IN COMPUTER SYSTEMS ENGINEERING

- First-Class Honours.

Bath, Uk

Oct. 2016 - Jun. 2020

Engineering Experience

Veist Engine : Vulkan renderer/game engine

Madrid, Spain

GRAPHICS PROGRAMMING PERSONAL PROJECT

2021

- Vulkan based engine written in C++17 with the objective of learning the graphics API.
- Command buffer recording abstraction layer that allows for easy creation and recording of Vulkan commands.
- Physically Based Rendering of GLTF models using GLSL shaders. The shaders can be compiled at runtime using SPIRV-Reflect or loaded from a cache.
- Image Based Lighting based on "Real Shading in Unreal Engine 4" from Siggraph 2013
- Entity component system handles scenes and provides a data driven approach to scene simulation that improves performance.
- Scene editor that allows to load/save scenes and add/edit/remove entities and their components.
- Forward and Deferred Rendering (Forward+ to be implemented soon)
- Framegraph rendering system that enables custom rendering architectures to be easily created and altered, as well as automatically handling descriptor sets and synchronisation of vulkan GPU structures such as barriers.

SRGB : Software Renderer

Madrid, Spain

GRAPHICS PROGRAMMING PERSONAL PROJECT

2020

- Software renderer written in C++ capable of physically based rendering of 3D models with multiple directional lights.
- The dependencies are SDL2 for window management and stbimage for texture loading. Everything else was written from scratch including math utilities such as vector and matrix operations and an .obj file parser. CMake can be used to build the project.
- Programmable vertex and fragment shader graphics pipeline using C++ virtual functions.
- Main shader achieves PBR rendering using Cook Torrance BRDF shader.
- Parallelism with OpenMP is used in multiple stages of the main render pipeline including vertex shader, primitive clipping, the rasterizer and fragment shaders to greatly increase performance.

BBS template matching algorithm and its application to cloud tracking

Bath, UK

UNIVERSITY FINAL YEAR PROJECT

Feb. 2020 - May. 2020

- Digital image processing based project that compared the Best Buddies Similarity (BBS) algorithm to other template matching algorithms when matching templates from sequences of clouds 15 and 30 minutes apart taken from satellite images that had been corrupted by different types and amounts of noise. Overall the BBS algorithm was deemed to be a useful method for cloud tracking, however not enough to de-throne other alternatives in either speed (Cross Correlation Coefficient) or noise performance (Ordinal Measures)

Distributed & Shared memory programming

Bath, UK

UNIVERSITY COURSEWORK

2019

- This coursework involved solving a simple mathematical task of recursively adding together adjacent slots in a 2D array using shared memory and distributed memory to learn the challenges of parallel programming in different memory architectures.
- The shared memory task involved using threads, barriers, semaphores and other structures to achieve the task.
- For distributed memory mainly MPI was used and the program run on the university's high performance cluster computer.

Skills

Programming *Experienced:* C++, C, MATLAB *Familiar:* C, Python, Java, JavaScript

Graphics *Experienced:* Vulkan, GLSL *Familiar:* OpenGL, HLSL

Other software *Familiar:* Unreal Engine, Unity

Developer Tools *Experienced:* Visual studio, Renderdoc, Git, Github

Languages *Native:* Spanish *Fluent:* English