Charles Wang

(301) 272-5740 ⊠ zwcharl@gmail.com ⊠ linkedin.com/in/zwcharl ⊠ charleszw.com ⊠ github.com/aczw

Education

University of Pennsylvania, School of Engineering & Applied Science *Philadelphia, PA*
B.S.E. in Digital Media Design *May 2026*
M.S.E. in Computer Graphics & Game Technology (Accelerated) *May 2026*

- ➔ Relevant undergraduate coursework: Data Structures & Algorithms, Computer Architecture, Linear Algebra
- ➔ Relevant graduate coursework: Interactive Computer Graphics, Procedural Graphics, Offline & Real-time Rendering, Computer Animation, 3D Modeling
- ➔ GPA: 3.69/4.00

Relevant Experience

Penn Engineering *Philadelphia, PA*
CIS 4600 Teaching Assistant *August 2024 — present*

- ➔ Answer questions regarding C++, OpenGL, Qt, and general computer graphics concepts including rasterization, the rendering pipeline, reflection models, coordinate transformations, and mesh data structures
- ➔ Grade homeworks and assist students in office hours on projects like a scene graph implementation, CPU rasterizer, half-edge mesh editor, and a voxel game engine

TikTok *San Jose, CA*
Intelligent Camera Effects Software Engineer Intern *May 2024 — August 2024*

- ➔ Contributed 1 new feature to the C++ SDK that powers TikTok's interactive effects and filters, which combines an in-house graphics rendering engine with generative AI models and object detection algorithms
- ➔ Revitalized a Unity-like effect creation tool used across internal teams after 7+ months of inactivity by writing C++ and CMake to integrate the newest SDK version, fixing 8 critical bugs and eliminating 3 crashes
- ➔ Prototyped an architecture that enables greater interplay between layered effects in the pipeline with JavaScript

Leadership

UPenn Game Research and Development Environment Club (UPGRADE)
Co-President *January 2023 — present*

- ➔ Collaborate with 40+ members to foster an inclusive environment for UPenn students interested in making games
- ➔ Initiated and lead the development of 6 Unity game projects including Minigame Collection, Galaxy Gacha, and DuckGooseDuckGoose, made with C# and Maya
- ➔ Organize annual school-wide game jams, PAX East and GDC trips, and host tutorial workshops and guest speakers

Projects

Monte Carlo Path Tracer
Offline physically-based renderer supporting cosine-weighted sampling, BSDF-based sampling, direct light sampling, MIS, and environment map lighting. Renders dielectric materials and Trowbridge-Reitz microfacet surfaces.

Mini Minecraft
Voxel game engine made with C++ and OpenGL in a team of three. Uses Qt for window and context creation. My contributions are terrain chunking, efficient rendering (block face culling), block texturing, day/night system, skybox, celestial objects (moon, stars, clouds), flood fill lighting, inventory system, GUI, and text rendering.

Mini Maya
OBJ mesh editor, inspired by Autodesk Maya. Internally uses a half-edge data structure to store geometry. Supports mesh operations like Catmull-Clark subdivision, face triangulation, edge splitting, and skinning via custom joint skeletons.

CPU Rasterizer
OBJ renderer written in C++. Uses Qt for context creation. Supports up to 16x antialiasing and Blinn-Phong shading.

Technical Skills & Interests

Languages: C++, OpenGL Shading Language (GLSL), C#, TypeScript/JavaScript, Python, Java, HTML/CSS
Software & Tools: Unity, Qt, Figma, Adobe CC (Illustrator, Photoshop, InDesign), Linux, Git, CMake, React, Node.js
Interests: Open source, UI/UX, making games, going running, bass guitar, making playlists, subway systems