Tony Yiding Tian

Philadelphia, PA | (267)249-1202 | tonytg@seas.upenn.edu | github.com/tonytgrt

EDUCATION

University of Pennsylvania - School of Engineering and Applied Sciences   Philadelphia, PA

B.S.E. in Computer Engineering, M.S.E. in Computer Graphics and Game Technology   *May 2027*

- GPA: 3.74 — Relevant Courses: GPU Programming, Advanced Rendering, Interactive Computer Graphics, Operating Systems Design & Implementation, Data Structure & Algorithm, Embedded Systems, Electrical Circuit

EXPERIENCE

Linux Kernel Policies Research Assistant - PURM Scholar   May 2025 – Present

Learning Directed Operating System (LDOS), Prof. Sebastian Angel   *Philadelphia, PA*

- Developed eBPF-based kernel monitoring infrastructure collecting real-time TCP networking metrics retrieved from 5 crucial kernel tcp functions of tcp_v4_rcv, v4_connect, state_process, congestion_control, and cubic
- Engineered high-performance data analysis pipeline processing 10,000+ TCP state transitions per second, identifying critical performance bottlenecks in kernel networking policies and congestion control algorithms
- Architected and contributed 2000+ lines of C and Python code to open-source KernMLOps repository, implementing kernel probing infrastructure used by 15+ researchers

Beta Test Engineer & QA Analyst   Dec 2022 – Oct 2023

miHoYo - Genshin Impact (AAA Mobile Gaming)   *Remote*

- Selected as exclusive beta tester for Genshin Impact, a $4B+ revenue mobile game with 60M+ monthly active users, participating in pre-release testing cycles every 6 weeks
- Conducted comprehensive quality assurance testing for 10+ character releases, each generating $1M+ in revenue, ensuring gameplay balance and identifying critical performance issues before public launch
- Performed systematic testing of open-world gameplay mechanics including combat systems, puzzle design, and performance optimization across multiple mobile platforms (iOS, Windows)
- Provided detailed feedback on character design, combat mechanics, and user experience, influencing design decisions for characters with combined revenue exceeding $10M

PROJECTS

Monte Carlo Path Tracer & Real-Time PBR Renderer | *C++, GLSL*   Jan 2025 – May 2025

- Implemented Monte Carlo path tracer supporting Cornell Box, glass refraction, microfacet materials, and environment mapping with direct light sampling and multiple importance sampling shaders
- Developed physically-based rendering (PBR) shader pipeline with albedo/metallic/roughness maps achieving real-time ray tracing on modern GPUs with 60+ FPS real-time performance
- Engineered custom BRDF models for materials including chrome, plastic, and complex surface properties
- Rendering demo reel: https://github.com/tonytgrt/TonyTianRenderDemo

PennOS - UNIX-like Operating System | *C, Make, GDB*   Mar 2025 – May 2025

- Architected and implemented a complete user-level operating system in C with team of 4, featuring 8000+ lines of systems code with full process lifecycle management
- Designed Process Control Block (PCB) data structure managing 50+ concurrent processes with metadata including PID allocation, priority levels, parent-child relationships, signal handling, and user/kernel stack management
- Implemented preemptive multi-level priority scheduler supporting 3 priority levels with Round Robin time-slicing (10ms quantum), preventing starvation through priority aging and achieving 95% CPU utilization
- Built POSIX-compliant interactive shell supporting 15+ built-in commands (ps, kill, jobs, fg/bg), I/O redirection, pipeline chaining, and batch script execution with robust error handling

Mini Minecraft - Voxel-based 3D Game | *C++, OpenGL, Qt*   Oct 2024 – Dec 2024

- Collaborated in team of 3 to develop fully-featured voxel game engine in C++ using OpenGL in Qt
- Engineered procedural terrain generation system using layered 2D/3D Perlin noise algorithms, creating 5 distinct biomes (Grassland, Mountain, Desert, Islands, Caves) with biome-specific block distributions and procedurally placed vegetation assets
- Developed dual physics simulation system: gravity-based collision detection with terrain for ground movement, and buoyancy calculations for water/lava interaction, plus creative fly-mode with 6-DOF movement
- Project demo showcasing all features: https://youtu.be/jRb4EHV5KQI

TECHNICAL SKILLS

Programming Languages: C/C++, CUDA, GLSL, Python, Java, eBPF, Assembly, JavaScript, LATEX
Graphics & Rendering: OpenGL, DirectX, Real-time Ray Tracing, Path Tracing, PBR, Procedural Generation
Systems & Tools: Linux Kernel Development, Docker, Qt, Git, CMake, Embedded Systems, TCP/IP
Frameworks & APIs: Qt, Raspberry Pi, ATMega, VMware, CloudLab, Jupyter Notebook