

Sheng Chen

shengchen2028@gmail.com | github.com/EMEEEEEMMMM | emeeemmmm.github.io

EDUCATION

Shanghai World Foreign Language Academy (WFLA)

International Baccalaureate Diploma Programme (IBDP) — Honor Class

Shanghai, China

Expected Grad: May 2028

- Selected for Honors Program

RESEARCH EXPERIENCE

Shanghai Jiao Tong University, School of Life Sciences and Biotechnology

Research Intern — Advanced Microscopy Lab (Prof. Boyang Qin)

Shanghai, China

Jul 2026 – Aug 2026

- Selected for summer research internship in Prof. Boyang Qin's laboratory, working on automation systems for high-throughput microscopy imaging
- Designed and implemented automated sample loading workflows for advanced microscopy platforms, integrating hardware control with imaging pipelines to streamline experimental throughput

COMPETITIVE PROGRAMMING

USA Computing Olympiad (USACO)

Bronze Division — Perfect Score (1000/1000)

Feb 2026

Advancing to Silver

- Achieved full marks in Bronze division during 10th grade, demonstrating strong proficiency in algorithmic problem-solving and data structures. Currently preparing for Silver division examination.

TECHNICAL PROJECTS

SPH Fluid Simulator | C++17, OpenGL 3.3, ImGui, OpenMP, CMake

May 2026 – Present

- Developed a from-scratch fluid simulation engine implementing both DFSPH and IISPH pressure solvers. Benchmarked DFSPH against IISPH, achieving 3-5 iterations for convergence compared to 20-30, demonstrating superior efficiency for large-scale simulations.
- Engineered a high-performance parallel pipeline using SoA spatial hashing with atomic operations and OpenMP multi-threading, achieving real-time performance at 33k particles in approximately 14ms per step on desktop CPU.
- Reconstructed water surfaces via Marching Cubes with custom Fresnel shaders for realistic rendering; built an ImGui control panel and integrated FFmpeg headless streaming for offline video encoding.

AtlasPhys: 3D Rigid Body Physics Engine | Python, OpenGL, PyQt6, NumPy, C++, ImGui Aug 2025 – Present

- Developed research-oriented physics simulation engine from scratch, implementing core algorithms for rigid body dynamics, collision detection, and constraint solving
- Implemented GJK (Gilbert-Johnson-Keerthi) and EPA (Expanding Polytope Algorithm) for convex shape collision detection in the python version, implemented SAT (Separating Axis Theorem) and XPBD (Extended PBD) in the c++ version, achieving robust handling of the collision of polyhedrons
- Designed impulse-based contact resolution with iterative solver and Baumgarte stabilization for realistic physics
- Built interactive 3D visualization environment with real-time rendering of physical forces, demonstrating strong integration of mathematical modeling and software engineering

Technical Blog & Research Notes | Hugo, Markdown, GitHub Pages, LaTeX

2025 – Present

- Maintain a technical blog documenting the design, mathematics, and implementation of my physics engine and related learning projects.
- Published articles on rigid-body dynamics, GJK/EPA collision detection, impulse-based contact resolution, numerical integration, quaternions, and linear algebra for physics simulation.
- Currently developing a technical reading series on the original AlphaGo paper, focusing on its algorithms, mathematical foundations, and implementation ideas.
- Integrates mathematical derivations, implementation notes, diagrams, and engineering reflections to document the learning process.

PyTorch Deep Learning Projects | PyTorch, CNN, Computer Vision

June 2025 – July 2025

- Implemented Chinese digit recognition network (70% accuracy after 100 epochs on 12K training samples)
- Developed facial emotion recognition CNN with batch normalization and dropout (60% accuracy on AffectNet)
- Practiced model training pipeline including data preprocessing, hyperparameter tuning, and TensorBoard logging

TECHNICAL SKILLS

Programming Languages: C++17, Python, JavaScript, SQL, LaTeX

Frameworks & Libraries: PyTorch, OpenGL, ImGui, NumPy, CMake, OpenMP

Tools & Technologies: Git, Linux, CUDA (learning), TensorBoard, FFmpeg, Hugo

Core Competencies: Physically-Based Simulation (SPH/DFSPH/IISPH, Rigid Body Dynamics), Computational Geometry (GJK/EPA, SAT, Marching Cubes), Collision Detection & Response, High-Performance Computing (OpenMP, SoA, Parallel Hashing), Deep Learning (CNNs, Computer Vision), Numerical Methods (PDEs, Integration, Linear Algebra), Algorithm Design & Optimization, 3D Graphics & Rendering