Xinle (Eric) Song

131-2227-3608 | erics311@ucla.edu | linkedin.com/in/xinle-song | github.com/EricSongXinLe

Education

University of California, Los Angeles — Los Angeles, CA
Bachelor of Science in Computer Engineering — *Sep. 2023 - June 2027 (Expected)*

- GPA: 3.9/4.0, Dean's Honors List, IEEE-HKN (ECE Honor Society) Board Officer
- Selected Coursework: Parallel Computing, Advanced Computer Architecture, Operating Systems, Data Structures and Algorithms, Digital Design
- John DeGroff Haller Memorial Scholarship (2025), John Richard Leffler Scholarship (2023)

Technical Skills

Languages: C/C++, Python, RISC-V/x86 Assembly, System Verilog, JavaScript
Parallel & Systems Programming:CUDA, OpenMP, MPI, SIMD (AVX2/AVX512), performance profiling
Tools & Frameworks: Linux, Git, GDB, GCC, NVCC, Nsight Compute, perf, Docker, vLLM

Experience

Tencent — Aug. 2025 – Sep. 2025
Software Engineer Intern — *Shanghai, China*

- Adapted and deployed **Shennong Agentic LLMs** on **NVIDIA H20/A10 GPUs**, completing accuracy validation and performance benchmarking for around 10 models.
- Developed a reusable **Python automation script** for end-to-end testing (data prep, model loading, output collection, report generation), reducing runtime from **7–8 hours to 30 minutes**.
- Developed vLLM modules for **weight decryption** and **license authentication**, ensuring correct model decryption and enforcing license checks for secure delivery.

UCLA ORCAS Lab — Jan. 2025 – Present
Undergraduate Research Assistant — *Los Angeles, CA*

- Extended **Tensor Core** functional and performance simulation on the RISC-V **Vortex GPGPU** simulator, adding 2:4 structured sparsity support.
- Implemented **C++ simulation** models for redesigned **Micro-Op** units, skipping **50%** of the FMA operations.
- Designed compact loaders for **Sparse Matrix A** (values and masks), **doubling** the throughput.
- Built tests to verify correctness by computing a reference **D matrix**, optionally enabled with a define flag.

Projects

Parallel Matrix Multiplication Algorithm | *C++, OpenMP, AVX2/512, perf, Linux* — Apr. 2025 – Jun. 2025

- Optimized **GEMM** for large matrices (4096*4096) using **OpenMP** and **SIMD** on x86 and ARM platforms.
- Achieved up to **633x speedup** by applying **loop reordering, tiling**, and AVX-based **FMA** operations.
- Identified cache miss bottlenecks using **perf**, tuned tiling for **L2** reuse, and enabled **AVX512** for GEMM.
- Reached **133 GFLOPS** on AWS Xeon CPU and **300 GFLOPS** on M1 MacBook, showcasing scalability.

Pipelined RISC-V CPU Simulator | *C++, Processor Design, GNU Make, Git* — Jan. 2025 – Mar. 2025

- Built a **5-stage RISC-V Out-of-Order** CPU simulator supporting branch prediction and OoO execution.
- Implemented a **GSharePlus** predictor combining global/local history with dynamic selection to boost accuracy.
- Implemented **RAT** to eliminate data hazards and optimized **RS, ROB, and CDB** for efficient scheduling.

Brewin Interpreter | *Python, Abstract Syntax Tree (AST), Static Typing, Git* — Oct. 2024 – Dec. 2024

- Developed an interpreter for **Brewin**, supporting **static typing**, **user-defined structs**, and **coercion**.
- Added **nested structs**, **default returns**, **value/object reference** passing, and **exception handling.**
- Implemented **static scoping** and **function calls**, optimizing variable lookup with deque and dictionary (**+30%**).
- Achieved **100%** unit test pass rate by verifying all edge cases against language specifications.