

Dimitri Chrysafis

dimitri.chrysafis@gmail.com • 650-305-9037
🔗 [DimitriChrysafis](#) • [dimitrichrysafis.github.io](#)

Education

University of Wisconsin–Madison
Bachelor of Science in Computer Science

Graduating May 2027
GPA: 4.0

Open Source Contributions

[Simplex Chat](#)

12 Approved Pull Requests

- Optimized X3DH key exchange protocol implementation, reducing handshake latency by **35%** through parallel ECDH computations and precomputed ephemeral key pools
- Fixed memory leaks in Core Data sync operations for macOS Ventura, reducing app memory footprint by **10%** and improving message sync reliability across devices

[Taichi](#)

8 Approved Pull Requests

- Redesigned Apple Metal backend implementing buffer–image copy routines at GPU command list level
- Enabled direct GPU memory transfers between buffers and textures on macOS, removing CPU-side staging overhead

Projects

Custom LLM Fine-Tuning for Automated Code Review [\[GitHub\]](#)

- Fine-tuned CodeLlama-7B on 25K code review comments from open-source projects, achieving **84% precision** in identifying bugs and code quality issues, outperforming GPT-3.5 baseline by 18% on custom evaluation dataset
- Implemented efficient training pipeline using 4-bit quantization and LoRA (rank=16), reducing memory usage from 28GB to 8GB while maintaining model quality, deployed on M4 Max MacBook Pro achieving **sub-400ms** inference for 500-token reviews

High-Performance Web-Based Fractal Renderer [\[Blog\]](#)

- Engineered real-time fractal visualization system supporting $10^{15} \times$ magnification for Mandelbrot, Newton's, and Kleinian limit sets using GPU-accelerated WebGL fragment shaders with double-precision emulation
- Achieved **60× performance improvement** through adaptive sampling, tile-based rendering, and optimized complex arithmetic reducing computational overhead by 85%

3D Real-Time Fluid Dynamics Simulator [\[Demo\]](#) [\[GitHub\]](#)

- Developed complete fluid dynamics engine implementing **incompressible Navier-Stokes equations** with semi-Lagrangian advection, supporting 2 million voxels (128^3 grid) at 30+ FPS
- Accelerated solver through Jacobi iteration, marching cubes, and vorticity confinement, reducing computation time by **40%**

Advanced Sphere Packing Optimization Engine [\[Blog\]](#) [\[GitHub\]](#)

- Tackled NP-hard 3D sphere packing problem achieving **74% packing density** for complex geometries with 10,000+ spheres using simulated annealing with adaptive cooling schedules and Metropolis-Hastings acceptance criteria
- Improved performance by **50%** through CPU/GPU unified memory architecture, ray-sphere intersection algorithms, and spatial hashing reducing complexity from $O(n^2)$ to $O(n)$

Skills and Interests

Languages: C++, Python, Swift, Rust, Go, Java, JavaScript/TypeScript, SQL, LaTeX, CUDA, OpenCL, Bash, Git

Tools: PyTorch, TensorFlow, JAX, NumPy, scikit-learn, Metal, Docker, Kubernetes, Hugging Face, OpenCV

Interests: Competitive marathon runner and triathlete; Calisthenics, Squash