XI HAN

39 Montclair Dr, Selden, New York 11784, United States
(+1) 631-710-8313 | xihan1@cs.stonybrook.edu | https://axihixa.github.io/

EDUCATION

Department of Computer Science, Stony Brook University, New York, United States Aug 2019 – Present
Ph.D. in Computer Science (In progress, expected by Spring 2026) | GPA: 3.9/4.0

Department of Computer Science and Technology, Tsinghua University, Beijing, China Aug 2015 – Jul 2019
B.E. in Computer Science and Technology | GPA: 3.25/4.0

PUBLICATIONS

➢ Jingwei Zhang*, Anh Tien Nguyen*, **Xi Han***, Vincent Quoc-Huy Trinh, Hong Qin, Dimitris Samaras, and Mahdi S. Hosseini, "2DMamba: Efficient State Space Model for Image Representation with Applications on Giga-Pixel Whole Slide Image Classification", In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. (*: **Equal Contribution**)

➢ **Xi Han**, Fei Hou and Hong Qin, "UGrid: An Efficient-And-Rigorous Neural Multigrid Solver for Linear PDEs", In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, 2024.

➢ Song-Hai Zhang, Ruilong Li, Xin Dong, Paul Rosin, Zixi Cai, **Xi Han**, Dingcheng Yang, Hao-Zhi Huang and Shi-Min Hu, "Pose2Seg: Detection Free Human Instance Segmentation", In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

WORK EXPERIENCE

Computer Graphics Lab

Stony Brook University, New York, United States | Research Assistant & Teaching Assistant Aug 2019 – Present
Advisor: Hong Qin, Professor at Department of Computer Science, Stony Brook University

➢ Conducted research in computer graphics (intelligent physics-based modeling). Involved concepts: Differentiable PDE-based vector graphics, data-driven neural PDE solvers, etc. Implemented multiple advanced research projects related to graphics and numerical analysis (Differentiable PDE solvers with customized CUDA operators).

➢ Cooperates with Computer Vision lab on training/inference efficiency optimization for AI models. Involved techniques: CUDA kernel fusing, performance profiling, and customized cache-friendly differentiable AI operators such as differentiable Monte-Carlo integrator, fused GEMM, 2D mamba scanner, etc.

➢ Hosted lectures on OpenGL programming with C++/Python, the implementation details of computer graphics applications and algorithms, and the state-of-the-art research topics on graphics and physics-based modeling.

Computer Graphics and Animation Lab

University of Texas at Dallas, Texas, United States | Research Assistant Sep 2018 – Nov 2018
Advisor: Xiaohu Guo, Professor at Department of Computer Science, University of Texas at Dallas

➢ Worked on the 3D face reconstruction project with a local Samsung research lab. Also constructed a human face model dataset for further research purposes.

➢ Configured a Linux workstation for deep learning purposes from zero and deployed neural network models on it.

Graphics and Geometric Computing Group

Tsinghua University, Beijing, China | Research Assistant Jan 2017 – Jul 2019
Advisor: Song-Hai Zhang, Professor at Department of Computer Science and Technology, Tsinghua University

➢ Deployed a MobileNet module on IOS platform with Apple's CoreML framework, and delivered an IOS app for a human segmentation (in Swift and Objective C++).

➢ Optimized the model used in the app (increased accuracy and added key point recognition) and achieved 10x speedup in FPS.

SKILLS

➢ Numerical analysis, high-performance computing, computer graphics, machine learning, and Linux system skills.

□ Expertise in computer graphics and numerical analysis: Neural PDE solvers, customized CUDA-level operators with back-propagation capability.

□ Expertise in AI/HPC: Customized AI operators, AI model training/inference efficiency optimization. Involved topics: PyTorch C++/CUDA extensions, kernel profiling, fine-tuning, operator fusing, cache optimization, etc.

□ Expertise in programming languages: C/C++ (OOP, STL, Metaprogramming and Concurrency), CUDA (including PTX) and Python.

□ Expertise in tools: PyTorch Profiler, CUDA-GDB, Nsight Compute and NVIDIA Compute Sanitizer.

    - Expertise in frameworks: PyTorch, OpenGL and Qt.
    - Other proficiencies: Bash, CMake, Assembly, MATLAB, Java, Objective C/C++ and Swift.
- Language Proficiencies:
    - Chinese (Mandarin) (Native speaker);
    - English (Proficient for working scenarios. TOEFL: 106/120; GRE: 324/340 + Writing 3.5);
    - Japanese (Sufficient for basic working scenarios. JLPT: N1 173/180, N2 169/180).

SELECTED PROJECTS

UGrid: An Efficient-And-Rigorous Neural Multigrid Solver for Linear PDEs

- TL;DR: UGrid is a neural solver for Partial Differential Equations (PDEs) with convergence guarantee.
- Built upon the combination of the U-Net architecture and the legacy MultiGrid PDE solver, provides users with high speed (up to 20x speedup against legacy solvers), high precision (relative residual as low as 1e-5), high robustness (against irregular and noisy input), high generalization power (to irregular boundary geometries and topology), and high scalability (without need for retraining).
- Involved techniques: Numerical analysis on convergence, customized AI operators (Python and CUDA based). Implements a customized CUDA convolution module to save computation for specific-shaped convolution kernels used in PDE solvers.
- Code available at https://github.com/AXIHIXA/UGrid.

2D-Mamba: Hardware-aware 2D Parallel Mamba Scanner

- TL;DR: 2D-Mamba scanner extends 1D Mamba into 2D while maintaining its modeling capabilities, high parallelism, and memory access efficiency.
- Implements a warp-scan based 2D parallel scan routine which supports scanning prefix callbacks for global tiling. Extends 1D Mamba scanning operation into 2D while maintaining its training/inference efficiency. Compared to a naïve implementation, achieves a throughput of 10x, while the GPU memory consumption is only 10%.
- Involved techniques: Warp shuffle and parallel scans, 2D tiling and caching, HBM access optimization, CUDA kernel and AI model profiling, and PyTorch CUDA extension encapsulation.
- Code available at https://github.com/AtlasAnalyticsLab/2DMamba (CUDA extension part).

CUDA Baseline Experiments

- TL;DR: Implements and fine-tunes multiple CUDA baseline algorithms.
- Implemented baselines and their optimizations:
    - Parallel reduction (with loop unrolling and warp shuffle primitives);
    - Histogram and Copy-If (with atomic primitives);
    - Parallel scan (WarpScan and Raking variants);
    - Fused Biased-Mask-Scale-Add (fp32 and fp16, for fp16, with half precision primitives like __hadd2);
    - SGEMM an GEMV (loop unrolling, SMEM padding, warp tiling and double buffer optimizations, reaching 90% throughput of cuBLAS);
    - Dropout (with cuRAND APIs);
    - Fused SoftMax/LayerNorm/RMSNorm, Im2Col, Matrix transpose, etc.
- Code available at https://github.com/AXIHIXA/CudaDemo.