# **Ronghan Che**

## **EDUCATION**

M.S. in Computer Science (Advisor: Prof. Robert J. Walls) GPA: 3.9 / 4.0 Worcester Polytechnic Institute **B.S. in Computer Science Minor in Mathematics** Virginia Tech

## SKILLS

- Languages & Frameworks: C, C++, CUDA, Python, PyTorch, OpenGL, MIPS, Node.js, React.js >
- Tools & Platforms: NVIDIA Nsight Systems, PostgreSOL, CMake, Git, LaTeX >
- Domains: High-Performance Computing (HPC), Deep Learning, Computer Graphics, Compiler Construction >

### **PROJECTS**

### Master's Thesis - RevDecode-CUDA: Accelerated Malware Detection

- Implemented a CUDA-accelerated sequence-decoding algorithm for malware detection in binary function libraries, transforming binaries into layered graphs for Viterbi traversal.
- Developed three optimized GPU kernels (Naive; Segment-Based Estimation; Fine-Grained), cutting runtime from >~16 min on CPU to ~10 s on GPU (~100× speedup).
- Incorporating committee feedback into final manuscript for USENIX Security Symposium 2025.

#### KernelNet - Modular Deep Learning Framework

- Designed and implemented from the ground up a C++/CUDA tensor engine with automatic differentiation- $\geq$ exposed via a C++ API supporting element-wise operations, broadcasting, and matrix multiplications.
- Built core neural modules—Dense (Linear), Activation functions (ReLU, Sigmoid, Tanh, Softmax), MaxPool,  $\geq$ Conv2D, LSTM, and Embedding—and an SGD optimizer with gradient management.
- Built CIFAR-10 and Penn Treebank pipelines (loading, batching, preprocessing) with integrated benchmarking >tools to assess custom CNN/RNN performance and computational efficiency.

### LightSim3D - Real-Time GPU Path-Tracing Engine

- Architected from scratch a C++/CUDA renderer with OpenGL integration for interactive visualization, including >dynamic camera and object controls.
- Built a Bounding Volume Hierarchy using Axis-Aligned Bounding Boxes and centroid-based splitting, cutting >ray-scene intersection from O(n) to  $\sim O(\log n)$ .
- Implemented a PBR Monte Carlo path tracer with tunable diffuse/metal materials, UV-mapped HDR environment >lighting, next-event estimation, cosine-weighted hemisphere sampling, Fresnel reflectance, and Russian roulette for realistic global illumination, soft shadows, and reflections.

#### **MiniJava** Compiler

- Implemented a modular compilation pipeline: regex-based lexer with line/column error reporting, LL(1) parser generating an AST, and semantic analyzer enforcing scope and type rules.
- Designed a recursive AST-traversing code generator that emits MIPS assembly (.data/.text, labels, control flow, >entry) while normalizing identifiers, validating calls/array dimensions, and reporting syntax/semantic errors.

## **EXPERIENCE**

### Virginia Tech

Teaching Assistant

- Led weekly office hours for 60+ students in CS 3304: Comparative Languages, covering programming paradigms and runtime behavior in Scala, C++, and Prolog.
- Graded programming assignments—from Scala parser projects to Prolog logic-rule implementations—and managed the course forum to resolve student questions and clarify requirements.

## **China Eastern Airlines**

Software Engineer Intern

- Architected end-to-end ETL pipelines in Pentaho Data Integration, creating Jobs (.kib) and Transformations (.ktr) with control-flow logic for error handling, branching, and variable injection to extract, validate, and normalize flight and passenger data from public APIs into a unified JSON schema for operational reporting.
- Developed responsive React is components for an internal health-monitoring mobile app-integrating with RESTful services and collaborating on API contract design to ensure seamless front-end/back-end communication.

Aug 2021 - Dec 2021

May 2020 - Aug 2020

Dec 2024 Dec 2021