XIAORUI HUANG

Always Fascinated 🖥️

- 👤 Preferred Name: Richard
- 📞 +1 (289) 772–8682
- 💼 xiaorui-richard-huang
- @ hxr.richard@gmail.com
- 📍 Toronto, Canada
- ⭕ Xiaorui-Huang

EXPERIENCE

Low Power AI Machine Learning Engineer
Qualcomm

- 📅 May 2023 — Aug 2023
- 📍 Markham, Canada

- Led efforts on **Neural Architecture Search (NAS)** and model compression within the Edge AI R&D team.
- Developed a NAS framework, leveraging Qualcomm's patented NAS techniques, optimizes a given models on a pre-profiled hardware, built with Pytorch's **torch.fx**
- Achieved **50% reduction** in **model size** and **60% drop in inference latency** without compromising accuracy across benchmark models, while reducing engineering time compared to manually applied NAS.
- Engaged in team-wide discussions on next-generation **eNPU** software stack, focusing on **quantization** and **attention mechanisms**.

`NAS` `Quantization` `Pytorch` `torch.fx` `Model Compression`

- -

RPA Backend Developer
IBM

- 📅 May 2022 — Apr 2023
- 📍 Markham, Canada

- Worked on backend development for IBM's Robotics Process Automation (RPA) platform, written in C# **OOP**.
- Augmented IBM RPA's *WAL* programming language, introducing a reflection feature resembling Java and C#.
- Collaborated with cross-functional teams, achieving a **15%** reduction in customer issues and defects per release.
- Employed **agile methodologies**, showed both independent and collaborative competencies in a hybrid environment.
- Presented solution strategies to RPA's senior architects and product teams.

`C#` `OOP` `Large Monorepo` `Language Design` `Agile`

EDUCATION

University of Toronto 🏛️
Honors BSc. in Computer Science

- 📅 Sep 2019 — Jun 2024

- CSC367 **Parallel Computing** (83%) — **CUDA** Arch & Reduction Algo, Parallel Arch & Algo, threading & **OpenMP**, Distributed Computing w/ **MPI**, Cloud Computing
- CSC317 **Computer Graphics** (97%) — Ray Tracing, Mass Spring Systems, BVH, Meshes, Kinematics, **OpenGL Shaders** in **C++** using **Eigen** and **libigl**
- ECE568 **Computer Security** (83%) — Buffer Overflow & Control Hijacking, **Cache Side-Channel Attacks**, Network Security, Cryptography, Web Security `C` `x86`
- CSC413 **Deep Learning** (96%) — Transformers, CNN, RNN, GAN, VAE, RL, GNN, Model Tuning techniques

`CSC369 OS` `CSC401 NLP` `CSC420 CV` `CSC412 Probabilistic ML`

RESEARCH

Distributed Online 3D Reconstruction
embARC Research Group

- 📅 Jan 2024 — July 2024
- 📍 University of Toronto

- **DISORF** — a **real-time Gaussian Splatting** & **NeRF** framework for online 3D reconstruction and visualization of scenes captured by resource-constrained mobile robots and edge devices.
- Proposed a novel shifted exponential frame sampling method to address the degradation in rendering quality caused by naive image sampling during online training
- Integrates novel techniques such as adaptive initalization to overcome challenges in real-time incremental learning.
- Paper is under review for RA-L and availble on *arXiv* and ⭕ Xiaorui-Huang/DISORF

`3D Gaussian Splatting` `SLAM` `NeRF` `Pytorch`

PROJECTS

CUDA Ray Tracing

- 📅 Nov 2023 ⭕ Xiaorui-Huang/cuda-ray-tracing

- Implemented a **CUDA** ray tracer with **BVH** acceleration structure, with Blinn-Phong shading.
- Achieved **real-time** ray-tracing of **30 FPS** and **2000x Speedup** on RTX3060-Ti compared to CPU.
- Designed framework for scene construction, allowing for rendering of new scenes via config and existing assets.

`CUDA` `C/C++` `Computer Graphics`

- -

Woodoku Learn

- 📅 Jul 2022 ⭕ EdwardHaoranLee/WoodokuLearn

- Replicated the mobile game Woodoku for the terminal using Python, enabling both human and AI gameplay through dedicated environment APIs.
- Employed Q-Learning, a **Reinforcement Learning** approach with Pytorch, targeting top scores on the Woodoku leaderboard.

`RL` `Pytorch` `OOP` `Agile` `CMake`

- -

Doodle Jumps in MIPS Assembly

- 📅 Dec 2021 ⭕ Xiaorui-Huang/doodle-jump

- Created a Minecraft-themed version of the **Doodle Jump** game using **MIPS Assembly**.
- Implemented game logic for player movement, collision detection, and scoring, key controls & graphic design.

`MIPS Assembly` `Game Development` `Emulation`

SKILLS

Programming Languages

`🐍 Python` `C/C++` `C#` `Rust` `Java` `LaTeX` `x86`

- -

Skills, Frameworks & Development Environments

`CUDA` `Pytorch` `OpenGL` `Parallel Algorithms` `MLIR` `Model Compression` `git` `Vim` `WSL` `Docker`