

## EDUCATION

**Masters in Computer Engineering**, University of Illinois at Urbana-Champaign

- Fall 2021 - Spring 2022

**Bachelors in Computer Engineering**, University of Illinois at Urbana-Champaign

- Fall 2018 - Spring 2021, GPA: 3.80/4

## EXPERIENCE

**Nvidia** Santa Clara, CA — *Software Intern*

May 2021 - August 2021

- Worked on GPU System SW team developing the GPU driver at a kernel level
- Implemented task boot and memory initialization sequence across multiple tasks in a real time operating systems
- Created design document and implemented run-time feature check using a register key
- Addressed escalation of privilege risk

**Data Acquisition** Illini Formula Electric — *Subteam Member*

2019, 2021

- Developed automotive software modules for the Illini Formula Electric Vehicle
- Designed a log parser in Python running on Linux (Raspberry Pi) to handle message packets from the Controller Area Network (CAN); logged data about the battery, motors, and suspension; displayed voltage, current, and state-of-health of the battery
- Emulated CAN messages using an Arduino for remote testing and development

## PAST COURSEWORK

**Computer Systems and Programming**

- Programmed a simple Linux OS in C and x86 assembly including file system, segmented memory, interrupts, system calls, scheduling, and user program execution
- Developed software drivers to interact with hardware devices such as a real time clock, keyboard, game controller, and VGA monitor

**Applied Parallel Programming**

- Analyzed common parallel algorithm patterns and how to identify bottlenecks
- Implemented multiple parallel algorithms including reduction, scan, and sparse matrix-vector multiplication
- Developed and optimized parallized CNN algorithm with input channel reduction tree that utilized shared and constant memory

**Computer Organization and Design**

- Discussed the architecture of computing hardware at the instructional set architecture and microarchitectural level
- Designed and implemented pipelined cpu in system verilog for RISC-V instruction set with branch prediction, multiplier, hazard detection, pipelined L1 cache, and 2-way set associative L2 cache with prefetching that performed in top 33 percentile.

## CURRENT COURSEWORK

- Manycore Parallel Algorithms, Computer Architecture, Languages and Compilers

## SKILLS

- C, C++, X86, RISC-V, CUDA, System Verilog, Python, PyTorch, OpenCV, Javascript, SQL

## PROJECTS

**Facial Recognition on Custom Data Set**

- Created python scripts to map a data set of labeled faces into 128-D vector space and trained a machine learning model on the data set to recognize faces through a webcam
- Trained on Scikit Learn's support vector machine(SVC) model, used OpenCV for image processing and a Torch deep learning model from OpenFace to vectorize faces