Jeff Niu [me@jeffniu.com](mailto:me@jeffniu.com)

Waterloo, ON [jeffniu.com](http://jeffniu.com/)

1 (905) 806-8846 [github.com/mogball](https://github.com/mogball)

work experience CoreAVI *Summer 2018*

*Intern, Embedded Graphics Developer*

- Implemented [EGL Compositor](https://www.khronos.org/registry/EGL/extensions/EXT/EGL_EXT_compositor.txt) Extension in ArgusSC OpenGL driver

- Added VxWorks 6.x/7 real-time process and multi-thread support to Argus

- Added support for RS-343A compliant analog video modes

- Ported Argus OpenCL driver to 64-bit Yocto embedded Linux

Yahoo! *Fall 2017*

*Intern, Software Engineer*

- Contributed data visualizations, task scheduling, and SQL/Druid query

to the open-source development of Apache Superset

- Built [a production tool](https://github.com/yahoo/sherlock) for real-time anomaly detection on Druid streams

- Created [ember-localforage](https://www.npmjs.com/package/ember-localforage), an EmberJS Data adapter that persists to browser cache

- Developed a [Bullet sprout](https://github.com/bullet-db/bullet-storm) to query in real-time the Twitter Firehose

teams Team Waterloop – Canada’s Hyperloop Team *Sep 2016*

*Lead, Software to present*

**-** Created [WLib](https://github.com/wloop), a collection of C++ libraries optimized for embedded systems

**-** Designed a **fail-safe**, reliable software infrastructure for the pod based on

a CAN network and distributed hubs

**-** Developed [Wio](https://github.com/wio/wio), a fully-featured build tool and package manager for C/C++

supporting native and embedded environments (AVR/ARM)

UW Nano Robotics Group *Sep 2016*

*Technical Lead, Controls to present*

- Used OpenCV to develop an occupancy grid localization algorithm that

tracks the microbot and nearby objects

- Applied a modified A\* procedure and 2D game physics to create a microbot AI

that can autonomously push an object through a labyrinth

- Main developer of [Minotaur](https://github.com/uwnrg/minotaur-cpp), UWNRG’s controls software built in Qt

projects Cerpent

*A Basic C-language interpreter, like Python but faster*

- Leverages clang’s libraries to generate line-by-line ASTs parsed by Cerpent

- Uses just-in-time compilation for user-defined functions, then added to symbol table

- Supports pre-compiled modules in C++ exposed to interpreter

education University of Waterloo

B.A.Sc. in Mechatronics Engineering

*Expected Spring 2021*

GPA 4.0 (Rank 1, 97%)

skills