

# Andrew Jemin Choi

✉ aj.choi@mail.utoronto.ca    📄 andrewjeminchoi    🌐 ajchoi

## Education

---

### University of Toronto

B.A.Sc. IN ELECTRICAL AND COMPUTER ENGINEERING

CGPA: 3.72 / 4.0

Toronto, ON

Spring 2019 (expected)

## Experience

---

### Amazon (AWS)

SOFTWARE DEVELOPMENT ENGINEER INTERN

Seattle, WA

Summer 2018

- Worked with the Internet of Things (IoT) team to create a one-touch onboarding script for Raspberry Pi's using the AWS Python SDK
- Developing an Android App in Java to automate and verify authentication and authorization of devices using Amazon Cognito and the AWS Mobile SDK

### University of California, Los Angeles (UCLA)

RESEARCH STUDENT

Los Angeles, CA

Summer 2017

- Explored new ways to perform faster inference on Bayesian Networks by compiling networks into hardware and optimizing graph structures
- Developed a Python/C tool that was 8 times faster than the state-of-the-art algorithm in finding marginal probabilities by compiling and optimizing feed-forward arithmetic circuits
- Research Paper published in the 2017 issue of RUCS, under the supervision of Dr. Guy Van den Broeck

### University of Toronto

TEACHING ASSISTANT – INTRODUCTION TO ENGINEERING

Toronto, ON

Fall 2016

- Led weekly tutorials to teach students about computer engineering practices, engineering ethics, and problem solving approaches
- Received an "Outstanding" TA rating from a class of 25 students, with an average rating of: 6.87/7

### Safran

SOFTWARE ENGINEERING INTERN

Peterborough, ON

Summer 2016

- Designed and documented architecture diagrams for ~1000 functions for libraries written in C and Assembly
- Developed a Python program to efficiently parse aviation requirements and cross-check data flow in the code, reducing documentation errors by ~50%
- Received the NSERC Experience Award, valued at \$5625, for undertaking an industrial research project in software engineering

## Projects

---

### Mapping the City of Toronto

C++11

- Led a team of 3 in developing a C++ mapping program on Linux using OpenStreetMap APIs
- Used unit testing and profiling for ~10000 lines of code to isolate bugs and pinpoint performance issues
- Created interactive and responsive graphics for the mapping interface and applied heuristics to optimize graph algorithms and to approximate Travelling Salesman solutions (~50% faster than expected metrics)

### FaceAverage

PYTHON

- Developed a Python app to overlay and average facial pictures using OpenCV's similarity transform
- Used the dlib landmark detector to find common facial features and align them using triangulation