Jerry Wang

(732) 853 – 5281 | jerryhhw23@gmail.com | jerrryw.github.io | github.com/jerrryw

Education

University of Maryland, College Park – MS in Applied Machine Learning Dec 2025
Rutgers University – New Brunswick – BS in Computer Science May 2023

Skills

Languages: C++, C, Python, Java, SQL
Frameworks: OpenCV, TensorFlow, PyTorch, Keras, scikit-learn

Experience

Software QA Engineer Intern, Eslite Corporation – Taipei, Taiwan Oct 2020 – Aug 2021

- Improved searching performance by implementing AWS CloudSearch to customize text relevance ranking and indexing data, resulting in a 15% increase in search performance
- Developed and executed test cases to optimize checkout flow using Python for automated testing, fixed 2 malfunctions regarding shipment information and delivery instructions
- Inspected Eslite e-commerce web page load time and provided possible suggestions to reduce web page load time by 200ms

Software Engineer Intern, Acer – Taipei, Taiwan June 2019 – Aug 2019

- Collaborated with 4 engineers in building an Automated Guided Vehicle with vision guidance using Python, OpenCV, and Microsoft Azure AI Vision
- Trained a classification model with TensorFlow and Microsoft Azure AI Vision, scoring 60% on classification accuracy
- Delivered remote tech support with customer service and collaborated in crafting multiple Microsoft product handbooks

Projects

Lane Detection

- Built a lane detection system in C++ and OpenCV that takes in a video input and process it to identify the lane in a moving vehicle
- Applied Gaussian smoothing and masks to extract information from unstructured image data to perform transformation and image rendering

Register Allocator

- Designed a bottom-up register allocator in C++ that produced an ILOC program using less than k number of registers
- Implemented Chaitin's and list scheduling algorithm to minimize register spilling and maximize functional unit utilization

Pascal Compiler

- Built a syntax-directed translation parser and code generator to generate ILOC code for a Pascal Subset in Flex and Bison
- Implemented the register-register model by transferring scalar variables from virtual to physical registers to maximize opportunities for register allocation