Jue Guo

336-473-1796 | guoj1995@gmail.com | Personal Website | github.com/BobGuo

Objective

Jue Guo is a upcoming PhD student in Computer Science and Engineering at The State University of New York at Buffalo. His current research interest is Neural Architecture Search. His advisor is Dr Wen Dong.

Education

Wake Forest University — Winston-Salem, NC
Bachelor of Science, Computer Science — *August 2015 – May 2019*

The State University of New York at Buffalo — Buffalo, NY
Master of Science, Robotics — *August 2020 – Current*

The State University of New York at Buffalo — Buffalo, NY
PhD, Computer Science — *August 2022 – Current*

Experience

Software Engineer — June 2017 – August 2017
Nanjing Ohappure Tech Co. Ltd — *Nanjing, China*

- Work as an intern and help develop new features for an outdated company website.
- Use WordPress to develop a front end also learn languages like Javascript, CSS, and HTML.
- Best Intern of Summer 2016

Software Engineer — June 2018 – August 2018
Nanjing Ohappure Tech Co. Ltd — *Nanjing, China*

- Intern for the tech team and volunteer for the research team
- Help with project " Fresh Water, Rural China"
- Update the old water quality database to better adjust the company product
- Successfully deliver a demo, which can access the database to select ID and provinces corresponding to the correct water quality

Machine Learning Engineer — June 2019 – August 2020
Zhejiang Society For Mathematical Medicine — *Hangzhou, China*

- National Lab under Zhejiang University
- Worked on Bone age detection using a machine learning approach for Zhejiang No.1 People's Hospital's Orthopaedics Department.

Projects

Pascal-Like Compiler in C++ | *C++, VScode* — September 2017 – December 2017

- This project uses C++ under the hood to read in and compile Pascal-like syntax language. It provides me with extensive practice in common data structures and solving the complex logic of a compiler. Finished the project perfectly, with the comment "the most perfect project in the class" from the instructor.

Course Registration Website | *Javascript, HTML, CSS* — September 2019 – December 2019

- The project aims to learn a new language fast and be able to deliver an up and running project at certain milestones. I am in charge of using PHP to develop a front-end to interact with the database and deliver the user story on time. I learned how to adapt to the user expectations and compromise some in order to deliver a final product.

Simple Parallel Cashier System | *C++, OpenMP, MPI* — September 2019 – December 2019

- With the appearance of more and more powerful hardware, we need to learn and adapt to this trend by using the potential provided by the hardware to us. This project simulates real-life grocery store check-out: Check for discount, promo code, a number of items purchased, and return a total price. Under the hood, it is written in C++ using MPI.

Multi-Human Pose Estimation | *Tensorflow, Python* September 2021 – December 2021

- Multi-person pose estimation aims at localizing 2d key points of an unknown number of people in an image. It has attracted much research interest because of its significance in various real-world applications, such as human behavior understanding, human-computer interaction, and action recognition. This can either be done by first detecting each human and then predicting their key points (top-down) or predicting all key points and then grouping them into separate objects(bottom-up). We are interested in the latter approach and its application of graph clustering.

Technical Skills

Languages: Python, Javascript, CSS, HTML, C++, Java
Libraries: Tensorflow, OpenMP, MPI

Courses

Undergraduate CS Courses

CSC 111A | *Intro CS — General Purpose — Java*

- "Wheels of fortune", "Black Jack", "Connect Four", "Connect Four-GUI"

CSC 112 | *Fundamentals of Computer Science — C++*

- "Pig Latin", " Normalization", " Grade Calculator" and other projects

CSC 165 | *Problem Solving Seminar*

- Problem-solving of competition questions

CSC 211 | *Computer Organization*

- Assembly Language and hardware aspect of computer science

CSC 221 | *Data Structure and Algorithm*

- "3-D array", "Doubly Linked lists", "Binary Search Tree", "Heap", "Hash Table", "Huffman Code", "Page Rank" and "SudokuBoard"

CSC 222 | *Data Structure and Algorithm 2*

- "RSA", "Matrix Multiplication and Facebook Friend Recommendation Algorithm" and "Memory saving on Knapsack Problem"

CSC 231 | *Programming Languages*

- A comparative study of programming language paradigms including imperative languages, functional programming, logic programming, and object-oriented programming. Syntax, semantics, parsing, grammar, and issues of language design are covered.

CSC 241 | *Computer System*

- Learn about how OS manages resources and provides an interface between the hardware and the user. "Simple Shell", "Sleeping Barber"," Conway's Problem"," Unisex Bathroom" and "Scheduling Method"

CSC 331 | *Software Engineering*

- An introduction to software engineering, the engineering discipline concerned with finding and applying solutions to problems encountered in delivering high-quality, large-scale, real-world software systems in a timely and cost-effective manner.

CSC 333 | *Principles of Translators*

- Study of techniques for translating high-level programming languages to a target language. Typical target languages include Java bytecode and assembly language. Topics include lexical analysis, parsing, intermediate representations, language semantics, code generation, and optimization.

CSC 346 | *Parallel Computation*

- Study of techniques for parallel and high-performance computing.

CSC 355 | *Intro to Numerical Methods*

- Numerical computation on modern computer architectures; floating-point arithmetic and round-off error.

CSC 399 | *Computer Science Mastery Exam*

- Study and test of overall knowledge of computer science and the requirement to graduate as a major in computer science student.

Graduate CS Courses

CSE 555LEC | *Intro Pattern Recognition*

- Foundations of pattern recognition algorithms and machines, including statistical and structural methods. Data structures for pattern representation, feature discovery and selection, classification vs. description, parametric and non-parametric classification, supervised and unsupervised learning, use of contextual evidence, clustering, recognition with strings, and small sample size problems. programming projects.

CSE 568LEC | *Robotics Algorithms*

- Robotics Algorithms provides a comprehensive introduction to the software side of making mobile robots autonomous. We begin with an understanding of various kinds of robots and their kinematics. We then study the various sensors that can be used, and algorithms to perceive what the robot is doing (proprioception) as well as what is around it (exteroception). Most sensing and actuation are uncertain. We study Bayes filter as the framework for probabilistically reasoning about this uncertainty while being able to predict the robot state at any given instant. Specifically, we look at the problems of localization (where am I?) and mapping (what is around me?), the two fundamental problems in mobile robots. We then conclude the course by looking at planning and navigation algorithms to perform a given task along with robot control architectures to put all these together to achieve autonomy

CSE 573LEC | *Computer Vision and Image Processing*

- This course is an introduction to those areas of Artificial Intelligence that deal with fundamental issues and techniques of computer vision and image processing. The emphasis is on physical, mathematical, and information-processing aspects of the vision. Topics to be covered include image formation, edge detection, and segmentation, convolution, image enhancement techniques, extraction of features such as color, texture, and shape, object detection, 3-D vision, and computer vision system architectures and applications.

CSE 574LEC | *Intro to Machine Learning*

- Involves teaching computer programs to improve their performance through guided training and unguided experience. Takes both symbolic and numerical approaches. Topics include concept learning, decision trees, neural nets, latent variable models, probabilistic inference, time series models, Bayesian learning, sampling methods, computational learning theory, support vector machines, and reinforcement learning.

CSE 673LEC | *Computational Vision*

- This will mainly be a project-oriented course that will introduce students to recent advances in computer vision. The course will cover topics starting from the basics of deep learning to its applications in solving a variety of computer vision tasks like recognition, detection, segmentation, etc..

CSE 676LEC | *Deep Learning*

- Deep Learning algorithms learn multi-level representations of data, with each level explaining the data in a hierarchical manner. Such algorithms have been effective at uncovering underlying structures in data, e.g., features to discriminate between classes. They have been successful in many artificial intelligence problems including image classification, speech recognition, and natural language processing. The course, which will be taught through lectures and projects, will cover the underlying theory, the range of applications to which it has been applied, and learning from very large data sets. The course will cover connectionist architectures commonly associated with deep learning, e.g., basic neural networks, convolutional neural networks, and recurrent neural networks.