RUIZHE ZHAO

Room 353, Huxley Building, 180 Queen's Gate, SW7 2RH ⋄ London, United Kingdom
(+44) 7743 574128 ⋄ ruizhe.zhao15@imperial.ac.uk

EDUCATION

Department of Computing, Imperial College London October 2017 — Present
Ph.D. Program in EPSRC High-Performance Embedded and Distributed System *London, United Kingdom*

- Thesis: *Compiling Deep Learning to Reconfigurable Platforms* — using a compiler-based approach to transform high-level Deep Neural Network descriptions to low-level hardware designs with customised transformations.
- Supervisor: Prof. Wayne Luk
- Research interests: Deep Learning, compiler and programming languages, reconfiguration techniques, etc.
- Teaching activities: undergraduate courses on Haskell/Java programming and Reconfigurable computing.

Department of Computing, Imperial College London October 2016 — September 2017
MRes in Advanced Computing *London, United Kingdom*

- Average score: 90.2% (ranked 1st).
- Awarded *The Corporate Partnership Programme Awards for Academic Excellence.*

School of EECS, Peking University Sept. 2012 — July 2016
B.S. in Computer Science & Engineering *Beijing, China*

- Accomplished all honour courses (6 in total) on algorithms, compiler, architecture, OS and network.
- Awarded *Excellent Research* (2014/15) and *Top-10 Undergraduate Thesis.*

SELECTED PUBLICATIONS

- **R Zhao**, B Vogel, T Ahmed, W Luk. Reducing Underflow in Mixed Precision Training by Gradient Scaling. *2020 International Joint Conference on Artificial Intelligence (IJCAI).*
- **R Zhao**, W Luk., *et al.* On the challenges in programming mixed-precision deep neural networks. *2020 International Workshop on Machine Learning and Programming Languages (MAPL).*
- **R Zhao**, W Luk. Efficient Structured Pruning and Architecture Searching for Group Convolution. *2019 International Conference on Computer Vision (ICCV) Workshop.*
- **R Zhao**, HC Ng, W Luk, X Niu. Towards Efficient Convolutional Neural Network for Domain-Specific Applications on FPGA. *2018 International Conference on Field Programmable Logic and Applications (FPL).*
- **R Zhao**, X Niu, Y Wu, W Luk, Q Liu. Optimizing CNN-based Object Detection Algorithms on Embedded FPGA Platforms. *2017 International Symposium on Applied Reconfigurable Computing (ARC).*
- WT Tang, **R Zhao**, M Lu, *et al.* Sparse Matrix-Vector Multiplication for Scale-Free Matrices on Intel Xeon Phi. *2015 International Symposium on Code Generation and Optimisation (CGO).*
- W Moses, L Chelini, **R Zhao**, O Zinenko. Polygeist: Affine C in MLIR. *2021 11th International Workshop on Polyhedral Compilation Techniques (IMPACT).*

WORKING EXPERIENCE

Facebook London September 2019 — December 2019
Software Engineering Intern, Machine Learning *London, UK*

- Implemented machine learning models for ads ranking that leverage additional training data with delayed feedback through carefully designed loss functions; idea based on `https://arxiv.org/abs/1907.06558`.
- Experienced with deploying models on large-scale machine learning platform and processing gigantic data.
- Collaborated and communicated effectively among team members; awarded return internship for June 2021.

Preferred Networks, Inc. May 2019 — July 2019
Research Intern *Tokyo, Japan*

- Supervisors: Dr. Brian Vogel and Dr. Tanvir Ahmed
- Worked on improving the training stability and efficiency when using *mixed-precision* on modern GPU platforms; achieved better performance than the state-of-the-art results from NVIDIA.
- Dived deep into the implementation of *Chainer*, a dynamic graph based Deep Learning framework similar to PyTorch; adapted DNN models training on large-scale GPU clusters.

- Published *Reducing Underflow in Mixed Precision Training by Gradient Scaling* at IJCAI '20; code accessible at `https://github.com/kumasento/gradient-scaling`.

Corerain Technologies Ltd. June 2016 — April 2019
Compiler Team Manager and Main Developer *London, UK and Shenzhen, China*

- Built a DNN-to-FPGA compilation and optimization framework and a cross-platform (CPU/GPU/FPGA) graph runtime from scratch, extensively used C/C++ and Python;
- Managed a team of 2 engineers and mentored 4 interns.

OPEN-SOURCE CONTRIBUTIONS

- Authored **Polymer**: a MLIR-based polyhedral compiler (`https://github.com/kumasento/polymer`) that first enables polyhedral transformations for MLIR.
- Co-developed **Polygeist** (`https://github.com/wsmoses/Polygeist`), a MLIR-based C/C++ polyhedral compilation framework, with collaborators from MIT, Google, and TU Eindhoven.
- Committer to compilation tools for software and hardware, including LLVM, MLIR, and CIRCT.

RESEARCH EXPERIENCE

Department of Computing, Imperial College London July 2015 — September 2015
Custom Computing Group, Summer Research Intern *London, UK*

- Supervisors: Prof. Wayne Luk and Dr. Timothy Todman.
- Introduced multi-pumping, a hardware optimisation technique for better resource utilization, to MaxCompiler, a High-Level Synthesis platform; code accessible at `https://github.com/imperial-summer-research`.

Center for Energy-efficient Computing and Application March 2013 — July 2016
Computer Architecture Group, Research Assistant *Beijing, China*

- Supervisor: Prof. Yun Liang.
- Undergraduate Thesis: Designed and implemented a Caffe-like deep learning framework, **SoCaffe**, on the Zynq System-on-Chip (SoC) platform with optimised GEneral Matrix Multiplication (GEMM) kernels; code accessible at `https://github.com/pku-ceca-research/SoCaffe`.

Institute of High Performance Computing, A* Star July 2014 — September 2014
Computing Science Group, Overseas placement *Singapore*

- Supervisors: Dr. Waiteng Tang and Dr. Mian Lu.
- Optimized memory and cache performance of Sparse Matrix Vector (SpMV) multiplication on Intel Xeon Phi many-cores co-processor, under the OpenMP framework;
- Explored the best formats for *scale-free matrix* that achieves high memory performance on the target platform;
- Co-authored *Sparse Matrix-Vector Multiplication for Scale-Free Matrices on Intel Xeon Phi* (CGO '15).

SELECTED COURSE PROJECTS

Singleton: A functional programming language with race-condition free type system May 2015 — June 2015
Two-members team, Design Principles of Programming Languages *Beijing, China*

- Co-designed Singleton, an experimental language, focusing on keeping race-condition type-checked statically by compiler. It is implemented in OCaml, another industrial-level functional programming language, originated from SML. Alms, a language that has utilised many theoretical concepts like affine types to support race-condition free type system, has been studied to build the core design of Singleton. The work contains implementation of the front-end lexer, parser, and type checker.
- Available on `https://github.com/network-hw/singleton`.

TECHNICAL STRENGTHS

Compiler	Contributing to the LLVM/MLIR framework
Algorithms	Practicing competitive programming in free time
Programming Languages	Strong in C++(11/14/17) and Python, know Haskell & OCaml
Hardware Deign	Xilinx tool-chain and boards, MaxCompiler