ZHENG ZHAN

140 The Fenway, Boston, MA 02115

(857)891-6751 zhan.zhe@northeastern.edu Zheng's Homepage Zheng Zhan

EDUCATION

Northeastern University Boston, MA
Ph.D. Candidate in Computer Engineering, GPA: 4.0/4.0 Sep 2019 – May 2024 (expected)

- Advisor: Prof. Yanzhi Wang
- Focus on *Model Compression, Continual Learning*.

Syracyse University Syracuse, NY
Master of Science in Computer Engineering, GPA: 3.833/4.0 Sep 2017 – May 2019

Xidian University Xi'an, Shaanxi, China
Bachelor of Engineering in Electronic Science and Technology Sep 2013 – Jun 2017
Excellent Class **(Undergraduate Honor Program)**

EXPERIENCE

Samsung Research America Mountain View, CA
Ph.D. Research Intern May 2022 – Aug 2022

- *Project: Efficient Vision Transformer using linear self-attention for large inputs*

Lawrence Livermore National Laboratory Livermore, CA
Ph.D. Research Intern @ DSSI program May 2021 – Aug 2021

- *Project: Multi-Prize Lottery Tickets of Vision Transformer*

Northeastern University Boston, MA
Research Assistant advised by Prof. Yanzhi Wang @ College of Engineering Sep 2019 – present

Efficient and Effective Continual Learning

We develop SparCL, which explores sparsity for efficient continual learning and achieves both training acceleration and accuracy preservation through the synergy of three aspects: weight sparsity, data efficiency, and gradient sparsity. (**NeurIPS-22, ICML-23**)

- Training acceleration through the TDM, DDR, and DGM. Leading to at most 23× fewer training FLOPs and an 1.7% improvement over SOTA accuracy.
- Achieve at most 3.1× training acceleration on a real mobile edge device.

Effective compression-DVFS co-design

We propose a highly representative pruning framework (a single neural network containing multiple sparsity ratios) to work with dynamic power management using DVFS. (**DAC-23**, **ICCAD-22**)

- Develop a framework which leverages the DVFS and compression techniques to get multiple subnetworks in one neural network to lower the variance of inference runtime for different hardware frequency levels. (**ICCAD-22**)
- Propose a two-level algorithm for obtaining subnets with arbitrary ratios in a single model with theoretical proof. It's a much more automatic framework. (**DAC-23**)

Effective compression-compiler co-design

Project: Compression-Compilation Co-design (CoCoPIE) Feb 2020 – present
Content: CoCoPIE , a **startup** developing a platform that optimizes AI models for edge devices, **that has raised $6 million in funding**.

Lead the Core project of achieving **Real-Time Super-Resolution on Mobile platform**, We are **the first** to achieve real-time SR inference (with only tens of milliseconds per frame) for implementing 720p resolution with competitive image quality (in terms of PSNR and SSIM) on mobile platforms. (**ICCV-21**, **ECCV-22**)

- Develop a framework that leverags pruning search and NAS to achieve real-time SR inference on the mobile. (**ICCV-21**)
- Propose a layer-wise and compiler-aware NAS algorithm with corresponding compiler-level optimizations. (**ECCV-22**)

Published papers in top-tier conferences. (**NeurIPS**, **ICCV**, **CVPR**, **ECCV**, **DAC**, **ICCAD** etc.)

University of Toronto Toronto, ON, Canada
Research Assistant advised by Prof. Baochun Li @ Department of ECE Jul 2018 – Feb 2019

- *Project: Scheduling Machine Learning Jobs with Reinforcement Learning* (IWQoS-19)

Syracuse University Syracuse, NY
Research Assistant advised by Prof. Yanzhi Wang @ College of ECS Sep 2017 – May 2019

- *Project: Stochastic Computing and Universal Approximation Theory*
  Prove the equivalence of Stochastic Computing-based Neural Networks (SCNN) and BNN by using Universal Approximation theory. (**Coauthor and present the work in AAAI-19**)

SELECTED PUBLICATIONS

Conference Papers, * means equal contribution. [...] is hyperlink button.

- Zifeng Wang*, **Zheng Zhan***, Yifan Gong, Yucai Shao, Stratis Ioannidis, Yanzhi Wang, Jennifer Dy. "DualHSIC: HSIC-Bottleneck and Alignment for Continual Learning." In submission.
- **Zheng Zhan**, Yifan Gong, Pu Zhao, et al, "Condense: A Framework for Device and Frequency Adaptive Neural Network Models on the Edge". **DAC 2023**.
- Zifeng Wang*, **Zheng Zhan***, Yifan Gong, et al, "SparCL: Sparse Continual Learning on the Edge". **NeurIPS 2022**. [paper] [code]
- Yifan Gong*, **Zheng Zhan***, et al, "All-in-One: A Highly Representative DNN Pruning Framework for Edge Devices with Dynamic Power Management". **ICCAD 2022**. [paper]
- Yushu Wu*, Yifan Gong*, Pu Zhao, Yanyu Li, **Zheng Zhan** et al, "Compiler-Aware Neural Architecture Search for On-Mobile Real-time Super-Resolution". **ECCV 2022**. [paper] [code]
- **Zheng Zhan***, Yifan Gong*, Pu Zhao* et al, "Achieving on-Mobile Real-Time Super-Resolution with Neural Architecture and Pruning Search". **ICCV 2021**. [paper]
- Geng Yuan, Xiaolong Ma, Wei Niu, Zhengang Li, Zhenglun Kong, Ning Liu, Yifan Gong, **Zheng Zhan** et al, "MEST: Accurate and Fast Memory-Economic Sparse Training Framework on the Edge". **NeurIPS 2021**. (**Spotlight paper, top 3%**) [paper] [code]
- Zhengang Li*, Geng Yuan*, Wei Niu*, Pu Zhao, Yanyu Li, Yuxuan Cai, Xuan Shen, **Zheng Zhan** et al, "NPAS: A Compiler-Aware Framework of Unified Network Pruning and Architecture Search for Beyond Real-Time Mobile Acceleration". **CVPR 2021**. (**Oral paper, top 5%**) [paper]
- Tianyun Zhang, Xiaolong Ma, **Zheng Zhan** et al, "A Unified DNN Pruning Weight Framework Using Reweighted Method". **DAC 2021**. [paper]
- Yanzhi Wang, **Zheng Zhan**, Liang Zhao et al, "Universal Approximation Property and Equivalence of Stochastic Computing-based Neural Networks and Binary Neural Networks". **AAAI 2019**. [paper]

Journal Papers

- Yifan Gong*, Geng Yuan*, **Zheng Zhan** et al, "Automatic Mapping of the Best-Suited DNN Pruning Schemes for Real-Time Mobile Acceleration", *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 2021. [paper]
- Tong Jian, Yifan Gong, **Zheng Zhan** et al, "Radio Frequency Fingerprinting on the Edge", *IEEE Transactions on Mobile Computing*, 2021. [paper]

SKILLS

- Machine Learning Framework: PyTorch, TensorFlow.
- Programming Languages: Python, C/C++ (wrote a Remote Test Harness), C# (wrote a Remote Build Server), MATLAB.