QIWEI (JOSHUA) YUAN

Salt Lake City, UT, USA | joshuayqw@gmail.com | https://joshuayy.github.io/

EDUCATION

University of Utah **Salt Lake City, UT, USA**
Doctor of Philosophy-Computer Science(GPA: 3.9/4) *Jan 2022∼*

- Research focus(2022-2023): High Performance Computing(HPC); GPU Kernel Design for Convolutional Neural Network Speedup; CUDA Algorithm Development for CPU-Code Adaption
- Research focus(2023-): Probabilistic Machine Learning, ML-based PDE Solver, Fourier Neural Operator, Tensor Decomposition
- Selected courses: *Parallel Programming*, *Operating Systems*, *Computer Architecture*, *Graduate Algorithms*, *Deep Learning Systems*, *Probabilistic Machine Learning*, *Parallel Program Many-Core System*

University of Washington **Seattle, WA, USA**
Master of Science in (Mechanical)Engineering(GPA: 3.73/4) *Sep 2017∼ Jul 2019*

- Selected courses: *Convex Optimization*, *Scientific Computing*, *Networked Dynamic System*, *Linear Multivariable Control*, *Linear System*, *Mathematics Foundation of System Theory*, *Data Programming*

University of Shanghai for Science and Technology **Shanghai, China**
Bachelor of Engineering(GPA: 3.37/4) *Sep 2006∼ Jun 2010*
Major: Mechanical Design, Manufacture and Automation(AUTOCAD, SOLIDWORKS)

PROFESSIONAL EXPERIENCE

University of Utah **Salt Lake City, UT, USA**
Research Assistant *Nov 2019∼Dec 2021*
Keywords: Reinforcement Learning (RL); RL application on combinatorial optimizations; Deep Learning Regulator; Machine Learning; Tensorflow; Image Classification; Inference Engine
Achievement: Complete a customer Deep Neural Network Inference Engine to replace TensorFlow's

Schneider Electric **Shanghai, China**
Project Leader(Experienced Technical Engineer) *2014∼2017*
Keywords: Project Management, Technology Evolving, Trouble Shooting

KOITO CO., LTD **Shanghai, China**
Senior Mechanical Engineer(Production Equipment Design and Manufacture) *2010∼2014*
Keywords: Machine Design, Technology Management, CATIA V5

PROJECTS

Contrastive Weight Regularization

- Configure new regularization called 'DReg' which can accelerate converge speed and improve validation accuracy of traditional deep learning especially in large batch;
- Test new 'DReg' on several DNNs (*ResNet18*, *MobileNetV2* and *VGG16*) in dataset (*MNIST*, *CIFAR10* and *CIFAR100*) training.(Pytorch, CUDA, Vim, Linux-coding involved over 1000 lines)

Heuristic Solver Development for Combinatorial Problems(RL-based)

- **Combinatorial Reinforcement Learning Based Scheduling for DNN Execution on Edge**
  - Map training layers of DNNs on Edge TPU pipelines using Reinforcement Learning based Scheduling (learning agent as Ptr-Net);
  - Configure synthesis datasets for training and testing (500 coding lines), modify original Ptr-Net to satisfy research requirement (400 coding lines), different combination of LSTM and Transformer on encoder and decoder (200 coding lines), post processing on result for compiling in Edge TPU (1000 coding lines). (Pytorch, CUDA, Vim, Linux-coding involved over 2500 lines);
  - Test running time of 10 middle-size DNNs (*Xception*, *ResNet50*, *ResNet101*, *ResNet152*, *Denset201*, *DenseNet121*, *ResNet101V2*, *ResNet152V2*, *DensetNet169*, *InceptionResNetV2*) on Edge TPU with proposed method against google commercial compiler. (up to $\sim 2.5\times$ speedups).

Effective Performance Modeling and Domain-Specific Compiler Optimization of CNNs for GPUs

- Configure the optimal CUDA parameters setting for Convolutional Neural Network(CNNs) execution by leveraging the knowledge of the register usage, tiling effectiveness, data mapping method and so on;
- Test **TVM/Ansor** auto-Tuning/Scheduling on CNN Layers of *Yolo9000* and *ResNet18* to investigate the stability and efficiency of active search, setting the baseline for performance evaluation of the proposed method(TVM/Ansor, Python, Visual Studio coding over 100 lines);
- Design script to post-treat experiment results automatically(Python, Visual Studio coding over 200 lines).

Kalman Filter GPU CUDA Kernel Design

- Transport related Kalman Filter Code from CPU to GPU(C++, CUDA-C);
- Optimize GPU CUDA Kernel by manipulating data in global memory, shared memory and register usage to fully take advantage of GPU computing power(C++, CUDA-C, Visual Studio coding over 200 lines);
- Profiling kernel execution performance using Nsight Compute to search for potential improvement.

Machine Learning Based Data Remapping to Reduce Branch Divergence in CUDA Parallel Programming

- Design a customer CUDA Kernel with severe branch divergence(C++, CUDA-C, 200 coding lines);
- Design a Deep Neural Network including input encoding and ground truth construction(Pytorch, 300 coding lines);
- Speed up the CUDA Kernel by feeding the remapped data points(up to $\sim 1.62\times$ speedups).

Develop Gaussian Process for Operator Learning

- Apply Fourier Neural Operator as data pre-processing method for High-Order Gaussian Process;
- Increase prediction accuracy on the dataset of *Navier Stokes*, *Burger*, *Advection*, and so on.(Pytorch, over 500 coding lines);
- Design active learning method for efficient sampling to reduce data generating burden and improve data usage(Pytorch, roughly 600 coding lines).

Tensor Decomposition on Data Imputation: Boost the Performance of Traditional Tensor Decomposition Method(CP or Tucker) using MADE Neural Network

- Design a systematic way to connect MADE neural network training and traditional tensor decomposition refining by feeding refreshed data of each;
- Increase the prediction accuracy on several benchmark dataset like *beijing air pollution*.(Pytorch, more than 800 coding lines);
- Investigate the potential of predicting future values based on history dataset.(Pytorch, over 1000 coding lines)

PUBLICATIONS

- **Qiwei Yuan**, Weizhe Hua, Yi Zhou and Cunxi Yu. *Contrastive Weight Regularization for Large Minibatch SGD* (arXiv-https://arxiv.org/abs/2011.08968)
- Yufan Xu, **Qiwei Yuan**, Eric Curtis Barton, Rui Li, P.Sadayappan and Aravind Sukumaran-Rajam. *Effective Performance Modeling and Domain-Specific Compiler Optimization of CNNs for GPUs* The International Conference on Parallel Architectures and Compilation Techniques(PACT'31)

OTHER SKILLS

Program Language : Python, C++, CUDA-C, Matlab, and Java. Platform: Linux (RHEL, Ubuntu)