
JST AIP Network Lab.
4-th JST-NSF-DATAIA International Joint Symposium
“ Cutting-Edge of AI Research
~ To Realize Society 5.0 / Smart and Connected Communities ~”

Dec 19, 2019

# Toward vastly large deep learning

Koichi Shinoda
(Tokyo Institute of Technology)

JST CREST Development and Integration of Artificial Intelligence Technologies for Innovation Acceleration

# Fast and cost-effective deep learning algorithm platform for video processing in social infrastructure


PI Koichi Shinoda (TokyoTech)

Co-PI Satoshi Matsuoka (Riken)

Masaki Onishi (AIST)

Rio Yokota (TokyoTech)

Tsuyoshi Murata (TokyoTech)

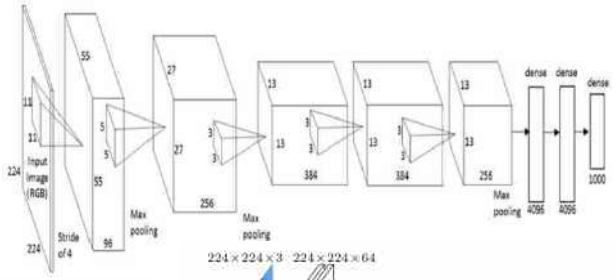
Hiroki Nakahara (TokyoTech)

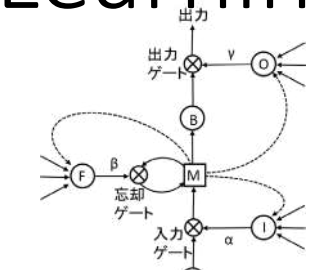
Taiji Suzuki (U Tokyo)

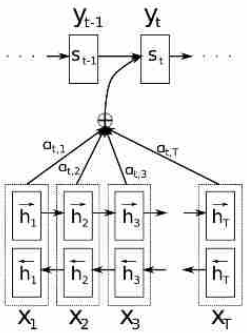
# Background

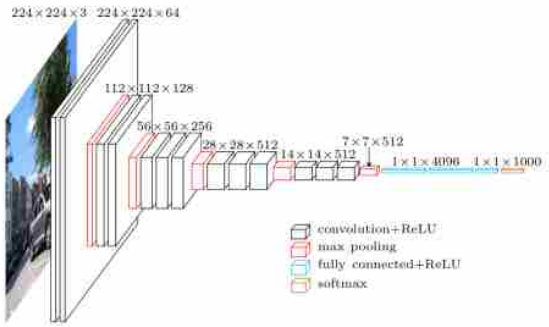
- Video processing for safe and secure society
  - Prevent traffic accidents (Dashcam)
  - Detect abnormalities (Surveillance camera)
- Deep Learning
  - Much better detection performance than before
  - IT Giants dominate the field

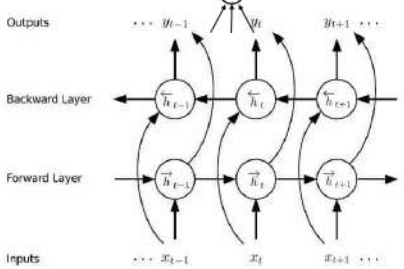
# Probably no need to explain what is Deep Learning…

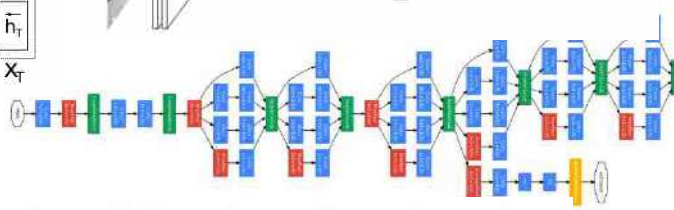


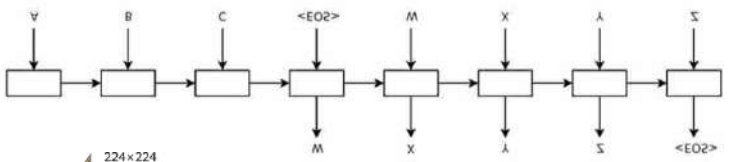


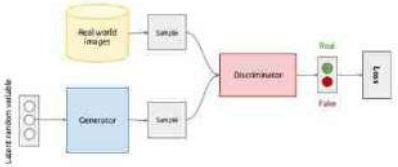



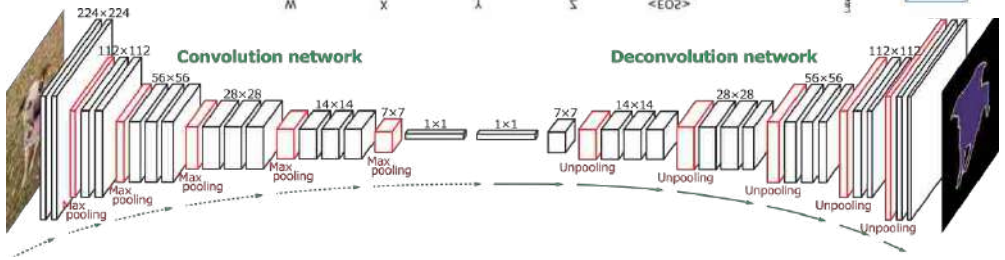


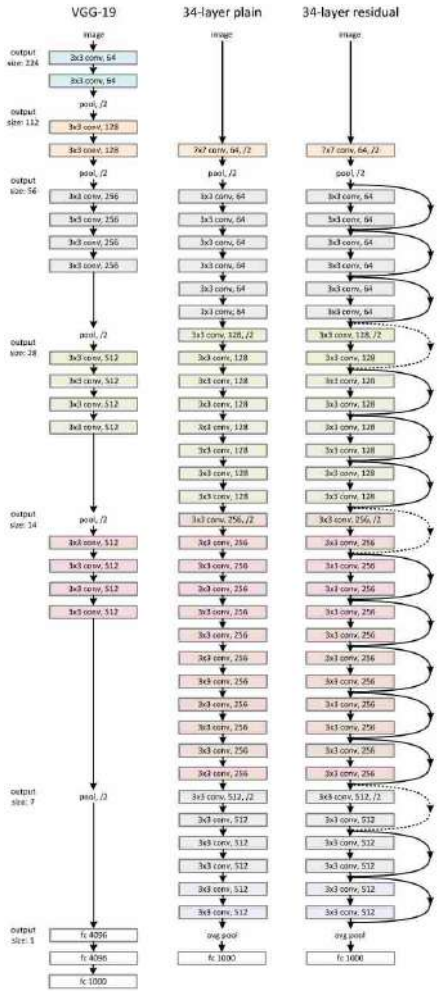









# Problem

1. Analyze a huge amount of images in real-time
2. Rapidly Adapt to the changes in environmental conditions
3. Edge Computing
   - Reduce traffics on Internet

These problems are deeply related with each other

→ Simultaneous optimization

# Our approach

- Develop fast and cost-effective deep-learning algorithm platform for video processing using TokyoTech supercomputer TSUBAME
  - TSUBAME 3.0, 2160 nodes, No.1 in Green 500
    It is 10 times faster than 「京」(Kei) for machine learning
  - High standard video search technologies
    The top group in NIST TRECVID workshop
- Co-Design framework (explain later..)

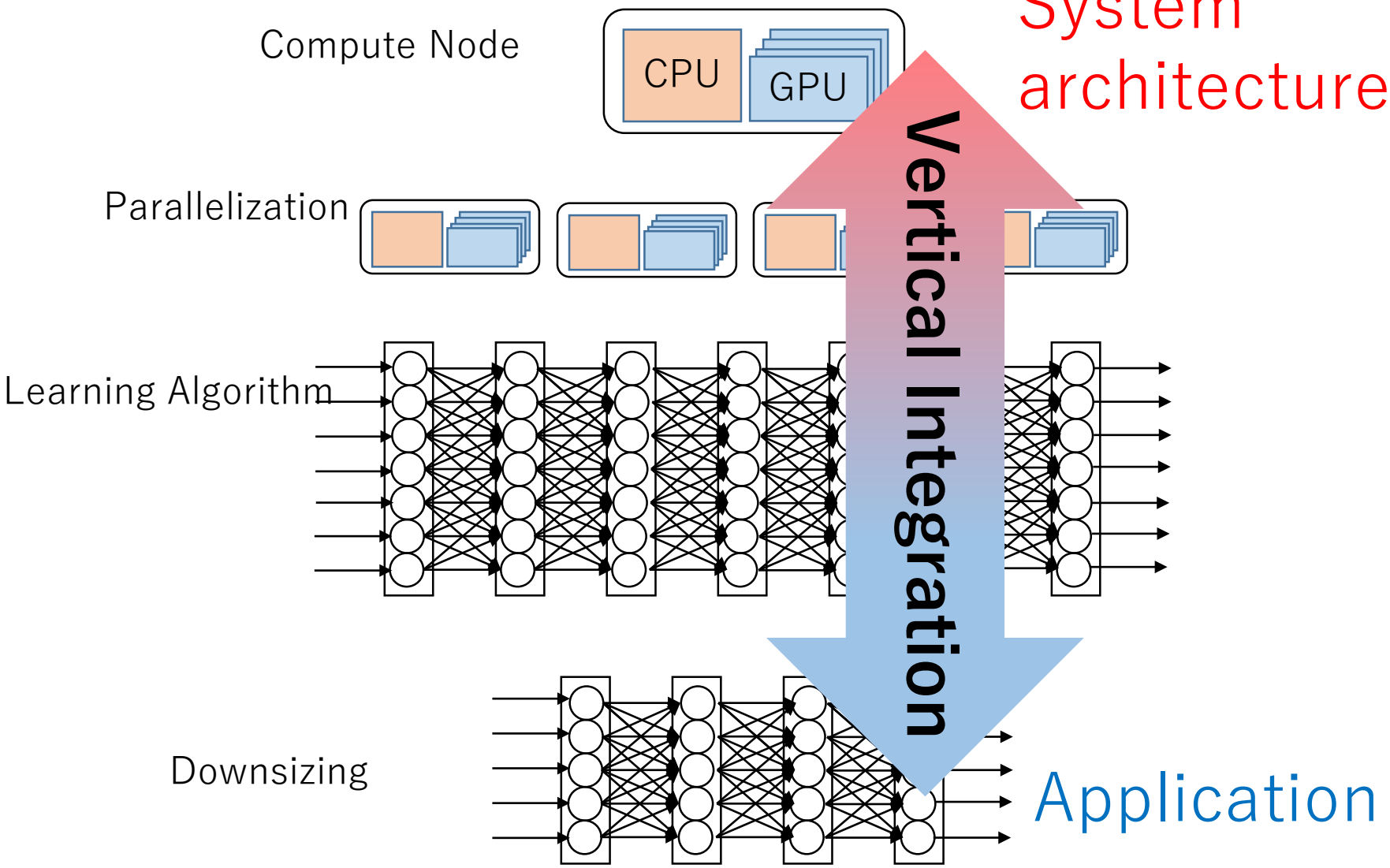
  From system architecture to applications, researchers in different areas collaborate together to maximize the total throughput
- Open platform

  Work with sensor and network companies to compete with IT giants

# Small Phase

Dec 2016 – Mar 2019

# Co-Design framework




1000x speed by 1/1000 memory

# Goal in Small Phase

| Component | Speed | Memory |
|---|---|---|
| Compute node (Yokota G) | 50x | 1/10 |
| Parallelization (Matsuoka G) | 10x | |
| Learning Algorithm (Shinoda G) | 10x | 1/10 |
| Downsizing (Murata G) | | 1/100 |
| Total | $>$ 1000x | $<$ 1/1000 |

# What we showed you two years ago in JST-NSF workshop 2017…

| Component | Speed | Memory |
|---|---|---|
| Compute node | 7.4x (50x) | 1/15(1/10) |
| Parallelization | 11.6x*(10x) | |
| Learning Algorithm | 11.6x*(10x) | 2*(1/10) |
| Downsizing | | 1/90(1/100) |
| Total | > x1000 | < 1/1000 |

* : Achievement obtained by the joint work of the two groups

# Our achievement in Small Phase

| Component | Speed | Memory |
|---|---|---|
| Compute node (Yokota G) | 18x (50x) | 1/5(1/10) |
| Parallelization (Matsuoka G) | 1536x (10x) | ? |
| Learning Algorithm (Shinoda G) | 10x (10x) | ?(1/10) |
| Downsizing (Murata G) | | 1/90(1/100) |
| Total | > 1500x | ? |

# Large Phase

Apr 2019 – Mar 2022

# Fugaku: Game Changer

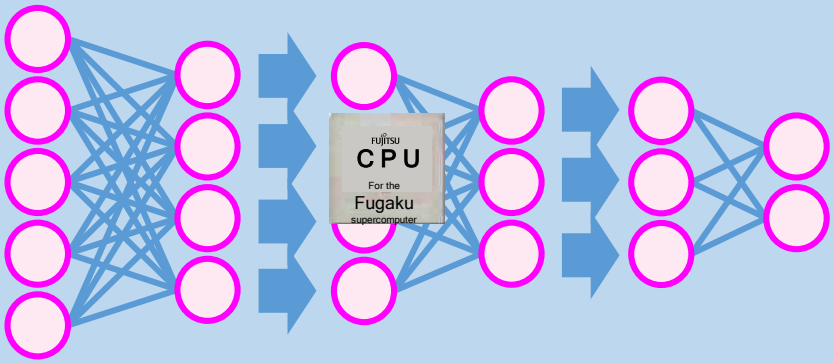
with 150,000 nodes

Prof. Satoshi Matsuoka

## Fugaku Processor

- High performance in FP16&Int8
- High mem band width
- Built-in scalable TOFU network

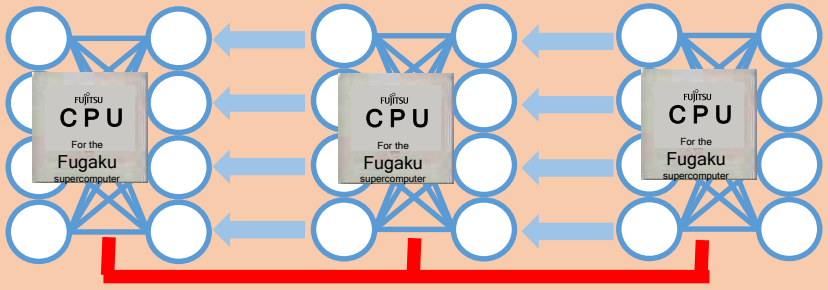
High Performance DNN Convolution




## Unprecedented scalability

Ultra-scalable network

Massive scaling
model & data parallelism



# Large Scale Public AI Infrastructures in Japan

| | Deployed | Purpose | AI Processor | Inference Peak Perf. | Training Peak Perf. | Top500 Perf/Rank | Green500 Perf/Rank |
|---|---|---|---|---|---|---|---|
| Tokyo Tech. TSUBAME3 | July 2017 | HPC + AI Public | NVIDIA P100 x 2160 | 45.8 PF (FP16) | 22.9 PF / 45.8PF (FP32/FP16) | 8.125 PF #22 | 13.704 GF/W #5 |
| U-Tokyo Reedbush-H/L | Apr. 2018 (update) | HPC + AI Public | NVIDIA P100 x 496 | 10.71 PF (FP16) | 5.36 PF / 10.71PF (FP32/FP16) | (Unranked) | (Unranked) |
| U-Kyushu ITO-B | Oct. 2017 | HPC + AI Public | NVIDIA P100 x 512 | 11.1 PF (FP16) | 5.53 PF/11.1 PF (FP32/FP16) | (Unranked) | (Unranked) |
| AIST-AIRC AICC | Oct. 2017 | AI Lab Only | NVIDIA P100 x 400 | 8.64 PF (FP16) | 4.32 PF / 8.64PF (FP32/FP16) | 0.961 PF #446 | 12.681 GF/W #7 |
| Riken-AIP Raiden | Apr. 2018 (update) | AI Lab Only | NVIDIA V100 x 432 | 54.0 PF (FP16) | 6.40 PF/54.0 PF (FP32/FP16) | 1.213 PF #280 | 11.363 GF/W #10 |
| AIST-AIRC ABCI | Aug. 2018 | AI Public | NVIDIA V100 x 4352 | 544.0 PF (FP16) | 65.3 PF/544.0 PF (FP32/FP16) | 19.88 PF #7 | 14.423 GF/W #4 |
| NICT (unnamed) | Summer 2019 | AI Lab Only | NVIDIA V100 x 1700程度 | ~210 PF (FP16) | ~26 PF/~210 PF (FP32/FP16) | ???? | ???? |
| C.f. US ORNL Summit | Summer 2018 | HPC + AI Public | NVIDIA V100 x 27,000 | 3,375 PF (FP16) | 405 PF/3,375 PF (FP32/FP16) | 143.5 PF #1 | 14.668 GF/W #3 |
| Riken R-CCS Fugaku | 2020 ~2021 | HPC + AI Public | Fujitsu A64fx > x 150,000 | > 4000 PO (Int8) | >1000PF/>2000PF (FP32/FP16) | > 400PF #1 (2020?) | > 16 GF/W |
| ABCI 2 (speculative) | 2022 ~2023 | AI Public | Future GPU ~ 3000 | Similar | similar | ~100PF | 25~30GF/W ??? |

# How to reach the goal?

- Massively parallel processing which can scale with 100,000 nodes.
  - Make second-order optimization De Facto
  - Model parallelism
- Video with higher resolution (HD → 4K → 8K)
- Scale up deep learning algorithm
  - Data augmentation using large-scale GAN
  - Use structured knowledge graphs as inputs
  - End-to-end model for video data
- FPGA implementation
- Theory, Simulation
- Evaluation in real environment (Benchmarking)

# Scale up Co-Design Framework

: NEW

100,000 nodes

Large-scale GAN

Graph DNN

Seq. model

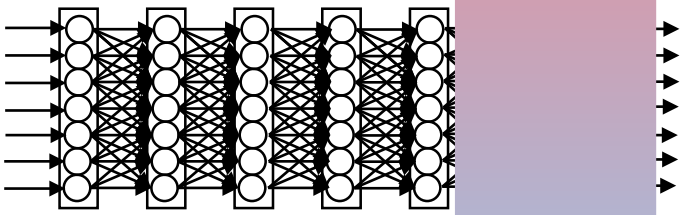
Node

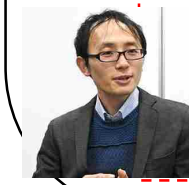
CPU

GPU

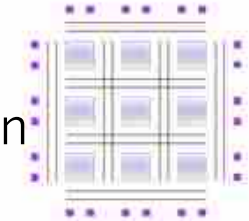
Parallelization

Training algorithm





FPGA implementation



Theory and simulation

Evaluation in real environment

We are studying many topics,
but here I introduce two of them...

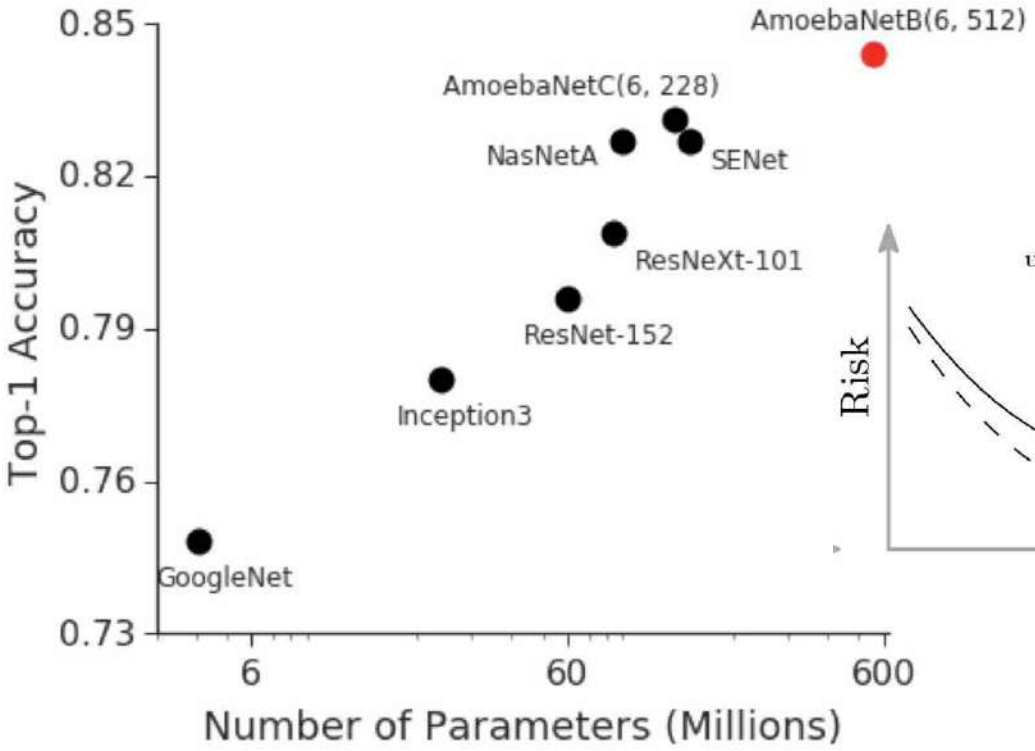
1. Second order optimization
   for massively parallel computing

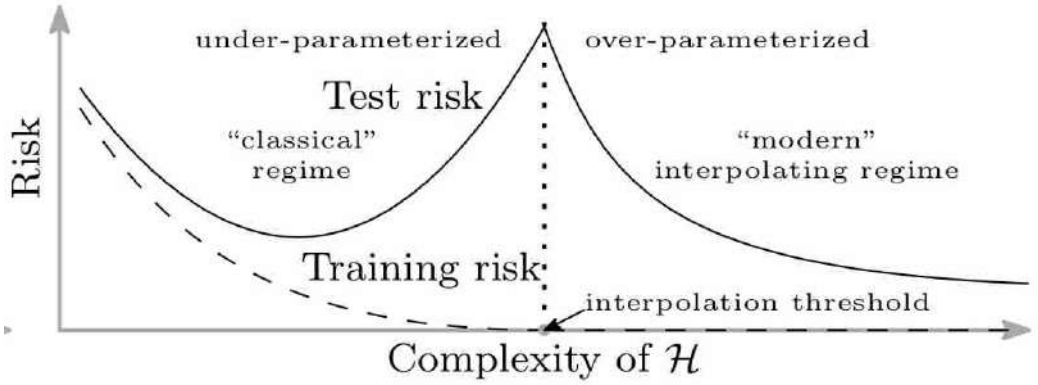
2. Ternary deep neural network
   accelerator for edge computing

# Second order optimization for massively parallel computing

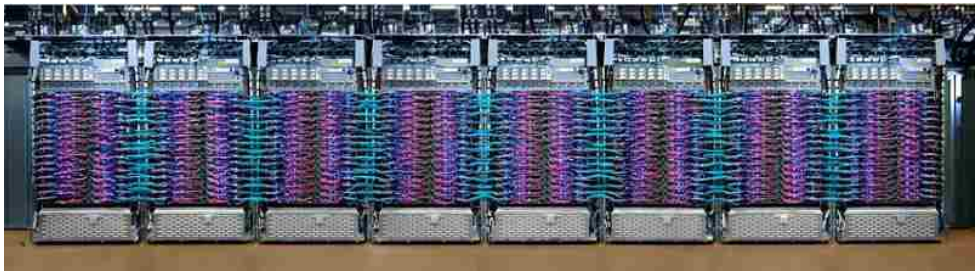
Prof. Rio Yokota

Training at a Scale Only Possible on Supercomputers

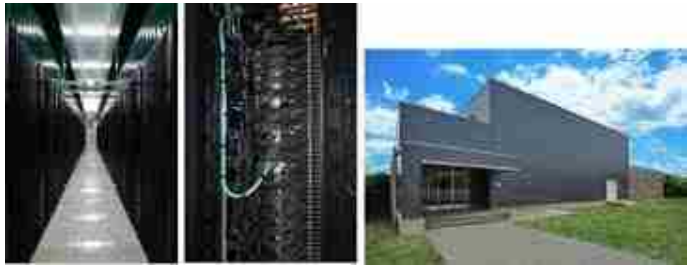


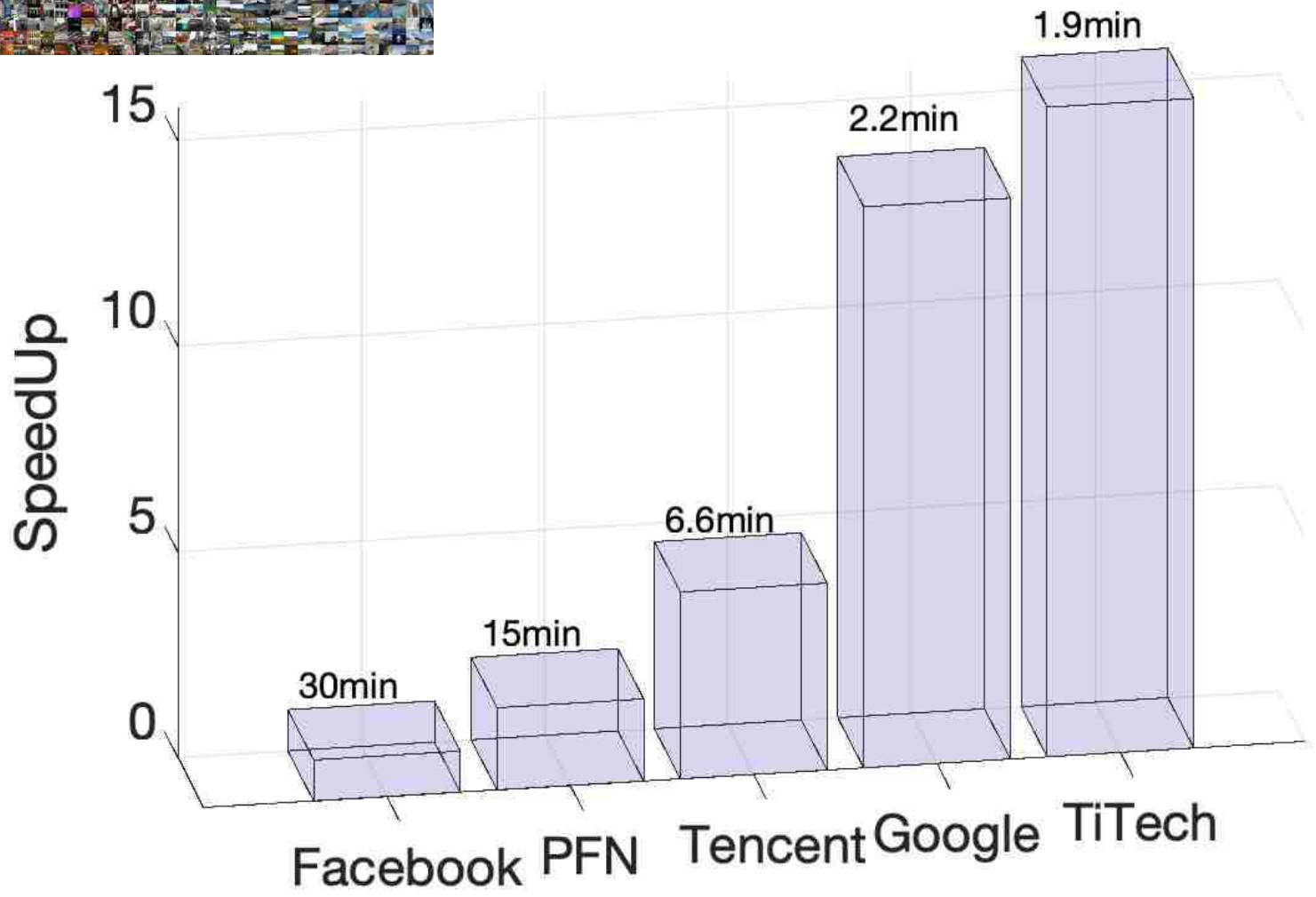

Google TPU v3 12.5PF



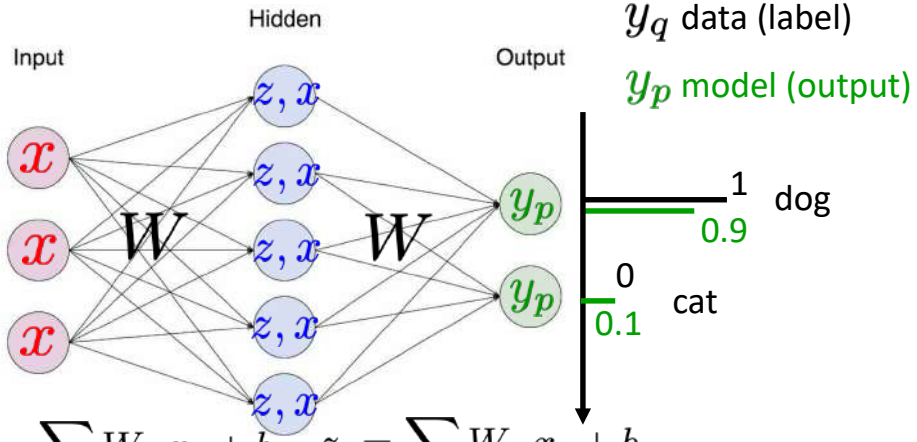
AIST ABCI 17 PF



ImageNet Can be Trained in a Few Minutes

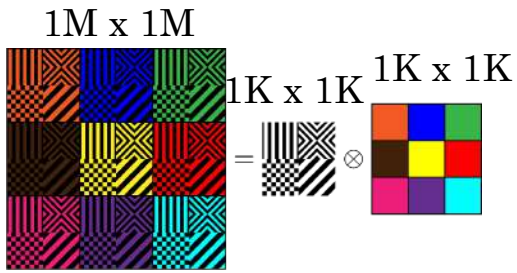

512GPU




Fisher Matrix

$$\begin{bmatrix} \frac{\partial J}{\partial W}^T \frac{\partial J}{\partial W} \end{bmatrix}$$

1Mx1M




Kronecker Factorization

## Cross Entropy Loss

$$J = \mathbb{E}_q \left( -\log y_p \right)$$

SGD $$W \leftarrow W - \eta \frac{\partial J}{\partial W}$$

Newton (Hessian) $$W \leftarrow W - \eta \mathbb{E}_q \left[ \frac{\partial^2 J}{\partial W^2} \right]^{-1} \frac{\partial J}{\partial W}$$

Gauss-Newton (Covariance) $$W \leftarrow W - \eta \mathbb{E}_q \left[ \frac{\partial J}{\partial W}^T \frac{\partial J}{\partial W} \right]^{-1} \frac{\partial J}{\partial W}$$

Natural Gradient (Fisher) $$W \leftarrow W - \eta \mathbb{E}_p \left[ \frac{\partial J}{\partial W}^T \frac{\partial J}{\partial W} \right]^{-1} \frac{\partial J}{\partial W}$$

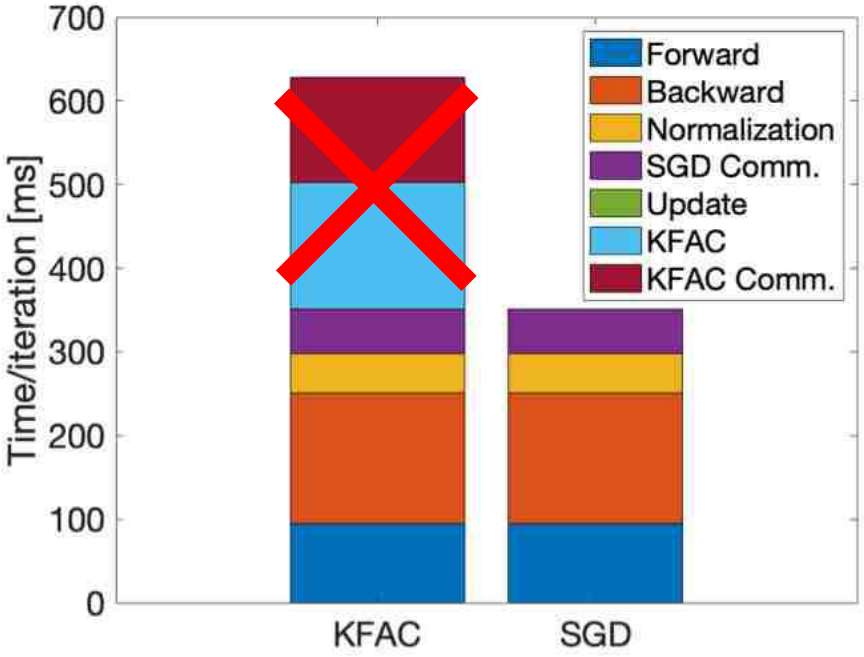
## Back propagation

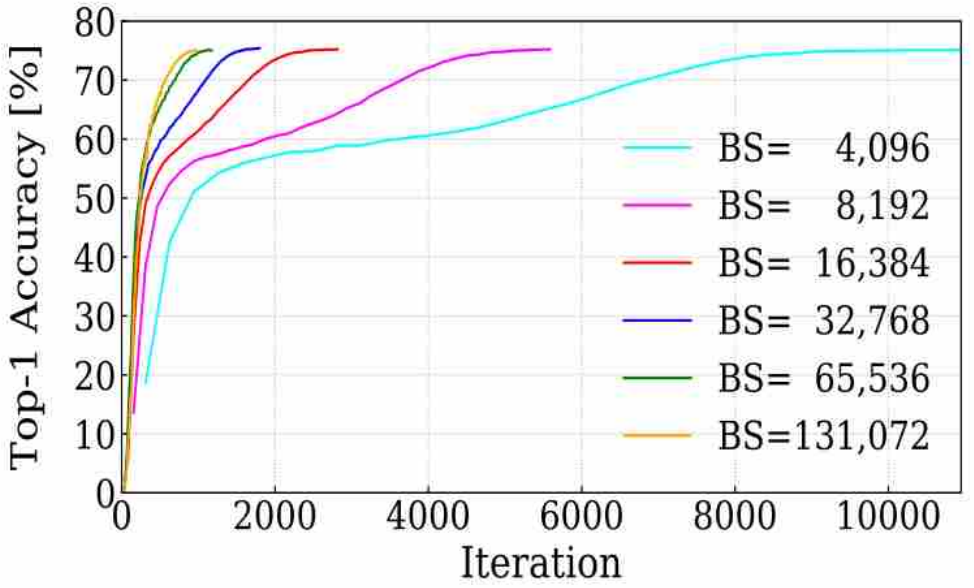
$$\frac{\partial J}{\partial W_{ij}} = \frac{\partial J}{\partial y_{pi}} \frac{\partial y_{pi}}{\partial z_i} \frac{\partial z_i}{\partial W_{ij}}$$

$$\underset{1M}{\frac{\partial J}{\partial W}} = \underset{1K}{\frac{\partial J}{\partial z}} \otimes \underset{1K}{\frac{\partial z}{\partial W}}$$

Kronecker Product

Eliminated the Overhead of Second Order Methods




| | Hardware | Software | Mini-batch size | Optimizer | Epoch | Time | Accuracy |
|---|---|---|---|---|---|---|---|
| Goyal *et al.* | Tesla P100 × 256 | Caffe2 | 8,192 | SGD | 90 | 1 hr | 76.3% |
| You *et al.* | KNL × 2048 | Intel Caffe | 32,768 | SGD | 90 | 20 min | 75.4% |
| Akiba *et al.* | Tesla P100 × 1024 | Chainer | 32,768 | RMSprop → SGD | 90 | 15 min | 74.9% |
| You *et al.* | KNL × 2048 | Intel Caffe | 32,768 | SGD | 64 | 14 min | 74.9% |
| Jia *et al.* | Tesla P40 × 2048 | TensorFlow | 65,536 | SGD | 90 | 6.6 min | 75.8% |
| Mikami *et al.* | Tesla V100 × 2176 | NNL | 34,816 → 69,632 | SGD | 90 | 3.7 min | 75.0% |
| Ying *et al.* | TPU v3 × 1024 | TensorFlow | 32,768 | SGD | 90 | 2.2 min | 76.3% |
| This work | Tesla V100 × 1024 | Chainer | 32,768 | K-FAC | 45 | ~~10~~ min | 74.9% |

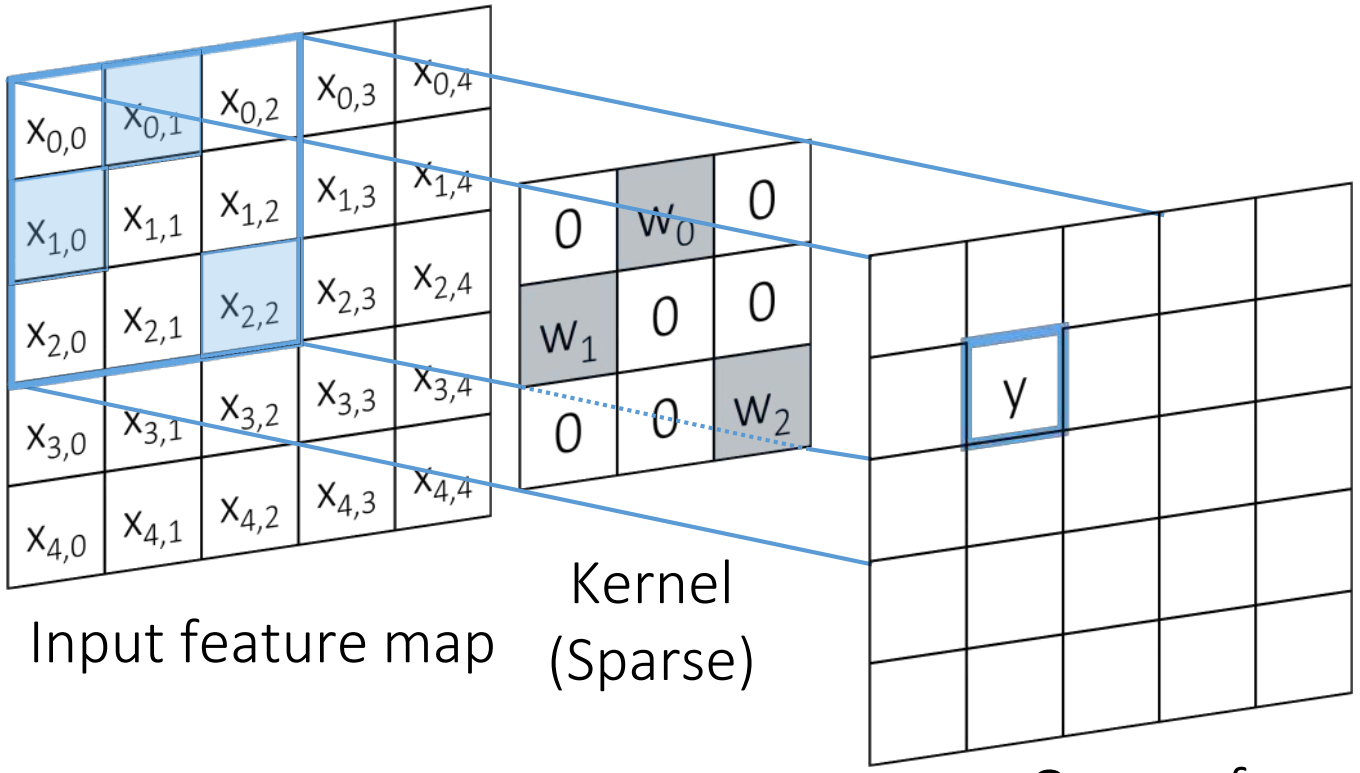
1.9

# Ternary deep neural network accelerator for edge computing

Prof. Hiroki Nakahara

# Ternary-weight convolutional NN

Quantize weight value into three values, -1, 0, 1
Most weights are zero! Can save computational cost.




$$w = \begin{cases} \text{Sign}(w_{hid},) & |w_{hid}| > \rho \\ 0, & \text{Otherwise} \end{cases}$$

$\boldsymbol{\rho}$: Threshold

# Demo: FPGA implementation

- YOLOv2 is implemented to FPGA(Intel Arria10)
- Three times faster than GPU(RTX2018Ti)  with ¼ power

# Our collaborators

TSUBAME3.0

Agency for Science, Technology and Research SINGAPORE

NVIDIA

SONY

DENSO IT LAB

CREST Deep

FUJITSU

R-CCS

AIST

RIKEN

REAL WORLD BIGDATA RWBC-OIL

AIST-Tokyo Tech
Real World Big-Data Computation
Open Innovation Laboratory
(RWBC-OIL)

Fugaku

ABCI