DEEPHi
深 鉴 科 技

DeePhi Tech

Bandwidth-Centric Deep Learning Processing through Software-Hardware Co-Design

Song Yao 姚颂

Founder & CEO

DeePhi Tech 深鉴科技

song.yao@deephi.tech

© 2017 DeePhi Tech. All Rights Reserved.

Outline

– About DeePhi Tech

– Background

– Bandwidth Matters

– Software-Hardware Co-Design Method and Results

– Summary

© 2017 DeePhi Tech. All Rights Reserved.

About DeePhi Tech (深鉴科技)

© 2016 DeePhi Tech. All Rights Reserved.

A Brief History

清華大學 TSINGHUA UNIVERSITY 1911

清華大學 ELECTRONIC ENGINEERING 1952

Early 2013
Research began

DEEPHi
深 鉴 科 技

2016.3
DeePhi founded

金沙江创业投资
GSR Ventures

BANYAN CAPITAL

2016.4
Angel Round

XILINX

MEDIATEK

清华控股
TSINGHUA HOLDINGS

方 和 资 本
SIGMA SQUARE CAPITAL

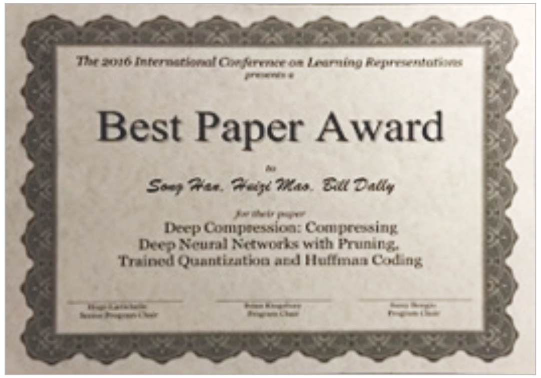
金沙江创业投资
GSR Ventures

BANYAN CAPITAL

Early 2017
A Round

© 2017 DeePhi Tech. All Rights Reserved.

DeePhi Team：Pioneer in Exploring Sparsity for Deep Learning



The 2016 International Conference on Learning Representations
presents a

Best Paper Award

to

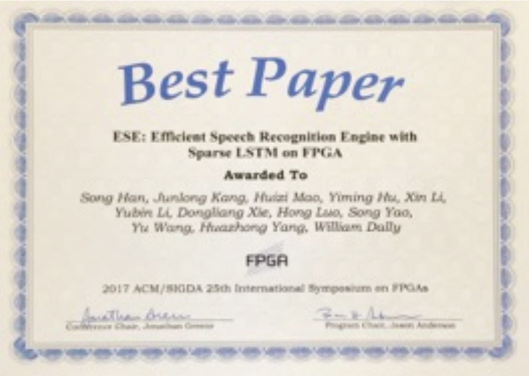
Song Han, Huizi Mao, Bill Dally

for their paper
Deep Compression: Compressing
Deep Neural Networks with Pruning,
Trained Quantization and Huffman Coding

Hugo Larochelle
Senior Program Chair

Brian Kingsbury
Program Chair

Samy Bengio
Program Chair



Best Paper

ESE: Efficient Speech Recognition Engine with
Sparse LSTM on FPGA

Awarded To

Song Han, Junlong Kang, Huizi Mao, Yiming Hu, Xin Li,
Yubin Li, Dongliang Xie, Hong Luo, Song Yao,
Yu Wang, Huazhong Yang, William Dally

FPGA

2017 ACM/SIGDA 25th International Symposium on FPGAs

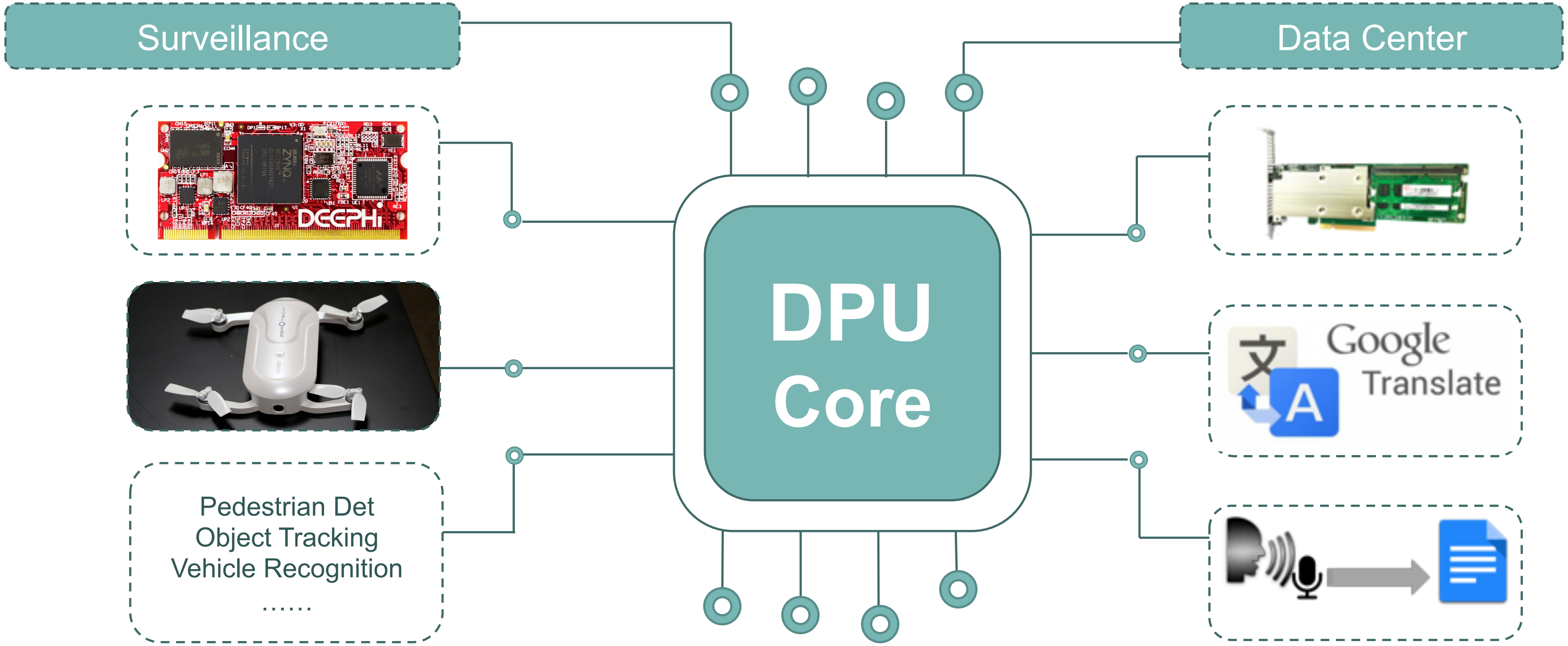
Conference Chair, Jonathan Greene

Program Chair, Jason Anderson

- First Paper in the World on Compressed and Sparse Neural Networks
  - “Learning both Weights and Connections for Efficient Neural Networks”, NIPS 2015
  - “Deep Compression”, ICLR 2016 **Best Paper**
- First Paper in the World on Sparse Neural Network Processor
  - “EIE: Efficient Inference Engine on Compressed Deep Neural Network”, ISCA 2016
- First Practical Case Using Sparse Neural Network Processor
  - Collaboration with Sogou Inc, partly revealed in：
  - “ESE: Efficient Speech Recognition Engine with Compressed LSTM on FPGA”, FPGA 2017 **Best Paper**
- First Full-Stack Development Kit for Sparse Neural Network：DNNDK^{TM}
- Filed Dozens of Patents in both China and the U.S.

© 2017 DeePhi Tech. All Rights Reserved.

Two Solutions for both Embedded Cases and Data Center



NetPosa 东方网力

川大智胜 WISESOFT

ZEROTECH 零度智控

amazon Lab126

amazon web services

有道

XILINX

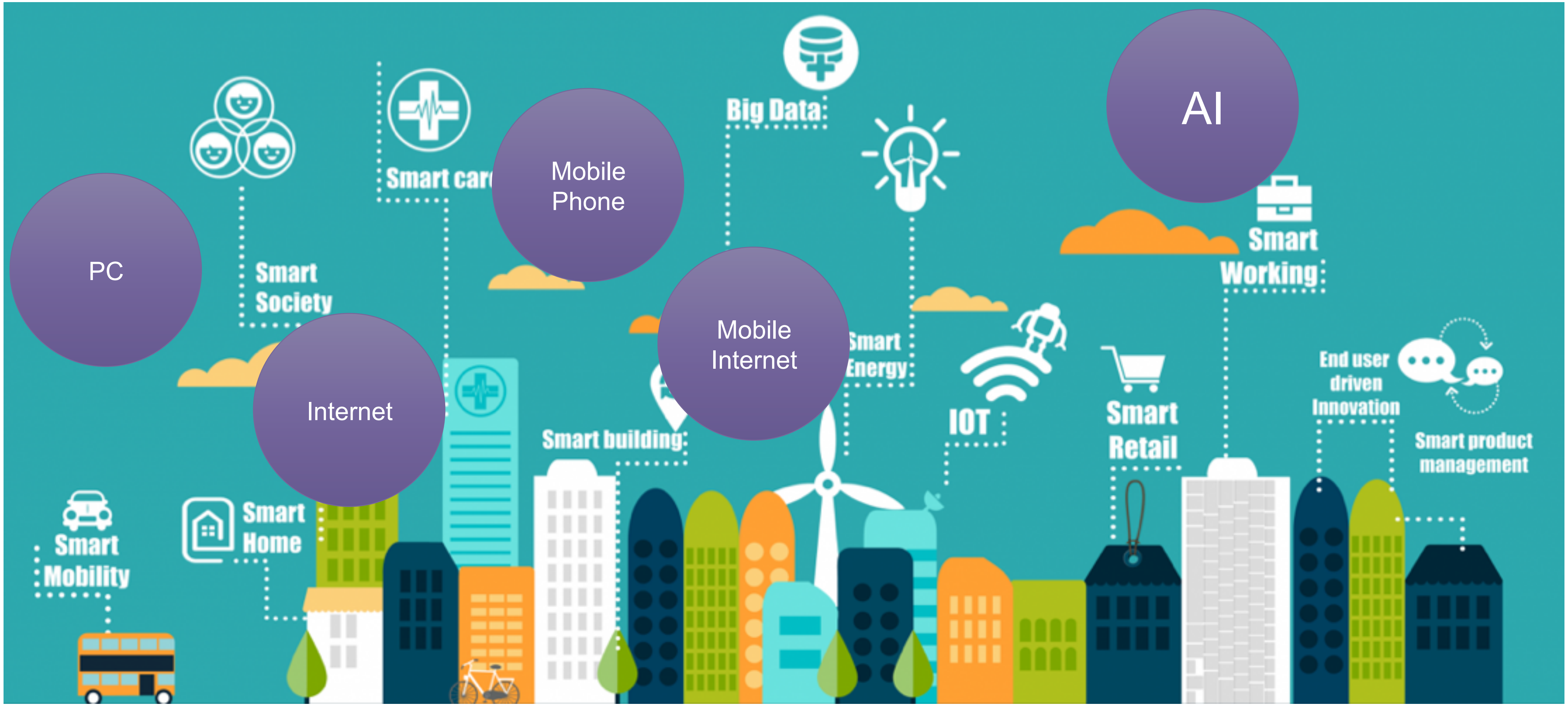
© 2017 DeePhi Tech. All Rights Reserved.

Background

Deep Learning for Everything

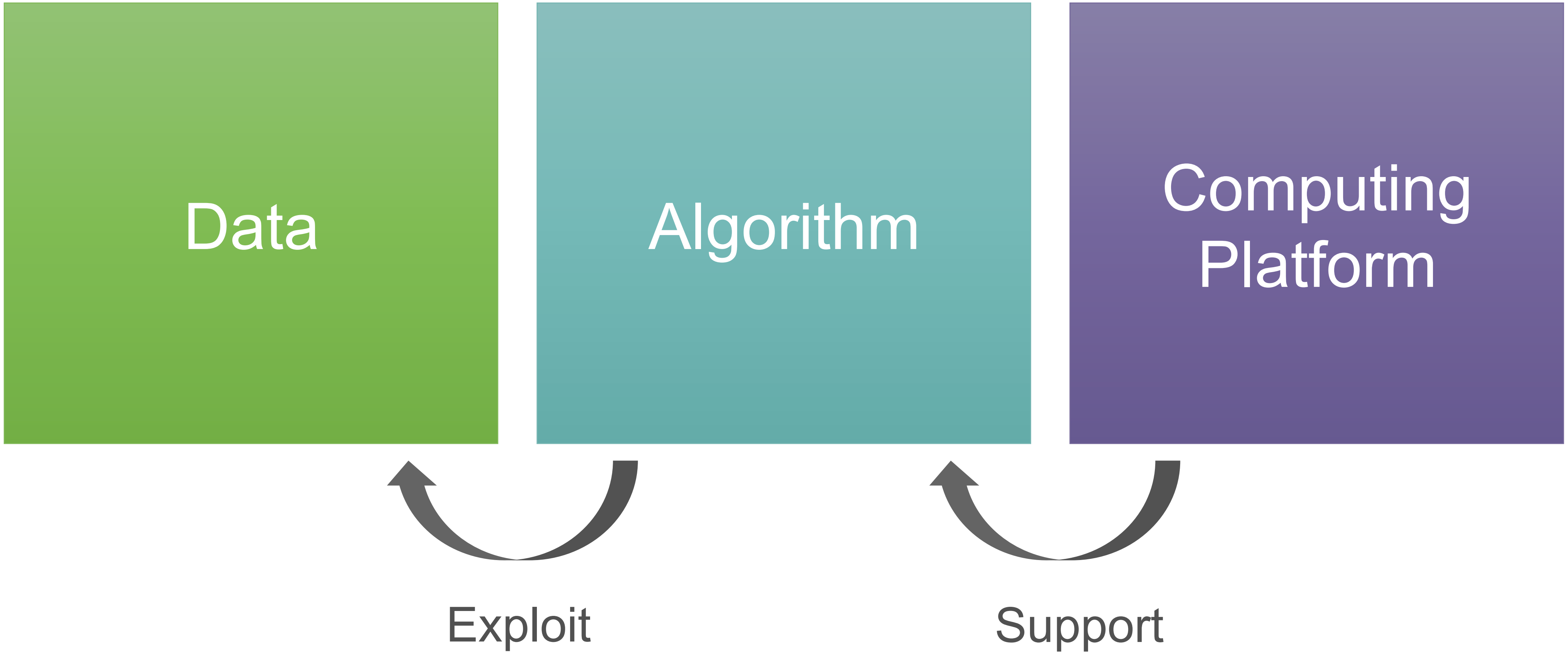
© 2017 DeePhi Tech. All Rights Reserved.

AI: The Future of Human Society



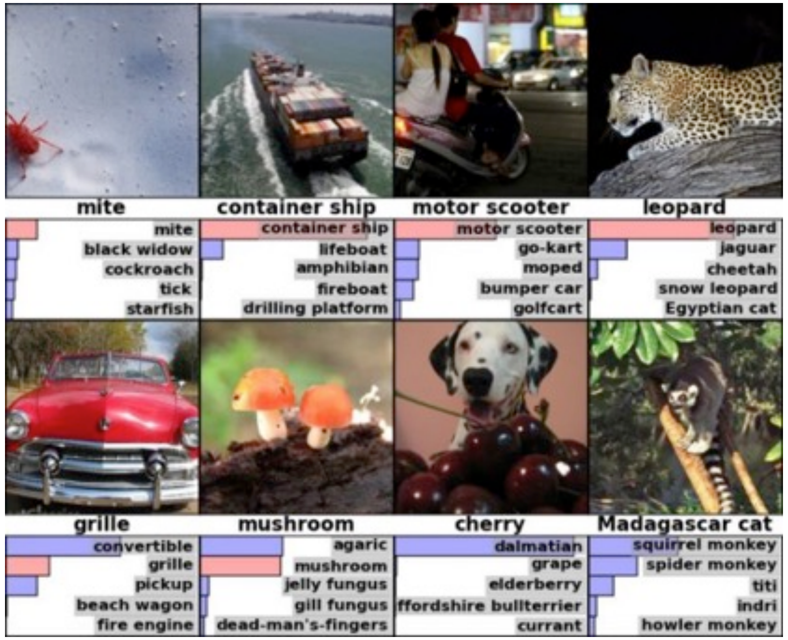
© 2017 DeePhi Tech. All Rights Reserved.

Success of AI = Data + Algorithm + Computing Platform



© 2017 DeePhi Tech. All Rights Reserved.

We Need To Thank GPU



Cat Face Recognition

1000 Servers， 16000 CPU

Andrew Ng & Jeff Dean， Google， 2012

AlexNet for ImageNet

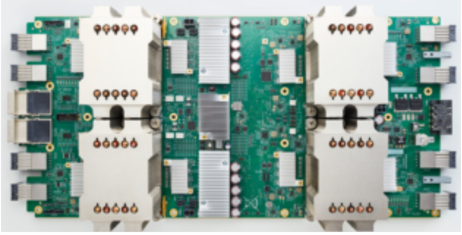
1 Server， 2 GPU

Geoffrey Hinton， 2012

© 2017 DeePhi Tech. All Rights Reserved.

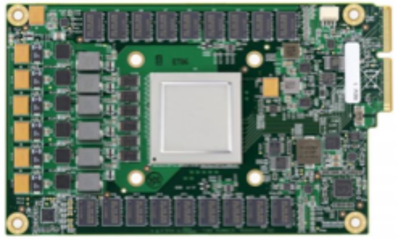
New Computing Platforms are Coming

intel + nervana



Google TPU V2

Microsoft, Amazon: FPGA



Google TPU V1

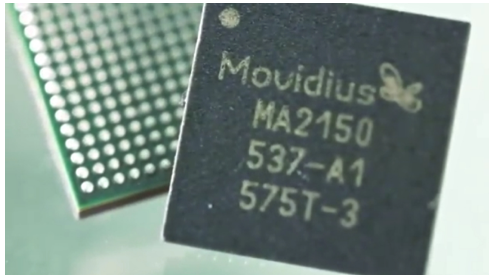
Cloud

Training

Inference

Edge

What is the key factor
in deep learning processing?



Movidius DSP

DeePhi DPU

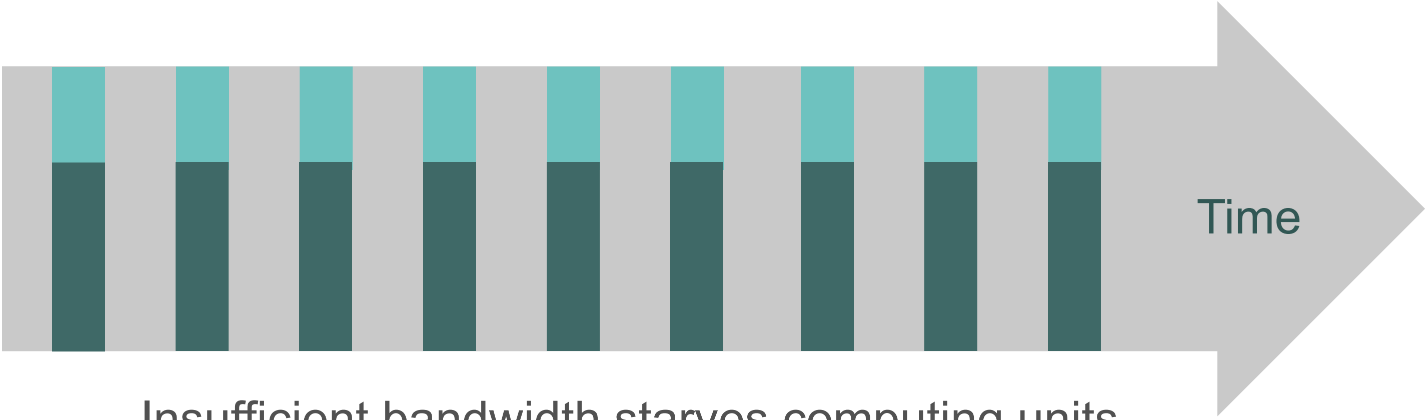
© 2017 DeePhi Tech. All Rights Reserved.

Motivation: Bandwidth Matters

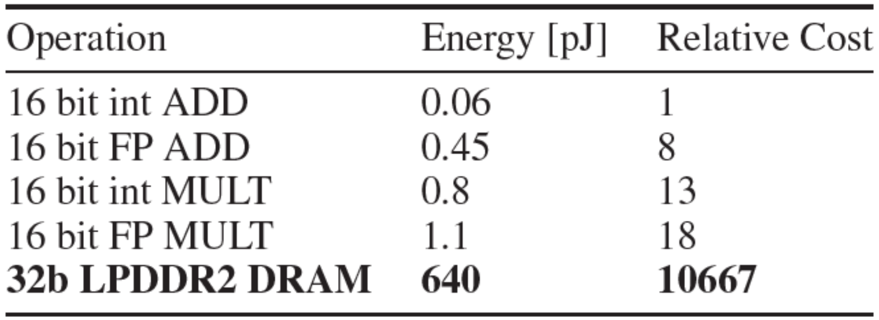
© 2017 DeePhi Tech. All Rights Reserved.

Low Utilization and High Power Come from Memory Access

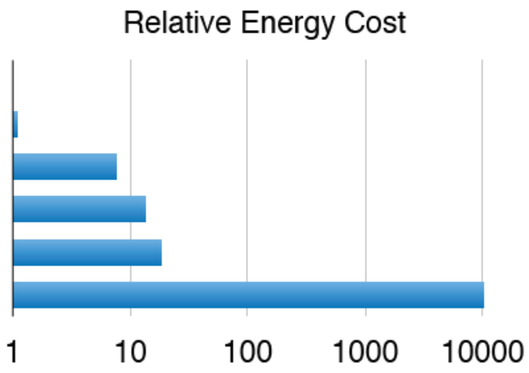
🟩 Utilized computing units



Insufficient bandwidth starves computing units



Operation	Energy [pJ]	Relative Cost
16 bit int ADD	0.06	1
16 bit FP ADD	0.45	8
16 bit int MULT	0.8	13
16 bit FP MULT	1.1	18
32b LPDDR2 DRAM	**640**	**10667**

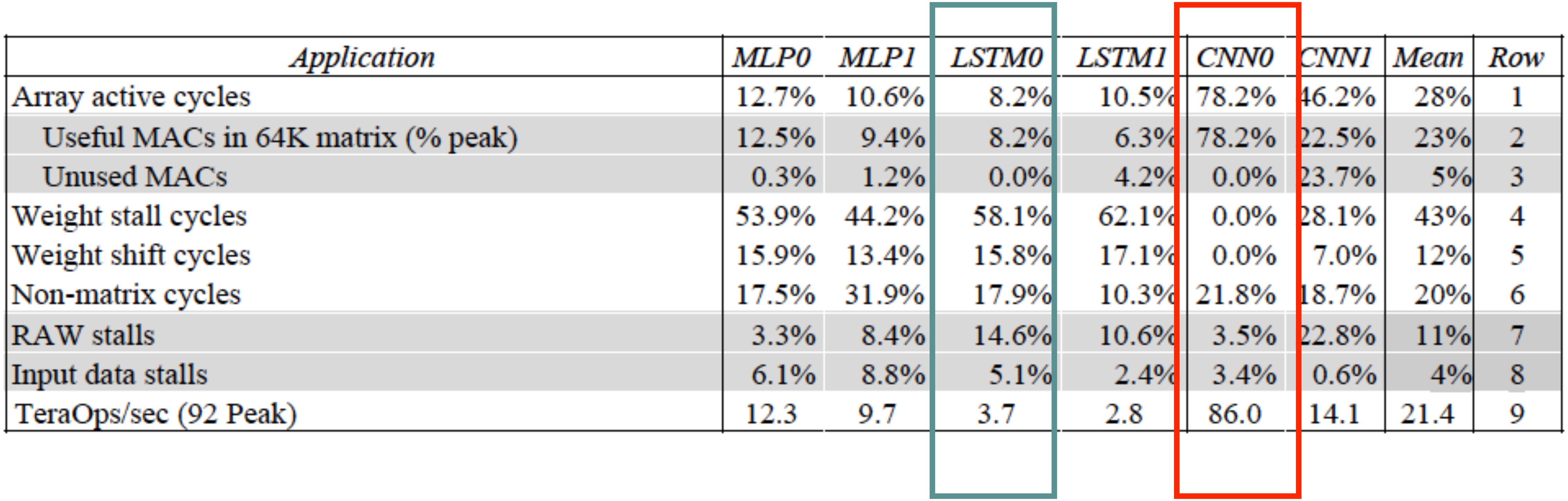


Source： Song Han et al., “Learning…”, NIPS 2015

Too many memory accesses result in high power

© 2017 DeePhi Tech. All Rights Reserved.

Bandwidth Limits Real Performance



Application	*MLP0*	*MLP1*	*LSTM0*	*LSTM1*	*CNN0*	*CNN1*	*Mean*	*Row*
Array active cycles	12.7%	10.6%	8.2%	10.5%	78.2%	46.2%	28%	1
Useful MACs in 64K matrix (% peak)	12.5%	9.4%	8.2%	6.3%	78.2%	22.5%	23%	2
Unused MACs	0.3%	1.2%	0.0%	4.2%	0.0%	23.7%	5%	3
Weight stall cycles	53.9%	44.2%	58.1%	62.1%	0.0%	28.1%	43%	4
Weight shift cycles	15.9%	13.4%	15.8%	17.1%	0.0%	7.0%	12%	5
Non-matrix cycles	17.5%	31.9%	17.9%	10.3%	21.8%	18.7%	20%	6
RAW stalls	3.3%	8.4%	14.6%	10.6%	3.5%	22.8%	11%	7
Input data stalls	6.1%	8.8%	5.1%	2.4%	3.4%	0.6%	4%	8
TeraOps/sec (92 Peak)	12.3	9.7	3.7	2.8	86.0	14.1	21.4	9

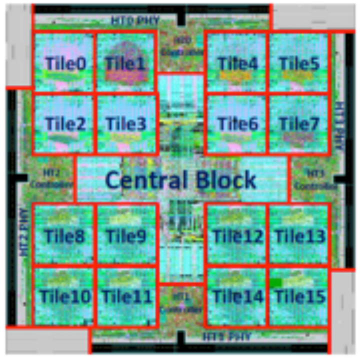
Too large to be stored on chip

Store everything on chip

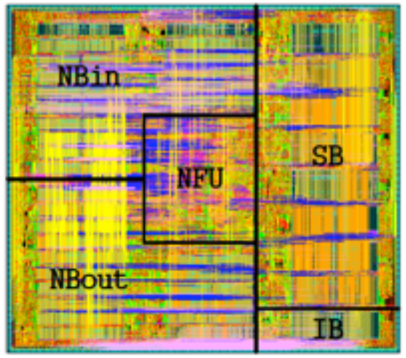
Source: Google TPU paper, ISCA 2017

© 2017 DeePhi Tech. All Rights Reserved.

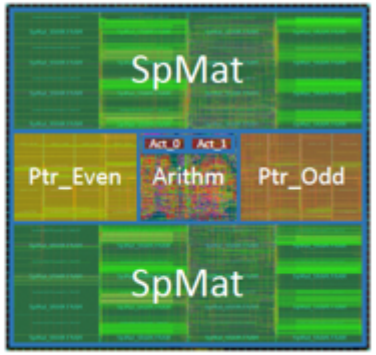
You Can Move Everything on Chip



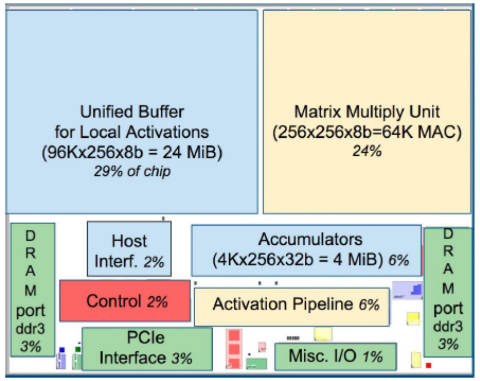
DaDianNao: 36MB eDRAM
Source: Yunji Chen et al., “DaDianNao…”, Micro 2014

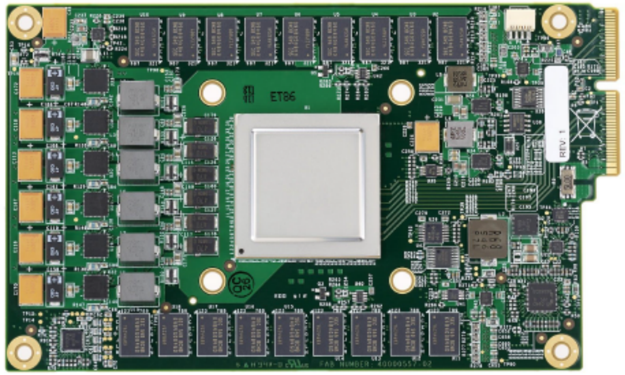


ShiDianNao: 256KB SRAM
Source: Zidong Du et al., “ShiDianNao …”, ISCA 2015



EIE:10.13MB SRAM
Source: Song Han et al., “EIE: …”, ISCA 2016

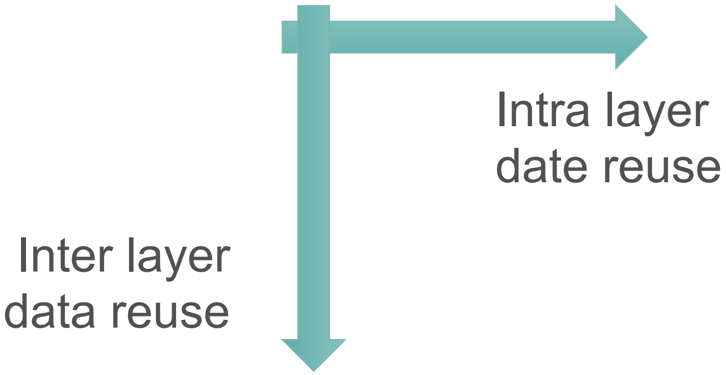




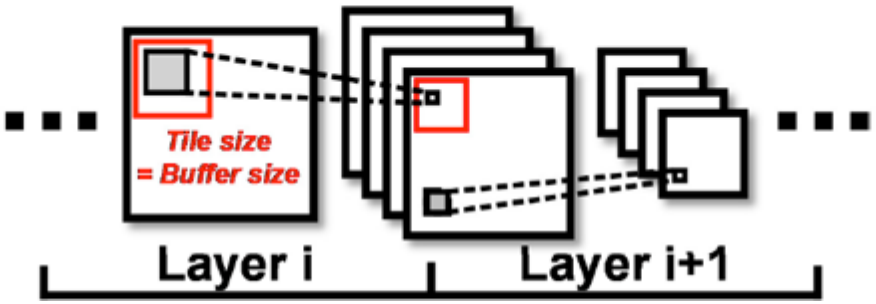
TPU:28MB SRAM
Source: Norman et al., “In-Datacenter: …”, ISCA 2017

© 2017 DeePhi Tech. All Rights Reserved.

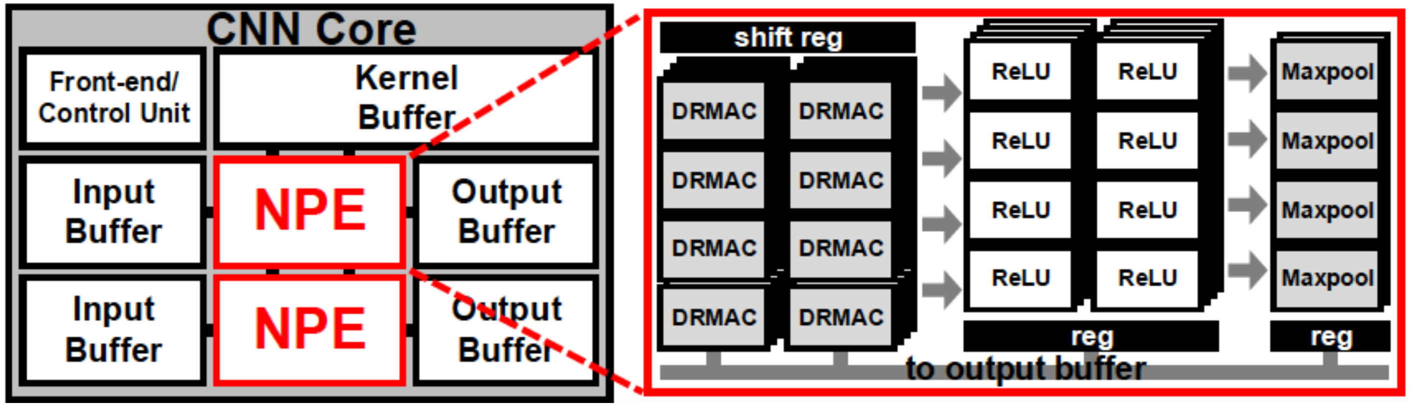
Efficient Architecture Design Exploiting Data Locality


Intra layer date reuse

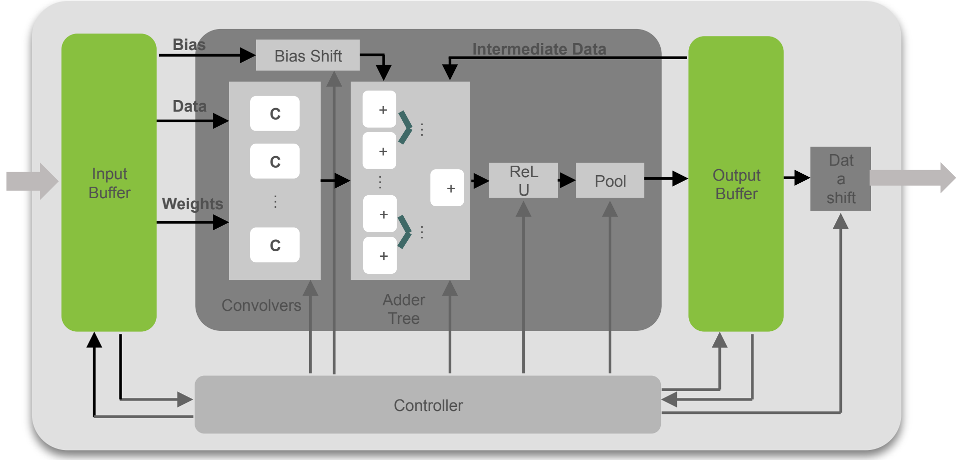
Inter layer data reuse



Jaehyeong Sim et al., “A 1.4TOPS/W…”, ISSCC 2016



Jaehyeong Sim et al., “A 1.4TOPS/W…”, ISSCC 2016



Jiantao Qiu et al., “Going deeper…”, FPGA 2016

© 2017 DeePhi Tech. All Rights Reserved.

You Should Also Combine Software Compression together

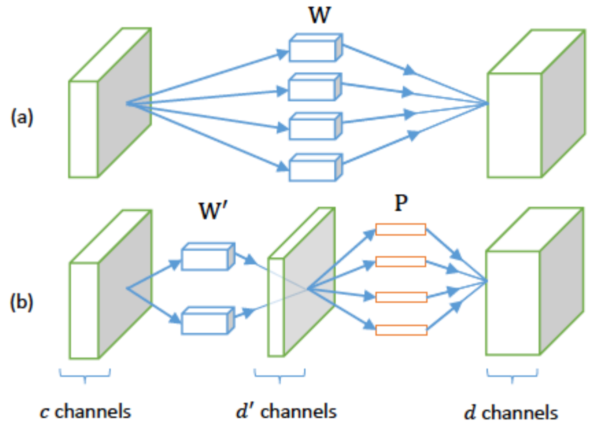
$$F(2,3) = \begin{bmatrix} d_0 & d_1 & d_2 \\ d_1 & d_2 & d_3 \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ g_2 \end{bmatrix} = \begin{bmatrix} m_1 + m_2 + m_3 \\ m_2 - m_3 - m_4 \end{bmatrix} \tag{5}$$

$$m_1 = (d_0 - d_2) g_0 \qquad m_2 = (d_1 + d_2) \frac{g_0 + g_1 + g_2}{2}$$

$$m_4 = (d_1 - d_3) g_2 \qquad m_3 = (d_2 - d_1) \frac{g_0 - g_1 + g_2}{2}$$

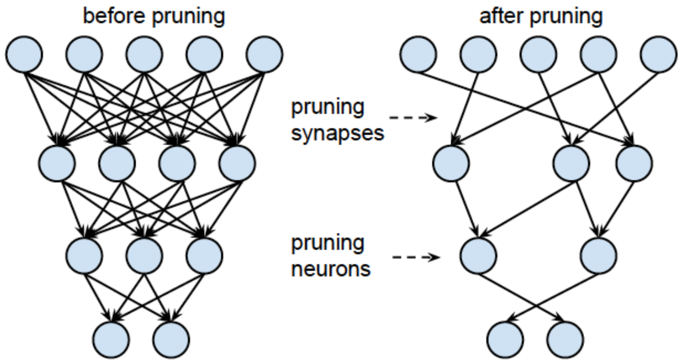
Winograd

Andrew Lavin et al., “Fast…”, arxiv:1509.09308



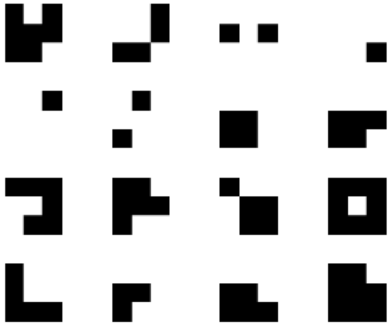
Decomposition like GSVD

Xiangyu Zhang et al., “Effcient…”, arxiv:1411.4229



Deep compression

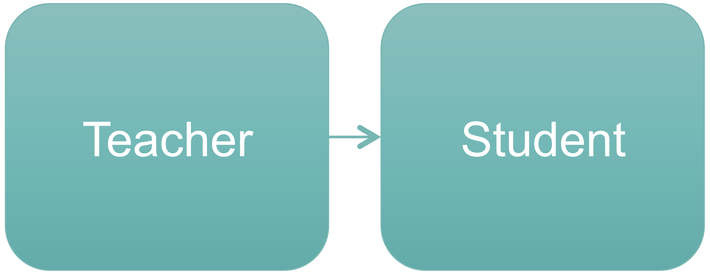
Song Han et al., ”Deep Compression…”, ICLR 2016



Binary Neural Network / XNOR-Net

Matthieu et al., arXiv:1602.02830v3

Mohammad et al., arXiv:1603.05279v1；



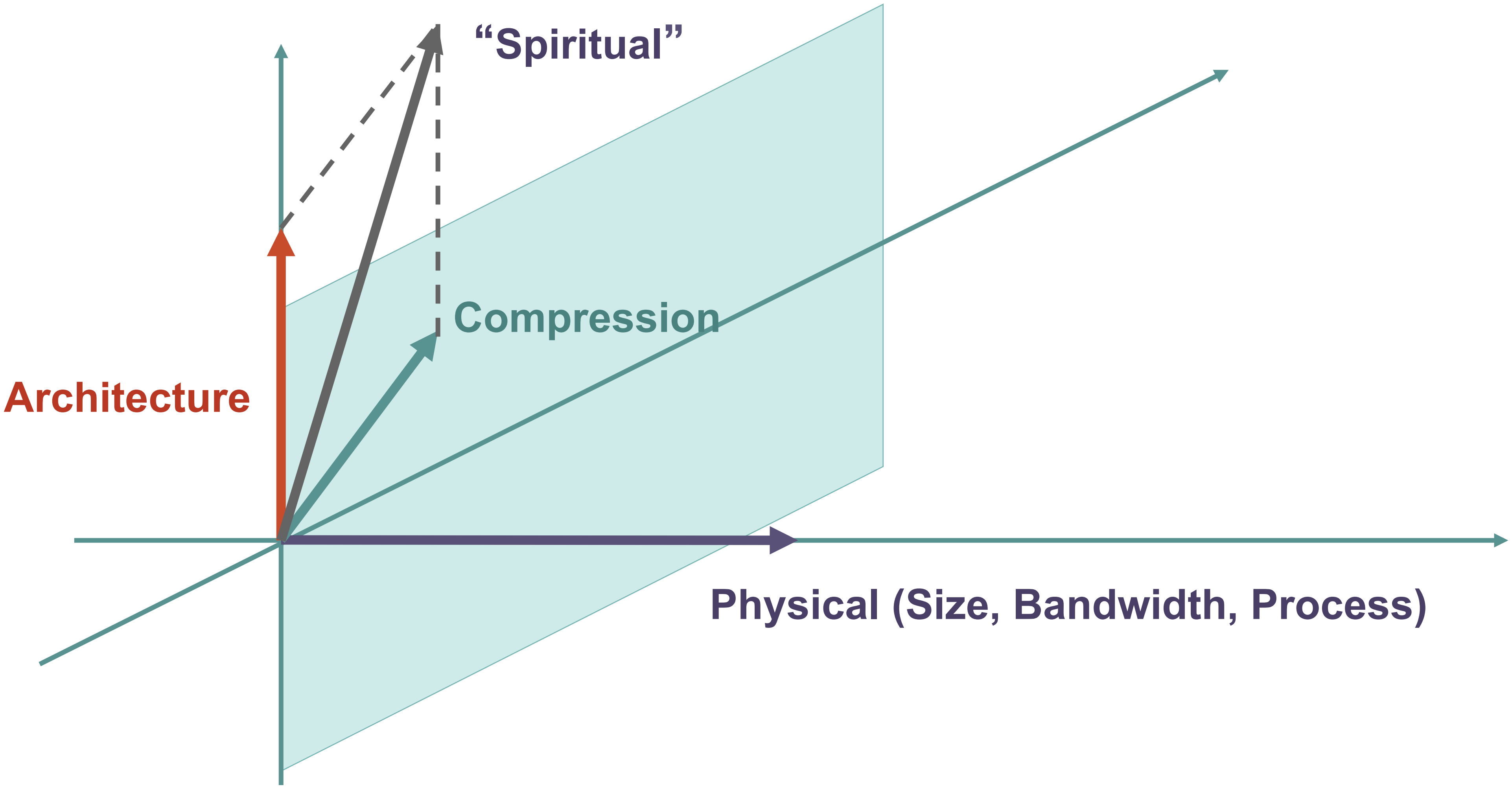
Teacher → Student

Distilling

Geoffrey Hinton et al., arxiv:1503.0253

© 2017 DeePhi Tech. All Rights Reserved.

Considering Software and Hardware Together

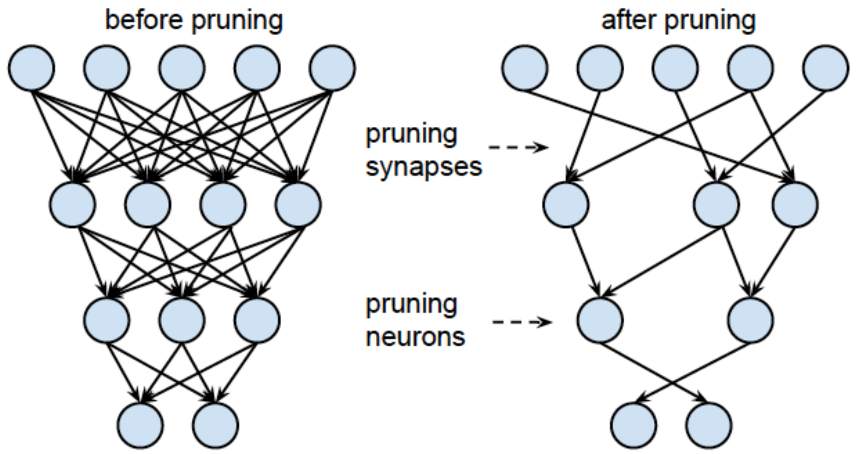


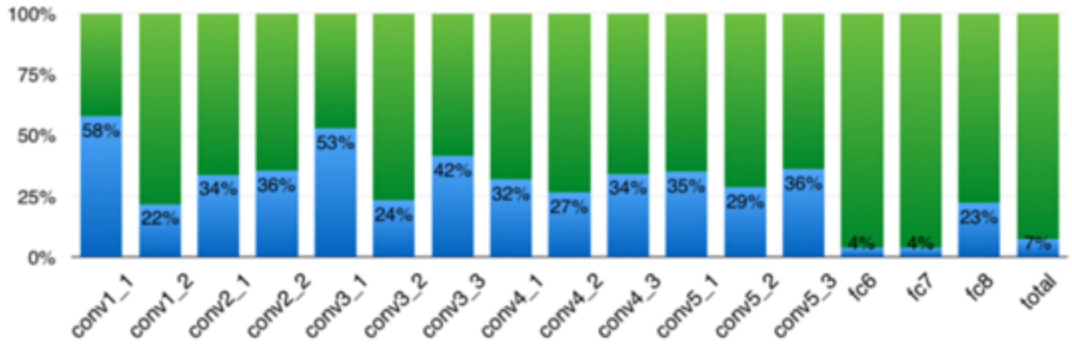
© 2017 DeePhi Tech. All Rights Reserved.

Software-Hardware Co-Design Method

© 2017 DeePhi Tech. All Rights Reserved.

Deep Compression: Make Neural Networks Sparse

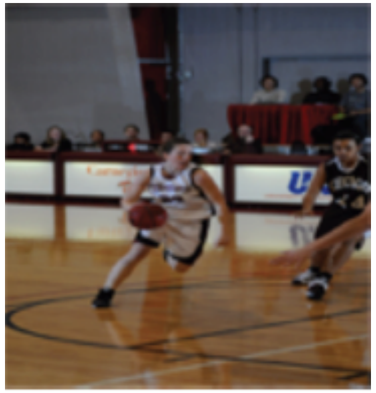




30x – 50x compression rate without hurting accuracy

Before pruning:
A brown dog is running through a grassy field

Prune to 10%:
a brown dog is running through a grassy area



Before pruning:
A basketball player in a white uniform is playing with a ball

Prune to 10%:
A basketball player in a white uniform is playing with a basketball

Not only weights, but also sparsity pattern encodes information

© 2017 DeePhi Tech. All Rights Reserved.

How Sparsity Benefits Computing

Smaller and Faster

Shorter latency in memory read and write

Fewer Computations

High equivalent performance

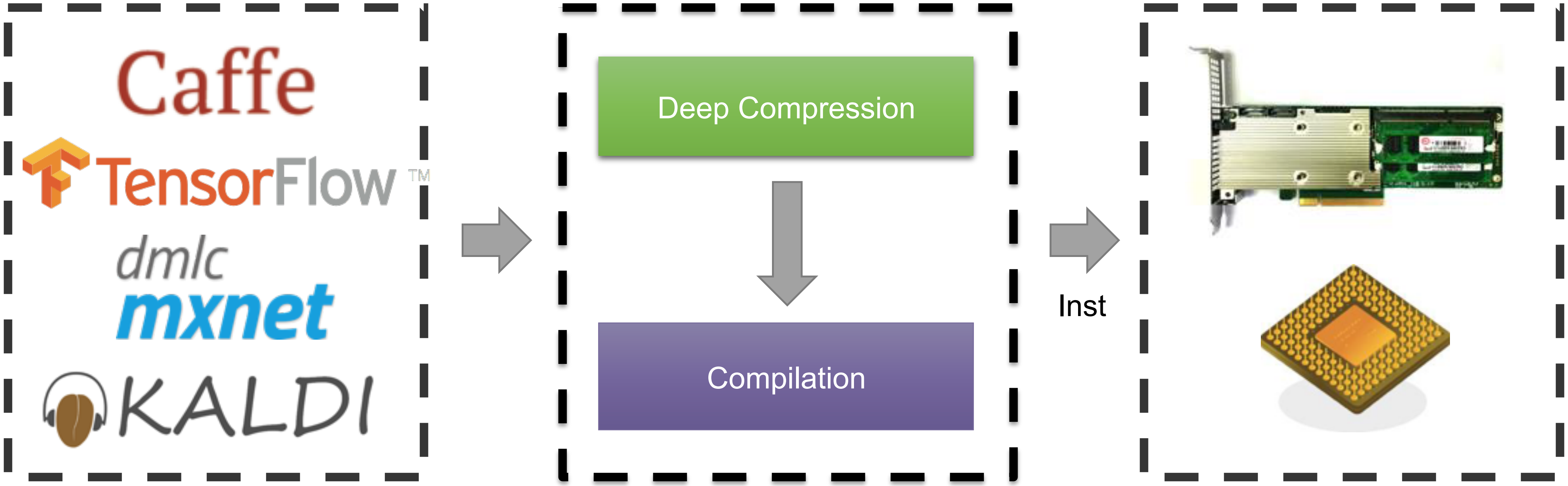
Lower Power

Significantly power reduction in memory operation

> “Sparsity will have high priority in future designs.”
>
> —— Google TPU Paper

© 2017 DeePhi Tech. All Rights Reserved.

Entire Workflow Considering Sparsity



Deep Compression

Compilation

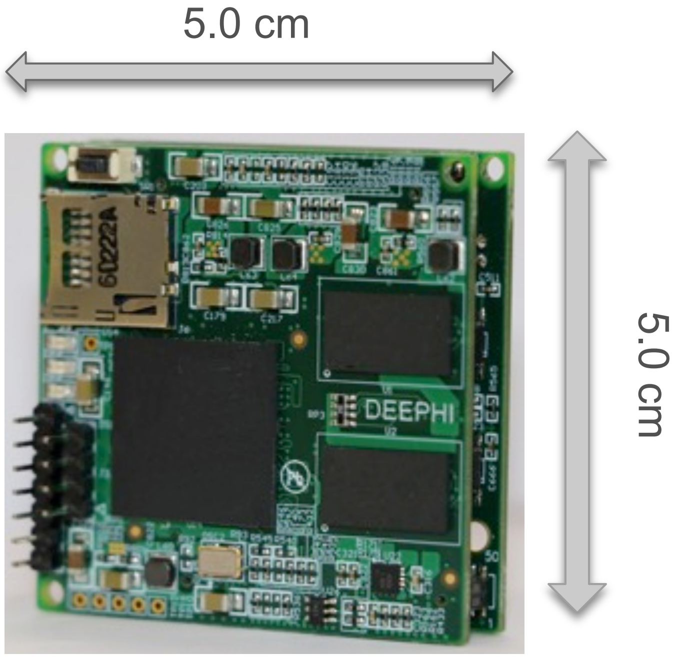
Algorithm
Development

Software
Stack

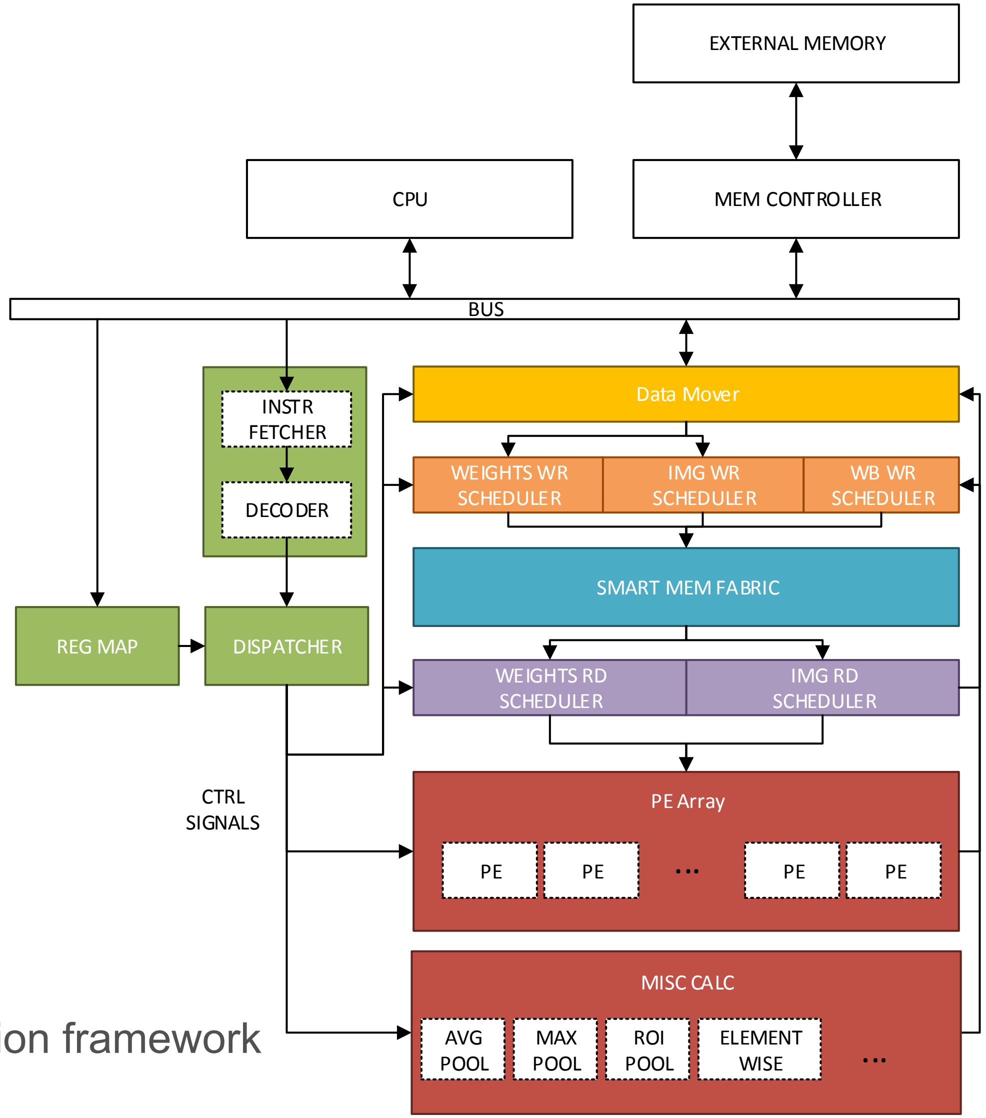
FPGA
ASIC

© 2017 DeePhi Tech. All Rights Reserved.

Aristotle: CNN Processing Platform



- Designed for CNN Acceleration
- Supports all Conv sizes, stride size
- Scalable design (1PE, 2PE, 4PE, 12PE)
- Supports mainstream deep learning object detection framework



EXTERNAL MEMORY

CPU

MEM CONTROLLER

BUS

INSTR FETCHER

DECODER

Data Mover

WEIGHTS WR SCHEDULER

IMG WR SCHEDULER

WB WR SCHEDULER

SMART MEM FABRIC

REG MAP

DISPATCHER

WEIGHTS RD SCHEDULER

IMG RD SCHEDULER

CTRL SIGNALS

PE Array

PE PE ... PE PE

MISC CALC

AVG POOL

MAX POOL

ROI POOL

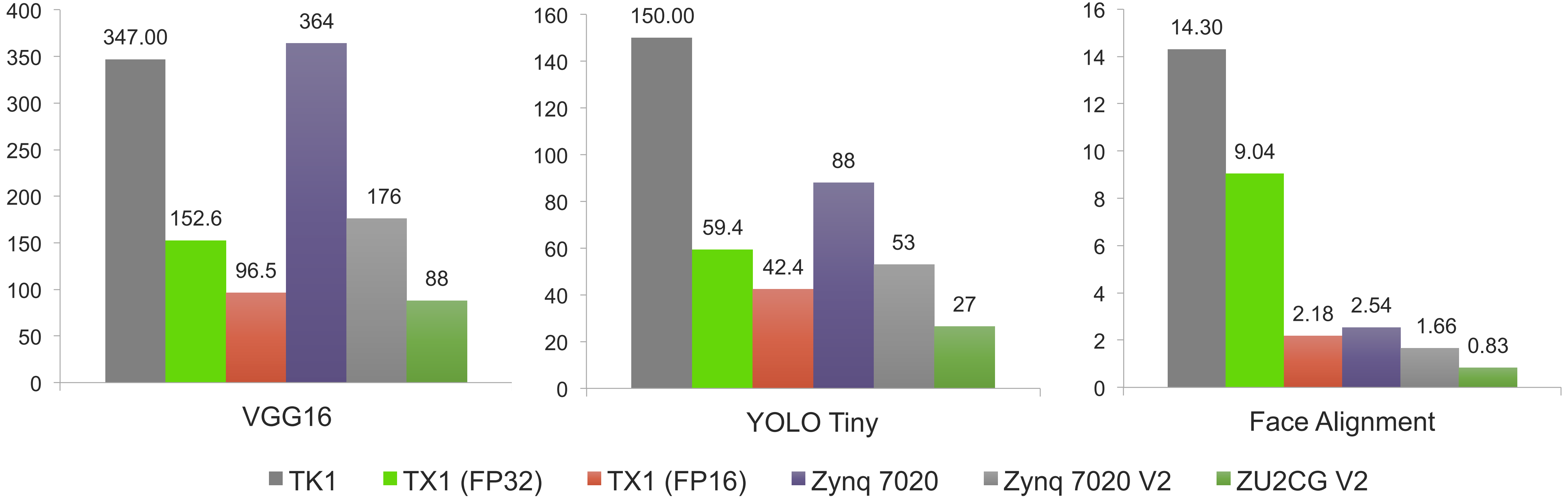
ELEMENT WISE

...

© 2017 DeePhi Tech. All Rights Reserved.

Aristotle (V2) on FPGA: Latency Comparison

Runtime with different network and platform (ms)

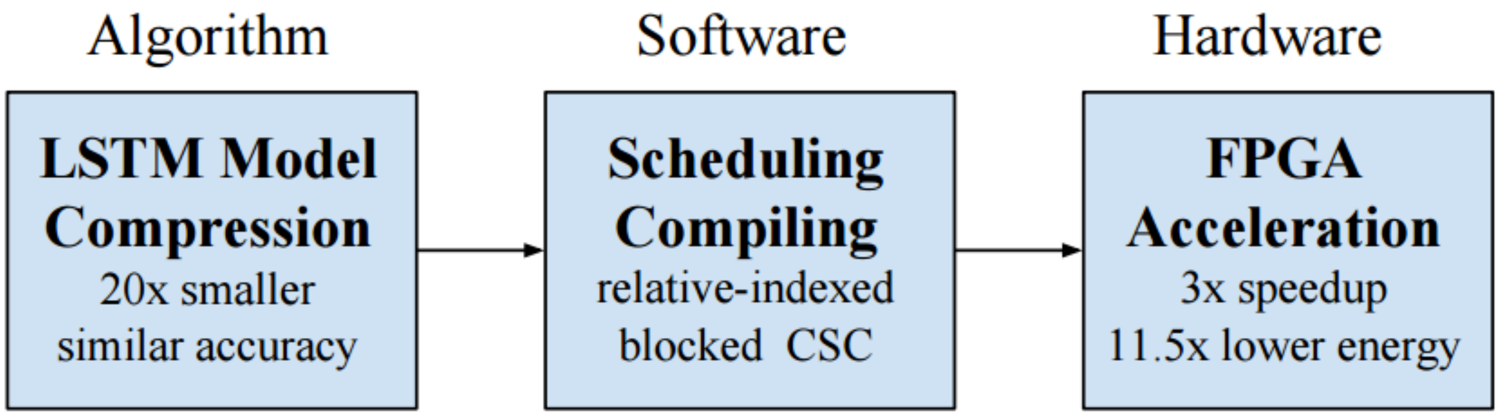


- Aristotle on Zynq 7020: 214 MHz
- Aristotle V2 on Zynq 7020 and ZU2CG：200 MHz and 400 MHz

All results are measured with CuDNN at single batch mode

© 2017 DeePhi Tech. All Rights Reserved.

Descartes: RNN Processing Platform



More computing units	Make full use of all computing units	Equivalent model with fewer computations	Higher bandwidth	Fewer memory accesses	Equivalent model with smaller size
12-bit quantization	Load balance	Pruning	Make full use of BRAM/Pipeline	Dedicated buffer	Pruning + quantization

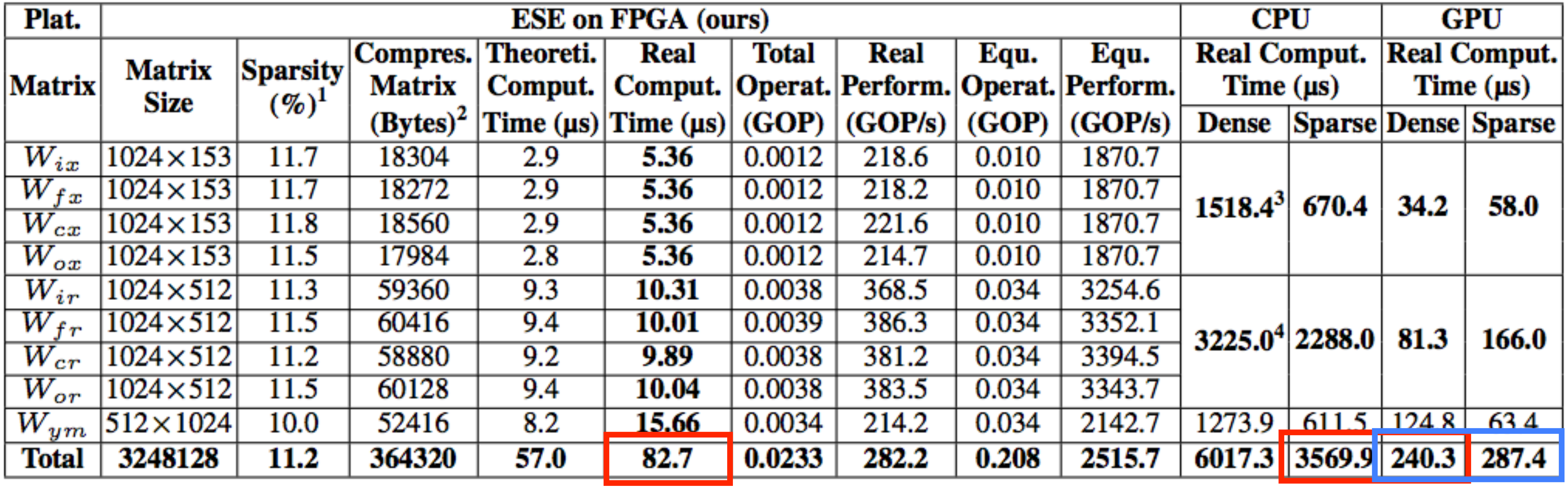
© 2017 DeePhi Tech. All Rights Reserved.

Descartes: Evaluation Platform Comparison

KU060 FPGA	Pascal Titan X GPU
20nm	16nm
4.75MB BRAM	480 GB/s
2 DDR3	12GB GDDR5X
2760 DSP	3584 CUDA Cores
200MHz	1.53 GHz
	CuBLAS/CuSPARSE

© 2017 DeePhi Tech. All Rights Reserved.

Descartes: Performance and Power Comparison



Plat.	ESE on FPGA (ours)									CPU		GPU	
Matrix	Matrix Size	Sparsity (%)[1]	Compres. Matrix (Bytes)[2]	Theoreti. Comput. Time (μs)	Real Comput. Time (μs)	Total Operat. (GOP)	Real Perform. (GOP/s)	Equ. Operat. (GOP)	Equ. Perform. (GOP/s)	Real Comput. Time (μs)		Real Comput. Time (μs)	
										Dense	Sparse	Dense	Sparse
W_{ix}	1024×153	11.7	18304	2.9	**5.36**	0.0012	218.6	0.010	1870.7	**1518.4[3]**	**670.4**	**34.2**	**58.0**
W_{fx}	1024×153	11.7	18272	2.9	**5.36**	0.0012	218.2	0.010	1870.7				
W_{cx}	1024×153	11.8	18560	2.9	**5.36**	0.0012	221.6	0.010	1870.7				
W_{ox}	1024×153	11.5	17984	2.8	**5.36**	0.0012	214.7	0.010	1870.7				
W_{ir}	1024×512	11.3	59360	9.3	**10.31**	0.0038	368.5	0.034	3254.6	**3225.0[4]**	**2288.0**	**81.3**	**166.0**
W_{fr}	1024×512	11.5	60416	9.4	**10.01**	0.0039	386.3	0.034	3352.1				
W_{cr}	1024×512	11.2	58880	9.2	**9.89**	0.0038	381.2	0.034	3394.5				
W_{or}	1024×512	11.5	60128	9.4	**10.04**	0.0038	383.5	0.034	3343.7				
W_{ym}	512×1024	10.0	52416	8.2	**15.66**	0.0034	214.2	0.034	2142.7	1273.9	611.5	124.8	63.4
Total	**3248128**	**11.2**	**364320**	**57.0**	**82.7**	**0.0233**	**282.2**	**0.208**	**2515.7**	**6017.3**	**3569.9**	**240.3**	**287.4**

	KU060@200MHz	Core i7-5930K CPU		Pascal Titan X GPU	
		Dense	Sparse	Dense	Sparse
Performance	3.48	0.048	0.081	1.20	1
Power	41W	111W	38W	202W	136W
Energy Efficiency	11.5	0.057	0.285	0.802	1

© 2017 DeePhi Tech. All Rights Reserved.

Summary

© 2017 DeePhi Tech. All Rights Reserved.

Do It through Software-Hardware Co-Design!

- Bandwidth is the bottleneck
- Methods to solve the problem
  - Move everything on chip
  - Use updated memory technology
  - Design efficient architecture
  - Employ compression on algorithm
- Software-Hardware Co-Design
  - Combining model compression and customized processor
- ~10X energy efficiency compared with GPU

© 2017 DeePhi Tech. All Rights Reserved.

THANKS FOR WATCHING

Song Yao 姚颂
Founder & CEO
DeePhi Tech 深鉴科技
song.yao@deephi.tech

© 2017 DeePhi Tech. All Rights Reserved.