A 400MHz NPU with 7.8TOPS^2/W High-Performance-Guaranteed Efficiency in 55nm for Multi-Mode Pruning and Diverse Quantization Using Pattern-Kernel Encoding and Reconfigurable MAC Units

Zhanhong Tan[1], Sia-Huat Tan[1], Jan-Henrik Lambrechts[1], Yannian Zhang[2], Yifu Wu[2], Kaisheng Ma[1]

[1]Tsinghua University, Beijing, China
[2]IIISCT, Xi'an, China

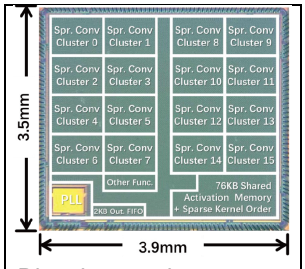
Deep neural networks present a promising future in applications, ranging from face ID on mobile phones to self-driving cars. Weight pruning and quantization act as valuable solutions to release the burden of computation and memory. Figure 1 shows the family of weight pruning, including the fine-grained and several structural pruning methods. With similar compression rates, coarse-grained pruning results in more accuracy drop. A new structural solution called pattern pruning [5] achieves excellent precision with uniform sparsity rates among kernels, which is friendly to hardware. Kernels are encoded into non-zero values with sparse pattern masks (SPM). This work adopts 16 types of patterns with 4b SPM for the 3x3 convolution, which gains up to 8x compression for eight-zero kernels. As for quantization, the optimal choice generally depends on models.

To support various quantization and pruning methods, including the new pattern granularity, we clarify the following challenges: 1) For pruning, the coarse-grained channel- and filter-level can be easily implemented by the layer width reconfiguration. However, the kernel-level leads to an unbalanced number of kernels in different filters, thus reducing resource utilization. Pattern pruning is a brand-new way that requires an efficient hardware design with a particular data-encoding and computation design that is also compatible with other pruning. However, prior works only focused on a single condition like unstructured sparsity [3] or relatively coarse-grained pruning [4] but did not cover multiple modes. 2) For quantization, resources are expected to be reused as much as possible for linear and nonlinear quantization. UNPU [2] studied various bit-width of weights but not for diversity. Consequently, there lacks a unified architecture facing to optimize diverse types of pruning and quantization.

We introduce a 55nm 400MHz 8.1TOPS/W versatile NPU with the following features: 1) Efficient processing engines (PE) and the input channel scheduler (ICS) supporting diverse pruning via pattern-kernel encoding; 2) An algorithm-enabled data layout for kernel pruning achieving better computation utilization; 3) Unified MAC units supporting 4b weight with linear and nonlinear quantization in a multiplier-free fashion that saves 30.7% power and 3.5% area.

Figure 2 shows the overall NPU architecture and processing flow for pattern and kernel pruning. The proposed architecture mainly consists of 64 PEs to process 64 output channels in parallel. Each PE comprises a sparsity interface (SI) and a pattern-based calculation core (PCC) that exploits the weight and activation sparsity. Each PE is interconnected with an 8-entry input feature map (IFM) FIFO and a 2.3KB weight buffer bank. ICS issues and multicasts input channels to PEs with order indices and masks, only requiring 6.8% extra memory of the 75KB activation buffer. These scheduling orders are generated from a software flow following the group-wise optimized kernel pruning. For the weight pre-processing shown at the bottom-right, the 4b weights are encoded in a pattern-kernel fashion: for pattern pruning, weights within a kernel are compressed in pattern-based format with a 4b SPM and non-zero values; for kernel pruning, kernels in each bank are rearranged to match the input channel order. Two pruning methods are compatible since the pattern-kernel encoding is decoupled by inter/intra-kernel.

Figure 3 depicts the high-performance and pattern-based sparsity-aware PE module with full pipeline design. The kernel in pattern-kernel format is decoded through the pattern decoder and restored to a full 9-weight kernel with a 9-bit mask. The mapping relationship between SPM and weight sparsity mask is stored in a 16-entry LUT reconfigured for each layer. An activation tile is pipelined into the local IFM tile register file, and simultaneously, a sparsity mask for activations is also produced. With sparsity masks, pair-masks for MAC units are generated by the AND logic, followed by a pointer generator to locate all the non-zero activation-weight pairs with the bisection-method. The sparsity rates across kernels are uniform in pattern pruning. The proposed group-wise kernel pruning attains balanced workloads for multicast. With a more uniform workload distribution profit from the optimized pruning on our hardware, the PE idle time is reduced by 27.3%~35.9%.



Die micrograph.

Figure 4 presents a high PE-utilization kernel pruning optimization method. The number of pruned kernels in different filters varies randomly for the regular kernel pruning, leading to unbalanced workloads. To tackle this problem, we propose a group-wise manner that each group of output channels shares the same set of input channels, and the number of connected input channels of all groups is restricted as the same. Evaluations under three models with 50% group-wise kernel pruning on CIFAR10 show an accuracy drop of less than 2%. To further improve PE utilization, we rearrange scheduling using the flow shown in figure 2, aiming to prevent PEs from being idle for too long. Results show that this policy efficiently skips ineffectual kernels, achieving up to 2.83x performance improvement.

Figure 5 presents the unified MAC unit to maximize resource reuse in different quantization manners. For the linear mode (LIN), we implement the multiplication via accumulating three parts (1×A, 2×A, 4×A). Two small shifters calculate 2×A and 4×A. Three weight bits serve as control signals to decide the effectual parts. For the single power-2 quantization (POW), only one shifter is activated to generate 2^0~2^6, and we utilize W[2:0] of 3'b111 as the code of zero. For the mixed power-2 quantization (MIX), two small adders leverage (α, β) as adjustment factors to fine-tune two power-2 levels. Both shifters generate results of two parts. Three modes reuse the same 28b accumulator to complete the final results. In summary, this unified MAC unit for 4b weight calculation replaces the traditional decoder and arithmetic multiplier with a few small adders and two shifters, which saves 30.7% power and 3.5% area without performance loss.

Figure 6 shows the measurement results of our NPU implemented in UMC CMOS 55nm technology, operating at 0.75-to-1.00V with 26-to-400MHz frequency. VGG-16 and ResNet-18, which employed different quantization and pruning methods with guaranteed accuracy, are evaluated by the NPU at 1.0V and 300MHz. The average performance and energy efficiency of convolution layers are 104.7-to-792.9GOPS and 1.07-to-3.64TOPS/W. We evaluate the peak energy efficiency using different sparsity scaling and show that this work achieves up to 8.1TOPS/W efficiency. Utilizing 4b weight quantization combined with pattern and kernel pruning, our NPU saves up to 91.7x weight storage. Compared to prior works that usually compromised performance for the peak efficiency [1~4], our NPU achieves the highest performance-guaranteed efficiency of 8.5TOPS^2/W (peak-efficiency x the-corresponding-performance). This newly defined metric is profound, aiming to guarantee that the NPU can achieve high performance and high efficiency at the same time. Besides, this work is the first one supporting inter/intra-kernel sparsity and linear/nonlinear quantization in a unified architecture with high performance and efficiency.

References:

[1] D. Shin, *et al*., "DNPU: An 8.1 TOPS/W Reconfigurable CNN-RNN Processor for General-Purpose Deep Neural Networks," ISSCC, Feb. 2017.

[2] J. Lee *et al*., "UNPU: A 50.6TOPS/W Unified Deep Neural Network Accelerator with 1b-To-16b Fully-Variable Weight Bit-Precision," ISSCC, Feb. 2018.

[3] C. Lin *et al*., "A 3.4-to-13.3TOPS/W 3.6TOPS Dual-Core Deep-Learning Accelerator for Versatile AI Applications in 7nm 5G Smartphone SoC," ISSCC, Feb. 2020.

[4] J. Yue *et al*., "A 65nm Computing-in-Memory-Based CNN Processor with 2.9-to-35.8TOPS/W System Energy Efficiency Using Dynamic-Sparsity Performance-Scaling Architecture and Energy-Efficient Inter/Intra-Macro Data Reuse," ISSCC, Feb. 2020.

[5] X. Ma *et al*., "PCONV: The Missing but Desirable Sparsity in DNN Weight Pruning for Real-Time Execution on Mobile Devices," AAAI, Feb. 2020.

978-1-7281-7581-2/21/$31.00 ©2021 IEEE

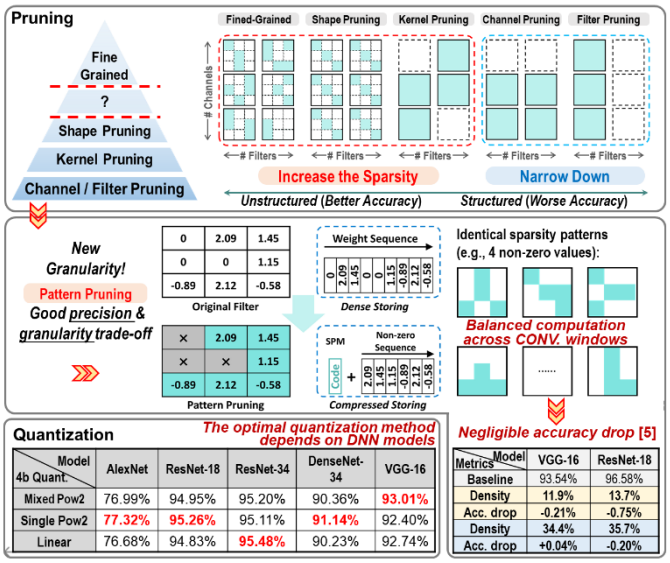


Fig. 1. The pruning family with a new promising pattern pruning and different optimal quantization across models.

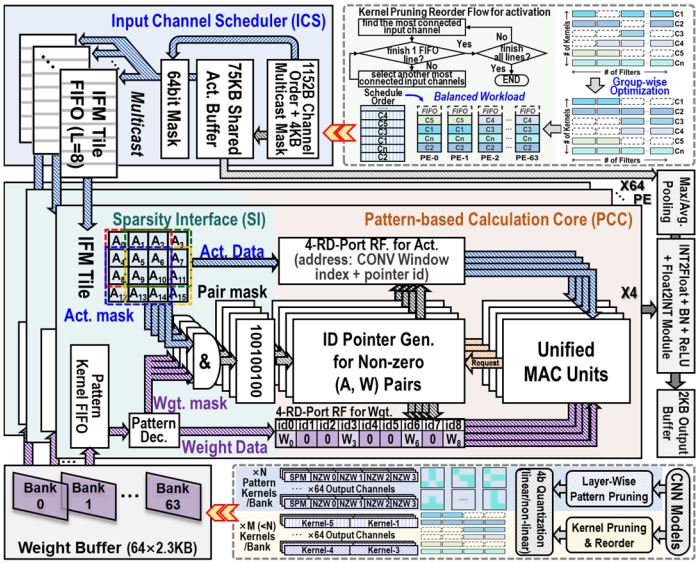


Fig. 2. Overall architecture and weight (bottom-right) & activation (top-right) pre-processing flow for pattern and kernel pruning.

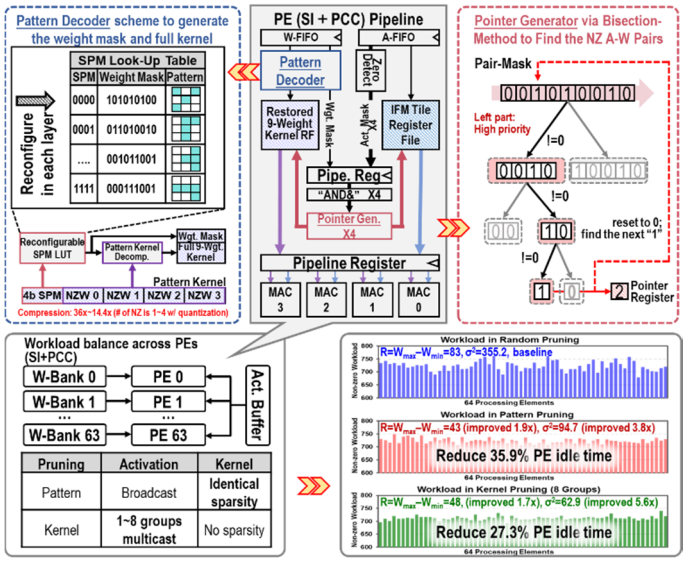


Fig. 3. Sparsity interface (SI) and pattern-based calculation core (PCC) for sparse computation.

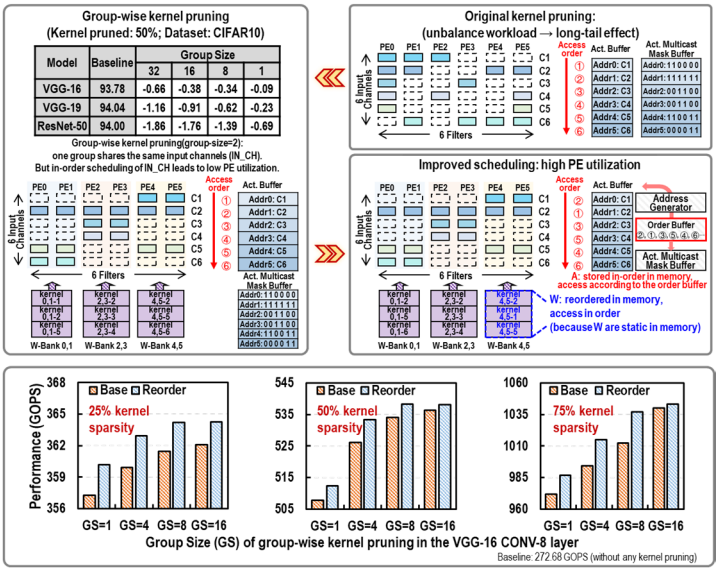


Fig. 4. The input channel scheduler (ICS) scheme for PE utilization optimization via group-wise kernel pruning and reordering.

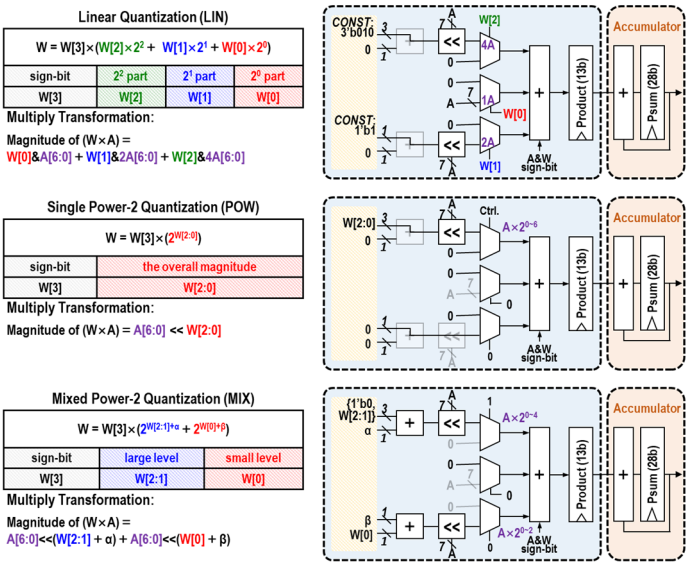
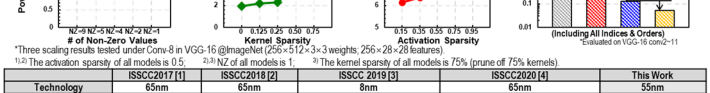


Fig. 5. Unified MAC unit for 4bit linear (LIN), single power-2 (POW), mixed power-2 (MIX) quantization for weights.

Model	VGG-16 (pre-trained acc: 92.85%)					VGG-16 (80.5%)	ResNet-18 (pre-trained acc: 94.57)				
Quantization	MIX	POW	LIN	MIX	MIX	MIX	MIX	POW	LIN	POW	POW
Dataset	CIFAR10					ImageNet100	CIFAR10				
Pruning	NZ=1	NZ=1	NZ=1	0.5 kernel pruning (GS=8) + NZ=1	0.75 kernel pruning (GS=8) + NZ=1	0.75 kernel pruning (GS=8) + NZ=1	NZ=1	NZ=1	NZ=1	0.5 kernel pruning (GS=8) + NZ=1	0.75 kernel pruning (GS=8) + NZ=1
Model Compression	x72	x72	x72	X144	X287	X287	x72	x72	x72	x144	X287
Top-1 Accuracy (%)	93.01	92.40	92.74	91.32	86.58	71.5	94.77	94.63	94.65	93.54	90.13
Average Conv. Layer Performance (GOPS)	104.8	104.7	104.8	143.6	294.2	792.9	138.0	162.1	136.5	174.4	205.9
Average Conv. Layer Efficiency (TOPS/W)	1.1	1.1	1.1	1.6	3.3	3.6	1.4	1.4	1.3	2.0	2.4
Latency (ms)	10.8	10.8	10.8	6.6	3.0	60.5	14.1	13.5	14.1	9.9	7.7
Power (mW)	93.3	91.9	92.2	88.6	87.8	162.9	101.5	110.5	100.8	88.9	88.0

*NZ = 1: only one Non-Zero (NZ) value in a kernel for pattern pruning; GS = 8: the Group Size (GS) in our group-wise kernel pruning.



*Three scaling results tested under Conv-8 in VGG-16 @ImageNet (256×512×3×3 weights; 256×28×28 features).
[1],[2] The activation sparsity of all models is 0.5; [2],[3] NZ of all models is 1; [3] The kernel sparsity of all models is 75% (prune off 75% kernels).

	ISSCC2017 [1]	ISSCC2018 [2]	ISSCC 2019 [3]	ISSCC2020 [4]	This Work
Technology	65nm	65nm	8nm	65nm	55nm
Chip Area (mm²)	16	16	5.5	12	13.65
Max. Frequency (MHz)	200	200	933	100	400
SRAM (KB)	290	256	1568	196	243
Weight Precision (bit)	4, 8, 16	1-16	8, 16	8	4
Sparsity Optimization	No	Skip loading zero Act.	Skip zero weights	Empty channel	Multi-mode pruning + activation-sparsity-aware
Weight Quantization Type	Q-Table	Linear	Linear	Linear	Linear, single power-2 mixed power-2
Performance (GOPS) (with similar bit-width)	1200 (Act.:8b; Weight:4b)	1382 (Act.:16b; Weight:4b)	6937 (Act.:8b; Weight:8b w/ 75% zeros)	32.8 (Act.:4/8b; Weight: 8b)	[1] 1407.2 (Act.8b; Weight:4b)
Power Consumption (mW)	34.6 @50MHz, 0.77V 279 @200MHz, 1.1V	3.2 @5MHz, 0.63V 297 @200MHz, 1.1V	39 @67MHz, 0.5V 1553 @933MHz, 0.8V	7.3 @25MHz, 0.51V 99 @100MHz, 1.00V	20 @26MHz, 0.75V 212.7 @400MHz, 1.00V
Peak Energy Efficiency (TOPS/W)	8.1 (Act.:8b; Wgt.:4b) @50MHz, 0.77V	11.6 (Act.:16b; Wgt.:4b) @5MHz, 0.66V	11.5 (Act.:8b; Wgt.:8b) @67MHz, 0.5V	13.3 (Act.:4/8b; Wgt.: 8b w/ 85% zeros) @50MHz, 0.90V	[2] 8.1 (Act.:8b; Wgt.:4b) @300MHz, 0.97V
[3] Performance-Guaranteed Efficiency (TOPS²/W)	2.3	0.4	5.2	5.6	8.5

[1] For VGG-16 model Conv-8 on ImageNet with 1-pattern & 75% kernel pruning and 95% activation sparsity operating at 400MHz 1.0V.
[2] For VGG-16 model Conv-11 on ImageNet with 1-pattern & 75% kernel pruning and 95% activation sparsity. Off-chip/IO power not included, but time included.
[3] **The Performance-Guaranteed Efficiency aims to avoid compromising the performance for the peak energy efficiency.**
Performance-Guaranteed Efficiency (TOPS²/W) = Peak energy efficiency (TOPS/W) × the corresponding performance (TOPS)
* 2OP=1 equivalent MAC=1 Mul+1 Add. The presented performance/efficiency is computed with the OP of original models and energy/time of actual execution.

Fig. 6. Measurement results and the comparison table.