A 141 uW, 2.46 pJ/Neuron Binarized Convolutional Neural Network based Self-learning Speech Recognition Processor in 28nm CMOS

Shouyi Yin[1], Peng Ouyang[2], Shixuan Zheng[1], Dandan Song[1], Xiudong Li[1], Leibo Liu[1], Shaojun Wei[1],
[1]Tsinghua University, [2]Beihang University, Beijing, China, yinsy@tsinghua.edu.cn

Abstract

An ultra-low power speech recognition processor is implemented in 28 nm CMOS technology, which is based on an optimized binary convolutional neural network (BCNN). A tailored self-learning mechanism is implemented to learn the features of users and improve recognition accuracy on the fly. Measurement results show that this processor supports real time speech recognition with power consumption of 141 uW and energy efficiency of 2.46 pJ/Neuron when working at 2.5 MHz, while achieving at most 98.6% recognition accuracy.

Introduction

Always-on speech interfaces are becoming prevailing in human-machine interaction, especially for wearable devices, Internet of Things, *etc*. Ultra-low power and real-time processing are critical for those battery-powered devices. Deep neural networks (DNN) has achieved great success on speech processing, but its massive parameters and computation produce too much power consumption. BCNNs has the potential to be used in speech recognition to reduce storage overhead and power consumption. But it results in roughly 10.0% relative accuracy loss [1]. In this paper, a BCNN based speech recognition processor is proposed. To achieve ultra-low power consumption, optimized computation flow and approximate computing units are employed. To compensate the accuracy loss, an on-chip self-learning mechanism is designed to incrementally update the weights based on user's speech input over time. This processor supports wake-up words detection, voice commands recognition and continuous speech recognition with peripheral decoder, as illustrated in Fig.1.

Overall Architecture

Fig.2 shows the top-level architecture, which consists of energy-based voice activity detection (VAD) unit, reconfigurable data-path for feature extraction and on-chip self-learning, BCNN unit, smoothing unit, and the main controller. The weights and activations of BCNN are totally stored on chip, using 37 KB weight memory and 4 KB shared data memory, respectively. The digital speech signals enter the VAD unit serially, and are accumulated to activate the downstream units. By configuring the datapath, a 512-point FFT-based filter bank (Fbank) is used to extract 40-dimension features. BCNN with 4 convolutional (CONV) layers and 3 fully-connected (FC) layers is used to generate acoustic models. The activations of first CONV layer, the weights and outputs of final FC layer are represented in 16-bit, while the rest in 1-bit. The outputs of the final FC layer are also sent to self-learning unit for user-specific weight update. A smoothing unit exploits the histogram of phoneme-level accuracy to detect the wake-up word and recognize voice commands. During runtime, the inactive units are clock-gated, saving power by 30%-40%.

Optimized BCNN Computation and Weights Compression

Fig.3 illustrates the optimized BCNN computation flow considering the time-domain speech data redundancy. Firstly, as two consecutive feature maps have 10 overlapped frames, their convolutional results are also overlapped by 80.48% in average. We eliminate this large redundancy by buffering the convolutional results, reusing and updating them in the computation of each frame. Secondly, we partition the input buffer into 3 banks. With the mapping rule in Fig.3, arbitrary 3 consecutive rows can be read out simultaneously, avoiding memory access collision, which supports the 3x3 kernels in CONV layers. Thirdly, we rearrange the positions of zeros in the 3-D weight matrix, put a given number of these zeros together for compression. To facilitate this mechanism, we design hybrid weight memory with different banks to store varied types of weights, flag tables to generate address, and decoder to decompress the weights. With negligible accuracy loss, the on-chip weight storage and the number of weights access times are both reduced by 24.25% ~ 27.5%.

Customized Approximate Computing

In the computation of BCNN, 99% of operations are additions, and 95.9% of additions are 1-increment additions. Therefore, we customize the adder to support accurate 1-increment additions and approximate 16-bits additions. As shown in Fig.4, this adder is designed as a combination of a 3b-RCA (Ripple Carry Adder), two 4b-RCAs and an approximate 5b-RCA. The carry chain of the approximate 5b-RCA is designed for correctly propagating the carry bit which guarantees the accuracy of increment additions. This adder's critical path (marked in red) is 49.28% shorter than an accurate 16b-RCA's carry chain, obtaining 48.08% PDP (Power-Delay-Product) reduction. Since it guarantees the accurate increment additions, the mean error rate is only 1.27%. For self-learning, we implement the exponential and logarithmic functions with piece-wise linear approximations. In this way, the absolute error of cross-entropy is only 0.087 in average.

On-chip Self-learning Mechanism

Some frames of high confidence are selected from user's input speech for self-learning. As shown in Fig.5, the entire flow is: computing *softmax* function, eliminating negative samples, generating one-hot labels, computing partial derivatives, and updating weights. Self-learning mechanism and FBank function share the same reconfigurable datapath (Fig.2). To implement *softmax* without large hardware overhead, we firstly subtract the maximum value from each *softmax* input. Therefore, each input value of exponential function is smaller than zero, making its implementation complexity greatly reduced. Secondly, as logarithmic function is also needed in the FBank, we replace the division in the initial formula by nesting exponential and logarithmic functions. Furthermore, we improve the performance by pipeline scheduling, which is shown in Fig.5. As BCNN takes more time than VAD and feature extraction (front-end), the spare time can be filled with self-learning. This pipelining scheme enhance the resource utilization rate by 18.9%, and shorten the latency by 45%.

Measurement results

The test chip is fabricated in 28nm CMOS technology. Fig. 8 shows the die photo and chip spec. The core size is 1.29 mm^2 including only 52KB memory which can store all the binary weights and activations on-chip. Measurement results are shown in Fig. 7. The test chip achieves 141uW~2.85mW

power consumption with 2.5MHz-50MHz clock frequency from 0.57V~0.9V. The peak power efficiency, neural network efficiency and energy per frame are 90TOPS/W, 2.56pJ/ Neuron and 141nJ/Frame respectively, while consuming 1.42mW at 0.58V and 20MHz (Fig.7). A variety of speech benchmarks have been tested on this chip (Fig.6). With the number of epochs in self-learning increases, the accuracy is much improved. Finally, we obtained 96% wake-up rate and 95% command recognition accuracy on a real-life Chinese domestic appliance control benchmark. The processing latency of one speech frame is 0.5ms~10ms, which guarantees the real-time applications. Compared to [2] and [5], this chip consumes only 7.8% and 44% power, and shows 6.50× and 14.79× higher neuron efficiency and energy per frame, respectively.

References

[1] X. Xiang, et al., Interspeech, pp. 533-537. 2017.
[2] M. Price, et al., ISSCC, pp. 244-245, 2017.
[3] M. Shah, et al, Journal of Signal Processing Systems, 2016.
[4] W.T Sai, et al., IEEE Trans. on Computers, 66 (6), pp. 996-1007.
[5] S. Bang, J. Wang, Z. Li, and et al., ISSCC 2017, pp. 250-251

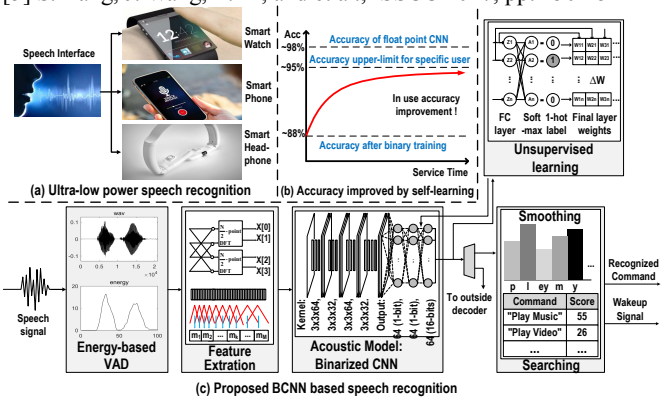


Fig.1 BCNN based Ultra-lower power speech recognition.

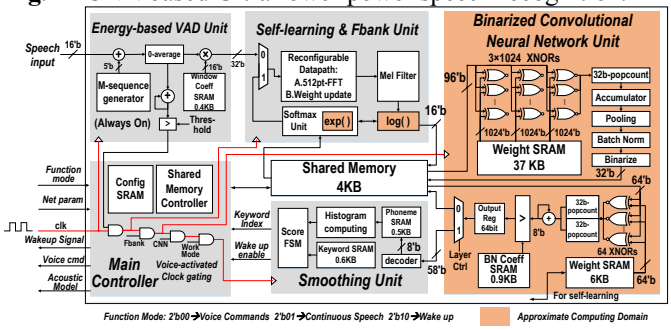


Fig.2 Architecture of proposed speech processing engine

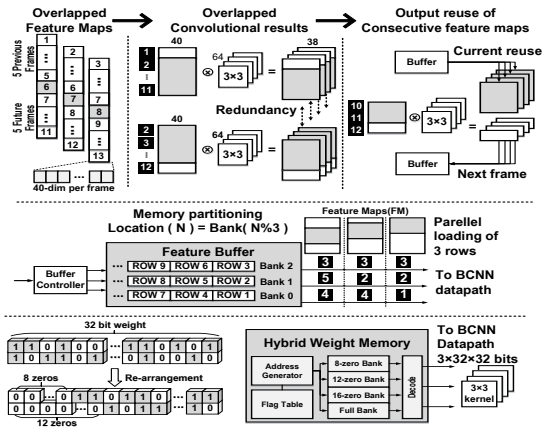
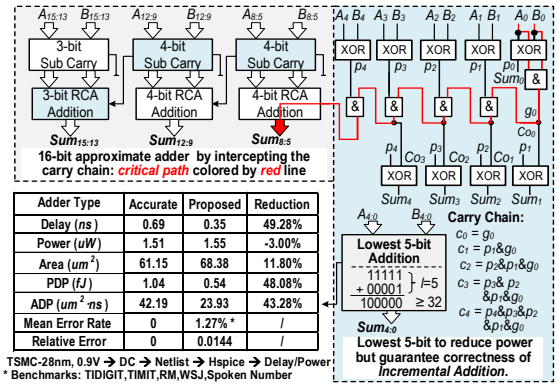


Fig.3 Data oriented BCNN computation techniques



Adder Type	Accurate	Proposed	Reduction
Delay (*ns*)	0.69	0.35	49.28%
Power (*uW*)	1.51	1.55	-3.00%
Area (um^2)	61.15	68.38	11.80%
PDP (*fJ*)	1.04	0.54	48.08%
ADP ($um^2 \cdot ns$)	42.19	23.93	43.28%
Mean Error Rate	0	1.27% *	/
Relative Error	0	0.0144	/

TSMC-28nm, 0.9V → DC → Netlist → Hspice → Delay/Power
* Benchmarks: TIDIGIT,TIMIT,RM,WSJ,Spoken Number

Fig.4 Approximate 16b Adder

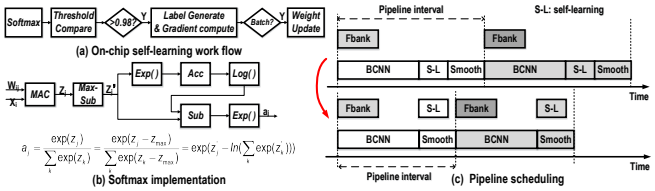
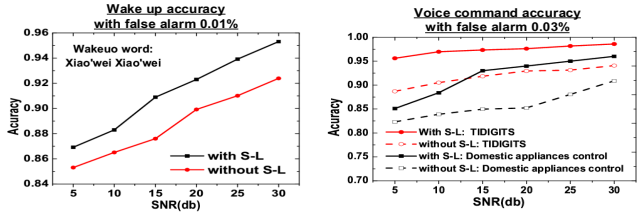
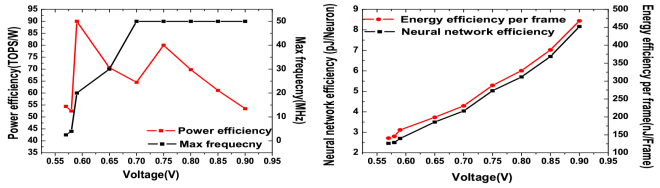


Fig.5 Self-learning flow, Softmax and pipeline scheduling



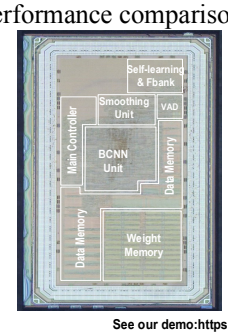
Benchmark	TIMIT	TIDIGITS	Domestic appliances control
Without S-L	25.20%	5.18%	3.97%
With S-L	21.58%	3.89%	1.94%
Relative Improvement	14.37%	24.9%	51.4%

Fig. 6 Measured results on recognition accuracy



Metric	ISSCC 2017 [2]	ISSCC 2017 [5]	JSPS 2016 [3]	TC 2016[4] TrueNorth	This work
Technology	65 nm	40 nm	40 nm	28 nm	28 nm
Area	9.61 mm^2	7.1 mm^2	12 mm^2	4.3 cm^2	1.29 mm^2
Voltage	0.6V - 1.2 V	0.63 V - 0.9 V	0.6 V	0.8 V	0.57 V - 0.9 V
Frequency	3 - 86 MHz	1.9 - 19.3 MHz	50 MHz	/	2.5 MHz - 50 MHz
Latency	Real-time	6.5 ms	10 ms	Real-time	0.5 ms - 25 ms
Power	SR (with decoder): 1.8 mW - 7.8 mW	Wakeup: 321 uW @3.9MHz 0.65 V	KWS: 11.2 mW SR (without decoder): 51.96 mW	Wakeup: 38.59 mW	Wakeup / Voice command / SR(without decoder): Min: 141 uW @2.5MHz, 0.57 Max: 2.85 mW @50 MHz, 0.9 V
Peak energy efficiency per Frame	/	2086 nJ / Frame	112 mJ / Frame	/	141 nJ / Frame
Peak neural network efficiency	16 pJ / Neuron	2.54 nJ / Neuron	140 nJ / Neuron	/	2.46 pJ / Neuron
Power efficiency	/	318 GOPS / W	/	/	90 TOPS / W

Fig.7 Performance comparison with voltage scaling,



Process	28 nm CMOS
Supply voltage	0.57 V - 0.9 V
Frequency	2.5 MHz - 50 MHz
Core size	1.74 × 0.74 mm^2
Die size	2 × 1 mm^2
Logic gates (NAND2)	0.78 million
SRAM	52 KB
Power consumption	141 uW – 2.85 mW
Power efficiency	90 TOPS / W
Energy per frame	141 nJ / Frame
Neural network efficiency	2.46 pJ / Neuron

See our demo:https://youtu.be/X_uYRsY5VHs

Fig.8 Die photo and chip specifications