A 250.3µW Versatile Sound Feature Extractor Using 1024-point FFT 64-ch LogMel Filter in 40nm CMOS

<u>Akiho Kawada*</u> (akihokawada@g.ecc.u-tokyo.ac.jp), Kenji Kobayashi*, Jaewon Shin, Rei Sumikawa, Mototsugu Hamada, Astutake Kosuge

*: equal contribution

the University of Tokyo

Outline

- Motivation
- Overview
 - Process Flow
 - Comparison with Prior Research
- Proposed Feature Extractor
 - FFT Implementation Based on R2²SDF
 - Zero-skipping Mel Filterbank
 - Log LUT
- Evaluations
 - Chip Implementation
 - Sound Recognition Performance
 - Performance Comparison (Power Efficiency etc.)
- References

Motivation

The widespread use of always-on voice applications



[Current] One app, one system: Inefficient design

- Our focuses
 - High versatility
 - Low power consumption

Overview: Process Flow



Overview: Comparison with Prior Research



Overview: Comparison with Prior Research



Proposed Method Details 1: <u>FFT</u> Implementation Based on R2²SDF

— The most power-consuming part

Our approach

- Serial implementation using the radix-2² algorithm
 - Reduces additions and multiplications
 - Simplifies with a serial implementation



Proposed Method Details 1: FFT Implementation Based on R2²SDF

Transforming the equation to set the first twiddle factor to -j makes the iteration radix 4

Algorithm	radix 2	radix 4	radix 2 ²
note	the radix of the butterfly operation is 2 (the most basic)	the radix of the butterfly operation is 4	the radix of the butterfly operation is <i>virtually</i> 4
additions and multiplications	more calculations	fower calculations	fower calculations
· · · · · · · · · · · · · · · · · · ·			
repeating structure	simple	complex	rather simple

Proposed Method Details 2: Zero-skipping mel <u>filterbank</u>

Filter weight value

→ A collection of filters

• A single feature map is formed by MACing each low-pass filter proposal1 with FFT output and merging all MAC results FFT The characteristics of mel filterbank are independent of sound proposal2 recognition tasks Mel \rightarrow Can be stored in ROM on hardware Filterbank Mel filterbank proposal3 • Mel filterbank: most filter coefficients are zero • Each filter acts as a low-pass filter Log Ch 0 Ch 1 \rightarrow introducing zero-skipping Ch 64 Compress the filter matrix size to 1/25 log melspectrogram Number of multiplications reduced to 1/25 Frequency [Hz] 9

Proposed Method Details 3: LogLUT

Logarithmic calculations generally require many cycles due to iterative computation

 \rightarrow Ours: Reduce cycles by **using an LUT**



proposal1

proposal2

FFT

Evaluations: Chip Implementation

- Implemented in 40nm CMOS
- 2064 cycles required for computation (4.13 ms delay at 500 kHz)
- Chip implementation area: 2340 μ m × 174 μ m
- Power consumption: 250.3 μ W at 1.1V



Process	40nm CMOS	Bit width	14bit
Area	2340µm × 174µm	Supply voltage	1.1V
Clock	500kHz	Total power	250.3mW

Evaluations: Accuracy Comparison on Sound Recognition

 Inferential performance of (3 kinds of LogMel) + DNN simulated for sound recognition tasks

	Feature	SW/HW	FFT sample	Environm ental	Linguistic Recog.	KWS- 12words	KWS- 35words	Average Accuracy
Numpy	LogMel	64bit SW	N=1024	78.1%	83.5%	86.5%	88.9%	84.3%
N=256 case		14bit HW	N=256	54.4%	63.8%	66.4 %	37.0%	52.8%
Ours		14bit HW	<i>N</i> =1024	74.7%	83.3%	85.1%	81.4%	81.1% (-3.2%)

<u>Results</u>

- only 3.2% accuracy drop compared to Numpy
- 16.3% improvement compared to N=256 HW FFT
- → Enough FFT data points *N* are essential

Evaluations: Performance Comparison (Power Efficiency etc.)

Achieves versatility and high power efficiency

- Various applications
- Power efficiency equal to or greater than conventional ASICs
- Digital configuration

	JSSC'21 [1]	JSSC'22 [3]	ISCAS'21 [2]	Ours
Applicable tasks	2-words KWS	12-words KWS	30-words KWS	(1) 35-words-KWS, (2) Language identification (3) Environmental sound
Filter type	MFCC	Analog filter	MFCC	LogMel
Filter output dimensions	10	16	40	64
Points of FFT	N=256	NA	N=512	N=1024
Process node	28nm CMOS	65nm CMOS	180nm CMOS	40nm CMOS
Feature extractor circuit area	0.054mm ²	1.60mm ²	2.39mm ²	0.41mm ²
Power consumption of feature extractor	2.00 mW	9.3 mW	26.4 mW	250.3 mW
Energy efficiency in KWS at normal supply voltage	16.0 nJ/frame/word	12.7 nJ/frame/word	8800.0 nJ/frame/word	14.9 nJ/frame/word (1/1.1)

Conclusions

Our versatile & energy-efficient audio feature extractor

- FFT: N=1024 & serial implementation of radix-2² algorithm
- mel filterbank: 64-ch & zero-skipping
- log: LUT

References

[1] W. Shan et al., "A 510-nW Wake-Up Keyword-Spotting Chip Using Serial-FFT-Based MFCC and Binarized Depthwise Separable CNN in 28-nm CMOS," in IEEE Journal of Solid-State Circuits, vol. 56, no. 1, pp. 151-163, Jan. 2021.

[2] L. Wu et al., "A High Accuracy Multiple-Command Speech Recognition ASIC Based on Configurable One-Dimension Convolutional Neural Network," in IEEE ISCAS, May 2021.

[3] K. Kim et al., "A 23-μW Keyword Spotting IC With Ring-Oscillator- Based Time-Domain Feature Extraction," in IEEE Journal of Solid- State Circuits, vol. 57, no. 11, pp. 3298-3311, Nov. 2021.

[4] R. Sumikawa et al., "A183.4-nJ/inference 152.8-µW 35-Voice Commands Recognition Wired-Logic Processor Using Algorithm- Circuit Co-Optimization Technique," in IEEE Solid-State Circuit Letters, vol. 7, pp. 22-25, 2024.

[5] D. Niizumi et al., "BYOL for Audio: Self-Supervised Learning for General-Purpose Audio Representation," in IEEE International Joint

Conference on Neural Networks, 2021. [Online]. Available: https://arxiv.org/abs/2103.06695

[6] D. Jaeon et al., "A Super-Pipelined Energy Efficient Subthreshold 240 MS/s FFT Core in 65 nm CMOS," in IEEE Journal of Solid-State Circuits, vol. 47, no. 1, pp. 23-34, Jan. 2012.

[7] S. He and M. Torkelson, "A New Approach to Pipeline FFT Processor," in IEEE Proceedings of IPPS '96, 1996, pp. 766 -770.

[8] A. Kosuge et al, "A 183.4nJ/inference 152.8uW Single-Chip Fully Synthesizable Wired-Logic DNN Processor for Always-On 35 Voice Commands Recognition Application," in IEEE Symposium on VLSI Circuits, June 2023.

[9] D. Llamocca and C. Agurto, "A Fixed-point implementation of the natural logarithm based on a expanded hyperbolic CORDIC algorithm", in XII Workshop IBERCHIP, 2006.