엣지 디바이스에서 NPU를 활용한 MobileNet 기반 모델의 추론 성능 분석

조은영, 김태구, 조용훈, 신기훈, 곽도균, 구동한, 나상진, 김태훈, 황우찬, 백윤주*

부산대학교

{ciara0124, tbg8577, kchoyh95, skh2929209, kwakdg, ehdgksrn, sktkdwls1222, bigteach0508, woochan629}@pusan.ac.kr,

*yunju@pusan.ac.kr

Analysis of Inference Performance of MobileNet-Based Models Utilizing an NPU on Edge Devices

Cho Eun Young, Kin Tae Gu, Cho Yong Hun, Shin Ki Hun, Kwak Do Gyun, Koo Dong Han, Na Sang Jin, Kim Tae Hun, Woo Chan Hwang, Baek Yun Ju*

Pusan National Univ.

요 약

엣지 디바이스에 경량 AI 모델을 탑재하여 외부 네트워크 연결 없이 디바이스 내부에서 자체적으로 추론하는 온디바이스 AI에 대한 수요가 증가하고 있다. 온디바이스 AI는 데이터 처리 지연을 줄이고 실시간 처리가 가능하다는 장점이 있으나, 엣지 디바이스의 연산 능력, 메모리, 전력 소모량 등 다양한 자원 제약이 존재한다. 이를 해결하기 위해 딥러닝 연산에 특화된 구조로 설계된 NPU 기반 하드웨어 가속 기술이 주목받고 있다. 본 논문에서는 엣지 디바이스 환경에서 NPU의 가속 효과를 검증하기 위해, NPU가 내장된 Renesas RA8P1 MCU에 경량 CNN 모델인 MobileNet을 탑재하여, CPU, NPU 실행 환경에 따른 추론 성능을 비교하였다. MobileNet 모델은 INT8 정밀도로 양자화 후, RUHMI Framework를 사용하여 MCU 환경에 최적화하였다. 실험 결과, NPU를 활용한 MobileNet 모델의 추론 성능은 CPU 실행 대비 추론 시간이 평균 약 87% 감소하였다.

Ⅰ. 서 론

최근 엣지 컴퓨팅 환경의 확산에 따라, 데이터가 생성되는 엣지 디바이스 내부에서 AI 모델을 추론하는 온디바이스 AI(On-device AI) 기술에 대한 수요가 증가하고 있다. 온디바이스 AI는 중앙 서버와의 통신 의존도를 낮춤으로써 네트워크 통신으로 인한 지연 시간을 감소시키고, 실시간 처리가 요구되는 응용 분야에 적합하다. 그러나 엣지 디바이스는 낮은 연산 능력, 수 kB에서 MB 수준의 메모리 용량 등 자원이 제한적이므로, 자원 제약적인 임베디드 환경에서 딥러닝 모델을 효율적으로 실행하는 것은 여전히 중요한 과제로 남아 있다 [1].

온디바이스 AI 구현을 위해, 실제 산업 환경에서는 합성곱 신경망(Convolutional Neural Network, CNN)을 경량화한 모델이 널리 활용되고 있다. MobileNet 계열 모델은 파라미터 수와 연산량을 효과적으로 줄인 대표적인 경량 CNN 모델로, 구조적 단순성과 다양한 모델 변형을 제공하여, 제한된 자원을 가진 디바이스 환경에서 성능과 효율성 간의 균형을 분석하기에 적합하다 [2].

최근에는 엣지 디바이스의 추론 성능을 향상시키기 위해 신경망 처리 장치(Neural Processing Unit, NPU)를 활용한 하드웨어 가속 기술이 주목받고 있다. NPU는 딥러닝 연산에 특화된 구조로 설계되어 연산 지연을 감소시키고 전력 효율을 향상시켜, 자원이 제한된 엣지 디바이스 환경에서 효과적인 추론 수단으로 평가된다 [3]. 이는 동일한 딥러닝 모델이라 하더라도, NPU의 활용 여부에 따라 추론 시간, 메모리 사용량 등 실행 특성이 상이하게 나타날 수 있음을 의미한다.

본 논문은 NPU가 내장된 Renesas RA8P1 Micro Controller Unit(MCU)의 평가 보드인 EK-RA8P1에 MobileNet 기반 모델을 탑재하여, CPU 및 NPU 실행 환경에 따른 추론 성능 차이를 실험적으로 비교한다. 이를 통해 엣지 디바이스 환경에서 NPU의 가속 효과를 검증하고, NPU 활용을 위한 딥러닝 모델 변환 과정과 메모리 최적화 측면에서 고려해야 할 요소들을 고찰한다.

Ⅱ. 본론

1. 실험 환경 및 시스템 구성

본 논문에서는 Renesas 사의 RA8P1 MCU를 실험에 사용한다. RA8P1은 ARM Ethos-U55 NPU가 내장된 엣지 디바이스로, CPU 기반 연산뿐만 아니라 NPU를 활용한 딥러닝 추론을 지원한다. Ethos-U55는 CNN 기반 모델의 연산을 가속하도록 설계되었으며, NPU에서 지원하지 않는 연산자는 CPU에서 처리된다. 이러한 구조는 자원이 제약적인 환경에서의 NPU 가속 효과를 평가하기에 적합하다.

딥러닝 모델의 효율적인 배포를 위해 Renesas 사의 Robust Unified Heterogeneous Model Integration(RUHMI) Framework를 사용하였다. RUHMI Framework는 학습된 딥러닝 모델을 RA8P1의 하드웨어 특성에 맞게 최적화하는 기능을 지원한다.

2. 모델 양자화

Ethos-U55 NPU는 INT8 정밀도로 양자화된 모델만을 지원한다. 이에 따라 NPU 활용 효과를 비교하기 위해, FP32 TFLite 모델을 RUHMI Framework를 통해 INT8로 양자화한 경우와 TensorFlow에서 사전 양자화된 TFLite INT8 모델을 대상으로, 모델 생성 방식에 따른 모델 구조 및 추론 시간의 차이를 분석하였다.

3. 모델 변환

본 논문에서는 사전 학습된 MobileNet 모델을 사용하여, RUHMI

Framework를 통해 RA8P1에서 실행 가능한 TFLite 기반 C 코드로 변환한다. 변환 과정에서 동일한 모델 구조를 유지한 상태에서 실행 환경을 CPU 또는 NPU로 구분하여 설정함으로써, 실행 환경에 따른 성능 차이를 비교할 수 있도록 구성하였다.

RA8P1 평가 보드는 1MB의 내부 MRAM과 64MB의 외부 OSPI 플래시 메모리를 포함한다. 내부 MRAM의 제한된 용량으로 인해 모델의 크기에 따라 빌드 단계에서 플래시 메모리 오버플로우 문제가 발생할 수 있다. 이를 해결하고자 모델 가중치를 외부 OSPI 플래시 메모리에 저장하도록 구성하였다. CPU 단독(CPU-only) 실행 환경에서는 다수의 CMSIS-NN 커널이 사용됨에 따라 메모리 사용량이 증가하여 내부 메모리 용량을 초과하는 문제가 발생하였다. 이를 해결하기 위해 입력 데이터 버퍼를 RA8P1 평가 보드에 포함된 64MB 외부 SDRAM에 할당하였다.

RUHMI Framework의 Visualizer를 사용하여 변환된 모델을 분석한 결과, FP32 정밀도의 TFLite 모델에 대해 RUHMI Framework의 양자화를 적용할 경우, 실행 환경에 관계 없이 입·출력 데이터 버퍼가 FP32 정밀도로 유지됨을 확인하였다. 또한, 동일한 모델을 NPU 실행 환경으로 변환한 경우, 일부 연산자(TFLiteQuantize, TFLitePad, TFLiteDequantize)는 NPU에서 지원되지 않아 CPU에 할당되는 것을 확인하였다.

Ⅲ. 실험

1. 실험 설정

실험에 사용한 MobileNet 모델은 약 1,000개의 객체 클래스를 포함하는 대규모 이미지 분류 데이터 세트인 ImageNet으로 사전 학습된 모델을 기반으로 한다. 본 연구에서는 MobileNet V1, V2, V3(Large, Small, minimalistic) 구조를 대상으로 비교 분석하였다. INT8 양자화를 위해 ImageNet 학습 데이터의 일부를 representative dataset으로 사용하여 학습 후 양자화(Post-Training Quantization, PTQ)를 적용하였다. 모든 입력 이미지는 224x224 해상도로 정규화하였으며, 모델 변환에는 RUHMI Framework의 기본 설정을 사용하였다. 표 1은 실험에 사용한 각 MobileNet 모델의 파라미터 수와 추론 시 요구되는 Multiply-ACcumulate(MAC) 연산량을 나타낸다. 엣지 디바이스에서 NPU의 가속 효과를 검증하기 위해 RA8P1의 평가 보드(EK-RA8P1)를 사용하였으며, 주요 제원은 표 2와 같다.

표 1. 실험에 사용한 MobileNet 모델

모델 종류		Params (M)	MACs (M)
MobileNet V1		4.221	572.4
MobileNet V2		3.488	304.4
MobileNet V3	Large	5.471	217.2
	L minimalistic	3.912	209.2
	Small	2.537	56.8
	S minimalistic	2.039	52.3

표 2. 실험에 사용된 EK-RA8P1 평가 보드 제원

	EK-RA8P1
CPU	Arm Cortex-M85@1 GHz, Arm Cortex-M35@250 MHz
NPU	Arm Ethos-U55@500 MHz, 250GOPS, 256 MACS/cycle
Memory	1 MB MRAM, 2 MB SRAM, 64 MB Octo-SPI flash, 64 MB SDRAM

모델 생성 방식과 실행 환경에 따른 성능 분석을 위해 네 가지 실험 구성을 정의하였다. Pre-Quantized(PQ) 모델은 TensorFlow에서 사전 양자화된 TFLite INT8 모델을 의미하며, RUHMI-Quantized(RQ) 모델은 RUHMI Framework를 통해 양자화된 모델을 의미한다. 각 모델은 CPU 또는 NPU에서 실행되었으며, 이에 따라 PQ-CPU, RQ-CPU, PQ-NPU, RQ-NPU의 네 가지 실행 구성을 비교하였다.

2. 실험 결과

실험에서는 PQ-CPU, RQ-CPU, PQ-NPU, RQ-NPU를 대상으로 MobileNet 모델의 추론 시간을 비교 분석하였다. 표 3은 MobileNet 모델의 구조 및 실행 환경에 따른 추론 시간 측정 결과를 나타낸다. 모든 MobileNet 모델에서 NPU를 활용한 경우, CPU 실행 대비 평균적으로 약 87%의 추론 시간이 감소하였다. 이는 엣지 디바이스 환경에서의 NPU 가속이 CNN 기반 딥러닝 모델의 추론 시간을 효과적으로 단축함을 보여준다. RQ-NPU 구성은 일부 연산자가 CPU에 할당됨에 따라 PQ-NPU 대비 추론 시간이 소폭 증가하는 경향을 보였다.

또한, MobileNet V3 Small 모델을 대상으로 PQ-NPU 및 Raspberry Pi 5에서의 추론 시 소비 전력을 비교하였다. 측정 결과, PQ-NPU는 0.33W, Raspberry Pi 5는 12ms의 추론 시간과 함께 2.3W의 소비 전력을 나타냈다. 이러한 결과는 MCU 기반 NPU 가속 추론이 Single Board Computer(SBC) 환경 대비 낮은 전력 소모로 딥러닝 모델 추론을 수행할 수 있음을 보여준다.

표 3. MobileNet 구조 및 실행 환경에 따른 추론 시간 비교 (단위: ms)

	PQ-CPU	RQ-CPU	PQ-NPU	RQ-NPU
V1	1451	1462	131	183
V2	1048	1138	113	175
V3 Large	1151	1138	113	159
V3 L minimalistic	737	764	96	142
V3 Small	456	470	40	60
V3 S minimalistic	230	244	32	53

Ⅳ. 결론

본 논문은 MCU 기반 엣지 디바이스 환경에서 NPU의 가속 효과를 검증하기 위해, NPU가 내장된 RA8P1 MCU에서 MobileNet 모델의 성능을 비교 분석하였다. 실험 결과, NPU를 활용한 경우 모든 MobileNet 모델에서 CPU 실행 대비 약 87%의 추론 시간 감소가 확인되었다. 이는 제한된 자원을 가진 엣지 디바이스 환경에서 NPU 가속이 CNN 기반 모델의 실시간 추론 성능 향상에 효과적임을 보여준다. 향후 연구로, self-attention 연산의 높은 연산량과 메모리 요구로 인해 MCU 기반 엣지 디바이스 환경에서의 적용에 제약이 있던 Transformer 모델을 최적화를 통해 MCU에 탑재한 온디바이스 AI에 대한 연구를 수행할 예정이다.

ACKNOWLEDGMENT

이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-대학ICT연구센터(ITRC)의 지원을 받아 수행된 연구임
(IITP-2026-RS-2023-00259967)

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 인공지능융합혁신인재양성사업 연구 결과로 수행되었음(IITP-2026-RS-2023-00254177)

참 고 문 헌

[1] Ngo, Dat, Hyun-Cheol Park, and Bongsoon Kang. "Edge Intelligence: A Review of Deep Neural Network Inference in Resource-Limited Environments." Electronics 14.12 (2025): 2495.

[2] Howard, Andrew, et al. "Searching for mobilenetv3." Proceedings of the IEEE/CVF international conference on computer vision. 2019.

[3] Fanariotis, Anastasios, Theofanis Orphanoudakis, and Vasilis Fotopoulos. "Evaluating the Energy Efficiency of NPU-Accelerated Machine Learning Inference on Embedded Microcontrollers." arXiv preprint arXiv:2509.17533 (2025).