USB형 가속기와 내장형 TPU의 에너지 효율 비교

강은선 O, 강민선, 박문주
인천대학교
dmstjs7047@hanmail.net, kangpenguin@naver.com, mpark@inu.ac.kr

Energy efficiency comparison of USB accelerator and an embedded TPU

Eunsun Kang O, Minseon Kang, Moonju Park
Incheon National University

요 약

하드웨어 가속기 사용으로 임베디드 시스템의 성능이 올라감에 따라서 인공지능 어플리케이션을 시스템 내에서 직접 돌리는 시도가 높아지고 있다. 어떤 형태의 가속기를 사용하는지가 임베디드 시스템의 성능과 에너지 효율 향상에 큰 영향을 줄 것으로 보인다. 따라서 본 논문에서는 USB형 가속기와 내장형 TPU의 성능 차이를 비교하기 위해 하나의 모델을 두 형태의 TPU로 돌려 소모된 전력과 에너지를 측정하였다. 내장형 TPU가 USB형 가속기보다 추론 시에는 더 적은 에너지를 소모하였으며 CPU를 사용했을 때와 비교하여 성능, 소모 에너지 면에서 더 높은 향상 정도를 보이고 있다. 따라서 USB형 가속기를 사용하는 것 보다 내장형 TPU를 사용하는 것이 CPU를 사용할 때와 비교하면 에너지 효율적 측면에서 더 좋다고 볼 수 있다.

1. 서 론

하드웨어 가속기 사용으로 임베디드 시스템의 성능이 올라감에 따라서 인공지능 어플리케이션을 클라우드 서버를 이용하지 않고 직접 돌리는 시도가 높아지고 있다. 클라우드 서버를 이용하지 않고 임베디드 시스템을 사용하여 인공지능 어플리케이션을 돌린다면 네트워크 트래픽과 지연시간을 줄일 수 있다.

임베디드 시스템에서 사용할 수 있는 가속기의 형태는 다양하다.[1] 이때 어떤 가속기를 사용하는지가 임베디드 시스템의 성능과 에너지 효율을 높이는데 중요한 변수가 될 것이다.

2. 관련 연구

가속기를 사용하는 방식은 USB형과 내장형으로 나뉜다. USB형 가속기에는 Intel neural compute stick2[2], coral USB accelerator[3] 등이 있고 내장형 가속기에는 Nvidia Jetson Nano[4], Coral Dev Board[5] 등이 있다. 이 중 USB형과 내장형을 모두 지원하는 가속기는 Coral사에서 개발한 TPU(Tensor Processing Unit)가 있다.

인공지능 어플리케이션을 돌리기 위한 가속기에 대한 연구는 다양하다. GPU(Graphic Processing Units)[6]를 사용한 연구가 가장 많으며 GPU와 비교하여 전력소모 대비 연산 속도를 향상시키는 가속기들도 제안되고 있다.[7] 기존 CNN 가속기보다 향상된 데이터 처리율과 에너지 효율을 보여주는 새로운 CNN 가속기 구조를 설계하는 방법을 설명하거나[8] 다양한 CNN 가속기의 공정과 동작 속도 등을 동일하게 재구성하여 각각의 면적, 에너지, 성능을 비교하는 연구도 있다.[9]

3. 실 험

3.1 하드웨어

이번 실험에서 USB형 가속기로는 Coral USB Accelerator(이하 Coral Accelerator로 표기)를 사용하였고 가속기와 연결할 보드로 raspberry pi 3 model B(이하 raspberry pi 3로 표기)를 사용하였다. 내장형 TPU로는 Coral Dev Board를 사용하였다. USB형과 내장형 가속기 보드의 스펙은 <표1>과 같다.

<표 1> 보드 비교

	Raspberry pi 3 + Coral Accelerator	Coral Dev Board
CPU	Quad Core 1.2GHz Broadcom BCM2837	Quad Cortex-A53, Cortex-M4F
가속기	Google Edge TPU coprocessor	
메모리	Micro SD	8 GB eMMC, MicroSD slot

3.2 사용한 모델

본 실험은 Coral 홈페이지에서 제공하는 기본 예제 중 image classification을 사용하였다[10]. 모델과 라벨 파일은 MobileNet V2 (iNat birds)의 모델과 라벨 파일을

이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No.2020R1F1A1053753).

사용하였다.[11] 해당 예제는 새 사진을 입력 받아 label list에 있는 새 종류 중 입력 받은 새 사진에 해당하는 새의 이름을 찾아주는 예제이다. 실험은 새 사진 400장을 불러와 가공하는 작업(이하 사진 전처리 또는 Input으로 표기)과 가공한 사진으로 추론하는 작업(이하 사진 추론 또는 Inference로 표기)을 세트로 하여 5번 반복하였다.

예제에서 제공하는 모델은 두 가지가 있다. 하나는 TPU를 사용하는 것이고 나머지는 CPU만 사용하는 것이다. CPU만 사용할 때는 가속기를 사용하지 않으므로 TPU 위임코드(delegate code)를 제거하여 사용하였다. 첫 번째는 Coral Accelerator 없이 CPU 모델로 실행하였고, 두번째는 Coral Accelerator와 TPU 모델, 위임 코드를 사용하였다.

3.3 위임 코드(delegate code)

TensorFlow가 정의한 위임이란 현재 사용하는 프로세서 이외에 가속기나 다른 프레임워크를 추가로 사용하기 위해 제공하는 방법이다. 위임 방법에는 external delegate를 사용하거나, delegate provider를 사용하는 방법 두가지가 있다.[12] 하지만 TPU에서는 위 방법이 아닌 바로 TPU를 사용하는데 필요한 라이브러리를 가져오는 방법을 사용한다.

USB형 가속기, 내장형 TPU 모두 위임할 때 해당 코드가 필요하다. 그러나 CPU만 사용하는 경우에는 디바이스를 감지할 수 없기 때문에 위임 코드를 제거하여 실험하였다.

3.4 실험 결과

IDLE의 경우 Coral USB Accelerator를 연결한 raspberry pi 3는 평균적으로 1.77W의 전력을, Coral Deb Board는 3.58W로 약 2배의 전력을 소모했다. Raspberry pi 3는 사진 전처리에 평균 189.008J, 추론에 평균 12.212J을 소모했다. Coral Dev board는 사진 전처리에 평균 218.294J, 추론에 평균 5.43J을 소모했다.

사진 전처리에 raspberry pi 3보다 Coral Dev Board가 더 많은 에너지를 소모했지만 이는 Coral Dev Board의 IDLE이 더 높은 탓으로 이를 감안하면 오히려 Coral Dev Board가 더 적은 에너지를 소모한 것으로 볼 수 있다. 사진 추론에서는 Coral Dev Board가 raspberry pi 3보다 더 적은 에너지를 소모했다. Coral Deb Board의 전력 소모가 더 컸지만 약 1/4의 속도로 매우 빠른 추론을 했기 때문에 결과적으로는 Coral Dev Board가 더 적은 에너지를 소모했다.

Coral Accelerator를 연결한 Raspberry pi 3의 성능 향상은 약 4.3%, 소모 에너지 향상은 약 4.6%의 향상 정도를 보였다. Coral Deb Board는 성능 향상이 약 2%, 소모 에너지 향상이 약 2.1%로 Coral Accelerator를 연결한 Raspberry pi 3보다 성능, 소모 에너지 향상 정도가 모두 높았다.

본 논문에서는 에너지 효율을 비교하기 위해 Energy Delay Product(이하 EDP로 표기)[13]를 사용한다. EDP는 소모한 에너지에 시간을 곱한 값으로 성능과 에너지 소모를 동시에 비교하기 위해 사용한다. 소모 에너지 만을 비교하는 경우 소모 에너지는 적어도 시간이 지연되는 경우가 발생할 수 있으므로 EDP를 비교하여 소모 에너지와 지연 시간을 모두 비교한다.

Coral Accelerator를 연결한 Raspberry pi 3는 CPU를 사용할 때보다 TPU를 사용할 때의 에너지x시간 값이 약 0.2%로 감소했다. Coral Dev Board는 CPU를 사용할 때보다 TPU를 사용할 때의 에너지x시간 값이 약 0.04%로 USB형 가속기보다 더 큰 감소율을 보였다.

<표 2> Raspberry pi 3 단계별 소요 시간, 전력, 에너지

TPU model, delegate code, Raspberry pi						
	Input			Inference		
	Time(ms)	Active Power(W)	Energy(J)	Time(ms)	Active Power(W)	Energy(J)
1	69402.7	2.800580645	194.3678583	4823	2.59025	12.49277575
2	64871.2	2.795946429	181.3764	4725.2	2.5682	12.13525864
3	67937.8	2.7692	188.1333558	4720.3	2.5774	12.16610122
4	69111.5	2.763983871	191.0230713	4740	2.564	12.15336
5	68561.3	2.773288136	190.1402399	4714	2.57	12.11498

<표 3> Coral Dev Board 단계별 소요 시간, 전력, 에너지

TPU model, delegate code, Coral						
	Input			Inference		
	Time(ms)	Active Power(W)	Energy(J)	Time(ms)	Active Power(W)	Energy(J)
1	45663.8	4.789547619	218.7089446	1140.9	4.925	5.6189325
2	45518	4.762641026	216.7858942	1091.4	4.964	5.4177096
3	45539.8	4.794025	218.3189397	1123.6	4.9275	5.536539
4	45382.5	4.80002439	217.8371069	1113.1	5.023	5.5911013
5	45534.3	4.827575	219.8202483	1092.2	4.569	4.9902618

<표 4> Raspberry pi 3에서의 CPU, TPU 시간, 소모 에너지 향상 비교

CPU vs TPU in Raspberry pi 3 when Inference (Time, Energy)						
	Time(s)			Energy(J)		
	CPU	TPU	TPU/CPU(%)	CPU	TPU	TPU/CPU(%)
1	109.335	4.823	4.411213244	270.0893394	12.49277575	4.625423491
2	109.2963	4.7252	4.323293652	270.5994228	12.13525864	4.484584083
3	108.8422	4.7203	4.336828914	266.7325501	12.16610122	4.561161063
4	107.4287	4.74	4.412228762	256.7411644	12.15336	4.733701363
5	107.3202	4.714	4.392462929	256.8307948	12.11498	4.717105676

<표 5> Coral Dev Board에서의 CPU, TPU 시간, 소모 에너지 향상 비교

CPU vs TPU in Coral Dev Board when Inference (Time, Energy)						
	Time(s)			Energy(J)		
	CPU	TPU	TPU/CPU(%)	CPU	TPU	TPU/CPU(%)
1	59.273	1.1409	1.924822432	265.3708128	5.6189325	2.117389038
2	59.2445	1.0914	1.842196322	268.1182506	5.4177096	2.020641858
3	59.2499	1.1236	1.896374509	269.3511848	5.536539	2.055509429
4	45.5492	1.1131	2.443731174	208.4597812	5.5911013	2.682100724
5	59.2454	1.0922	1.843518653	271.6541625	4.9902618	1.836990737

<표 6> Raspberry pi 3 CPU와 TPU의 EDP 비교

CPU vs TPU in Raspberry pi 3 when inference (EDP)			
	CPU(J*s)	TPU(J*s)	TPU/CPU(%)
1	29530.21792	60.25265744	0.204037294
2	29575.51569	57.34152413	0.193881739
3	29031.75757	57.42764759	0.197809752
4	27581.36953	57.6069264	0.208861733
5	27563.13227	57.11001572	0.207197118

<표 7> Coral Dev Board CPU와 TPU의 EDP 비교

CPU vs TPU in Coral Dev Board when Inference (EDP)			
	CPU(J*s)	TPU(J*s)	TPU/CPU(%)
1	15729.32419	6.410640089	0.040755979
2	15884.5317	5.912888257	0.03722419
3	15959.03077	6.22085522	0.038980157
4	9495.176265	6.223454857	0.065543332
5	16094.25952	5.450363938	0.033865267

4. 결 론

Coral Dev Board는 항상 TPU가 켜져 있기 때문에 기본 IDLE의 전력 소모 자체가 크다. 이로 인해 사진 전처리 시 Coral Accelerator를 연결한 raspberry pi 3보다 더 많은 에너지를 소모한다. 그러나 추론 시에는 Coral Accelerator를 연결한 raspberry pi 3보다 우수한 성능을 보이기 때문에 더 적은 에너지를 소모한다. 또한 Coral Dev Board가 Coral Accelerator를 연결한 Raspberry pi 3보다 성능 면에서도 소모 에너지 면에서도 더 높은 향상 정도를 보인다.

USB의 입출력 딜레이로 인해 Coral Accelerator를 연결한 Raspberry pi 3보다 Coral Dev Board의 EDP 향상 정도가 0.2%, 0.04%로 더 높다. 따라서 USB형 가속기보다 내장형 TPU가 가속기를 사용할 때 더 높은 에너지 효율의 향상을 보인다는 것을 유추할 수 있다.

본 실험에서는 Coral Accelerator와 Coral Dev Board를 비교하기 위해 TPU와 CPU만 사용 가능한 모델을 선택했지만 이후에는 다양한 가속기의 비교를 위해 직접 모델을 만들어 각 가속기마다 어떤 에너지 효율 정도를 보이는지 비교할 필요가 있다.

참고 문헌

[1] 김재준, 김형준, 김태수, 김율화, 김진석. 추론 전용 저항성 메모리 기반 뉴럴 네트워크 가속기 하드웨어 연구 동향. 전자공학회지, 45(7), 46-53. 2018.

[2] https://software.intel.com/content/www/us/en/develop/hardware/neural-compute-stick.html

[3] https://coral.ai/products/accelerator/

[4] https://developer.nvidia.com/embedded/jetson-nano-developer-kit

[5] https://coral.ai/products/dev-board/

[6] https://www.intel.com/content/www/us/en/products/docs/processors/what-is-a-gpu.html

[7] 박종현, 김민식, 김윤수, 이경민, 윤명국, 노원우. 인공신경망 연산을 위한 하드웨어 가속기 최신 연구 동향. 정보과학회지, 34(9), 21-26. 2016.

[8] 최동우, 이한호. 필터 분해 기법을 이용한 에너지 효율적 재구성형 CNN 가속기 구조. 전자공학회논문지, 57(7), 22-33. 2020.

[9] 김휘수, 최재완, 이선정, 안정호. 다양한 CNN 가속기에서 아키텍처에 따른 면적, 에너지, 성능 분석. 한국정보과학회 학술발표논문집, 708-710. 2020.

[10] https://github.com/google-coral/pycoral/blob/master/examples/classify_image.py

[11] https://coral.ai/models/

[12] https://www.tensorflow.org/lite/performance/implementing_delegate?hl=en

[13] R. Gonzalez and M. Horowitz, "Energy Dissipation In General Purpose Microprocessors," IEEE Journal of Solid-State Circuits, Vol. 31, Issue 9, pp. 1277-1284, Sep. 1996.