ISSN : 1226-7244 (Print)
ISSN : 2288-243X (Online)
논문번호 18-03-43
284
j.inst.Korean.electr.electron.eng.Vol.22,No.3,805~809,September 2018
http://dx.doi.org/10.7471/ikeee.2018.22.3.805

이진 가중치 신경망의 하드웨어 구현을 위한 고정소수점 연산 정확도 분석

Accuracy Analysis of Fixed Point Arithmetic for Hardware Implementation of Binary Weight Network

김 종 현*, 윤 상 균*★

Jong-Hyun Kim*, SangKyun Yun*★

Abstract

In this paper, we analyze the change of accuracy when fixed point arithmetic is used instead of floating point arithmetic in binary weight network(BWN). We observed the change of accuracy by varying total bit size and fraction bit size. If the integer part is not changed after fixed point approximation, there is no significant decrease in accuracy compared to the floating-point operation. When overflow occurs in the integer part, the approximation to the maximum or minimum of the fixed point representation minimizes the decrease in accuracy. The results of this paper can be applied to the minimization of memory and hardware resource requirement in the implementation of FPGA-based BWN accelerator.

요 약

본 연구에서는 이진 가중치 신경망(BWN)을 부동소수점 데이터를 사용하여 학습시킨 후에, 학습된 파라미터와 주요연산을 고정소수점으로 근사화시키는 과정에서 정확도의 변화를 분석하였다. 신경망을 이루고 있는 각 계층의 입력 데이터와 컨볼루션 연산의 계산에 고정소수점 수를 사용했으며, 이때 고정소수점 수의 전체 bit 수와 소수점 이하 bit 수에 변화를 주면서 정확도 변화를 관찰하였다. 각 계층의 입력 값과 중간 계산값의 정수 부분의 손실이 발생하지 않으면 고정소수점 연산을 사용해도 부동소수점 연산에 비해 큰 정확도 감소가 없었다. 그리고 오버플로가 발생하는 경우에 고정소수점 수의 최대 또는 최소값으로 근사시켜서 정확도 감소를 줄일 수 있었다. 이 연구결과는 FPGA 기반의 BWN 가속기를 구현할 때에 필요한 메모리와 하드웨어 요구량을 줄이는 데 사용될 수 있다.

Key words : Binary Weight Network, Low precision network, Fixed point approximation, FPGA, CNN

Ⅰ. 서론

심층신경망은 컴퓨터 비전과 음성인식 등 다양한 분야에서 성능이 입증되고 있다. 여러 가지 심층신경망 방법 중에서 CNN(Convolutional Neural Network)이 컴퓨터 비전 분야에서 가장 많이 사용되고 있다[1, 2].

CNN을 수행하기 위해서는 매우 많은 부동소수점 파라미터를 계산에 사용하는 데 부동소수점 연산이 복잡하기 때문에 계산 복잡도를 줄이기 위하여 신경망 내에 존재하는 데이터나 파라미터의 크기를 줄이는 저정밀도 신경망에 대한 연구가 여러

* Department of Computer and Telecomm. Engineering, Yonsei University, Wonju
★ Corresponding author
E-mail : skyun@yonsei.ac.kr, Tel : +82-33-760-2267
※ Acknowledgment
This work was supported by the Yonsei University Wonju Campus Future-Leading Research Initiative of 2017 (2017-52-0074)
Manuscript received Sep. 8, 2018; revised Sep. 18, 2018; accepted Sep. 19, 2018

연구자에 의해서 수행되었다[3, 4].

부동소수점 수를 적은 비트의 고정소수점 수로 양자화시키는 방법들이 연구되었고, 특히 BNN (Binarized Neural Network)은 CNN의 데이터와 가중치(weight)를 모두 +1/-1의 값으로 이진화시킨 구조로 저정밀도 신경망의 가장 극단적인 형태이지만 복잡도가 작은 이미지에 대해서 높은 정확도를 보였다[5]. 그리고 BWN(Binary Weight Network)은 가중치만 +1/-1로 이진화시킨 신경망인데, ImageNet과 같은 복잡도가 큰 이미지에 대해서도 높은 정확도를 보였다[6].

저정밀도 신경망은 파라미터 적재에 필요한 메모리 요구량을 원래의 신경망과 비교할 때에 현저히 감소시키기 때문에 하드웨어 기반의 가속처리에 적합하다. 그래서 BNN을 FPGA기반으로 가속시키는 연구가 많이 수행되었다[7,8,9,10]. BWN은 가중치만 이진화시키고 데이터는 부동소수점 수를 그대로 사용하여 학습시킨다. BWN에 대한 FPGA 구현은 제대로 되어 있지 않으며, 이러한 BWN 기반의 분류를 FPGA를 사용하여 하드웨어로 구현하기 위해서는 구현 복잡도와 메모리 요구량을 줄이기 위해서 학습된 파라미터와 주요연산을 부동소수점 대신에 고정소수점 연산으로 근사화시키는 것이 필요하다.

BWN 연산을 고정소수점 연산 기반 FPGA로 구현할 때에 하드웨어 복잡도를 줄이기 위해서는 원래의 부동소수점 연산과 비교할 때에 정확도의 손실이 크지 않는 최소의 고정소수점 수의 크기를 찾아서 이 크기로 고정소수점 수 연산을 구현하는 것이 필요하다. 이를 위하여 본 연구에서는 작은 크기의 이미지 데이터 집합인 CIFAR-10 [11]을 처리하는 CNN인 ConvNet에서 고정소수점 수의 전체 비트 수와 소수점 이하 비트 수의 변화에 따른 BWN 추론의 정확도의 변화를 실험적으로 분석하여 BWN 연산의 FPGA 구현에 적합한 최소의 고정소수점 크기를 결정하였다.

II. 관련 연구

심층신경망에서 사용하는 전형적인 CNN 구조는 그림 1과 같이 convolution(Conv) 계층과 데이터 양을 축소하는 pooling 계층의 반복으로 구성되며 마지막에 fully connected(FC) 계층들이 이어진다. pooling 계층은 선택적으로 수행되는 maxpooling과 batch 정규화(BN)로 구성된다.

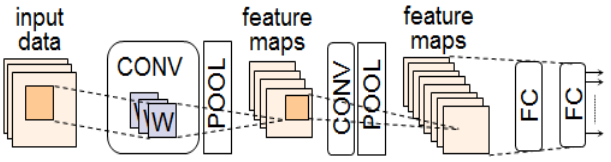

Fig. 1. Typical Deep CNN model structure.
그림 504. 전형적인 심층 CNN 모델 구조

BWN는 일반적인 CNN에서 가중치를 +1/-1로 이진화시킨 저정밀도 신경망이다. 가중치를 이진수로 표현하며, CNN에서의 벡터 내적은 곱셈이 없이 데이터들 간의 덧셈과 뺄셈만으로 계산된다. Rastegari 등은 CIFAR-10을 분류하는 BWN 구조인 ConvNet 구조[6]를 제시하였으며, 표 1과 같은 구조를 갖는다. 여기서 Input feature map의 깊이와 폭은 각각 D와 W_{in}로 나타내고, output feature map의 깊이와 폭은 각각 N과 W_{out}로 나타낸다.

Table 1. ConvNet for BWN implementation.
표 1. BWN 구현을 위한 ConvNet

Layer	D	W_{in}	W_{out}	N
Conv1	3	32	32	128
Conv2	128	32	32	128
Pool2	128	32	16	128
Conv3	128	16	16	256
Conv4	256	16	16	256
Pool4	256	16	8	256
Conv5	256	8	8	512
Conv6	512	8	8	512
Pool	512	8	4	512
FC1	8192	1	1	1024
FC2	1024	1	1	1024
FC3	1024	1	1	10

표 1에서 최초 입력은 32×32 이미지의 R, G, B값으로서 깊이가 3이고 폭이 32이다. 6개의 Conv/Pool 계층 연산을 수행한 후에 출력된 512개의 4×4 데이터는 8192개의 1차원 데이터로 간주되어 3개의 FC 계층 연산을 수행한 후에 10개의 분류 데이터가 최종적으로 출력된다.

III. 고정소수점 형식에 따른 정확도 분석

Tensorflow를 사용하여 BWN기반의 ConvNet를 구현하고, CIFAR-10의 5만개의 학습용 데이터를 사용하여 학습하여 이진가중치 파라미터와 BN용 32비트 부동소수점 파라미터를 얻었다. 그리고 1만개의 테스트용 데이터를 적용하여 학습된 파라미터를 사용하여 테스트를 한 결과 85.77%의 정확도를 얻었다.

고정소수점 수는 전체 비트수와 소수점이하 비트수가 정해져 있다. 32비트 부동소수점을 고정소수점으로 근사시킬 때에 가능한 한 정수부분의 오버플로를 발생하지 않도록 하고 오차를 줄이기 위해서 반올림을 적용하였다.

데이터와 학습된 BN 파라미터의 적절한 비트수를 찾기 위해 Tensorflow내의 Tensorboard 기능을 사용해서 신경망에서의 중간계산값과 activation data의 분포를 확인하였다.

Table 2. Max/min values of intermediate results and activation data.
표 2. 중간계산값과 activation data의 최대/최소 값

Layer	intermediate data		activation data	
	min	max	min	max
Input	-	-	-12.1	9.2
Conv1	-12.1	9.2	0.0	42.2
Conv2	-187.3	172.6	0.0	47.4
Conv3	-2447.7	2380.2	0.0	29.0
Conv4	-2084.0	1831.6	0.0	22.6
Conv5	-2310.9	1754.3	0.0	22.2
Conv6	-1349.5	842.3	0.0	18.6
FC1	-1039.0	1301.5	0.0	9.3
FC2	-286.2	254.1	0.0	11.9
FC3	-61.3	544.3	-11.3	32.8

Layer	BN parameter	min	max
Conv	scale	0.0048	0.1825
	offset	-2.4890	2.2724
FC	scale	0.0153	0.0691
	offset	-4.7207	0.3251

표 2는 10,000개의 CIFAR-10 데이터에 대해서 중간 계산값과 activation 데이터의 최대/최소값과 Batch Normalization(BN) 연산에 필요한 파라미터인 scale과 offset의 최대/최소값을 나타낸 것이다. 여기서 최초의 Input은 정규화를 시킨 CIFAR-10 데이터이며, activation 데이터는 ReLu 계층 이후의 데이터이고, 중간계산값은 2D convolution 연산의 결과로서 Maxpool과 Batch Normalization 연산으로 이어지는 pooling 계층의 입력으로 사용 된다. 이 결과를 참고하여 여러 데이터와 파라미터 크기를 참고하여 정확도 분석에 사용될 고정소수점 수의 크기로 사용하였다.

C언어를 비롯한 대개의 프로그래밍 언어는 고정소수점 형식을 지원하지 않는다. 고정소수점 수의 표현은 C언어에서 정수형으로 표현하였으며, 고정소수점 연산은 32비트 정수 연산을 이용하여 수행하였다. 정수 연산 결과는 고정소수점 형식에 맞도록 조정했다. 이러한 연산을 사용하여 표 1의 ConvNet 신경망의 고정소수점 연산을 수행하는 C 프로그램을 작성하여 형식 변화에 따른 고정소수점 정확도 분석하고 검증하였다.

IV. 고정소수점 형식의 정확도 분석 결과

1. 학습된 파라미터

BWN에서 필요한 파라미터는 이진 가중치와 Batch 정규화 연산에 필요한 scale, offset이다. 이진 가중치는 +1/-1의 값이므로 이진수로 나타낸다. BN 연산의 scale과 offset은 여러 실험을 통해 정확도 손실이 가장 적은 경우를 찾아서 표 3과 같이 고정소수점을 적용하였다. 전체 bit수는 10-bit를 사용하였고 Conv 계층과 FC 계층에 대해서 서로 다른 소수점 위치를 적용하였다.

Table 3. Fraction point location of BN parameters.
표 3. BN parameter 소수점 위치

		total	sign	integer	fraction
Conv	scale	10	1	0	9
	offset	10	1	2	7
FC	scale	10	1	0	9
	offset	10	1	0	9

2 Activation data

표 4은 activation data의 bit수와 소수점 자리에 따른 정확도를 비교한 것이다. 중간계산값은 32bit

를 사용해서 오버플로가 발생하지 않게 하였다. 실험결과 activation data의 정수 bit수는 최소 3bit를 사용해야 정확도 감소가 크지 않은 것을 알 수 있다. 여기서 len은 activation data의 bit수, f는 소수자리 bit수이다.

Table 4. Accuracy comparison by bit size and fraction size of activation data.

표 4. activation data bit수와 소수점에 따른 정확도비교

len,f	5,1	5,2	5,3	5,4	-	-
acc(%)	83.3	84.8	83.7	72.7	-	-
len,f	8,2	8,3	8,4	8,5	8,6	8,7
acc(%)	84.7	85.1	84.9	85.0	84.1	74.1
len,f	10,4	10,5	10,6	10,7	10,8	10,9
acc(%)	84.9	85.0	84.9	85.0	84.1	74.3

3 중간계산값

표 5는 중간계산값의 bit수에 변화를 주면서 정확도를 비교한 것이다. activation data=(l,f)에서 l은 activation data의 bit수이며 f는 소수자리의 bit수이다. 실험결과 중간계산값의 정수자리 bit수는 최소 11bit (소수자리 포함 14bit)가 필요한 것을 알 수 있다. 중간계산값의 bit수가 충분하지 않을 경우 덧셈을 할 때 오버플로우에 의한 정보 손실이 발생하고 정확도가 감소하는 것을 알 수 있다.

Table 5. Accuracy comparison by bit size of intermediate values.

표 5 중간계산 값의 bit수에 따른 정확도 비교

	activation data=(8,3)				
intermediate data bit size	16	14	13	12	11
accuracy(%)	85.1	85.0	56.4	0.09	0.09

같은 구조의 신경망에 대해서 CIFAR-10 데이터에 대해서 이진화되지 않은 CNN을 사용하는 경우에는 88.0%의 정확도를 보이고 부동소수점 연산을 사용하는 BWN에서는 85.7%의 정확도를 보여서 약간의 정확도 손실이 발생한다. 고정소수점 연산을 사용하는 BWN은 85.1%의 정확도를 보여서 부동소수점 연산에 비해 0.6%의 정확도 손실이 발생하며, 고정소수점 크기를 잘 정하면 정확도가 거의 같음을 확인할 수 있다.

4. 오버플로 제어에 따른 정확도 변화

중간계산값의 bit수를 정할 때 오버플로가 발생하지 않는 최소의 bit수를 사용하는 방법을 사용하였다. 따라서 그 최소 bit수보다 낮은 bit수를 사용할 경우 정확도가 급격히 감소하게 된다. 이러한 중간계산값 bit수를 적용했을 때 오버플로가 발생했을 경우 최대 또는 최소값으로 만들어 주어서 오차를 최소화해주는 방법을 적용하여 실험해보았다. 2D 컨볼루션 연산의 결과값을 누적시켜 더할 때 이 오버플로 제어를 하였고 표 6에 오버플로 제어를 한 것과 하지 않은 것의 비교결과를 나타내었다. 실험결과 오버플로 제어를 하지 않았을 때 정확도 감소가 급격히 줄어든 것을 알 수 있다.

Table 6. Accuracy comparison by overflow control(OVC) of intermediate value.

표 6. 중간계산값 오버플로우 제어 방법에 따른 정확도비교(OVC는 OVerflow Control 이다)

No OVC	intermediate data bit size	16	14	12	11
	accuracy(%)	85.1	85.0	0.09	0.09
OVC	intermediate data bit size	16	14	12	11
	accuracy(%)	85.1	85.1	84.5	75.5

V. 결론 및 향후 연구방향

본 연구에서는 부동소수점으로 BWN을 학습시킨 뒤 학습된 파라미터와 주요연산을 고정소수점으로 근사화시키는 과정에서 정확도 변화를 분석하였다. 학습된 파라미터 중 Batch 정규화 파라미터는 값이 전반적으로 작았기 때문에 소수 부분의 bit수를 크게 적용하였다. BWN 연산과정에 고정소수점 연산을 적용하였고 사용하는 데이터의 bit수와 소수점 위치에 따라 정확도 변화를 확인하는 실험을 하였다. 부동소수점을 사용한 BWN에서는 정확도가 85.7%이며 고정소수점을 적용하고 activation 데이터도 소수점이하 3자리, 전체 8자리를 사용하고 중간계산값은 16bit를 사용하였을 때의 정확도는 85.1%였다. 따라서 고정소수점을 사용하되 중간계산을 할 때에 덧셈/뺄셈에 필요한 충분한 크기의 정수 bit수를 적용해 준다면 큰 정확도 손실이 없는 것을 확인하였다.

본 연구에서 확인한 데이터의 bit수에 대한 실험 결과를 FPGA기반 BWN가속기를 설계하는 데 반영하면 메모리 요구량과 하드웨어 요구량이 감소한다.

References

[1] A. Krizhevsky, I. Sutskeve, and G.E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, pp.1097-1105, 2012. DOI:10.1145/3065386

[2] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint* arXiv:1409.1556, 2014.

[3] D. D. Lin and S. S. Talathi, "Overcoming challenges in fixed point training of deep convolutional networks," *arXiv preprint* arXiv: 1607.02241, 2016.

[4] D. Miyashita, E.H. Lee, and B. Murmann, "Convolutional neural networks using logarithmic data representation," *arXiv preprint* arXiv: 1603.01025, 2016.

[5] M. Courbariaux, et al., "Binarized neural networks: Training deep neural networks with weights and activations constrained to +1 or -1," *arXiv preprint* arXiv:1602.02830, 2016.

[6] M. Rastegari, et al., "XNOR-Net: ImageNet Classification Using Binary Convolutional Neural Networks," in *European Conf. on Computer Vision* (ECVV'16), Springer, pp.525-542, 2016.

[7] R. Zhao, W. Song. "Accelerating binarized convolutional neural networks with software programmable FPGAs," in *Proc. ACM/SIGDA Int. Symp. on Field-Programmable Gate Arrays*, ACM, pp.15-24, 2017.DOI;10.1145/3020078.3021741

[8] Y. Umuroglu, et al., "FINN: a framework for fast, scalable binarized neural network inference," in *Proc. ACM/SIGDA Int. Symp. on Field-Programmable Gate Arrays*, ACM, pp.65-74, 2017. DOI:10.1145/3020078.3021744

[9] S. Liang, et al., "FP-BNN: Binarized Neural Network on FPGA," *Neurocomputing* 275, pp. 1072-1086, 2018. DOI:10.1016/j.neucom.2017.09.046

[10] R. Andri, et al. "YodaNN: An ultra-low power convolutional neural network accelerator based on binary weights," *IEEE Computer Society Annual Symp. on VLSI* (ISVLSI'16), pp.236-241, 2016. DOI:10.1109/ISVLSI.2016.111

[11] CIFAR-10 and CIFAR-100 datasets, https://www.cs.toronto.edu/~kriz/cifar.html

BIOGRAPHY

Jong-Hyun Kim (Member)

2016 : BS degree in Biomedical Eng.ineering, Yonsei University.
2018 : MS degree in Computer Science, Yonsei University.
2018.8~ : Researcher, R&D Center, Telechips Inc.

SangKyun Yun (Member)

1984 : BS degree in Electronics Eng.ineering, Seoul National University.
1986 : MS degree in Electrical Engineering, KAIST.
1995 : PhD degree in Electrical Engineering, KAIST.
1992~2001 : Professor, Department of Computer Science, Seowon University.
2001~ : Professor, Department of Comptuer and Telecom. Engineering, Yonsei Univ. Wonju.