Duplicate Max-pooling 기반 이미지 분류 경량 모델

*김상훈 *김원준

건국대학교 전기전자공학부

*lalabo918@konkuk.ac.kr *wonjkim@konkuk.ac.kr

A Light-weight Model Based on Duplicate Max-pooling for Image Classification

*Sanghoon Kim *Wonjun Kim

Electrical and Electronics Engineering Konkuk University

*lalabo918@konkuk.ac.kr *wonjkim@konkuk.ac.kr

요약

고성능 딥러닝 모델은 학습과 추론 과정에서 고비용의 전산 자원과 많은 연산량을 필요로 하여 이에 따른 개발 환경과 많은 학습 시간을 필요로 하여 개발 지연과 한계가 발생한다. 따라서 HW 또는 SW 개선을 통해 파라미터 수, 학습 시간, 추론시간, 요구 메모리를 줄이는 연구가 지속 되어 왔다. 본 논문은 EfficientNet에서 사용된 Linear Bottleneck을 변경하여 정확도는 소폭 감소 하지만 기존 모델의 파라미터를 55%로 줄이는 경량화 모델을 제안한다.

I. 서론

CNN(Convolutional Neural Network)를 통한 이미지 분류 연구는 정확도를 향상을 위해서 크고 깊은 모델이 연구되어 왔다. 모델이 크고 깊어지는 속도에 비해 GPU의 발전은 비교적 더뎌 예산이 부족한 기업이나 대학에서는 점차 어려운 과제가 되었다. 이러한 큰 모델은 임베디드 시스템에 적용하기 적절하지 않다. 최근에는 정확도 외에 효율성을 중요시하는 경량화에 대한 연구가 지속되고 있다.

최근에는 사람이 아닌 시스템이 최적의 딥러닝 모델을 찾는 AutoML이라는 연구가 진행됐다. NASNet[1], mNASNet이 이에 해당하고 Compound Scaling을 통해 효율적으로 모델을 키우는 EfficientNet[2]은 mNASNet을 기본 모델로 한다.

EfficientNet는 Inverted Residual 구조에 Linear Bottleneck을 사용해 확장과 축소를 하였고, 중간에는 Depthwise Convolution과 Squeeze and Excitation 연산을 사용했다. 효율적인 구조로 되어 있으나 Linear Bottleneck은 모델이 깊어짐에 따라 채널이 많아져 파라미터 수가 급격하게 늘어나며 비중도 과도하게 크다.

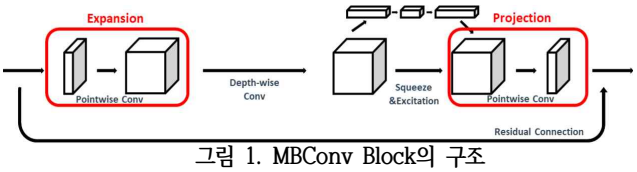

그림 1. MBConv Block의 구조

Linear Bottleneck의 파라미터를 줄이기 위해 EfficientNet에 사용된 MBConv(Mobile Inverted Bottleneck Convolution) Block(그림1)의 구조를 보면 Linear Bottleneck을 제외하고는 채널 간의 연산은 없다. Depthwise Convolution은 채널별로 Convolution을 진행하고, Squeeze and Excitation은 채널마다 다른 값을 곱해준다.

Convolution을 대체하기 위해 Projection을 Expansion 했던 비율로 Maxpooling한 결과 파라미터는 기존 모델 대비 77%로 줄였으나, 성능은 많이 감소하지 않았다. 채널을 늘리는 방법으로 단순 복제한 결과 파라미터는 기존 모델 대비 55%로 줄이면서 Maxpooling만 적용했을 때보다 성능이 소폭 상승했다.

본 논문에서는 Linear Bottleneck 대신 파라미터를 줄이면서 성능을 최대한 유지 시키는 방법을 제안한다. 제안하는 방법은 Linear Bottleneck 대신 채널을 복제시켜 연산을 거치고 그 채널들을 Maxpooling 하는 구조이다. 이를 통해 분류의 성능은 소폭 감소했으나, 기존 모델의 파라미터를 55%로 줄일 수 있었다. 추가로 Duplicate와 Maxpooling을 따로 적용했을 때보다 같이 적용했을 때 파라미터가 감소하는 동시에 성능이 향상했다.

II. 제안하는 방법

Linear Bottleneck은 Expansion과 Projection 모두 Pointwise Convolution을 사용하기 때문에 총 파라미터 수는 $2 \times C_{in} \times C_{out}$다. C_{in}은 Block Input 채널 수, C_{out}은 확장된 채널 수다.

본 논문은 Linear Bottleneck에서 Pointwise Convoltion 대신 Duplicate Maxpooling 사용하여 확장과 축소한다. 이 방법을 사용하면 파라미터가 없어 기존 Block 대비 $2 \times C_{in} \times C_{out}$ 만큼 줄어든다. 이 방법의 구조는 그림 2와 같이 나타낼 수 있다.

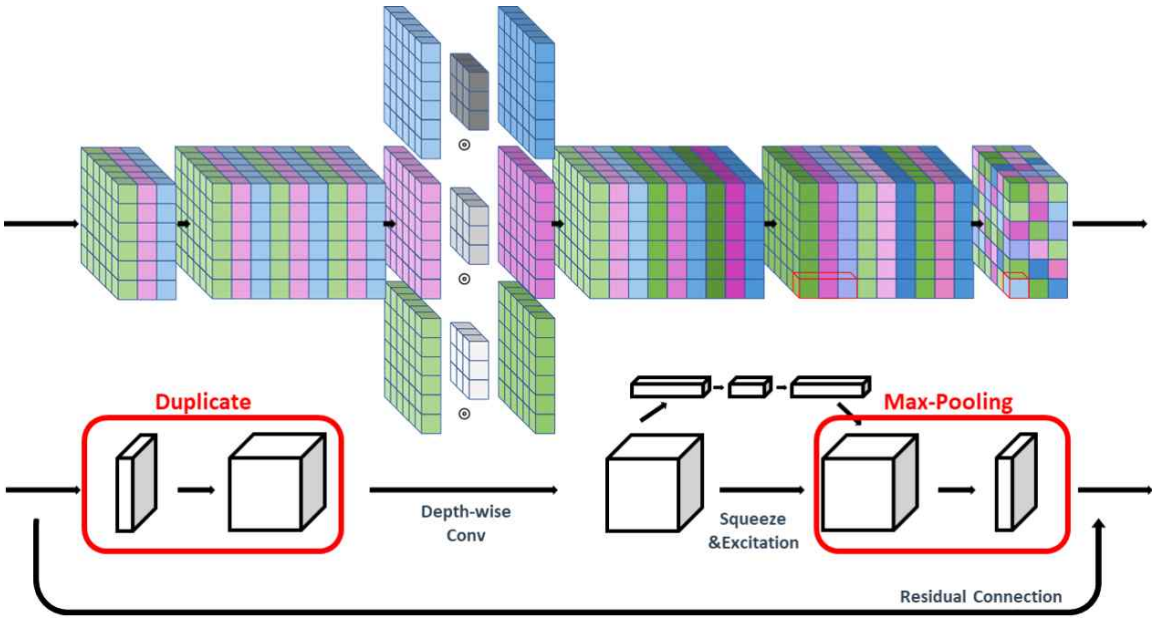

그림 2. Duplicate Max-pooling 구조

Ⅲ. 구현

본 논문은 Pytorch를 이용하여 구현하였으며, 성능 평가를 위해 CIFAR10, CIFAR100, ImageNet10 데이터 셋을 사용하였다. 손실 함수로는 Cross Entropy, Optimizer은 Adam, Learning rate는 0.01로 시작하고 Scheduler는 CosineAnnealingLR을 사용해 학습했다.

EfficientNetB0을 기본 모델로 Input, Output 채널 수가 같은 Block만 대체했다.

Network	# of Params
EfficientNetB0	4,020,358
Max-pooling Only	3,114,374
Duplicate Only	3,104,134
Duplicate Max-pooling (Ours)	2,198,150

표 1. 각 모델의 파라미터 개수(Class 10개 기준)

Network	ImageNet10	CIFAR10	CIFAR100
EfficientNetB0	88.66%	93.36%	74.38%
Only Max-pooling	89.25%	89.12%	68.25%
Only Duplicating	88.79%	89.29%	68.30%
Duplicate Max-pooling	89.35%	89.33%	67.94%

표 2. 각 데이터 셋 학습에 따른 성능(정확도) 비교

Ⅳ. 결론 및 향후 연구 방향

표 1과 표 2에서 볼 수 있듯이, Duplicate Max-pooling을 통해 EfficientNet의 파라미터 개수를 약 45% 감소시킬 수 있으며 해상도가 높은 모든 데이터 셋에서는 성능이 소폭 상승하기도 했다. 이 방법은 Linear Bottleneck을 사용하는 모델에 적용하여 경량화할 수 있다.

감사의 글

본 연구는 2021년도 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학지원사업의 결과로 수행되었음 (No.2018-0-00213, SW중심대학(건국대학교)).

참고문헌

[1] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, “Learning transferable architectures for scalable image recognition,” in *Proc. IEEE Int. Conf. Compute. Vis. Pattern Recognit. (CVPR)*, pp. 8697-8710, Jun. 2018.

[2] M. Tan and Q. Le, “EfficientNet: Rethinking model scaling for con-volutional neural networks,” in *Proc. 36th International Conference on Machine Learning (ICML)*, pp. 6105-6114, Jun. 2019.