CVPR 2020
EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks

2022.08.01

논문 리뷰

배성훈

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

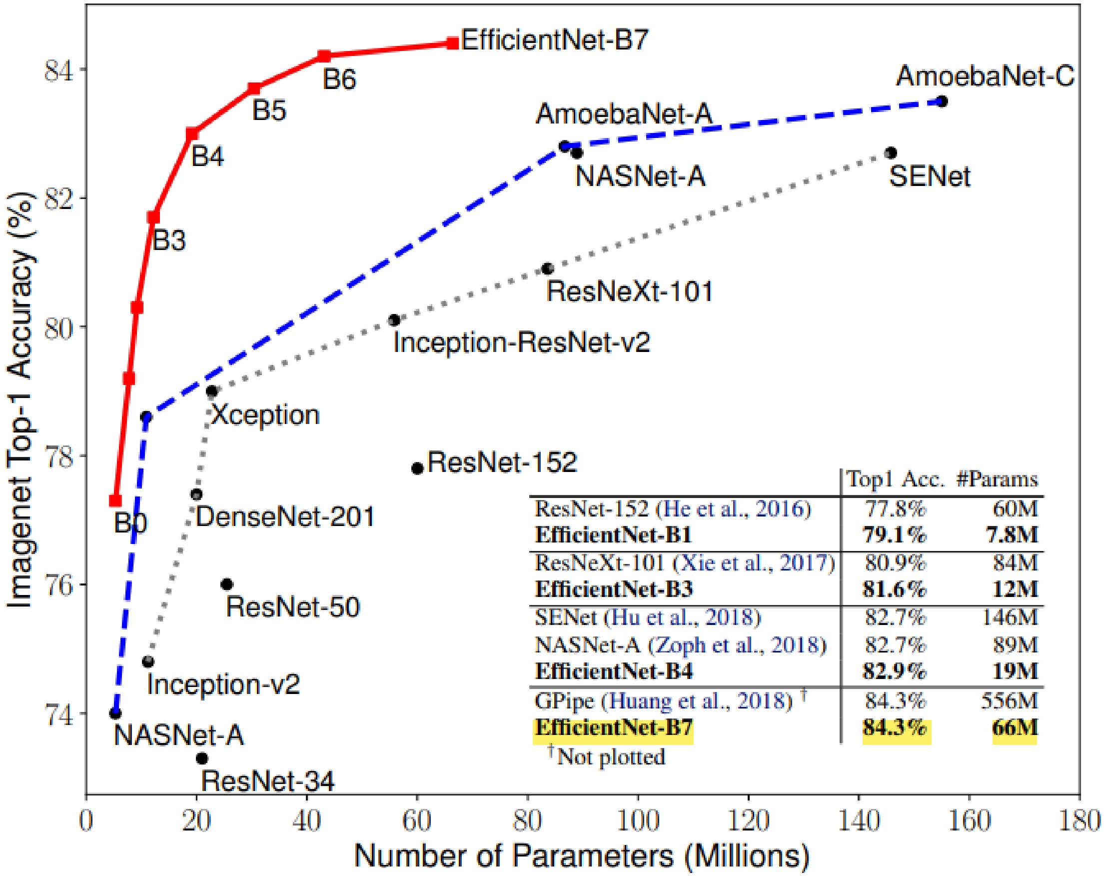
- **Research Background:**
  - 기존의 연구는 CNN 정확도를 높이기 위해 <u>Depth, Width, Resolution 중 한 가지 방법</u>만을 선택해 성능을 향상
  - 이에 따라 저자는 Convolution Network의 **정확성**과 **효율성**을 향상시키는 원칙에 따른 **Scale up 방법**에 대한 의문
  - **<u>Compound Coefficient</u>**를 통해 **<u>Depth, Width, Resolution을 균형 있게 Scaling</u>**해 성능 향상

*__Depth__: Layer 수 증가

*__Width__: Filter(channel 수) 증가

*__Resolution__: Input image의 해상도(크기) 증가

Model Size vs ImageNet Accuracy

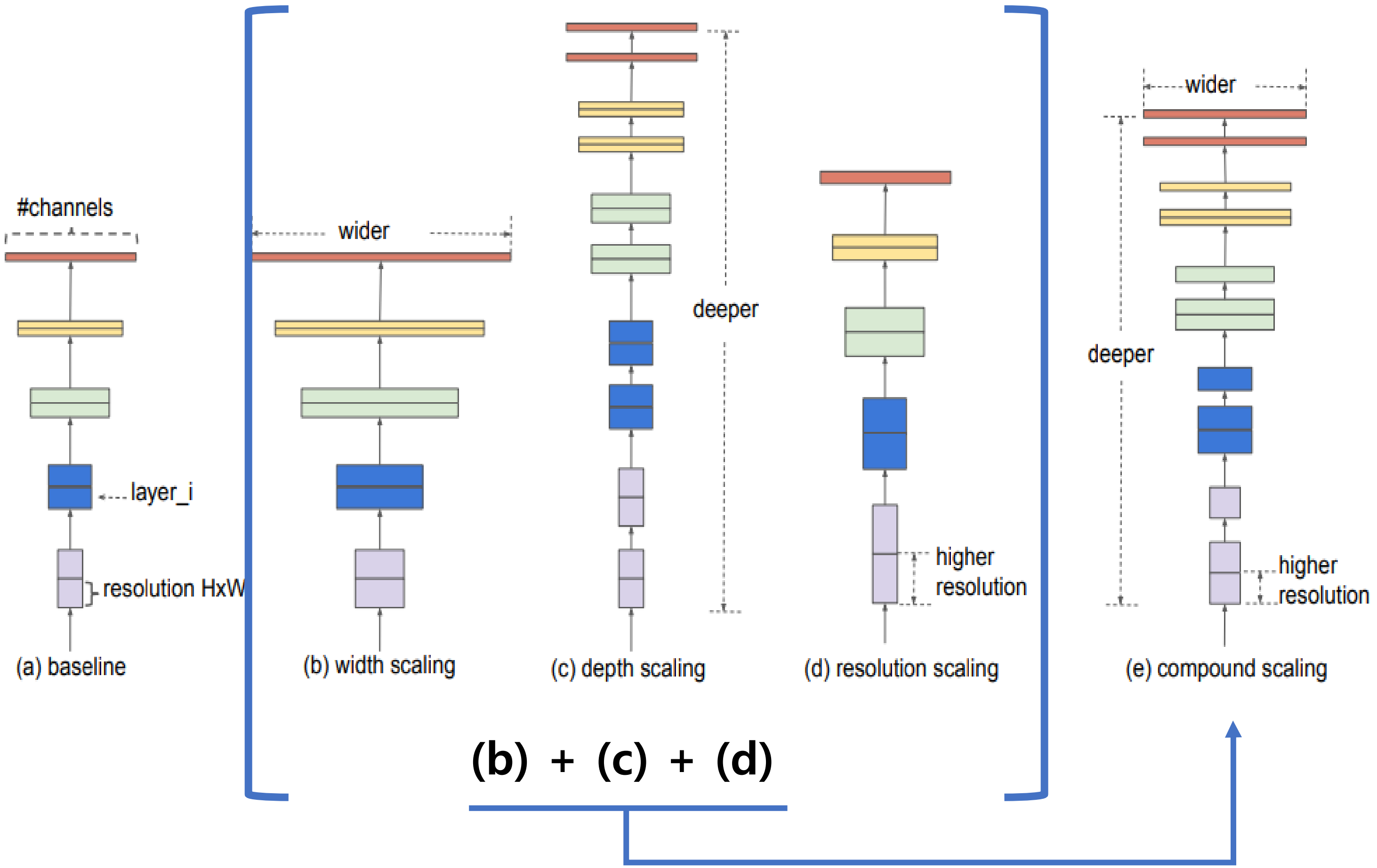


	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**79.1%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.6%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**82.9%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.3%**	**66M**

†Not plotted

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

- **Method:**
  - Compound Scaling: Small Grid Search에 의해 결정되는 Width, Depth, Resolution scaling을 위한 **Constant coefficients**(α, β, γ)
  - 각 Layer에서 수행하는 연산 F는 고정해 Architecture 설계를 단순화
  - 제한된 Resource 환경에서 모델의 **정확도는 최대한 높이면서, 연산량은 최대한 줄임**



(a) Standard Convolution

$$\boldsymbol{N} = \odot_{i=1\ldots s} \underline{\boldsymbol{F}_i}^{L_i}(\boldsymbol{X}_{<H_i, W_i, C_i>})$$

연산 F 고정

(e) Compound Scaling

$$\max_{d,w,r} \quad Accuracy(\mathcal{N}(d, w, r))$$

$$s.t. \quad \mathcal{N}(d, w, r) = \bigodot_{i=1\ldots s} \hat{\mathcal{F}}_i^{d \cdot \hat{L}_i}(X_{\langle r \cdot \hat{H}_i, r \cdot \hat{W}_i, w \cdot \hat{C}_i \rangle})$$

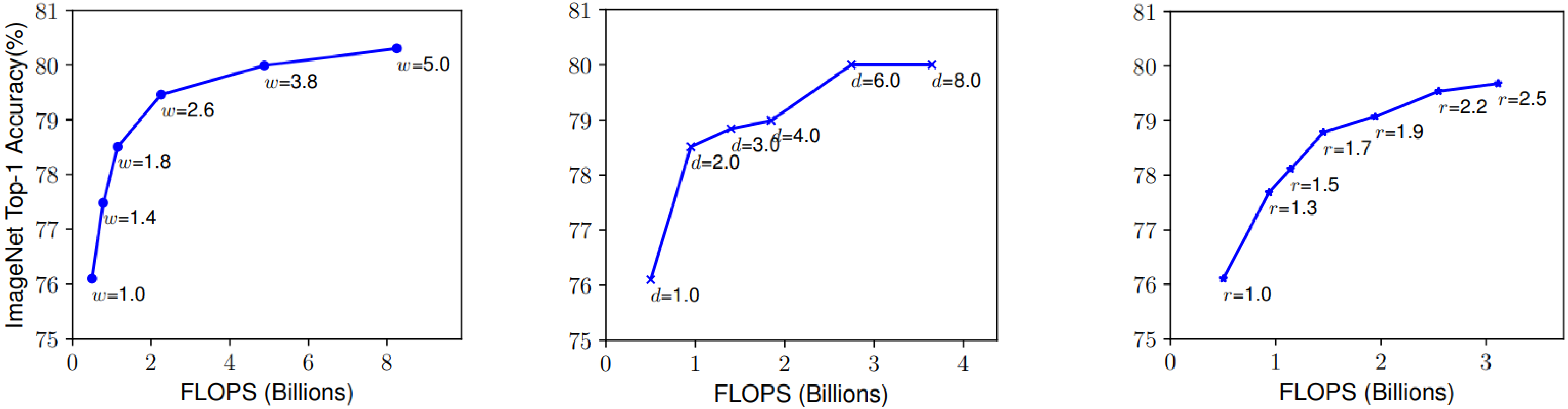
$$\left.\begin{array}{l} \text{Memory}(\mathcal{N}) \leq \text{target\_memory} \\ \text{FLOPS}(\mathcal{N}) \leq \text{target\_flops} \end{array}\right\} \text{제한된 Resource}$$

$\boldsymbol{d, w, r}$ 은 Network Width, Depth, Resolution scaling을 위한 **coefficients**

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

- **Method:**
  - **<Compound Scaling 단계>**
    Step 1. ∅ =1로 고정한 상태로, 앞서 정의한 수식에 **α, β, γ** 을 순차적으로 입력해 최적 값을 구함
    Step 2. step 1에서만 진행해 얻어진 α, β, γ 고정하고 ∅를 임의적으로 입력해 비교

Scaling Up a Baseline Model with Different Network Width (w), Depth (d), and Resolution (r) Coefficients



w ↑
Fine grained feature 잘 포착
학습이 쉬움
Higher level feature 포착이 어려움

d ↑
Rich, complex feature 잘 포착
Vanishing gradient 증가
Accuracy gain 감소

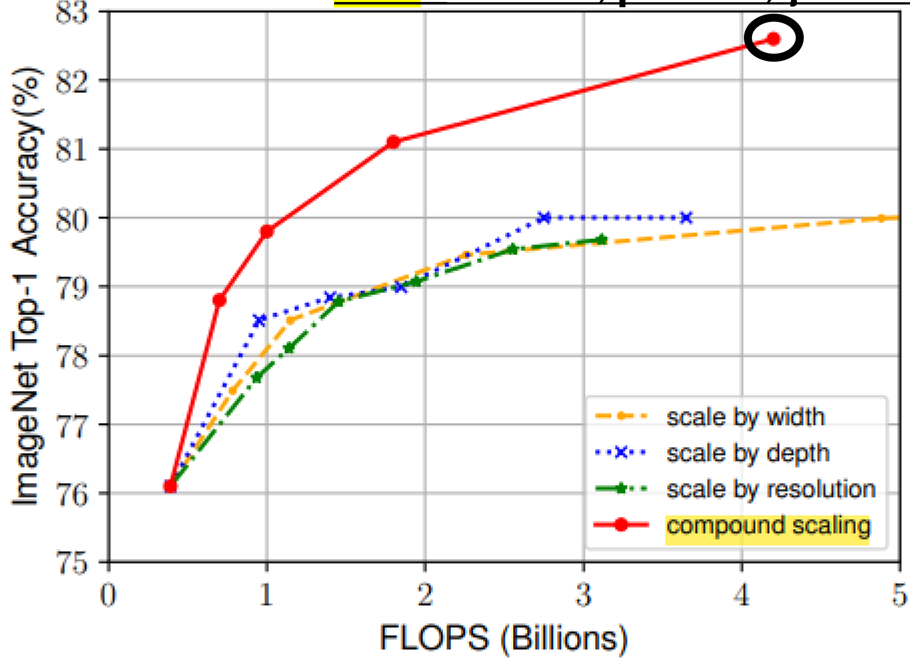
r ↑
Accuracy gain 감소

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

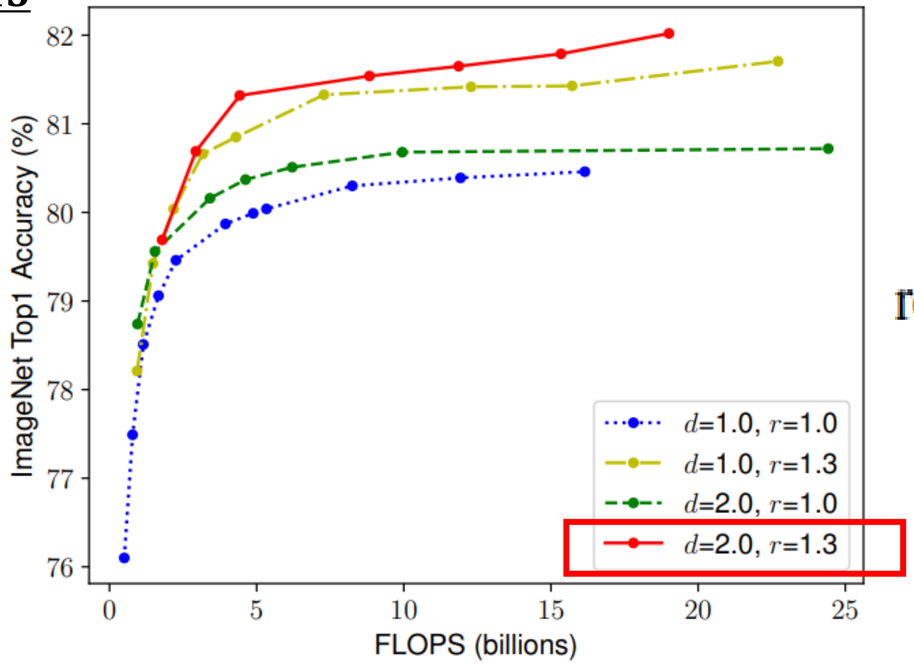
- **Method:**
  - 3개의 dimension은 <u>독립적이지 않음</u> **=> w, d, r** 각각에 다른 값을 scaling해 균형을 맞출 필요가 있음
  - 네트워크가 깊어지고, 해상도가 높아질수록 <u>**정확도 향상**</u>
  - 논문은 $\alpha \cdot \beta^2 \cdot \gamma^2$ = 2로 제한, 제한된 범위에서 <u>**Compound Coefficient ∅**</u>를 사용해 <u>α, β, γ Scaling</u>함

Scaling Network Factor vs Compound Scaling

<u>**Best**: $\alpha = 1.2; \beta = 1.1; \gamma = 1.15$</u>



Scaling Network Width for Different Baseline Networks



정확도 향상

Compound Scaling

$$\text{depth: } d = \alpha^{\phi}$$

$$\text{width: } w = \beta^{\phi}$$

$$\text{resolution: } r = \gamma^{\phi}$$

$$\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$

$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

- **Method:**
  - MNasNet와 유사한 **Neural Architecture Search** 접근 방식을 사용
  - **Accuracy**와 **FLOPS**를 모두 최적화하는 multi-objective search를 활용

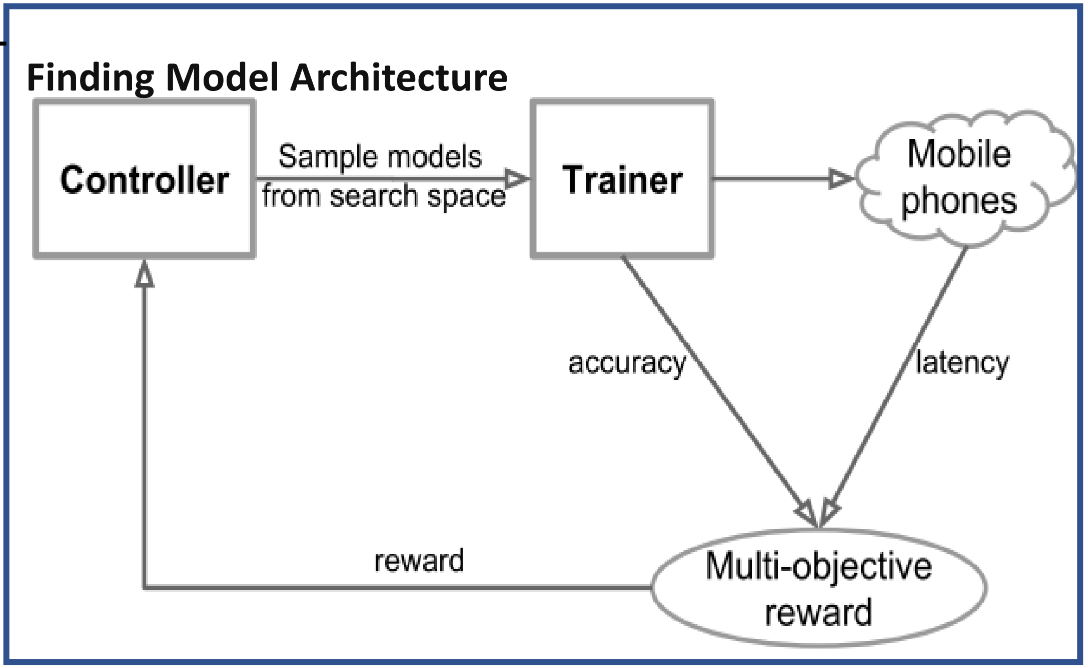
MNasNet Approach

Model architecture :
ImageNet에서 학습하는 데 사용
<u>Accuracy and Latency 계산</u>

$$\underset{m}{\text{maximize}} \quad ACC(m) \times \left[\frac{LAT(m)}{T}\right]^{w}$$

where w is the weight factor defined as:

$$w = \begin{cases} \alpha, & \text{if } LAT(m) \leq T \\ \beta, & \text{otherwise} \end{cases}$$



Latency가 특정 지정된 값보다 낮을 때 Accuracy가
최대가 되도록 최적의 아키텍처가 달성될 때까지 반복.

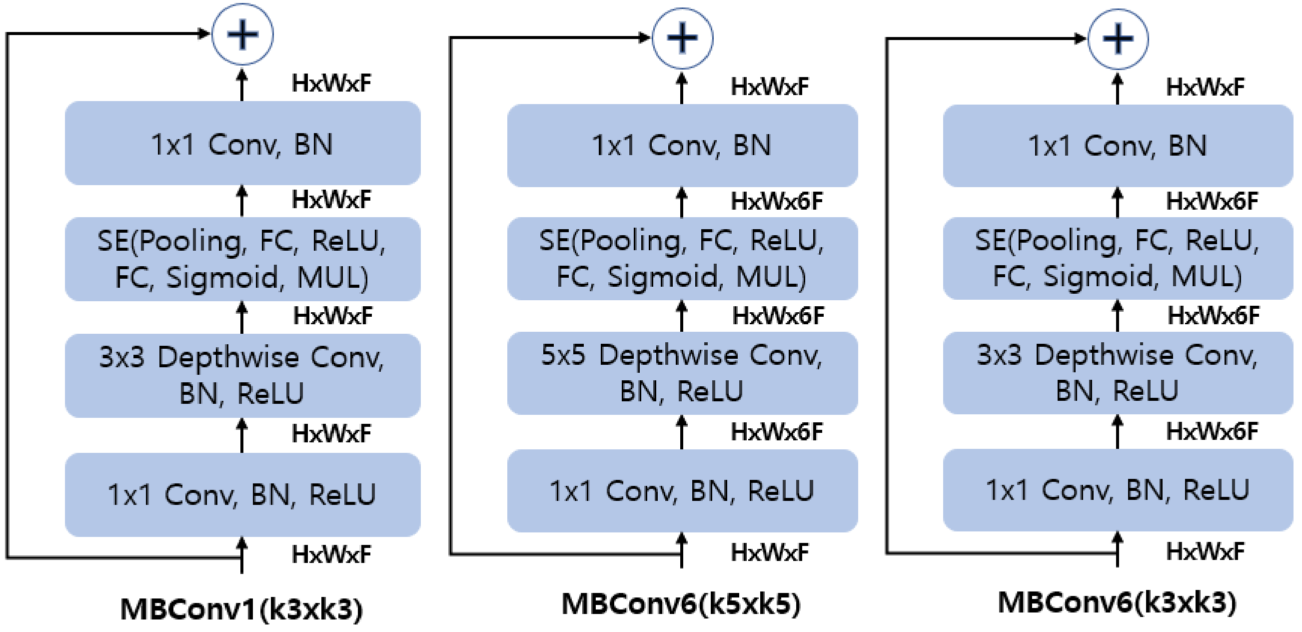
Model accuracy (on ImageNet)와 latency을
모델 목표로 사용해 <u>**Best architecture를 찾음**</u>

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

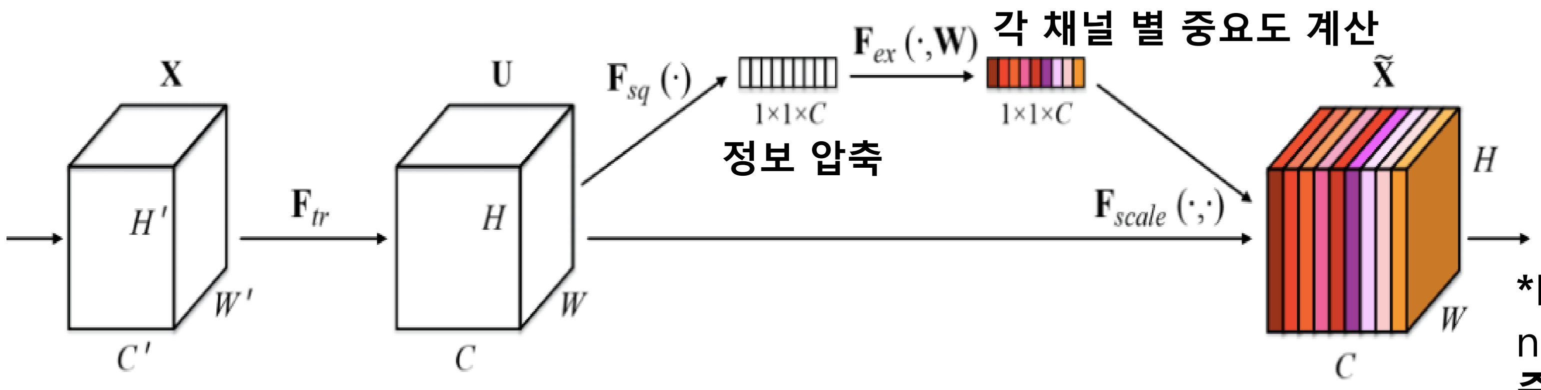
- **Method:**
  - NAS에서 정확도 및 FLOPS를 모두 챙길 수 있도록 최적화한 **EfficientNet**

EfficientNet-B0 Baseline Network (ø 의 값에 따라 EfficientNet-B0, 1, …, 7로 분류)

Stage i	Operator $\hat{\mathcal{F}}_i$	Resolution $\hat{H}_i \times \hat{W}_i$	#Channels $\hat{C}_i$	#Layers $\hat{L}_i$
1	Conv3x3	224 × 224	32	1
2	MBConv1, k3x3	112 × 112	16	1
3	MBConv6, k3x3	112 × 112	24	2
4	MBConv6, k5x5	56 × 56	40	2
5	MBConv6, k3x3	28 × 28	80	3
6	MBConv6, k5x5	14 × 14	112	3
7	MBConv6, k5x5	14 × 14	192	4
8	MBConv6, k3x3	7 × 7	320	1
9	Conv1x1 & Pooling & FC	7 × 7	1280	1



Squeeze-and-excitation optimization (SE) 추가



***Excitation:** 압축된 정보를 weighted layer와 non-linear activation function으로 **각 채널 별 중요도**를 계산해 기존 input에 곱을 해주는 방식

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

- **Experiment:**
  - EfficientNet-B0 ~ B7 까지의 성능을 SOTA 모델과 비교
  - 같은 성능 대비 Parameters와 FLOPS가 EfficientNet에서 더 낮음 (파라미터 수 **약 8.4배 적음**)

EfficientNet Performance Results on ImageNet

Model	Top-1 Acc.	Top-5 Acc.	#Params	Ratio-to-EfficientNet	#FLOPs	Ratio-to-EfficientNet
EfficientNet-B0	**77.1%**	**93.3%**	**5.3M**	**1x**	**0.39B**	**1x**
ResNet-50 (He et al., 2016)	76.0%	93.0%	26M	4.9x	4.1B	11x
DenseNet-169 (Huang et al., 2017)	76.2%	93.2%	14M	2.6x	3.5B	8.9x
EfficientNet-B1	**79.1%**	**94.4%**	**7.8M**	**1x**	**0.70B**	**1x**
ResNet-152 (He et al., 2016)	77.8%	93.8%	60M	7.6x	11B	16x
DenseNet-264 (Huang et al., 2017)	77.9%	93.9%	34M	4.3x	6.0B	8.6x
Inception-v3 (Szegedy et al., 2016)	78.8%	94.4%	24M	3.0x	5.7B	8.1x
Xception (Chollet, 2017)	79.0%	94.5%	23M	3.0x	8.4B	12x
EfficientNet-B2	**80.1%**	**94.9%**	**9.2M**	**1x**	**1.0B**	**1x**
Inception-v4 (Szegedy et al., 2017)	80.0%	95.0%	48M	5.2x	13B	13x
Inception-resnet-v2 (Szegedy et al., 2017)	80.1%	95.1%	56M	6.1x	13B	13x
EfficientNet-B3	**81.6%**	**95.7%**	**12M**	**1x**	**1.8B**	**1x**
ResNeXt-101 (Xie et al., 2017)	80.9%	95.6%	84M	7.0x	32B	18x
PolyNet (Zhang et al., 2017)	81.3%	95.8%	92M	7.7x	35B	19x
EfficientNet-B4	**82.9%**	**96.4%**	**19M**	**1x**	**4.2B**	**1x**
SENet (Hu et al., 2018)	82.7%	96.2%	146M	7.7x	42B	10x
NASNet-A (Zoph et al., 2018)	82.7%	96.2%	89M	4.7x	24B	5.7x
AmoebaNet-A (Real et al., 2019)	82.8%	96.1%	87M	4.6x	23B	5.5x
PNASNet (Liu et al., 2018)	82.9%	96.2%	86M	4.5x	23B	6.0x
EfficientNet-B5	**83.6%**	**96.7%**	**30M**	**1x**	**9.9B**	**1x**
AmoebaNet-C (Cubuk et al., 2019)	83.5%	96.5%	155M	5.2x	41B	4.1x
EfficientNet-B6	**84.0%**	**96.8%**	**43M**	**1x**	**19B**	**1x**
EfficientNet-B7	**84.3%**	**97.0%**	**66M**	**1x**	**37B**	**1x**
GPipe (Huang et al., 2018)	84.3%	97.0%	557M	8.4x	-	-

Scaling Up MobileNets and ResNet

Better Performance

Single-dimension Scaling vs Compound Scaling

Model	FLOPS	Top-1 Acc.
Baseline MobileNetV1 (Howard et al., 2017)	0.6B	70.6%
Scale MobileNetV1 by width (w=2)	2.2B	74.2%
Scale MobileNetV1 by resolution (r=2)	2.2B	72.7%
compound scale (d=1.4, w=1.2, r=1.3)	**2.3B**	**75.6%**
Baseline MobileNetV2 (Sandler et al., 2018)	0.3B	72.0%
Scale MobileNetV2 by depth (d=4)	1.2B	76.8%
Scale MobileNetV2 by width (w=2)	1.1B	76.4%
Scale MobileNetV2 by resolution (r=2)	1.2B	74.8%
MobileNetV2 compound scale	**1.3B**	**77.4%**
Baseline ResNet-50 (He et al., 2016)	4.1B	76.0%
Scale ResNet-50 by depth (d=4)	16.2B	78.1%
Scale ResNet-50 by width (w=2)	14.7B	77.7%
Scale ResNet-50 by resolution (r=2)	16.4B	77.5%
ResNet-50 compound scale	**16.7B**	**78.8%**

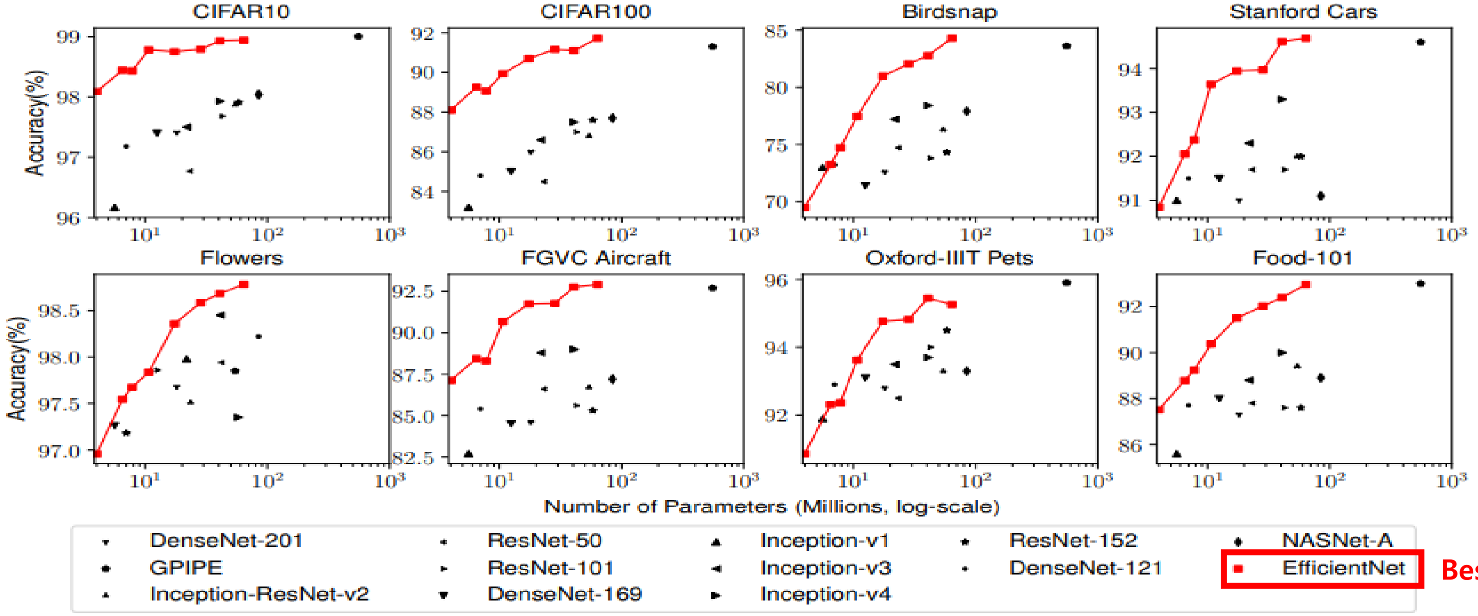
EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

- **Experiment:** 기존의 Top level 모델보다 적은 Parameters와 FLOPS로도 좋은 성능을 보임

EfficientNet Performance Results on Transfer Learning Datasets

	Comparison to best public-available results						Comparison to best reported results					
	Model	Acc.	#Param	Our Model	Acc.	#Param(ratio)	Model	Acc.	#Param	Our Model	Acc.	#Param(ratio)
CIFAR-10	NASNet-A	98.0%	85M	EfficientNet-B0	98.1%	4M (21x)	†Gpipe	**99.0%**	556M	EfficientNet-B7	98.9%	64M (8.7x)
CIFAR-100	NASNet-A	87.5%	85M	EfficientNet-B0	88.1%	4M (21x)	Gpipe	91.3%	556M	EfficientNet-B7	**91.7%**	64M (8.7x)
Birdsnap	Inception-v4	81.8%	41M	EfficientNet-B5	82.0%	28M (1.5x)	GPipe	83.6%	556M	EfficientNet-B7	**84.3%**	64M (8.7x)
Stanford Cars	Inception-v4	93.4%	41M	EfficientNet-B3	93.6%	10M (4.1x)	‡DAT	**94.8%**	-	EfficientNet-B7	94.7%	-
Flowers	Inception-v4	98.5%	41M	EfficientNet-B5	98.5%	28M (1.5x)	DAT	97.7%	-	EfficientNet-B7	**98.8%**	-
FGVC Aircraft	Inception-v4	90.9%	41M	EfficientNet-B3	90.7%	10M (4.1x)	DAT	92.9%	-	EfficientNet-B7	**92.9%**	-
Oxford-IIIT Pets	ResNet-152	94.5%	58M	EfficientNet-B4	94.8%	17M (5.6x)	GPipe	**95.9%**	556M	EfficientNet-B6	95.4%	41M (14x)
Food-101	Inception-v4	90.8%	41M	EfficientNet-B4	91.5%	17M (2.4x)	GPipe	93.0%	556M	EfficientNet-B7	**93.0%**	64M (8.7x)
Geo-Mean						**(4.7x)**						**(9.6x)**

Model Parameters vs. Transfer Learning Accuracy



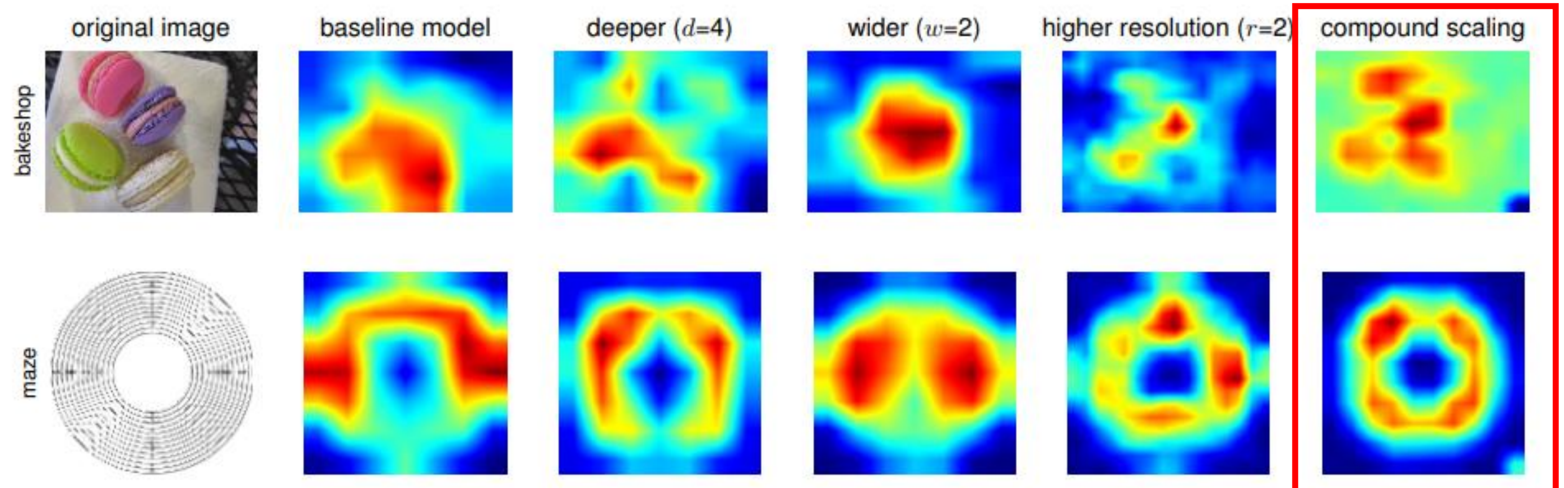
Best Performance

EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks (CVPR 2020)

- **Experiment:**
  - Class Activation Map을 시각화 했을 때 Width, Height, Resolution을 각각 따로 Scaling up한 것 보다 Compound Scaling 했을 때 좀 더 객체들을 잘 담고 있고 정확함

EfficientNet Performance Results on Transfer Learning Datasets

Better Performance



Scaled Models

Model	FLOPS	Top-1 Acc.
Baseline model (EfficientNet-B0)	0.4B	77.3%
Scale model by depth (d=4)	1.8B	79.0%
Scale model by width (w=2)	1.8B	78.9%
Scale model by resolution (r=2)	1.9B	79.1%
Compound Scale (d=1.4, w=1.2, r=1.3)	**1.8B**	**81.1%**

한줄평: 기존의 CNN Scaling 방식을 효과적인 Scaling 방법인 Compound Scaling을 제안해 성능을 효율적으로 향상했다. 단순하면서 직관적인 방법으로 좋은 성능을 달성한 만큼 인상 깊은 논문이다.