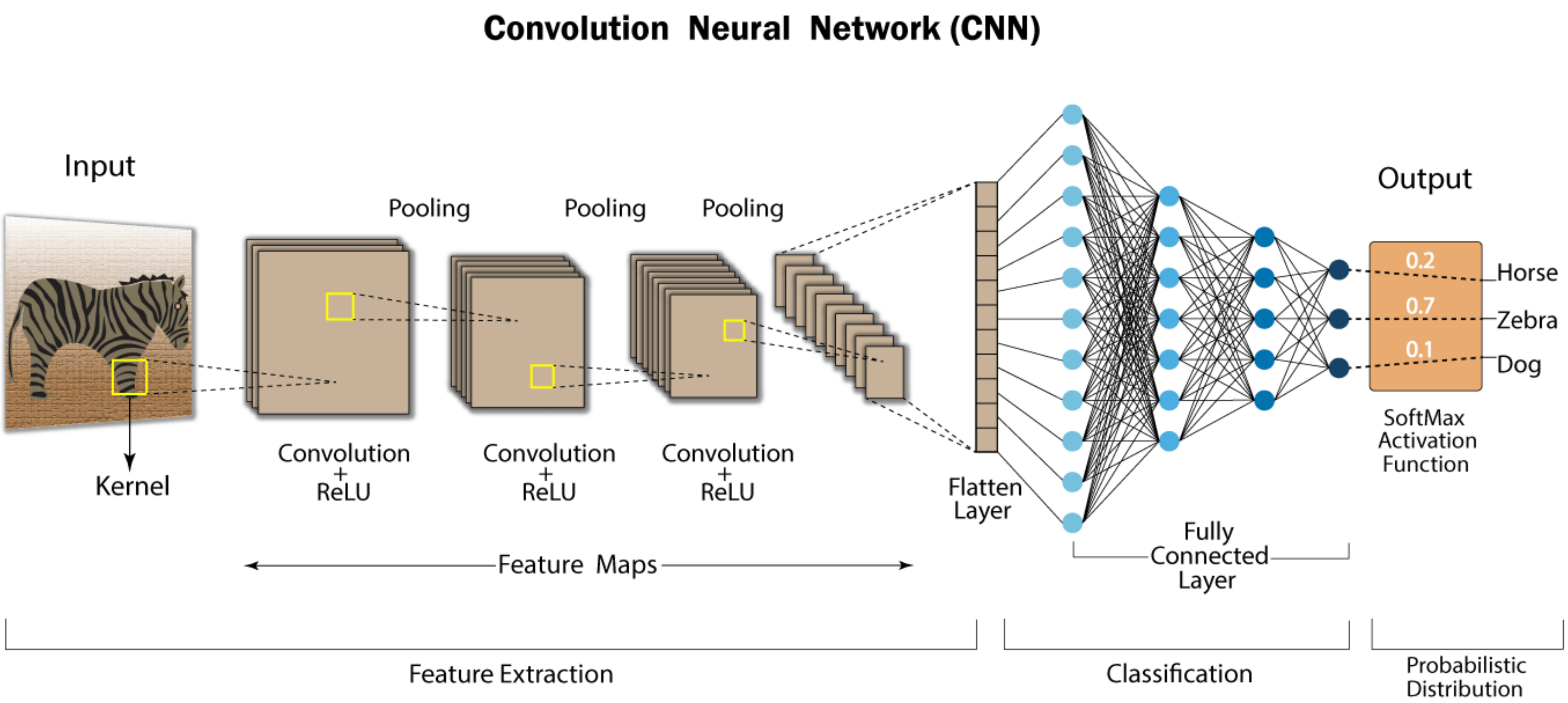
EfficientNet
&
Good guy, bad guy classification

Mì Ai

CNN



Convolution Neural Network (CNN)
Input
Pooling
Pooling
Pooling
Output
0.2
0.7
0.1
Horse
Zebra
Dog
SoftMax Activation Function
Kernel
Convolution + ReLU
Convolution + ReLU
Convolution + ReLU
Flatten Layer
Fully Connected Layer
Feature Maps
Feature Extraction
Classification
Probabilistic Distribution

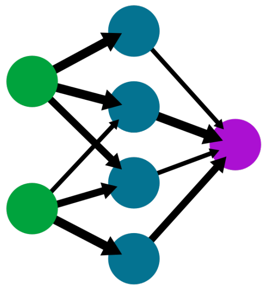
CNN

Architecture	Year	Accuracy	Parameters
AlexNet	2012	56.55%	62M
GoogleNet	2014	74.8%	6.8M
SENet	2017	82.7%	145M
GPipe	2018	84.3%	557M

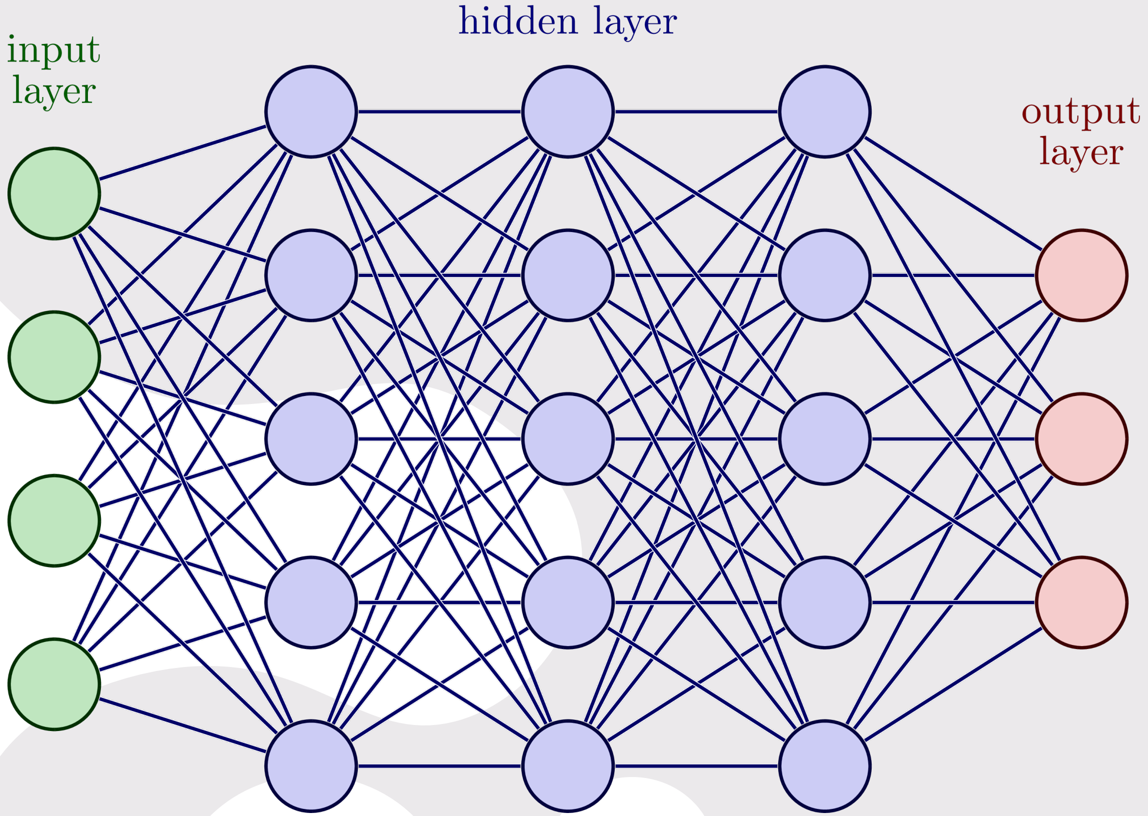
Model Accuracy vs Model Size Trade-off

https://learnopencv.com/efficientnet-theory-code/

Model Scaling



How?



Model Scaling

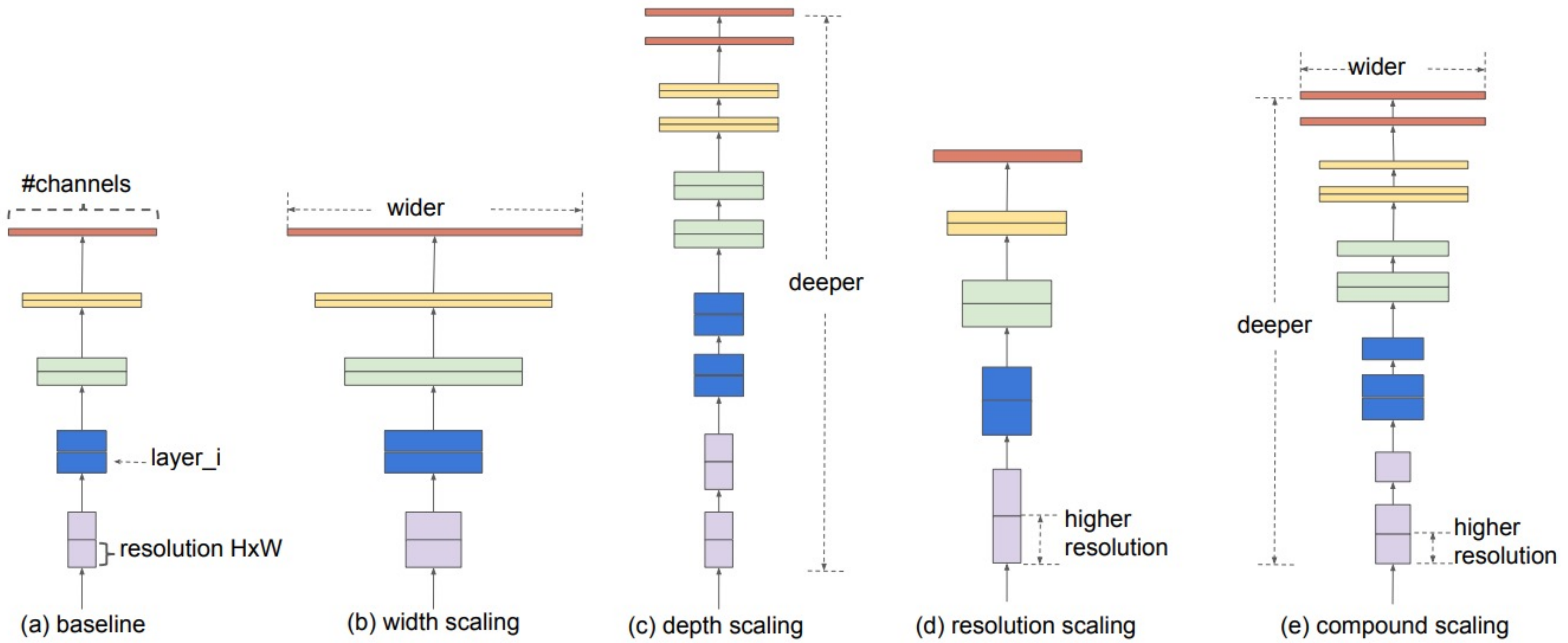
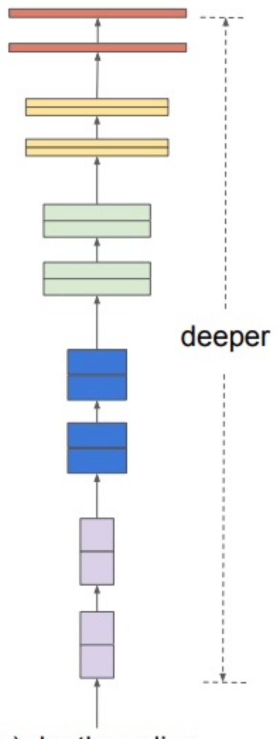


Figure 2. **Model Scaling.** (a) is a baseline network example; (b)-(d) are conventional scaling that only increases one dimension of network width, depth, or resolution. (e) is our proposed compound scaling method that uniformly scales all three dimensions with a fixed ratio.

https://learnopencv.com/efficientnet-theory-code/

Model Scaling



(c) depth scaling

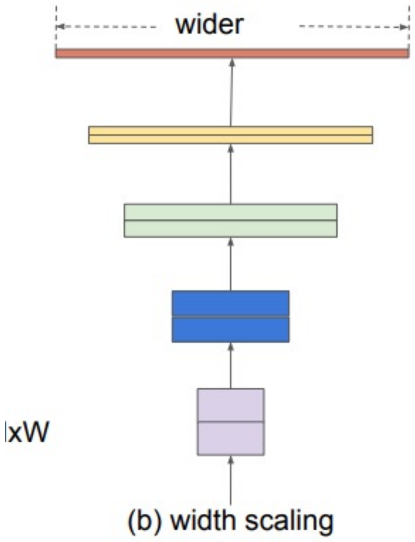
Pros
Largers number of layers hold richer details about the image -> more accurate.

E.g. ResNet-18 and ResNet-200 are both based on the ResNet architecture,
but ResNet-200 is much deeper than ResNet-18 and is, therefore, more accurate. On the other hand, ResNet-18 is smaller and faster to run.

Cons
They are difficult to train due to the vanishing gradient problem.
The gain in accuracy saturates after a certain depth.

https://learnopencv.com/efficientnet-theory-code/

Model Scaling

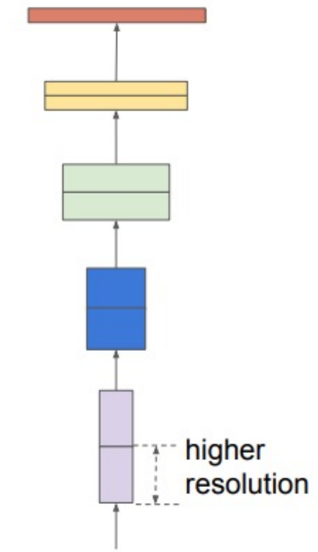


Pros
CNN layer also have multiple filters.
Network with more filters per layer is considered wider.
Easier to train.

Cons
Extremely wide and shallow network have difficulty capturing higher-level features.
The gain in accuracy saturates after a certain width..

https://learnopencv.com/efficientnet-theory-code/

Model Scaling



(d) resolution scaling

Pros

Larger image has more information than a smaller one→ change architect by taking in a larger input image and improve accuracy.

Cons

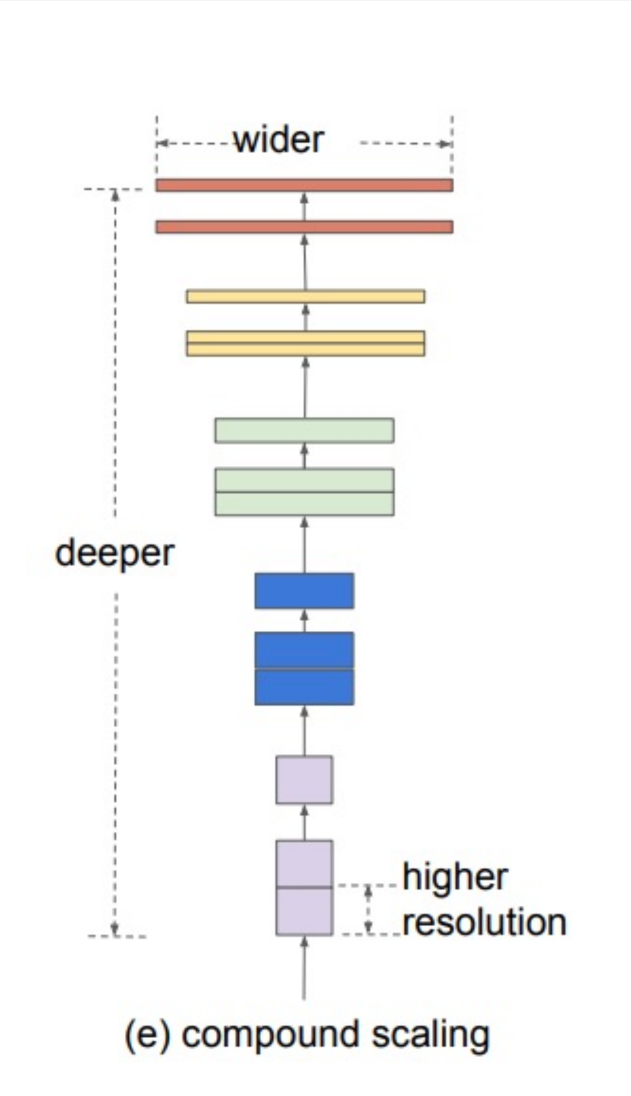
Requires more processing power

The gain in accuracy tends to saturate after a certain resolution.

https://learnopencv.com/efficientnet-theory-code/

Model Scaling

Balance all of above factor to gain better accuracy -> **Compound Scaling**



https://learnopencv.com/efficientnet-theory-code/

EfficientNet

Link:

https://arxiv.org/abs/1905.11946

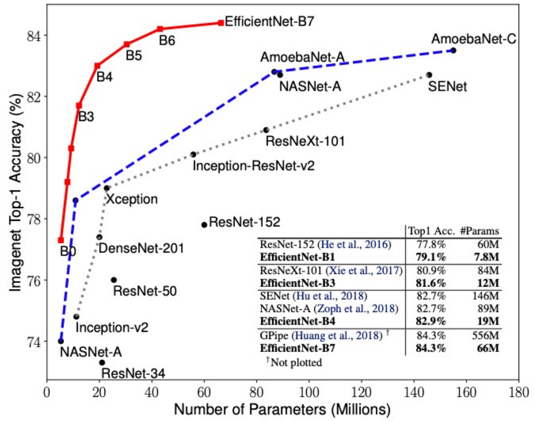
EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks

Mingxing Tan [1] Quoc V. Le [1]

Abstract

Convolutional Neural Networks (ConvNets) are commonly developed at a fixed resource budget, and then scaled up for better accuracy if more resources are available. In this paper, we systematically study model scaling and identify that carefully balancing network depth, width, and resolution can lead to better performance. Based on this observation, we propose a new scaling method that uniformly scales all dimensions of depth/width/resolution using a simple yet highly effective *compound coefficient*. We demonstrate the effectiveness of this method on scaling up MobileNets and ResNet.

To go even further, we use neural architecture search to design a new baseline network and scale it up to obtain a family of models, called *EfficientNets*, which achieve much better accuracy and efficiency than previous ConvNets. In particular, our EfficientNet-B7 achieves state-of-the-art 84.3% top-1 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

Figure 1. **Model Size vs. ImageNet Accuracy.** All numbers are for single-crop, single-model. Our EfficientNets significantly outperform other ConvNets. In particular, EfficientNet-B7 achieves new state-of-the-art 84.3% top-1 accuracy but being 8.4x smaller and 6.1x faster than GPipe. EfficientNet-B1 is 7.6x smaller and 5.7x faster than ResNet-152. Details are in Table 2 and 4.

EfficientNet

$$Y_i = F_i(X_i)$$

$$N = F_k \odot F_{k-1} \odot \ldots\ldots \odot F_1(X_1) = \bigodot_{j=1\ldots k} F_j(X_1)$$

Model has stages in that layers are repeated

$$N = \bigodot_{j=1\ldots s} F_i^{L_i} X_{\langle H_i, W_i, C_i \rangle}$$

$$\begin{aligned} \max_{d,w,r} \quad & Accuracy\big(\mathcal{N}(d,w,r)\big) \\ s.t. \quad & \mathcal{N}(d,w,r) = \bigodot_{i=1\ldots s} \hat{\mathcal{F}}_i^{d \cdot \hat{L}_i}\big(X_{\langle r\cdot\hat{H}_i, r\cdot\hat{W}_i, w\cdot\hat{C}_i\rangle}\big) \\ & \text{Memory}(\mathcal{N}) \leq \text{target\_memory} \\ & \text{FLOPS}(\mathcal{N}) \leq \text{target\_flops} \end{aligned} \tag{2}$$

https://learnopencv.com/efficientnet-theory-code/

EfficientNet

Compound scaling using ϕ to control the resource to scale the model.

α, β, γ : Constants find by grid search

FLOPS is usually proportional with d, w^2, r^2.

$$
\begin{aligned}
\text{depth: } & d = \alpha^{\phi} \\
\text{width: } & w = \beta^{\phi} \\
\text{resolution: } & r = \gamma^{\phi} \\
& \boxed{\text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2} \approx 2 \\
& \alpha \geq 1, \beta \geq 1, \gamma \geq 1
\end{aligned}
$$

$$\rightarrow (\alpha \cdot \beta^2 \cdot \gamma^2)^{\varphi} \rightarrow 2^{\varphi}$$

https://learnopencv.com/efficientnet-theory-code/

EfficientNet-B0

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

https://learnopencv.com/efficientnet-theory-code/

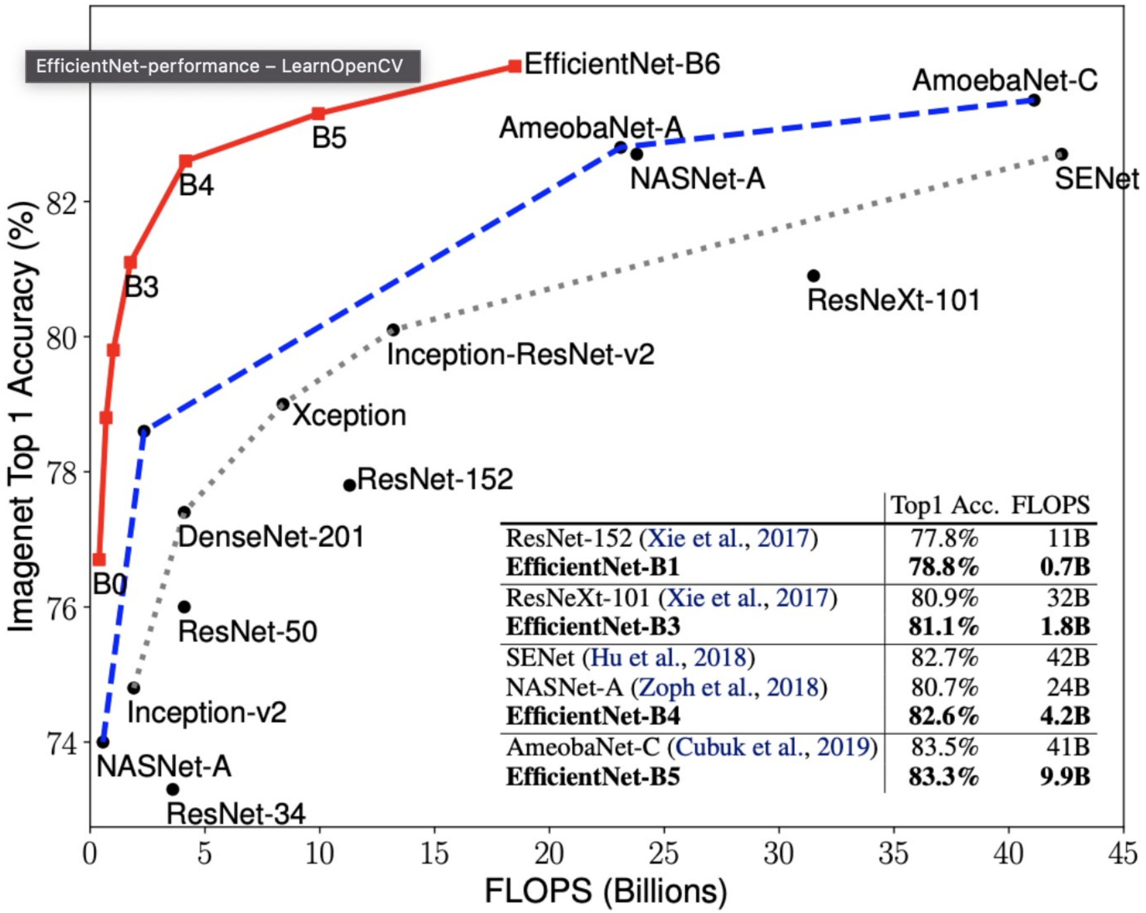
EfficientNet-BX (scaling)

Starting with EfficientNet-B0, the authors used the following strategy to scale it up.

1. Fix $\phi = 1$.
2. Assume the resource available at any step of scaling is twice of the resource at the previous step.
3. Do a small grid search over α, β, and γ such that the constraint in the above equation is not violated.
4. The authors found the parameters α = 1.2, β = 1.1, γ = 1.15 to work the best.
5. Fix α, β, and γ as constants and scale up EfficientNet-B0 with different ϕ to obtain new scaled networks EfficientNet-B1 to B7.

https://learnopencv.com/efficientnet-theory-code/

EfficientNet-BX (scaling)



	Top1 Acc.	FLOPS
ResNet-152 (Xie et al., 2017)	77.8%	11B
EfficientNet-B1	**78.8%**	**0.7B**
ResNeXt-101 (Xie et al., 2017)	80.9%	32B
EfficientNet-B3	**81.1%**	**1.8B**
SENet (Hu et al., 2018)	82.7%	42B
NASNet-A (Zoph et al., 2018)	80.7%	24B
EfficientNet-B4	**82.6%**	**4.2B**
AmeobaNet-C (Cubuk et al., 2019)	83.5%	41B
EfficientNet-B5	**83.3%**	**9.9B**

https://learnopencv.com/efficientnet-theory-code/

Hands-on

- EfficientNet B0 on Bad guy/good guy classification
- ImageDataGenerator from Dataframe