EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks

National Research University “Higher School of Economics”
Faculty of Computer Science
Maxim K.

Moscow, 2020.

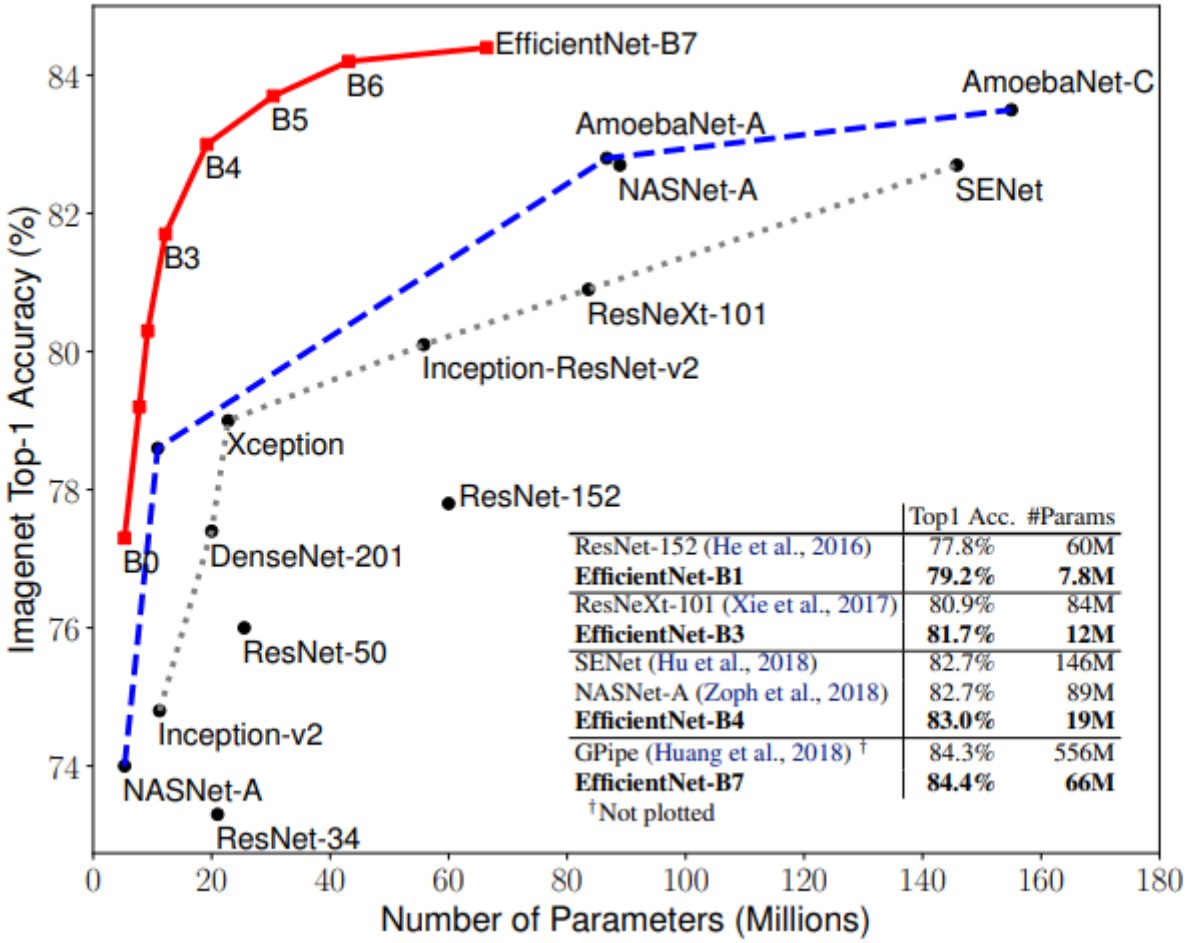
Plan

- Introduction. Main idea
- Caused improvements
- Simple scaling and methods' actuality
- Compound Model Scaling
- EfficientNet Architecture
- Baseline and learning to scale
- Comparisons and results
- Conclusion

Model Scaling. Main idea.

- Scaling FullyConnected Networks impossible
- ConvNets developed in fixed resource budget.
- Scale up for better accuracy!
- Ok.
  ResNet-18
- ResNet-32
- ResNet-50-152-1000
- May be we doing something wrong???

Caused improvements



	Top1 Acc.	#Params
ResNet-152 (He et al., 2016)	77.8%	60M
EfficientNet-B1	**79.2%**	**7.8M**
ResNeXt-101 (Xie et al., 2017)	80.9%	84M
EfficientNet-B3	**81.7%**	**12M**
SENet (Hu et al., 2018)	82.7%	146M
NASNet-A (Zoph et al., 2018)	82.7%	89M
EfficientNet-B4	**83.0%**	**19M**
GPipe (Huang et al., 2018) †	84.3%	556M
EfficientNet-B7	**84.4%**	**66M**

†Not plotted

Picture 1. Model Size vs ImageNet Accuracy.

Simple scaling and methods actuality

- Three ways to scale Convolutional Network:
  1. Depth (He et al., 2016)
  2. Width (Zagoruyko & Komodakis, 2016)
  3. Resolution (Huang et al., 2018)
- Clear to scale network in one way.
  But most commonly used 2-3 ways scaling – very random & often reach sub-optimal accuracy and efficient.
- Theoretically (Raghu et al., 2017; Lu et al., 2018) and practically (Zagoruyko & Komodakis, 2016) shows relationship between network width and depth.
- Let’s find parameters’ global dependence

Convolutional model scaling as is

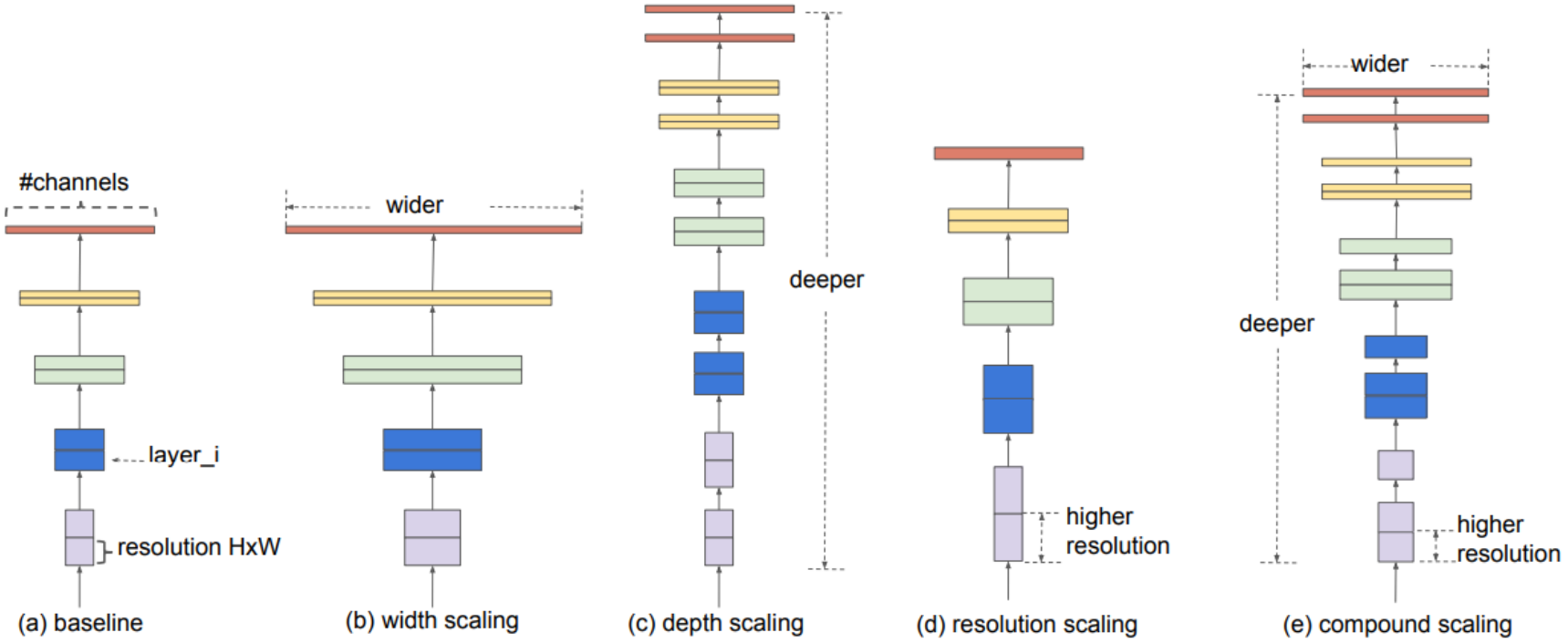


Figure 2. Model Scaling. (a) is a baseline network example; (b)-(d) are conventional scaling that only increases one dimension of network width, depth, or resolution. (e) is our proposed compound scaling method that uniformly scales all three dimensions with a fixed ratio.

Compound Model Scaling

$i-th\ ConvNet\ layer$:
$Y_i = \mathcal{F}_i(X_i), where\ X\ - input, Y - output.$

$ConvNet\ \mathcal{N}\ is\ list\ of\ composed\ layers$
$\mathcal{N} = \mathcal{F}_k \circ \ldots \circ \mathcal{F}_2 \circ \mathcal{F}_1(X_1)$

$Optimisation\ problem$:

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

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

$$\text{Memory}(\mathcal{N}) \leq \text{target\_memory}$$

$$\text{FLOPS}(\mathcal{N}) \leq \text{target\_flops}$$

Compound Model Scaling

Optimization difficulty: d, w, r depend on each other.

Due to this – most commonly scale in one of these.

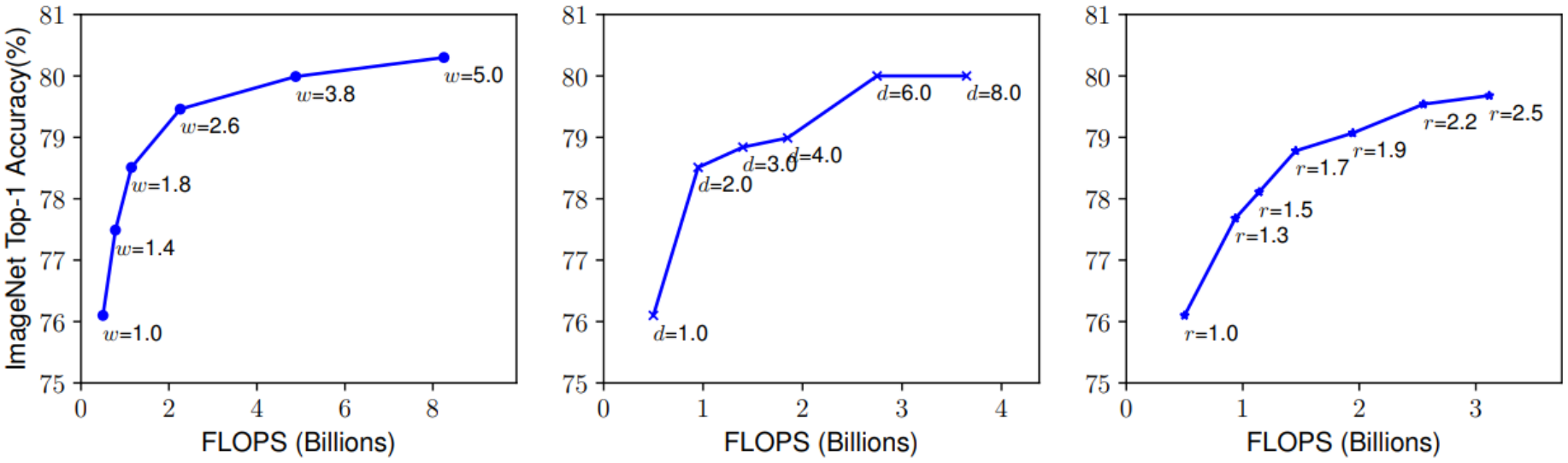


Figure 3. Scaling Up a Baseline Model with Different Network Width, Depth, Resolution Coefficients.

Compound Model Scaling

- Depth (d).
  Intuition: deeper ConvNet can capture richer and more complex features and generalize well on new tasks (Transfer L).
- Width (w).
  Commonly use for a small-size models. However extremely wide but shallow networks feel problems with capturing higher level features
- Resolution (r).
  With higher resolution input images, ConvNet can capture more fine-grainted patterns. GPipe (sota ImageNet accuracy) on 480x480.

Scaling up any dimension of network width, depth, or resolution improves accuracy, but the accuracy gain diminishes for bigger models.

Compound Model Scaling

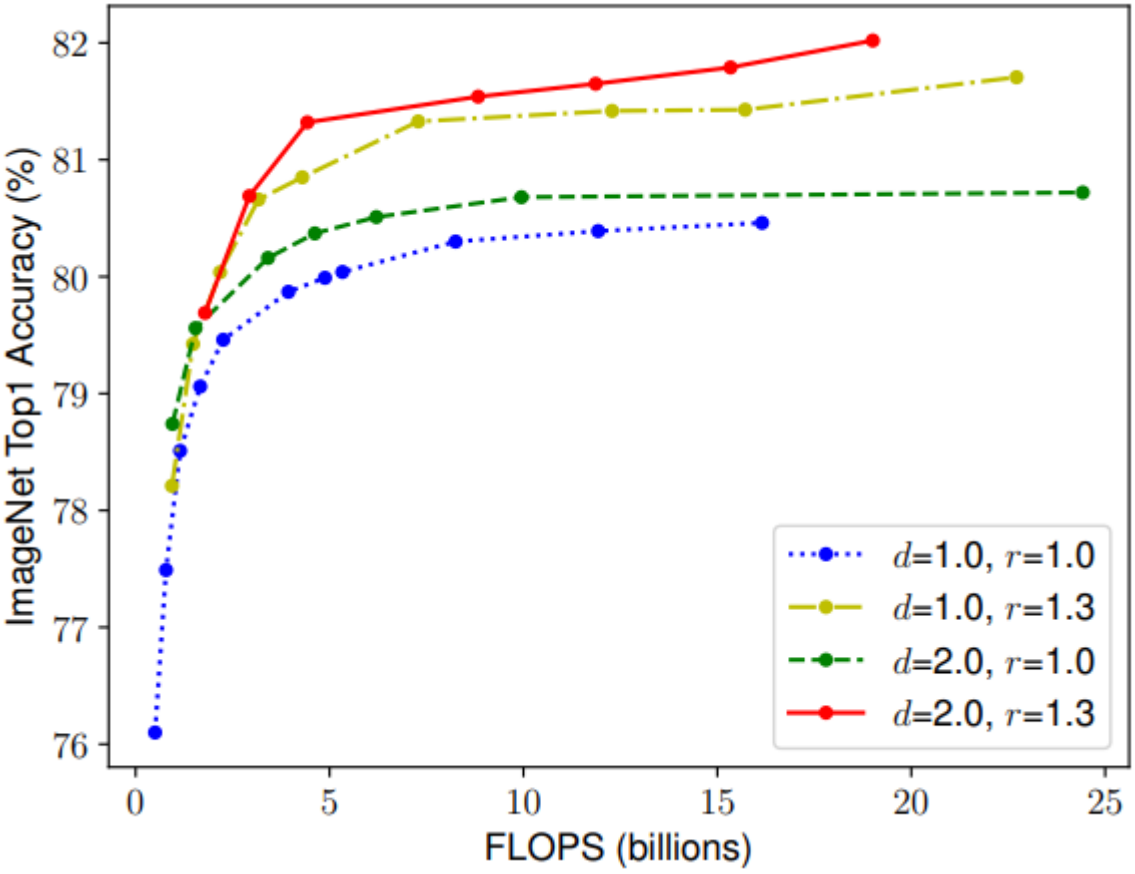


Figure 3. Scaling Network Width for Different Baseline Networks. Each dot in a line denotes a model with different width coefficient. The first baseline has 18 conv layers with 224x224, the last baseline has 36 conv layers with 299x299.

Compound Model Scaling

$$\text{Conv FLOPS} \approx d,\ w^2,\ r^2$$

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

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

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

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

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

$$\text{total FLOPS} \approx \left(\alpha \cdot \beta^2 \cdot \gamma^2\right)^\phi \approx 2^\phi$$

Baseline Architecture

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

Table 1. EfficientNet-B0 baseline network.

Learning to scale

- STEP 1:

  $\varphi = \varphi\ if\ \varphi$ is not None else 1; (here we fix φ)

  $\alpha, \beta, \gamma = GridSearch($ *ConvNet* $\mathcal{N}$ *is list of composed layers*

  $\mathcal{N} = \mathcal{F}_k \circ ... \circ \mathcal{F}_2 \circ \mathcal{F}_1(X_1)$

  Optimisation problem:

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

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

  $\text{Memory}(\mathcal{N}) \leq$ target_memory

  $\text{FLOPS}(\mathcal{N}) \leq$ target_flops

  $)$

  depth: $d = \alpha^\phi$

  width: $w = \beta^\phi$

  resolution: $r = \gamma^\phi$

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

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

- STEP 2:

  $for\ fixed\ \alpha, \beta, \gamma\ scaling\ up\ network\ with\ different\ \varphi$ $and\ get\ EfficientNet - B\{1-7\}.$

$$e.g.\ for\ EfficientNet - B0:$$

$$\alpha = 1.2, \quad \beta = 1.1, \quad \gamma = 1.15$$

Results

Model	Top-1 Acc.	Top-5 Acc.	#Params	Ratio-to-EfficientNet	#FLOPS	Ratio-to-EfficientNet
EfficientNet-B0	**77.3%**	**93.5%**	**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	**79.2%**	**94.5%**	**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	**80.3%**	**95.0%**	**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	**81.7%**	**95.6%**	**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	**83.0%**	**96.3%**	**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	**83.7%**	**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	**84.2%**	**96.8%**	**43M**	**1x**	**19B**	**1x**
EfficientNet-B7	**84.4%**	**97.1%**	**66M**	**1x**	**37B**	**1x**
GPipe (Huang et al., 2018)	84.3%	97.0%	557M	8.4x	-	-

Table 2. EfficientNet Performance Results on ImageNet. ConvNets with similar top-1/top-5 accuracy are grouped together.

Compound Scaling on other networks.

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%**

Table 3. Scaling Up MobileNet and ResNet.

Results

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%**

Table 4. Scaled Models Used in Figure 7.

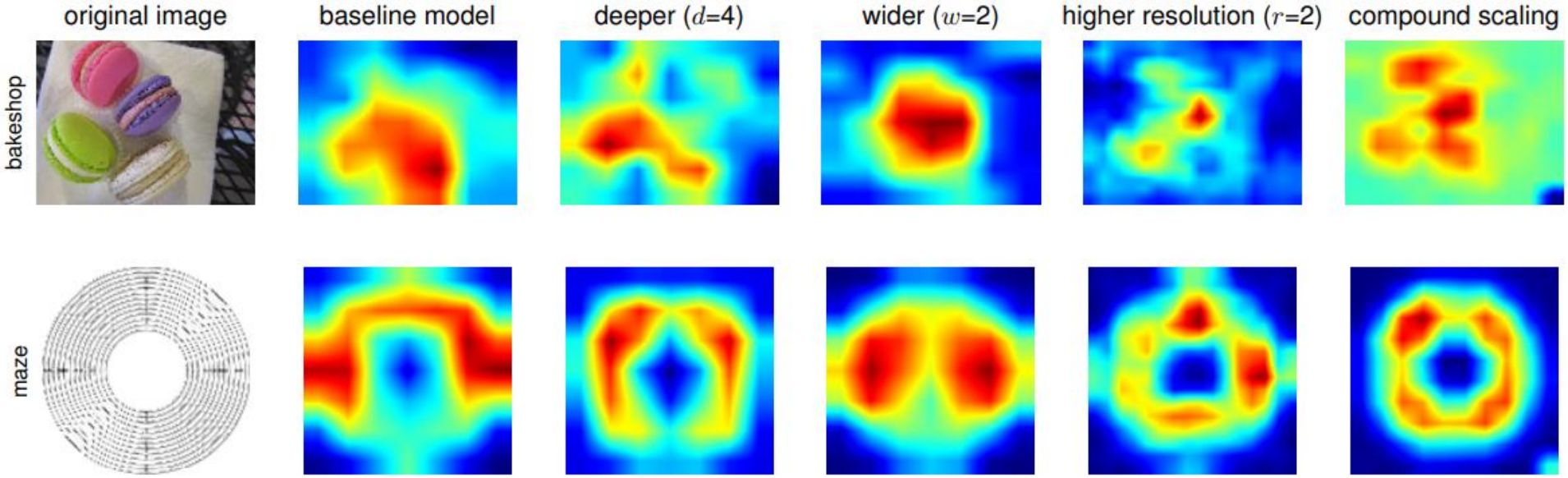


Figure 7. Class Activation Map (CAM) for Models with different scaling methods.

Results

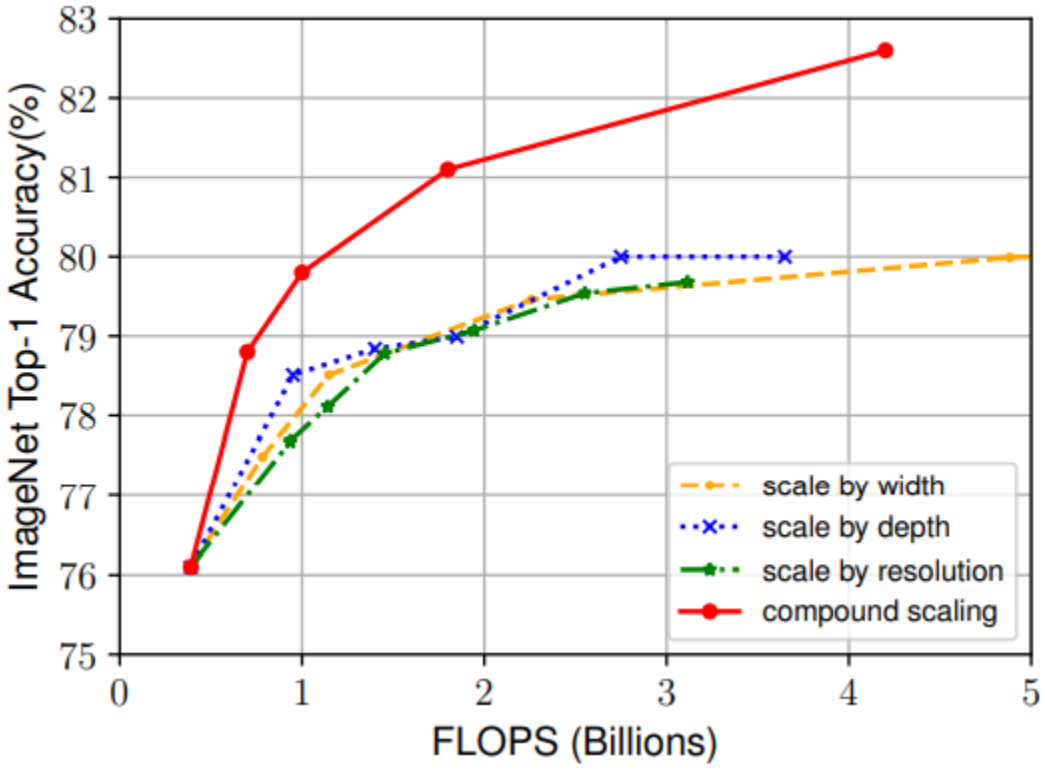


Figure 8. Scaling Up EfficientNet-B0 with Different Methods.

Conclusion

- Carefully balanced network width, depth, and resolution gives better accuracy and efficiency.
- Compound Scaling enables to scale up a baseline ConvNet to any target resource.
- Compound Scaling allows to focus on more relevant regions with more object details.
- Founded mobile-size EfficientNet model (n-times lower in parameters and FLOPS) can be scaled up very effectively, surpassing state-of-the-art accuracy.