EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks

GROUP 22:

HE Yichen	A0236326H
LOU Xinyun	A0251245M
SONG Mingyue	A0251125U
WANG Jiangyi	A02346307J
YANG Sizhe	A0236299N

CONTENTS

- BACKGROUND
- MOTIVATION
- REPRODUCTION
- EXTENTION
- CONCLUSION

BACKGROUND

EVOLUTION OF CNNs

01 **AlexNet (2012)**

First winner of the ImageNet challenge based on a CNN

02 **ZFNet (2013)**

Change in hyper parameter

03 **VGGNet (2014)**

Depth is critical in CNN design

04 **GoogLeNet (2014)**

Reduce parameter count, memory usage, and computation

05 **ResNet (2015)**

Residual blocks skip connection

EfficientNet

Refer to: https://medium.com/appyhigh-technology-blog/convolutional-neural-networks-a-brief-history-of-their-evolution-ee3405568597

EVOLUTION OF CNNs

Observation 1:

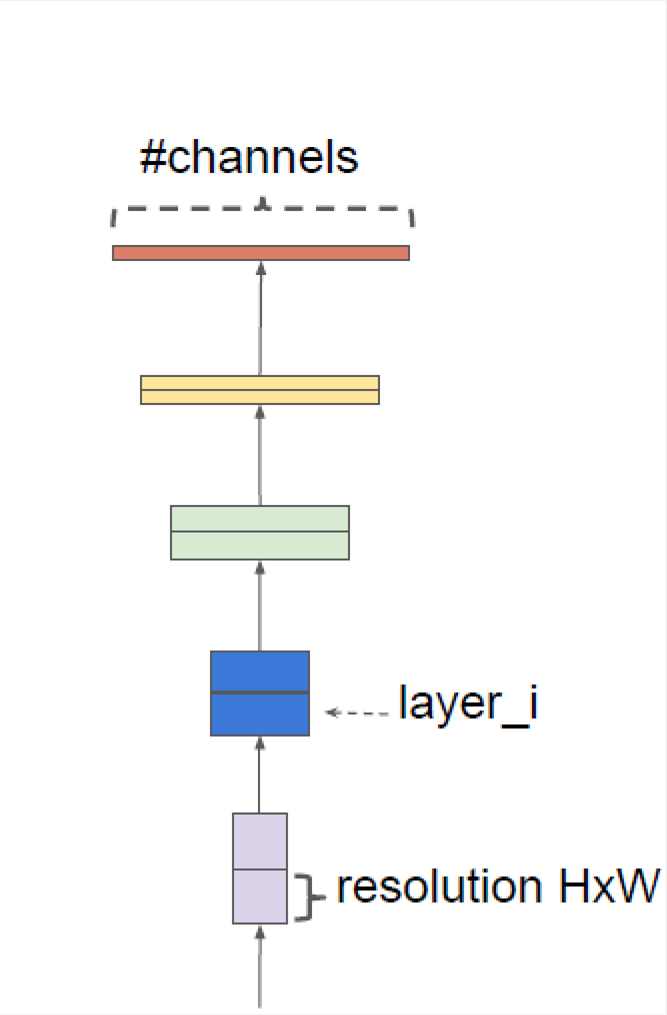
More sophisticated architecture

Better performance

Larger scale

2 MOTIVATION

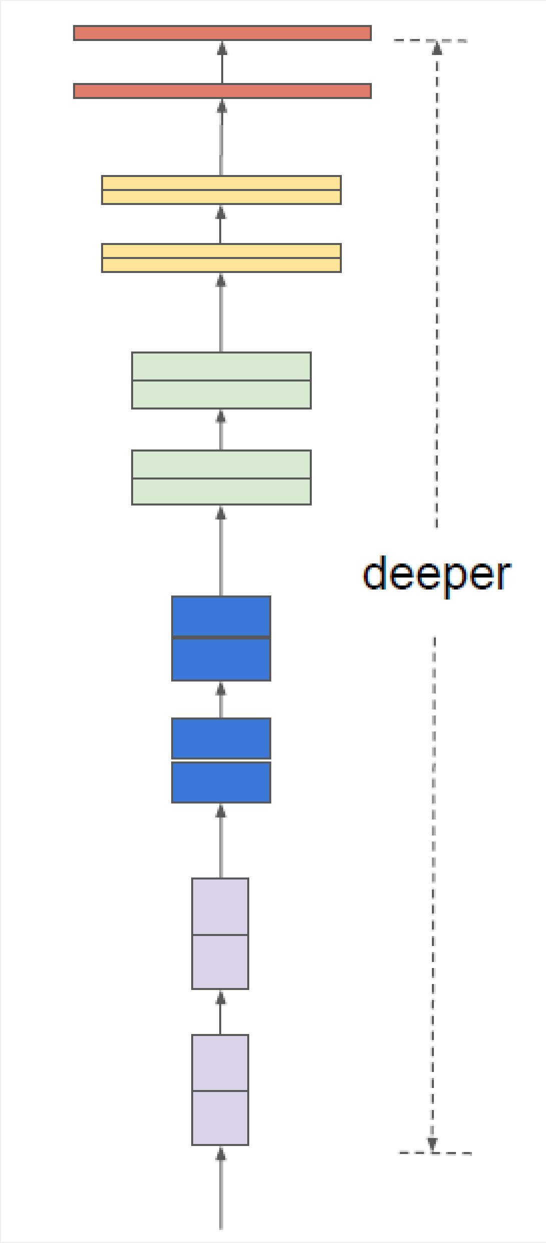
SCALING UP CNNs



Baseline

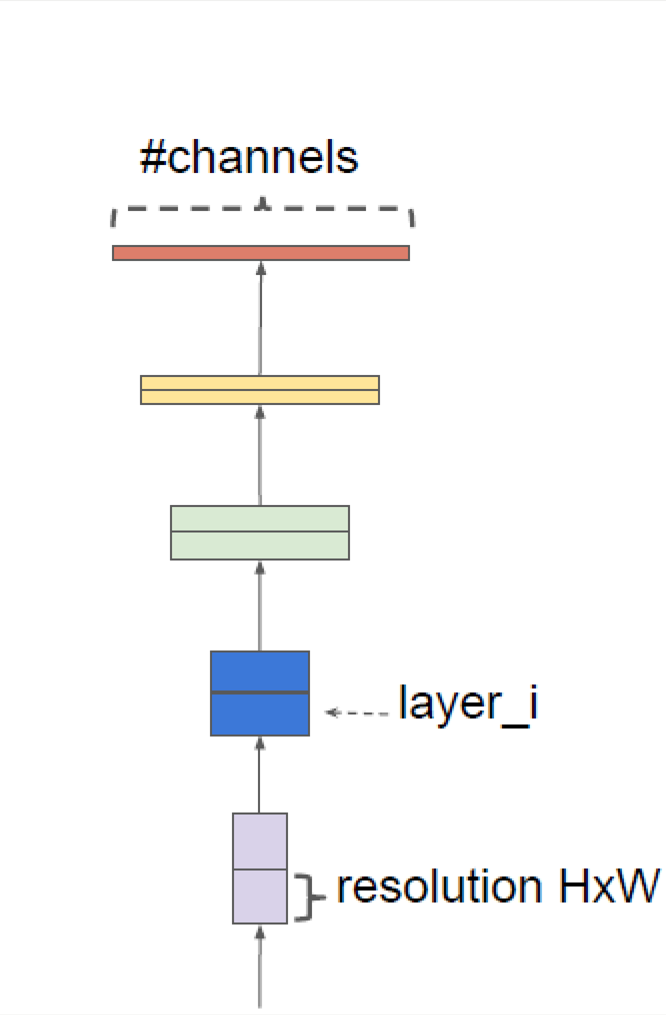
Add more layers

Depth coefficient: d
(In this case, d=2)



Depth scaling

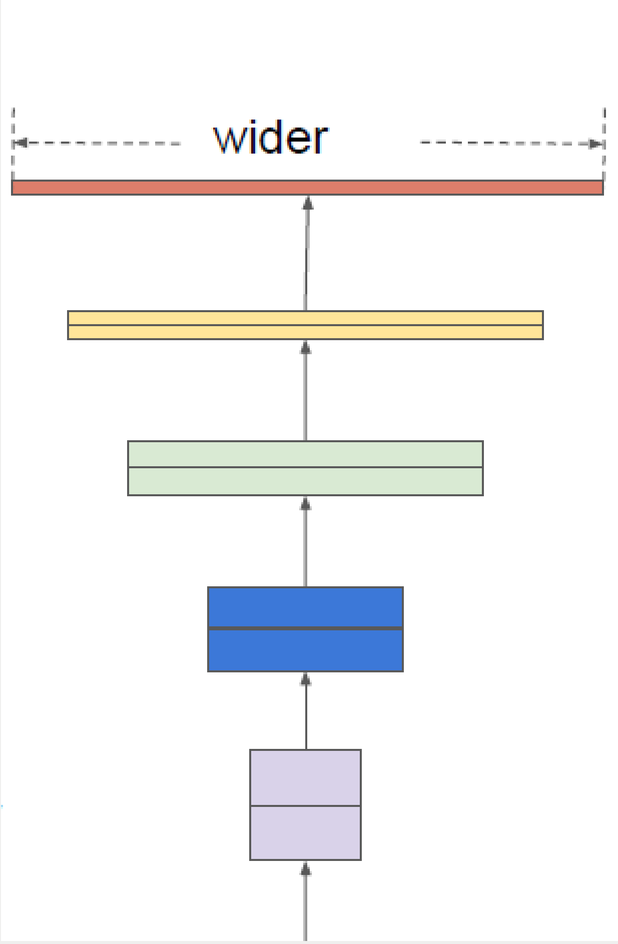
SCALING UP CNNs



Baseline

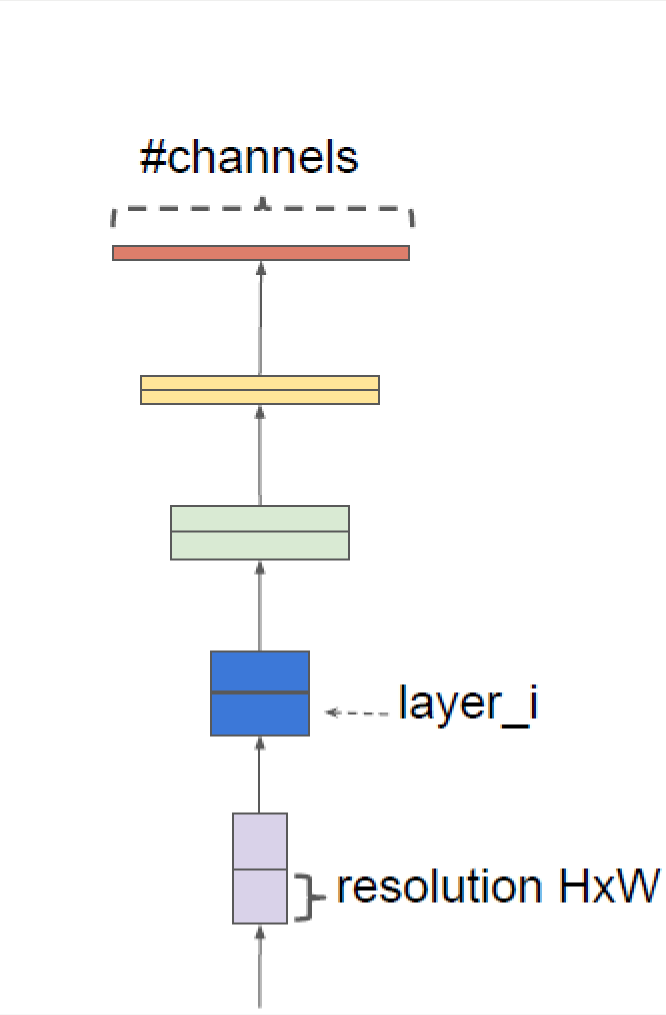
Add more channels

Width coefficient: w



Width scaling

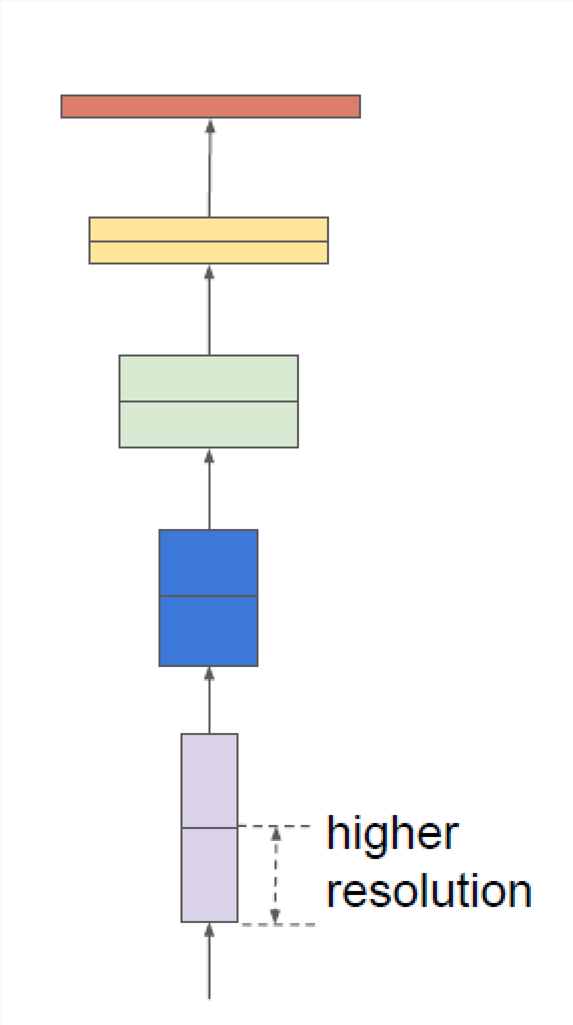
SCALING UP CNNs



Baseline

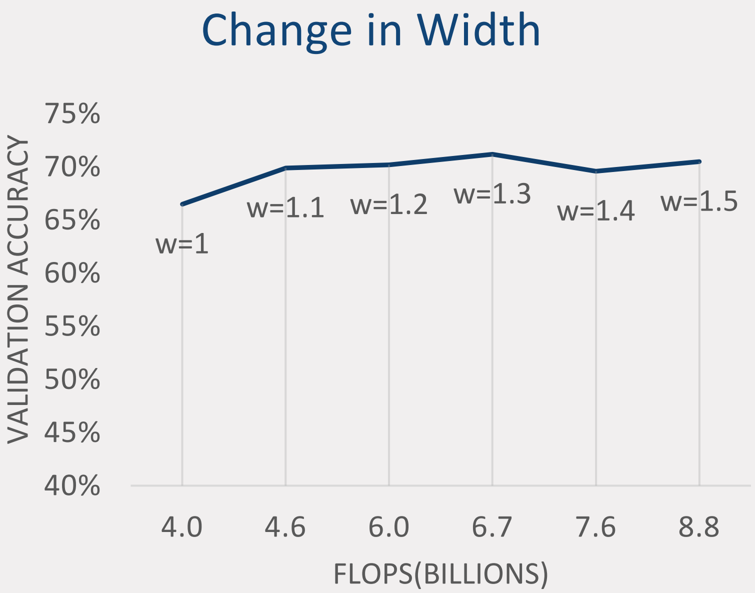
Higher resolution of input image

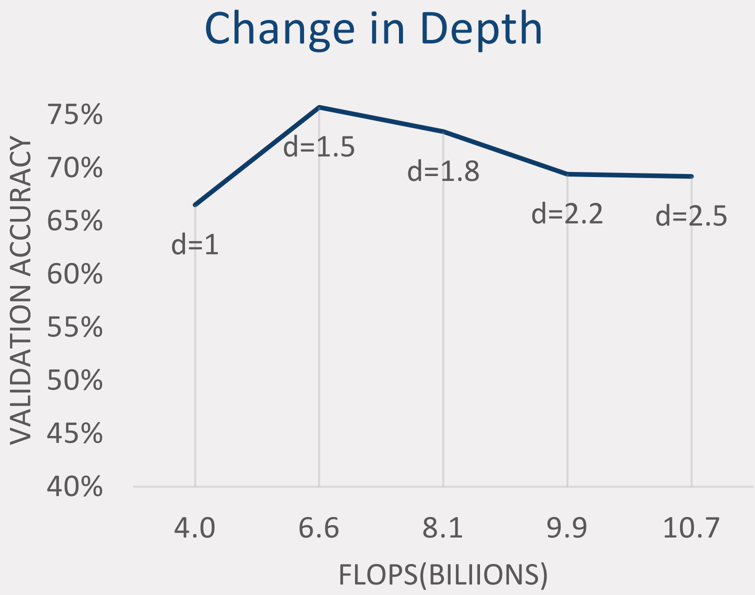
Resolution coefficient: r

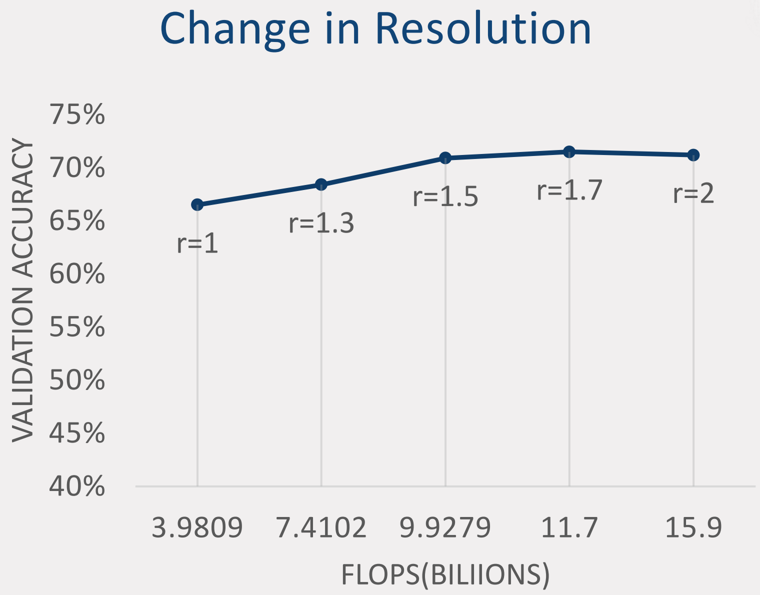


Resolution scaling

EMPIRICAL STUDY







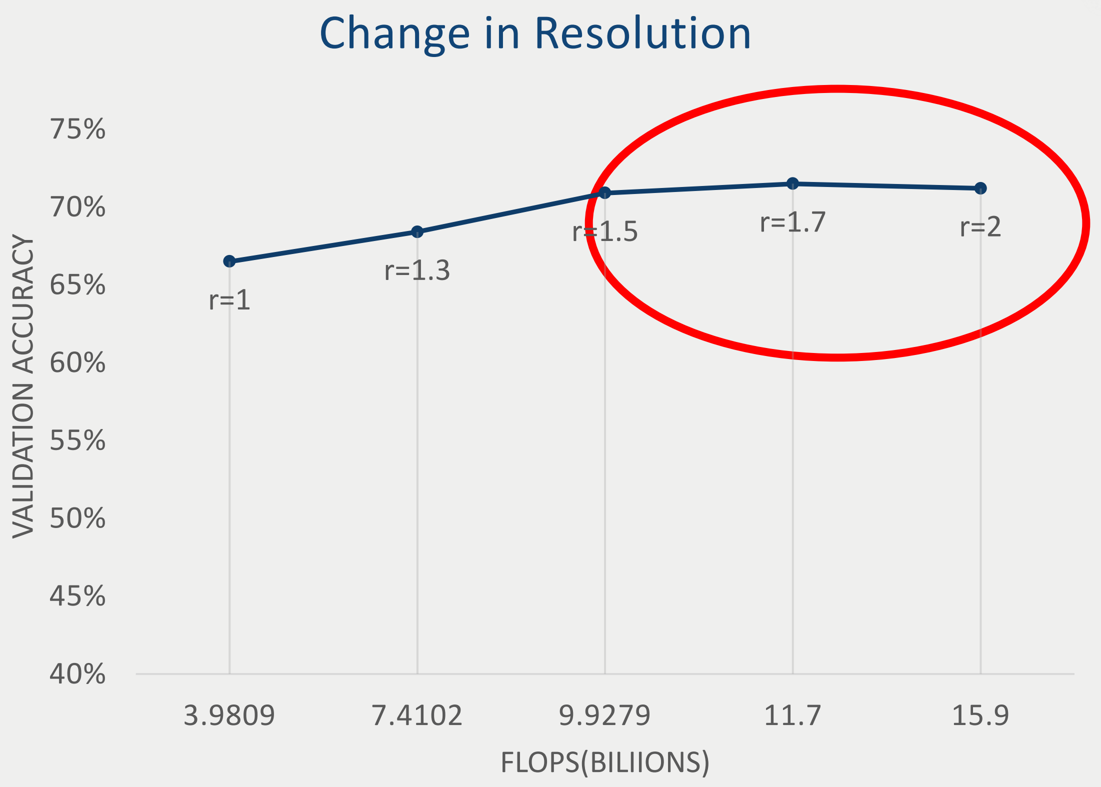
Experiment setting:

- Baseline: EfficientNet-B0 (Structure will be detailed discussed later)
- Dataset: CIFAR-10 with only 3 classes (cat, deer and dog),
  15k~ training samples, 3k~ testing samples
- Epochs: 40

EMPIRICAL STUDY

Observation 2:

Scaling up any dimension of network width, depth, or resolution improves accuracy, but the degree of benefit diminishes as the models grow larger.



MOTIVATION

But different scaling dimensions are **not independent**.

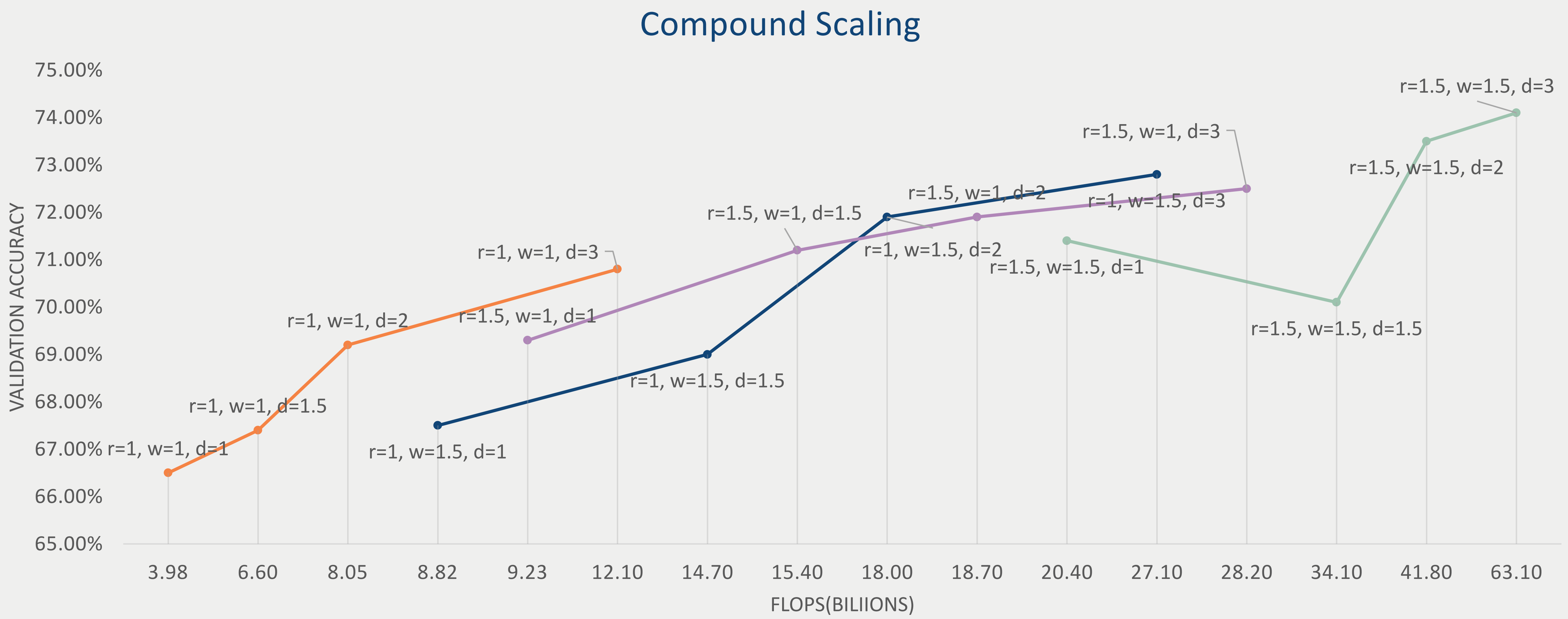
They are **interrelated**.

MOTIVATION

So…….

Uniformly scaling up the coefficients!

Compound Scaling



Compound Scaling

Observation 3:

In order to pursue better accuracy and efficiency, it is critical to balance all dimensions of network width, depth, and resolution during ConvNet scaling.

MOTIVATION

Based on *observation 2 & 3* ——

Observation 2:

Scaling up any dimension of network improves accuracy, but the degree of benefit diminishes as the models grow larger.

Observation 3:

We need to balance all dimensions of network during ConvNet scaling.

MOTIVATION

Based on *observation 2 & 3* ——

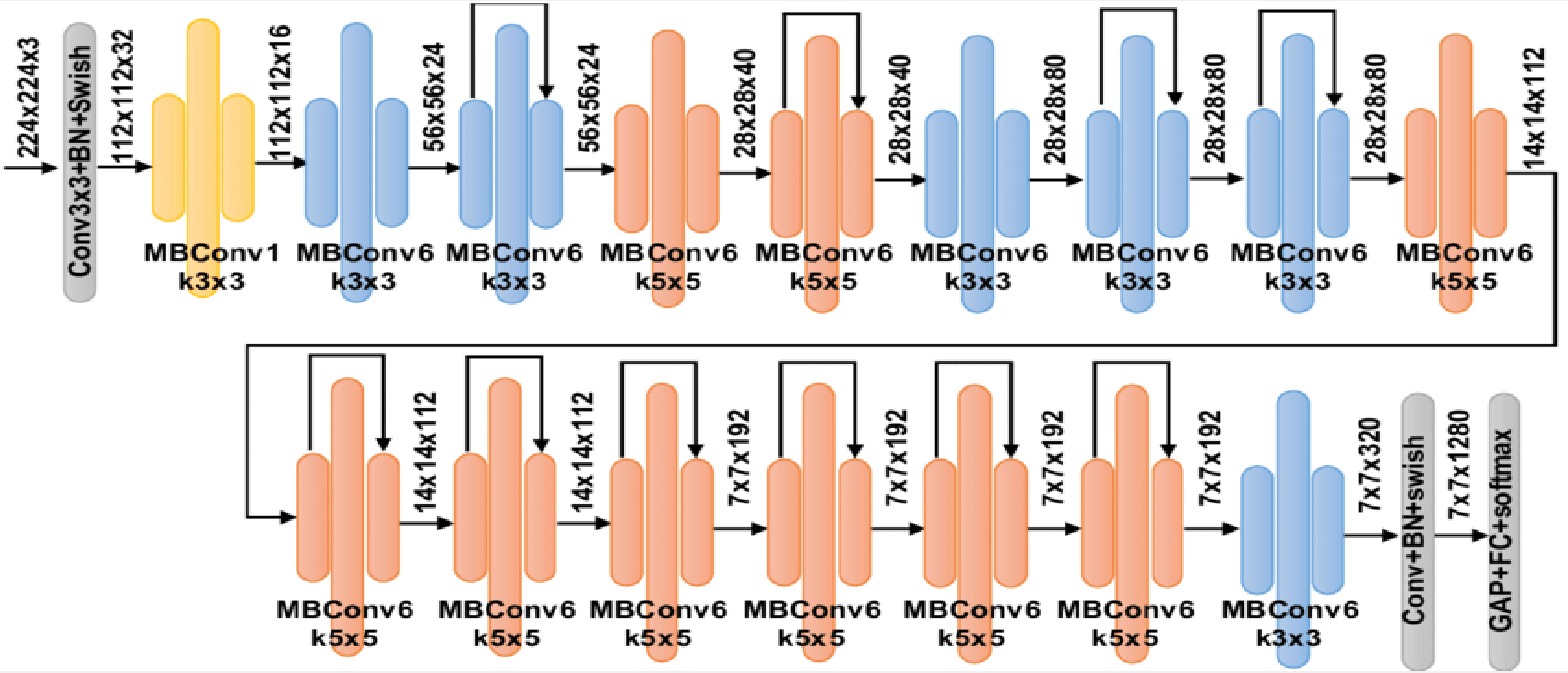
Question:

How to balance the three dimensions of neural networks?

REPRODUCTION

EFFICIEFFICIENTNET B0 ARCHITECTURE

Before talking about result, go through the general architecture of EfficientNet B0 (Baseline).



EFFICIENTNET REPRODUCTION

To conclude, there are two main parts for our reproduction:

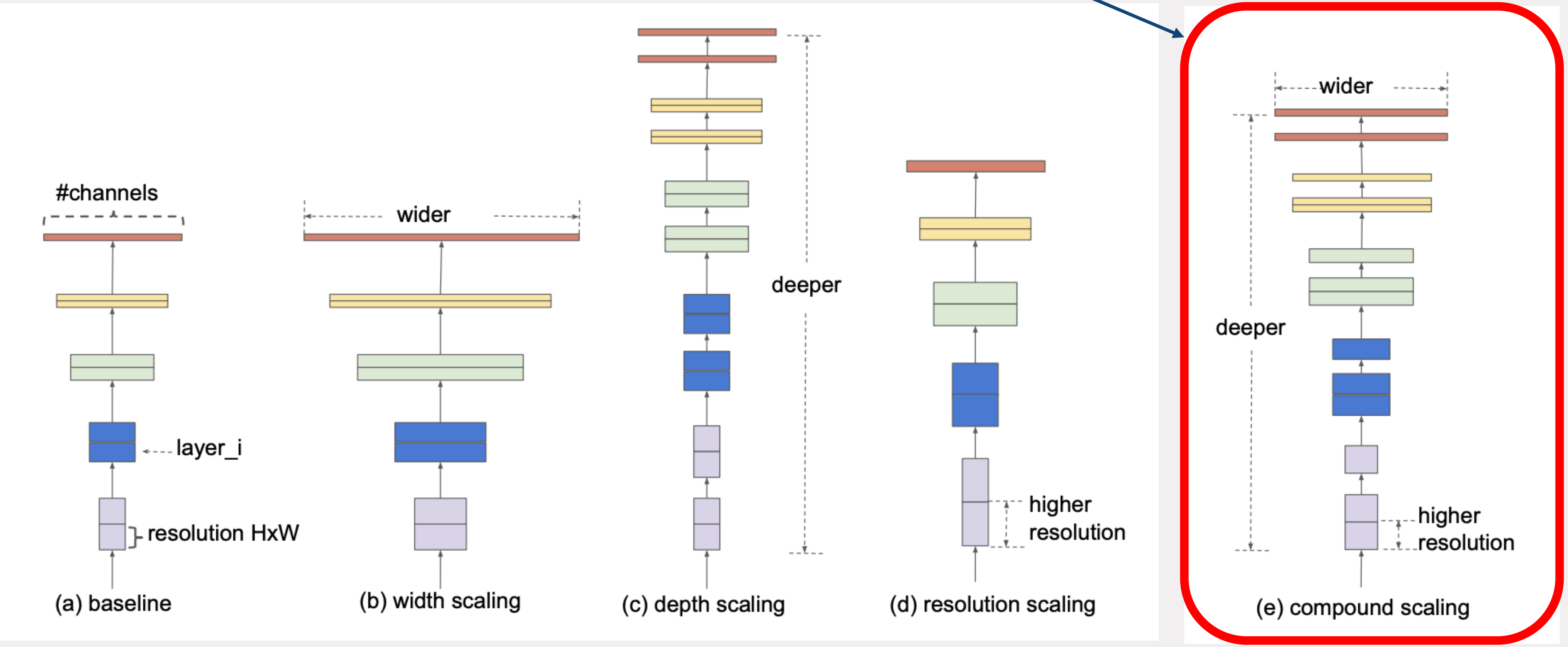
1. To determine the optimal combination of depth, width, and resolution for specific computation resources.

2. Compare the performance of EfficientNet with other convolutional neural network (CNN) architectures.

EFFICIENTNET REPRODUCTION

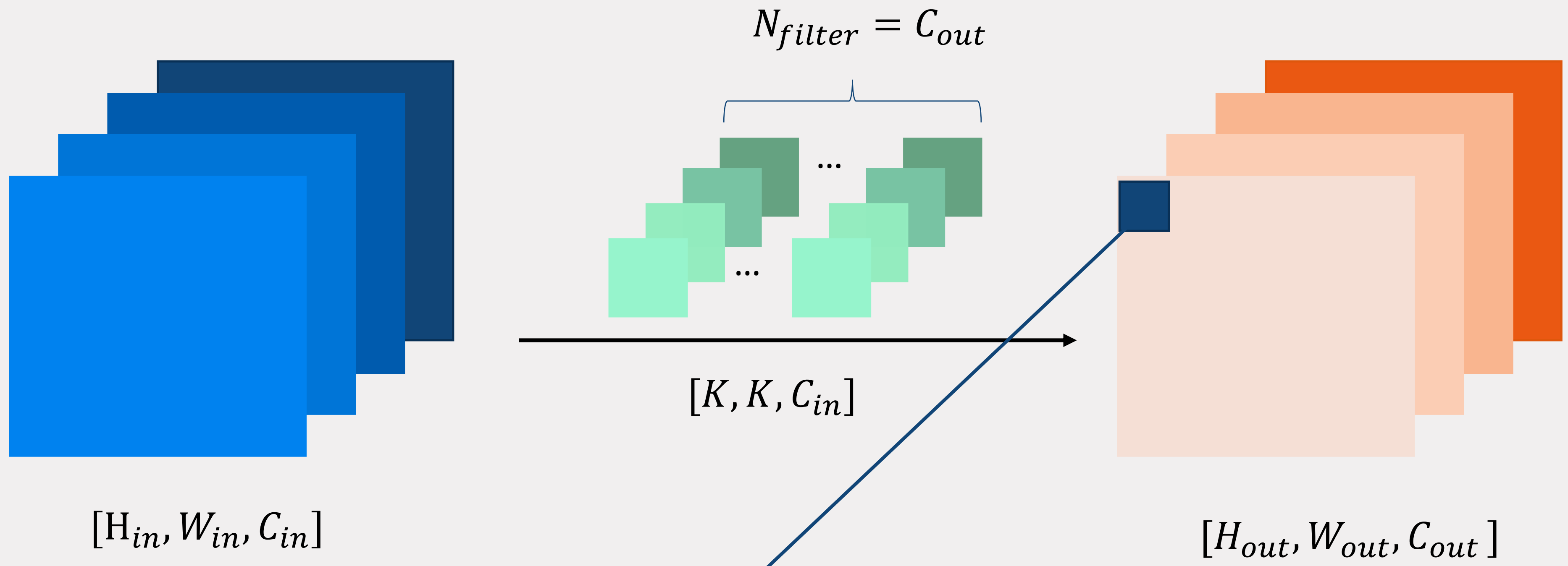
First part: Determine combination of depth, width, resolution

Efficient Net proposes ‘Compound Scaling’ Scheme to determine the optimal scaling factors.



EFFICIENTNET REPRODUCTION

First part: Determine combination of depth, width, resolution



Computational Complexity $\sim O(\overbrace{C_{in} \times K^2} \times \underbrace{H_{out} \times W_{out} \times C_{out}}_{\text{\# of pixels}})$

EFFICIENTNET REPRODUCTION

First part: Determine combination of depth, width, resolution

Result: Computational Complexity $\sim O(C_{in} \times K^2 \times H_{out} \times W_{out} \times C_{out})$

Scaling factor: β $\longleftrightarrow$ Width: w $\longleftrightarrow$ (C_{in}, C_{out})

Scaling factor: γ $\longleftrightarrow$ Resolution: r $\longleftrightarrow$ (H_{out}, W_{out})

Conclusion1: if we enlarge width by β and resolution by γ, then the computational costs will be proportional to $(\beta^2 \cdot \gamma^2)$ times.

Scaling factor: α $\longleftrightarrow$ Depth: d $\longleftrightarrow$ $(\# \; of \; convolution \; modules)$

Conclusion2: if we enlarge depth by α, then the computational costs will be proportional to α times.

Conclusion3: if we enlarge 3 factors by (α, β, γ) respectively, then the computational costs will be **proportional to $(\boldsymbol{\alpha \cdot \beta^2 \cdot \gamma^2})$ times.**

EFFICIENTNET REPRODUCTION

First part: Determine combination of depth, width, resolution

<u>'Compound Scaling' Scheme</u>:

Question: if now, our resources are N times larger, how to allocate for these 3 scaling factors?

- Step 1: Assume twice more resources are available, as we illustrate in the previous slide, we are required to solve:

$$\begin{cases} \max\limits_{\{\alpha,\beta,\gamma\}} : validation\_accuracy(\alpha,\beta,\gamma) \\ s.t: \quad \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \end{cases}$$

<u>Grid Search!</u>

EFFICIENTNET REPRODUCTION

First part: Determine combination of depth, width, resolution

Grid Search Result:

Comparison between Different Grid Search Combinations

index	α	β	γ	best_train_accuracy	best_val_accuracy
1	**1.2**	**1.1**	**1.15**	**86.02%**	**93.17%**
2	1.2	1.15	1.1	84.85%	92.40%
3	1.3	1.05	1.15	85.10%	92.12%
4	1.3	1.15	1.05	84.21%	92.03%
5	1.15	1.1	1.2	85.07%	92.26%
6	1.1	1.2	1.15	84.71%	92.52%

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

Experiment setting:

- Baseline: EfficientNet-B0
- Dataset: CIFAR-10 with 10 classes
- Epochs: 100

EFFICIENTNET REPRODUCTION

First part: Determine combination of depth, width, resolution

<u>'Compound Scaling' Scheme</u>:

Question: if now, our resources are N times larger, how to allocate for these 3 scaling factors?

- Step 1: Assume twice more resources are available, as we illustrate in the previous slide, we are required to solve:

$$\begin{cases} \max\limits_{\{\alpha,\beta,\gamma\}} : validation\_accuracy(\alpha,\beta,\gamma) \\ s.t: \quad \alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \end{cases}$$

- Step 2: Now we achieve the optimal $(\hat{\alpha}, \hat{\beta}, \hat{\gamma})$ and in the question, we have <u>N times</u> more computation resources. In this setting, the optimal scaling factors $(\alpha^*, \beta^*, \gamma^*)$ can be determined as follows:

$$\begin{cases} \alpha^*, \beta^*, \gamma^* = \hat{\alpha}^{\phi}, \hat{\beta}^{\phi}, \hat{\gamma}^{\phi} \\ \phi = \log_2 N \end{cases} \longrightarrow \begin{cases} d^* = d_0 \cdot \alpha^* \\ w^* = w_0 \cdot \beta^* \\ r^* = r_0 \cdot \gamma^* \end{cases}$$

EFFICIENTNET REPRODUCTION

First part: Determine combination of depth, width, resolution

Big, Intractable Grid Search Problem
(N times more resources)

Uniform Scaling Parameter ϕ

Small, Tractable Grid Search Problem
(2 times more resources)

Trade-off

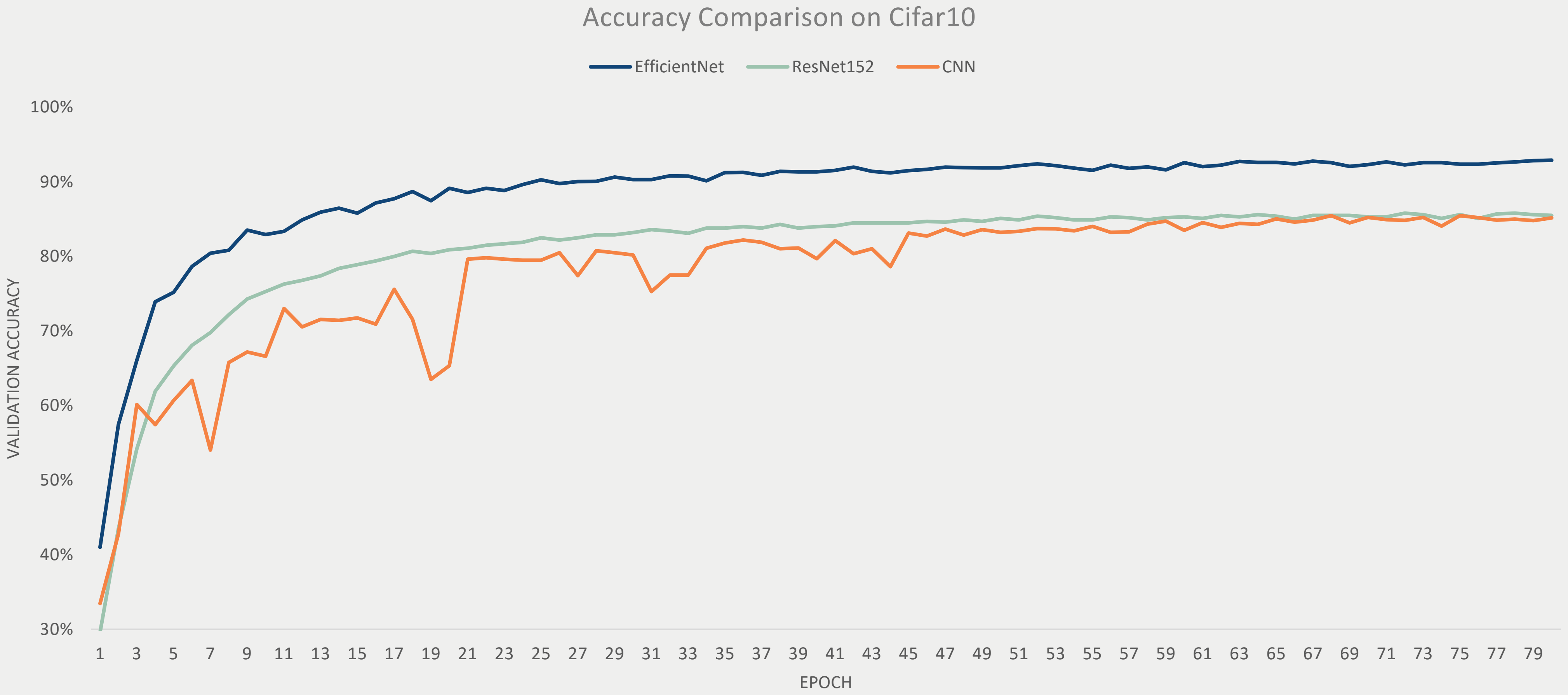
Tractability

Loyalty

EFFICIENTNET REPRODUCTION

Second Part: Comparison between EfficientNet and other Network Architectures

Compare between EfficientNet and other CNN architectures including the CNN given in our lecture and famous ResNet152 on CIFAR10 Dataset.



4

EXTENSION

EXTENSION

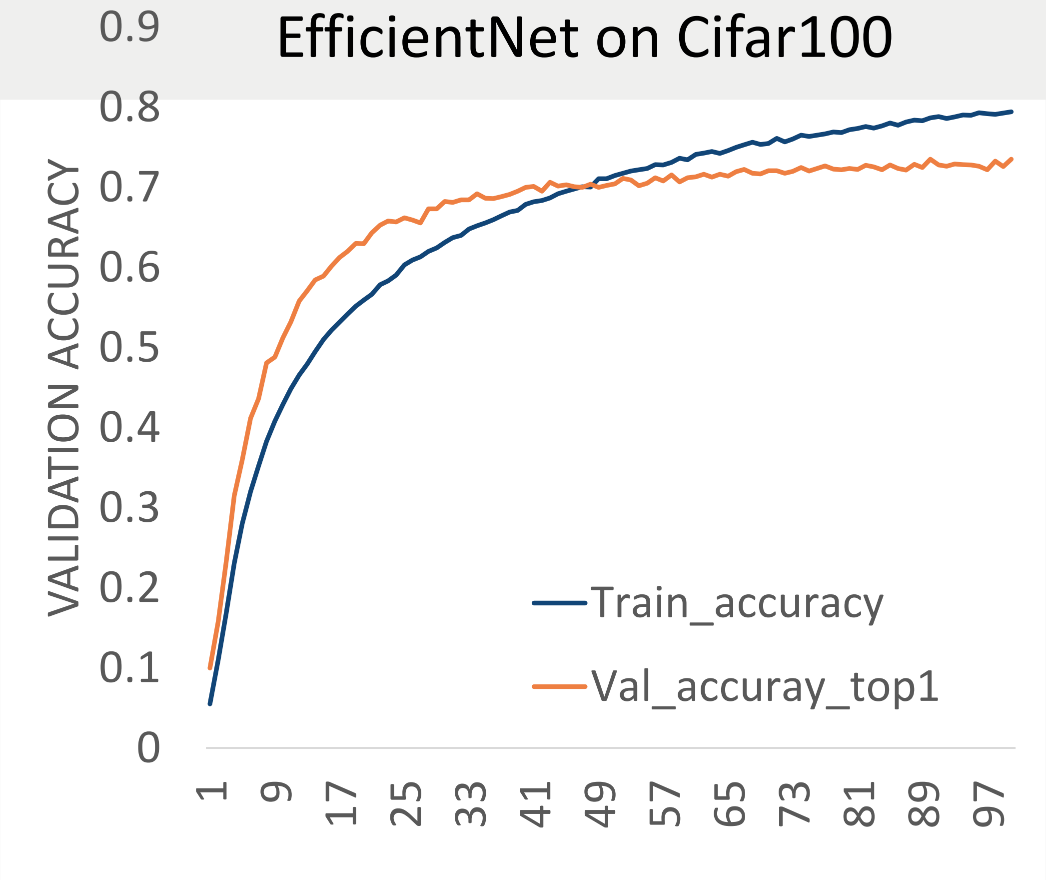
Two possible directions of extension:

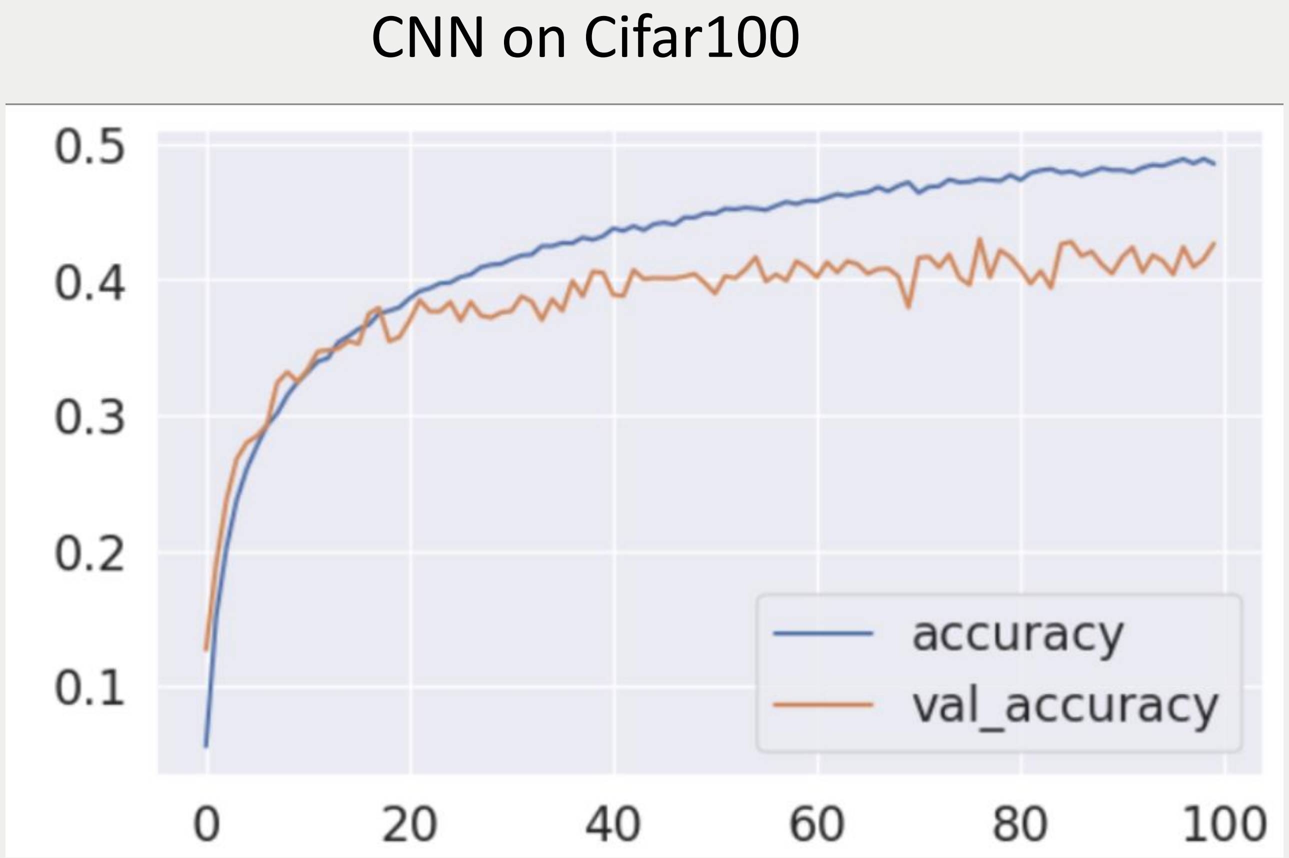
1. Apply EfficientNet on CIFAR100 dataset
2. Improve the EfficientNet Architecture

EFFICIENTNET EXTENSION

First part: Apply EfficientNet on CIFAR100 Dataset

Compare between EfficientNet and the CNN given in our lecture on CIFAR100 Dataset.





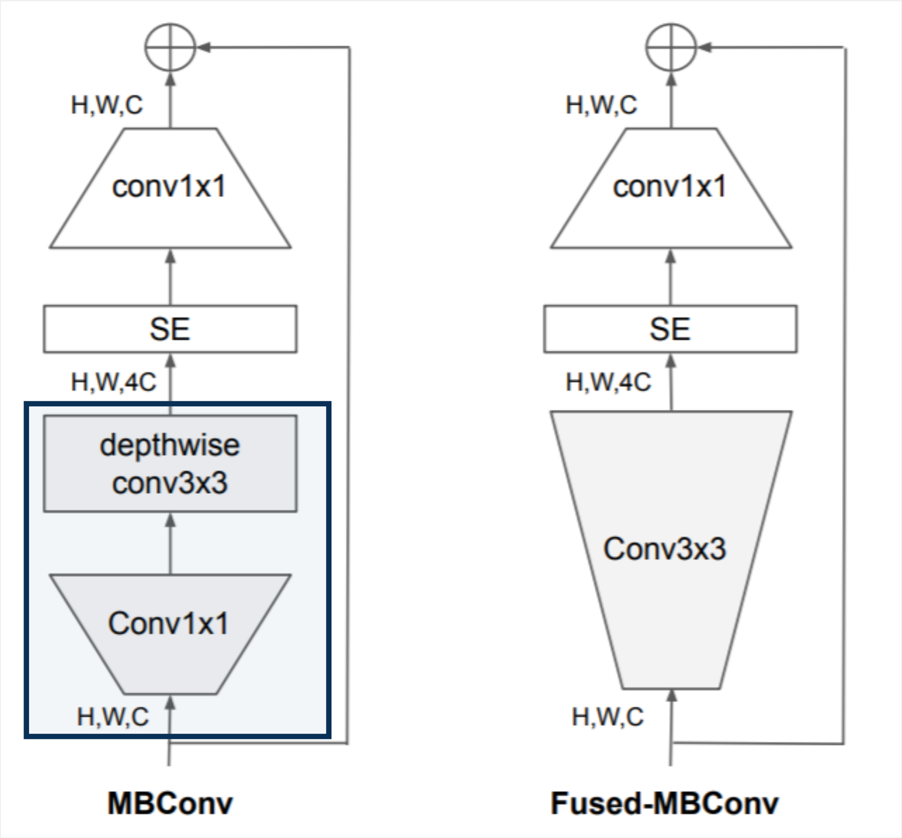
EFFICIENTNET EXTENSION

Second part: Improve EfficientNet Architecture

Limitations of the EfficientNet architecture,

1. Training with very large image sizes is slow
2. Depthwise convolutions are slow in early layers

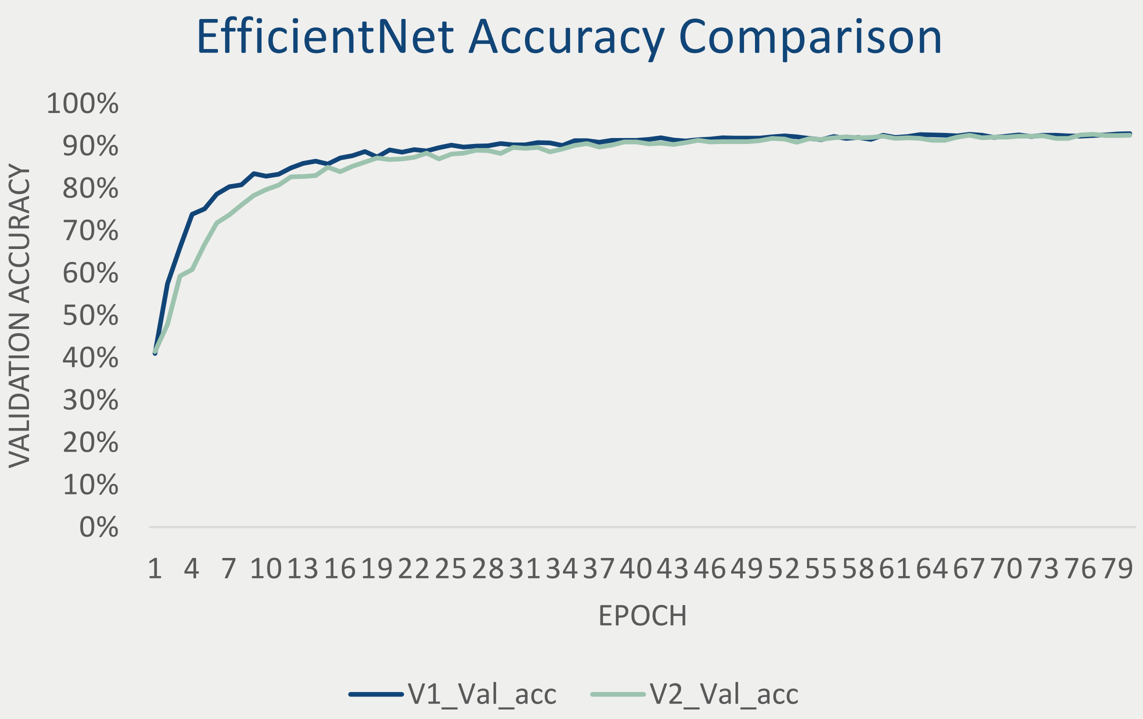
To overcome these limitations, we try to replace the depthwise 3x3 convolution and expansion 1x1 convolution in MBConv in EfficientNet with a regular 3x3 convolution.

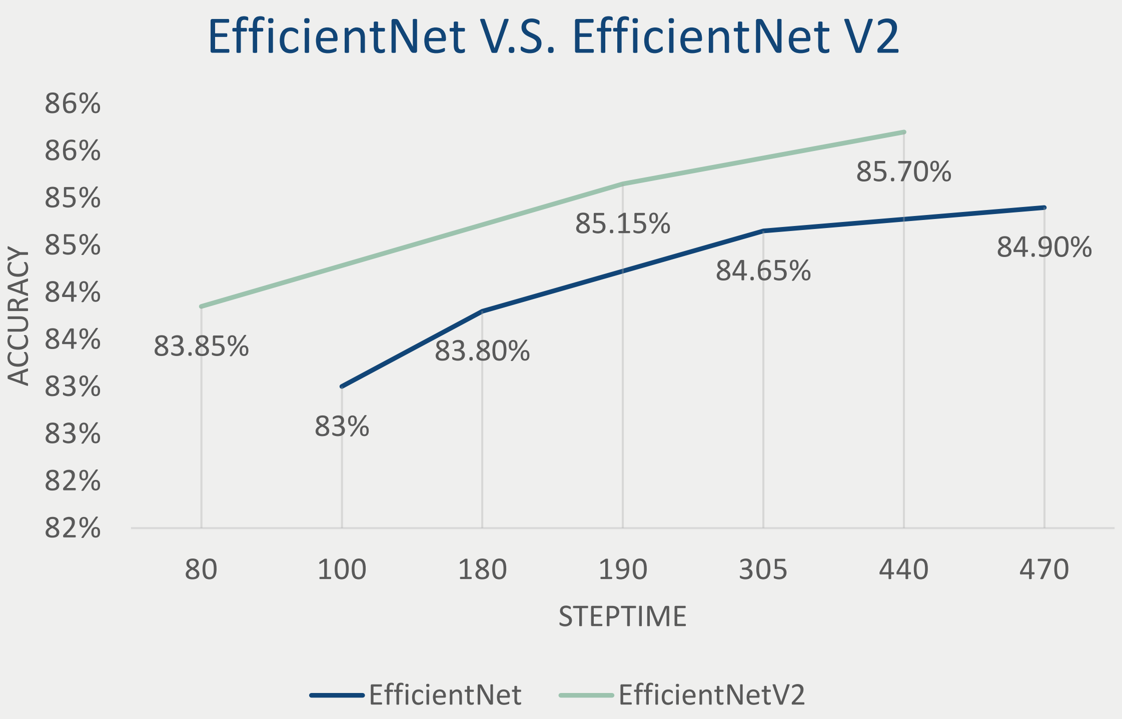


EFFICIENTNET EXTENSION

Second part: Improve EfficientNet Architecture

Compare between EfficientNet and EfficientNet V2 with respect to model performance (accuracy) and model efficiency (steptime).





CONCLUSION

SUMMARY

ResNet

How to improve training performance? →

1. Improve architecture
2. Compound scaling

Work on other datasets? Other improvements? →

1. Performance on other datasets
2. MBConv → Fused MBConv

SUMMARY

Dataset: CIFAR-10 & CIFAR-100

Platform: Colab with GPU

Library: PyTorch

Links: https://github.com/SizheYang512/DSA5204-2023-Group20

Reference (paper and code):

1. ResNet, https://arxiv.org/abs/1512.03385
2. EfficientNet, https://arxiv.org/abs/1905.11946
3. EfficientNet V2, https://arxiv.org/abs/2104.00298
4. Code: https://github.com/qubvel/efficientnet
5. Code: https://github.com/WZMIAOMIAO/deep-learning-for-image-processing/tree/master/pytorch_classification/Test9_efficientNet

THANKS

HAVE A NICE DAY!