EfficientNet

There are 3 main dimensions to scale up ConvNet

a. depth : # of layers.

b. width : # of channels

c. resolution : the size of images.

In Efficient paper, they think

" arbitrary scaling requires tedious manual tuning and still often yields sub-optimal accuracy and efficiency. "

⇒ Uniformly scales network width, depth and resolution with a set of fixed scaled coeffs.

depth by α^N, width by β^N, resolution by γ^N

Observations:

Choose a baseline conv layer $\hat{F}_i$, all layers must be scaled uniformly with const. ratio.

⇝ any kind of $\hat{F}$ is acceptable.

- We can see from paper that increase one of these params will enhance performace, but accuray gain diminishes for bigger models ⇐ less efficient.

Methods:

optimisation problem:

depth : $d = \alpha^\phi$

width : $w = \beta^\phi$

resolution : $r = \gamma^\phi$

s.t. $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$. $\alpha \geq 1$, $\beta \geq 1$, $\gamma \geq 1$

EfficientNet Architecture: Using mobile inverted bottleneck MBConv (refer to original paper)

with squeeze-and-excitation optimisation

Some details:

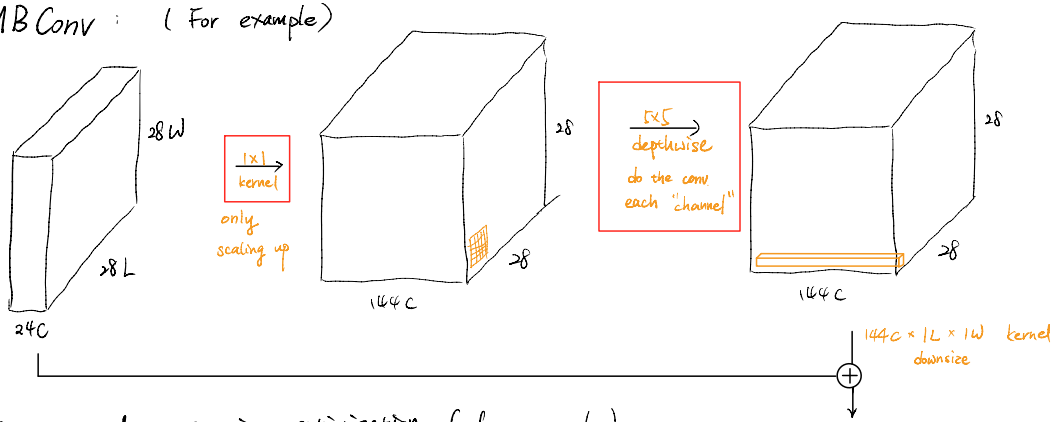
I. Using SiLU instead of ReLU

II. AutoAugment for image augmentation

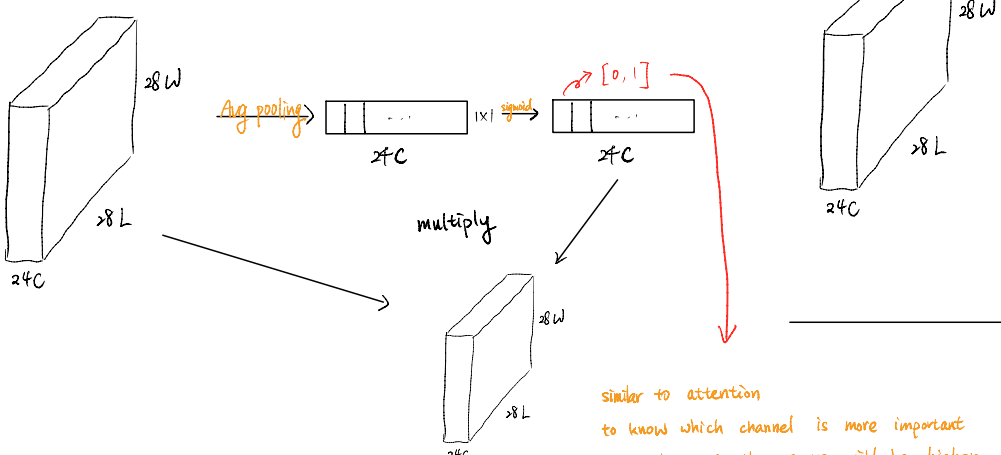
III. Stochastic Depth with survival prob. = 0.8

IV. Dropout ratio from 0.2 for B0 to 0.5 for B7

1. MBConv : (For example)

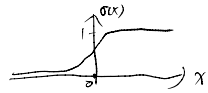


2. Squeeze-and-excitation optimisation (for example)



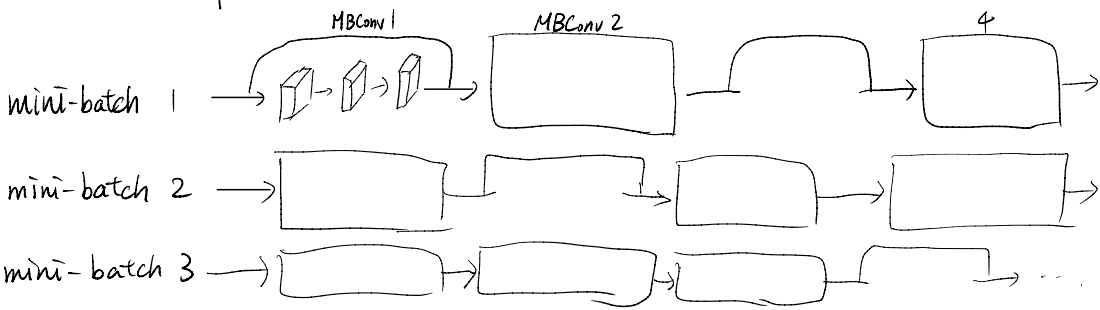
⚠ Actually, there are one conv (scaling down) and one conv (scaling up) after Avg pooling (in implementation) in order to know the attention score between diff channels (or make non-linear relationship)

3. SiLU



$$\text{silu} = x \cdot \sigma(x), \quad \sigma(x): \text{logistic sigmoid}$$

4. Stochastic Depth.



Randomly drop MBConv block (similar to dropout)

5. Optimisation problem:

$$\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2 \quad \text{since} \quad \text{depth} \cdot \text{width}^2 \cdot \text{resolution}^2 \approx \text{FLOPs}$$

For one CNN layer, FLOPs is

$$\text{output size} \times \text{kernel size}^2 \times \text{in channels} \times \text{out channels}$$

$$\gamma^2 \qquad\qquad w \qquad\qquad w$$

$$\text{total layers} = d \longrightarrow d \times w^2 \times \gamma^2$$