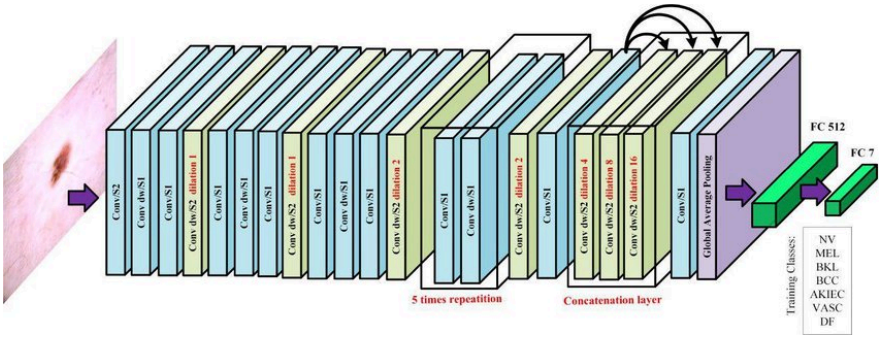
What is MobileNet?

- MobileNet is a CNN architecture that is much faster and smaller. These models are useful to be implemented on mobile and embedded devices.

- MobileNet is known to give good results while being computationally inexpensive comparatively.

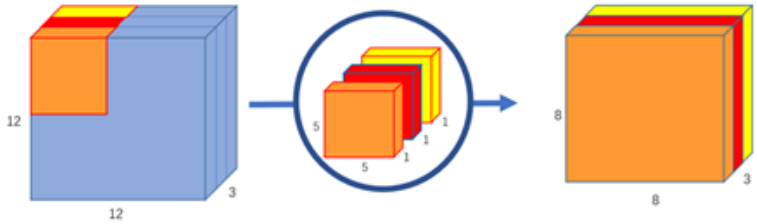
- It was the first CNN architecture to use pointwise convolutions. It makes use of a new kind of convolutional layer, known as Depthwise-Separable convolution.

- It has 10x less parameters compared to ResNet50 and still offer great results !



What is Depthwise Convolution?

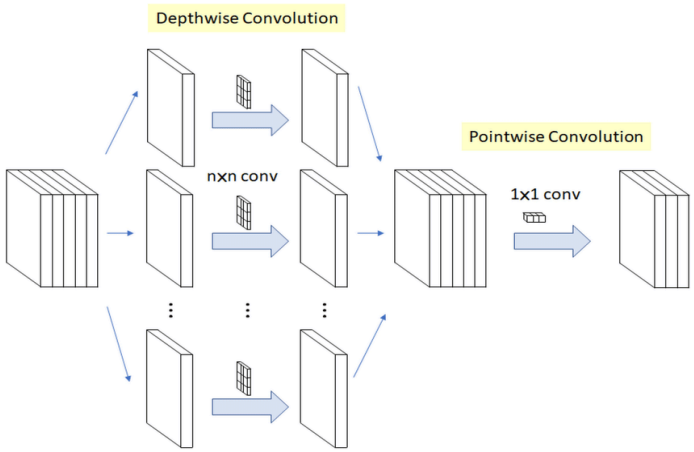
In Depthwise Convolution, we apply a single convolutional filter for each input channel



What is Depthwise-Separable Convolution?

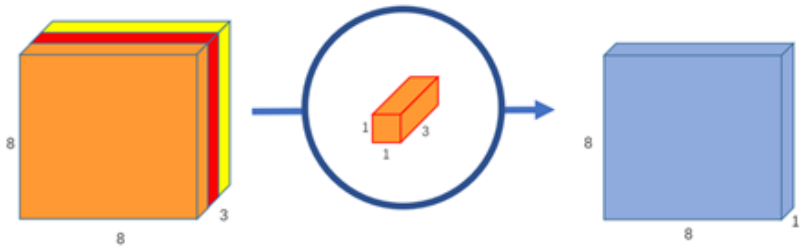
Depthwise Separable Convolution splits the computation into two steps:

Depthwise convolution: It applies a single convolutional filter per each input channel.
Pointwise convolution: It is used to create a linear combination of the output of the depthwise convolution.



What is Pointwise Convolution?

In Pointwise Convolution, we use a 1x1 kernel, which iterates across every point. This kernel has a depth equal to the number of channels in the input picture.



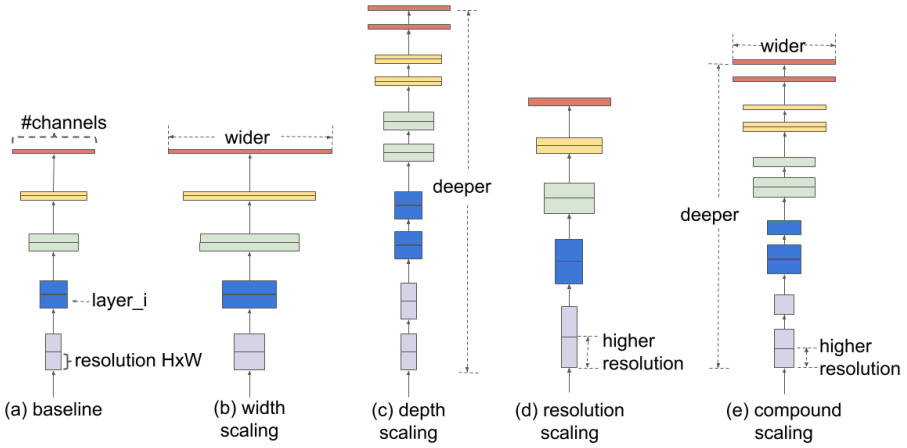
What is EfficientNet?

- EfficientNet is a state-of-the-art CNN architecture that outperforms all previous ones while using fewer parameters.

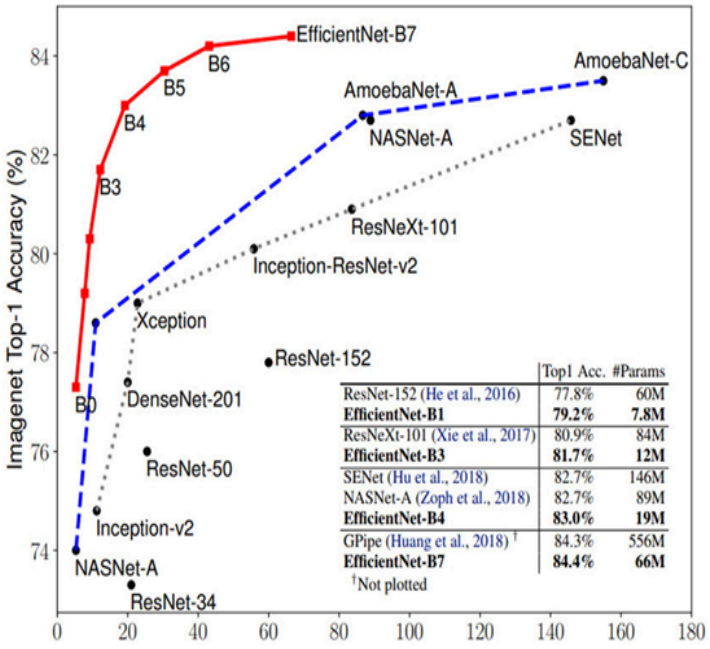
- It is a family of neural networks of different sizes, where a lot of attention was paid to the scaling of the networks in the family.

- It uses a method that uniformly scales all dimensions of depth/width/resolution using a technique called compound coefficient.

- It has eight models(B0->B7), with increasing size



Comparison of EfficientNet with Other Famous CNN Models



Model Name	Number of params	Top 1 Acc	Top 5 Acc
EfficientnetB0	5.3M	77.3	93.5
MobileNet	2.3M	71.0	90.5
ResNet50	25.6M	83.2	96.5
Inception	22.9M	79.0	94.5
VGG16	138M	74.4	91.9
AlexNet	62M	63.3	84.6

Factors to consider while using Transfer Learning

New Dataset	Similarity of new dataset with the original dataset	Action
Small and similar	Fine-tuning might lead to overfitting.	Consider using regularization techniques such as dropout or weight decay.
Large and similar	Fine-tune the pre-trained network.	This can leverage the knowledge learned from the original dataset and adapt it to the new dataset.
Large and different	Train a convolutional neural network from scratch.	This is necessary when the new dataset is significantly different from the original dataset.