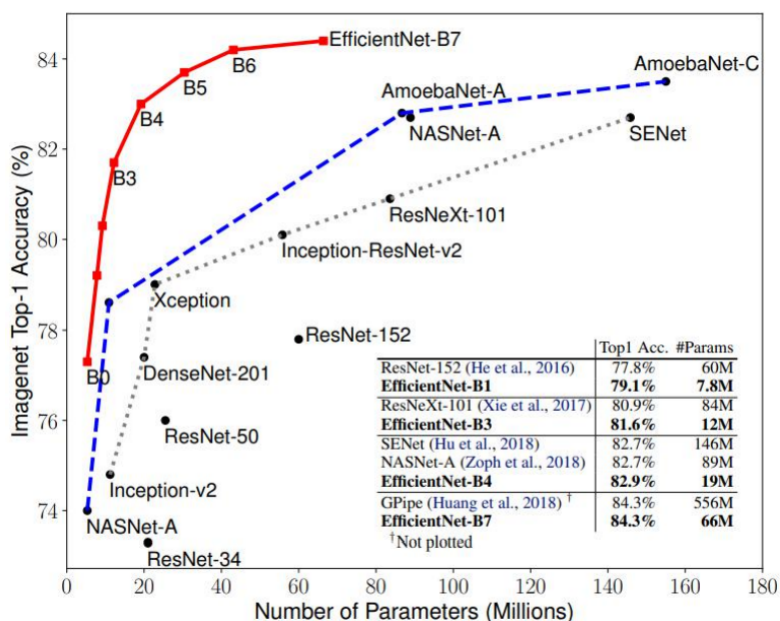


EfficientNet

- 概述

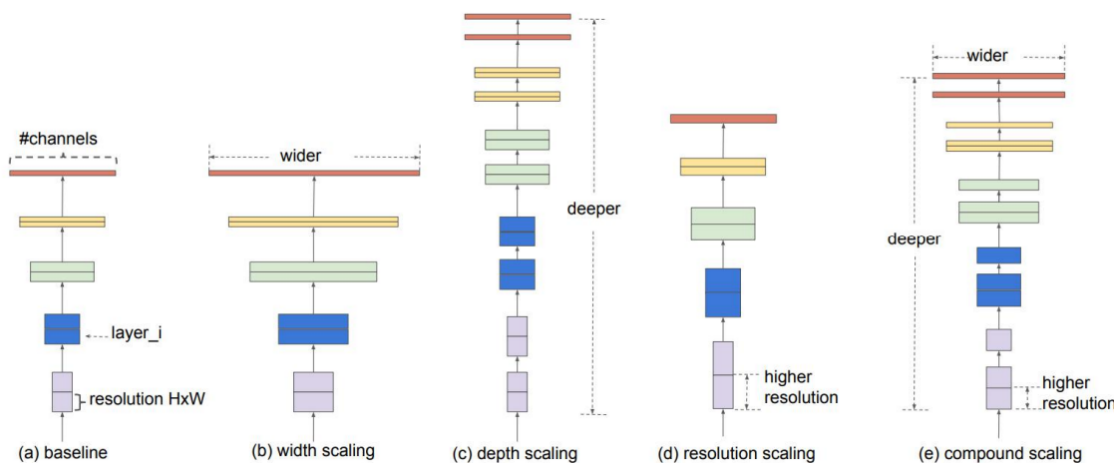
效果碾压，B0 - B7 表示各种参数的版本，B0往上考虑的因素越来越多

现在做什么都要先做个EfficientNet提取特征再做其他的



- 出发点

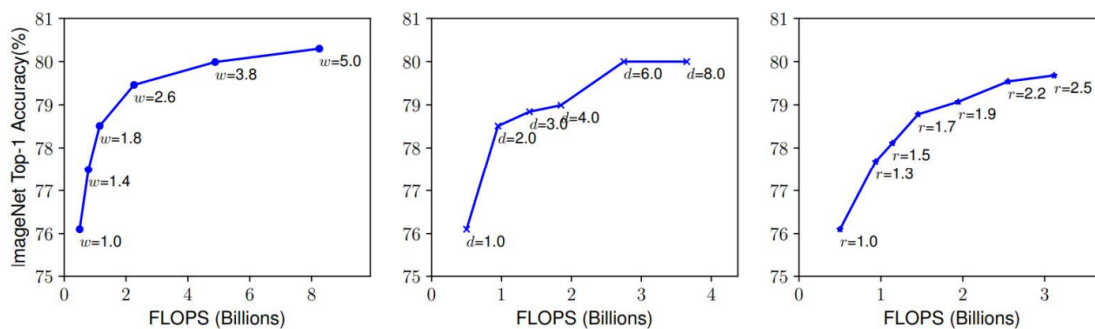
网络的特征图个数，层数，输入分辨率都会对结果产生影响



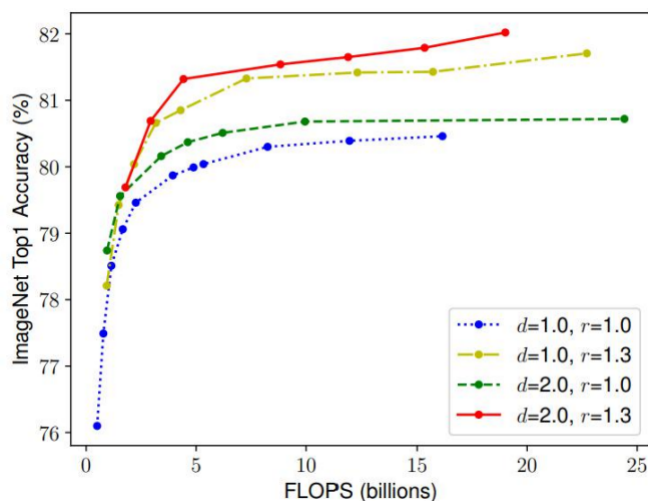
单独提升这些指标，都能使得效果有所提升，但是会有瓶颈

FLOPS表示计算复杂度，例如卷积计算量：

$$H \text{ (high)} * W \text{ (width)} * M \text{ (channels)} * K \text{ (卷积核大小)} * K * N \text{ (卷积核个数)}$$



综合提升这些指标，谷歌用参数搜索（类似于遍历）的方法来得出结果



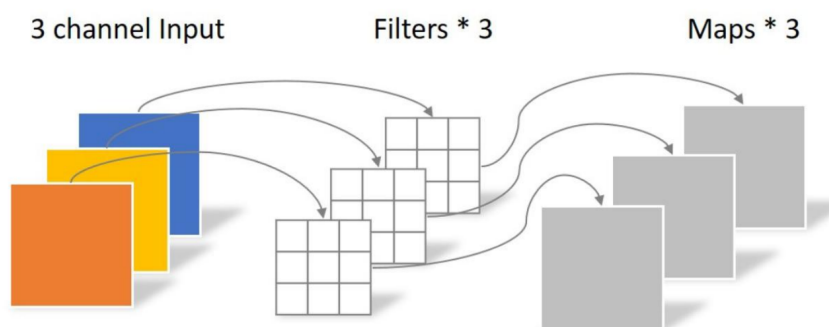
- 基本网络架构

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

MBConv6 表示权重参数升维 6倍

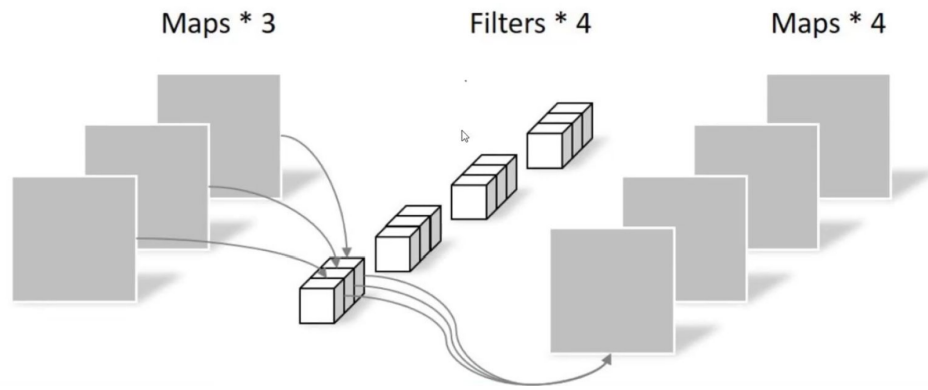
- Depthwise卷积

深度可分离卷积，原来是每个filter都要与前面的通道层相乘，参数量太大，现在是一指定然后生成对应的map特征图



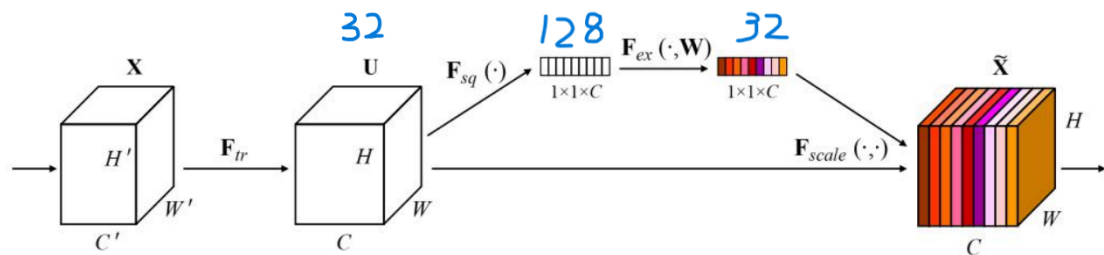
- Pointwise卷积

这里对于前面生成的map特征图再经过 1*1 的卷积核，这里的filters就可以设置很多个了



- SE模块

对于下图右边的特征图集合，对于每个特征图而言，其重要性往往不一样，需要设置权重
这里对每个特征图计算其权重（注意力机制）



上线路中，从32到128做了一次升维，从128到32做了一次降维，第二个32即是权重参数，前面 Conv6 就表示先升维6倍

- 计算流程

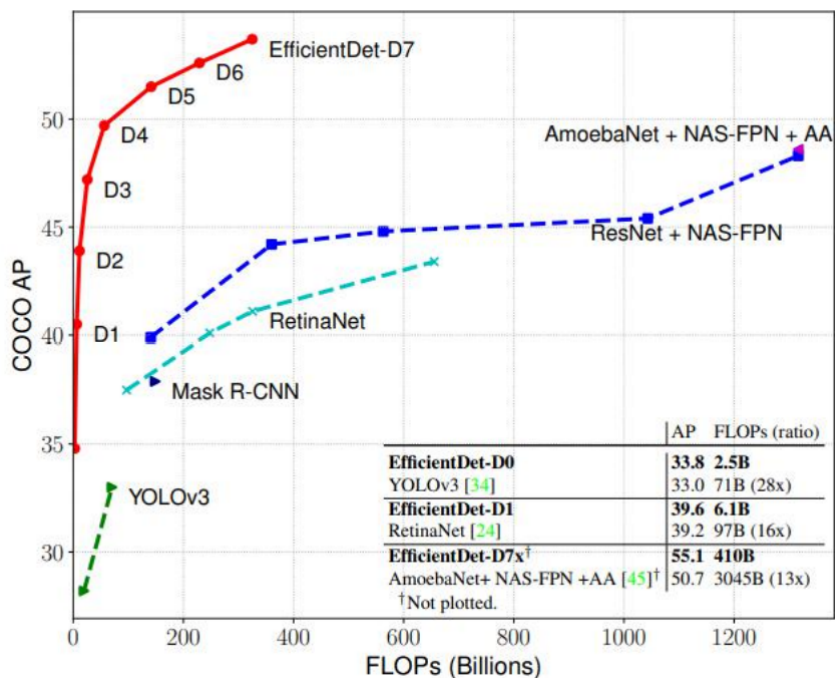
下面的res表示分辨率大小， dropout表示是否需要

```
def efficientnet_params(model_name):
    """ Map EfficientNet model name to parameter coefficients. """
    params_dict = {
        # Coefficients: width, depth, res, dropout
        'efficientnet-b0': (1.0, 1.0, 224, 0.2),
        'efficientnet-b1': (1.0, 1.1, 240, 0.2),
        'efficientnet-b2': (1.1, 1.2, 260, 0.3),
        'efficientnet-b3': (1.2, 1.4, 300, 0.3),
        'efficientnet-b4': (1.4, 1.8, 380, 0.4),
        'efficientnet-b5': (1.6, 2.2, 456, 0.4),
        'efficientnet-b6': (1.8, 2.6, 528, 0.5),
        'efficientnet-b7': (2.0, 3.1, 600, 0.5),
        'efficientnet-b8': (2.2, 3.6, 672, 0.5),
        'efficientnet-l2': (4.3, 5.3, 800, 0.5),
    }
    return params_dict[model_name]
```

EfficientDet

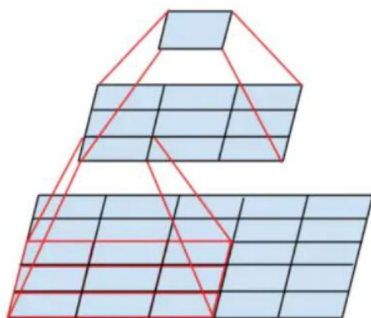
- 概述

效果很好，相同重量级优势明显



整体 = EfficientNet + BiFPN

原金字塔结构的融合，若只做上采样（尺寸小的感受野大的→尺寸大的感受野小的，下图即是往下，即往初始的方向）



特征融合时这里若全部都映射成最下面的大尺寸特征图，则只能优化识别小物体，而偏置中大物体