

VGG 加深了网络的层数，使用几个小卷积核代替一个大卷积核，不仅减少了参数量，而且还加深了网络，这样类似的做法在很多网络中都有出现过，比如在 EfficientNetV2 中，将 V1 中的 5×5 卷积换为 3×3 ，考虑到感受野的问题所以就多叠加了几次。

像 VGG 这样深的网络必然会引起参数量增加，那么 InceptionV1 是通过将原来卷积层从使用多个单一的卷积操作变为使用多种卷积核，减少了冗余的参数的同时又丰富了特征，相比于 InceptionV1 认为冗余的特征信息对网络性能不是一件好事，GhostNet 认为一个神经网络其提取到的丰富甚至冗余的特征信息保证了模型对输入图像的全面理解，为了利用冗余的特征，同时兼顾网络计算量以及参数量，使用一种简单而有效的线性运算生成更多冗余的特征。GhostNet 就是探索了一种简单的方法达到了相同的目的，那么 ResNet 也是如此，本来网络要学习的是一个复杂的变换，加上跳跃连接后，网络需要学习的东西就改变了，简化为学习输出和输入差值的那一部分。

从另一个角度来看，ResNet 其实也是一种简单的特征复用，而 DenseNet 则是一种密集的特征复用，使用 `concat` 拼接方式实现将浅层和深层特征结合，但是说回到特征冗余这一方面，`concat` 拼接后的特征肯定也有大量的冗余，这里解决这个问题的办法是直接限制每个模块输出特征图的个数，那么如果说要改进的话，我自己最先想到的就是像 MobileNetV3 中的那样使用 SE 模块学习通道的权重参数，不过这样的操作已经很常见了。

CSPNet 更像是将 ResNet、DenseNet、Inception 揉在一起，通过不同的融合方式寻找最有效的组合方式。

MobileNetV1 使用深度可分离卷积减少计算量，这在 V2V3 中都有继承。V2 里主要是使用了倒残差的结构，其实这种结构并没有带来计算量上的改变，它影响的主要是不让网路的特征被 ReLU 函数破坏掉，从而保证最后得到的低维特征的有效性。V3 减小计算量的手段就是更换了激活函数，又根据激活函数的特点来决定更换哪些层的激活函数。

MobileNetV1 中提到的宽度因子和分辨率因子就简单地做了几个实验说明了一下哪种组合效果更好，没有把这两个因素作为轻量化网络的重点改造对象，而在 EfficientNetV1 中对网路的宽度、深度、分辨率大小展开研究，并从这三个

因素来轻量化网络。并且在 MobileNet 系列中使用的深度可分离卷积虽然减少了计算量，但是速度慢，将其换为 3×3 卷积虽然参数量略有提高单数速度有大幅提高。那么在以后在设计模型时，假如设计了一个很有效果的结构，可以考虑是不是不用将所有模块都进行替换？替换掉哪些层的效果更好？EfficientNetV2 中将 V1 中 5×5 换为 3×3 那么结合 RepVGG 中结构重参数化的思想不替换 5×5 ，而是将 3×3 重参数化为 5×5 ，也不用堆叠那么多次，是不是效果还会更好一些？

再从信息交互的角度看，ShuffleNet 中的通道重排、MicroNet 中的 Dynamic Shift-Max 激活函数和 SwinTransformer 中的 cyclic shift 操作都关注到了如果要将图像分块单独处理的话一定要引入一种方式加强信息交互。