Inception-V3 Notes

1. 背景

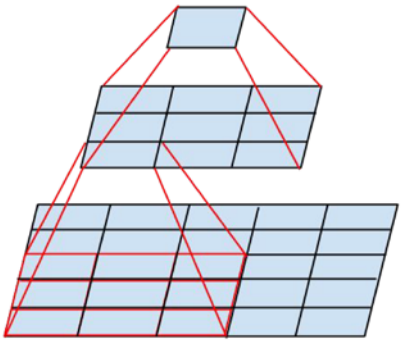
GoogLeNet 的结构很难被调整去适应新的应用场景，因为这一结构本身的优点就是计算代价较小，如果直接增加网络的深度，这一优点也就没有了。

2. 设计的一般原则

- 要避免特征描述瓶颈（representational bottleneck）。所谓特征描述瓶颈就是中间某层对特征图在空间维度进行较大比例的压缩（比如使用 pooling 时），导致很多特征丢失。这里的特征图压缩指的是输入的特征图，而不是输出的特征图。对于输入的特征图，先进行卷积操作，然后再池化降维，这种做法是不违反这一规则的。
- 特征的数目越多（特征图 channel 数）收敛的越快。相互独立的特征越多，输入的信息就被分解的越彻底，分解的子特征间相关性低，子特征内部相关性高，把相关性强的聚集在了一起会更容易收敛。
- 可以压缩特征维度数，来减少计算量。Inception-v1 中提出的用 1x1 卷积先降维再作特征提取就是利用这点。
- 整个网络结构的深度和宽度（特征维度数）要做到平衡。文章中提到了“the optimal improvement for a constant amount of computation can be reached if both are increased in parallel”，这句话指的应该就是 ResNet 中提到的当特征图长宽减少为原来的一半时，把特征图的 Channel 增加为原来的二倍，这样每个卷积层计算的复杂度都是一样的。

3. 卷积分解

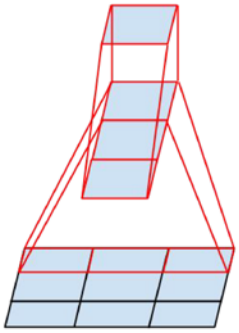
3.1. 将大卷积核分解成多个小卷积核



左图是将一个 5*5 的卷积核替换成两个 3*3 的卷积核
卷积核替换之后，一个需要考虑的问题是这两个 3*3 卷积之间要不要用非线性激活函数，作者通过实验发现非线性激活函数比线性激活函数效果要好

Figure 1. Mini-network replacing the 5×5 convolutions.

3.2. 空间分解成不对称的卷积结构



左图是将一个 3*3 卷积分解成一个 1*3 的卷积+一个 3*1 的卷积
理论上来说，任何一个 n*n 的卷积都可以被分解成一个 1*n+一个 n*1 的卷积，不过作者通过实验发现，这种卷积结构用在特征图大小为 12 到 20 之间时才比较有效果

Figure 3. Mini-network replacing the 3×3 convolutions. The lower layer of this network consists of a 3×1 convolution with 3 output units.

4. 辅助分类器

作者认为辅助分类器只是起到了正则化的作用

5. Grid Size Reduction

指的就是在网络中的某一层，将输入的特征图大小减半，同时通道数增加一倍

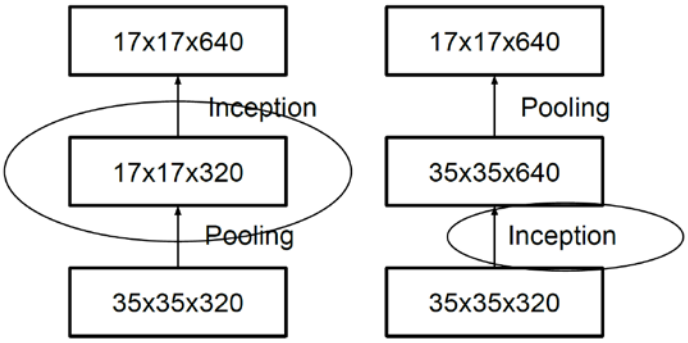


Figure 9. Two alternative ways of reducing the grid size. The solution on the left violates the principle 1 of not introducing an representational bottleneck from Section 2. The version on the right is 3 times more expensive computationally.

上图是两种可能的做法。第一种是对输入的特征图先进行池化，然后再卷积，作者认为这种做法违反了 representational bottleneck 设计原则。第二种做法是先进行卷积，再做池化，不过这种做法计算代价太大了。

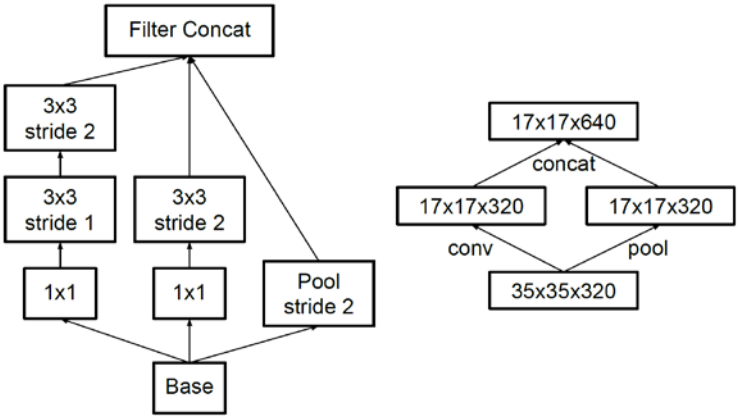
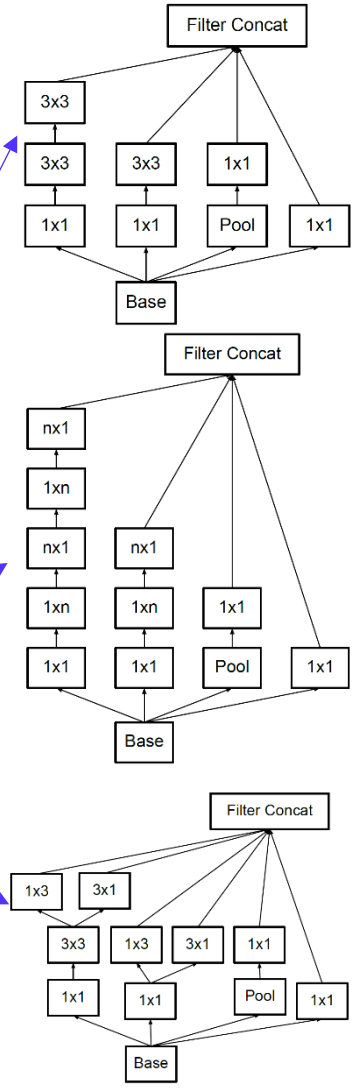


Figure 10. Inception module that reduces the grid-size while expands the filter banks. It is both cheap and avoids the representational bottleneck as is suggested by principle 1. The diagram on the right represents the same solution but from the perspective of grid sizes rather than the operations.

上图是作者提出的改进方案，对于输入的特征图，分别用 stride=2 的卷积和池化进行运算，然后把这两种操作得到的特征图在 Channel 维度上拼接起来。

6. Inception-V2

type	patch size/stride or remarks	input size
conv	3×3/2	299×299×3
conv	3×3/1	149×149×32
conv padded	3×3/1	147×147×32
pool	3×3/2	147×147×64
conv	3×3/1	73×73×64
conv	3×3/2	71×71×80
conv	3×3/1	35×35×192
3×Inception	As in figure 5	35×35×288
5×Inception	As in figure 6	17×17×768
2×Inception	As in figure 7	8×8×1280
pool	8 × 8	8 × 8 × 2048
linear	logits	1 × 1 × 2048
softmax	classifier	1 × 1 × 1000



7. **降低输入图片分辨率**

   若要降低模型的时间代价，可以采用浅层模型，也可以降低输入图片的分辨率，作者认为第二种方法更好。

8. **Inception-V3**

Inception-v2 BN-auxiliary	**21.2**%	**5.6**%	4.8