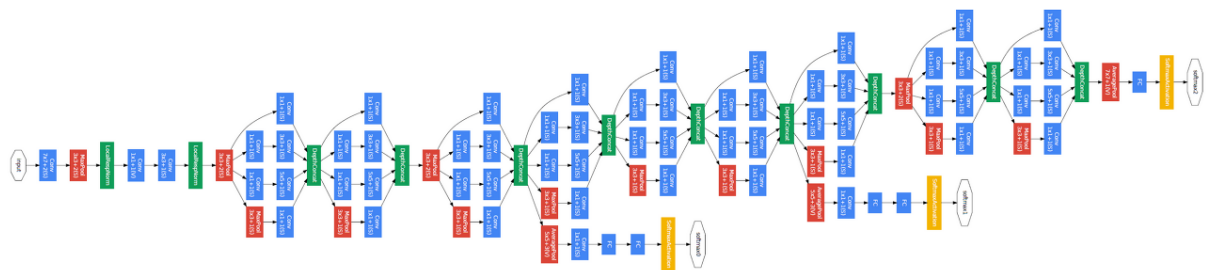


GoogLeNet

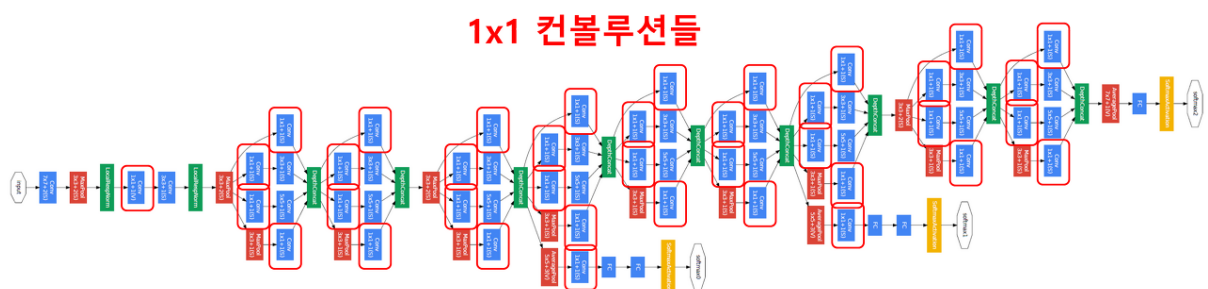
GoogLeNet은 2014년 ILSVRC에서 VGGNet을 이기고 우승을 차지한 알고리즘이다. 19층보다 좀 더 깊은 22층으로 구성되어 있다.

논문: Going Deeper with Convolutions

GoogLeNet의 구조



1 x 1 컨볼루션

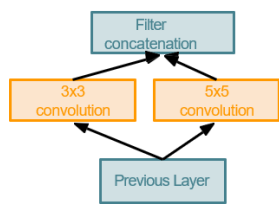


1x1 컨볼루션은 어떤 의미를 갖는 것일까? GoogLeNet에서 1x1 컨볼루션은 feature map의 갯수를 줄이는 목적으로 사용된다>>요약이라 생각하면 됨

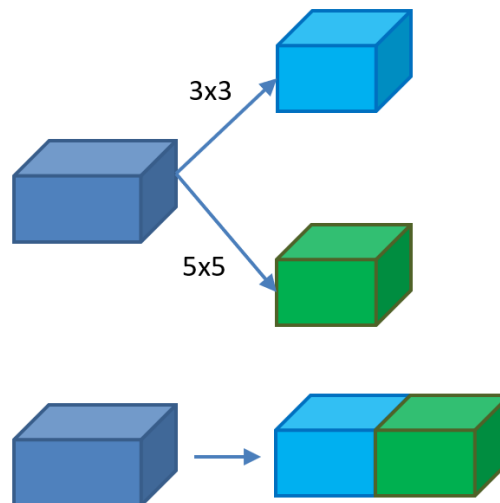
특성맵의 갯수가 줄어들면 그만큼 연산량이 줄어든다.

Case Study: GoogLeNet

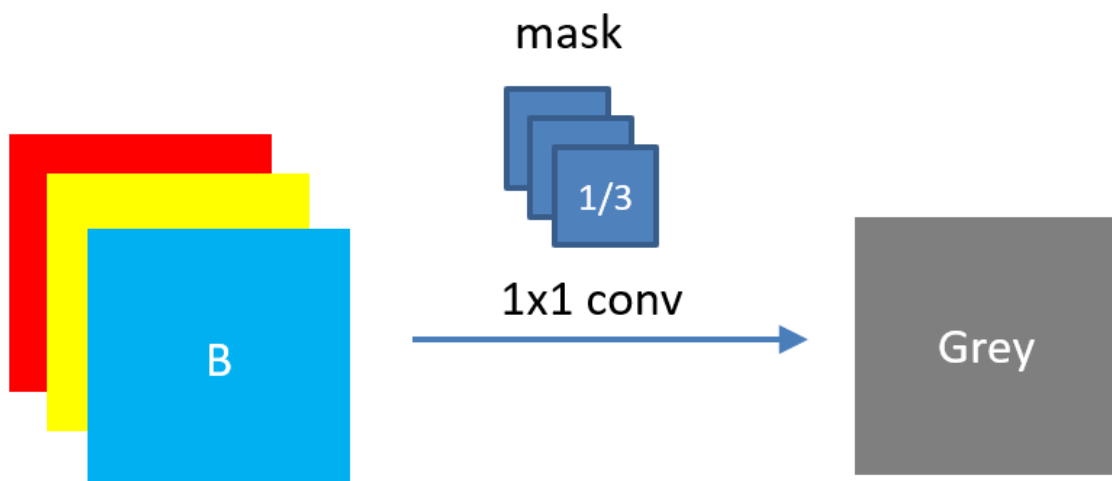
[Szegedy et al., 2014]

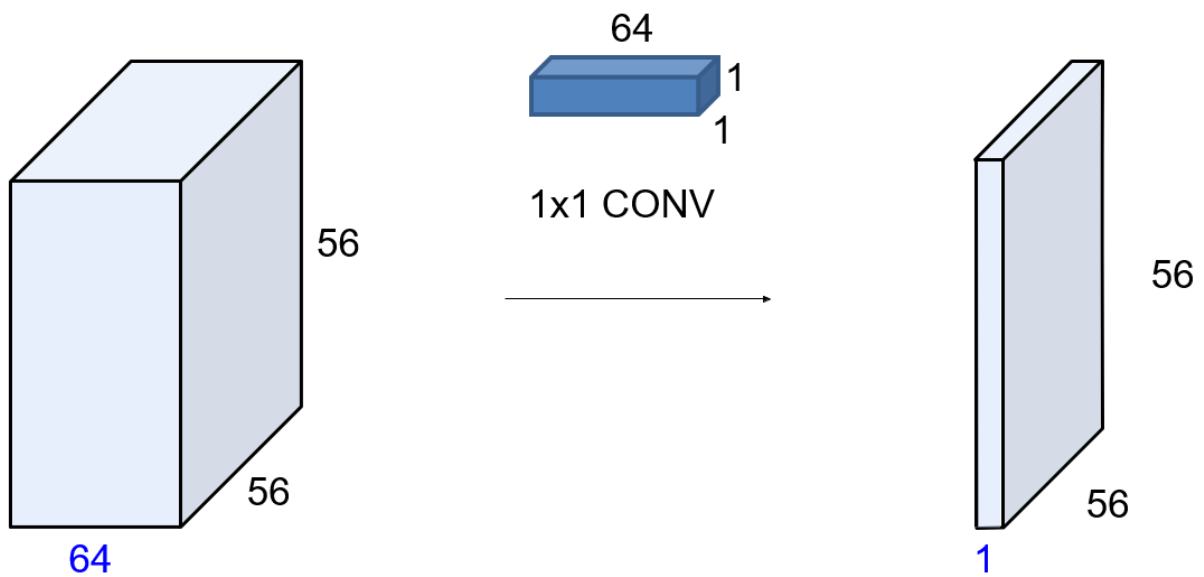


Naive Inception module

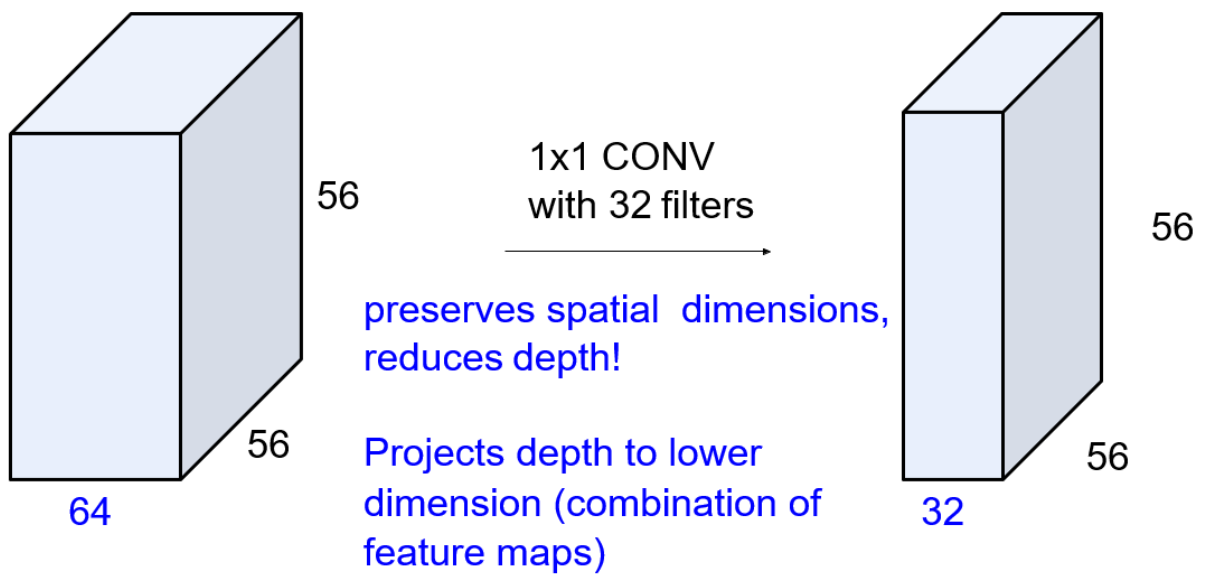


이런식으로 concatenate

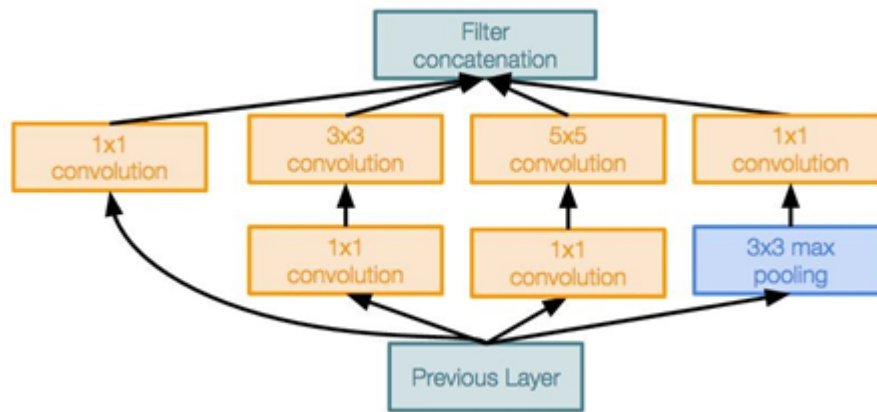




이 그림을 보면 64개의 input이 1개의 input으로 축약되었다.

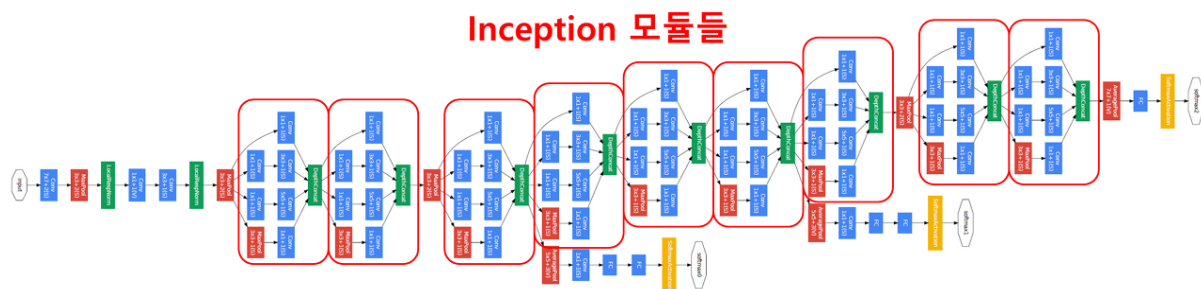


계산량이 준다.

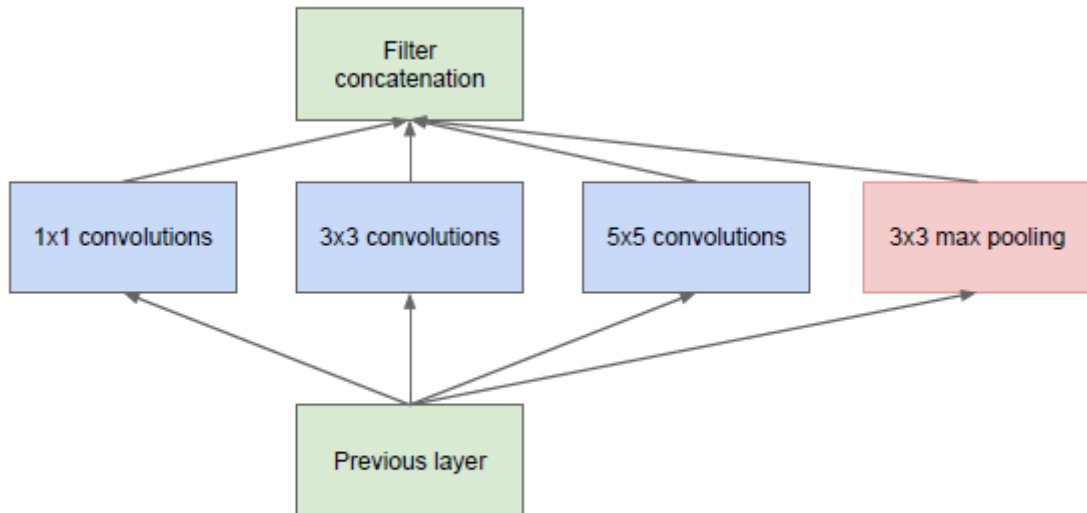


Inception 모듈

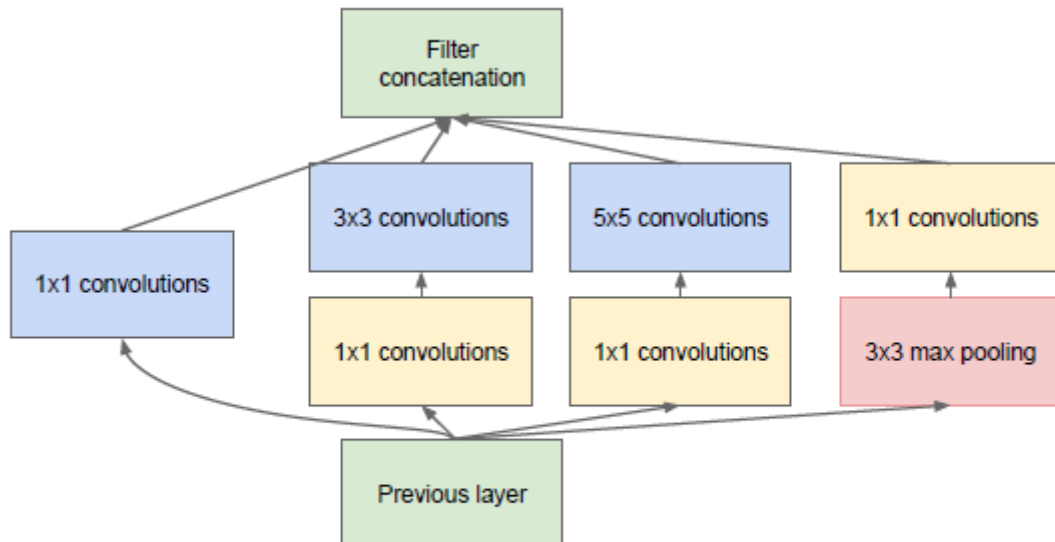
위에서 잠깐 그림으로 보였듯이 GoogLeNet의 핵심인 Inception모듈이다.



GoogLeNet은 총 9개의 인셉션 모듈을 포함하고 있다.



(a) Inception module, naïve version



(b) Inception module with dimensionality reduction

GoogLeNet에 실제로 사용된 모듈은 1x1 컨볼루션이 포함된 (b) 모델이다. 아까 살펴봤듯이 1x1 컨볼루션은 특성맵의 장수를 줄여주는 역할을 한다. 노란색 블록으로 표현된 1x1 컨볼루션을 제외한 나이브(naive) 버전을 살펴보면, 이전 층에서 생성된 특성맵을 1x1 컨볼루션, 3x3 컨볼루션, 5x5 컨볼루션, 3x3 최대풀링해준 결과 얻은 특성맵들을 모두 함께 쌓아준다. AlexNet, VGGNet 등의 이전 CNN 모델들은 한 층에서 동일한 사이즈의 필터커널을 이용해서 컨볼루션을 해줬던 것과 차이가 있다. 따라서 좀 더 다양한 종류의 특성이 도출된다. 여기에 1x1 컨볼루션이 포함되었으니 당연히 연산량은 많이 줄어들었을 것이다.

>>그니까 연산량을 줄이고 비선형성도 증가!!

오른쪽에 보면 Max Pooling을 왜할까?? 아 참고로 stride=1이다.아 zero padding도 있다.

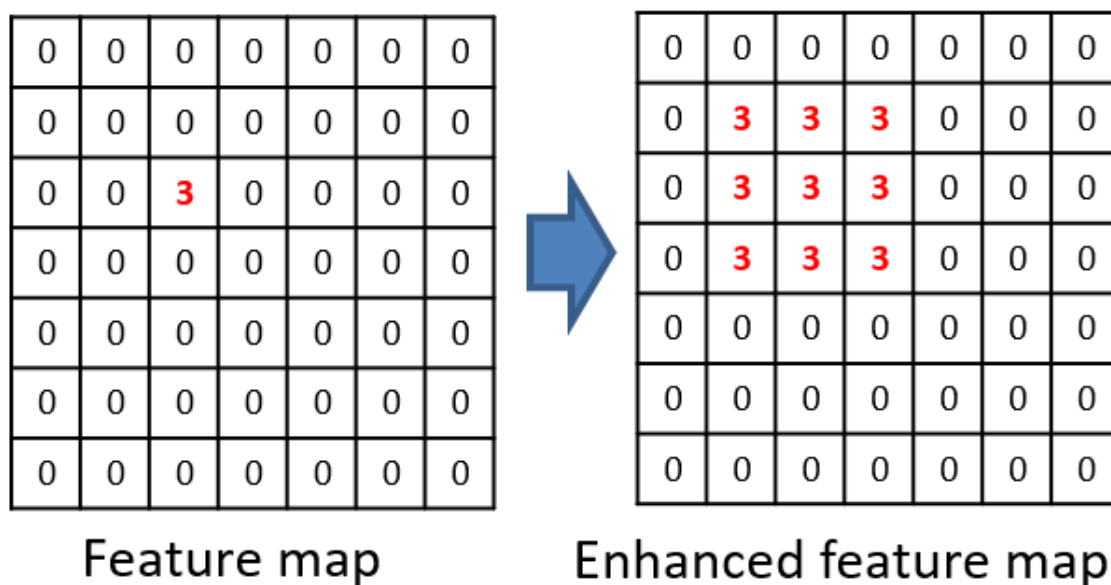
그렇다면

input:32x32x64

output:32x32x64

형태가 전혀 바뀌지 않는다. 그러나 pooling을 하면

3x3 max pooling, stride=1



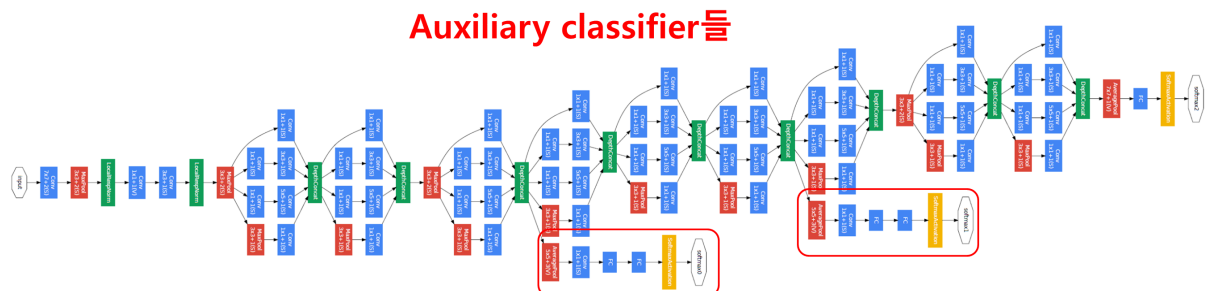
그후 1x1 convolution을 하면 feature map 축약(channel 수만 바뀜)+ 1x1 convolution을 추가함으로써 non-linearity 증가

1x1 Convolution



auxiliary classifier

네트워크의 깊이가 깊어지면 깊어질수록 vanishing gradient 문제를 피하기 어려워진다. 그러니까 가중치를 훈련하는 과정에 역전파(back propagation)를 주로 활용하는데, 역전파과정에서 가중치를 업데이트하는데 사용되는 gradient가 점점 작아져서 0이 되어버리는 것이다. 따라서 네트워크 내의 가중치들이 제대로 훈련되지 않는다. 이 문제를 극복하기 위해서 GoogLeNet에서는 네트워크 중간에 두 개의 보조 분류기(auxiliary classifier)를 달아주었다.



보조 분류기의 구성을 살펴보면, 5 x 5 평균 풀링(stride 3) -> 128개 1x1 필터커널로 컨볼루션 -> 1024 FC 층 -> 1000 FC 층 -> softmax 순이다. 이 보조 분류기들은 훈련시에만 활용되고 사용할 때는 제거해준다.