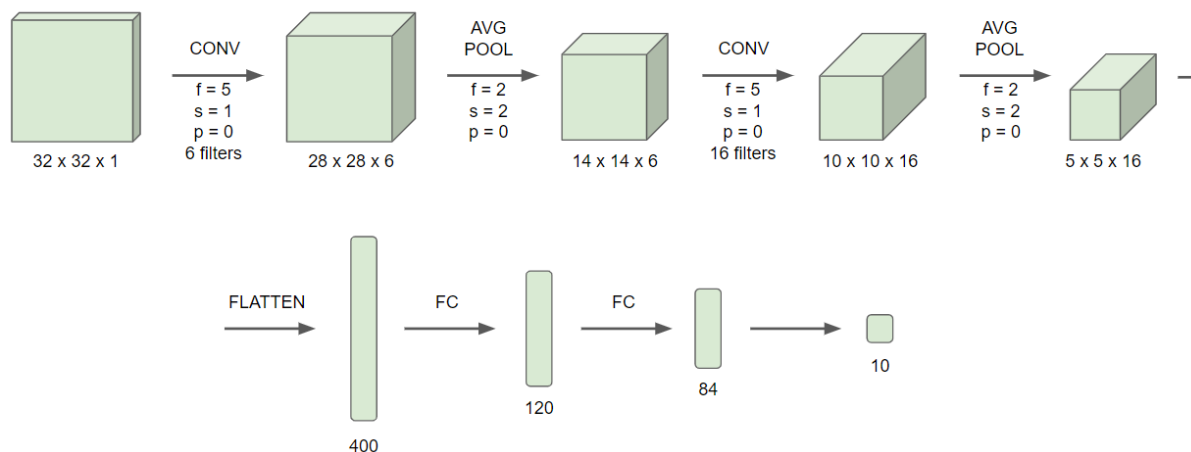


2. 케이스 스터디

Classic Network

LeNet-5

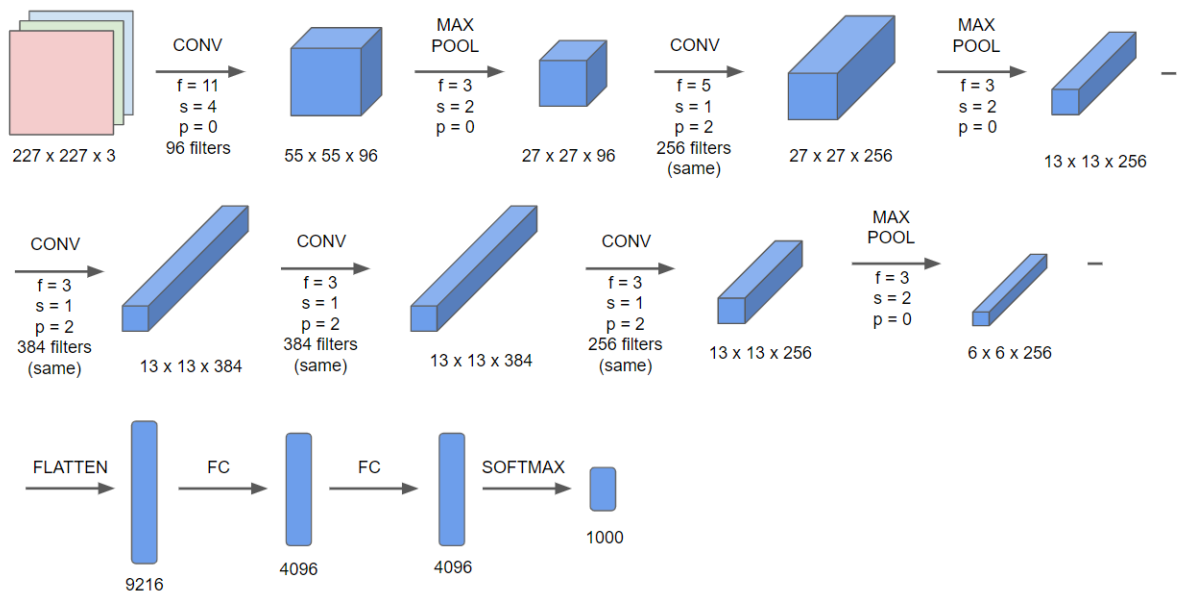
LeNet - 5



- 목적: 손글씨 인식
- input size: 32x32x1 (흑백)
- average pooling 사용
- FC: 400 → 120 nodes / 120 → 84
- output size: 84 units → 0~9 classifications
- classifier: 옛날 모델 (softmax 아님)
- 신경망이 깊어질수록 높이와 너비가 감소하고 채널 수가 증가함
- Sigmoid와 tanh 사용 (Relu x)
- 컴퓨터가 느려 변수의 계산으로 줄이기 위해 각각의 필터가 서로 다른 채널에 적용됨 (요즘에는 사용x)
- LeNet-5의 비선형성이 pooling층 뒤에 적용됨 (요즘과 다름)

AlexNet

AlexNet



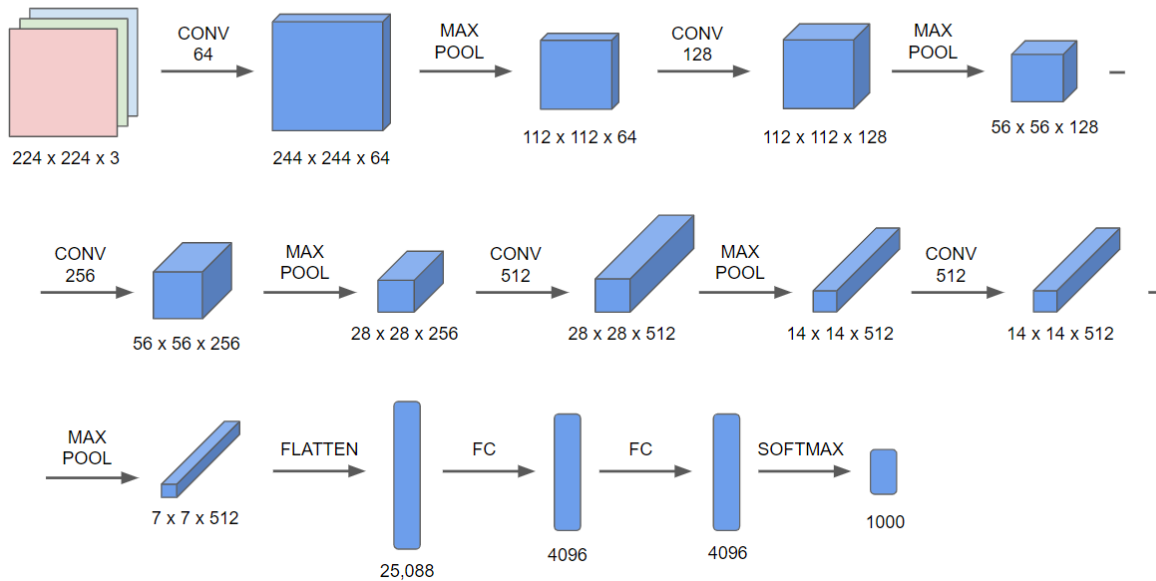
- input size: 227x227x3
- ~60m parameters
- LeNet과 비슷하지만 훨씬 큰 신경망, LeNet보다 뛰어난 성능
- ReLu 사용
- Multiple GPUs
- Local Response Normalization이라는 층이 있었음
 - 높이와 너비가 지정된 한 지점의 모든 채널을 정규화 → 그리 유용하지 않음

VGG-16

VGG-16

CONV : 3 x 3 filter, s=1, same

MAX POOL: 2 x 2, s=2

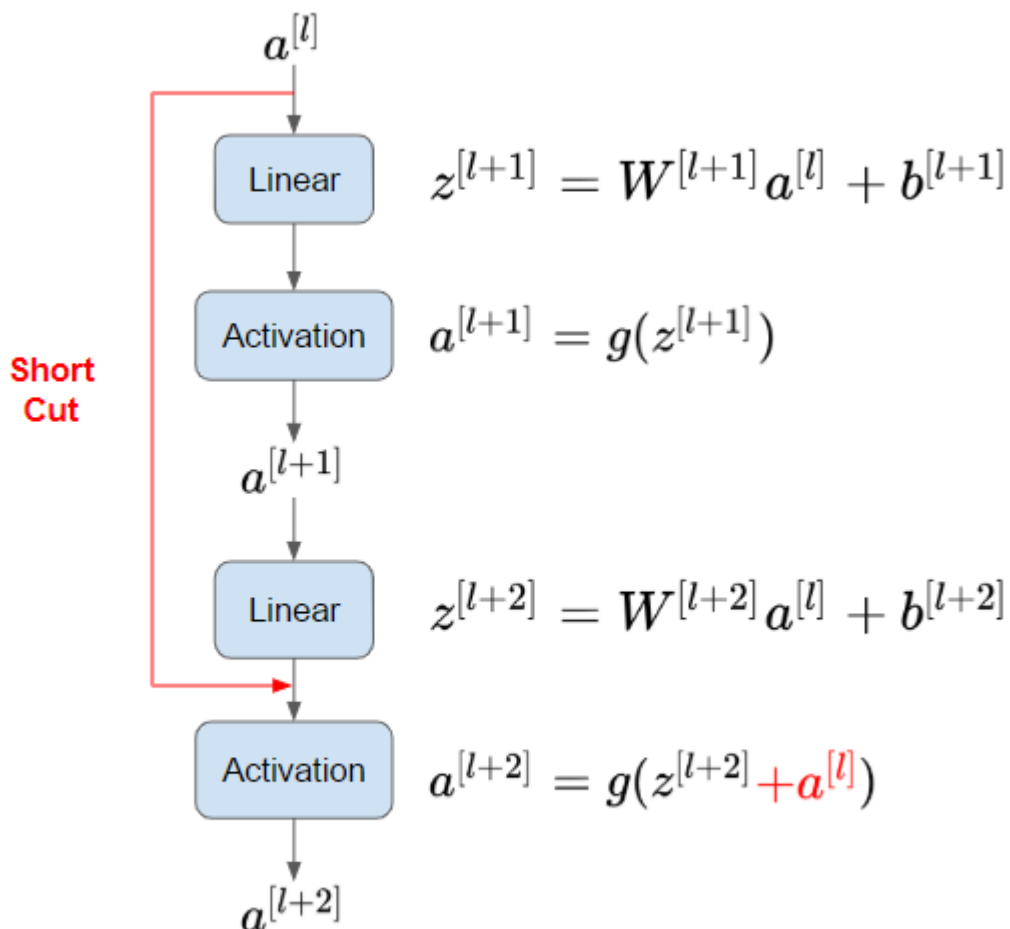
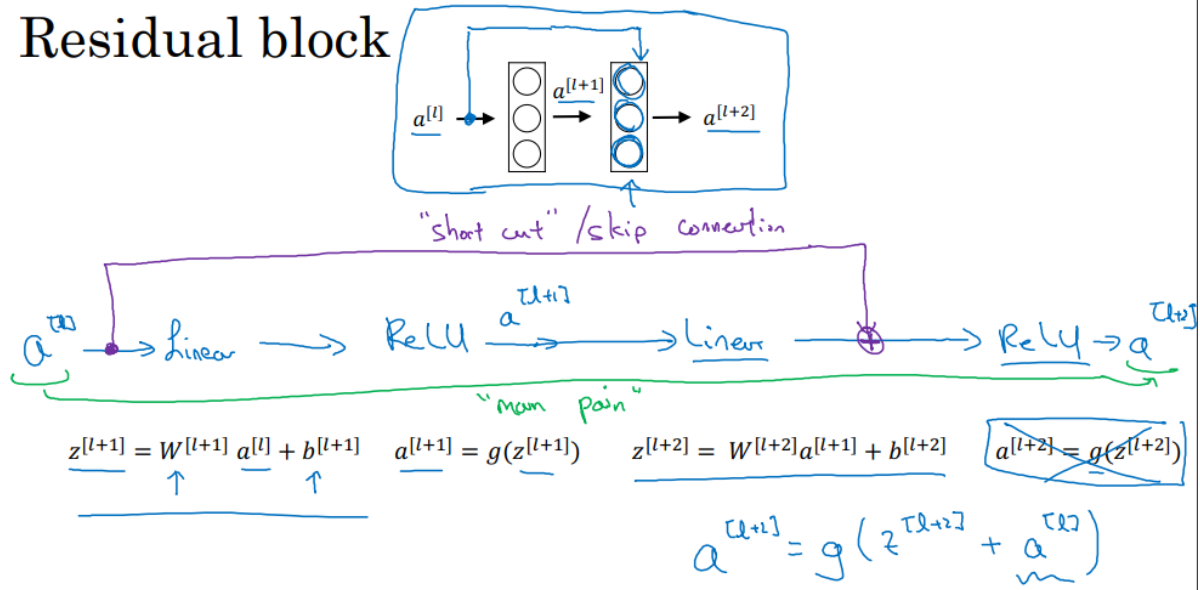


- conv layer: stride가 1인 3x3 filter, same
- max pooling: stride가 2인 2x2 filter
- 균일하고 간결한 구조
 - conv layer에서 채널 수를 두 배씩 증가시킴
- ~138M parameters
- 단점: 파라미터가 많아 네트워크의 크기가 커짐

Resnets

Residual block

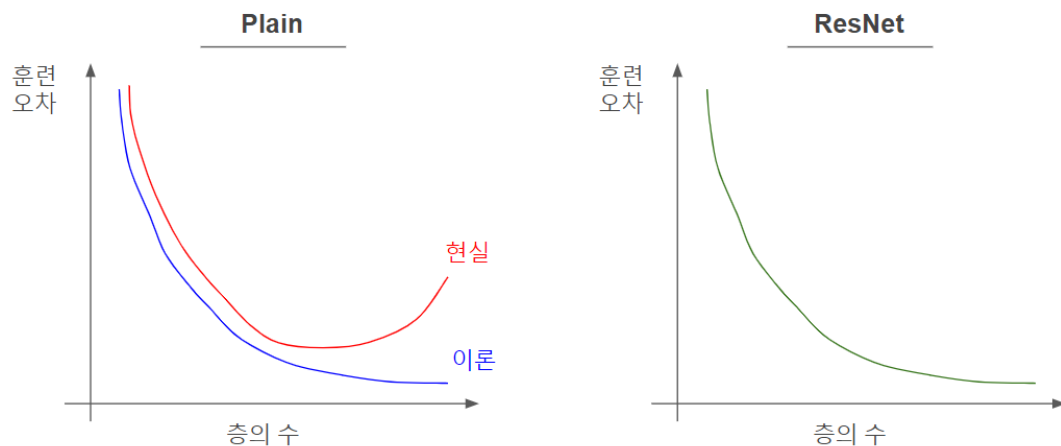
Residual block



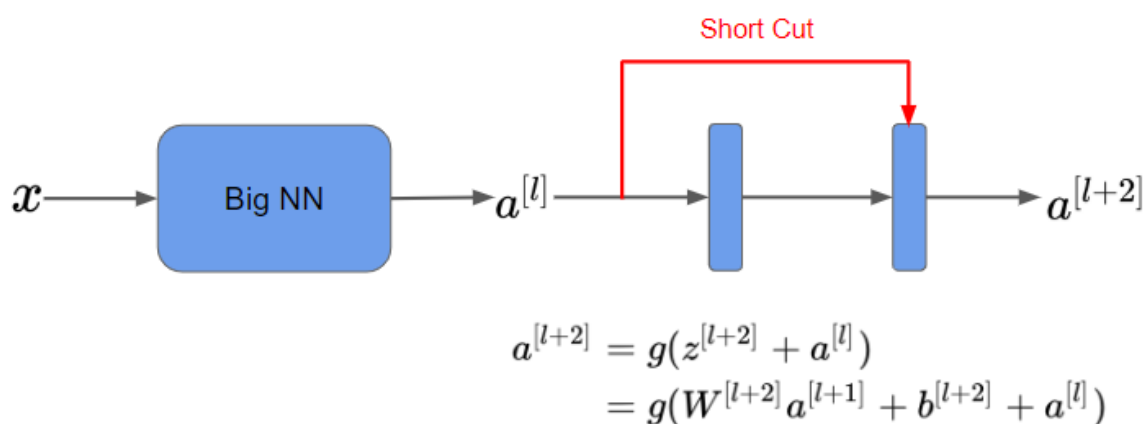
- Residual block을 사용하면 훨씬 더 깊은 신경망을 훈련시킬 수 있음

Residual Network

- 두개의 layer마다 하나의 residual block 형성
- 신경망이 깊어질수록 training loss가 커짐
- ResNet에서는 층이 깊어져도 traing loss가 계속 감소
 - vanishing/exploding gradient problem 해소



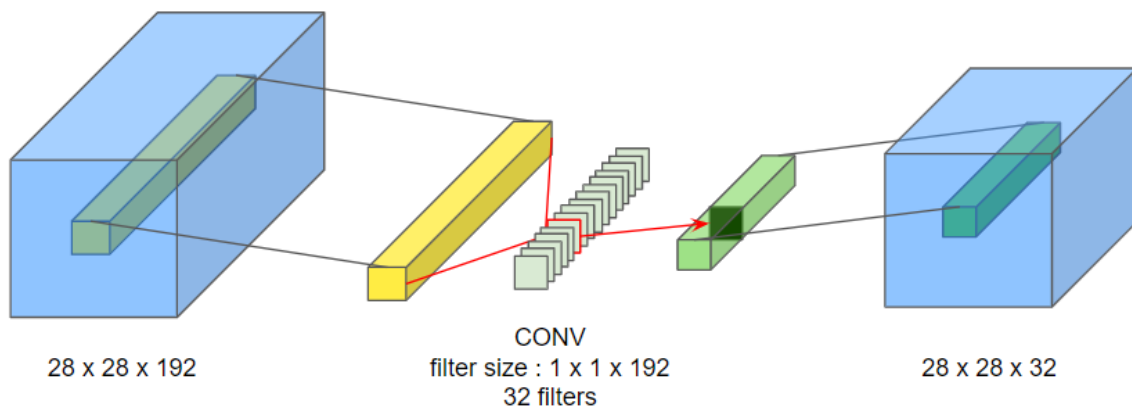
Why ResNets Work/h3



- $W^{[l+2]}$ 와 $b^{[l+2]}$ 의 값이 0이 되면 $a^{[l+2]} = g(a^{[l]}) = a^{[l]}$ 로 항등식이 됨

- 위와 같은 short cut을 통해 더 깊은 층에서도 높은 성능을 유지함
- 동일 합성곱 또는 행렬 W_s 를 이용해 $z^{[l+2]}$ 와 $a^{[l]}$ 의 차원을 같게 맞춰줌

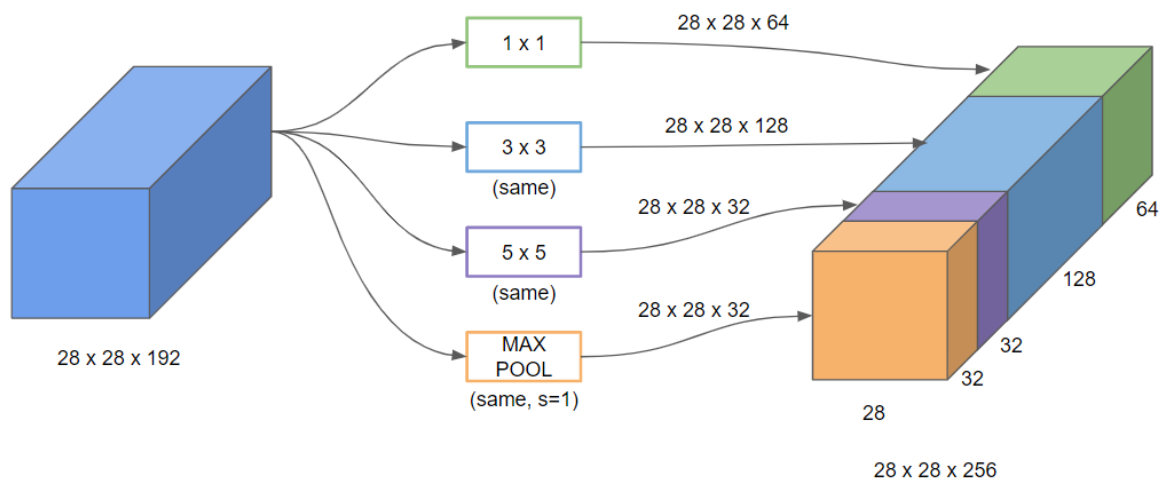
Network in Network



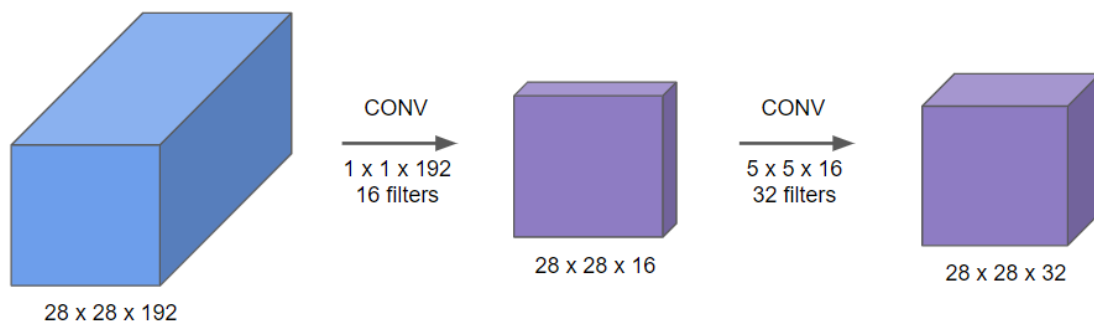
1:1 합성곱

- 1x1 kernel 을 이용한 합성곱
- 채널 수를 조절함
- 비선형성을 증가시켜 더 복잡한 함수를 학습할 수 있음
- 연산량을 줄임

Inception Network Motivation

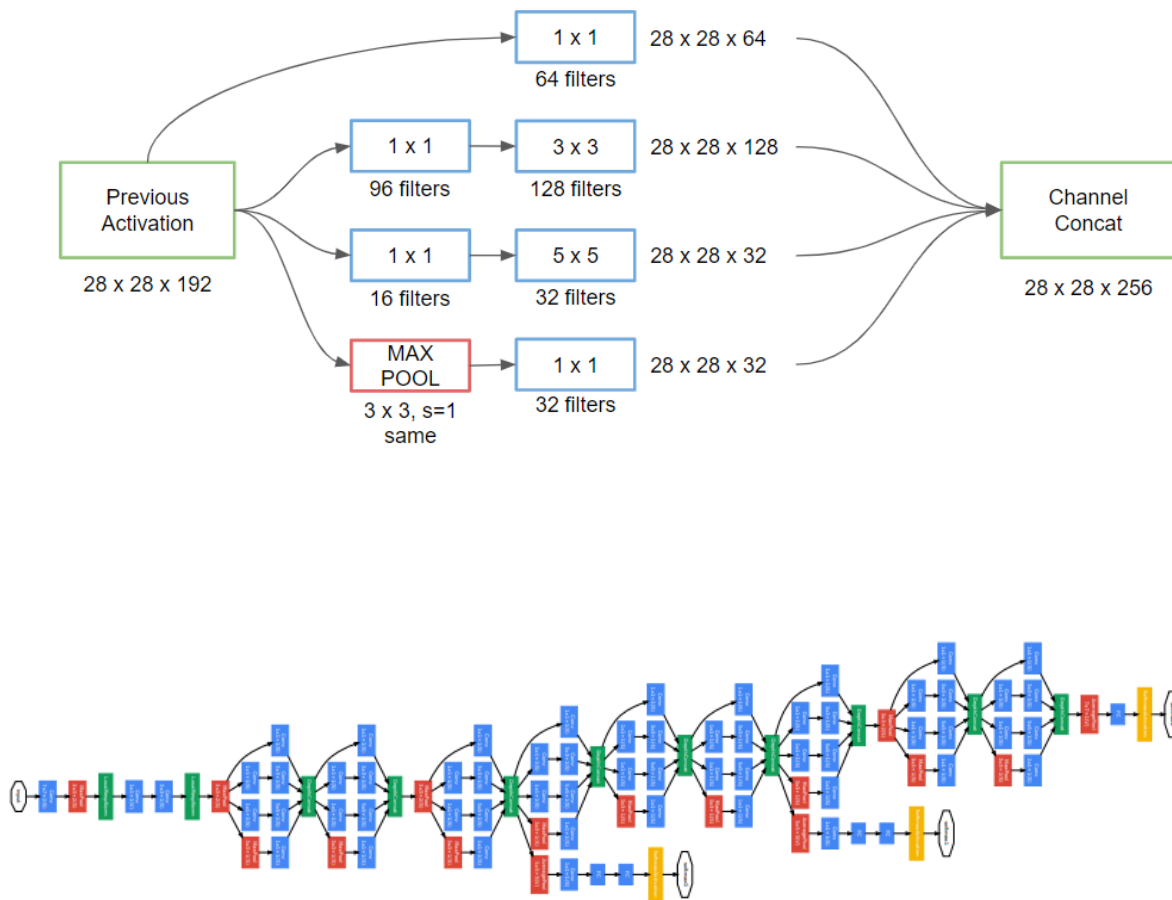


- 필터의 크기나 풀링을 하나로 정하는 대신 여러 개를 전부 적용한 뒤 각각의 output을 합침
- 1x1 convolution을 사용해 연산의 수를 줄임
 - 5x5 conv: $28 \times 28 \times 192 \times 5 \times 5 \times 32 = \text{약 } 1\text{억 } 2000\text{만개}$
 - 1x1 → 5x5: $28 \times 28 \times 192 \times 1 \times 1 \times 16 + 28 \times 28 \times 16 \times 5 \times 5 \times 32 = \text{약 } 1240\text{만개}$



Inception Network

Inception Module



- 1x1 합성곱을 이용해 계산 줄임
- Pooling layer는 그 이후에 1x1 conv layer 적용